Instituto Alberto Luiz Coimbra de Pós-Graduação e Pesquisa de Engenharia

COPPE UFRJ

EXPLORING PRODUCT LINE CONCEPTS IN VIDEO GAMES BUILDING

Diego Cardoso Borda Castro

Tese de Doutorado apresentada ao Programa de Pós-graduação em Engenharia de Sistemas e Computação, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Doutor em Engenharia de Sistemas e Computação.

Orientador: Cláudia Maria Lima Werner D.Sc.

Rio de Janeiro
Junho de 2025

EXPLORING PRODUCT LINE CONCEPTS IN VIDEO GAMES BUILDING

Diego Cardoso Borda Castro

TESE SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Orientador: Cláudia Maria Lima Werner D.Sc.

Aprovada por: Profa. Cláudia Maria Lima Werner
Prof. Geraldo Bonorino Xexéo
Prof. Esteban Walter Gonzalez Clua
Prof. Edson Alves de Oliveira Junior
Profa. Regina Maria Maciel Braga Villela

RIO DE JANEIRO, RJ – BRASIL
JUNHO DE 2025

Castro, Diego Cardoso Borda

Exploring product line concepts in video games building/Diego Cardoso Borda Castro. – Rio de Janeiro: UFRJ/COPPE, 2025.

XV, 154 p.: il.; 29,7cm.

Orientador: Cláudia Maria Lima Werner D.Sc.

Tese (doutorado) – UFRJ/COPPE/Programa de Engenharia de Sistemas e Computação, 2025.

Referências Bibliográficas: p. 98 – 109.

1. Software Reuse. 2. Software Product Line. 3. Serious Games. I. D.Sc., Cláudia Maria Lima Werner. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia de Sistemas e Computação. III. Título.

Agradecimentos

Agradeço primeiramente à minha mãe e tio, Rosane Cardoso e Ricardo Cardoso, pelo amor, apoio incondicional e por sempre acreditarem em mim. E, com todo carinho, agradeço também às minhas parceiras, minha filha Eevee Cardoso e minha noiva Bianca Albuquerque, pela parceria, paciência e por estar sempre ao meu lado, me apoiando e me motivando a seguir em frente.

À minha orientadora, Cláudia Werner, pela oportunidade e direcionamento que recebi no doutorado. Obrigado pelos conselhos, confiança depositada ao longo dos trabalhos e ensinamentos que guiaram essa pesquisa. Espero ter contribuído face à confiança em mim depositada.

Aos meus amigos, colegas, agradeço pelo suporte e pelas trocas de experiências que tornaram esta jornada mais enriquecedora.

Aos professores do PESC/COPPE que, com muita dedicação e competência, apoiaram meu desenvolvimento.

Aos amigos do grupo de Reutilização de Software e do Lab3D da COPPE/UFRJ, pela convivência e amizade.

À CAPES, pelo apoio financeiro durante o doutorado.

A todos aqueles que passaram em minha vida e contribuíram de alguma forma.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências (D.Sc.)

EXPLORANDO CONCEITOS DE LINHA DE PRODUTO NA CONSTRUÇÃO DE VIDEO GAMES

Diego Cardoso Borda Castro

Junho/2025

Orientador: Cláudia Maria Lima Werner D.Sc.

Programa: Engenharia de Sistemas e Computação

Alguns alunos, com tempo reduzido de aula, desinteressaram-se completamente pelo conteúdo que está sendo ministrado. Esse fato pode ser consequência do método de ensino atual, que é amplamente focado no professor e nos materiais apresentados em slides. Considerando isso, um expressivo número de pesquisadores está desenvolvendo novas abordagens pedagógicas que destacam métodos ativos, tais como aprendizagem orientada por tarefas, utilização de vídeos e jogos. Entretanto, a criação de um video game abrange diversas fases e requer a colaboração de profissionais com habilidades variadas, configurando-se como uma tarefa desafiadora. Os jogos sérios, em especial os educacionais, constituem uma divisão dos jogos que vão além de sua finalidade recreativa, com o objetivo de promover a assimilação de conhecimentos para os estudantes. Diante dos desafios relacionados ao desenvolvimento, a maior parte dos Jogos Educacionais Digitais (JEDs) é elaborada com base em jogos já existentes, o que pode ser compreendido como Reutilização de Software (RS). A Linha de Produtos de Software (LPS) representa uma das áreas de destaque da RS, consistindo em uma abordagem que visa agrupar sistemas que compartilham um determinado conjunto de características semelhantes, como no caso de uma coleção de jogos. Através de revisões e entrevistas com especialistas, foi levantado que seriam necessárias várias LPSs para JEDs, dependendo do contexto em que o mesmo está inserido, demonstrando a complexidade de construção dessa LPS. Diante disso, essa tese visa apresentar um modelo genérico de uma LPS para JEDs. A partir do modelo genérico, uma instância foi criada com o intuito de especializar o desenvolvimento de JEDs. Investigações por meio de questionários e de entrevistas foram realizadas para validar a instância, sendo demonstrada como uma solução promissora para o desenvolvimento de JEDs.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

EXPLORING PRODUCT LINE CONCEPTS IN VIDEO GAMES BUILDING

Diego Cardoso Borda Castro

June/2025

Advisor: Cláudia Maria Lima Werner D.Sc.

Department: Systems Engineering and Computer Science

Some students with limited class time have lost all interest in what is being taught. This may result from the current pedagogical approach, which predominantly emphasizes the instructor and the content displayed on slides. Many researchers are formulating innovative pedagogical strategies that prioritize active methodologies, including task-oriented learning and games. Creating a video game involves multiple phases and necessitates the collaboration of various individuals with diverse skills, rendering it a challenging task that can require significant time to complete. Serious games, particularly Educational Games (EGs), represent a category of games that extend beyond mere entertainment, aiming to facilitate acquiring knowledge for students. Due to developmental challenges, a large number of EGs are created by adapting or modifying existing games, a process known as Software Reuse (SR). Software Product Line is one of the primary areas of SR, and it is a technique that seeks to bring together systems that have a particular set of similar features, such as a series of similar games. Reviews and interviews revealed that many SPLs are required for the development of EGs, dependent upon the specific context of their application, and that a Generic MOdelo (MG) is requisite to facilitate the development of this SPL due to its complexity and variety. This research intends to introduce a MG of an SPL for EGs. An instance was generated from the MG to specialize the development of EGs. Investigations through questionnaires and interviews were carried out to validate the instance, which was demonstrated as a promising solution for the development of EGs.

Contents

List of Figures **x**

List of Tables **xii**

List of Abbreviations **xiv**

1 Introduction **1**

1.1 Motivation and Context 1

1.2 Objective 3

1.3 Methodology 3

1.4 Text organization 7

2 Theoretical foundation **9**

2.1 Educational Games 9

2.2 Software Reuse 12

2.3 Variability management 13

2.4 Engine VS Framework VS SPL 15

3 Literature Review **17**

3.1 How and what features are used in SPL for the development of EVGs? 20

3.2 Why and when should SPL be used in the development of EVGs? . . 26

3.3 Who and what current tools use SPL for the development of EVGs? . 27

3.4 How much effort/cost is involved in developing EVGs through SPL? 29

3.5 Final consideration 30

4 Proof of concept **31**

4.1 Dynamic Tetrad Game 31

4.2 Classic Tetrad Game 34

4.3 Evaluation 37

4.4 Planning 37

4.5 Participants sample 38

4.6 Procedure 39

4.7 Results . . . 39
4.7.1 Participants sample . . . 40
4.7.2 Analysis of results . . . 40
4.8 Final Consideration . . . 47

5 Survey **49**
5.1 Discussion . . . 55
5.2 Final Consideration . . . 61

6 Converting FM to GM **62**
6.1 Levels of abstraction . . . 62
6.2 GM Notation . . . 64
6.3 Notation Evaluation . . . 71

7 Generic Model Development **74**
7.1 SPLEG Generic Model . . . 74
7.2 Model Instance . . . 80
7.3 Instance Evaluation . . . 84
7.4 Final Consideration . . . 90

8 Conclusion **92**
8.1 Contribution . . . 93
8.2 Bibliographic Production . . . 93
8.3 Threats to validity . . . 95
8.4 Ethical issues . . . 96
8.5 Future Work . . . 97

References **98**

A Additional Information from the Literature Review **110**
A.1 Summary of the Research Protocol . . . 110
A.2 Papers analyzed in the Literature Review . . . 112

B TAM + MEEGA Questionnaire (English version) **115**
B.1 Job description . . . 115
B.2 Characterization questionnaire . . . 116
B.3 Evaluation questionnaire . . . 117

C TAM + MEEGA Questionnaire (Portuguese version) **120**
C.1 Descrição do trabalho . . . 120
C.2 Questionário de caracterização . . . 121
C.3 Questionário de avaliação . . . 122

D Survey **125**
D.1 Support text for the respondent 125
D.2 Characterization questionnaire 125
D.3 Specific questionnaires 127

E Survey (Portuguese version) **130**
E.1 Texto de apoio para o respondente 130
E.2 Questionário de caracterização 130
E.3 Questionários específico 132

F Generic Model Questionnaire (Portuguese version) **135**
F.1 Descrição geral 135
F.2 Elementos do modelo de características 136
F.3 Regras de Transformações 137
F.4 Atividades 141
F.5 Questionário de caracterização 141
F.6 Questionário de avaliação 143

G Student Questionnaire (Portuguese version) **145**
G.1 Descrição do trabalho 145
G.2 Questionário de caracterização 145

H Teacher Questionnaire (Portuguese version) **149**
H.1 Descrição do trabalho 149
H.2 Questionário de caracterização 149
H.3 Questionário de avaliação da proposta de ideia 151
H.4 Questionário de avaliação do jogo 152

List of Figures

1.1 Research methodology, adapted from (LACERDA *et al.*, 2013) 4

2.1 Development flow. 11
2.2 Feature Oriented Domain illustration (SCACCHI, 2011a,b). 13

3.1 SPL for EVGs. 22

4.1 Elemental tetrad Generation Game (CASTRO and WERNER, 2021). 32
4.2 Configuration panel (CASTRO and WERNER, 2021). 32
4.3 Elemental tetrad Feature tree (CASTRO and WERNER, 2021). . . . 33
4.4 Levels generated by the game. 35
4.5 Game feature selection trees, grouped according to the elemental tetrad. 35
4.6 Game characteristics selection trees. 35
4.7 Tetrad SPL Classic game feature tree. 36
4.8 Characterization of specialists. 41
4.9 Experience of specialists. 42
4.10 Characterization of students. 43
4.11 Experience of students. 44
4.12 Experience of community. 44
4.13 Characterization of community. 45
4.14 Meega / TAM Questionnaire with Specialists Response. 46
4.15 Meega / TAM Questionnaire with Students Response. 47
4.16 Meega / TAM Questionnaire with Community Response. 48

5.1 Demographic data. 50
5.2 Participant's Experience. 52
5.3 Questionnaire for the second part of the survey 53

6.1 Abstraction levels for representing an SPL, adapted from (MAIA, 2024) and (DE OLIVEIRA, 2006) 63
6.2 Participants' experiences. 72
6.3 TAM Questionnaire Answers. 73

7.1 Levels of abstraction for SPL development. FM defined. 75
7.2 GM after step 1. 75
7.3 GM after step 2. 76
7.4 GM after step 3. 77
7.5 GM after step 4. 78
7.6 GM final version. 78
7.7 Feature Model, GM, and Model. 79
7.8 Floor Question Filling Form. 81
7.9 Game environment. 82
7.10 Locked and released states of the elevator. 82
7.11 Examples of questions to be answered and feedback for the student. . 82
7.12 A* algorithm touchpoints. 83
7.13 Dialogue with the NPC to describe the floor. 84
7.14 Characterization of group A. 85
7.15 Characterization of group B. 86
7.16 Group A Usability Assessment (MEEGA questionnaire). 87
7.17 Group A Experience Assessment (MEEGA questionnaire). 87
7.18 Perception of ease of use by group B (TAM questionnaire). 88
7.19 Perception of usefulness of use by group B (TAM questionnaire). . . . 88
7.20 Group B Usability Assessment (MEEGA questionnaire). 89
7.21 Group A Experience Assessment (MEEGA questionnaire). 89
7.22 FM transformation stage until game development. 91

A.1 Summary of the research protocol based on the model created by CALDERÓN *et al.* 110
A.2 Search flow of research on games and SPL approach. 112

List of Tables

2.1 Description of FODA notation elements (KANG *et al.*, 1990). 14
2.2 Engine VS Framework VS SPL 16

3.1 Characteristics of EGs 23

4.1 Changeable game features; 32
4.2 Game mechanics, dynamics and aesthetics. 33

6.1 Description of notation elements, adapted from MAIA (2024) and DE OLIVEIRA (2006). 64
6.2 [Rule 1] Transformation of characteristics into classes or methods. . . 66
6.3 [Rule 2] Transforming simple relationships into associations. 67
6.4 [Rule 3] Transforming alternative VP into specialization/specialization. 68
6.5 [Rule 4] Transforming optional VP into a composition relationship. . 69
6.6 [Rule 5] Transforming mandatory elements into mandatory stereotypes. 70
6.7 [Rule 6] Transforming an inclusion relationship into a dependency relationship. 70
6.8 [Rule 7] Transforming an exclusion relationship into an association with OCL. 71

8.1 Papers published as first author. 94
8.2 Papers published as co-authorship. 95

A.1 Search String. 111
A.2 Papers selected for the study. 112

F.2 [Regra 1] Transformação de características em classes ou métodos. . . 137
F.3 [Regra 2] Transformando relacionamentos simples em associações. . . 138
F.4 [Regra 3] Transformando pontos de variação alternativo em especialização/especialização. 138
F.5 [Regra 4] Transformando pontos de variação opcionais em um relacionamento de composição. 139
F.6 [Regra 5] Transformando elementos obrigatórios em estereótipos. . . . 140

F.7 [Regra 6] Transformando uma relação de inclusão em uma relação de dependência. 140
F.8 [Regra 7] Transformando uma relação de exclusão em uma associação com OCL. 140

List of Abbreviations

AL	ACTIVE LEARNING	vi
CBD	COMPONENT BASED DEVELOPMENT	vi
CIM	COMPUTATIONAL INDEPENDENT MODEL	vi
EA	EDUCATIONAL ACTIVITIES	vi
EG	EDUCATIONAL GAME	vi
ET	ELEMENTAL TETRAD	vi
EVG	EDUCATIONAL VIDEO GAME	vi
FM	FEATURE MODEL	vi
GAWP	GAMES WITH A PURPOSE	vi
GBL	GAME-BASED LEARNING	vi
GM	GENERIC MODEL	vi
MDD	MODEL DRIVEN DEVELOPMENT	vi
MR	MULTIVOCAL REVIEW	vi
OMG	OBJECT MANAGEMENT GROUP	vi
PIM	PLATFORM INDEPENDENT MODEL	vi
PSM	PLATFORM SPECIFIC MODEL	vi
SE	SOFTWARE ENGINEERING	vi
SLR	SYSTEMATIC LITERATURE REVIEW	vi
SPL	SOFTWARE PRODUCT LINE	vi

SPLEG SOFTWARE PRODUCT LINE FOR EDUCATIONAL VIDEO GAMES . . . vi

SR SOFTWARE REUSE . . . vi

TAM TECHNOLOGY ACCEPTANCE MODEL . . . vi

VP VARIATION POINT . . . vi

Chapter 1

Introduction

This chapter presents the context, motivation, and problems this thesis addresses. Additionally, this chapter presents the objectives, the methodology employed to achieve them, and the structure of the text.

1.1 Motivation and Context

Educators are increasingly exploring creative learning methodologies that combine enjoyment with education to enhance student engagement and motivation. Active Learning (AL) methodologies (BONWELL and EISON, 1991), such as Project-Based Learning (LARMER *et al.*, 2015) and Flipped Classroom (BERGMANN and SAMS, 2012), empower students and give them a central role in their learning journey, with reports that indicate good results (FREEMAN *et al.*, 2014).

Among AL methodologies, Game-Based Learning (GBL) (PIVEC *et al.*, 2003), especially with video games (VGs) (GEE, 2003; PRENSKY, 2001), plays a prominent role. Games in education have been found to improve knowledge and skill acquisition, perceptual and cognitive skills, and promote empathy, among other benefits (BOYLE *et al.*, 2016). According to Kalmpourtzis (KALMPOURTZIS, 2019), "Games have a fantastic ability to change the presentation and delivery of problems to players, making them invisible to the eyes of players while they are still engaged in the game context". Given the advantages, it is to be expected that games will become an increasingly important force in education.

Despite the benefits of Educational Games (EGs), their design is complex, resource-intensive and requires multiple sets of interdisciplinary skills (BELLOTTI *et al.*, 2013; GONZÁLEZ GARCÍA *et al.*, 2019; JENKINS *et al.*, 2004). Furthermore, if digital, significant technical expertise and resources are also needed to support their development. Due to these obstacles, EG designers frequently concentrate on adapting (modifying) existing games rather than creating new ones (ABBOTT, 2018). Similarly to the concept of opportunistic Software Reuse (SR) (NCUBE *et al.*, 2008), this method involves modifying something already created and altering it for a different function. This strategy reduces the experience required for game development, as the ability to change is based on recognizing/adapting game mechanics rather than designing/creating them. Adaptations are inevitable, but this strategy, called modding in the game world, drastically reduces the experience and resources required to develop an EG (ABBOTT, 2018). This demonstrates that the gaming community is already using SR approaches but in an ad-hoc manner.

SR is a subdiscipline of Software Engineering (SE) that seeks to create new products from existing ones. There are several areas of study within this field of study, including Componentization, Model Driven Development (MDD), and Software Product Line (SPL). Research and development of initial prototypes indicated that the SPL technique is the most appropriate for developing EGs, primarily attributable to its capacity to generate multiple products from one origin. SPL can be thought of as a collection of software that shares similar characteristics and can be modified from the same base by adding and eliminating characteristics at "variation points" (KRUEGER, 1992), with the primary objective of increasing productivity and reducing production time (MEFTAH *et al.*, 2019). From the analysis of games from large companies, it is possible to identify the SPL patterns that were used in the development of these games (GARCIA *et al.*, 2022); however, research carried out in this study shows that this technique is not frequently used in the creation of EGs (FURTADO *et al.*, 2011).

As previously mentioned, the process of creating EGs, from conception to execution, can be pretty complex. Scratch, eAdventure, Alice, Roblox Studio, and a few engines are tools that seek to facilitate this process and have already been used successfully. However, even with these tools, creating EGs takes significant time and can be too much for a single teacher to manage (GONZÁLEZ GARCÍA *et al.*, 2019). To increase the productivity of EGs, especially for teachers with little or no knowledge of educational VGs, the research question can be formulated as follows: **How can the use of SPL be specialized for the development of EG?** This thesis focuses on VGs; so, when reading EGs, recognize that VGs are referenced.

To design a SPL, it is essential to first determine all the characteristics that will constitute the line, thereby organizing the variety that will be offered (MAAZOUN *et al.*, 2014). These attributes are typically represented by Feature Models (FMs), structured in trees that delineate all the characteristics of the product to be developed, emphasizing its specifics through branches referred to as variation points (VP) (MAAZOUN *et al.*, 2014; SILVA, 2016).

FMs simply display the characteristics of the software to be constructed, lacking sufficient details for its development (CAVALCANTI *et al.*, 2011). Consequently, class diagrams are constructed to represent this information. The level of abstraction in a FM significantly differs from that in a class diagram, necessitating an intermediate step to facilitate an easier transition. A prospective approach that can function as a link between these two representations is a Generic Model (GM), which can be defined as a high-level abstraction from which models are generated. They can be represented using UML, displayed as a class diagram that illustrates entities, attributes, and relationships, albeit at an abstract rather than a concrete level(CAVALCANTI *et al.*, 2011; CHEN *et al.*, 2009). In view of this, this thesis also aims to present an approach to SPL development through the construction of GM.

Finally, it is important to emphasize that the current study assumes an impartial position with respect to specific teaching methodologies, ideologies, or models, and is not associated with a singular approach.

1.2 Objective

As previously stated, creating an EG is a complex process. The primary aim of this thesis is to create a generic SPL that facilitates the development of EGs for any discipline or subject to be taught. Furthermore, an approach for converting FM to GM is presented, with the aim of facilitating the SPL development process. These general objectives can be decomposed into specific goals:

- Characterize the state-of-the-art in the use of software reuse approaches for VG development, highlighting the adoption of SPL strategies;
- Facilitating the SPL development process through a FM to GM conversion approach;
- Enable EGs development for educators lacking prior training in the field.
- Implement an SPL for Educational VGs (SPLEG);
- Conduct an evaluation study to validate the ease and agility of developing a VG through the SPL created.

1.3 Methodology

The technique used in this research was inspired by the Design Science Research (DSR) model (LACERDA *et al.*, 2013), but was adjusted according to this research need. Each stage contributed to and strengthened the path to reach the research objectives.

Figure 1.1 presents the methodology used, which may be interpreted in two ways: each line details how each activity was designed, while each column details all activities performed (first column), their associated artifacts (second column), and their associated objectives (third column). Orange and green colors represent exploratory and conclusive efforts, respectively. Exploratory activities may be defined as those that involve the validation or analysis of early data. Conclusive activities seek to explore the overarching issue deeper, which is to investigate reuse strategies for improving the game development process. Each activity in Figure 1.1 is described below.

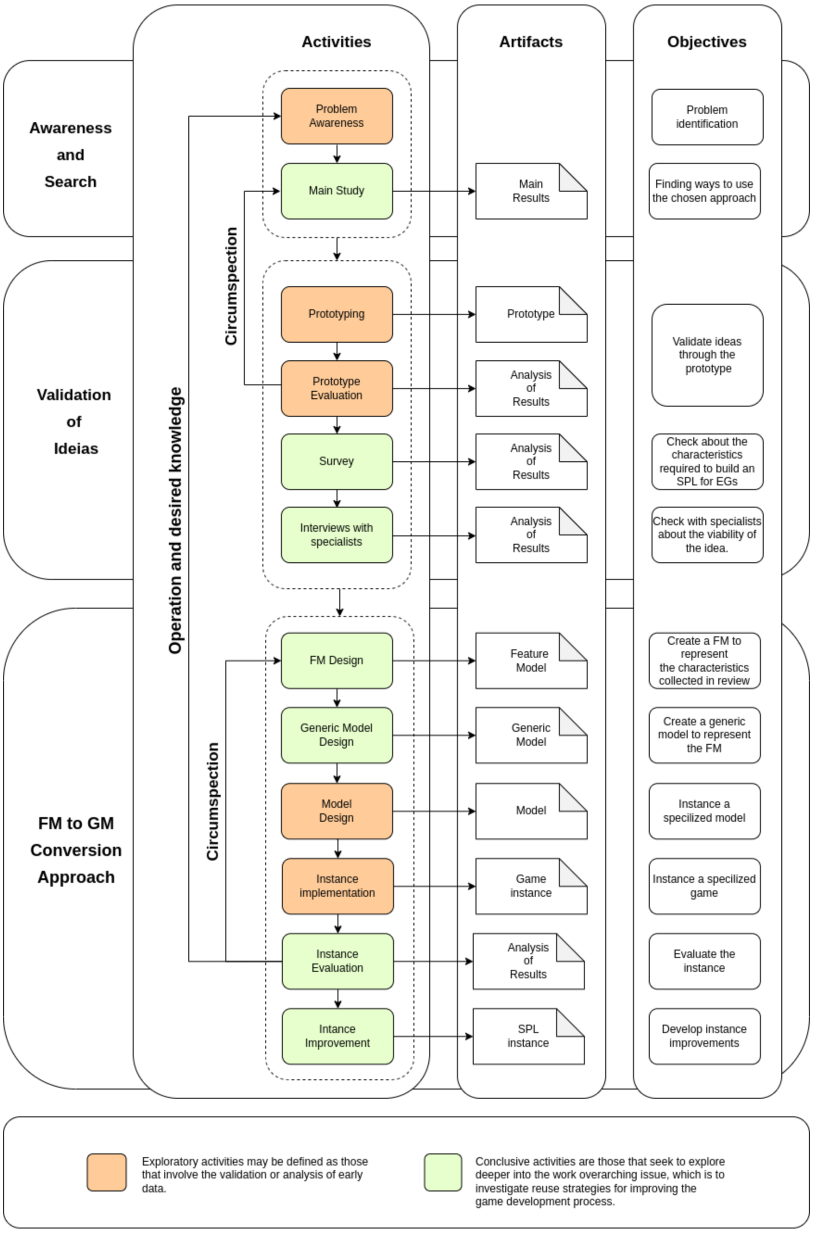


Figure 1.1: Research methodology, adapted from (LACERDA *et al.*, 2013)

- **Awareness and Search:** Seeks to identify the problem to be solved. Three revisions were performed and will be described below.

- **Problem Awareness:** Elaborate an initial study to have first contact with the researched area. Seeking to understand how Educational Video Games (EVGs) are being created currently. Ad hoc revision.
  - **Objective:** Identify the problem to be solved with the thesis. Identify what problems exist in creating EVGs through SR.
- **Literature review:** Develop a study to find out what has already been produced on game development through the SPL.
  - **Artifact:** Analysis of the results of the review.
  - **Objectives:** Define state-of-the-art on using the SPL for game development.

- **Validation of ideas:** Activities to generate a practical basis.
  - **Prototyping:** Develop one (or more) game(s) based on previou studie to validate SPL concepts to game development.
    - **Artifact:** Games that were developed.
    - **Objectives:** Validate some SPL concepts to the game development process.
  - **Prototype Evaluation:** Conduct a viability study to evaluate the game created in the previous phase.
    - **Artifact:** Game evaluation.
    - **Objectives:** Evaluate the games from a development perspective. Evaluate whether the developed games indicate ease of construction and whether such a method may aid game development.
  - **Survey:** Check with specialists, students, and enthusiasts about the characteristics required to build an SPL for EGs.
    - **Artifact:** Analysis of survey results.
    - **Objectives:** Determine the hypotheses of the characteristics necessary to build an SPL for EGs.
  - **Interviews with Specialists:** Check with specialists about the validity/viability of the solution of SPL.
    - **Artifact:** Analysis of interview results.
    - **Objectives:** Determine from the opinions of the specialists whether the approach development is valid and viable.

- **FM to GM Conversion Approach:** Based on the information collected in previous activities, this thesis aims to present an SPLEG and a GM to manage the characteristics variability of this SPL.
  - **FM Design:** Based on the reviews, an SPL was created with the aim of specializing the development of EVGs.
    - **Artifact:** FM created to represent the SPL.
    - **Objectives:** Create an SPL to specialize the development of EGs.
  - **Generic Model Design:** Based on the reviews, survey, and interviews, it was discovered that more than one SPL would be necessary for EVGs, highlighting the complex nature of the SPL. Therefore, a GM solution was created to manage the variabilities. The objective of this activity was the development of this GM.
    - **Artifact:** GM created.
    - **Objectives:** Use all collected information to create the GM.
  - **Model Design:** Instantiate a model from the GM that was created in the previous activity. The purpose was to create a specialized SPLEG.
    - **Artifact:** The SPL model that was instantiated.
    - **Objectives:** Instantiate the SPL model to solve the thesis research question.
  - **Instance implementation:** Implements the model that was created in the previous activity.
    - **Artifact:** A specialized game that was instantiated.
    - **Objectives:** Instantiate a game from the model created.
  - **Instance Evaluation:** Conduct a viability study to evaluate the SPL created in the previous phase.
    - **Artifact:** Analysis of the evaluation results.
    - **Objectives:** To evaluate the developed SPL to see if it helps to develop EVGs faster and easier compared to traditional methods.
  - **Instance Improvement:** Improve the SPL created.
    - **Artifact:** SPL with identified improvements.
    - **Objectives:** Develop the improvements found in the SPL evaluation.

1.4 Text organization

This work is organized into eight chapters. This chapter presents the research's context, methodology, motivation, and problem.

To enhance comprehension of the chapters, a theoretical foundation on important topics is provided. Themes like GMs, educational games, software reuse, and product lines will be discussed and illustrated in **Chapter 2**.

A study was conducted to enhance the comprehension of games generated via SPLs, identifying the advantages and essential characteristics required for the formulation of a specialized SPLEG. More details about the review can be read in **Chapter 3**.

To validate the information found in the review, prototypes were built to validate the idea of using SPL concepts to develop games. More information regarding the prototypes' development and evaluation can be found in **Chapter 4**.

As previously stated, a set of characteristics essential for the development of SPL was selected from the reviews. A survey was conducted with specialists, students, and enthusiasts in the field of gaming in order to verify these features. Subsequent to the survey, hypotheses were formulated and validated by interviews with professionals with over a decade of experience in EGs. **Chapter 5** contains further information regarding the survey and the interviews.

An SPL is usually represented by a FM, which illustrates only the high-level characteristics that the software must have. During the investigation, it was noted that additional information is essential for the improvement of the SPL, necessitating a reduction in the abstraction level of the FM. Furthermore, research has demonstrated that numerous SPLs are necessary for the development of EGs, dependent upon the specific context of the game application, emphasizing the complex nature of the SPL. Confronted with intricate SPLs, some projects advise for the utilization of GMs for managing this huge diversity of characteristics. Consequently, a conversion approach from FM to GM was developed to address the diversity within an SPLEG. Additional details regarding the GM are available in **Chapter 6**.

From the GM, an instance was generated to provide a specialized SPLEG. This instance was proposed and developed, and it was demonstrated through a simple and user-friendly interface platform. Its objective is to enable educators lacking game design expertise to create various EGs across several subjects by merely inputting the information. It is important to note that all characteristics utilized in the SPL were collected through literature research, questionnaires, and expert interviews, combining theory with best practices. Finally, an evaluation was executed involving master's degree students in games at the Federal University of Rio de Janeiro (UFRJ) and experts in EGs with over a decade of expertise in the field.

The evaluations indicated that, despite the complexity of the concept, the SPLEG platform demonstrated ease of use and potential in developing EGs. More details about the instanced model can be seen in **Chapter 7**.

The final remarks of this thesis are presented, demonstrating its limitations and future work, in **Chapter 8**.

Chapter 2

Theoretical foundation

This chapter aims to provide the theoretical foundation essential to comprehend the research that may be unfamiliar to the reader. In this sense, some basic concepts about educational games (EG), Software Reuse (SR) and Generic Model (GM) are described in the following.

2.1 Educational Games

The academic literature presents numerous definitions of play and games (SALEN and ZIMMERMAN, 2004; XEXÉO *et al.*, 2017).

Huizinga provides a classical definition of *play*, that is frequently cited as referring to games as a free activity that is very deliberately detached from conventional life and is not serious while simultaneously absorbing the player wholly and intensely (HUIZINGA, 1999). It is an activity that carries no material interest and yields no profits. It operates within its designated parameters of time and space, adhering to established rules in an organized manner. It promotes the formation of social groupings, which tend to surround themselves with secrecy and to stress their difference from the everyday world by disguise or other means (HUIZINGA, 1999).

On the other hand, definitions focused on game design often characterize games as systems with uncertain outcomes (COSTIKYAN, 2013), emphasizing formalism through states, constraints, and goals, and underscoring players' decision-making related to resource management (SALEN and ZIMMERMAN, 2004).

Although some definitions echo Huizinga's stance that games remain insulated from the real world within a "magic circle," contemporary perspectives like Juul's (JUUL, 2005) argue that games produce negotiable consequences in real life. Consequently, modern discourse posits that the "magic circle," if it exists, is permeable (BROWN, 2015). This premise is essential to support the application of games in education.

Games go beyond entertainment, passing messages and values (FLANAGAN and NISSENBAUM, 2014). This can be a side effect of some design choices in an entertainment game, such as adolescents learning Greek mythology while playing "God of War". When the main objective of game design is not entertainment but teaching, persuasion, or other objectives, they are termed serious games (ADAMS and DORMANS, 2012) or Games With a Purpose (GAWP).

Many serious games, including educational and training games, are used in Game-Based Learning (GBL) methods. These approaches explore game features like simulations, safety from undesirable consequences, immediate feedback, error-based learning, etc, to improve educational results and enhance the learning experience. Digital Game-Based Learning (DGBL) narrows this down to digital games (BAU, 2015; PIVEC *et al.*, 2003).

There are several ways to organize or define a game, and one well-known formal approach for this purpose is the Elemental Tetrad (SCHELL, 2008), which aims to divide the properties of games into four categories, which are (SCHELL, 2008):

- **Mechanics:** can be interpreted as the rules and activities that may occur during the game.
- **Second level mechanics:** Although the second level mechanics are not part of the elemental Tetrad, they will be added in this thesis to describe mechanics generated from the combination of primary mechanics. This addition was made to bring more dynamism to the proposed change of the games to be built. It is worth remembering that the game designer only has direct control over the mechanics and sprites. However, the mechanics are all the parts necessary to play a game (HUNICKE *et al.*, 2004).
- **Story:** describe the narrative aspect of the game.
- **Aesthetics:** It is the most apparent part to the player, being everything that the five senses can capture. It is represented by the game's appearance, sprites, sounds, etc.
- **Technology:** refers to the tools and the systems used to implement the gameplay.

.

From an educational point of view, it is possible to argue that how a user learns is fundamentally more essential than the environment through which the learning occurs (ARNAB *et al.*, 2015). However, this step of how content will be taught is significantly influenced by the type of the mechanics used in its development (BOLLER and KAPP, 2017a; DE ARAUJO and DA SILVA ARANHA, 2013). Games can have various mechanics and features. The complexity of a game can increase in proportion to the number of elements. As a result, methods for arranging game concepts are required. There are several ways to organize ideas for creating a game, including flow models (DORMANS, 2011), gameplay (GUARDIOLA, 2016), and features (XEXÉO *et al.*, 2013).

Along with methods for organizing game concepts, various game creation cycles are described in the literature; nonetheless, they may be simplified into four primary stages (RAMADAN and WIDYANI, 2013). Each of them is highlighted as follows:

- **Pre-production:** aims to define and improve the game's original concepts; this is the stage during which documentation, such as the Game Design Document (GDD), concept art, and game design, is created. Specific procedures begin with a phase called Pitch, which is responsible for the game's idea and basic design.

- **Production:** Some processes divide this stage into development and refinement. However, it may be viewed as a continuous cycle centered on creating assets and source code. Finally, the entire technique is validated internally.

- **Testing:** intends to evaluate the game's usability, playability, and balance characteristics. This process is typically followed by two releases (alpha and beta), depending on the game's integrity.

- **Release:** final stage of production of the game, it is ready to be released to the public. The release process comprises product launch, project documentation, game maintenance, and expansion planning.

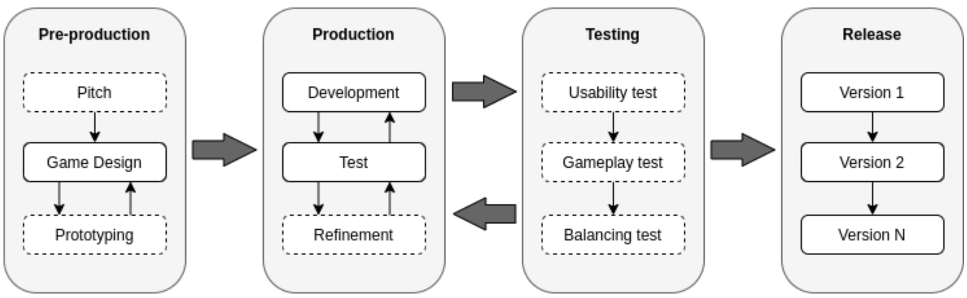


Figure 2.1: Development flow.

Figure 2.1 is a simplified flow chart illustrating the many stages that might occur during game development. The dotted steps are optional and may be skipped.

As a result of the procedure outlined above, it is reasonable to deduce that developing games may be pretty tricky and time-consuming. As a result, Educational game (EG) designers often adapt existing games instead of creating them from scratch (ABBOTT, 2018). This adaptation in the gaming community is termed **mod** (MCARTHUR and TEATHER, 2015; UNGER, 2012).

2.2 Software Reuse

Multiple industries, such as manufacturing, automotive, and electronics, utilize reuse effectively. SR is defined as the process of developing systems from one or more existing ones rather than starting from scratch; that is, it is the process of utilizing existing software artifacts and knowledge to create something new (KRUEGER, 1992). Based on the SWEBOK book, several applicable areas can be identified in SR, including Construction for and by Reuse, Component-Based Design, Model-Driven Architecture and (BOURQUE and FAIRLEY, 2014), the latter being the main focus of this thesis.

Software Product Line (SPL) refers to a collection of strategies, techniques, and tools for the methodical development of comparable systems with a common core but exhibiting distinct characteristics. Utilizing these subjects is predicted to decrease development time, more straightforward maintenance and evolution of systems, enhance programmer satisfaction, and improve the quality of code (KRUEGER, 1992).

SPL divides its approach into two stages: domain engineering, which involves the creation of shared assets, and application engineering, which consists of the reuse of common elements and the addition of unique elements. SPL is distinct from other forms of reuse in that it contrasts predictive and opportunistic methods. Instead of storing generic software components in a library in the expectation of reusing them, SPL requires the development of software artifacts (i.e., assets) only when their reuse in one or more products is anticipated (KRUEGER, 1992).

Every SPL is categorized according to the similarities and differences of the products on the line. It begins with a resource called root, which provides the starting point and is the only node that does not have a top node. The nodes are then branched by **required** that are present in all applications and constitute the heart of SLP (demonstrated by filled balls); **alternative** that are restrictive characteristics; an application may have one or more of these features (indicated by unfilled circles **optional** that are eatures that specific applications may or may not have (demonstrated by unfilled balls); and **exclusions** that when only one of these characteristics can be used (shown by filled circles). These branches are also known as variation points, an essential characteristic that makes SPL a large-scale development standard (MEFTAH *et al.*, 2019).

In examining the variation points of an SPL, one can determine the presence of variants and invariants. The first can be characterized as the implementation alter-

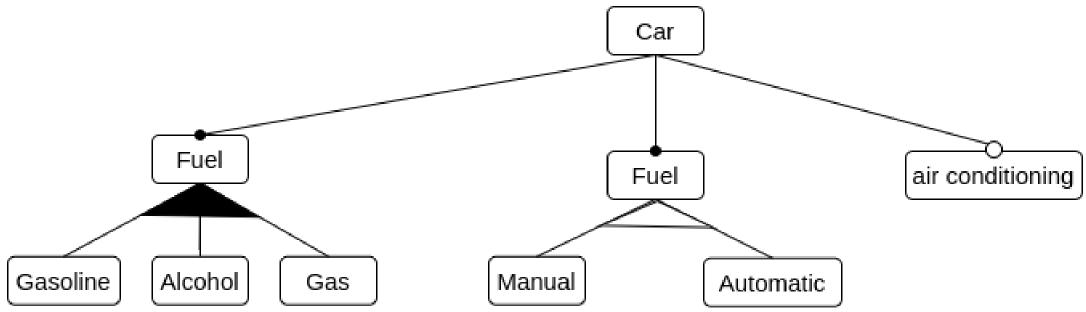


Figure 2.2: Feature Oriented Domain illustration (SCACCHI, 2011a,b).

natives offered for a specific point. In other words, they are components inherently associated with a variation point, serving as alternatives to configure that variation point. The invariant refers to fixed elements that remain unconfigured within the application area (DE OLIVEIRA, 2006). Figure 2.2 demonstrates an example of these elements based on Feature-Oriented Domain Analysis FODA (KANG, 2010). The example illustrates that a car can utilize multiple fuel types, possess either automatic or manual transmission, and may or may not be equipped with air conditioning.

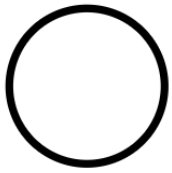
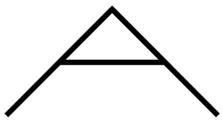
2.3 Variability management

Variability refers to the capacity of a system or software product to be altered, adjusted, or configured for a different context (BOSCH, 2004). Modeling can be used as a support mechanism to define and express the variability inherent in an SPL in a systematic and traceable manner, along with the mappings between the components constituting an SPL. Numerous SPL projects are conceived and sustained by model-based methodologies (CAVALCANTI *et al.*, 2011).

Typically, SPLs define their requirements through a Feature Model (FM). FODA is one of the most used notations to represent an FM, as previously illustrated. FM includes all potential features that the line can produce, with each product originating from an FM represented by a Feature Configuration (FC) (SILVA, 2016), indicating a possible instance of attributes that form a new product.

In addition to the FODA notation, there are several other notations, including the Feature Oriented Reuse Method (FORM) (KANG *et al.*, 1998), Reuse-Driven Software Engineering Business (RSEB) (POUR, 2000), and proposals by VAN GURP *et al.*, Riebisch (RIEBISCH *et al.*, 2002), Cechticky (CECHTICKY *et al.*, 2004), and Czarnecki (CZARNECKI *et al.*, 2004), among others. However, each of these notations possesses deficiencies, including a lack of standardization, a vague connection between characteristics, and inadequate expressiveness to represent interactions among variants. Finally, these models only describe the software at a high level, lacking development-level information such as composition, aggregation, specialization, interfaces, and association. Table 2.1 demonstrates each of the elements of the FODA notation.

Table 2.1: Description of FODA notation elements (KANG *et al.*, 1990).

Element	**Description**
. 	Mandatory characteristic, indicates that a feature must exist.
. 	Optional characteristic, indicates that a feature may or may not exist.
. 	Alternative relationship, indicates a relationship between a variation point and its variations, having cardinality equal to 1.
. 	Optional relationship, indicates a relationship between a variation point and its variations, having cardinality equal to 0 or N.
	Simple relationship, indicates that one characteristic is related to another.
	Exclusion relationship, indicates that if one of the characteristics exists, the other related one cannot exist.
	Inclusion relationship, indicates that the inclusive characteristic can only exist if the other included characteristic exists.

GMs serve as a high-level abstraction from which models are derived, but with a focus on development. They can be represented using UML (Unified Modeling Language) and are intended to illustrate entities, properties, and associations (CAVALCANTI *et al.*, 2011). Upon the instantiation of a GM, other models are generated that are quite similar to the feature selections made in an SPL, demonstrating how the selected features will be implemented in the software to be created. In this manner, they can be interpreted as a method of demonstrating the variability development of an SPL. They are designed to be instantiated in a variety of contexts, similar to SPLs; however, with a lower level of abstraction than FM. Their structure must be able to be extended or reduced as necessary, and they must provide options for choice, allow for various levels of detail, and implement reusable components (CÉRET *et al.*, 2013; MAIA, 2024).

2.4 Engine VS Framework VS SPL

As previously mentioned, game development is a multifaceted activity that might need considerable time. In light of this, various tools have been created to enhance development, including engines, frameworks, and the concept of SR, particularly focusing on SPL. Due to a variety of terms, it is essential to distinguish between each one and explain their distinctions.

An engine can be considered a game development platform that offers an environment for game creation. The tool is designed to simplify the game development process by integrating several game components, such as animations, interactions, and collision detection, into one cohesive platform (BARCZAK and WOŹNIAK, 2019). It incorporates the previously introduced concept of components to offer reusable software elements applicable in several games or scenes. However, it does not provide an initial structure for the project, making it necessary to use some structure created by the developer or even use a framework (POLITOWSKI *et al.*, 2021).

A framework may be theoretical or technical, with the first one defined as a conceptual structure that supports the investigation and comprehension of a subject within a research effort. In other words, it can be defined as a collection of questions and suggestions that support a research. A technical framework is a reusable collection of software libraries, classes, components, or structures (GARVEY and JONES, 2021). A framework offers a practical option for fundamental structures, enabling developers to concentrate on more intricate tasks by providing a preset architecture and reusable components that simplify and improve programmers' work (JOHNSON, 1997). A framework, in contrast to engines, does not include a development platform and just offers an initial structure for the project with a collection of components.

An SPL, as previously noted, is a collection of strategies, techniques, and methodologies for the systematic development of similar systems that share a common core but have distinctive characteristics. In contrast to frameworks, it does not provide an initial structure for a project or offer components and libraries for utilization; the development tools employed are abstract to SPL and are not their responsibility. It does not offer a development environment or reusable components, in contrast to engines. The responsibility of SPL is to distinguish the shared characteristics of the project from those that differentiate it, offering an abstraction that allows for the visualization and future development of the software's features (KRUEGER, 1992). In contrast to frameworks, it does not provide an initial structure for a project or offer components and libraries for utilization; the development tools employed are abstract to SPL and are not their responsibility. Table 2.3 shows the comparison between each of the terms.

Table 2.2: Engine VS Framework VS SPL

	Focus	**Definition**	**Advantages**
Engine	Development	Platform that offers an environment for game development.	Simplify the game development process by integrating several components.
Theoretical Framework	Research	Collection of questions and suggestions that support a research	Offers a structure of clearly defined steps to facilitate the systematization of research.
Technical Framework	Development	Initial structure for a project with a series of reusable components.	Reduces the effort of building a system by providing an initial structure to use.
SPL	Development	Collection of software-based systems or products that share a common core.	It enables the enhancement of comprehension and control over the common and variable functionalities of a system.

Chapter 3

Literature Review

This chapter aims to elucidate the concepts and methodologies used in the literature review conducted in this thesis, also demonstrating each of the results obtained. A Systematic Literature Review (SLR) is a structured methodological examination of research that analyzes and classifies studies within a particular domain and systematically presents an overview of a specific topic through an organized and replicable process (KITCHENHAM *et al.*, 2009).

The revision carried out in this work followed the stages of the protocol proposed by KITCHENHAM *et al.* (2009). The search string was executed on the leading search engines, Scopus[1], ScienceDirect3[2], IEEEXplore[3], and El Compendex[4], as recommended by CUSKER (2013) and KITCHENHAM *et al.* (2009).

The concept of mod, is similar to the opportunistic SR approach. Consequently, the initial study conducted was an exploratory investigation aimed at identifying the building process of mods, their benefits, and development methodologies. This research was conducted utilizing a methodology known as Multivocal Review (MR). MR provides a comprehensive analysis of the literature to extract extensive information on a particular topic; hence, it integrates data from both white (academic papers, books, etc.) and gray sources (blogs, websites, videos, etc). This technique is commonly employed when there is significant community backing for the study topic, like the gaming community, and it is essential to validate practical knowledge on a particular subject (GAROUSI *et al.*, 2019). An MR can be divided into two phases: the initial phase entails pursuing academic information (utilizing an SLR), while the subsequent phase focuses on exploring gray literature. This preliminary research revealed that the majority of EVGs are generated via mods, the processes that involve their construction, and highlighted a gap in the literature regarding

[1]https://www.scopus.com
[2]https://www.sciencedirect.com
[3]https://ieeexplore.ieee.org/
[4]https://www.engineeringvillage.com/home.url

tools for mods development (CASTRO and WERNER, 2023). However, while the concept of mod similarities the opportunistic SR approach, literature indicates that it contains an important characteristic relating to the player's mode of expression, specifically considering the manipulations and concepts introduced by the modder. Consequently, the research scope was extended to include games generically. In addition, it was established that despite the extensive size of the gaming community, all content identified in the gray literature was also characterized via SLR; thus, it was decided to employ just the SLR method in subsequent investigations. In conclusion, it is important to emphasize that the current study concentrates on EVGs; however, the literature reviews exclusively employed the term "game" in order to avoid neglecting works that are pertinent to the investigation.

In addition to the mod review, another exploratory study was carried out, with the purpose of demonstrating which SR approach would be the most appropriate for specializing educational games. This study demonstrated numerous technologies, including frameworks, methodologies, and engines, were discovered to facilitate the building of games through reuse. However, these tools are mainly in the experimental stage or are usually associated with specific problems, such as a steep learning curve and complicated implementation. Among the tools identified, the most often used strategy to reuse was componentization, which is still utilized in modern engines. However, this approach was discarded because it alone did not allow for the specialization of games. The Model Driven Development is another approach and is a model-intensive approach to RS that emphasizes the use of abstract models that can be automatically converted into functional code, rather than explicitly writing code for all components of a system. This approach reduces development effort and enhances consistency. Nevertheless, automatic code generation frequently overlooks the importance of efficacy and performance in games. In view of this, the MDD approach was discarded. Finally, the SPL approach was demonstred as the most appropriate for specializing the development of EGs, primarily attributable to its capacity to generate multiple products from one origin.

The search string was developed using the PICOC framework (Population [5], Intervention [6], Comparison[7], Outcome[8], and Context[9]) at four levels (ARAÚJO *et al.*, 2022; PETTICREW and ROBERTS, 2008). Each of the levels will be described below. The search string was constructed by combining related domain-specific keywords using the logical operator "OR" and fields using the logical operator "AND."

[5]**Population**: the primary theme of the subject under investigation.

[6]**Intervention**: supplementary theme of the subject of investigation.

[7]**Comparison**: This dimension facilitates the comparison of one object of study with another.

[8]**Outcome**: what the study seeks to find.

[9]**Context**: aims to define the search string by including terms that demonstrate the subject's relevance.

To validate the search string, two control papers were used to create and run the string in the Scopus database, the first database where the string was applied. This validation technique aims to ensure the search string's quality by ensuring that it returns relevant papers of the author's knowledge (MORAES and SOUZA, 2011).

According to (MOTTA *et al.*, 2016) and (MATALONGA *et al.*, 2017), snowballing processes can compensate for the absence of other search engines and supplement the approach by doing research via the references and citations of the papers. Therefore, to minimize the loss of some papers and increase the search range, the forward[10] and backward[11] (one-level) snowballing procedure was used, which checks the references and citations of articles seeking relevance (WOHLIN, 2014). Following, each stage of the implementation procedure is described. The Apendix A shows the search string and a summary of the research protocol model carried out.

The search string was executed and returned the following number: Ei Compendex 265 papers; Scopus 262, Science Direct 223, and IEEEXplore 48, totaling 798 papers to be analyzed. All works not related to the work, book chapters, and conference invitations were eliminated in the selection by title, leaving 434 items to be analyzed. The abstracts of all works that did not align with the context were eliminated during the abstract selection process, resulting in 47 works that were eligible for analysis. The papers from the previous stage were thoroughly reviewed and analyzed to determine whether they addressed at least one research question. After this initial filter, 15 papers persisted and were chosen for the study.

After the procedure above, all remaining papers were applied to the snowballing process. The backward procedure returned 391 papers, verifying all references to the 15 selected papers. 41 papers were returned through the forward procedure, with all citations to the 15 selected papers being verified. An additional 432 items were added to the analysis. The new papers were subjected to the entire selection process, including title, abstract, and full-text reading. After the analysis, five more papers were included, resulting in 20 selected papers to extract research information. In total, 1230 papers were analyzed throughout the research. Appendix A shows the search flow utilized in this review, as well as the papers analyzed in it.

Most of the works analyzed merely designate the tree's characteristics but do not implement them. Furthermore, it is also possible to observe that there have been only three studies on the subject in the past three years. This lack of studies can be attributed to various challenges, including the selection of characteristics and manager the variability, the definition of the domain, and the line size (SARINHO *et al.*, 2018). Despite the low number of works found, only 20, most papers in the search

[10]**Snowballing Forward**: refers to the identification of new papers based on the works that referenced the paper that was analyzed (WOHLIN, 2014).

[11]**Snowballing Backward**: refers to the identification of new papers based on the works that were referenced in the paper that was analyzed (WOHLIN, 2014).

string addressed SPL issues in the context of game development without an educational context, indicating the importance of the topic. Then, each research question will be answered with the information discovered during the research period.

3.1 How and what features are used in SPL for the development of EVGs?

To create an EG, it is essential to comprehend the manner, timing, and characteristics of the game that should be applied to the player to establish a progressive and appropriate learning curve. The correct application of these characteristics assists the students in feeling more relaxed, motivated, and prepared to continue learning while enjoying themselves (MEFTAH *et al.*, 2018). Therefore, a series of characteristics must be considered, including gender, player count, difficulty progression, rules, victory conditions, score, effort, virtual environment, objective, time, and control, all of which are relevant to the overall development of games. In the educational context, it is possible to identify specific characteristics, such as assessment, exercise, content, feedback, and external references (MARTINS *et al.*, 2018). It is also crucial to know the appropriate time to collect each piece of information and to comprehend the players' emotions and psychological state, which includes everything related to their ability to take the initiative, their authority, their commitment, and so forth (MEFTAH *et al.*, 2018).

Specific works utilize SPL to develop serious games in the cognitive rehabilitation field, focusing on characteristics that enhance memory, attention, and concentration, which are critical topics for developing EVGs. These games can be created using a variety of platforms, interfaces, and technologies, with a particular emphasis on 2D and 3D interfaces, platforms such as computers, smartphones, and consoles, as well as technologies such as augmented reality, virtual reality, or traditional methods such as controls or a keyboard (TAVARES *et al.*, 2014). However, some authors contend that platform, interface, and technology are not essential for teaching a particular discipline, with the content of the game being the primary source of concern (MORAN, 2000).

Following the same theme of characteristics that enhance memory, attention, and concentration, certain studies investigate using educational activities that involve multiple-choice questions. To illustrate, this type of activity was employed to teach fractions, in which the player was required to select all sentences equivalent to the question (SÖBKE *et al.*, 2014). In addition to the academic quiz activity, other authors are also utilizing other activities with basic mechanics to enhance students' effort, motivation, concentration, and self-improvement. These activities include information correlation, crossword puzzles, organization, and image search (RINCÓN *et al.*, 2018). Even though these activities possess basic mechanics, they are regarded as an excellent method for providing self-assessment. They are repurposed by other game subdomains, serving as activities within a larger game and being used as activities within teaching tools, such as Moodle (SARINHO *et al.*, 2018). These activities are all called Educational Activities (EA).

The classifications of EVGs can be diverse and vary depending on the type of game, with a particular emphasis on action, adventure, construction, simulation, RPG, and so forth, or even a combination of multiple types (COOPER and SCACCHI, 2015; TAVARES *et al.*, 2014). Some mechanics and types are more appropriate for the context depending on the game's intention to be created. Games such as quizzes and matching are more relevant if the objective is to recall previously acquired knowledge. If the aim is to encourage students to think and analyze the circumstances, the most suitable games are puzzles, RPGs, and strategy. Simulators or construction games are recommended to stimulate students' creativity. It is important to remember that these types of games are not exclusive in their objectives; they can be combined to create a new game (BOLLER and KAPP, 2017b). Finally, it is recommended that games include a narrative for the player, facilitating the comprehension of content by individuals without prior experience. Some authors emphasize the importance of explicitly stating the objectives that guide the tasks that the player is performing. One method of elucidating the meaning is through the narrative characteristic, which involves the development of an engaging justification that fosters a connection between the game and the player by recounting the events that transpired (MARTINS *et al.*, 2015).

Progress tracking techniques were also emphasized as effective features for developing serious games. These techniques focus on the collection of each play to enhance the game, increase player engagement, and understand the player's challenges through the analysis of collected information (GEISLER and KAVAGE, 2021; O'ROURKE *et al.*, 2014; TAMLA *et al.*, 2019; TAVARES *et al.*, 2014).

The simplified SPL is illustrated in Figure 3.1, which is accompanied by Table 3.1, providing a detailed explanation of each characteristic, including its intended use and some considerations.

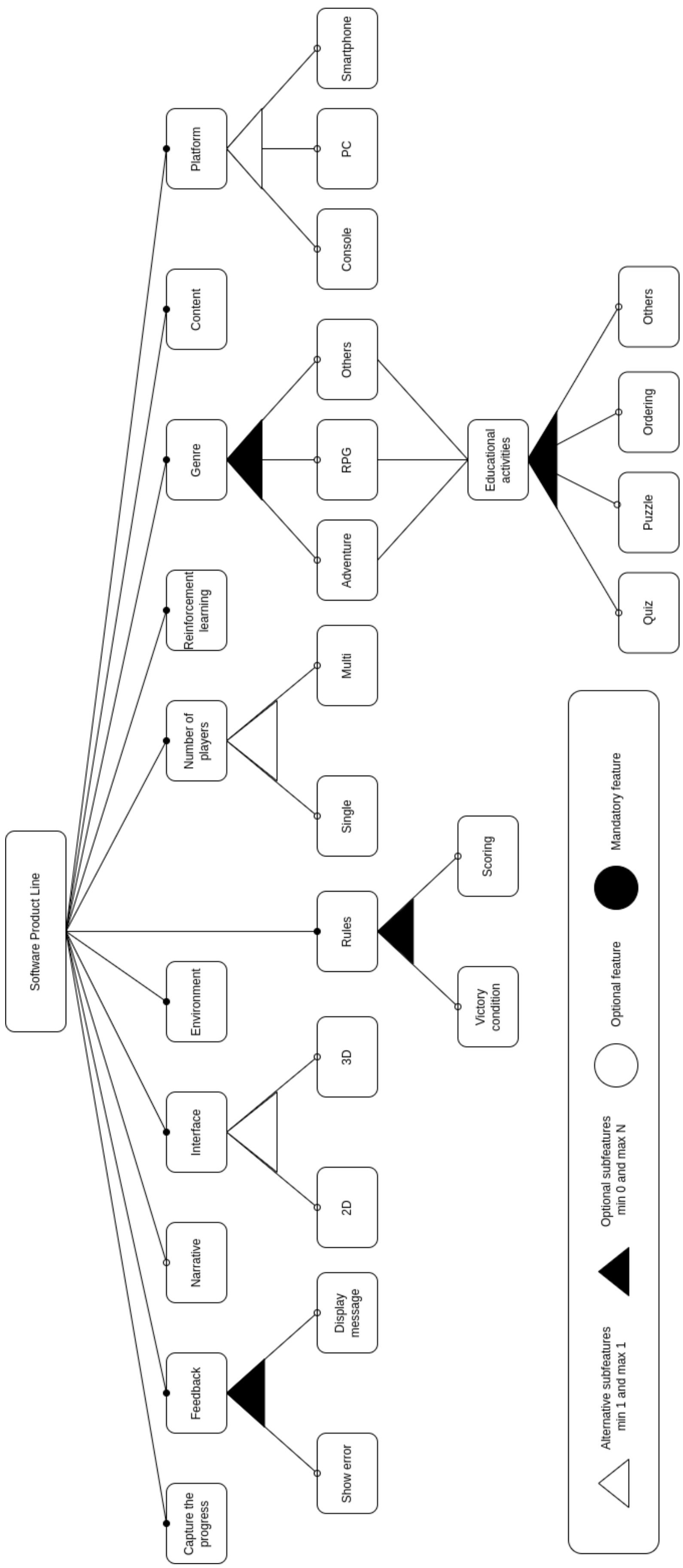


Figure 3.1: SPL for EVGs.

Table 3.1: Characteristics of EGs

Characteristics	Considerations
Genre	Aims to define the game in accordance with its gameplay style. Despite the existence of a variety of games and recommendations for each educational objective, the creation of an SPL that includes all of these types is practically impossible due to the sheer number of mechanics that the line would have to consider. As a result, it is advised that it be created for a single type.
Number of players	Establish the maximum number of players that the game allows. Multiplayer can enhance teamwork in EGs. However, interactions must be implemented twice in an SPL: once for single-player and again for multiplayer.
Rules	Determine the game's functionality and the manner in which players should interact with the game environment, thereby guiding the learning experience and the educational objective. It is necessary to assess each game characteristic and specialize the rules in order to enable their use in the various games that are to be created composing the SPL rules.
Victory condition	Conditions that must be satisfied in order to win a specific game. It is necessary to ensure that the player has assimilated the knowledge that the game is trying to transmit.
Scoring	Players' performance metrics. Reward the student who is acquiring knowledge along the game. The point value may be selected or not in an SPL; however, it is recommended that it be selected to increase the student's motivation.

Environment	Establish the environment in which the player will be immersed. In an EG, the player must first learn the game in order to subsequently learn the material being taught. Therefore, while it is feasible to exchange environments in the SPL, it is recommended to use the same environment for games that teach concepts that are within the same discipline, with the intention of reducing the learning curve of the game.
Goal	Establish the game's objective. The objective of an EG is to instruct on a specific discipline. Consequently, this objective is established based on the game's characteristics and the content that has been incorporated.
Platform	The new engines already enable the creation of builds for various platforms, such as cell phones, computers, and consoles, thereby simplifying the game's implementation. However, it is still necessary to exercise caution regarding the size of the images selected for the game. For an EG, the mobile version facilitates mobility; however, it is recommended that you use computers for larger games.
Technology and interface	Define the technology that will be employed in the game. The technology used in EGs has the potential to alter the way information is presented by incorporating 2D, 3D, and augmented reality objects, thereby increasing the player's motivation. However, it is recommended that only one technology be used to construct an SPL, as the forms of interaction are complex and vary depending on the technology. However, as an illustration, the 2D and 3D interfaces were incorporated into the image, while the forms of technologies such as virtual reality were omitted and augmented due to the difficulty of implementing them in a single SPL.

Learning progression	It is essential for an EG to be effective in order for learning to progress. However, the burden of the professor and the material he/she includes is on the SPL. Again, this characteristic is not present in the SPL image.
Progression tracking	Determine the method of data collection for the game. This monitoring is employed in EGs to assess the student's learning progress and comprehend the actions that were taken throughout the game. This feature is not affected by SPL; it is only necessary to employ a technique to accumulate the information.
Educational activities	These are basic mechanical exercises and evaluations that are regarded as an excellent method of self-assessment, and they are employed in EGs and tools such as Moodle. Examples include matching, sorting, image search, and quizzes.
Content	There is no value in an EG that lacks a suitable content for teaching. However, these characteristics are the responsibility of the teacher, who must select the appropriate material for the game.
Feedback	In order for learning to occur, it is necessary that the student comprehends all of the errors that occurred during the game and to demonstrate them in the appropriate manner. This method of error presentation occurs through feedback mechanics, which can be implemented in a variety of ways, including: the removal of points, damage to the player, the demonstration of the correct response, or even an explanation of the response.
Reinforcement	By presenting new activities in the game, it is possible to reinforce the knowledge gained from the students' errors, thereby establishing reinforcement learning.

Narrative	In order to gain a comprehensive understanding of the game and the discipline being taught, it is essential to have a narrative that unfolds, creating a progressive environment for the information that is presented. These narratives must be relevant to the subject matter being taught, and it is the responsibility of the teacher to develop a motivating narrative.

3.2 Why and when should SPL be used in the development of EVGs?

The development of games in their digital versions is an even more complex task, being multifaceted and requiring the integration of several fields of computing and technical skills. Most significant software development challenges arise during the development of digital games, including design, requirements, interface, and so forth. This is a costly endeavor that necessitates a significant amount of development time (BOAVENTURA and SARINHO, 2019). While this endeavor is discussed, standard card or board games highlight the prospect of inventing a mechanism or tool that distributes or reuses artifacts to support numerous games (SCACCHI, 2017).

A low budget often characterizes serious games and is usually called "the poor cousins of games" (WESTERA, 2019). Rather than developing new games, most EG designers focus on adapting (changing) existing ones (ABBOTT, 2018). This strategy reduces the experience required for game creation, as the ability to modify is based on the recognition/adaptation of game mechanics rather than the creation. These adaptations are inevitable; however, they can significantly reduce the effort and resources required to develop a serious, effective game (ABBOTT, 2018). These adaptations are necessary due to the significant cost, time, and effort needed to construct a game. These costs can be reduced if active and pedagogical components can be repurposed and integrated into the development process. Despite the existence of engines, frameworks, and development packages for game creation, the literature lacks tools to assist in creating EVGs, as they are only available for game development and lack educational focus (TAMLA *et al.*, 2019). EVGs can significantly contribute to the process of recapturing students' interest (LESSA FILHO and HERNÁNDEZ-DOMÍNGUEZ, 2016). Consequently, it would be optimal for teachers to create their games. However, it is illogical to assume that a single teacher could independently develop games for their classroom due to the effort required for game development, the absence of EVG tools, and the limited time a teacher has to teach, correct work, and administer exams (LEE *et al.*, 2020; SÖBKE *et al.*, 2014).

In general, serious games require the characteristics mentioned in Table 3.1, with many of them being similar to any type of game to be created. Considering all previously mentioned, applications that belong to the same market segment and possess a significant number of common characteristics can be developed using an SPL. This is the primary motivation for creating an SPL for designing EVGs (LESSA FILHO and HERNÁNDEZ-DOMÍNGUEZ, 2016; PERDEK and VRANIĆ, 2023; TAVARES *et al.*, 2014). In summary, an SPL enables the efficient management of products that belong to a specific domain and possess common and variable elements, which can be edited using variation points. The primary benefits of this approach include the reduction of time and costs associated with the development of an application and the enhancement of quality (MARTINS *et al.*, 2018).

Another reason for the use of SPL in the development of games is the construction of franchises. For instance, numerous successful games were converted into franchises through the development of an SPL, with a particular emphasis on Quake, Quake II, and Quake III, and all editions of Unreal Tournament (COOPER and SCACCHI, 2015; PERDEK and VRANIĆ, 2023). In the same way, it is possible to consider the use of this franchise strategy to construct EVGs. For a game to teach something, the player must first learn the game's functionality before beginning to understand the content being taught. Consequently, there are two learning curves: one for the game's functionality and another for the content intended to be taught through the game. To reduce the learning curve of the game and allow the student to concentrate just on the material being taught, EVG franchises can be established through the use of SPL. This allows for teaching subjects within the same discipline through games that share the same characteristics, sprites (images used in the game), characters, histories, and contexts. Consequently, all games that teach, for example, algebra, would employ similar games, presenting only the content to be learned.

3.3 Who and what current tools use SPL for the development of EVGs?

For digital games that put first entertainment, it is feasible to locate SPL in prominent franchises, including Quake, Quake II, and Quake III, and all editions of Unreal Tournament (COOPER and SCACCHI, 2015). Additionally, engines, frameworks, and development kits are accessible to facilitate game development. However, only three SPLs were developed, and there were not many tools for developing EVGs through SPL in the review. In general, most of the identified works only discussed the modeling of SPL without demonstrating its development or the tools necessary to use the line. The literature has specifically highlighted the lack of tools to support the creation of serious games (TAMLA *et al.*, 2019).

The initial SPL discovered is a simple tool designed to develop educational questionnaire activities, allowing for the selection of different platforms for game execution, such as a console, mobile device, and computer (SARINHO *et al.*, 2018). The second SPL discovered was for cognitive rehabilitation, focusing on characteristics that enhance memory, attention, and concentration, which are critical for developing EVGs. This SPL, known as SATReLO, also employs educational activities, such as dominoes, to enhance comprehension of patterns, object organization, search activities to enhance observation and comparison, and image sequencing activities to improve interpretation and activity progression. Furthermore, the tool records the player's scoring activities and accomplishments to later analyze them (MARTÍNEZ ARIAS *et al.*, 2021).

Another SPL discovered was not entirely automated; instead, it required coding by the user who wished to create. This SPL had a set of components that the user could edit, thereby facilitating the game's development and reducing the time and cost of implementation. However, the paper demonstrates that three games were developed from the SPL, known as JIndie, with inspiration from existing EVGs. For instance, the original game inspiration has 6.937 lines of code in the repository, while the SPL version has 20.210 lines, of which only 1.849 were implemented by the developer (LESSA FILHO and HERNÁNDEZ-DOMÍNGUEZ, 2016). The paper has chosen this metric even though code lines are not ideal for software comparison. Nevertheless, observing a substantial reduction in the number of lines is feasible. However, it is essential to note that the tool was not automated and required a technical understanding of the PHP programming language to develop the game. Consequently, the teacher was required to learn this language and allocate significant time to the development process.

Nonconventional SPLs were also discovered throughout the investigation, with the first one being constructed through reverse engineering of the bot source codes found in the version repository of GitHub for the educational Robocode game (MARTINEZ *et al.*, 2018). The objective of this game is to command a tank in battle to destroy one's opponent, with the bot's commands being developed in Java. Another unusual technique was the creation of the game Cat King, which was characterized by the reuse of a game designed with objectives to assist in the memory of basic legal knowledge; however, the game was designed to teach environmental engineering (SÖBKE *et al.*, 2014).

Finally, tools were also discovered that may assist in developing SPL, but they do not directly address the subject. One addresses progress tracking techniques, which were identified as crucial for the effective development of serious games. Game

Analytics is one of the tools identified for this purpose. It is focused on collecting user interactions throughout the game, aiming to increase user engagement. In addition, AI techniques were employed to monitor the player's progress; however, no specific instrument was provided (TAMLA *et al.*, 2019; TAVARES *et al.*, 2014). In addition to the notion of tools to facilitate the construction of the line, the Moodle tool was also discovered during the research, with a particular emphasis on the available educational activities (MARTINS *et al.*, 2018). These activities include quizzes, organization, matching, memory games, and more (DIAS *et al.*, 2024; SINNAYAH *et al.*, 2021).

3.4 How much effort/cost is involved in developing EVGs through SPL?

The dearth of knowledge exchange on networks may contribute to game development's costly and time-consuming nature, which often results in amateur developers "reinventing the wheel," an inefficient process (WESTERA, 2019). To attain widespread market acceptance, it is necessary to reduce development costs and time to market. This can be accomplished by enhancing software reuse and knowledge exchange (WESTERA, 2019). Furthermore, three aspects of EVGs restrict their development: the necessity for numerous resources, as they incorporate audio, image, animation, and story artifacts, in addition to the complexity of software development; the challenge of ensuring that the content to be taught is transparent; and the rapid degradation of technical and technological capabilities (MARTINS *et al.*, 2018).

As previously mentioned, one of the primary advantages of SPL is the reduction of the time and cost associated with the development of similar software (CAPILLA *et al.*, 2019). As a result, the time required to develop a game following the construction of an SPL is significantly reduced, in addition to enhancing its quality (MARTINS *et al.*, 2018)). However, creating an SPL, whether from the ground up or through reengineering existing variants, is typically a multi-year project that requires a significant investment. This process involves the collection of product characteristics, the selection of similarities, the definition of variation points, the coding of the tree, and other activities (MARTINEZ *et al.*, 2018; MARTÍNEZ *et al.*, 2018). To illustrate the magnitude of the effort, a game developed using reverse engineering to create a game with simple mechanics and assessment activities accumulated 1000 hours of effort (SÖBKE *et al.*, 2014).

In general, there were few comments regarding the time and cost of developing an SPL for EVGs, which may be directly linked to the difficulty of analyzing and quantifying the investment in this activity (MARTINEZ *et al.*, 2018; MARTÍNEZ *et al.*, 2018) and the low number of SPLs found due to the development difficulty (TAMLA *et al.*, 2019).

3.5 Final consideration

The information collected in this study was used to determine the important features for building an SPLEG, the logic behind its development, the current tools that can assist in the development of this activity, and the amount of effort required to develop it.

Among the main characteristics found, the following stand out: learning progression, educational activities, narrative, feedback, and reinforcement learning, among others. Moreover, the motivation for development was evidenced, as it indicated that this effort could reduce both the time and cost of game production while concurrently lowering the learning curve for students by employing games with similar structures, visuals, and narratives.

Given the information collected, it was possible to observe that the solution of developing an SPLEG was viable, allowing the construction of prototypes to verify the difficulty of developing this SPL.

Chapter 4

Proof of concept

The main objective of this chapter is to show the games that were developed to validate and verify the possibility of developing games through the use of SPL concepts.

4.1 Dynamic Tetrad Game

As mentioned, any game can be explained through its mechanics, second-level mechanics, and aesthetics. In light of this, it can be asserted that a new game can be created if each of these elements is changed. Thus, a new prototype based on the elemental tetrad was constructed, in which changes are applied to the game at runtime, producing new mods periodically.

The prototype built applies the concept of Dynamic Software Product Line (DSPL) for building VGs. As a prototype, this technique was utilized to construct the game due to its higher efficiency in arranging future changes since the game does not need to be built each time a change occurs. The characteristics of the classic product line are developed during the design phase; however, the development of the prototype of a DSPL (AYALA *et al.*, 2021) was chosen due to the ease in coordinating the changes within the game.

The game functions like an infinite runner (CASTRO and WERNER, 2021). Each time the player reaches a checkpoint (marked by a flag), a new random combination of game mechanics, second-level mechanics, and aesthetics is generated, thus producing a new version of the game at runtime. Figure 4.1 shows four phases generated at random by the game. Each time the player meets a flag, a new one game is produced. The game displays a panel showing the selected configuration to visualize the specified attributes. Figure 4.2 shows the panel. From this figure, it is possible to observe all the characteristics being modified throughout the game. Table 4.1 describes each feature in more detail.

Figure 4.1: Elemental tetrad Generation Game (CASTRO and WERNER, 2021).

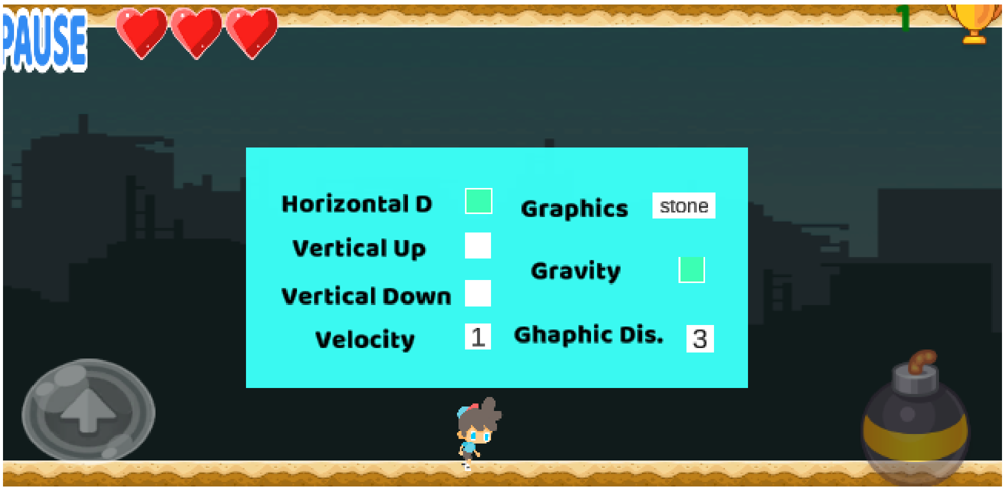


Figure 4.2: Configuration panel (CASTRO and WERNER, 2021).

Table 4.1: Changeable game features;

Characteristic	**Explanation**
Movement type	Controlled through 3 booleans. **Horizontal:** The character will run horizontally. **Vertical up:** The character will run from top to bottom in a vertical position. **Vertical down:** The character will run from bottom to top in a vertical position.
Velocity	It is controlled through a number with 3 values (1, 2 and 3) that creates a vector with fast, medium and low speeds.
Graphics	The game has 3 stages: Desert, Ice, stone and swamp.

Gravity	It is controlled through a boolean that defines whether or not the character will float.
Graphics Display	It is controlled through a number with 3 values (1, 2 and 3) that creates a vector of distance between the objects on the screen. Control through which the game will check if there are more or less spaced objects.

Table 4.2: Game mechanics, dynamics and aesthetics.

Mechanics	Run and jump or fly, destroy objects with bombs, destroy objects by clicking on the object, lives, score, pick up coins to buy new levels, lose lives by hitting objects
Second Level Mechanics	Increase or decrease speed, increase or decrease gravity, change character directions
Aesthetics or Interface	Fire environment Ice environment Earth environment Air environment

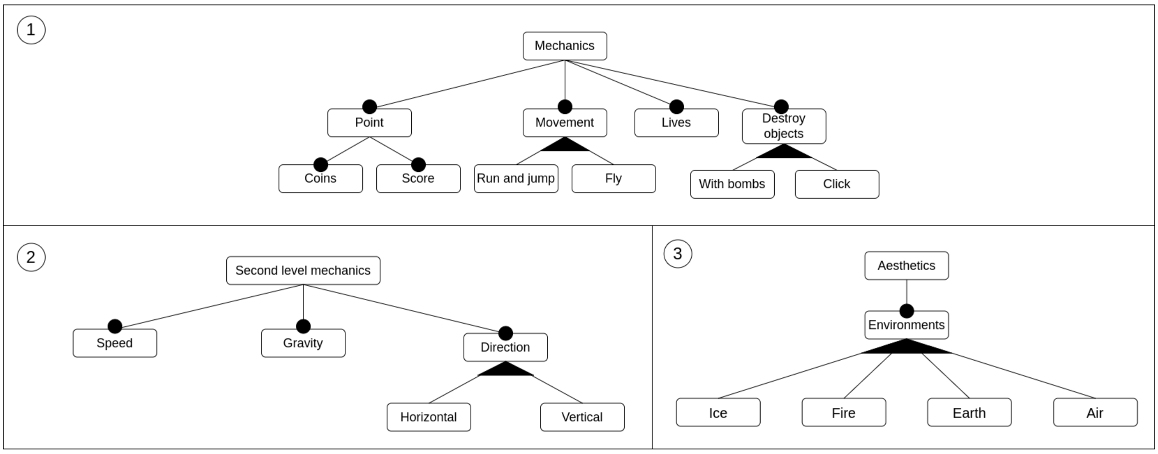


Figure 4.3: Elemental tetrad Feature tree (CASTRO and WERNER, 2021).

Table 4.2 categorizes the game's properties according to some levels of the elemental tetrad. The required gameplay elements include lives, scores, money, and the loss of life when colliding with level objects. Note that any random mod will possess these characteristics. A boolean variable for each random game feature determines whether it will be implemented or not. For each mechanism, the game will pick between walking, jumping, or flying, destroying items with bombs, and damaging objects by clicking, serving the same rule for the second-level mechanics. Each step will pick a terrain type for aesthetic purposes.

Figure 4.3 demonstrates the feature tree for full-game functionality, where filled balls are required, and empty balls are optional. Triangles with fill indicate that just one feature may be selected. This graphic depicts a model based on the FODA characteristics model (KANG *et al.*, 1990). The graphic has been separated into the three components of elemental tetrad to facilitate viewing.

4.2 Classic Tetrad Game

This work's first prototype aimed to develop a manual product line derived from an original game built from scratch with changes made directly in the source code. The previous section described a prototype of a game that aimed to simulate a dynamic product line; this one was a bit more challenging, but the changes made during runtime were easier to manage because the game did not need to be recompiled for each change. The game described in this section seeks to replicate a classic SPL, with the game being constructed throughout development, hence increasing the implementation difficulties.

The game may be interpreted as a platform where the protagonist must complete the level's three tasks. At the beginning of the game, the player must select the characteristics they want to incorporate. There are three feature trees for this purpose, one for each level of elemental tetrad. Figure 4.4 demonstrates some levels generated by the game, from which it is also possible to see the objectives selected to be conquered. Figure 4.6 demonstrates the game's characteristics selection menu, having a characteristics tree for each part of the elemental tetrad. This figure also displays the game's rules menu, which describes how to play. Figure 4.5 demonstrates the positioning of each elemental tetrad element in the game.

The game works as a product line where the player can choose each part of the game, such as its mechanics, objectives, enemies, and other elements. The player will select their preferred characteristics in each game, and the game builder will create a game with the selected characteristics. Next, each element that the game can choose or modify will be described. Figure 4.7 demonstrates all three FODA (KANG *et al.*, 1990) trees with the characteristics that can be chosen.

Finally, several features were added to make the game more dynamic and complex, including characters with varying amounts of health, distinct attacks, and an A* algorithm to follow the character.

Elements that can be selected or modified:

Figure 4.4: Levels generated by the game.

Figure 4.5: Game feature selection trees, grouped according to the elemental tetrad.

Figure 4.6: Game characteristics selection trees.

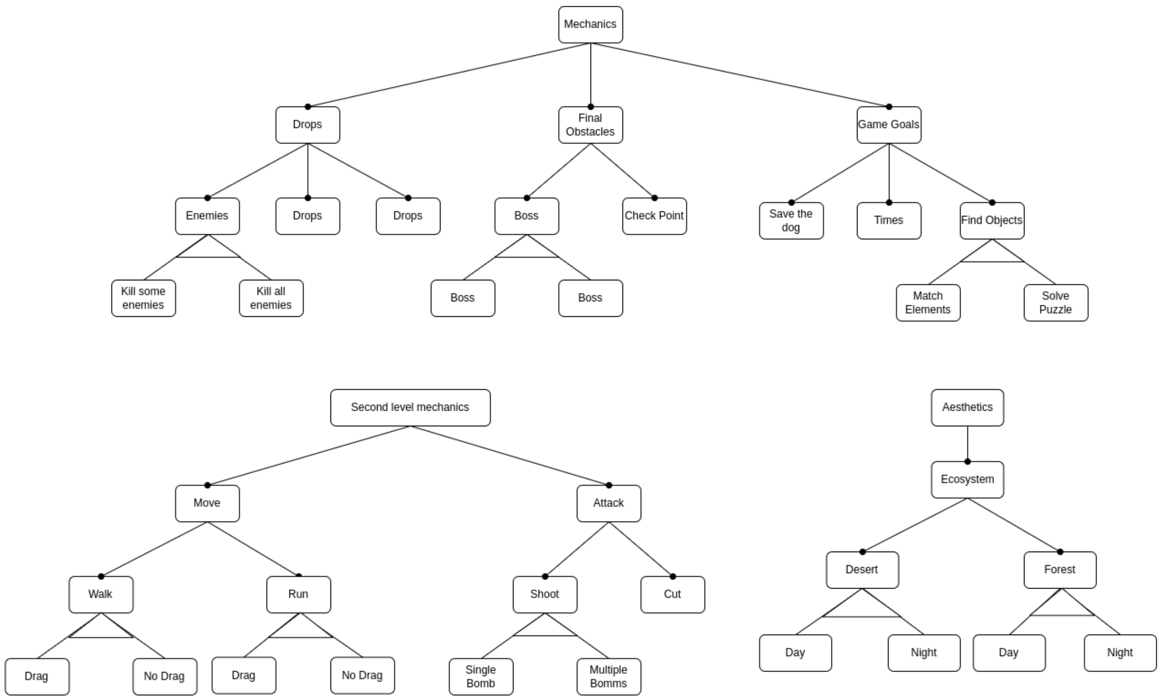


Figure 4.7: Tetrad SPL Classic game feature tree.

- **Mechanics:**
  - **Drops:** Drops in the language of games can be understood as the act of an enemy dropping an item in a match; however, in this game, it was used with a sense of items or elements that are dispersed across the game.
    * **Enemies:** Choose whether the player should kill all enemies.
    * **Coins:** Items must be collected in the game.
    * **Lifes:** Items must be collected in the game.
  - **Final obstacles:** obstacles to be overcome by the player.
    * **Boss:** choose if the game will have one or multiple bosses.
    * **Check Point:** choose whether the game will have checkpoints.
  - **Game goal:** main objective of the game.
    * **Save dog:** save the dog from the cage.
    * **Timer:** choose if the game should be won before time runs out.
    * **Find objects:** Select the items the player must find in the game. There are two options: match elements, which must find the secret object, and puzzle solver, which must find similar objects.
- **Second level mechanics:**
  - **Move:** choose the second level movement mechanics.

    - **Walk:** the character can only walk.
    - **Run:** the character can only run.
  - **attack:** choose the second-level attack mechanics.
    - **Shoot:** Choose between the two possibilities of weapons, with one shot at a time or many shots.
    - **Cut:** choose weapon with sword option.

- **Aesthetics:**
  - **Ecosystem:** choose the game stage.
    - **Desert:** choose between day or night.
    - **Forest:** choose between day or night.

This game allowed the player to select the game's elements, rules, and objectives, creating a unique game for each tree configuration. This game was hard-coded, being necessary to develop several conditionals for its creation, resembling a real SPL.

4.3 Evaluation

This session will present the planning and preliminary qualitative results of the prototypes built to validate this work's approach. It will describe the planning, participants, procedure, and results.

4.4 Planning

A study was carried out to validate this approach and find evidence about the gaming community's importance, need, and approval in the face of the thesis to create the platform.

The primary item in this initial evaluation would be the community's approval of the platform's development, which would be a technology validation. Then, some questions from an adapted Technology Acceptance Model (TAM) questionnaire were used (DAVIS, 1993). TAM collects data primarily on the usefulness and usability of the presented approach, allowing users to determine whether the approach to be built will be helpful to the community. This model is well-known in the academic field for measuring technology acceptance. Its strengths include focusing on technology-specific information, being extensible, allowing it to be applied in different contexts, and being used during and after adopting a specific technology. However, some questionnaire questions were modified in this work to validate the idea's possible usefulness before the platform's development. However, since the concept is validated through a game, the MEEGA (PETRI *et al.*, 2016) questionnaire was also used. This questionnaire is used primarily to validate the usability and experience provided by a game; however, in this evaluation, it will be used to determine whether or not the usability and experience of the game influenced the acceptance of the technology. It is mainly used to ensure that if the game has a problem, it does not inhibit the approach from being accepted. Therefore, the MEEGA questionnaire will validate the game's usability and experience, and the TAM questions will evaluate the tool to be built.

The evaluation procedure ran from 10/10/2022 to 10/21/2022, with a pilot evaluation on 10/07/2022 with a participant to make sure that the game and questionnaire had no issues that would affect the assessment, as well as to confirm the execution time that a candidate would take to experiment. Following the pilot execution, it was possible to determine that the procedure took an average of 30 minutes and did not present any problem that would affect the experiment.

4.5 Participants sample

For the pilot study, only one undergraduate student was used to validate the games and questionnaire to identify possible problems. Three groups of participants were chosen for the main study. This division of groups was created to divide the participants into different experience levels. Each of these groups will be described in more detail below.

- **Experts:** The first sample was selected from graduate students from the Federal University of Rio de Janeiro (UFRJ), UNIRIO, and Rio de Janeiro State University (UERJ) who had prior experience with games and Software Reuse. Because this group had more experience in the area, it attempted to validate the idea more prudently.

- **SR students:** The second population was drawn from a sample of students enrolled in an SR course at UFRJ. This population had less experience with reuse than the first, but they were younger participants familiar with games and programming. From this population, it was intended to obtain a less rigorous perspective than the first group and with a vision a little bit focused on people who already used games and programming.

- **Gaming community:** The third group of participants was chosen from the gaming community, leaving the invitation open to anyone wanting to participate. This group was formed to gain a less academic and rigorous perspective and understand what the gaming community looks for since they are the biggest mod creators.

4.6 Procedure

The study was conducted remotely through the availability of the materials required for the study's execution. However, for the pilot study and the first two populations, the entire experiment was conducted via a Google Meeting call, with the think-aloud protocol (JÄÄSKELÄINEN, 2010). This protocol was used to collect additional information, such as whether the player was having difficulty with the game and whether they liked the proposed idea, among other things. Using this protocol, it was also possible to capture the environment's sound, which aided in understanding some sensations felt by the player through their reactions, such as claims, sighs, and expressions of fatigue or stress. The main stages of this study are as follows:

- **Study and game description:** This step involved distributing a form containing the primary information required to conduct the evaluation, such as explanatory texts about the study, basic game commands, and URLs for the questionnaire and installer. It is worth mentioning that the questionnaire was made available in two versions through Google Forms, English and Portuguese, seeking a greater number of participants. The questionnaires used can be found in Appendix B and C.

- **Participants' characterization:** A characterization questionnaire was made available to each participant.

- **Game execution:** Participants must install and test the two games sent to them.

- **Completing the qualitative questionnaire:** The TAM (DAVIS, 1993) and MEEGA questionnaires (PETRI *et al.*, 2016) adapted to the context were made available to each of the participants.

4.7 Results

The main objective of this evaluation is to determine the feasibility of the SPL development concept for the development of EVGs. Due to this, the two games described in Sections 4.1 and 4.2 were evaluated using a single questionnaire, as the purpose of the evaluation was not to assess each game but rather to introduce the concept of the platform and determine its viability. This decision was taken to reduce or optimize the evaluation time and thus obtain a better result. The other two games were not evaluated because they were handheld.

Despite the questionnaire used to collect information about usability and game experience, this information was collected only if any game feature was misunderstood or was not to the user's liking; however, this does not affect the final result of the evaluation.

4.7.1 Participants sample

The study involved 46 evaluators, who were divided into three groups: specialists, who were defined as expert users with experience in the field of games and SR; students who took the SR course; and the gaming enthusiast community.

It is worth remembering that many different answers were given in relation to the participants' time of experience. In order to present this result in a concise manner, these answers were divided into six categories: More than 10 years, Up to 10 years, Up to 5 years, Up to 3 years, Less than a year, and Without experience.

4.7.2 Analysis of results

Figures 4.8 and 4.9 show how the study specialists were classified, showing that three specialists evaluated the study, each with a distinct level of expertise in each subject. From these figures, it can be inferred that two users have been involved with games for over ten years. However, it is essential to note that this time is only about the time they had contact with games, whether they were playing a video game or doing more research on the topic. However, when it comes to game development, it should be noted that only one member has been an expert in the field for more than 10 years, while the other has only 5 years of expertise. It is important to note that this experience time decreases even more when mods are discussed, with 5 and 3 years of experience for each participant. About SR, note that all participants have at least 10 years of experience in the subject, but when dealing with SPL, only one participant has this time of experience. When analyzing this information, it is possible to notice that the group consists of a somewhat more game-focused specialist, one more expert about reuse, and a more specialtist user.

When looking at the student research group's data, it is possible to see that the students are all around the same age and are still enrolled in graduation. The students' characteristics are included in Figures 4.8 and 4.9.

Another noteworthy statistic is that while half of the students have more than 10 years of expertise with digital games when it comes to game development, this experience drops to 3 or even zero. Finally, in terms of expertise with SR, all students had less than a year of experience with SR, which was to be expected given

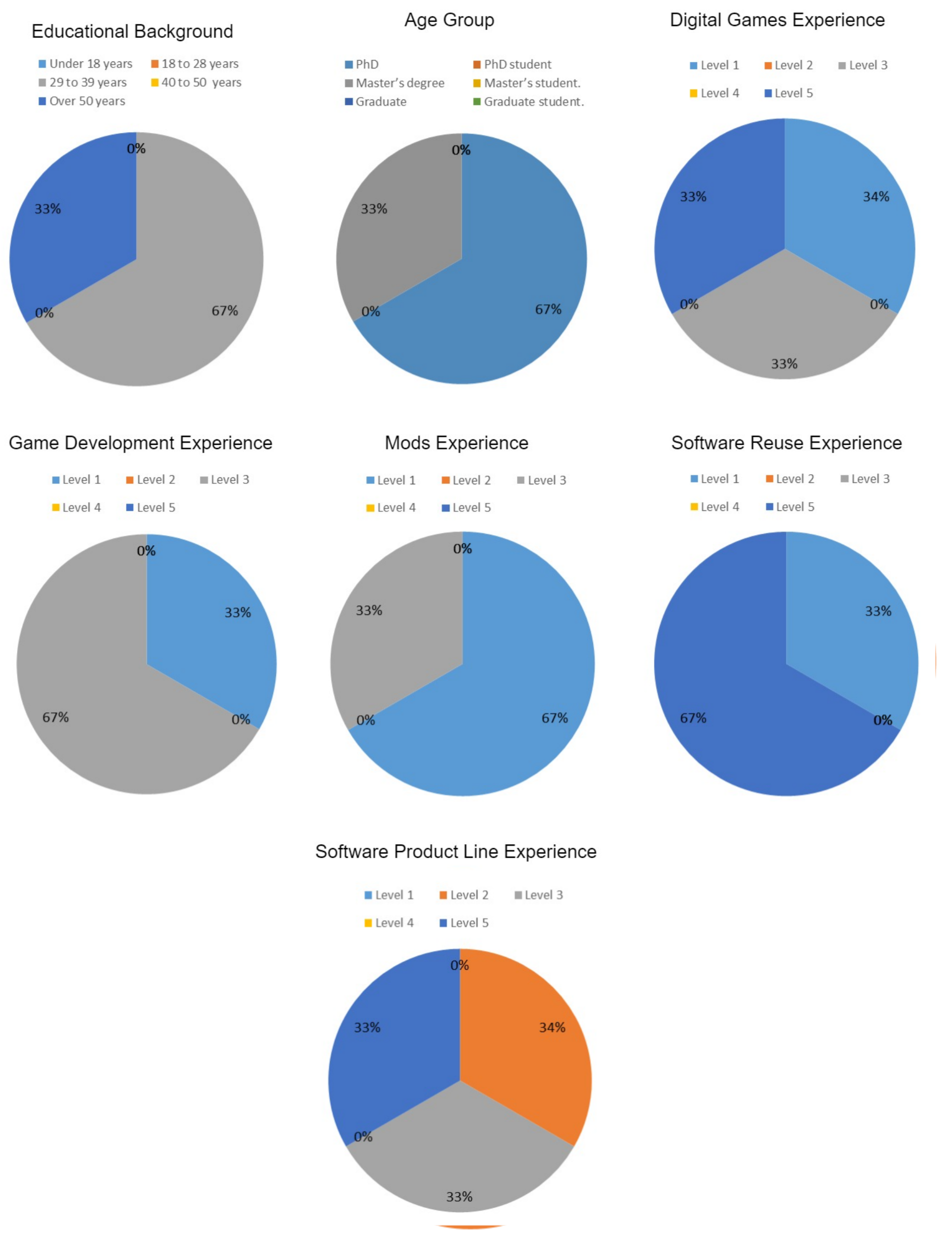


Figure 4.8: Characterization of specialists.

that they are undergraduate students and are only now familiar with the field. It is important to note that although everyone was familiar with software development and SE, they knew little about the SR.

When looking at the community assessment group, it is clear that most participants are between the ages of 18 and 28 and are either graduates or in the process of graduating. When the experiences of these participants are examined, it is clear that most of them have been in contact with games for a long time, having been players for more than ten years and having knowledge about game development for about three years. However, when it comes to SR and SPL, it is possible to notice that most evaluators lack or have limited experience in the field. This was

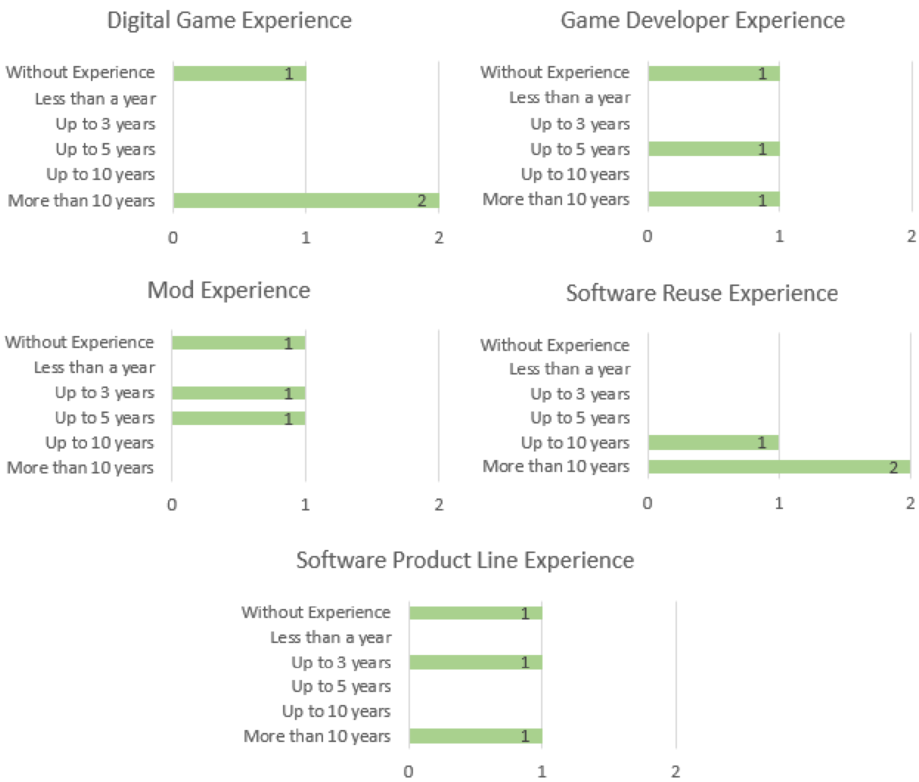


Figure 4.9: Experience of specialists.

expected because the study was conducted with the gaming community in mind. Figures 4.13 and 4.12 demonstrate information about the characterization of the community study group.

Despite having three groups of participants with varying knowledge about games and SR, the answers generally followed a pattern. However, it was possible to notice slightly lower scores in the specialists' answers, which was expected given that they are specialists and expect a more accurate result. Figures 4.14, 4.15 and 4.16 demonstrate the answers regarding usability, experience, and usefulness of the proposed game.

It should be noted that only one questionnaire was used for both games, considering that the main purpose of the evaluation was to check the viability of building the specialized SPL for the development of ECGs. However, a few Meega questionnaire questions were also used to see if the game's usability and experience could influence the evaluation's final result.

Regarding the game's usability and experience, it was possible to notice that it did not directly affect the study's results; however, some improvements that could be made were identified, such as improving the character's movement, having more instructions on how to play, and using some context variables to improve the player experience by randomly generating the game. Even though only one questionnaire

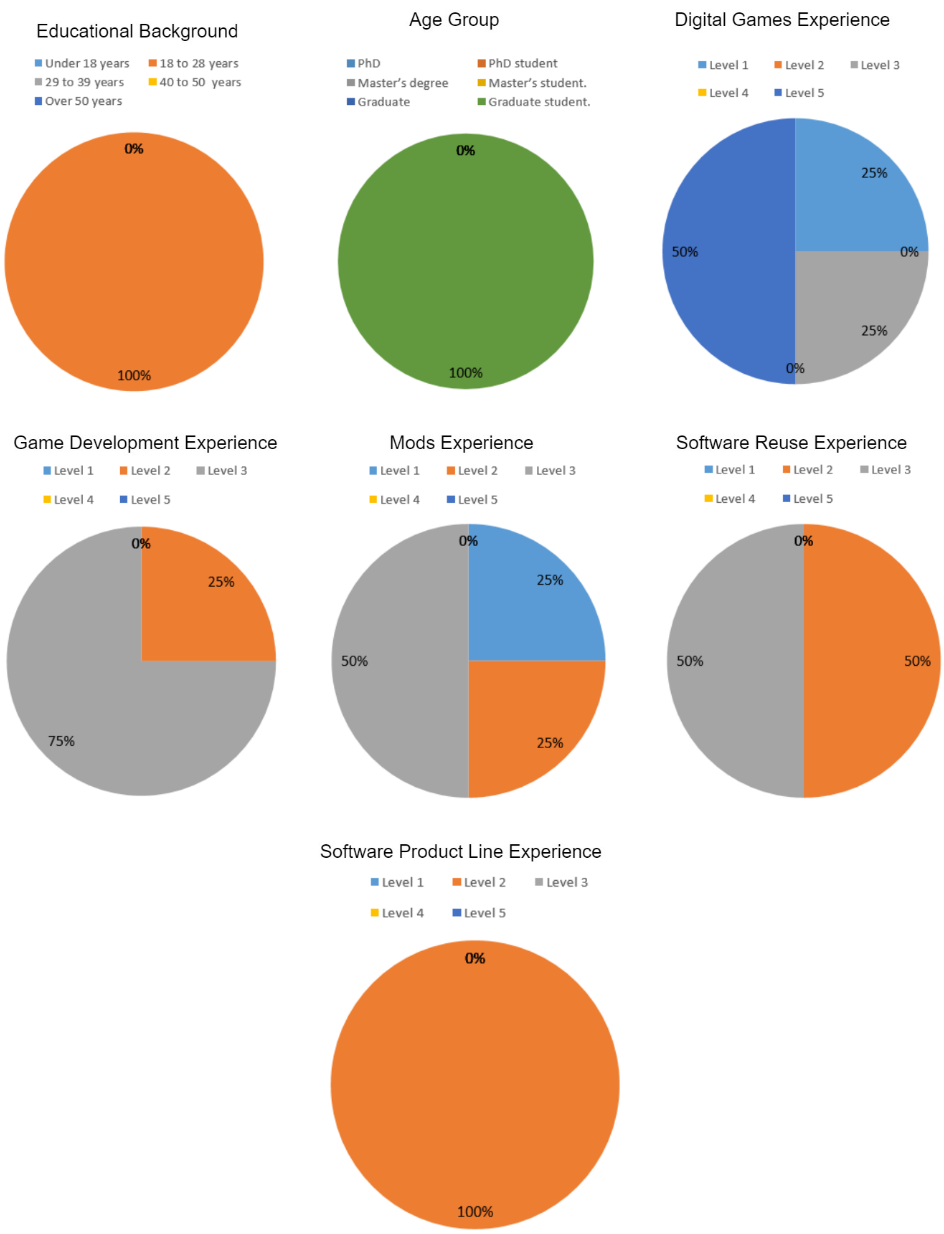


Figure 4.10: Characterization of students.

was used for both games, it is believed that most usability issues are related to the first game, based on comments such as utilizing context factors to generate the game at random to enhance the user experience. Engaging, beautiful, and intuitive game. However, understanding how games are generated can be difficult for non-gamers.

Regarding the verification of the utility of the game, a considerable number of favorable answers were observed, indicating that users could construct games from the notion of SPL and comprehend how the concept of a feature tree would function. The questionnaire generated responses such as: Regarding the possibility of deriving games from the selection of line elements, it appears to be a solution option with actual application and contribution to the community. Exciting concept: I created

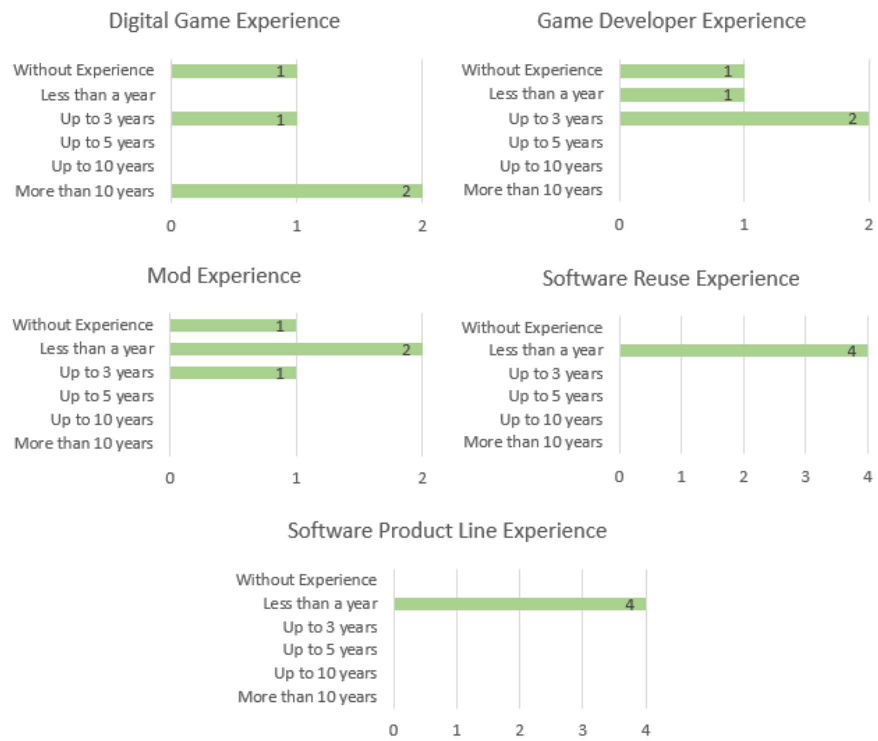


Figure 4.11: Experience of students.

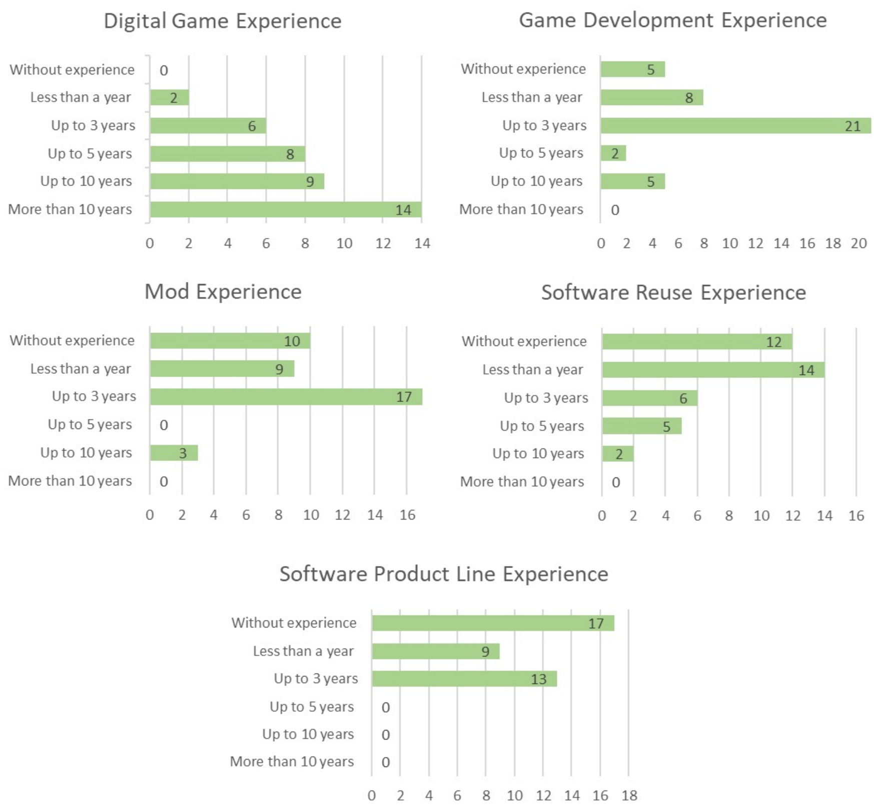


Figure 4.12: Experience of community.

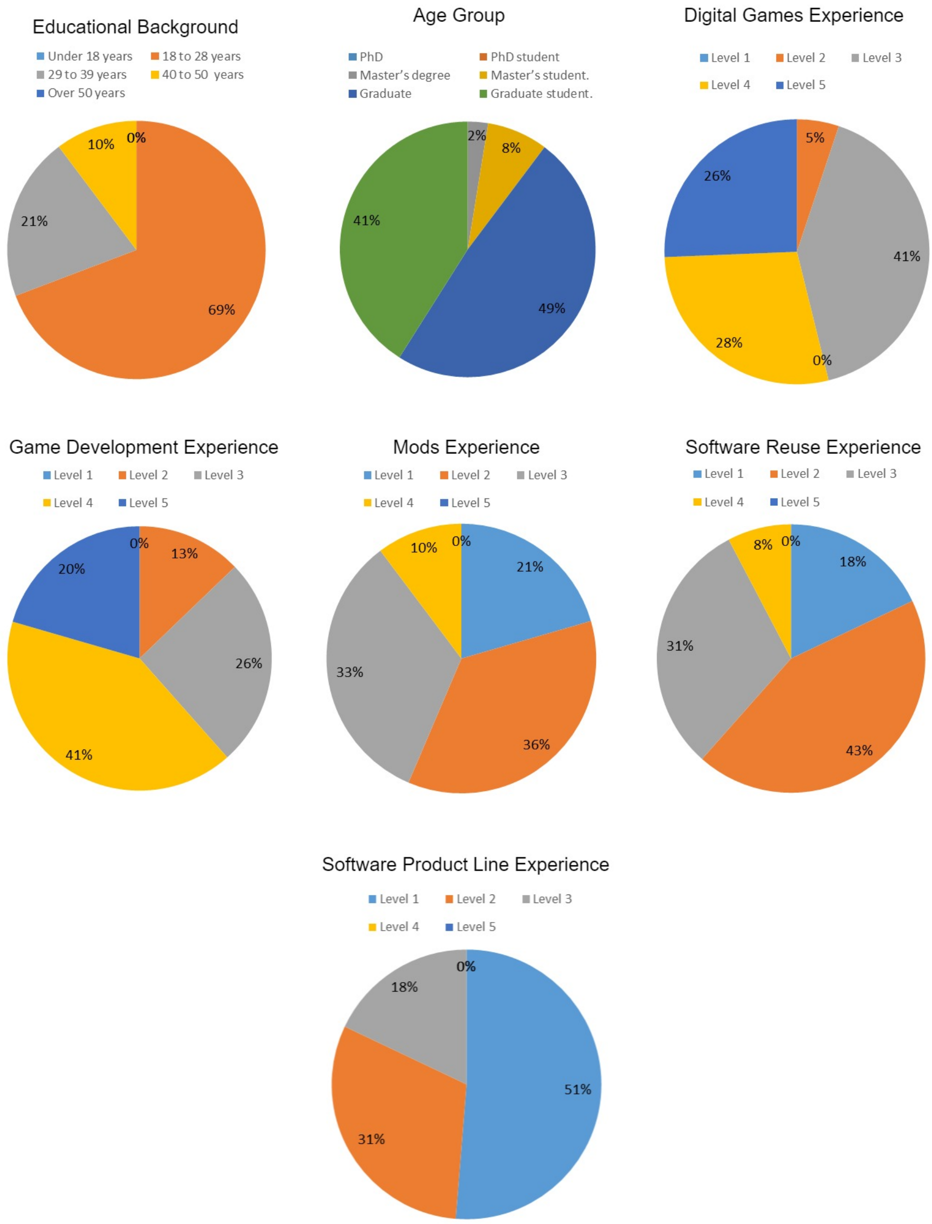


Figure 4.13: Characterization of community.

three completely different games, which is an interesting concept that would greatly benefit the game development community.

Looking at the answers in general, it is possible to notice that more than 70% of the answers obtained positive results (Agree and Strongly Agree) and that approximately 95% of the respondents gave answers greater than or equal to Indifferent, with only 5% of the answers obtaining a negative note (Disagree and Strongly Disagree), based on a total of 46 participants. This percentage is even higher when considering only the answers that discuss the tool's usefulness, with more than 81% of positive responses, more than 98% of responses above indifferent, and fewer than 2% of adverse reactions. From this, it is clear that the suggested games has a great

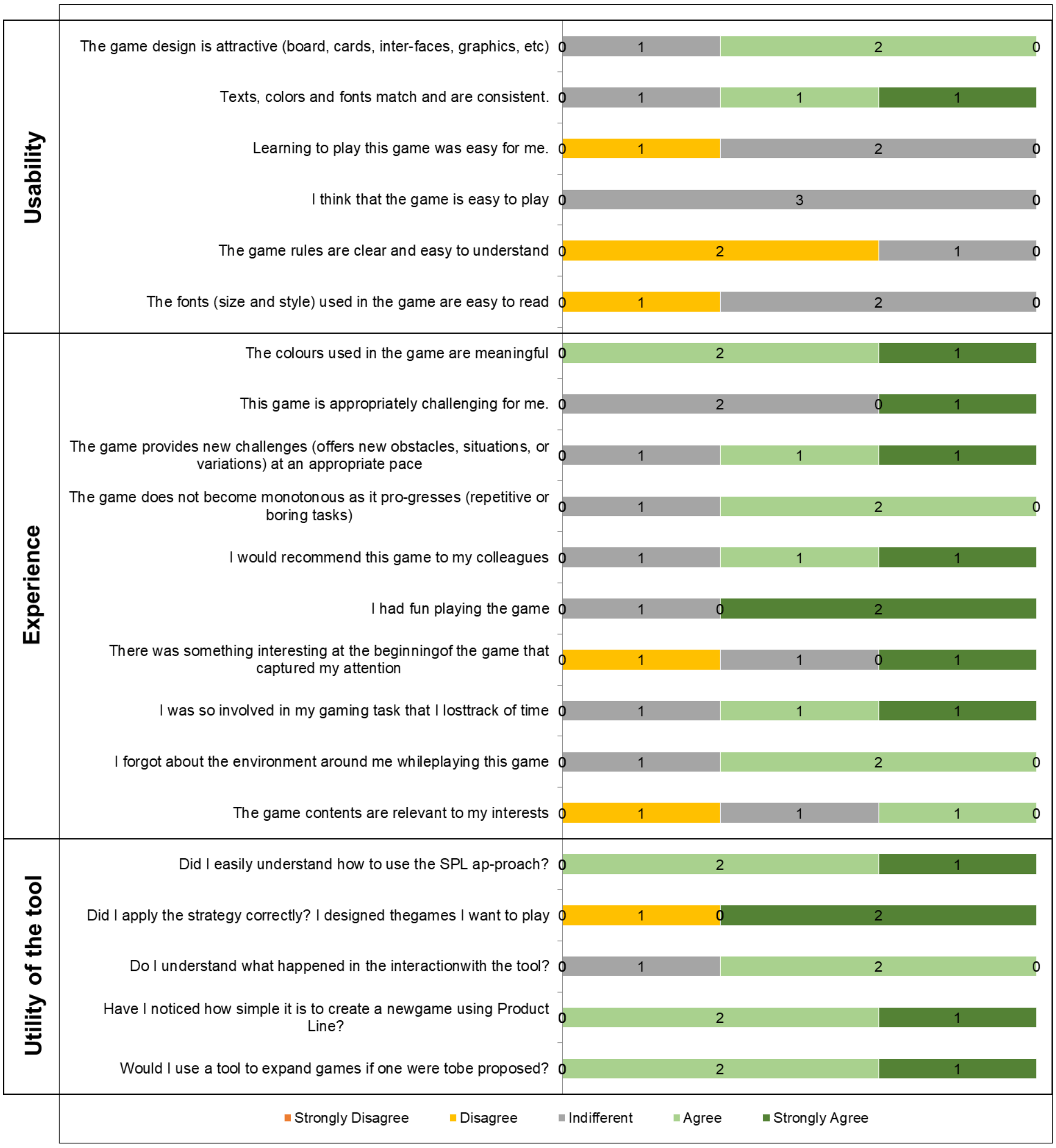


Figure 4.14: Meega / TAM Questionnaire with Specialists Response.

deal of promise and utility and has the potential to aid the community in developing games.

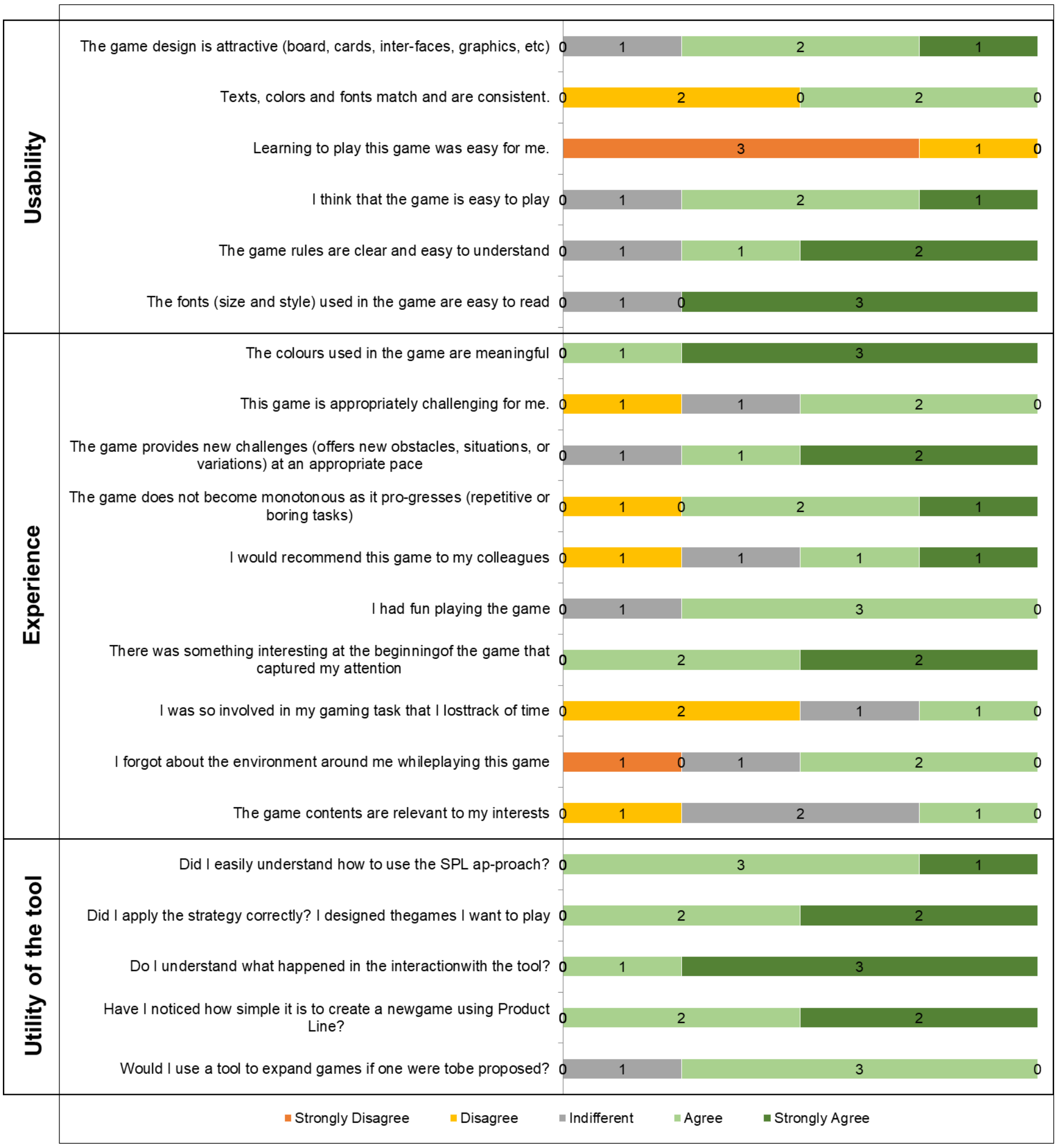


Figure 4.15: Meega / TAM Questionnaire with Students Response.

4.8 Final Consideration

This chapter introduced two games developed utilizing diverse SPL concepts and creative endeavors. The first concept involved developing games via a dynamic product line, wherein the game would randomly select all of its new features. The second game was developed, allowing the user to choose the characteristics of each game, in contrast to the first.

The two games created attempted to demonstrate the concept of building the solution to be proposed, and an evaluation was conducted with them to verify the feasibility of building the SPL. In general, regardless of the group of evaluators, it was possible to notice many positive responses for both usability and experience,

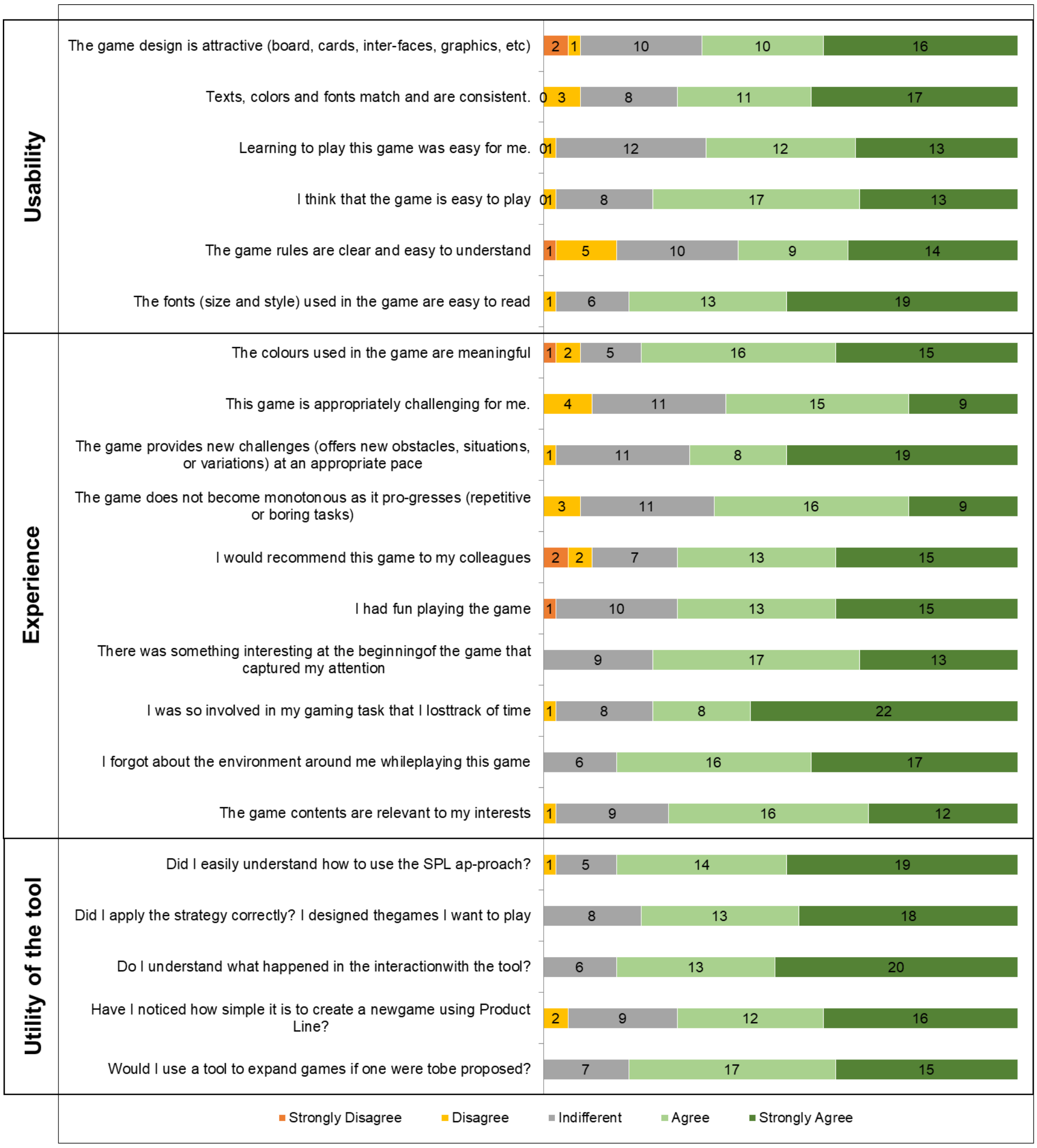


Figure 4.16: Meega / TAM Questionnaire with Community Response.

thus demonstrating that these variables did not influence the final result of the research. Regarding usefulness, it was also possible to perceive positive responses, showing that the solution is valid and would be helpful in the community. With this, the specialized SPL was developed.

Chapter 5

Survey

Games are characterized by numerous factors, including objectives, rules, constraints, interaction, challenges, competition, rewards, feedback, and other features. Based on each characteristic that originates a game, it is possible to measure its teaching effectiveness. Many factors influence teaching directly, including satisfaction, motivation, interface, usability, experience, and the selected characteristics (CASTRO, 2020).

The review in Chapter 3 found several characteristics that should be part of an SPL to develop EGs. Based on these characteristics, questions were formulated to confirm the review results; the forms used in the Surveys can be seen in Appendices C and D. The survey consisted of two groups, each focusing on separate respondent groups. The first group included 23 participants, including researchers and educators in EGs, game developers, students, and enthusiasts. The second group, consisting of 12 participants, including participation of a game development company in London, with the company's name omitted to maintain the confidentiality of the information provided. The first group of the study will be called Group A, and the second Group B.

The first survey section collected demographic data: (1) the gender, (2) age group, (3) research area, (4) level of education, (4) years of experience with EGs, (5) years of experience with SPL, (6) number of SPLs developed and (7) number of SPLs developed for EGs. The second part collected data regarding what characteristics an SPL should have.

Group A consisted of a significant amount of gaming experts, including 14 participants aged over 39, 11 with PhD degrees, 2 with PhDs in progress, 5 with completed master's degrees, 3 with undergraduate degrees, and just 2 participants with undergraduate degrees in progress. These two individuals indicated they were not involved in the gaming sector. Consequently, these two were excluded from the study, resulting in 21 participants remaining in group A. In group B, just two members had a master's degree, 10 had completed their degrees, and one had a degree in progress.

It is evident that most of the participants in this group consisted of recent graduates or had been working in the area for approximately three to five years. The majority of the participants were between 18 and 29. Figure 5.1 demonstrates demographic information in more detail.

Figure 5.1: Demographic data.

Group A consists of professionals with solid academic backgrounds; more than fifty percent of the participants in this group completed their doctoral degrees. This characteristic coincides with the prevalent age group since most respondents are older, indicating that they have a solid basis of experience. When questioned about their experience with EGs, about half of the participants stated they had more than ten years of experience. On the other hand, this proportion is not mirrored in the SPL area, where more than half of the respondents declared that they had no expertise and had never used this method. Most participants stated that they had never created an SPL concerning the quantity of software product lines developed for educational games. Despite this, seven participants reported having already developed SPLs focused on games. The period of the experience in group B is significantly shorter, which is indicative of the lower age group of the participants. Nevertheless, all members of group B participate directly in developing games in the job market.

Only two participants admitted working with SPL despite their daily involvement. It is possible that most participants in both groups only created a few or no SPLs for games, reducing the number when discussing SPL for educational games. Nevertheless, one participant indicated that had already generated 10 SPL for games; however, the number was considered an outlier due to its significant divergence. The statistics suggest that the term SPL is uncommon in game development. The surveys revealed that the term is frequently linked to franchises. While there are theoretical parallels between franchises and SPLs, where several games have similar characteristics and minor modifications generate distinct products, significant differences exist at the level of development, where an SPL is constructed by a tree that exhibits all product features, any of which may be selected, edited, deleted, or added. Their combination of these features produces the final result. Each participant's experience can be seen in Figure 5.2.

The second section of the survey aims to collect data about the essential features required for developing an SPL. All these features were derived from the literature review presented in Chapter 2. Each feature identified in the review was categorized into four groups: aesthetics (visual aspects), mechanics, educational mechanics, and others. Additionally, to mitigate prejudices in the answers, all answers included in the questionnaire were balanced and categorized as required and unnecessary, with the options presented randomly to prevent automatic responses. The characteristics identified as "not necessary" were expressed to avoid explicitly including "no" in the sentences to reduce the impact of biased replies. Furthermore, participants were questioned about the importance, motivation, challenges, and tools for developing an SPL for educational games. The number of answers for each of the characteristics asked in the survey can be seen in Figure 5.3.

Participants predominantly cited the reduction in the time necessary to develop EGs as their primary motivation for implementing SPL. This was followed by the ability to create games with similar features that reduce the learning curve for students, enabling players who were previously habituated to the game or its mechanics to focus on the knowledge that the playful experience seeks to provide. Some participants pointed out that all games offer a didactic function, as they need to explain their mechanics and the context in which it is situated fundamentally. Furthermore, those involved identified benefits such as increased productivity, increased component reuse, reduced costs, and standardization, which promotes consistency among products developed within the same franchise. Finally, two of the participants believed that the use of an SPL for this objective was irrelevant. However, given the population's size, these instances may be considered outliers without affecting the general results.

In addition to motivation, participants were also asked about their prior knowl-

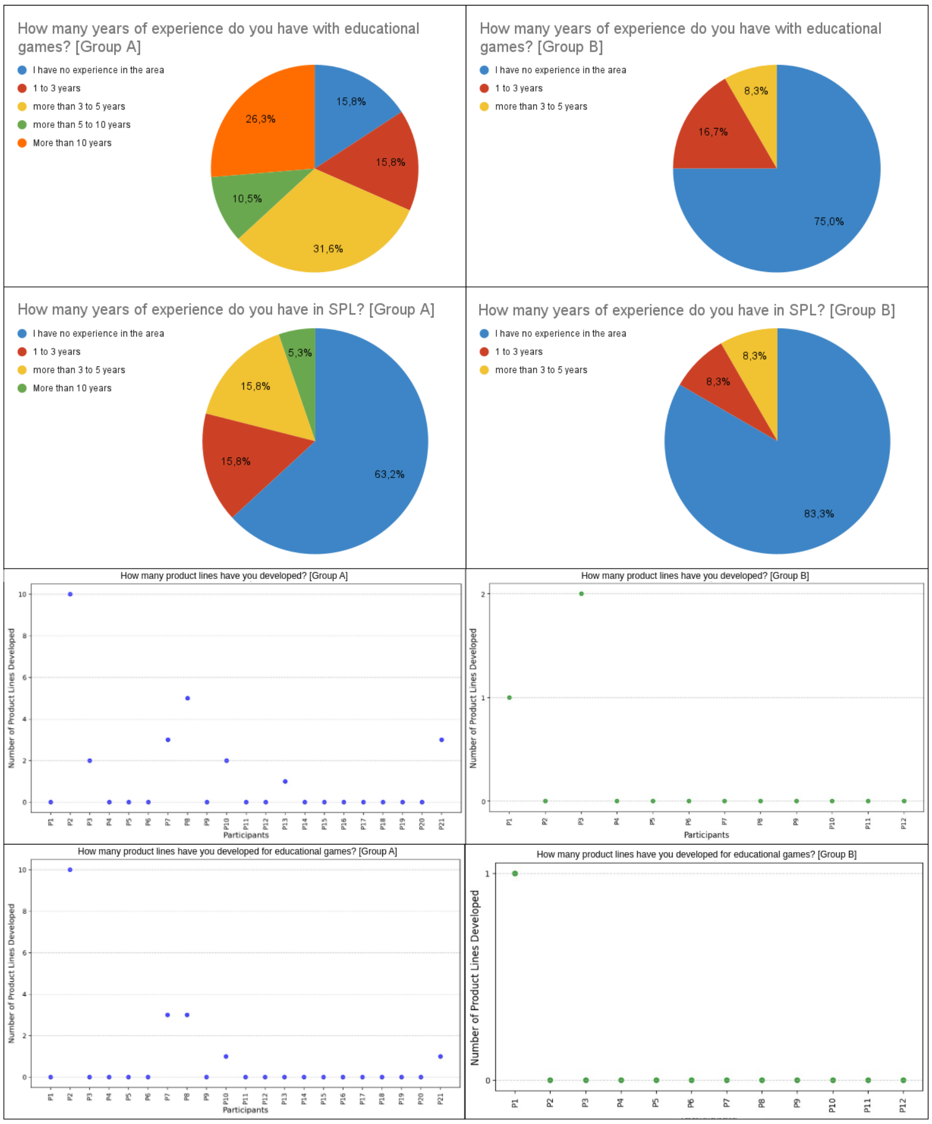


Figure 5.2: Participant's Experience.

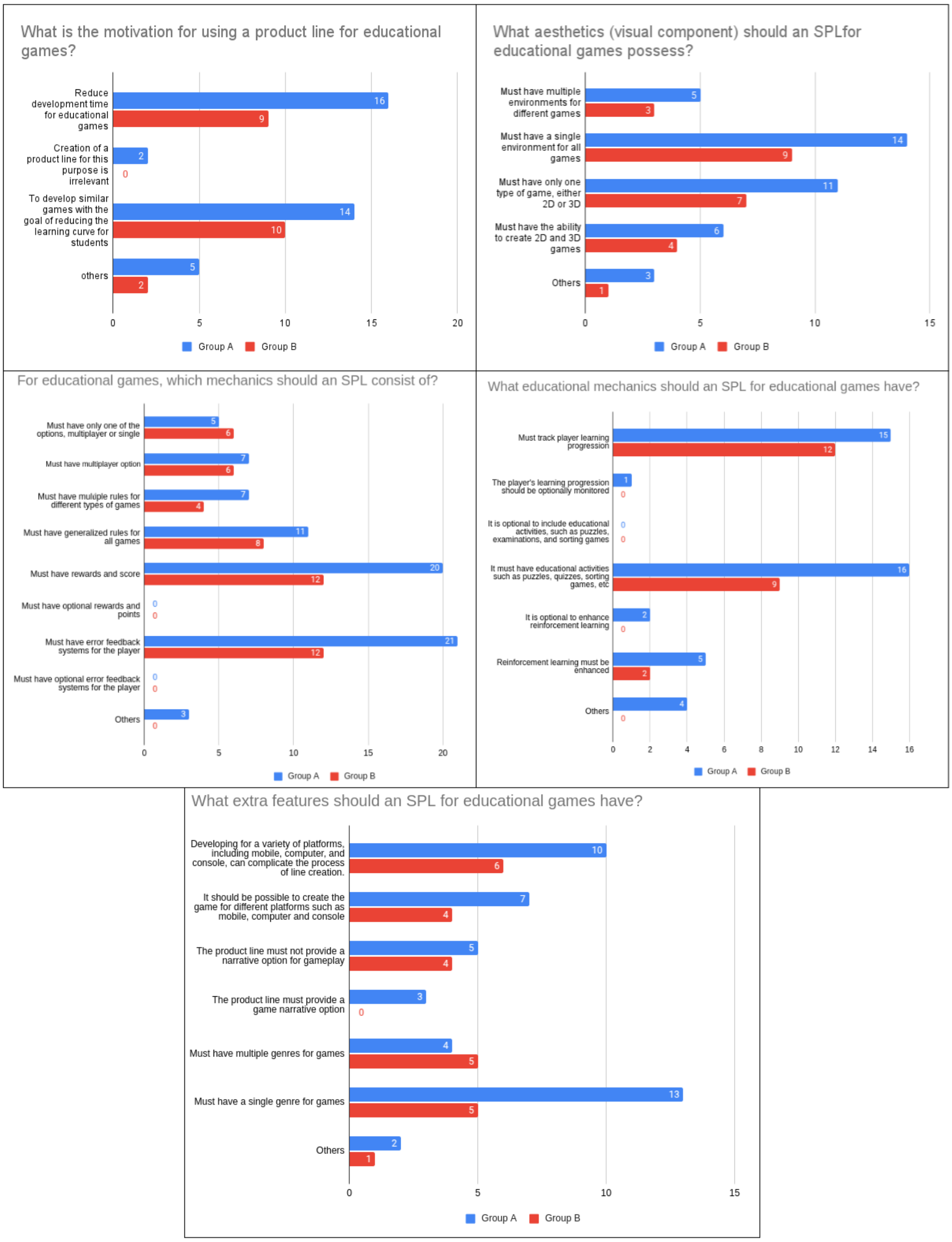


Figure 5.3: Questionnaire for the second part of the survey

edge of SPLs for developing EGs and the main obstacle in creating an SPL. Many challenges related to development were reported, particularly highlighting the complexity of the use case domain of the line since the field of EGs is perceived as extensive and varied, making it challenging to formulate specializable solutions. Finally, some participants reported SPLs aimed at creating EGs, such as the Coelho Sabido franchise, Askme, Reeborg, Laby, Educameeple, Arcade Game Maker, Move2Learn, and Kahoot.

Participants were asked about the characteristics that an SPL needed to develop EGs. The visual dimension showed the need for a single environment for all games, providing a unique experience for the students. This method significantly reduces the learning curve, as the player is already familiar with the game environment. This characteristic is strongly recommended since it substantially reduces development work, given that in an SPL, it is assumed that most features are generic and context-adaptable, where numerous participants emphasized the necessity of multiple SPLs for EGs, depending on the context of use, demonstrating the complexity of the SPL to be built. Another factor stresses using 2D or 3D settings, which differ based on educational needs. In this context, the SPL was advised to pick just one environment because of the substantial effort necessary to provide an SPL compatible with both formats. Thus, creating an SPL for 2D games and another for 3D games would be more effective, emphasizing the importance of many SPLs based on the application environment.

Concerning the mechanics, three factors were especially evident. At first, it was noted that it is necessary to include continuous feedback that explicitly explores correct and incorrect behaviors inside the game. Moreover, including awards was vital since they encouraged the user during the learning process, improving the entire experience. Finally, it was emphasized that adopting specialized rules that apply to all games is essential for guaranteeing consistency across all games. Nevertheless, formulating generally applicable rules that can be employed across several games can be extremely difficult. One participant addressed this issue by emphasizing the need to develop appropriately contextualized rules to accommodate the specificities of each situation.

In the domain of educational mechanics, it was emphasized that educational activities provide the main characteristic for facilitating the specialization of pedagogical knowledge. This methodology enables the creation of games with standard rules applicable across numerous circumstances, facilitating considerable flexibility in their use. Thus, instructors may modify learning dynamics to accommodate diverse requirements and circumstances, enhancing the teaching and learning experience. This suggested technique would enable the development of educational games designed to educate various courses and disciplines. This strategy would align with the specific requirements and interests of educators and teachers engaged in the teaching and learning process. Furthermore, tracking the player's learning progression was considered essential. This technique has become crucial to ensure the effectiveness of the learning process and the achievement of the specified objectives. This feature can demonstrate how players understand the instructional objectives incorporated in the game, showing their difficulties through the actions performed throughout the various matches. This approach allows modifications and improvements in the teaching-learning process, thus promoting a more effective advancement of the educational activities associated with the game. Lastly, the issue of reinforcement learning remained unanswered, as it did not receive many answers. However, a relevant comment was made regarding the potential for this characteristic to reduce the player's motivation, making the game a repetitive task.

Concerning additional features for SPL, almost all participants preferred using a singular genre for all SPL-developed games, emphasizing the necessity to create different SPLs customized to other contexts of use. Furthermore, it was determined that the SPL must enable the development of games across several platforms, hence maximizing the product's distribution. Nevertheless, several participants mentioned the developing process. When a game is created with the environment in which it will be executed in mind, the characteristics of the environment are better explored; leaving these aspects generic can highlight usability problems for the end player.

The findings analysis shows that, despite significant obstacles, SPLs focused on EGs have massive potential for promoting substantial transformation in this sector's development. This change may provide considerable advancements and improvements, benefiting the educational sector. The standardization of procedures, significant reduction of operating costs, and subsequent improvement in operational efficiency were recognized as benefits. Nevertheless, despite these benefits, the execution of SPLs continues to be a very complex and challenging task. This mainly arises from the need to develop several product lines, each customized for specific environments and diverse application contexts. To solve implementation issues and enhance the use of educational activities, a potentially advantageous approach would be to mitigate these obstacles.

5.1 Discussion

Fourteen hypotheses were developed based on the answers to the research questions. Three experts with more than a decade of expertise in educational games were interviewed to verify these hypotheses. This section will examine each hypothesis in further depth.

The first two quantitative questions investigated the factors influencing developers' decisions to develop an SPL for EGs. The results demonstrated that both highlighted alternatives are essential for progressing the line, encouraging hypotheses 1 and 2.

Following an expert investigation, both hypotheses were confirmed as valid. The first was emphasized as a general advantage of the SPL, but the second was recognized as a direct benefit for education. The SPL Reader Rabbit (BAYTAK and LAND, 2010) was referenced, wherein the characters are used for teaching several themes in early childhood education. This type of SPL allows children to remember the characters and the environment, facilitating and promoting learning through previous knowledge, meaning the child only needs to focus on the subjects being taught.

> Product lines are created to reduce the development time of educational games.

Hypothesis 1.

> Product lines are created to develop similar games to decrease the learning curve for students.

Hypothesis 2.

The two subsequent hypotheses were not formulated using quantitative analysis of the questionnaire answers. They were formulated based on the study of answers to open questions in the questionnaire, which sought to understand the difficulty of developing an SPL for EGs. After analyzing these responses, hypotheses 3 and 4 were formulated.

Hypothesis 3 postulated that the main challenge in game development is to find the creative thinking required to create them. The challenge for EGs increases as it requires converting learning into an entertaining experience. Developing a serious game is challenging; transferring this concept to a generic format involves even more complexities. Consequently, the theory has not been fully validated, as the issue is not only technological; nevertheless, technical challenges persist. Hypothesis 4 adheres to this thinking, positing several possibilities for developing serious games involving many mechanisms. Nonetheless, the complexity of development increases when trying to specialize this process, as the level of abstraction necessitates the identification of mechanics that can be applied to any subject to be taught.

The challenge of creating an SPL for educational games is inherently associated with technical complexities since it represents a project characterized by a significant degree of abstraction.

Hypothesis 3.

The initial question on the characteristics necessary for developing an SPL for EGs presented in the survey was about the need for multiple environments for different games or the creation of just one environment. Concerning the amount, eight respondents indicated a need for various habitats. At the same time, 23 responses favored the establishment of a single environment for all games in the line, considering the two populations included in the research. To facilitate the analysis of the results, the quantities mentioned from here on will be the totals of the responses from the two populations seen in Figure 5.2. Experts indicated that the decision to create one or multiple environments for games is conditioned on the project's requirements; nevertheless, they argue that for a generic SPL, it is advisable to concentrate on a single environment, considering the complexities associated with integrating multiple environments. Therefore, hypothesis 5 was validated.

The domain of educational games is broad and abstract, making it extremely difficult to establish generic characteristics that can cover all educational aspects.

Hypothesis 4.

An SPL for educational games must have a single environment for all games to facilitate their development.

Hypothesis 5.

The survey indicated that 10 respondents argued for including 2D and 3D environments in the game's graphic dimensions, while 18 preferred focusing on a single dimension. Hypothesis 6 was evaluated based on the number of responses. The experts pointed out that the dimension of the game can affect the player's motivation, thus improving their engagement time, when they were questioned about hypothesis 6. Nevertheless, this variable is irrelevant in the context of the quality of learning. Whether the game is in 3D or has high-resolution graphics, the determining factor is its consistency with the educational purpose. Furthermore, they emphasized the existence of 2D games with relatively straightforward visuals that still attract millions of players. Due to this, they advised selecting only one dimension of games for the SPL, thereby validating hypothesis 6.

An SPL for EGs should not be focused on 3D or 2D; the environment's interface is not directly linked to the quality of the game's teaching; therefore, it would be indifferent.

Hypothesis 6.

The quantitative results indicated a similar choice for the multiplayer feature in the SPL, with 11 respondents choosing either single or multiplayer inclusion and 13 respondents expressing a specific need for a multiplayer feature. Consequently, an additional hypothesis was formulated, highlighting the importance of this feature. Experts consulted for this hypothesis indicated that the multiplayer feature is extensively utilized in educational games since it promotes player collaboration and improves the game's educational value, facilitating knowledge exchange among players. This feature depends on the area in which the game will be utilized. Two students can collaborate if intended for classroom use, eliminating the necessity for a multiplayer mode. Therefore, it would be an extra feature, not highly mandatory. Consequently, the relevance of this characteristic for developing an SPL is reduced, thereby refuting the initial idea.

An SPL for educational games must have a multiplayer option so that it is possible to create interaction between players who are learning.

Hypothesis 7.

Concerning game rules, 11 participants indicated that each game must have distinct rules, with the SPL providing an extensive selection of rules for developers to select from. Nineteen participants favored the implementation of general rules applicable to all games. In response to hypothesis 8, the experts argued that, similar to hypothesis 2, games require substantial originality from their developers, rendering them distinctive and engaging for players. However, to enhance the potential for abstraction, it was anticipated that the rules would be specialized, permitting increased practicality and reduced development time for the game developer.

An SPL for educational games must have specialized rules for all games, given the vast number of games it must develop.

Hypothesis 8.

Reward mechanisms are extensively employed in both educational and non-educational games. They are a conventional element that encourages players by affirming correct actions. All survey participants agreed that including awards in the SPL is necessary due to its widespread use. Additionally, feedback is an essential component of educational games. Consequently, all survey participants considered it necessary. When asked about its importance, experts unanimously stated that every educational game should include feedback, referencing Marc Prensky's well-known quote: 'Games have outcomes and feedback. That gives us learning' (PRENSKY, 2001) and validating hypothesis 9.

> An SPL for educational games must provide error feedback mechanisms enabling players to identify mistakes.

Hypothesis 9.

Feedback is essential for students to comprehend their errors, and similarly, tracking a player's learning growth is crucial. The survey aimed to assess the necessity of this element in the SPL; only one respondent indicated that it was unnecessary. The specialists consulted for hypothesis 10 stated that every educational game must provide this feature, although rare. Furthermore, they argued that an educational game needs to serve not merely as an instrument for transmitting content but also provide tools for assessing player performance on specific tasks, thereby facilitating instruction on optimized task execution. They finally commented on the GameAnalytics (GAMEANALYTICS, 2025) library designed to monitor all player activities during a match.

> An SPL for educational games should track the player's learning progression.

Hypothesis 10.

In line with the principles of feedback and knowledge progress, the survey assessed the necessity for reinforcement learning attributes, interpreted as new challenges presented to the learner upon failure of a prior challenge (LITTMAN, 2015). Only 7 participants considered this feature essential, whereas 2 opposed it. Only 7 participants considered this feature. In addition, 23 participants neither designated it mandatory nor expressed opposition to it. Upon inquiry on the feature, the experts indicated that it frequently enhances involvement in the game; however, it necessitates precise calibration rather than a mere trial-and-error approach. Regarding its need in the SPL, experts did not consider it a mandatory feature but rather an optional one.

> An SPL for educational games must have characteristics that strengthen knowledge through errors.

Hypothesis 11.

A few SPLs for games have been identified in the literature, each concentrating on a singular genre. Consequently, the survey explored the possibility of creating either a single genre or multiple genres for the SPL; 9 respondents supported the development of various genres, whereas 19 favored the creation of a single genre. Hypothesis 12 was formulated based on the quantitative data. When questioned regarding this hypothesis, the experts claimed that it was essential to concentrate on a singular game genre; if an alternative genre was required, it necessitated the creation of a separate SPL. This is mainly attributable to the challenges of implementation and the potential for varying rules and mechanics across different game genres. Consequently, the SPL has to adapt to a distinct niche of regulations, mechanics, and applications; should this niche evolve, an alternative SPL will be required.

> An SPL for educational games should focus on a single game genre due to the difficulty of implementation.

Hypothesis 12.

The literature indicates that a significant challenge in creating the SPL is specializing game mechanics while considering the creative dimension of games. Another aspect to contemplate is the specialty of educational mechanics, enabling the application of the same mechanics to any subject intended for instruction. The research indicated that one potential avenue for this specialization was implementing educational activities, defined as interactive learning tasks designed to enhance student engagement through various challenges (GEORGIEV *et al.*, 2016). Given this, the survey also sought to identify the need to use these activities within the SPL. 27 participants responded favorably to educational activities, while only one was against them. The experts questioned hypothesis 13 and reported that using educational activities in EGs is frequent, providing mini-exercises to strengthen the student's knowledge and validate the hypothesis.

> The use of educational activities can enable the specialization of teaching mechanics.

Hypothesis 13.

The respondents were ultimately queried about the necessity of launching the game on several platforms, including PCs and mobile devices. The final hypothesis arose from the results indicating that 16 participants favored developing games for several execution environments, while 11 focused solely on a single creation environment. Upon inquiry regarding this viewpoint, the experts expressed dissent, asserting that games foster a creative environment that ought to enhance the distinctive attributes of each locale, thereby seeking to render the experience more engaging. SPL could forfeit the opportunity to investigate any platform's potential by permitting development for several types of devices.

An SPL for educational games must have the possibility of being cross-platform built to serve more students.

Hypothesis 14.

5.2 Final Consideration

The development of EGs is seen as a challenging and labor-intensive endeavor that necessitates both educational and technological expertise. A teacher's everyday responsibilities leave little time to acquire new technology skills and develop games for classroom usage.

SPL seeks to develop analogous products characterized by a common set of features, positioning itself as a potential solution to the problem described so far. The study conducted with game developers and experts in the field of EGs yielded 14 hypotheses regarding the essential elements for the development of this SPL. After validating the hypotheses with experts, the results indicate that the area continues to exhibit gaps that might be enhanced through the benefits of utilizing SPL, including the specialization of mechanics and the creation of games with analogous rules. Nonetheless, although SPLs can enhance the efficacy of EG development, it remains difficult to balance creativity with educational requirements.

A significant concern emphasized in the interviews and surveys was the necessity to develop many SPLs, depending on the type and context in which the EVGs would be inserted. Consequently, one can observe the complex structure of the line and the multitude of characteristics it must have. So, it was determined to create a GM to manage this large variability of characteristics that this SPL must support.

Chapter 6

Converting FM to GM

The main objective of this chapter is to present an approach for converting FM to a GM, introducing several rules that must be followed for this conversion to be possible. This conversion is necessary to facilitate the transition from a FM to a class diagram, as FMs merely provide abstractions of the requisite characteristics of software, rendering the creation of a class diagram a complicated task.

6.1 Levels of abstraction

As described in Chapter 2, it is very challenging to identify a notation that is adequately comprehensive to encapsulate the variabilities of an SPL, particularly for complex software. This representation is crucial as it facilitates a deeper comprehension of the features that the SPL can produce, elucidating the aspects that may lead to distinct and specialized applications or products. These models are representations with a high level of abstraction and have some limitations, mainly regarding the form of development, lacking information such as composition, aggregation, specialization, interfaces, and association.

Such limitations may yield a shallow depiction of variability, culminating in a flawed understanding of the system family that will be subsequently reused. Consequently, it is essential to incorporate lower-level models that provide a clear and straightforward description of the SPL while simultaneously conveying significant information regarding its development. However, this model still needs some abstraction, given that the SPL should possess sufficient flexibility to facilitate the generation of many versions.

SPLs generally describe their requirements via a FM. From this FM, the essential features of the software to be created are selected, and implementations are carried out, often guided by a class diagram (MAAZOUN *et al.*, 2014; SILVA, 2016). However, the level of abstraction depicted in a FM and in a class diagram are very different, requiring some prior steps, meaning that the evolution of the SPL occurs through stages. A potential method that may serve as an intermediary between these two representations is a GM (SILVA, 2016).

GMs can be understood as a high-level abstraction from which models are derived. They can be represented using UML (Unified Modeling Language) (OBJECT MANAGEMENT GROUP, 2017), being diagrammed as a class diagram illustrating entities, properties, and associations (CAVALCANTI *et al.*, 2011), however only at an abstract and not concrete level. After instantiating a GM, models are generated, thus creating concrete class diagrams available for implementation. This type of modeling has already been used to represent SPL variability, demonstrating good results (CAVALCANTI *et al.*, 2011; CHEN *et al.*, 2009; GARCÉS *et al.*, 2007; SANTANA *et al.*, 2009; SILVA, 2016). Figure 6.1 demonstrates the abstraction levels for developing an SPL.

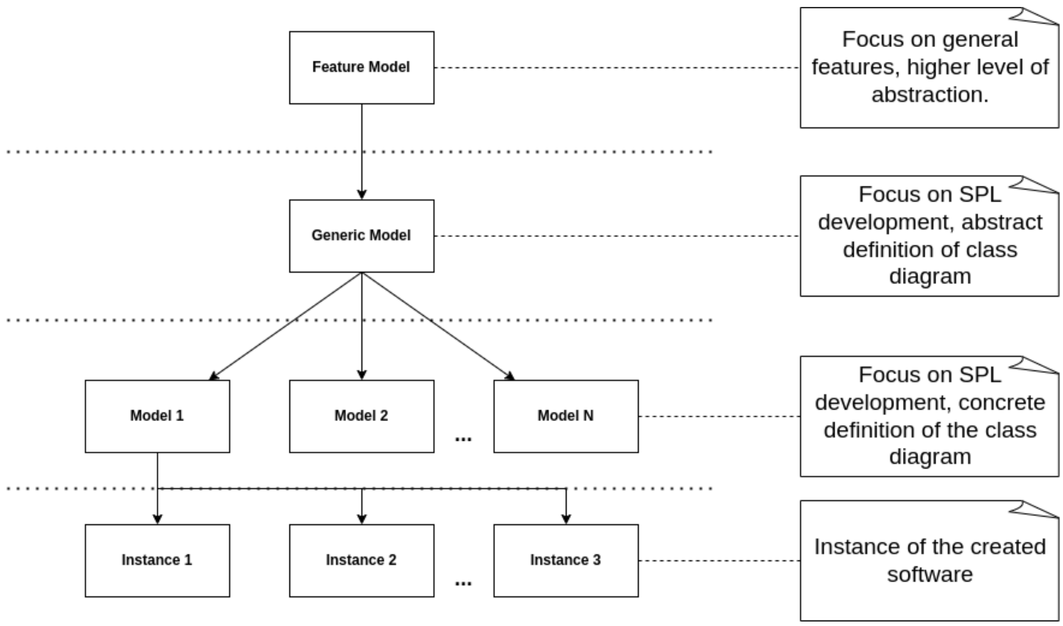


Figure 6.1: Abstraction levels for representing an SPL, adapted from (MAIA, 2024) and (DE OLIVEIRA, 2006)

From the figure, it is possible to observe that FM is at the highest level of abstraction to demonstrate the characteristics of an SPL. From this, GMs can be generated representing a class diagram with only abstract objects. From the definition of these abstraction objects, models are generated that in turn can give rise to several instances, this being the final application.

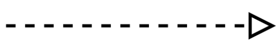
UML establishes a GM for each of the models represented by it, which is described by means of a class diagram (OBJECT MANAGEMENT GROUP, 2017). However, there is no universal standard for FM, with FODA being the most widely used. Because of this, UML does not have a defined model to represent a GM from an SPL (DE OLIVEIRA, 2006). A FM merely delineates high-level properties and lacks information regarding the development of the SPL. Thus, the GM will function as a diagram constructed post-FM development, depicting the interactions among characteristics to be integrated into the SPL.

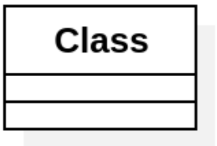
6.2 GM Notation

Odyssey-FEX (Feature EXtended) can be understood as a notation that attempts to add some of the objects of the class diagram within FMs. It is represented by a GM through class diagrams, where each class represents an element present in the FM or a relationship between these elements. The content is segmented into three diagram packages: a taxonomy illustrating the FM features, a diagram depicting the relationships that delineate the properties among the features in a feature model, and a diagram outlining the composition rules that define the dependency and exclusivity regulations between the model's features (DE OLIVEIRA, 2006).

The main objective of the notation is to formalize the existing concepts in a feature model, reducing the complexity of creating them by defining guidelines that assist the developer (DE OLIVEIRA, 2006). Therefore, the notation was designed to represent a level of abstraction above the FMs level. However, a GM represents a development-level modeling, unlike FM which only characterizes high-level features of a line. With this in mind, some researchers use GMs to represent a development-level SPL, functioning as an intermediate modeling between FM and class diagrams (SILVA, 2016). Considering this, a new version of the Odyssey-FEX notation is presented, modified to convey information regarding an SPL at the development stage. Moreover, Odyssey-FEX seeks to delineate functional, technological, and conceptual characteristics, whereas the current GM focuses only on functional and conceptual characteristics. Table 6.1 illustrates all components of the GM.

Table 6.1: Description of notation elements, adapted from MAIA (2024) and DE OLIVEIRA (2006).

Element	Description
	Association relationship indicates a link between two classes. (OBJECT MANAGEMENT GROUP, 2017).
	A dependency relationship signifies a weak connection, indicating that one class utilizes another without establishing a direct association (OBJECT MANAGEMENT GROUP, 2017).

	Type of association. Composition relationship indicates that one class is contained within another. Subelements do not exist without a parent element (OBJECT MANAGEMENT GROUP, 2017).
	Special type of composition. Aggregation relationship, indicates that a class is composed of one or more elements. The subelement of the aggregation can exist without the parent element (OBJECT MANAGEMENT GROUP, 2017).
	specialization/specialization relationship, indicates a superclass (specialized) or a subclass (specialized). All attributes and methods are inherited by the superclass (OBJECT MANAGEMENT GROUP, 2017).
	Class element, indicates a characteristic within the SPL OBJECT MANAGEMENT GROUP (2017). From it, it is possible to observe the methods and attributes that such an element must possess.
«Mandatory»	Stereotype created to explain the mandatory elements that come from the SPL. The lack of this stereotype describes an optional class.
Object Constraint Language (OCL)	A language employed to delineate limits and rules of business within UML models. It serves to articulate circumstances that cannot be expressed just through UML diagrams OBJECT MANAGEMENT GROUP (2017).

From Table 6.1, it is possible to observe the main components of the GM to be generated, demonstrating several elements that are used in class diagrams but are not used in an FM. Consequently, it is essential to outline transformation rules to convert the elements of the FODA diagram to a GM (DE OLIVEIRA, 2006). The initial rule indicates that every FM characteristic must be converted into a class or function to be incorporated into the class representing its parent characteristic. It is essential to categorize each attribute as either functional or conceptual previously:

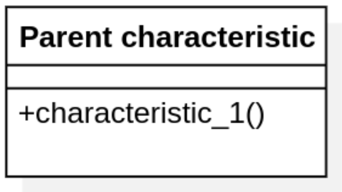
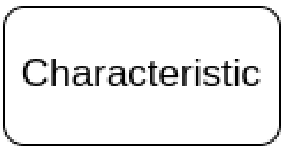
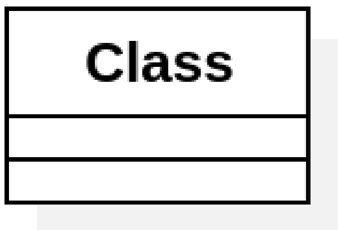
- **Functional characteristics:** represent single functionalities. These should be represented as methods within the class associated with the parent feature.

- **Conceptual characteristics:** represent more complex behaviors that encompass several functionalities or actions. They must be mapped to new classes

.

While the tree's root is not a characteristic of the SPL, it represents the main class of the software to be developed; hence, it must be converted into a class that possesses characteristics corresponding to each feature at the second level of the tree. The conversion of the tree to the GM should be examined by levels; so, if a tree comprises 5 levels, there must be 5 transformation steps, with each rule requiring analysis. It is essential to note that every node in the tree possessing child nodes or leaves must be converted into a class. Table 6.2 demonstrates these transformations.

Table 6.2: [Rule 1] Transformation of characteristics into classes or methods.

FODA Element	**Generic Model Element**
Transform functional characteristics into methods in the parent feature that will become a class.	
	
Transform conceptual characteristics into classes.	
	

Upon examining the initial two levels of the tree, it is possible to identify relationships without VP, indicative of simple relationships, which should be represented via simple associations in the GM and with an attribute in the parent node or characteristic represented by a class. It is worth remembering that these associations also occur at other levels of the tree. Table 6.3 demonstrates these transformations.

In the original Odyssey-FEX, it was advised to establish an interface or a class when facing a VP. However, a GM must be as generic as possible, not taking into account particularities of any language. With this in mind, each of the alternative VPs must be transformed into a specialization or a class with two methods, taking into account whether the characteristic is functional or conceptual. Nonetheless, an alternate VP indicates that only one of the child characteristics may be selected; hence, a logical XOR operation was assigned to the class methods, ensuring that only one method could be implemented. In the same way, the Singleton design pattern was incorporated into specialization, ensuring that only a single instance of a class could be created. This pattern aims to ensure that only a single instance of a given class can be instantiated during the entire execution of the program (HUNT and HUNT, 2013). To implement it, it is necessary to make the class constructor private; this prevents other parts of the code from instantiating the class directly. A static variable is subsequently instantiated within the class to retain the unique instance, and a public static method is implemented to verify whether this instance has been generated; if not, the method creates and stores it; if it exists, it returns the existing instance (HUNT and HUNT, 2013). Table 6.4 demonstrates these transformations.

Table 6.3: [Rule 2] Transforming simple relationships into associations.

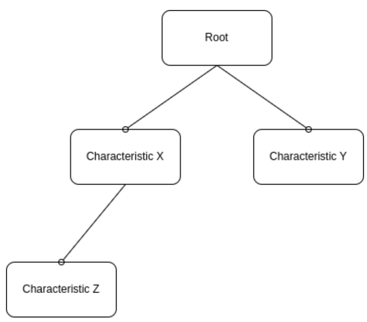
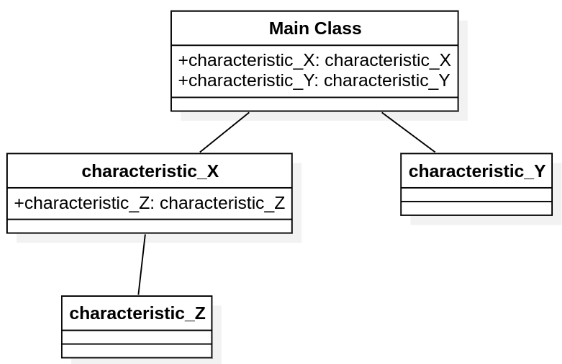
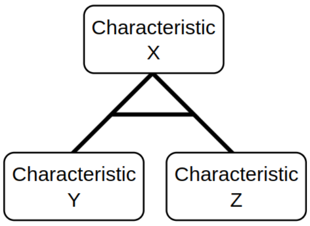
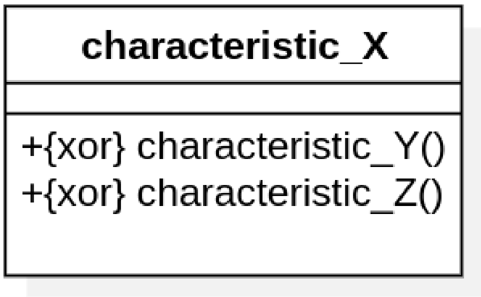
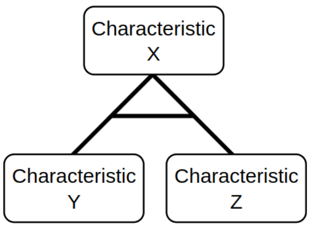
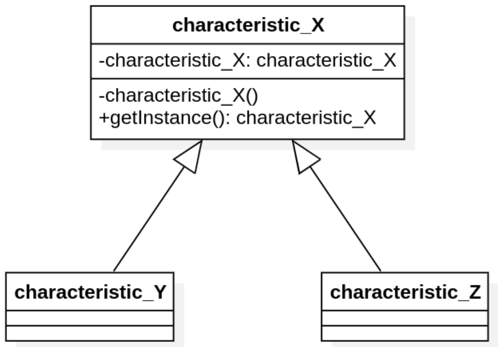
FODA Element	Generic Model Element
	

Table 6.4: [Rule 3] Transforming alternative VP into specialization/specialization.

<table>
<tr><th>FODA Element</th><th>Generic Model Element</th></tr>
<tr><td colspan="2">Transform alternative VPs with functional characteristics into classes with two methods.</td></tr>
<tr><td>


</td><td>


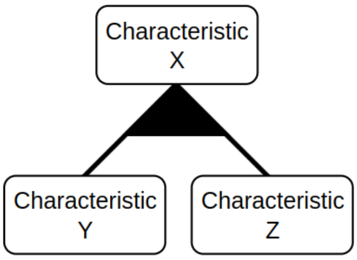
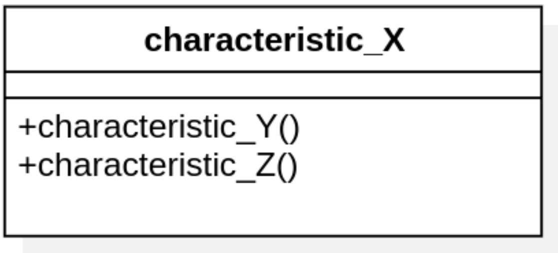
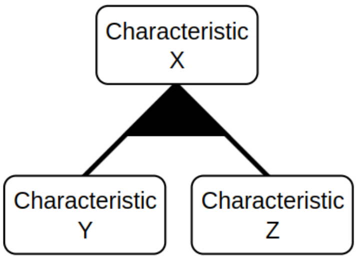
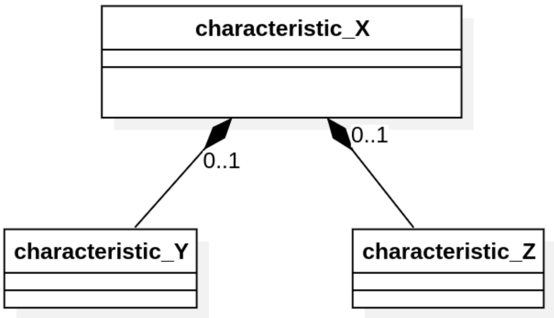
</td></tr>
<tr><td colspan="2">Transform alternative VPs with conceptual features into classes with specialization.</td></tr>
<tr><td>


</td><td>


</td></tr>
</table>

The optional VP was not delineated in the original notation and is explained in this thesis. Similar to the alternative VP, the decisive element for the applicable rule is the type of the feature, whether functional or conceptual. For conceptual characteristics, a composition must be established for each element that forms this relationship, indicating that the superclass comprises zero or more classes. Finally, the cardinalities must be delineated to classify the element as optional, employing the cardinality 0..1. To establish functional characteristics, a parent class must be instantiated containing all methods, analogous to the approach utilized in the alternative VP. Table 6.5 demonstrates this transformation.

Table 6.5: [Rule 4] Transforming optional VP into a composition relationship.

FODA Element	Generic Model Element
Transform optional VPs with functional characteristics into classes with two methods.	
	
Transform alternative VPs with conceptual features into classes with association.	
	

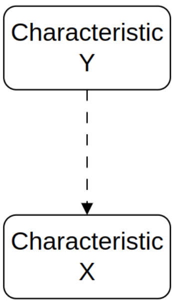
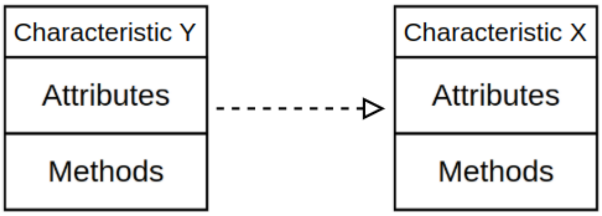
The original notation lacked a mapping to denote mandatory or optional characteristics, as this is not within the purview of the GM. The model instance is responsible for determining the elements utilized in the software construction, so ensuring the GM remains as generic as possible. However, to ensure a complete conversion of the FMs requirements called FODA (KANG, 2010), one stereotype was created, one to explain the mandatory elements of the SPL. Table 6.6 demonstrates the transformations for the mandatory element. The lack of a mandatory stereotype represents an optional class.

Table 6.6: [Rule 5] Transforming mandatory elements into mandatory stereotypes.

FODA Element	Generic Model Element
	«Mandatory»

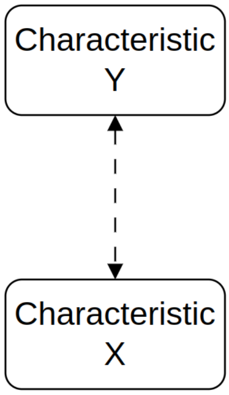
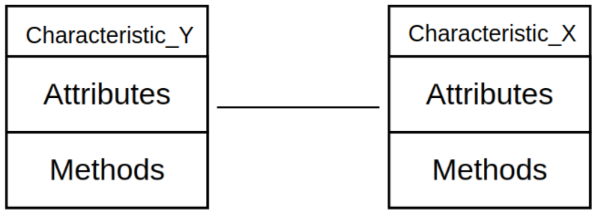
The original notation did not include the inclusion and exclusion relationships originating from FM FODA (KANG, 2010). The inclusion relationship indicates that the inclusive characteristic can only exist if the other included characteristic exists; that is, there is a relationship of dependencies between them, but there is no direct relationship. Therefore, every inclusion relationship must be transformed into a dependency relationship in the GM. Table 6.7 demonstrates this transformation.

Table 6.7: [Rule 6] Transforming an inclusion relationship into a dependency relationship.

FODA Element	Generic Model Element
	

An exclusion relationship indicates that the presence of one element excludes the existence of another. The class diagram lacks a visual representation for this type of constraint; hence, the Object Constraint Language (OCL) (OBJECT MANAGEMENT GROUP, 2017) will be employed to express this restriction. OCL is a language utilized to define constraints and business rules within UML models, articulating conditions that cannot be represented just through diagrams. Therefore, each exclusion representation will be demonstrated through an association combined with an OCL. Table 6.8 demonstrates this transformation.

Table 6.8: [Rule 7] Transforming an exclusion relationship into an association with OCL.

FODA Element	Generic Model Element
	 **context** Characteristic_Y **inv Exclusion:** Characteristic_Y.allInstances()->size() = 0 or Characteristic_X.allInstances()->size() = 0

Finally, it is important to emphasize that a GM can be specified according to the developer's preferences; thus, the rules shown assist the conversion of the FM into a GM, but it is the developer's responsibility to improve it with any additional rules, methods, or attributes he/she considers essential.

6.3 Notation Evaluation

The previous section demonstrated all the requisite steps to convert an FM into a GM, using inspiration from the Odyssey-FEX (DE OLIVEIRA, 2006) notation. The original notation aimed to formalize FM principles, simplifying their complexity, with a special focus on the formulation of the model formation rules. The study was evaluated via an observational study in which participants were provided with transformation rules and attempted to construct a GM from a FM (DE OLIVEIRA, 2006). Therefore, the present study will use the same evaluation standard to carry out a theoretical evaluation; the think-aloud (JÄÄSKELÄINEN, 2010) protocol was used to capture any feelings during the study.

For the evaluation, one FM was provided to each participant, accompanied by the previously described transformation rules; participants received 30 minutes to convert the FM into a GM. Upon task completion, participants addressed each evaluation question, which was designed according to the TAM questionnaire (DAVIS, 1993) to assess the acceptance and utility of the GM, using the Likert scale (AGUIAR *et al.*, 2011) to capture the degree of agreement or disagreement for each question. The instrument used in the study can be seen in more detail in Appendix F.

A pilot study was conducted with a master's student in the SPL research domain to evaluate the viability of the study's duration and to identify potential challenges or issues. Throughout the pilot, no issues were detected that would compromise the subsequent study. Feedback indicated that the transformations were coherent; however, it was suggested that a clearer explanation of the functional and conceptual characteristics be provided, as well as the inclusion of a stereotype to delineate the transformation of optional relations, in addition to mandatory ones. In response to the initial feedback, the evaluation form was revised to clarify the descriptions of functional and conceptual features, as illustrated in Appendix F. Nonetheless, concerning the optional stereotype, it was determined not to establish it, considering that the absence of a mandatory stereotype inherently categorizes it as optional; hence, it was not developed to avoid complicating the diagram.

Subsequent to the validation of the pilot study, an additional investigation was conducted with specialists to evaluate the coherence, simplicity, and usability of the transformation rules. This study comprised researchers in the field of SE with over 10 years of expertise who were previously involved with SPL and GMs. All participants possessed doctoral degrees and were over 50 years old, reflecting the high qualifications of the selected group. The group consisted of three individuals who, despite possessing substantial knowledge in SE, exhibited varying levels of expertise in the particular area of interest chosen for the study. Figure 6.2 illustrates this particular expertise.

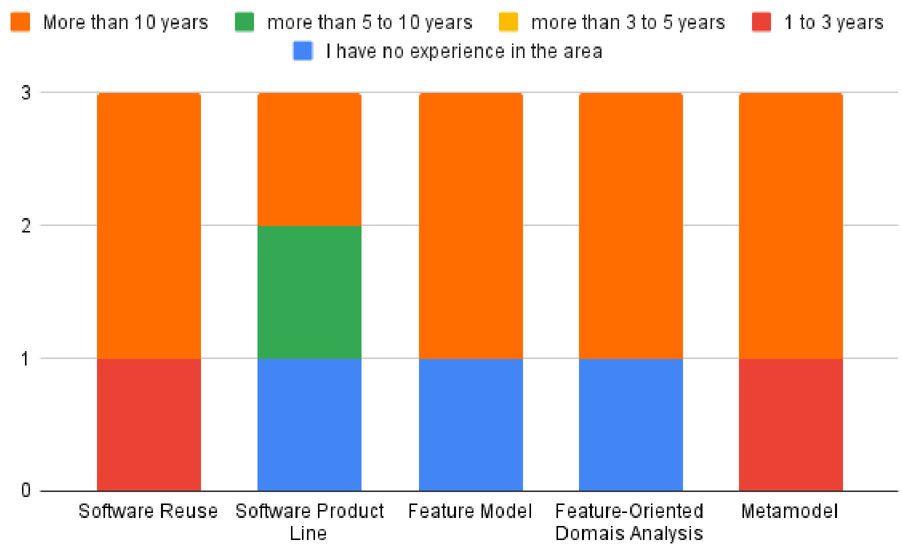


Figure 6.2: Participants' experiences.

The TAM questionnaire was provided to specialists post-activity to assess acceptance and utility. The comments indicate that the rules were simple to implement and the usage guidelines were clear effectively illustrating their utility in enhancing comprehension of the software development context, thereby achieving the intended objective of its construction. Figure 6.3 illustrates these findings. Alongside the positive results obtained from the TAM, the questions allowed collecting responses that indicated the GM approach was a beneficial effort, providing effective guidelines for the implementation of coherent class diagrams. Although positive findings were obtained, recommendations for enhancements were also collected, including the potential for incorporating additional rules to elaborate on the OCL declarative language, as it is not frequently utilized alongside UML.

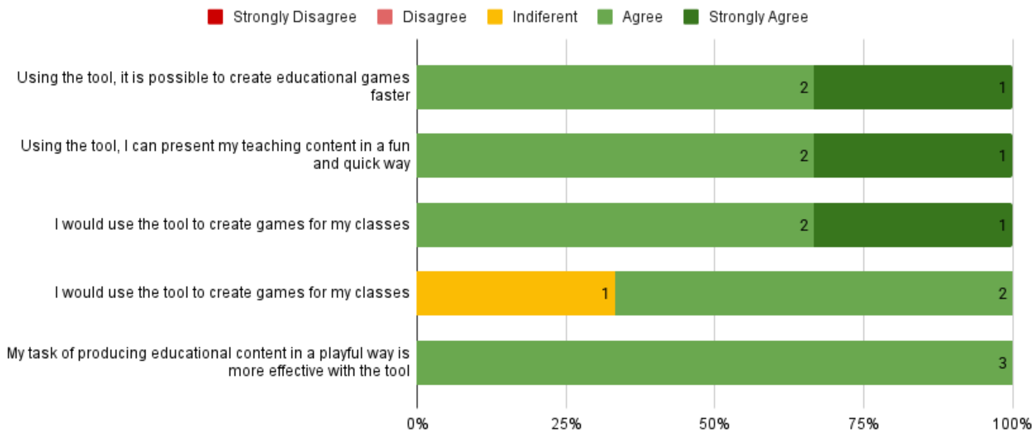


Figure 6.3: TAM Questionnaire Answers.

Finally, a question was made concerning rule 4, which pertains to composition (whole-part relationship) in UML. The formal definition states that the lifespan of "part" objects is contingent upon the longevity of the "whole" entity. A component cannot be instantiated independently; it must always be integrated within a larger system. The assertion was made that the multiplicity on the composition side must invariably be one, as the independent existence of the portion (suggested by a multiplicity of 0.1) contradicts the notion of essential dependency between part and whole.

However, the multiplicity 0..1 in a composition reduces the semantic weight of the relationship, as it suggests that the part can exist independently of the whole. This aligns more closely with the definition of aggregation, which denotes a weaker association where the part can exist autonomously.

Nevertheless, it was determined to maintain the composition within the model, even with a cardinality of 0..1, to illustrate a robust and exclusive relationship wherein the part cannot exist independently of the whole or be shared with other entities. In this instance, cardinality 0..1 signifies that the whole may or may not encompass the portion, but it does not imply that the component exists independently of the whole.

Chapter 7

Generic Model Development

The main purpose of this chapter is to present the transformation and the implementation of the FM of EGs into a GM so that it is possible to implement the SPLEG.

7.1 SPLEG Generic Model

Chapter 3 demonstrated the FM that was generated from the survey of the characteristics necessary for the creation of an SPLEG, being the first level of abstraction presented in Figure 7.1. Upon defining the FM, it can generate a GM that abstractly delineates the information necessary for product development, illustrating aspects such as composition, aggregation, specialization, interfaces, and association. Figure 7.1 demonstrates FM as the first level of abstraction.

Upon producing the FM, each of the aforementioned rules must be executed to convert it into a GM. The tree, containing 5 levels, necessitates 5 transformation steps, during which all rules must be applied at each step. At the initial level, only the tree's root is displayed, accompanied by several associated characteristics, thereby representing a conceptual attribute. Consequently, this node in the tree must be converted into a class, according to transformation rule 1. Figure 7.2 illustrates the initial phase of transformation, establishing the main class for the instance to be created, identified as game. The first level will always become a class, being a very simple transformation; however, this was done to illustrate.

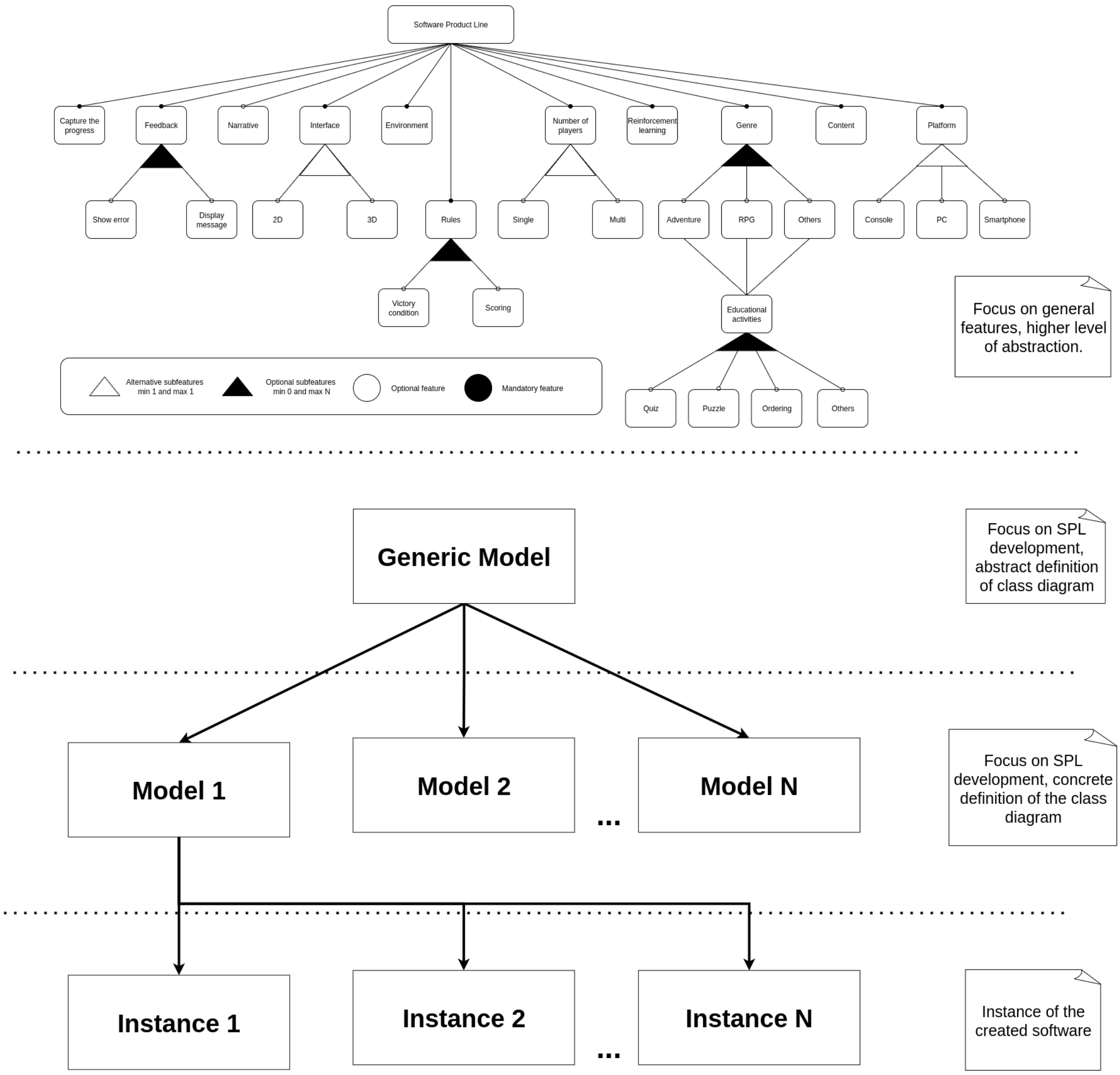


Figure 7.1: Levels of abstraction for SPL development. FM defined.

Figure 7.2: GM after step 1.

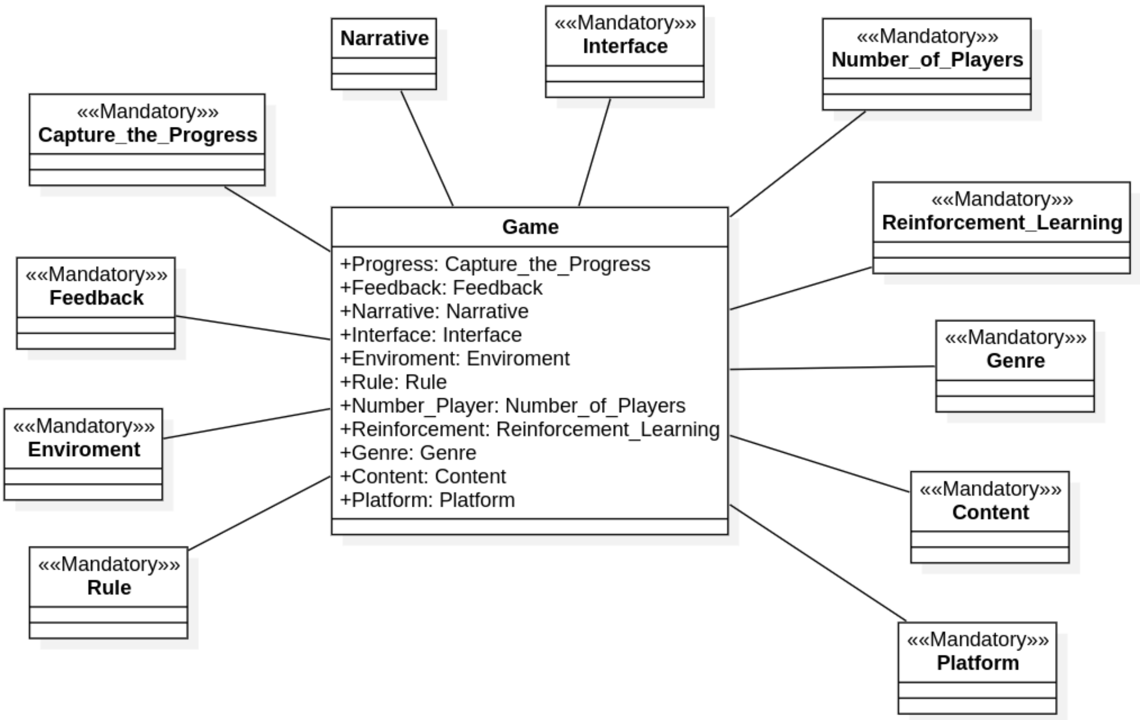


Figure 7.3: GM after step 2.

At the secondary level, every characteristic associated with the preceding level must be examined to determine whether they are conceptual or functional. At this level of the tree, only conceptual characteristics were identified, resulting to their conversion into classes. Furthermore, rules 2 and 5 must be applied to illustrate the classes that are required for an instance and to establish the relationships among the classes. Consequently, each requisite attribute was established within the game class to represent the corresponding classes associated with it. Figure 7.3 illustrates the GM subsequent to these modifications.

The application of rules 3 and 4 is necessary to discern alternative and optional variation points at the third level. It appears that the functional characteristics of "show error", "display message", "victory condition", and "scoring" can only be represented as a single responsibility to resolve. Nevertheless, the remaining characteristics may be interpreted as conceptual characteristics, which involve more complex implementation methods and responsibilities. Therefore, for these last characteristics, composition and specialization rules were applied, in addition to the use of the Singleton design pattern. Figure 7.4 illustrates the GM subsequent to these modifications.

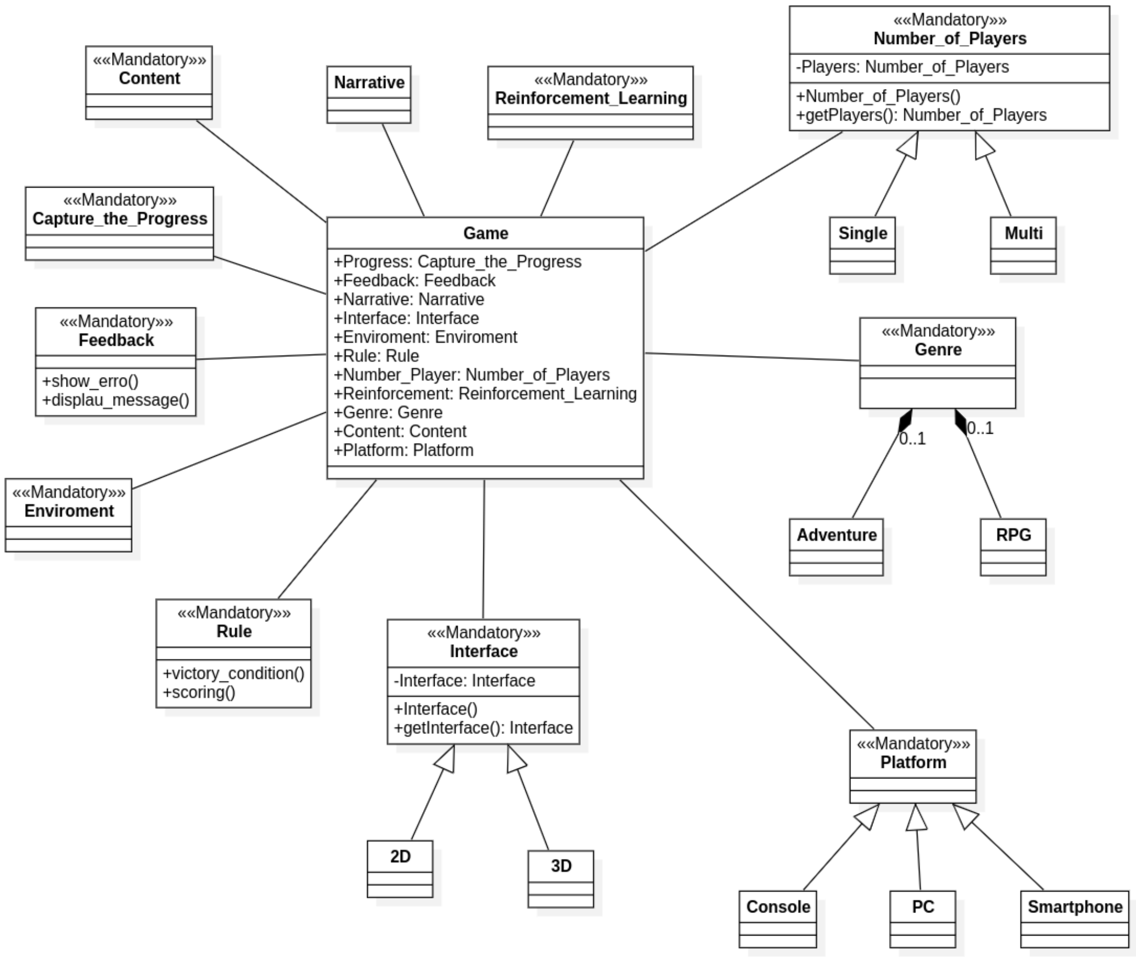


Figure 7.4: GM after step 3.

The only characteristic that is exhibited at level 4 of the tree is EA. It is important to note that this characteristic is associated with all gender characteristics; therefore, its relationship is direct with gender and not with its children. Additionally, this node has children and is therefore represented by a class, according to rules 1 and 2. Figure 7.5 illustrates the GM subsequent to these modifications.

At the final level of the tree, it is possible to identify the characteristics that are children of EA, represented by an optional VP and functional characteristics; thus, rule 4 will be applied to execute the transformation. Figure 7.6 shows the final version of the GM after applying the rules. A class diagram or GM can be as detailed as the designer desires; hence, it is the developer's responsibility to incorporate the rules he/she considers essential.

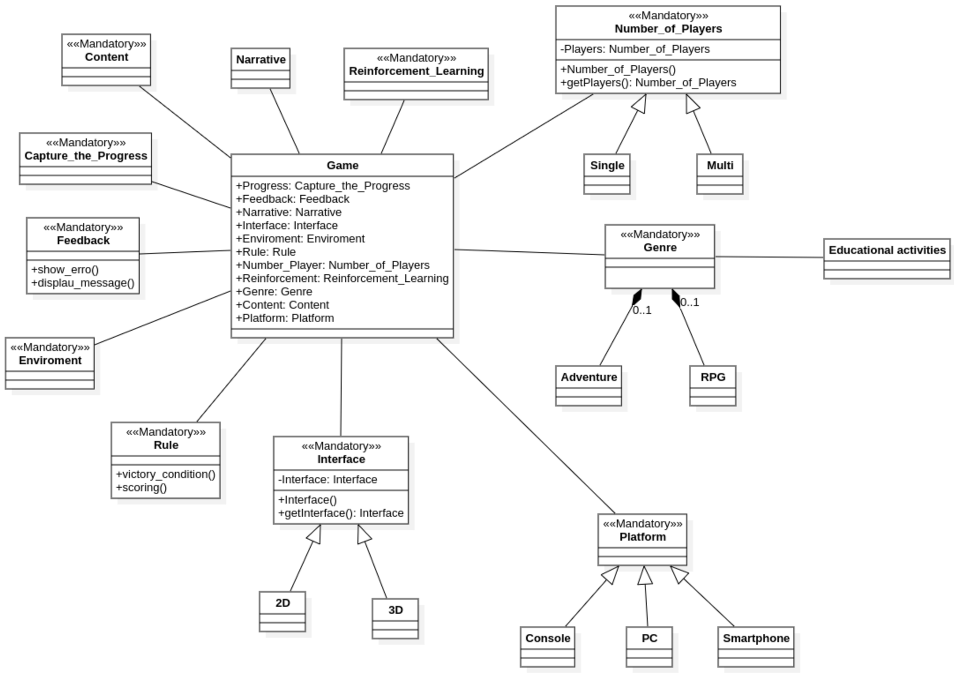


Figure 7.5: GM after step 4.

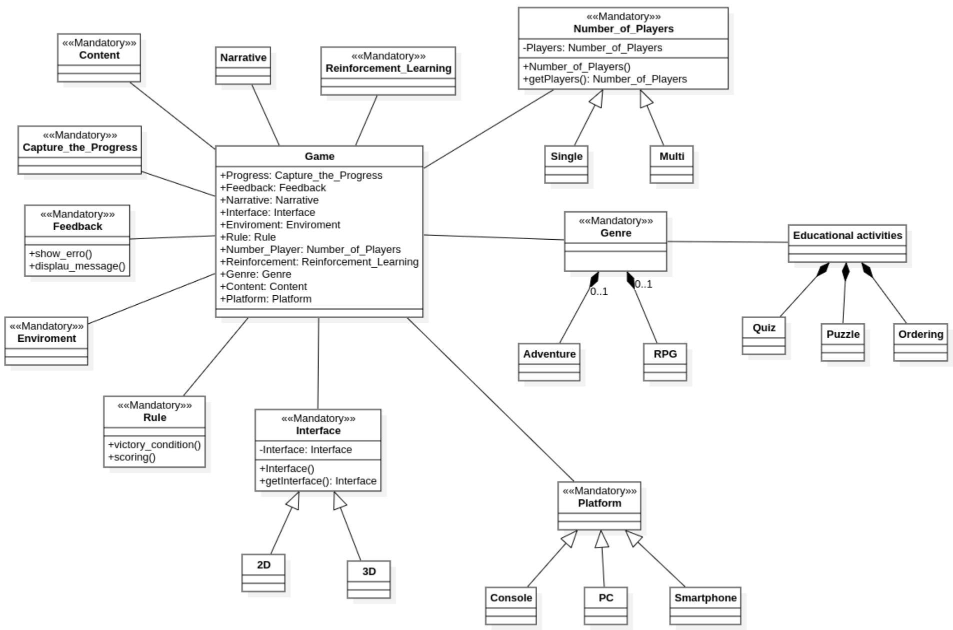


Figure 7.6: GM final version.

Upon constructing the GM, it is essential to select the attributes that will be utilized to produce the intended product. The selection of these characteristics was based on the hypotheses validated by the experts; thus, only those characteristics representing confirmed hypotheses were chosen. Figure 7.7 presents each of the transformation levels already discussed, represented by the characteristics tree, GM, and model.

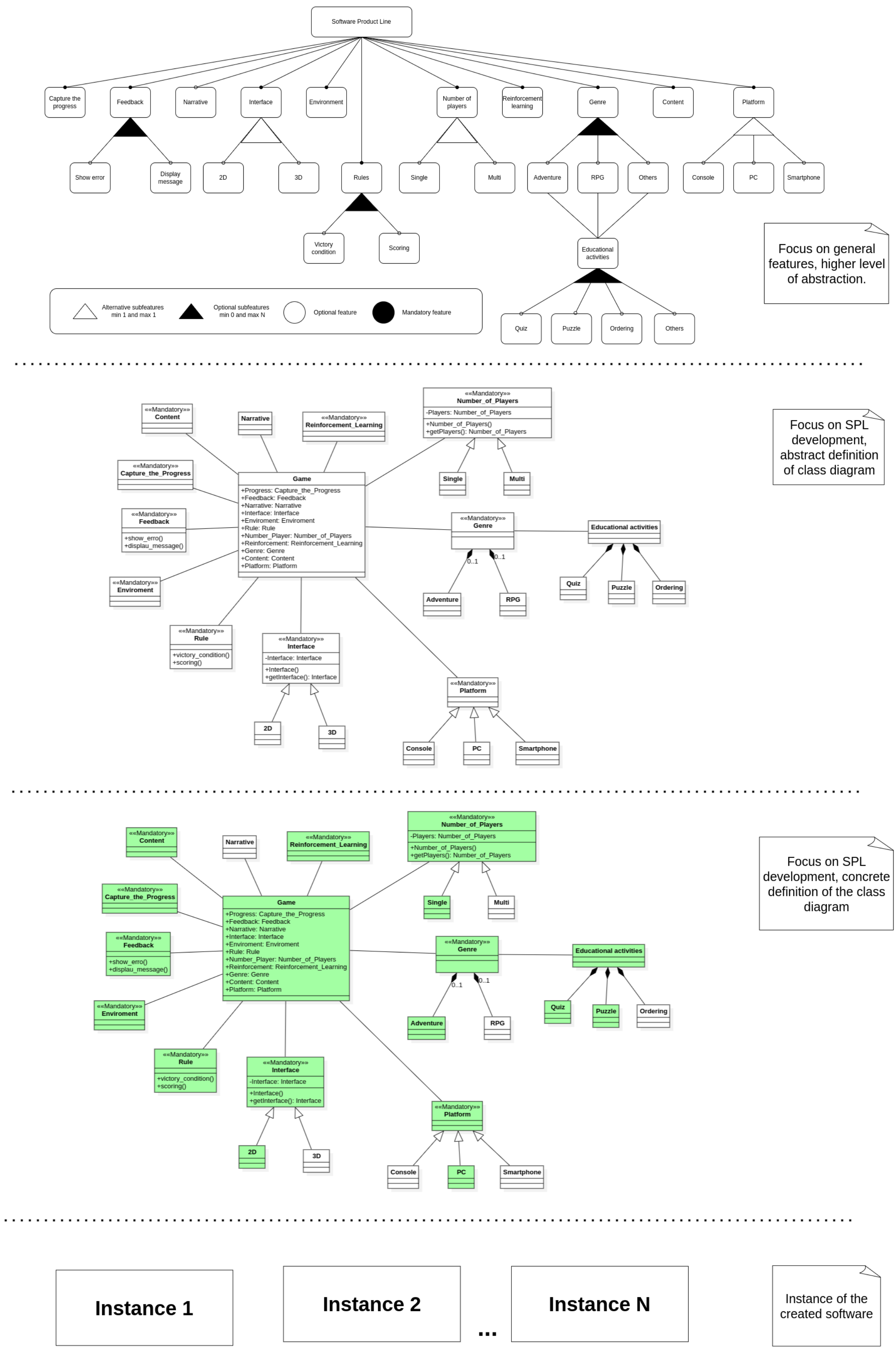


Figure 7.7: Feature Model, GM, and Model.

7.2 Model Instance

Upon establishing the transition rules from the feature tree to the GM, it became possible to develop a model as illustrated in the previous section, with the last step involving the instantiation of the model to produce a product. Therefore, a game was proposed to illustrate the implementation of an SPLEG.

The game was designed to enable the creation of educational games for classroom use, accessible even to educators lacking game development expertise; hence, it features a simple and intuitive user interface. In view of this, the feature tree was omitted from the teacher, as demonstrated in previous prototypes, and it was replaced with a simple form that required only the teacher to complete the content to be taught, and the game will be created based on this content.

The game's inspiration originated from a scientific initiation project that developed a game for teaching software requirements. The game mentioned was created using the Gather Town (RODRIGUES *et al.*, 2024) platform, structured in order that each floor represented a level, with activities and areas that the user had to interact with or explore in order to progress through the phase (RODRIGUES *et al.*, 2024). With this in mind, an instance inspired by the aforementioned game was created, called LEAP (Learning, Exploring And Progressing), and adhering to the features gathered from the survey.

The LEAP game takes place in a structure like a school, where players must engage in EAs to earn points and progress to the subsequent floor. The quantity of points necessary for progression, the number of EAs per level, and the content of each EA are entirely configurable by the instructor. Figure 7.8 shows the form that must be filled out by the teacher to create each floor of the game, having to add each of the questions and the number of NPCs for each phase. The image illustrates the four selected EAs to be incorporated into the game, along with supplementary configurations, including point allocation per floor and the assigned number of NPCs (non-playable characters) per floor, which inhabit the game's environment but are not controllable by the player. These NPCs were designed to represent educators, learners, and several other characters. Access to these screens is exclusively permitted with a password established by the teacher.

To offer educators optimal customization of the game and flexibility in configuring the floors regarding number and content, a database architecture was established, comprising distinct tables for each modifiable aspect inside the game. Consequently, each stage corresponds to a record in the stage table; each element within a stage, including plants and seats, constitutes a record with its corresponding Cartesian coordinates linked to the stage; additionally, each NPC possesses its own records. Consequently, the educator can ascertain the quantity of floors in the game, the number of NPCs in each level, the amount of questions presented at every stage, and the points necessary for advancement, all in an editable manner. To facilitate the instructor's use, a function was developed that enables the teacher to define simply the appropriate number of floors, while the environment is generated randomly.

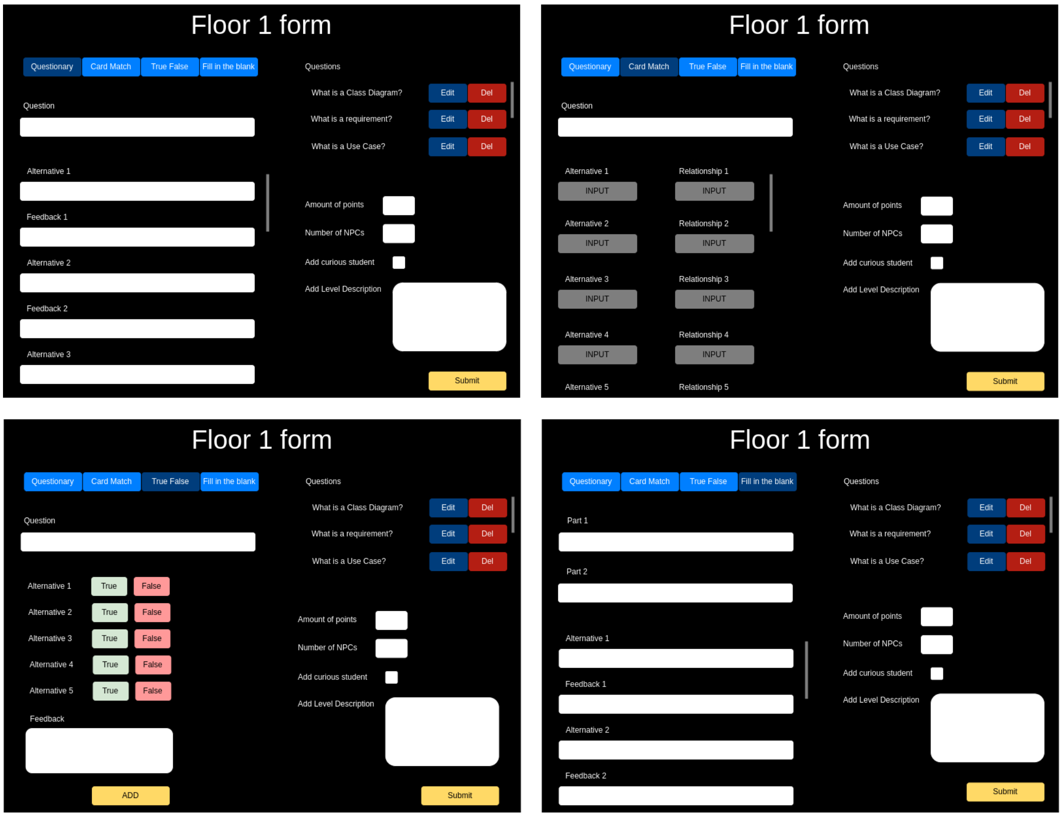


Figure 7.8: Floor Question Filling Form.

The student starts on the first floor and must interact with the characters to answer the EAs registered by the teacher to acquire the requisite points for elevator access, thus progressing in the game. Figures 7.9 and 7.10 show the game environment and the elevator in both obstructed and unobstructed states, permitting progression to the next level and showcasing the panel for level selection by the user. Considering feedback, each question generates a response dependent upon the player's selected answer, so enabling the teacher to clarify to the learner the rationale behind their correct or incorrect response. Furthermore, certain reinforcement learning elements were used, whereby a student received an additional point upon answering a question incorrectly, enabling their progression in the elevator. Figure 7.11 illustrates instances of EAs in the game and the feedback provided when a student answers a question incorrectly. It is important to remember that the order of every potential answer was randomized so that they did not appear in the same order; moreover, the questions that had already been answered in each phase were saved for each user so that no questions were duplicated.

Figure 7.9: Game environment.

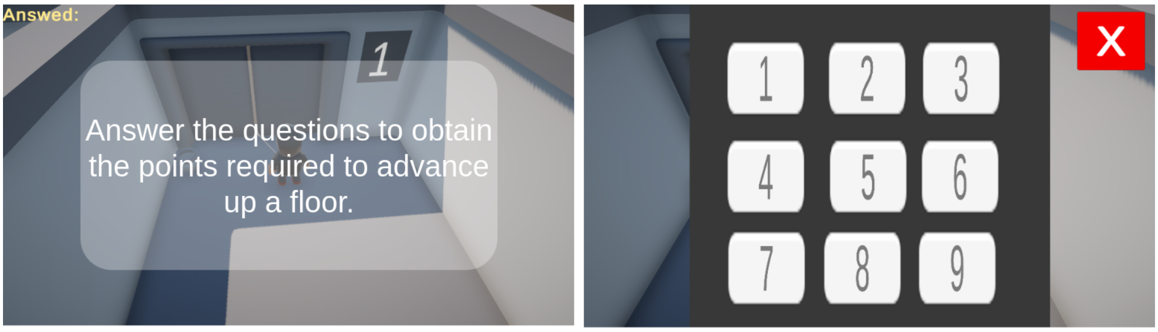


Figure 7.10: Locked and released states of the elevator.

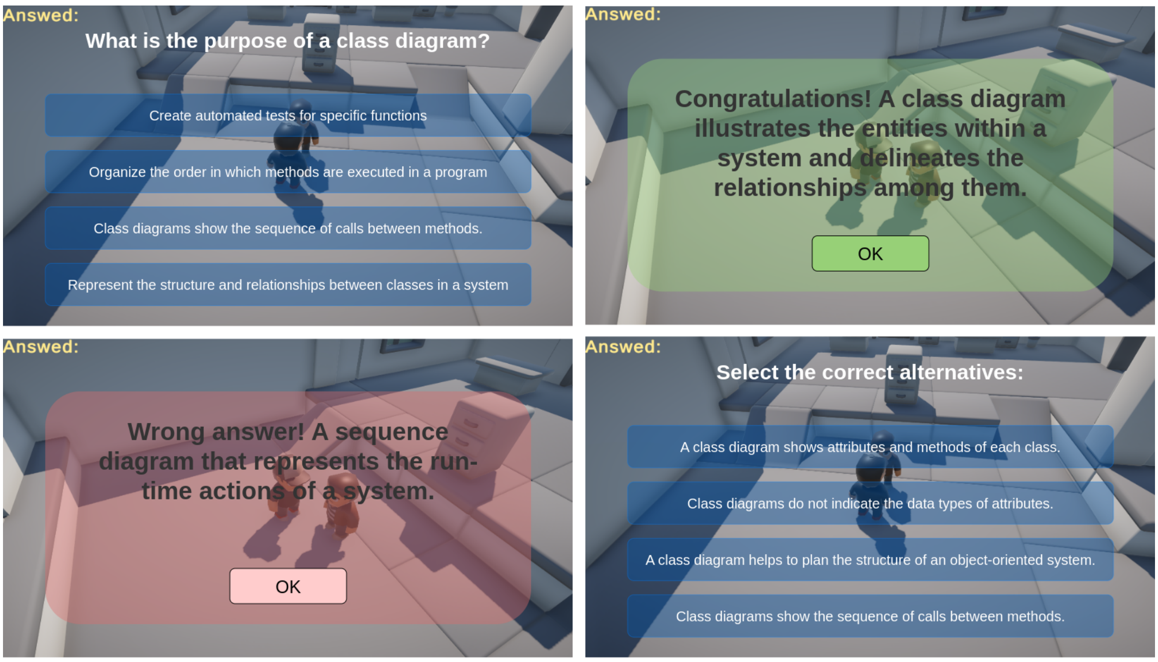


Figure 7.11: Examples of questions to be answered and feedback for the student.

New features were incorporated to enhance game interest and present additional challenges. The implementation of an A* algorithm enabled NPCs to follow the player, the feature was called as "curious student". Figure 7.8 illustrates the feasibility of selecting this feature. This capability enabled that, upon an NPC encountering a player, a new question would be asked, with every wrong answer resulting in a deduction of one point from the total score. This required the player to complete the level quickly to avoid losing points for incorrect answers and to maintain the tension of the game. Figure 7.12 illustrates the touchpoints where the NPC was permitted to follow the user. Alongside the following feature, an additional function was implemented to provide an overview of each floor, represented by an NPC positioned near to the elevator. Figure 7.8 also illustrates the potential for including this feature. This feature was implemented to enhance comprehension of the material presented in each phase and to provide the user with a general context of the learning requirements for each level. Figure 7.13 shows an example of this feature.

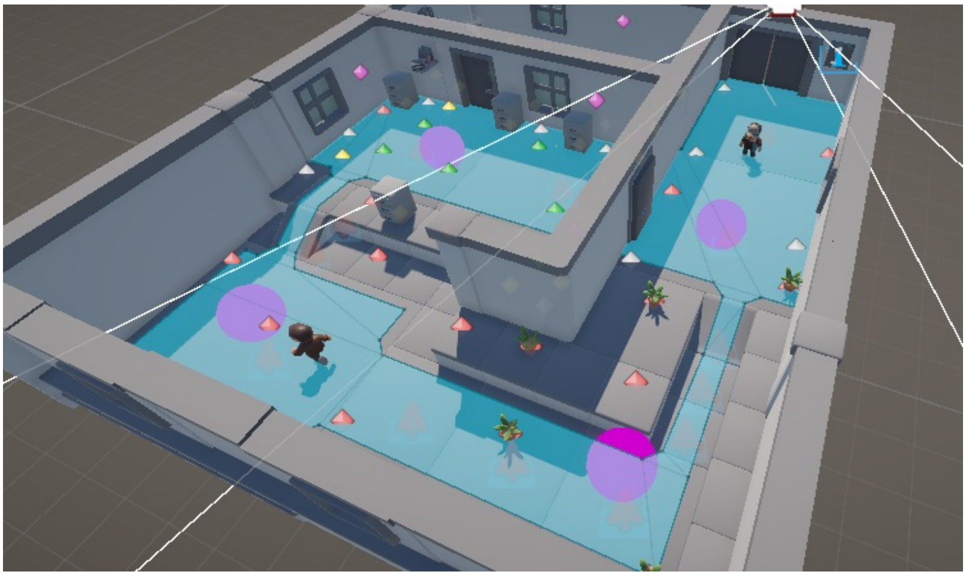


Figure 7.12: A* algorithm touchpoints.

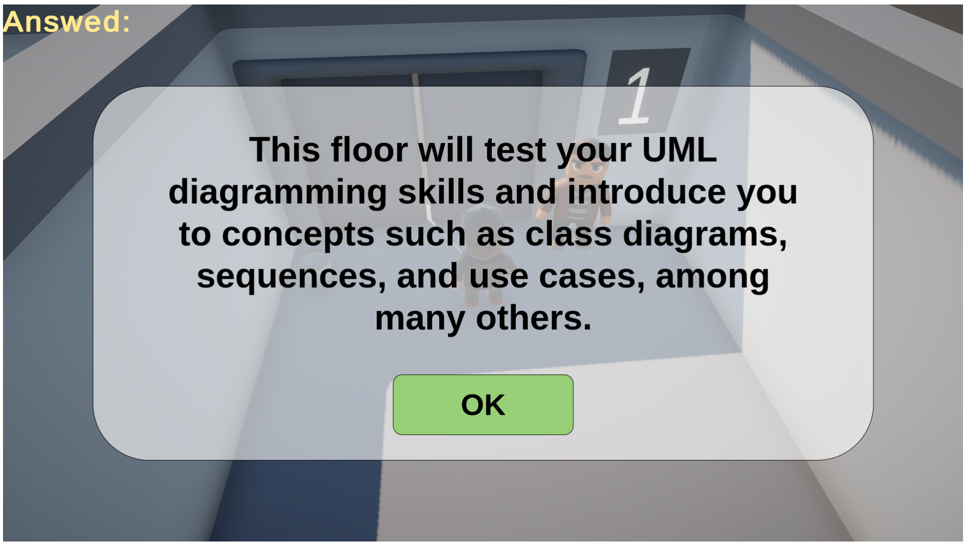


Figure 7.13: Dialogue with the NPC to describe the floor.

7.3 Instance Evaluation

Research was conducted to validate the proposal and construction of SPLEG. This study required the validation of the acceptance and necessity of SPLEG, necessitating an adaptation of the Technology Acceptance Model (TAM) to align with the study's (DAVIS, 1993). However, it was still necessary to validate the input interface of the questions by the teachers, as well as to validate the games created. For this, the MEEGA questionnaire was used. Therefore, the MEEGA questionnaire will validate the game's usability and experience, and the TAM questions will evaluate the usefulness of the idea behind the game. The forms used in the assessment can be found in Appendices G and H.

The evaluation process occurred from January 27, 2025, to February 30, 2025, preceded by a pilot evaluation on January 10, 2025, involving two participants, one to each group already mentioned, to ascertain that the game and questionnaire were free of issues that could compromise the evaluation, as well as to verify the duration required for a candidate to complete it. Following the pilot execution, it was determined that the procedure averaged 40 minutes in duration and encountered no issues detrimental to the experiment. Nevertheless, suggestions for enhancements were noted, including the incorporation of additional elements to increase engagement during gameplay and the introduction of an NPC to elucidate the context of each floor, both of which were added prior to the final assessment. These features have been previously referenced in the preceding section.

The evaluation procedure established two groups of respondents. The initial group, which consisted of 16 students from UFRJ who studied the games discipline (group A), responded exclusively to the MEEGA questionnaire. The second group completed the TAM and MEEGA questionnaires and employed a sample of six educators and researchers in EGs (group B), each of whom had over ten years of experience in the field. It is important to mention that two of the participants had previously participated in the survey. Figures 7.14 and 7.15 illustrate the characterization of the participants in each group. From the characterization of group A, it is possible to observe that a quarter of the population has already had classes through games and almost half of the population has already developed an EG. Furthermore, it is evident that nearly all participants in group A possessed a PhD and were older, which would be expected due to their over 10 years of experience in the field. Furthermore, it is shown that fifty percent of the population has utilized games in the classroom, while the remaining fifty percent has not, rendering the population ideal for study. This includes professionals with expertise in SPL and games, possessing over ten developed products, alongside educators lacking prior knowledge in the field.

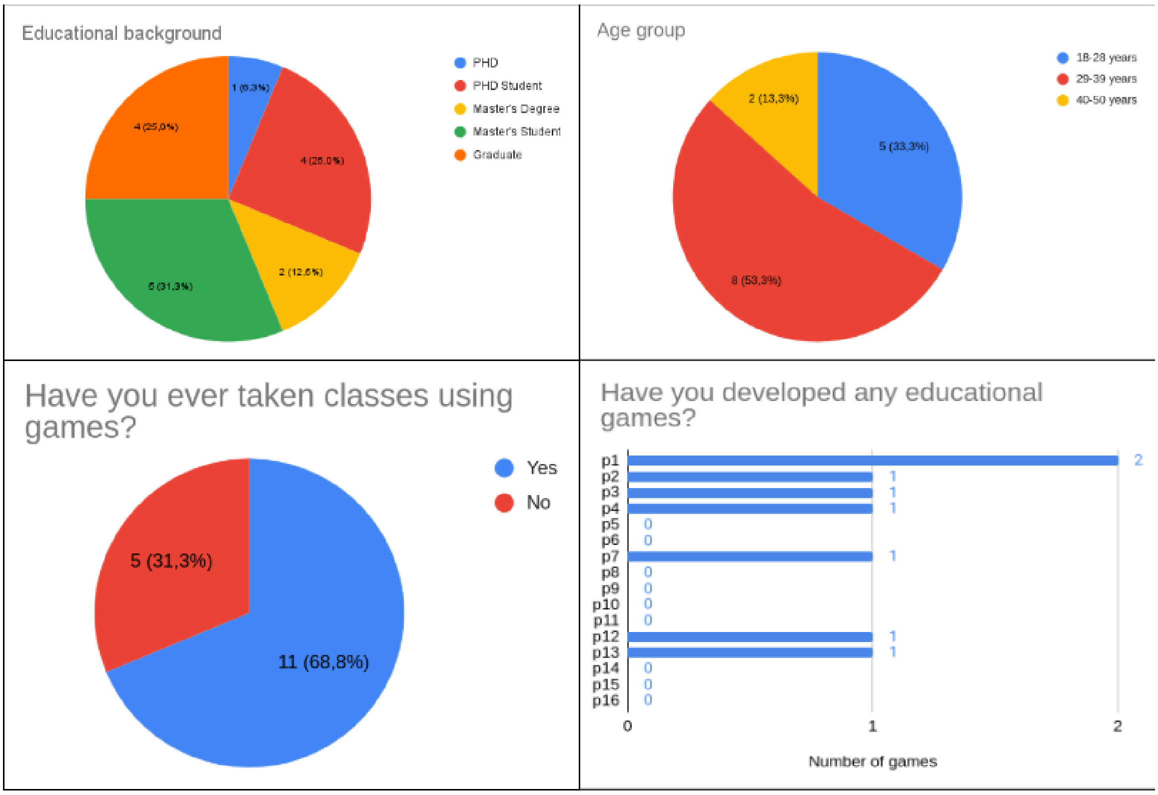


Figure 7.14: Characterization of group A.

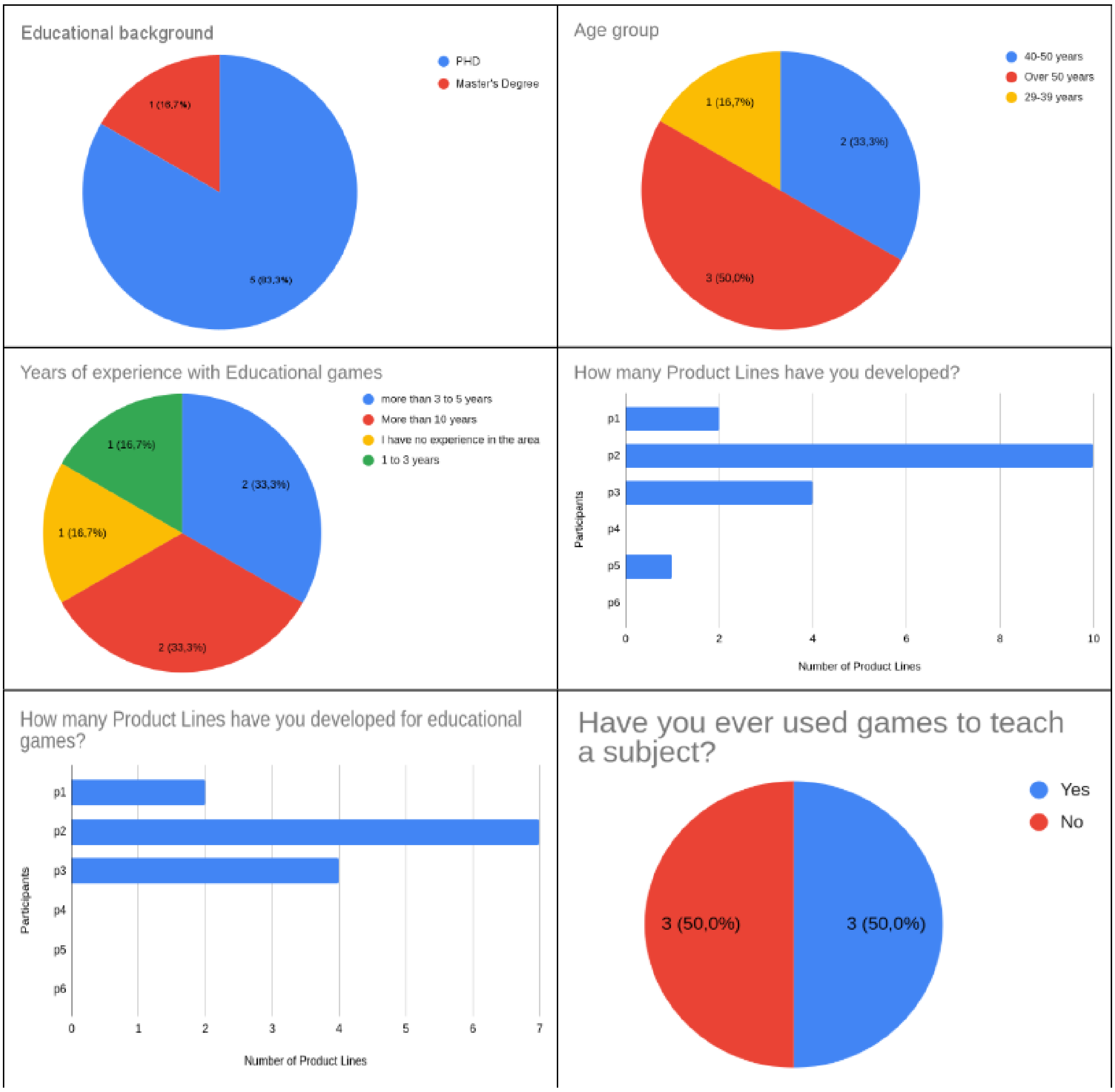


Figure 7.15: Characterization of group B.

The primary objective of group A was to guarantee that the students who would utilize the game would have a positive and cohesive experience. The results of the usability test for the game created are depicted in Figure 7.16. The vast majority of respondents provided positive responses, which indicates that the game has good usability. However, it is important to note that five respondents provided negative responses regarding the colors and fonts used in the game, which suggests that the fonts and colors displayed throughout the game may need to be changed. In terms of the experience, the scenarios remain constant, with the majority of outcomes being positive. Nevertheless, it is important to understand that the questions that were specifically designed to captivate the user's attention received some indifferent responses, as well as the relevance of the teaching content. Nevertheless, the last one may be a result of the SE instructional case study.

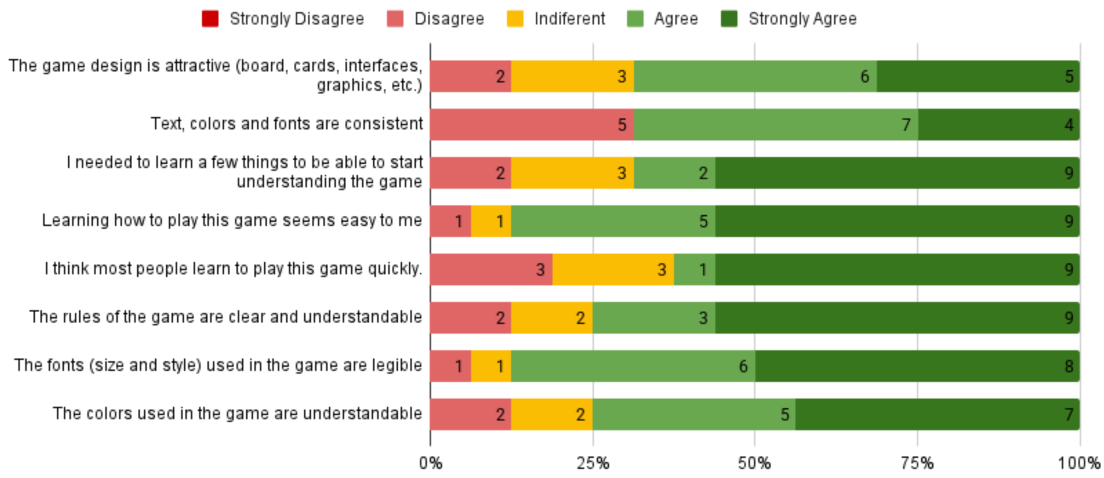


Figure 7.16: Group A Usability Assessment (MEEGA questionnaire).

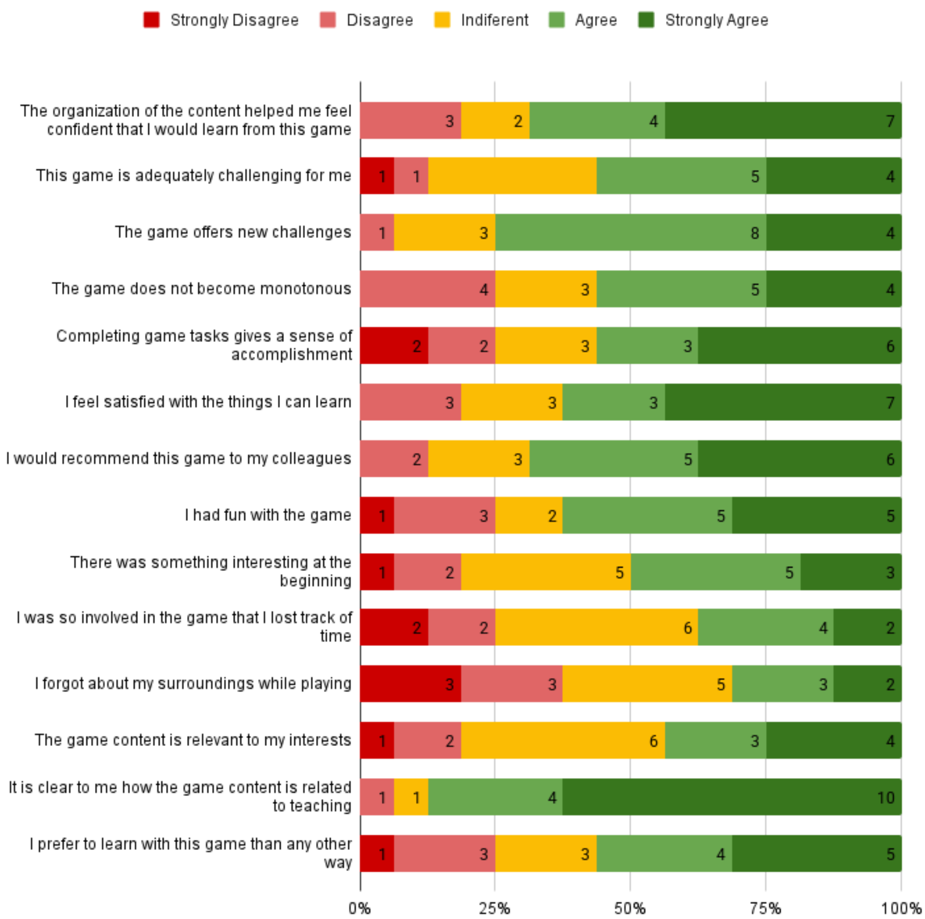


Figure 7.17: Group A Experience Assessment (MEEGA questionnaire).

The primary objective of the TAM questionnaire, as previously stated, was to assess the tool's functionality and ease of use. The evaluation results are illustrated in Figures 7.18 and 7.19. Concerning usability, the majority of responses were favorable; however, it is important to note that some respondents expressed indifference towards the tool's flexibility, which is the problem discussed in the approach outlined in this thesis. It is important to note that a game may encompass various concepts, and a game designed to instruct on a certain subject will always be stimulating for the player, as it was specifically created to share that knowledge. Nonetheless, the game faced two primary challenges: enabling teachers without previous experience to build games and abstracting the subject to be taught, hence employing the EA strategy. Consequently, it is expected that a degree of flexibility will be compromised in the effort of balancing these two objectives. Regarding usefulness, it is clear from the responses that the tool is useful for building games and would help teachers build games for their subjects.

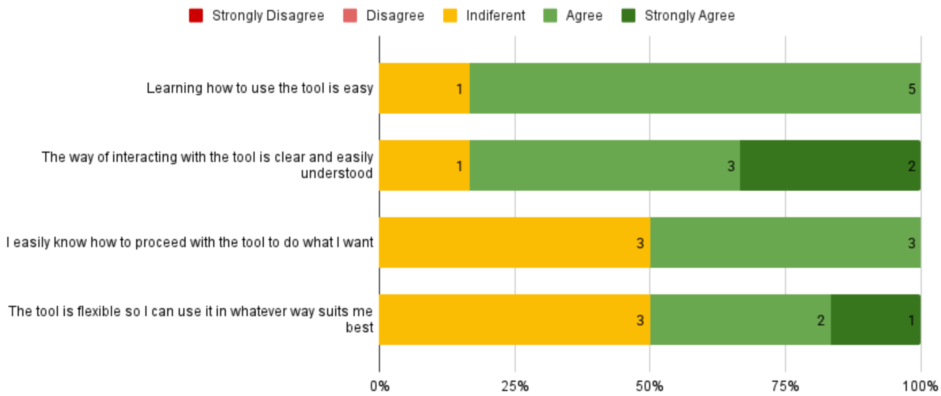


Figure 7.18: Perception of ease of use by group B (TAM questionnaire).

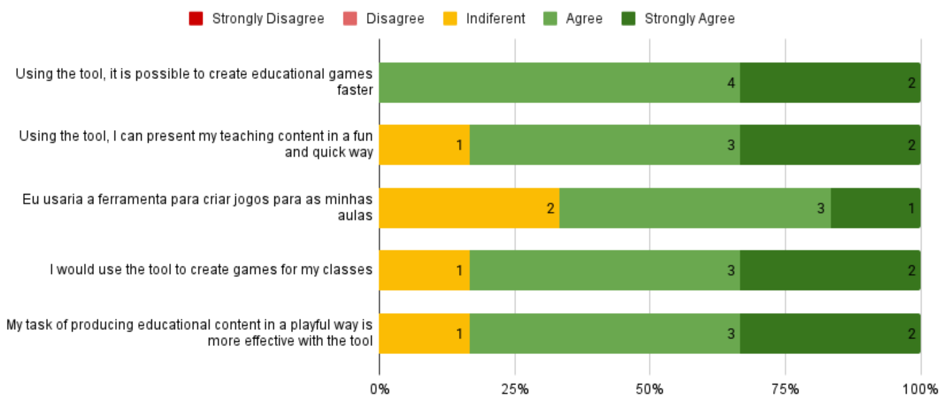


Figure 7.19: Perception of usefulness of use by group B (TAM questionnaire).

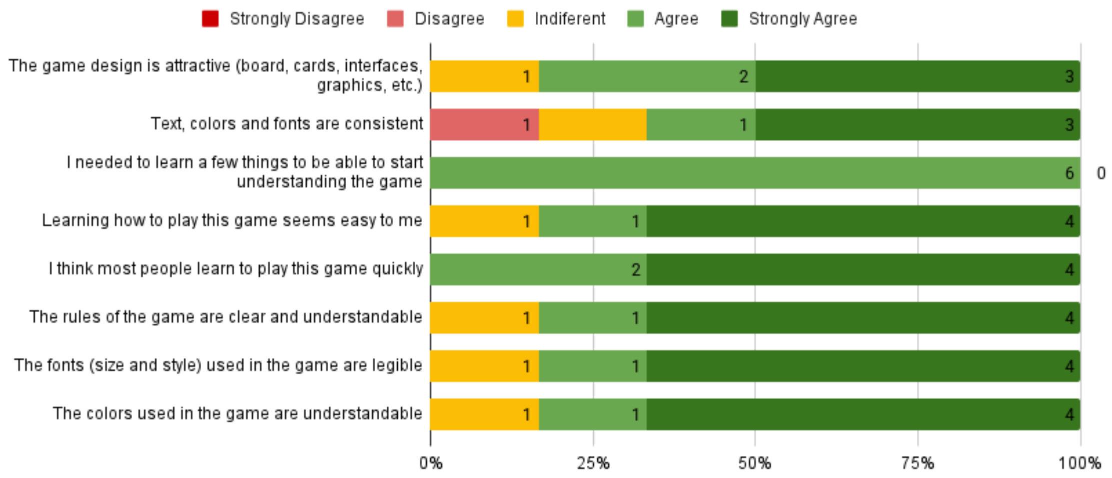


Figure 7.20: Group B Usability Assessment (MEEGA questionnaire).

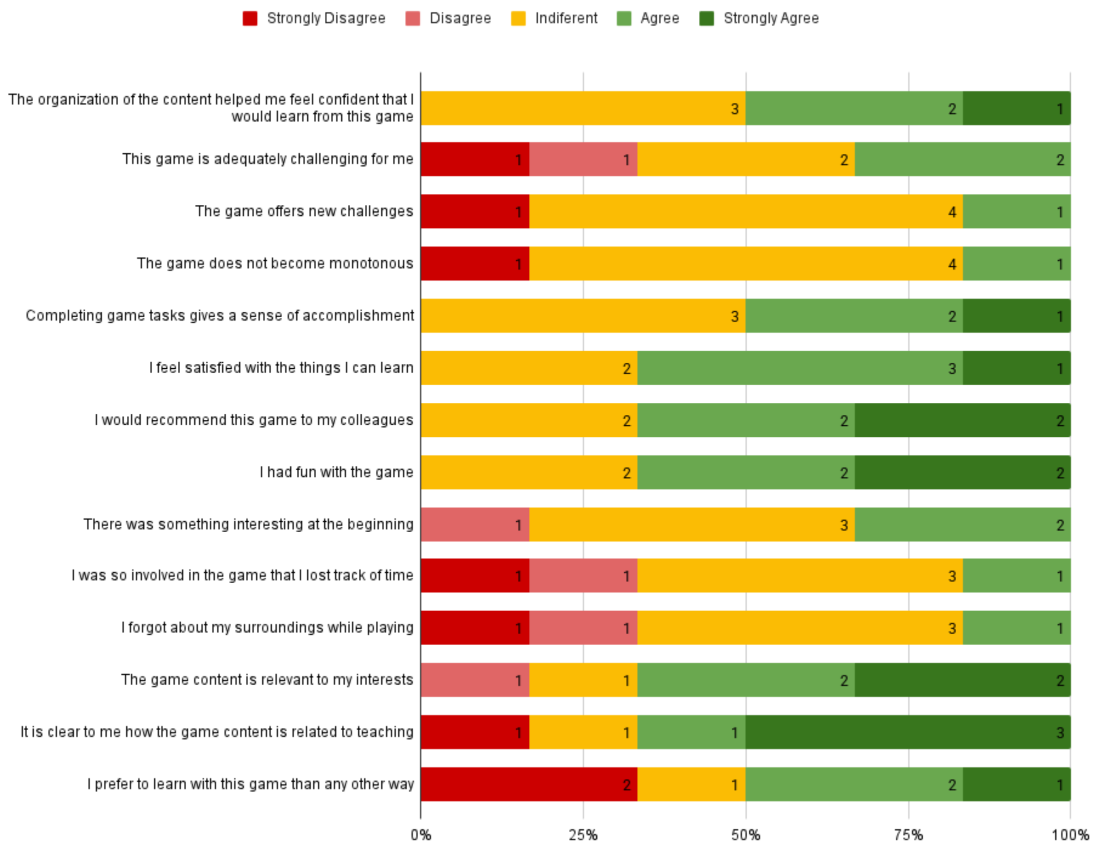


Figure 7.21: Group A Experience Assessment (MEEGA questionnaire).

Finally, the MEEGA questionnaire was carried out with experts in order to collect their perspectives on the effectiveness, usability, and experience of the games that were developed. In terms of usability, the results in Figure 7.20 are positive, indicating the tool has good usability, both in terms of the game created and in filling in the information to generate the game, confirming the answers given by group A. Regarding the experience provided by the game, it is possible to perceive a more careful analysis in relation to group A, which was to be expected, given the greater experience in the subject by group B. Figure 7.21 illustrates a similarity between positive and neutral results, alongside a small number of negative results. The tool currently offers a satisfactory user experience; however, enhancements are necessary, particularly regarding the incorporation of new challenges for the game, as previously noted in the pilot study, which was improved by adding the mechanics of the user's chase functionality by NPCs.

7.4 Final Consideration

The chapter illustrated the conversion of the FM developed in the literature study of Chapter 3 into a GM, aiming to enhance model description and diminish abstraction to support SPL development. The conversion utilized the rules described in Chapter 6, with each of the transformation steps described in detail.

The developed GM allowed the selection of characteristics for its implementation, each chosen based on the hypotheses supported by the experts from the survey presented in Chapter 4. Consequently, a game was developed from the model, named LEAP. The objective was to enable teachers without prior expertise to construct EGs, necessitating a specialization of this growth through the application of the EA technique. Figure 7.22 illustrates each step of this SPL development methodology, from the formulation of the FM to its development.

The game was evaluated by students, enthusiasts, and specialists in the domain of EG development. The results demonstrated great potential in the use of SPL for EG development, also demonstrating good usability and experience for the player.

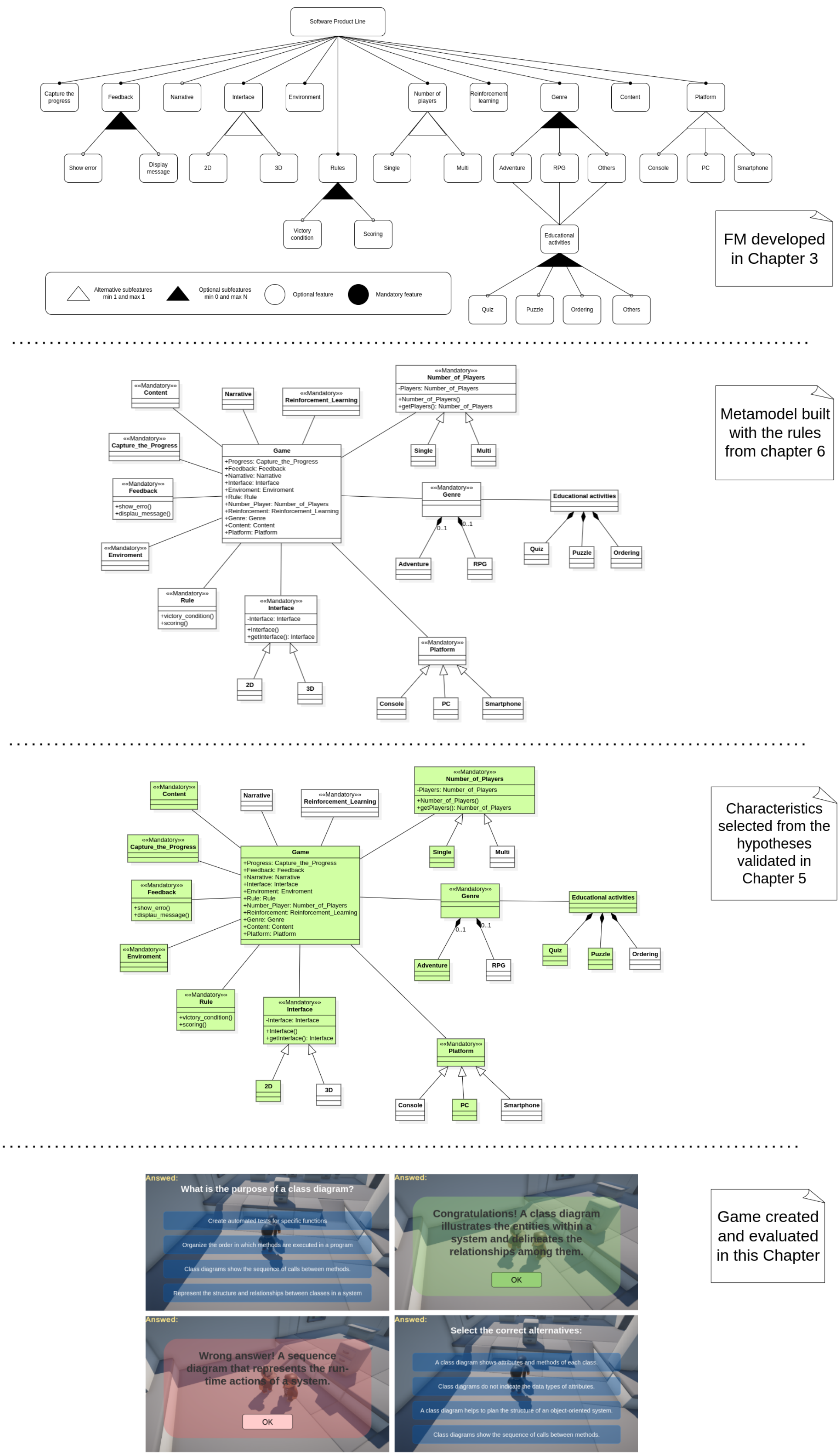


Figure 7.22: FM transformation stage until game development.

Chapter 8

Conclusion

This thesis aimed to illustrate the specialization of SPL for the development of EGs, necessitating the completion of multiple steps until its final achievement. The initial step involved conducting a literature review to identify existing contributions in the field. The study focused on enhancing the understanding of SPL utilization for development, which was crucial for understand the characteristics applicable in an SPL for EGs and the methodology for its construction.

The study conducted allowed for the identification of a number of characteristics that would be essential for the development of the SPLEG, with a focus on feedback mechanics, 2D or 3D interfaces, game categories, and educational activities. The combination of these characteristics gave rise to a FM that was validated through surveys with students, enthusiasts and experts in EGs. Hypotheses were generated to determine the characteristics of the SPL to be developed after the survey results were analyzed. Each hypothesis was verified by experts with over a decade of experience in EGs.

A FM just illustrates the essential characteristics that software must have, serving as a high-level representation of the variability within an SPL. To design software, it is essential to elucidate the relationships among these features to enable the developer's implementation. An approach for expressing the variety of software at a lower level of abstraction is the utilization of GMs, represented as class diagrams that depict entities, properties, and associations, albeit in an abstract rather than tangible manner. Consequently, a series of transformations were developed to facilitate the translation of the FM into a GM, with each rule being confirmed by SPL specialists.

In order to expedite its development, the FM constructed from the literature reviews was converted to a GM after the aforementioned rules were established, thereby illustrating the sequential process of each transformation. A model was generated by this GM by selecting the characteristics that led to the hypotheses that were subsequently validated by specialists. A game instance was developed using the model that was developed and was evaluated by students, enthusiasts and game experts. This demonstrates the significant potential of using SPL to specialize the development of EGs.

8.1 Contribution

A comprehensive examination of the provided thesis reveals multiple contributions to the domains of EGs and SR. It starts with literature review that elucidate the current advancements concerning the application of SPL concepts in game developments, indicating that SPL is the most recommended approach to specialize the development of EVGs, thus increasing the number of games developed, especially by people who do not have prior knowledge in the area. Furthermore, the game development sector was characterized through SPL, ultimately facilitating the construction of a FM that meets the requirements for the production of EVGs. These contributions were presented in Chapter 3.

Following the generation of FM, prototypes were constructed and evaluated to illustrate the viability of developing SPLEG. Hypotheses regarding the characteristics necessary for the SPLEG were formulated based on these prototypes and the FM, and they were subsequently discussed and assessed with specialists, encompassing both theoretical and practical dimensions. Chapters 4 and 5 provide a comprehensive account of the development process and the associated discussions.

A FM just illustrates the characteristics that an SPL must have, without exhibiting developmental structures; consequently, class diagrams are created. Transforming an SPL into a class diagram is a complicated activity, as it necessitates multiple transformation operations. Chapter 6 presents an additional contribution by illustrating transformation rules from a FM to a GM, which serves as an intermediary diagram between the FM and the class diagram, with the objective of facilitating this transformation.

The last contribution described in this thesis and the most important is the development of the SPLEG, following all the previous processes, from the design of the FM, through the GM, generating a model and eventually developing a generic EVGs that can be tailored to teach other sorts of subjects.

8.2 Bibliographic Production

Several papers were published, illustrating the significance of the contributions made in this thesis. Tables 8.1 and 8.2 demonstrate these publications.

Table 8.1: Papers published as first author.

Complete Reference	**Type**
Guide to Development of educational games through Product Line: a literature review, **D. Castro**, **C. Werner**, Interciencia Journal, (vol. 50, no. 1&3, 2025), ISSN 0378-1844. (To appear).	Journal
A multivocal review on derivation games: a software reuse study, **D. Castro**, **C. Werner**, International Journal on Advances in Software, (vol. 17, no. 1&2, 2024), pp. 68-79, ISSN 1942-2628.	Journal
Reusing and Deriving Games for Teaching Software Reuse, **D. Castro**, **C. Werner**, International Journal on Advances in Software, (vol. 13, no. 3&4, 2020), pp. 207-216, ISSN 1942-2628	Journal
Desenvolvendo Jogos por Meio de Linha de Produto Dinâmica, **D. Castro**, **C. Werner**, In: Congresso IberoAmericano em Engenharia de Software, 2024, Brasil. Anais do XXVII Congresso Ibero-Americano em Engenharia de Software (CIbSE 2024). pp. 380-387. (in portuguese)	Conference
Construindo Jogos Educacionais Através de Linha de Produto. **D. Castro**, **C. Werner**, In: Congresso IberoAmericano em Engenharia de Software, 2024, Brasil. Anais do XXVII Congresso Ibero-Americano em Engenharia de Software (CIbSE 2024). pp. 327-334. (in portuguese)	Conference
Extending Educational Games Across Product Lines, **D. Castro**, G. Xexéo, **C. Werner**, 13th International Conference on Videogame Sciences and Arts (VJ 2023), Aveiro, Portugal, novembro 2023, pp. 134-149	Conference
Castro, D.; **Werner, C. M. L.** A Multivocal Review on Derivation Games. in: the 2023 IARIA Annual Congress on Frontiers in Science, Technology, Services, and Applications, Valencia, Spain. 2023. p. 144-149.	Conference
Castro, D.; **Werner, C. M. L.** Exploring Product Line Concepts in Game Building. in: the 2023 IARIA Annual Congress on Frontiers in Science, Technology, Services, and Applications, Valencia, Spain. 2023. p. 150-152.	Conference
A structured review of game coding through modeling, **D. Castro**, **C. Werner**, XX Simpósio Brasileiro de Games e Entretenimento Digital (SBGames), Gramado, novembro 2021, pp. 1-5	Conference
Rebuilding games at runtime, **D. Castro**, **C. Werner**, First International Workshop on Automated Software Engineering for Computer Games (ASE4Games), Melbourne, Australia, novembro 2021, pp. 73-77	Conference

Systematic Mapping on Software Reuse Teaching, **D. Castro** e **C. Werner**, 12th International Conference on Information and Communication Systems (ICICS), Valencia, Espanha, maio 2021, pp. 257-264	Conference
Systematic mapping on the use of games for software engineering education, **D. Castro**, D. Costa, **C. Werner**, XXIII Iberoamerican Conference on Software Engineering (CIbSE 2020), Curitiba, Paraná, novembro 2020, pp. 512-525	Conference
Unfolding for creation of educational games, **D. Castro**, **C. Werner**, Trilha de Educação do XIX Simpósio Brasileiro de Games e Entretenimento Digital (SBGames), Recife, novembro 2020, pp. 822-825	Conference

Table 8.2: Papers published as co-authorship.

Complete Reference	**Type**
Immersive Learning Research from SVR Publications: A Re-conduction of the Systematic Mapping Study, F. Fernandes, **D. Castro**, **C. Werner**, Journal on Interactive Systems, (vol. 13, no. 1, 2022), pp. 205-220, ISSN 2763-7719	Journal
Evaluating User Experience of a Software Engineering Education Virtual Environment, F. A. Fernandes, **D. C. Castro**, C. S. Rodrigues, **C. M. Werner**, 24th Symposium on Virtual and Augmented Reality, outubro 2022, pp. 137-141.	Conference
Development of the Software Engineering Education Virtual Classroom Prototype: An Experience Report, F. Fernandes, **D. Castro**, C. Rodrigues, **C. Werner**, 30o Workshop sobre Educação em Computação (WEI), Niterói, agosto 2022, pp. 85-96	Conference
A Systematic Mapping Literature of Immersive Learning from SVR Publications, F. Fernandes, **D. Castro**, **C. Werner**, 23rd Symposium on Virtual and Augmented Reality (SVR), Gramado, novembro 2021, pp. 1-13	Conference

8.3 Threats to validity

Threats to validity are potential risks that are involved in the design and execution of studies. These threats can limit the ability to produce reliable results or specialize them to a larger population than those used in the experiments. From a critical analysis of the thesis, it is possible to find some threats to validity IHANTOLA and KIHN (2011). These threats were split into four types, as follows:

- **Conclusion validity:** These threats relate to the risk of obtaining incorrect conclusions from the results found.
  - The assessments carried out in this thesis followed some types of protocols, such as surveys, interviews, and remote and in-person assessments, which may have caused a different level of reliability across studies.
- **Internal validity:** These threats are related to the degree of confidence on the information found and the influence of other factors or variables on this information. Their presence is associated with cause and effect problems.
  - This study assessed enthusiasts, students and experts with different levels of knowledge, which may result in different results.
- **Construct validity:** These threats are associated with the risk that the scenarios built for the experiments (and therefore the collected data) do not reflect theory.
  - For remote users and specialists, information of study execution was sent by e-mail, which could cause a misunderstanding of some instructions in the tool or the study.
- **External validity:** These threats are related to the risk of specializing observed results for a larger population, i.e. beyond the sample used in the experiments.
  - Sample Size Limitations. There is a risk that replicating this research with a larger sample may lead to different results.
    - The survey had only 35 participants.
    - The interview with experts to validate hypotheses had only 3 participants.
    - The evaluation of the GM notation had only 3 SPL experts.
    - Only 21 participants carried out the final game study, with only 6 experts.
  - The experiments were carried out remotely in uncontrolled environments, making it difficult to specialize the results.

8.4 Ethical issues

Data collection via TAM, MEEGA questionnaires, surveys, and interviews entails acquiring personal information, perceptions, and opinions from subjects, hence necessitating ethical consideration in research involving human subjects. In accordance with the recommendations of certain studies, all data were reported anonymously, thereby preventing participant identification and reducing the risks associated with privacy and confidentiality ALBUQUERQUE (2019); SANTOS (2020).

8.5 Future Work

Upon the execution of the thesis analysis, it is possible to identify potential future studies in each of the research areas discussed. MDD is the initial area that demands further investigation. This area was underscored by the demonstration of only one prototype throughout the course of the research. Nevertheless, the research also revealed that it is a potential area of SR that could be beneficial in the development of educational games.

Another promising area of research is the exploration of modeling FMs, as UML lacks a specified structure to represent this type of model. The thesis exclusively demonstrated manual rules for transforming a FM into a GM. An interesting study would involve developing a tool that executes this process semi-automatically, as the developer's analysis is a requirement at certain stages.

The conclusive evaluation of the LEAP game reveals that enhancements are required in the color palettes, arrangement of screen components, and the integration of additional difficulties. Future plans include ongoing development of the game to address the stated issues. Furthermore, a single game was developed using the exhibited GM. New instances are expected to be generated from it.

Ultimately, the survey demonstrated that the development of EVGs would necessitate the implementation of multiple SPLs, each of which would be contingent upon its specific application context. As a result, the examination of these trees may yield positive results that can be implemented in various contexts.

References

ABBOTT, D., 2018, “Modding Tabletop Games for Education”, *Proceedings of International Conference on Games and Learning Alliance*, pp. 318–329.

ADAMS, E., DORMANS, J., 2012, *Game mechanics: advanced game design*. Hoboken, New Jersey, New Riders.

AGUIAR, B., CORREIA, W., CAMPOS, F., 2011, “Using the Likert scale in game analysis”, *Salvador: SBC-Proceedings of SBGames Anais*, v. 7, n. 2, pp. 2. (in Portuguese).

ALBUQUERQUE, A. O., 2019, “Pesquisa com prontuário: análise ético-jurídica à luz dos Direitos Humanos dos Pacientes”, *Cadernos de Ética em Pesquisa*, v. 1, n. 1, pp. 41–52.

ARAÚJO, J. C., ARAÚJO, M. C., NEIVA, F. W., 2022, “An integrated and interdisciplinary learning approach through a cross-platform development process”, *ANALECTA-Centro Universitário Academia*, v. 7, n. 2. (in Portuguese).

ARNAB, S., LIM, T., CARVALHO, M. B., et al., 2015, “Mapping learning and game mechanics for serious games analysis”, *British Journal of Educational Technology*, v. 46, n. 2, pp. 391–411.

AYALA, I., PAPADOPOULOS, A. V., AMOR, M., et al., 2021, “Prodspl: Proactive self-adaptation based on dynamic software product lines”, *Journal of Systems and Software*, v. 175, pp. 110909.

BARCZAK, A. M., WOŹNIAK, H., 2019, “Comparative study on game engines”, *Studia Informatica. Systems and Information Technology. Systemy i Technologie Informacyjne*, pp. 1–2.

BAU, D., 2015, “Droplet, a blocks-based editor for text code”, *Journal of Computing Sciences in Colleges*, v. 30, n. 6, pp. 138–144.

BAYTAK, A., LAND, S. M., 2010, “A case study of educational game design by kids and for kids”, *Procedia-Social and Behavioral Sciences*, v. 2, n. 2, pp. 5242–5246.

BELLOTTI, F., KAPRALOS, B., LEE, K., et al., 2013, “Assessment in and of serious games: An overview”, *Advances in human-computer interaction*, v. 2013, pp. 1–1.

BERGMANN, J., SAMS, A., 2012, *Flip Your Classroom: Reach Every Student in Every Class Every Day.* San Antonio, Texas, International Society for Technology in Education.

BOAVENTURA, F. M., SARINHO, V. T., 2019, “A Feature-Based Approach to Develop Digital Board Games”. In: *Entertainment Computing and Serious Games: First IFIP TC 14 Joint International Conference, ICEC-JCSG 2019, Arequipa, Peru, November 11–15, 2019, Proceedings 1*, pp. 175–186. Springer.

BOLLER, S., KAPP, K., 2017a, *Play to learn: Everything you need to know about designing effective learning games.* Alexandria, United States, Association for talent development.

BOLLER, S., KAPP, K., 2017b, *Play to learn: Everything you need to know about designing effective learning games.* Alexandria, VA, Association for talent development.

BONWELL, C. C., EISON, J. A., 1991. “Active Learning: Creating Excitement in the Classroom”. ASHE-ERIC Higher Education Report No. 1. Washington, D.C.: The George Washington University, School of Education and Human Development.

BOSCH, J., 2004, “Software variability management”. In: *Proceedings. 26th International Conference on Software Engineering*, pp. 720–721. IEEE.

BOURQUE, P., FAIRLEY, R. E., 2014, *Guide to the Software Engineering Body of Knowledge SWEBOK.* 3rd ed. , IEEE Computer Society. ISBN: 978-0-7695-5166-1.

BOYLE, E. A., HAINEY, T., CONNOLLY, T. M., et al., 2016, “An update to the systematic literature review of empirical evidence of the impacts and outcomes of computer games and serious games”, *Computers & Education*, v. 94, pp. 178–192. ISSN: 0360-1315. doi: https://doi.org/10.1016/j.compedu.2015.11.003. Available at: <https://www.sciencedirect.com/science/article/pii/S0360131515300750>.

BROWN, D., 2015, "Games and the Magic Circle". In: Lee, N. (Ed.), *Encyclopedia of Computer Graphics and Games*, pp. 1–4, Cham, Springer International Publishing. ISBN: 978-3-319-08234-9. doi: 10.1007/978-3-319-08234-9_32-1. Available at: <https://doi.org/10.1007/978-3-319-08234-9_32-1>.

CALDERÓN, A., TRINIDAD, M., RUIZ, M., et al., 2018, "Teaching Software Processes and Standards: A Review of Serious Games Approaches". In: *International Conference on Software Process Improvement and Capability Determination*, pp. 154–166. Springer. doi: 10.1007/978-3-030-00623-5_11.

CAPILLA, R., GALLINA, B., CETINA, C., et al., 2019, "Opportunities for software reuse in an uncertain world: From past to emerging trends", *Journal of software: Evolution and process*, v. 31, n. 8, pp. e2217.

CASTRO, D., WERNER, C., 2023, "A Multivocal Review on Derivation Games". In: *IARIA Congress*, pp. 144–149, Valencia, Spain, November.

CASTRO, D., WERNER, C., 2021, "Rebuilding games at runtime". In: *2021 36th IEEE/ACM International Conference on Automated Software Engineering Workshops (ASEW)*, pp. 73–77. IEEE.

CASTRO, D. C. B., 2020. "Use of Games as a Strategy for Teaching Software Reuse". Master's Dissertation, Universidade Federal do Rio de Janeiro, (In Portuguese).

CAVALCANTI, Y. C., DO CARMO MACHADO, I., DA MOTA, P. A., et al., 2011, "Towards metamodel support for variability and traceability in software product lines". In: *Proceedings of the 5th International Workshop on Variability Modeling of Software-Intensive Systems*, pp. 49–57.

CECHTICKY, V., PASETTI, A., ROHLIK, O., et al., 2004, "XML-based feature modelling". In: *Software Reuse: Methods, Techniques, and Tools: 8th International Conference, ICSR 2004, Madrid, Spain, July 5-9, 2004. Proceedings 8*, pp. 101–114. Springer.

CÉRET, E., DUPUY-CHESSA, S., CALVARY, G., 2013, "M2Flex: a process metamodel for flexibility at runtime". In: *IEEE 7th International Conference on Research Challenges in Information Science (RCIS)*, pp. 1–12. IEEE.

CHEN, L., ALI BABAR, M., ALI, N., 2009, "Variability management in software product lines: A systematic review". pp. 81–90, 01. doi: 10.1145/1753235.1753247.

COOPER, K., SCACCHI, W., 2015, "Introducing computer games and software engineering", *Computer games and software engineering*, pp. 1–27. Publisher: Chapman and Hall/CRC Boca Raton.

COSTIKYAN, G., 2013, *Uncertainty in Games*. Massachusetts, EUA, MIT Press.

CUSKER, J., 2013, "Elsevier Compendex and Google Scholar: a quantitative comparison of two resources for engineering research and an update to prior comparisons", *The Journal of Academic Librarianship*, v. 39, n. 3, pp. 241–243.

CZARNECKI, K., HELSEN, S., EISENECKER, U., 2004, "Staged configuration using feature models". In: *Software Product Lines: Third International Conference, SPLC 2004, Boston, MA, USA, August 30-September 2, 2004. Proceedings 3*, pp. 266–283. Springer.

DAVIS, F. D., 1993, "User acceptance of information technology: system characteristics, user perceptions and behavioral impacts", *International journal of man-machine studies*, v. 38, n. 3, pp. 475–487.

DE ARAUJO, G. G., DA SILVA ARANHA, E. H., 2013, "Formative assessment of skills and abilities: instrumentation for digital games", *RENOTE*, v. 11, n. 3. (In Portuguese).

DE OLIVEIRA, R. F., 2006. "Formalization and Consistency Verification in the Representation of Variabilities". Master's Dissertation, Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brasil. (In Portuguese).

DIAS, D. M., BATTESTIN, V., DE OLIVEIRA, M. G., 2024, "H5P for educators: enhancement of interactive learning experiences in Moodle environment", *Revista Eletrônica Sala de Aula em Foco*, v. 13, n. 1, pp. 97–110. (In Portuguese).

DORMANS, J., 2011, "Simulating mechanics to study emergence in games". In: *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, v. 7, pp. 2–7.

FLANAGAN, M., NISSENBAUM, H., 2014, *Values at Play in Digital Games*. Massachusetts, EUA, MIT Press.

FREEMAN, S., EDDY, S. L., MCDONOUGH, M., et al., 2014, "Active learning increases student performance in science, engineering, and mathematics", *Proceedings of the National Academy of Sciences*, v. 111, n. 23, pp. 8410–8415.

FURTADO, A. W., SANTOS, A. L., RAMALHO, G. L., et al., 2011, "Improving digital game development with software product lines", *IEEE software*, v. 28, n. 5, pp. 30–37.

GAMEANALYTICS, 2025. "GameAnalytics: Turning Data Into Insights". Available at: <https://www.gameanalytics.com/>. Accessed on: October 3, 2024.

GARCÉS, K., PARRA, C., ARBOLEDA, H., et al., 2007, "Variability management in a model-driven software product line", *Revista Avances en Sistemas e Informática*, v. 4, n. 2, pp. 3–12.

GARCIA, L. A., OLIVEIRAJR, E., MORANDINI, M., 2022, "Tailoring the Scrum framework for software development: Literature mapping and feature-based support", *Information and Software Technology*, v. 146, pp. 106814.

GAROUSI, V., FELDERER, M., MÄNTYLÄ, M. V., 2019, "Guidelines for including grey literature and conducting multivocal literature reviews in software engineering", *Information and Software Technology*, v. 106, pp. 101–121.

GARVEY, C. M., JONES, R., 2021, "Is there a place for theoretical frameworks in qualitative research?" *International Journal of Qualitative Methods*, v. 20. doi: 10.1177/1609406920987959. Available at: <https://doi.org/10.1177/1609406920987959>.

GEE, J. P., 2003, "What video games have to teach us about learning and literacy", *Computers in entertainment (CIE)*, v. 1, n. 1, pp. 20–20.

GEISLER, B. J., KAVAGE, S. L., 2021, "A multi-engine aspect-oriented language with modeling integration for video game design". In: *Evaluation of Novel Approaches to Software Engineering: 15th International Conference, ENASE 2020, Prague, Czech Republic, May 5–6, 2020, Revised Selected Papers 15*, pp. 336–359. Springer.

GEORGIEV, A., GRIGOROV, A., BONTCHEV, B., et al., 2016, "The RAGE advanced game technologies repository for supporting applied game development". In: *International Conference on Games and Learning Alliance*, pp. 235–245. Springer.

GONZÁLEZ GARCÍA, C., NÚÑEZ-VALDEZ, E. R., MORENO-GER, P., et al., 2019, "Agile development of multiplatform educational video games using a Domain-Specific Language", *Universal Access in the Information Society*, v. 18, n. 3, pp. 599–614.

GUARDIOLA, E., 2016, "The gameplay loop: a player activity model for game design and analysis". In: *Proceedings of the 13th International Conference on Advances in Computer Entertainment Technology*, pp. 1–7.

HUIZINGA, J., 1999, *Homo Ludens: A Study of the Play-Element in Culture*. London, United Kingdom, Routlegde.

HUNICKE, R., LEBLANC, M., ZUBEK, R., 2004, "MDA: A formal approach to game design and game research". In: *Proceedings of the AAAI Workshop on Challenges in Game AI*, v. 4.

HUNT, J., HUNT, J., 2013, "Gang of four design patterns", *Scala Design Patterns: Patterns for Practical Reuse and Design*, pp. 135–136.

IHANTOLA, E.-M., KIHN, L.-A., 2011, "Threats to validity and reliability in mixed methods accounting research", *Qualitative Research in Accounting & Management*. doi: 10.1108/11766091111124694.

JÄÄSKELÄINEN, R., 2010, "Think-aloud protocol", *Handbook of translation studies*, v. 1, pp. 371–374.

JENKINS, H., SQUIRE, K., TAN, P., 2004, "You can't bring that game to school. Designing Supercharged", *Design research*, pp. 244–252.

JOHNSON, R. E., 1997, "Frameworks=(components+ patterns)", *Communications of the ACM*, v. 40, n. 10, pp. 39–42.

JUUL, J., 2005, *Half-real: Video games between real rules and fictional worlds*. Massachusetts, EUA, MIT press.

KALMPOURTZIS, G., 2019, *Educational Game Design Fundamentals: Journey to Creating Intrinsically Motivating Learning Experiences*. Florida, EUA, CRC Press.

KANG, K. C., 2010, "FODA: Twenty Years of Perspective on Feature Modeling". In: *Proceedings of VaMoS'10*, v. 37, *ICB-Research Report*, p. 9. Universität Duisburg-Essen.

KANG, K. C., COHEN, S. G., HESS, J. A., et al., 1990, *Feature-oriented domain analysis (FODA) feasibility study*. Technical report, Carnegie-Mellon Univ Pittsburgh Pa Software Engineering Inst.

KANG, K. C., KIM, S., LEE, J., et al., 1998, "FORM: A feature-; oriented reuse method with domain-; specific reference architectures", *Annals of software engineering*, v. 5, n. 1, pp. 143–168.

KITCHENHAM, B., BRERETON, O. P., BUDGEN, D., et al., 2009, "Systematic literature reviews in software engineering–a systematic literature review", *Information and software technology*, v. 51, n. 1, pp. 7–15.

KRUEGER, C. W., 1992, "Software reuse", *ACM Computing Surveys (CSUR)*, v. 24, n. 2, pp. 131–183.

LACERDA, D. P., DRESCH, A., PROENÇA, A., et al., 2013, "Design Science Research: research method for production engineering", *Gestão & produção*, v. 20, n. 4, pp. 741–761. (In Portuguese).

LARMER, J., MERGENDOLLER, J. R., BOSS, S., 2015, *Setting the Standard for Project Based Learning: A Proven Approach to Rigorous Classroom Instruction*. San Antonio, Texas, Association for Supervision and Curriculum Development (ASCD).

LEE, J., KIM, T., KANG, S., 2020, "Recovering software product line architecture of product variants developed with the clone-and-own approach". In: *2020 IEEE 44th Annual Computers, Software, and Applications Conference (COMPSAC)*, pp. 985–990. IEEE.

LESSA FILHO, C. A. C., HERNÁNDEZ-DOMÍNGUEZ, A., 2016, "JIndie: A Software Product Line for Educational Games with a Focus on Constructionism". In: *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, v. 27, p. 350. Issue: 1.

LITTMAN, M. L., 2015, "Reinforcement learning improves behaviour from evaluative feedback", *Nature*, v. 521, n. 7553, pp. 445–451.

MAAZOUN, J., BOUASSIDA, N., BEN-ABDALLAH, H., 2014, "A bottom up SPL design method". In: *2014 2nd International Conference on Model-Driven Engineering and Software Development (MODELSWARD)*, pp. 309–316. IEEE.

MAIA, V. C., 2024, *Towards a Metamodel to Support the Creation of Interactive Software Systems for Managing the Location and Use of Medical Equipment in Health Centers*. Ph.D. Thesis, Federal University of Rio de Janeiro.

MARTINEZ, J., TËRNAVA, X., ZIADI, T., 2018, "Software product line extraction from variability-rich systems: the robocode case study". In: *Proceedings of the 22nd International Systems and Software Product Line Conference-Volume 1*, pp. 132–142.

MARTINS, G., CAMPOS, F., BRAGA, R., et al., 2015, “BROAD-PLG: Computational Model for Building Educational Games”. In: *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, v. 26, p. 449. Issue: 1.

MARTINS, G., VEIGA, W., CAMPOS, F., et al., 2018, “Building educational games from a feature model”. In: *Proceedings of the XIV Brazilian Symposium on Information Systems*, pp. 1–7.

MARTÍNEZ, J.-C., PABÓN, M.-C., RINCÓN, L., et al., 2018, “Using software product lines to support language rehabilitation therapies: An experience report”. In: *2018 ICAI Workshops (ICAIW)*, pp. 1–6. IEEE.

MARTÍNEZ ARIAS, J. C., ÁLVAREZ VARGAS, G. I., SIERRA GALVIS, M. V. A., et al., 2021, “SATReLO: A tool to support language therapies for children with hearing disabilities using video games”, *Revista Facultad de Ingeniería Universidad de Antioquia*, , n. 99, pp. 99–112. Publisher: Universidad de Antioquia.

MATALONGA, S., RODRIGUES, F., TRAVASSOS, G. H., 2017, “Characterizing testing methods for context-aware software systems: Results from a quasi-systematic literature review”, *Journal of Systems and Software*, v. 131, pp. 1–21.

MCARTHUR, V., TEATHER, R. J., 2015, “Serious mods: A case for modding in serious games pedagogy”, *Proceedings of IEEE Games Entertainment Media Conference (GEM)*, pp. 1–4.

MEFTAH, C., RETBI, A., BENNANI, S., et al., 2018, “Exploration of software product line to enrich the modeling of mobile serious games”. In: *Proceedings of the 12th International Conference on Intelligent Systems: Theories and Applications*, pp. 1–5.

MEFTAH, C., RETBI, A., BENNANI, S., et al., 2019, “Mobile Serious Game Design using User Experience: Modeling of Software Product Line Variability”, *International Journal of Emerging Technologies in Learning (iJET)*, v. 14, n. 23 (Dec), pp. 55–66. doi: 10.3991/ijet.v14i23.10899. Available at: <https://online-journals.org/index.php/i-jet/article/view/10899>.

MORAES, T. M., SOUZA, A. D., 2011, “Systematic review of communication within the software development process”, *Universidade Federal de Goiás-GO*, p. 57. (in Portuguese).

MORAN, J. M., 2000, *New technologies and pedagogical mediation.* Campinas, São Paulo, Papirus Editora. (In Portuguese).

MOTTA, R. C., DE OLIVEIRA, K. M., TRAVASSOS, G. H., 2016, "Characterizing Interoperability in Context-aware Software Systems", *Proceedings of VI Brazilian Symposium on Computing Systems Engineering (SBESC)*, pp. 203–208.

NCUBE, C., OBERNDORF, P., KARK, A. W., 2008, "Opportunistic software systems development: making systems from what's available", *IEEE Software*, v. 25, n. 6, pp. 38–41.

OBJECT MANAGEMENT GROUP, 2017. "Unified Modeling Language (UML), Version 2.5.1". December. Available at: <`https://www.omg.org/spec/UML`>. Accessed: 2025-03-31.

O'ROURKE, E., HAIMOVITZ, K., BALLWEBER, C., et al., 2014, "Brain points: A growth mindset incentive structure boosts persistence in an educational game". In: *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 3339–3348.

PERDEK, J., VRANIĆ, V., 2023, "Lightweight aspect-oriented software product lines with automated product derivation". In: *European Conference on Advances in Databases and Information Systems*, pp. 499–510. Springer.

PETRI, G., VON WANGENHEIM, C. G., BORGATTO, A. F., 2016, "MEEGA+: an evolution of a model for the evaluation of educational games", *INCoD/GQS*, v. 3, pp. 1–40.

PETTICREW, M., ROBERTS, H., 2008, *Systematic reviews in the social sciences: A practical guide.* John Wiley & Sons.

PIVEC, M., DZIABENKO, O., SCHINNERL, I., 2003, "Aspects of game-based learning". In: *3rd International Conference on Knowledge Management, Graz, Austria*, v. 304.

POLITOWSKI, C., PETRILLO, F., MONTANDON, J. E., et al., 2021, "Are game engines software frameworks? A three-perspective study", *Journal of Systems and Software*, v. 171, pp. 110846.

POUR, G., 2000, "Integrating component-based and reuse-driven software engineering business into software and information engineering curriculum".

In: *30th Annual Frontiers in Education Conference. Building on A Century of Progress in Engineering Education. Conference Proceedings (IEEE Cat. No. 00CH37135)*, v. 1, pp. T2C–18. IEEE.

PRENSKY, M., 2001, "The Digital Game-Based Learning Revolution". In: *Digital Game-Based Learning*, chap. 1, McGraw-Hill. Available at: <`http://www.marcprensky.com/writing/Prensky%20-%20Ch1-Digital%20Game-Based%20Learning.pdf`>.

RAMADAN, R., WIDYANI, Y., 2013, "Game development life cycle guidelines", *2013 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, pp. 95–100.

RIEBISCH, M., BÖLLERT, K., STREITFERDT, D., et al., 2002, "Extending feature diagrams with UML multiplicities". In: *6th World Conference on Integrated Design & Process Technology (IDPT2002)*, v. 23, pp. 1–7. Citeseer.

RINCÓN, L., MARTÍNEZ, J.-C., PABÓN, M. C., et al., 2018, "Creating a software product line of mini-games to support language therapy". In: *Advances in Computing: 13th Colombian Conference, CCC 2018, Cartagena, Colombia, September 26–28, 2018, Proceedings 13*, pp. 418–431. Springer.

RODRIGUES, C. S. C., NOBRE, T. S., NAZARETH, V., et al., 2024, "ReqJourney: Gamified Tool in the 2D Metaverse for Teaching Requirements Engineering". In: *Workshop sobre Interação e Pesquisa de Usuários no Desenvolvimento de Jogos (WIPlay)*, pp. 64–74. SBC.

SALEN, K., ZIMMERMAN, E., 2004, *Rules of Play: Game Design Fundamentals.* Massachusetts, EUA, MIT press.

SANTANA, E. F., DE OLIVEIRA, R. P., DO PRADO, A. F., et al., 2009, "Domain-Specific Modeling of Software Product Lines in Ubiquitous Computing". In: *Simpósio Brasileiro de Componentes, Arquiteturas e Reutilização de Software (SBCARS)*, pp. 209–222. SBC. (In Portuguese).

SANTOS, V. G. C. P., 2020, "Privacidade violada: os limites éticos do uso de dados na propaganda digital", *Trabalho de Conclusão de Curso (Bacharelado em Administração)-Faculdade de Administração e Ciências Contábeis, Universidade Federal do Rio de Janeiro, Rio de Janeiro.*

SARINHO, V. T., DE AZEVEDO, G. S., BOAVENTURA, F. M., 2018, "Askme: A feature-based approach to develop multiplatform quiz games". In: *2018*

17th Brazilian Symposium on Computer Games and Digital Entertainment (SBGames), pp. 38–3809. IEEE.

SCACCHI, W., 2011a, "Modding as an open source approach to extending computer game systems", *Proceedings of IFIP International Conference on Open Source Systems*, pp. 62–74.

SCACCHI, W., 2011b, "Modding as a basis for developing game systems", *Proceedings of the 1st international workshop on Games and software engineering*, pp. 5–8.

SCACCHI, W., 2017, "Practices and technologies in computer game software engineering", *IEEE Software*, v. 34, n. 1, pp. 110–116. Publisher: IEEE.

SCHELL, J., 2008, *The Art of Game Design: A book of lenses.* Florida, EUA, CRC press.

SILVA, F. P., 2016. "A metamodel-based approach for feature representation and modeling in dynamic software product lines". Masters Dissertation, Universidade Federal de Goiás, Brasil. (In Portuguese).

SINNAYAH, P., SALCEDO, A., REKHARI, S., 2021, "Reimagining physiology education with interactive content developed in H5P", *Advances in Physiology Education*, v. 45, n. 1, pp. 71–76. Publisher: American Physiological Society Rockville, MD.

SÖBKE, H., CHAN, E., VON BUTTLAR, R., et al., 2014, "Cat King's Metamorphosis: The Reuse of an Educational Game in a Further Technical Domain". In: *Games for Training, Education, Health and Sports: 4th International Conference on Serious Games, GameDays 2014, Darmstadt, Germany, April 1-5, 2014. Proceedings 4*, pp. 12–22. Springer.

TAMLA, P., BOHM, T., NAWROTH, C., et al., 2019, "What do serious games developers search online? A study of GameDev StackExchange", *Proceedings of the 5th Collaborative European Research Conference (CERC 2019). CEUR-WS.org*, v. 2348, pp. 131–142.

TAVARES, D., SILVA, C., ROUSY, D., 2014, "A feature model for health-focused serious games: a software product line-based approach". In: *CIbSE*, pp. 520–532. (In Portuguese).

UNGER, A., 2012, "Modding as part of game culture", *Computer Games and New Media Cultures*, pp. 509–523.

VAN GURP, J., BOSCH, J., SVAHNBERG, M., 2001, "On the notion of variability in software product lines". In: *Proceedings Working IEEE/IFIP Conference on Software Architecture*, pp. 45–54. IEEE.

WESTERA, W., 2019, "Serious Gaming Coming of Age: Implementing a European Innovation Policy to Amplify Serious Game Development." In: *GamiLearn*.

WOHLIN, C., 2014, "Guidelines for snowballing in systematic literature studies and a replication in software engineering". In: *Proceedings of the 18th international conference on evaluation and assessment in software engineering*, pp. 1–10.

XEXÉO, G., CARMO, A., ACIOLI, A., et al., 2013, "What are games?" *LUDES. Rio de Janeiro*, v. 1, pp. 1–30. (in Portuguese).

XEXÉO, G., CARMO, A., ACIOLI, A., et al., 2017, "What are games: An introduction to ludes object of study", *Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brasil.* (In Portuguese).

Appendix A

Additional Information from the Literature Review

A.1 Summary of the Research Protocol

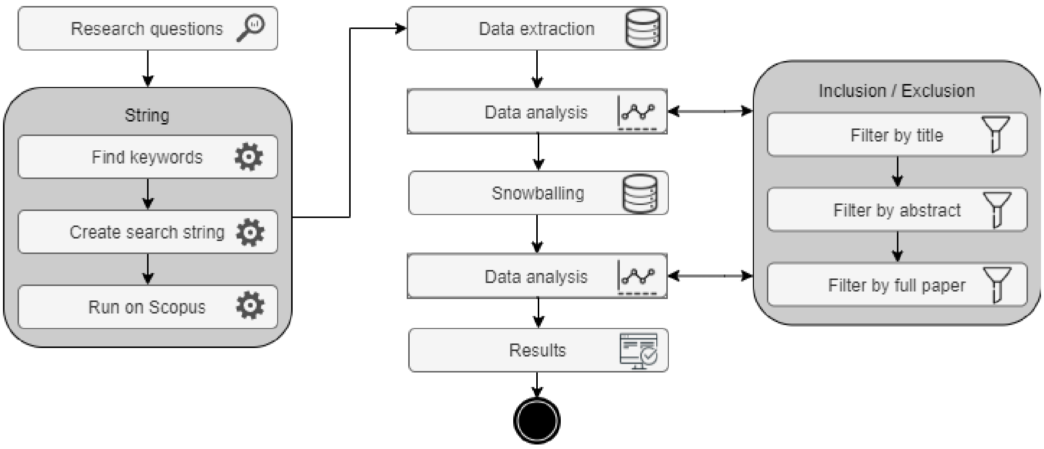


Figure A.1: Summary of the research protocol based on the model created by CALDERÓN *et al.*

.

Implementation procedure

1. Execute the search string;
2. Apply the inclusion / exclusion criteria based on the title;
3. Apply the inclusion / exclusion criteria based on the abstract;
4. Apply the inclusion / exclusion criteria based on the full text;
5. Apply snowballing backward; and
6. Apply snowballing forward

Inclusion criteria

1. The paper must be in the context of game development through SPL;
2. The paper must provide data to answer at least one of the research questions;
3. The paper must be written in English.

Exclusion Criteria

1. Conference call;
2. Studies that can not be fully accessed;
3. Studies that are not in Computer Science or Engineering;
4. Studies that are not related to digital games.

Research questions

1. How and what features are used in SPL to develop EVGs?
2. Why and when should SPL be used to develop EVGs?
3. Who and what current tools use SPL to develop EVGs?
4. How much effort/cost is involved in developing EVGs through SPL?

Table A.1: Search String.

P	Game*, gami*, play*, edutainment
I	Software Product Line, SPL, Software Reuse
C	Unused dimension
O	Tool*, engine*, mechanism*, application*, app*,platform*, instrument*, system*, console
C	Unused dimension
TITLE-ABS-KEY ((game* OR gami* OR play* OR edutainment) AND ("Software Product Line" OR spl OR "Software Reuse") AND (tool* OR engine* OR mechanism* OR application* OR app* OR platform* OR instrument* OR system* OR console)) AND (PUBYEAR > 2014 AND PUBYEAR < 2025) AND (LIMIT-TO (SUBJAREA , "COMP") OR LIMIT-TO (SUBJAREA , "ENGI"))	
Control papers	1 - Creating a software product line of mini-games to support language therapy

	2 - AsKME: A Feature-Based Approach to Develop Multiplatform Quiz Games

A.2 Papers analyzed in the Literature Review

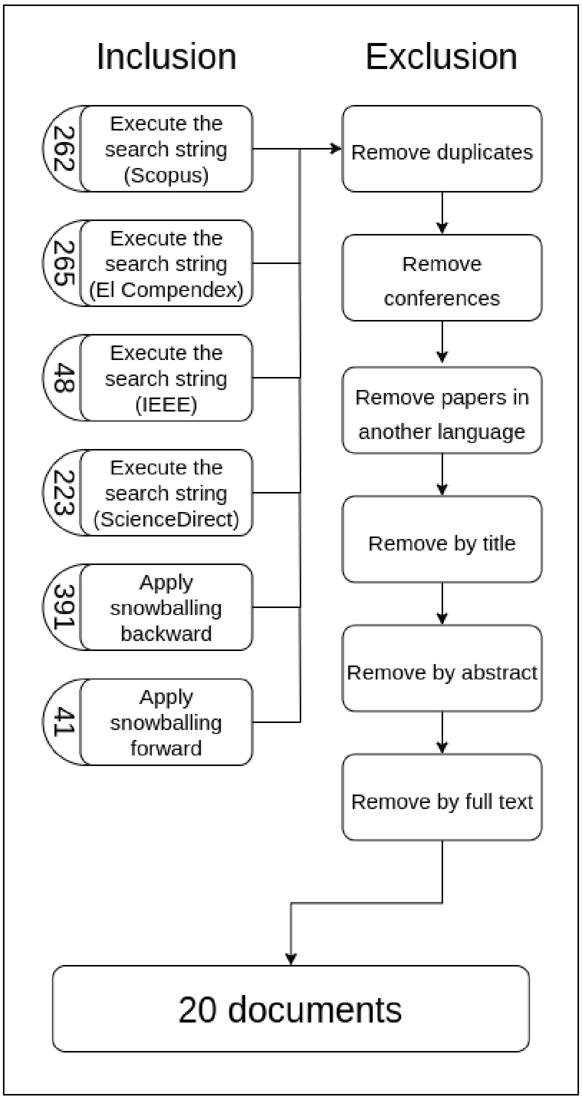


Figure A.2: Search flow of research on games and SPL approach.

Table A.2: Papers selected for the study.

Title	Reference	Q1	Q2	Q3	Q4
Lightweight Aspect-Oriented Software Product Lines with Automated Product Derivation	Perdek and Vranic (PERDEK and VRANIĆ, 2023)	X	X		

A Multi-engine Aspect-Oriented Language with Modeling Integration for Video Game Design	Geisler and Kavage (GEISLER and KAVAGE, 2021)	X	X		
SATReLO: A tool to support language therapies for children with hearing disabilities using video games	Martínez Arias et al. (MARTÍNEZ ARIA *et al.*, 2021)	X	X	X	
Recovering Software Product Line Architecture of Product Variants Developed with the Clone-and-Own Approach	Lee et al (LEE *et al.*, 2020)	X	X		
Serious gaming coming of age: Implementing a European innovation policy to amplify serious game development	Westera (WESTERA, 2019)	X	X		X
What do serious games developers search online? a study of GameDev StackExchange	Tamla et al (TAMLA *et al.*, 2019)	X	X		
A Feature-Based Approach to Develop Digital Board Games	Boaventura and Sarinho (BOAVENTURA and SARINHO, 2019)	X	X	X	
Using Software Product Lines to Support Language Rehabilitation Therapies: An Experience Report	Martinez et al. (MARTÍNEZ *et al.*, 2018)	X	X	X	
Building educational games from a feature model	Martins et al. (MARTINS *et al.*, 2018)	X	X		X
AsKME: A Feature-Based Approach to Develop Multiplatform Quiz Games	Sarinho et al. (SARINHO *et al.*, 2018)	X	X	X	
Exploration of software product line to enrich the modeling of mobile serious games	Meftah et al. (MEFTAH *et al.*, 2018)	X	X	X	
Software product line extraction from variability-rich systems: The robocode case study	Martínez et al. (MARTINEZ *et al.*, 2018)	X		X	X

Creating a software product line of mini-games to support language therapy	Rincon et al. (RINCÓN *et al.*, 2018)	X	X		
Practices and Technologies in Computer Game Software Engineering	Scacchi (SCACCHI, 2017)	X	X		
JIndie: A Software Product Line for Educational Games with a Focus on Constructionism (In Portuguese)	Lessa Filho and Hernández-Domínguez (LESSA FILHO and HERNÁNDEZ-DOMÍNGUEZ, 2016)	X	X	X	X
Introducing computer games and software engineering	Cooper and Scacchi (COOPER and SCACCHI, 2015)		X	X	
BROAD-PLG: Computational Model for Building Educational Games (In Portuguese)	Martins et al. (MARTINS *et al.*, 2015)	X			
Brain points: A growth mindset incentive structure boosts persistence in an educational game	O'Rourke et al. (O'ROURKE *et al.*, 2014)	X	X		
A feature model for health-focused serious games: a software product line-based approach (In Portuguese)	Tavares et al. (TAVARES *et al.*, 2014)	X	X		
Cat King's metamorphosis: The reuse of an educational game in a further technical domain	Sobke et al. (SÖBKE *et al.*, 2014)	X	X		X

Appendix B

TAM + MEEGA Questionnaire (English version)

B.1 Job description

Games is one of the industries that has grown significantly over the years, attracting enthusiasts of all ages, genres, and tastes and reaching a community of billions of consumers. However, game development can be time-consuming, with numerous participants and stages, which makes some titles to take years to complete. With such a large community, some customers cannot wait that long for the game to be released. As a result, they end up making his/her own versions of the game; this process of modifying an existing game to make a new one is known as a mod.

Although the development of mods is common in the gaming community, a study revealed some difficulties in the process, which stand out: the lack of specialized tools for building mods, the difficulty of understanding the original game's source code, and, at times, the need to recreate the original game from scratch.

The mod concept is very similar to the concept of opportunistic software reuse, in which specific software is copied and modified. Through a study on games and software reuse it was possible to conclude that Software Product Line would be one of the most recommended approaches for building mods. As a result, the goal of this work is to demonstrate the concept of a product line for building games using two existing games. The first aims to generate a new game automatically modifying the game's mechanics, dynamics, and aesthetics. The second lets the player design his/her own game by combining mechanics, dynamics, and aesthetics.

Basic commands and notes:

- Game 1
  - Use the joystick to move the character

- Avoid hitting obstacles

- Game 2
  - Within the feature tree, select all desired mechanics, dynamics, and aesthetics for the game. If you have two options for a trait, choose at least one.
  - Use the joystick to move the character
  - To attack or hold the boxes, press the buttons on the right.
  - To jump, press the left side button.
  - Complete the game objectives outlined on the splash screen to win.

B.2 Characterization questionnaire

Please answer the following questions based on your experience with the games. All data collected will be used exclusively for research purposes and will be published completely anonymously, without compromising the participant.

1. **Educational background:**
   - ☐ PhD.
   - ☐ PhD student.
   - ☐ Master's degree.
   - ☐ Master's student.
   - ☐ Graduate.
   - ☐ Graduate student.
   - ☐ Others (please specify): ____________________

2. **Age group:**
   - ☐ Less than 18 years
   - ☐ 18-28 years
   - ☐ 29-39 years
   - ☐ 40-50 years
   - ☐ Over 50 years

3. **Experience**

Consider: (1) no experience with the activity; (2) theoretical knowledge but no practice; (3) personal or classroom projects; and (4) industry projects. (5) Has extensive knowledge

	1	2	3	4	5
Digital games	☐	☐	☐	☐	☐
Game development	☐	☐	☐	☐	☐
Mods	☐	☐	☐	☐	☐
Software Reuse	☐	☐	☐	☐	☐
Software product line	☐	☐	☐	☐	☐

4. Time experience

Please provide details in your answer. Include how many months of experience you have in each of the knowledge areas.

Technology	Months
Digital games	
Game development	
Mods	
Software Reuse	
Software product line	

B.3 Evaluation questionnaire

5. Usability

	1	2	3	4	5
The game design is attractive (board, cards, interfaces, graphics, etc).	☐	☐	☐	☐	☐
Texts, colors and fonts match and are consistent.	☐	☐	☐	☐	☐
Learning to play this game was easy for me.	☐	☐	☐	☐	☐
I think that the game is easy to play.	☐	☐	☐	☐	☐
The game rules are clear and easy to understand.	☐	☐	☐	☐	☐
The fonts (size and style) used in the game are easy to read.	☐	☐	☐	☐	☐
The colours used in the game are meaningful.	☐	☐	☐	☐	☐

6. Experience

	1	2	3	4	5
This game is appropriately challenging for me.	☐	☐	☐	☐	☐
The game provides new challenges (offers new obstacles, situations, or variations) at an appropriate pace.	☐	☐	☐	☐	☐
The game does not become monotonous as it progresses (repetitive or boring tasks).	☐	☐	☐	☐	☐
I would recommend this game to my colleagues.	☐	☐	☐	☐	☐
I had fun playing the game.	☐	☐	☐	☐	☐
There was something interesting at the beginning of the game that captured my attention.	☐	☐	☐	☐	☐
I was so involved in my gaming task that I lost track of time.	☐	☐	☐	☐	☐
I forgot about the environment around me while playing this game.	☐	☐	☐	☐	☐
The game contents are relevant to my interests.	☐	☐	☐	☐	☐

7. Utility of the proposed tool:

	1	2	3	4	5
Did I easily understand how to use the SPL approach?	☐	☐	☐	☐	☐
Did I apply the strategy correctly? I designed the games I want to play.	☐	☐	☐	☐	☐
Do I understand what happened in the interaction with the tool?	☐	☐	☐	☐	☐
Have I noticed how simple it is to create a new game using Product Line?	☐	☐	☐	☐	☐
Would I use a tool to expand games if one were to be proposed?	☐	☐	☐	☐	☐

8. Have you identified any positive or negative aspects of using the game in your opinion? If so, which one(s)?

____

____

9. Do you have any idea on how to improve the game or the platform? If so, please explain.

10. **This question is for any additional comment (difficulties, criticisms, and/or suggestions) about the study. We depend on your help to improve the work.**

Thanks for your collaboration!

Diego Cardoso Borda Castro
Cláudia Maria Lima Werner

Appendix C

TAM + MEEGA Questionnaire (Portuguese version)

C.1 Descrição do trabalho

Uma das indústrias que mais vem crescendo ao longo dos anos é a de jogos, atraindo entusiastas de todas as idades, gêneros e gostos, chegando a ter uma comunidade de bilhões de consumidores. No entanto, o desenvolvimento de jogos pode ser algo demorado, com muitos participantes e etapas, o que faz com que alguns títulos levem anos até sua entrega final. Com uma comunidade de entusiastas tão grande, alguns consumidores não conseguem esperar tanto tempo até o lançamento do jogo. Devido a isso, acabam criando suas próprias versões do jogo. Esse procedimento de utilizar um jogo já existente para construção de um novo é conhecido como mod.

Apesar da construção de mods ser algo recorrente na comunidade de jogos, através de um estudo realizado, foi possível perceber algumas dificuldades nesse processo, onde se destacam: a falta de ferramentas especializadas para construção de mods, a dificuldade de entender o código fonte do jogo original ou até mesmo, em alguns momentos, a necessidade de recriar o jogo original do zero.

O conceito de mod se assemelha muito com o conceito de Reutilização de Software oportunista, onde um determinado software é copiado e modificado. Tendo isso em mente, foi realizado um estudo sobre jogos e Reutilização de Software, onde a abordagem de Linha de produto de Software se destacou entre as demais para a construção de mods. Devido a isso, esse trabalho visa demonstrar o conceito de linha de produto para construção de jogos por meio de dois jogos disponibilizados. O primeiro visa criar um novo jogo de forma automática de tempos em tempos através da modificação das mecânicas, dinâmicas e estéticas do jogo. O segundo permite que o próprio jogador construa seu jogo por meio da seleção das mecânicas, dinâmicas e estéticas.

Comandos básicos e observações:

- Jogo 1
  - Utilize o joystick para movimentar o personagem
  - Evite bater nos obstáculos
- Jogo 2
  - Selecione todas as mecânicas, dinâmicas e estéticas desejadas para o jogo dentro da árvore de características. Se tiver duas opções para uma características, lembre-se de selecionar pelo menos uma.
  - Utilize o joystick para movimentar o personagem
  - Utilize os botões do lado direito para atacar ou segurar as caixas
  - Utilize o botão do lado esquerdo para pular
  - Para vencer, cumpra os objetivos do jogo descritos na tela inicial

C.2 Questionário de caracterização

Por favor, responda as questões abaixo com base na experiência que obteve ao utilizar os jogos. Todos os dados coletados são apenas para melhoria da pesquisa e serão publicados de forma totalmente anônima, não comprometendo o participante.

11. Formação Acadêmica:

☐ Doutorado concluído

☐ Doutorado em andamento

☐ Mestrado concluído

☐ Mestrado em andamento

☐ Graduação concluída

☐ Graduação em andamento

☐ Outro (Qual): ____________________

12. Faixa etária:

☐ Menos de 18 anos

☐ 18 a 28 anos

☐ 29 a 39 anos

- ☐ 40 a 50 anos
- ☐ Mais de 50 anos

13. Experiência

Conside: (1) Sem experiência com a atividade; (2) Possui conhecimento teórico, sem prática; (3) Pratiquei em projetos em pessoais ou em sala de aula; (4) Pratiquei em projetos na indústria; (5) Possui um vasto conhecimento

	1	2	3	4	5
Jogos digitais	☐	☐	☐	☐	☐
Desenvolvimento de jogos	☐	☐	☐	☐	☐
Mods	☐	☐	☐	☐	☐
Reutilização de Software	☐	☐	☐	☐	☐
Linha de produto de Software	☐	☐	☐	☐	☐

14. Tempo de experiência

Por favor, detalhe sua resposta. Inclua o número de meses de experiência para cada uma das áreas de conhecimento.

Tecnologia	Meses
Jogos digitais	
Desenvolvimento de jogos	
Mods	
Reutilização de Software	
Linha de produto de Software	

C.3 Questionário de avaliação

15. Usabilidade

	1	2	3	4	5
O design do jogo é atraente (tabuleiro, cartas, interfaces, gráficos, etc.).	☐	☐	☐	☐	☐
Os textos, cores e fontes combinam e são consistentes.	☐	☐	☐	☐	☐
Eu precisei aprender algumas coisas antes que eu pudesse jogar o jogo.	☐	☐	☐	☐	☐
Eu considero que o jogo é fácil de jogar	☐	☐	☐	☐	☐
As regras do jogo são claras e compreensíveis.	☐	☐	☐	☐	☐

As fontes (tamanho e estilo) utilizadas no jogo são legíveis.	☐	☐	☐	☐	☐
As cores utilizadas no jogo são compreensíveis.	☐	☐	☐	☐	☐

16. Experiência

	1	2	3	4	5
Este jogo é adequadamente desafiador para mim.	☐	☐	☐	☐	☐
O jogo oferece novos desafios (oferece novos obstáculos, situações ou variações) com um ritmo adequado.	☐	☐	☐	☐	☐
O jogo não se torna monótono nas suas tarefas (repetitivo ou com tarefas chatas).	☐	☐	☐	☐	☐
Eu recomendaria este jogo para meus colegas.	☐	☐	☐	☐	☐
Eu me diverti com o jogo.	☐	☐	☐	☐	☐
Houve algo interessante no início do jogo que capturou minha atenção.	☐	☐	☐	☐	☐
Eu estava tão envolvido no jogo que eu perdi a noção do tempo.	☐	☐	☐	☐	☐
Eu esqueci sobre o ambiente ao meu redor enquanto jogava este jogo.	☐	☐	☐	☐	☐
O conteúdo do jogo é relevante para os meus interesses.	☐	☐	☐	☐	☐

17. Utilidade da ferramenta a ser proposta:

	1	2	3	4	5
Eu compreendi facilmente como usar a abordagem de SPL?	☐	☐	☐	☐	☐
Eu usei a abordagem da maneira correta? Criei os jogos que gostaria	☐	☐	☐	☐	☐
Compreendi o que aconteceu na interação com a ferramenta?	☐	☐	☐	☐	☐
Eu notei a facilidade de criar um novo jogo por meio de Linha de produto?	☐	☐	☐	☐	☐
Caso existisse a ferramenta a ser proposta, eu usaria uma ferramenta dessa para expandir jogos?	☐	☐	☐	☐	☐

18. De acordo com sua opinião, foi identificado algum aspecto positivo / negativo da utilização do jogo? Se sim, qual(ais)?

19. **Você possui alguma sugestão para melhoria do jogo ou da plataforma? Em caso positivo, por favor, especifique-a.**

20. **Este espaço é reservado para quaisquer comentários adicionais (dificuldades, críticas e/ou sugestões) a respeito do estudo executado. Contamos com sua contribuição para que o trabalho seja aprimorado.**

Obrigado pela sua colaboração!

Diego Cardoso Borda Castro
Cláudia Maria Lima Werner

Appendix D

Survey

D.1 Support text for the respondent

Educational games (EGs) are an effective tool in the pedagogical process, combining enjoyment with education to improve student engagement while increasing knowledge retention. These games provide a novel and creative method of delivering content, enabling students to develop their cognitive and creative abilities in a playful environment.

A Software Product Line (SPL) is an approach to software development that focuses on the advantages of decreased costs, time, and effort in the development of a range of related products. It aims to classify software with shared attributes and implement modifications through variation points, defined as variation points for a specific segment of the product. This enables the creation of a product tree that includes comparable characteristics, with specific components subject to modification. This research aims to investigate the connection between both of these fields and the utility of the SPL in the production of EGs, while also identifying the related advantages and challenges.

Your participation is crucial to improving comprehension of these issues and for promoting the creation of effective and accessible educational solutions for everyone.

D.2 Characterization questionnaire

21. **What is your research area?**

    ____

    ____

22. Educational background:

☐ PhD.

☐ PhD student.

☐ Master's degree.

☐ Master's student.

☐ Graduate.

☐ Graduate student.

☐ Others (please specify): ____________________

23. Age group:

☐ Less than 18 years

☐ 18-28 years

☐ 29-39 years

☐ 40-50 years

☐ Over 50 years

24. How many years of experience do you have with educational games?

☐ I have no experience in the area

☐ 1 to 3 years

☐ more than 3 to 5 years

☐ more than 5 to 10 years

☐ More than 10 years

25. How many years of experience do you have in Software Product Line?

☐ I have no experience in the area

☐ 1 to 3 years

☐ more than 3 to 5 years

☐ more than 5 to 10 years

☐ More than 10 years

26. How many product lines have you developed?

0	1	2	3	4	5	6	7	8	9	10

27. How many product lines have you developed for educational games?

0	1	2	3	4	5	6	7	8	9	10

D.3 Specific questionnaires

28. Are you familiar of any educational game product lines? Would you be able to provide any websites or references that refer to them?

____

____

29. What is the biggest challenge in creating a product line for educational games?

____

____

Select as many features as you deem necessary.

30. What is the motivation for using a product line for educational games?

- ☐ Reduce development time for educational games
- ☐ Creation of a product line for this purpose is irrelevant
- ☐ To develop similar games with the goal of reducing the learning curve for students
- ☐ other

31. What aesthetics (visual component) should an SPLfor educational games possess?

- ☐ Must have multiple environments for different games
- ☐ Must have a single environment for all games
- ☐ Must have only one type of game, either 2D or 3D
- ☐ Must have the ability to create 2D and 3D games

☐ other

32. For educational games, which mechanics should an SPL consist of?

☐ Must have multiplayer option

☐ Must have only one of the options, multiplayer or single

☐ Must have multiple rules for different types of games

☐ Must have specialized rules for all games

☐ Must have rewards and score

☐ Must have optional rewards and points

☐ Must have error feedback systems for the player

☐ Must have optional error feedback systems for the player

☐ other

33. What educational mechanics should an SPL for educational games have?

☐ Must track player learning progression

☐ The player's learning progression should be optionally monitored

☐ It is optional to include educational activities, such as puzzles, examinations, and sorting games

☐ It must have educational activities such as puzzles, quizzes, sorting games, etc

☐ It is optional to enhance reinforcement learning

☐ Reinforcement learning must be enhanced

☐ other

34. What extra features should an SPL for educational games have?

☐ Developing for a variety of platforms, including mobile, computer, and console, can complicate the process of line creation

☐ It should be possible to create the game for different platforms such as mobile, computer and console

☐ The product line must not provide a narrative option for gameplay

☐ The product line must provide a game narrative option

☐ Must have multiple genres for games

☐ Must have a single genre for games

☐ other

35. Would you like to contribute with any suggestions or comments?

Appendix E

Survey (Portuguese version)

E.1 Texto de apoio para o respondente

Os Jogos Educacionais (JEs) podem ser uma poderosa ferramenta no processo de ensino, pois combinam diversão e educação para aumentar o engajamento do aluno, enquanto tentam ajudar na retenção de conhecimento. Esses jogos oferecem uma maneira diferente e inovadora de apresentar conteúdos, permitindo que os alunos desenvolvam suas habilidades cognitivas e criativas em um ambiente lúdico.

Linha de Produtos de Software (LPS) é uma abordagem de desenvolvimento de software que tem como principal vantagem a redução dos custos, tempo e esforço necessários para se criar uma família de produtos semelhantes. Ela visa agrupar softwares com características comuns e fazer modificações nos mesmos por meio de pontos de variação, que podem ser compreendidos como alterações que são feitas em uma parte específica do produto. Por meio disso, é possível criar uma árvore de produtos que possuem características semelhantes, mas que certas partes podem ser modificadas.

O objetivo desta pesquisa é examinar a relação entre essas duas áreas e como a abordagem de LPS pode ser útil no desenvolvimento de JEs, bem como encontrar os benefícios e problemas associados.

Sua participação é essencial para melhorar a compreensão desses problemas e ser possível avançar no desenvolvimento de soluções educacionais viáveis e acessíveis para todos.

E.2 Questionário de caracterização

36. Qual sua área de pesquisa?

____

____

37. Formação Acadêmica:

☐ Doutorado concluído

☐ Doutorado em andamento

☐ Mestrado concluído

☐ Mestrado em andamento

☐ Graduação concluída

☐ Graduação em andamento

☐ Outro (Qual): ____________________

38. Faixa etária:

☐ Menos de 18 anos

☐ 18 a 28 anos

☐ 29 a 39 anos

☐ 40 a 50 anos

☐ Mais de 50 anos

39. Quantos anos de experiência você possui com jogos educacionais?

☐ Não possuo experiência na área

☐ 1 a 3 anos

☐ mais de 3 a 5 anos

☐ mais de 5 a 10 anos

☐ Mais de 10 anos

40. Quantos anos de experiência você possui em Linha de produtos de Software?

☐ Não possuo experiência na área

☐ 1 a 3 anos

☐ mais de 3 a 5 anos

☐ mais de 5 a 10 anos

☐ Mais de 10 anos

41. **Quantas Linhas de produtos você já desenvolveu?**

0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**

42. **Quantas Linhas de produtos você já desenvolveu para jogos educacionais?**

0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**

E.3 Questionários específico

43. **Conhece alguma linha de produtos para jogos educacionais? Poderia citar sites ou referências delas?**

____

____

44. **Qual a maior dificuldade de criação de uma linha de produtos para jogos educacionais?**

____

____

Selecione quantas características julgar necessário.

45. **Qual a motivação para utilizar uma Linha de produtos para jogos educacionais?**
    - ☐ Diminuir o tempo de desenvolvimento de jogos educacionais
    - ☐ Criar um linha de produtos para esse propósito não é relevante
    - ☐ Para criar jogos semelhantes com a intenção de diminuir a curva de apresendizado dos alunos
    - ☐ outros

46. **Quais estéticas (parte visual) uma LPS para jogos educacionais deve possuir?**
    - ☐ Deve possuir varios ambientes para diversos jogos
    - ☐ Deve possuir um único ambiente para todos os jogos
    - ☐ Deve possuir apenas um tipo de jogo, sendo 2D ou 3D
    - ☐ Deve possuir a possibilidade de criar jogos 2D e 3D

☐ outros

47. Quais mecânicas uma LPS para jogos educacionais deve possuir?

☐ Deve possuir opção de multiplayer

☐ Deve possuir apenas uma das opções, multiplayer ou single

☐ Deve possuir várias regras para diferentes tipos de jogos

☐ Deve possuir regras especializadas para todos os jogos

☐ Deve possuir recompensas e pontuação

☐ Deve possuir recompensas e pontuação de maneira opcional

☐ Deve possuir sistemas de feedback dos erros para o jogador

☐ Deve possuir sistemas de feedback dos erros para o jogador de forma opcional

☐ outros

48. Quais mecânicas educacionais uma LPS para jogos educacionais deve possuir?

☐ Deve rastrear a progressão de aprendizado do jogador

☐ Deve rastrear a progressão de aprendizado do jogador de forma opcional

☐ Deve possuir atividade educacionais como puzzle, quiz, jogos de ordenação, etc de forma opcional

☐ Deve possuir atividade educacionais como puzzle, quiz, jogos de ordenação, etc

☐ Deve fortalecer o aprendizado por reforço de forma opcional

☐ Deve fortalecer o aprendizado por reforço

☐ outros

49. Quais características extras uma LPS para jogos educacionais deve possuir?

☐ O desenvolvimento para diversas plataformas como celular, computador e console pode dificultar a crição da linha

☐ Deve ser possivel criar o jogo para diversas plataformas como celular, computador e console

☐ A linha de produto não deve fornecer uma opção de narrativa para jogo

☐ A linha de produto deve fornecer uma opção de narrativa do jogo

☐ Deve possuir varios gêneros para os jogos

☐ Deve possuir um único gênero para os jogos

☐ outros

50. Gostaria de contribuir com alguma sugestão ou comentários?

____

____

Appendix F

Generic Model Questionnaire (Portuguese version)

F.1 Descrição geral

Linha de Produto de Software (LPS) constitui uma abordagem no âmbito da pesquisa sobre Reutilização de Software, cuja finalidade é integrar um conjunto de estratégias, técnicas e ferramentas destinadas ao desenvolvimento sistemático de sistemas que compartilham um núcleo comum, embora apresentem características distintas. As principais vantagens consistem na diminuição do tempo necessário para o desenvolvimento, bem como na facilidade para a manutenção e alteração do produto.

Essa abordagem é comumente estruturada por meio de modelos de características, os quais se configuram em árvores que ilustram todas as características do produto a ser desenvolvido, evidenciando suas particularidades por meio de ramificações conhecidas como pontos de variação. O modelo mais amplamente reconhecido é denominado *Feature-Oriented Domain Analysis* (FODA).

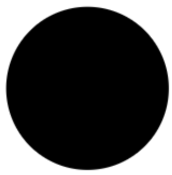
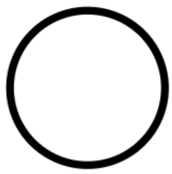
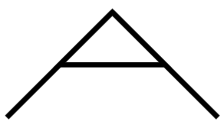
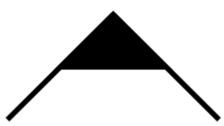
Apesar da já existência dos modelos de características, essa notação só demonstra em alto nível as propriedades do software a ser construído, não demonstrando informações suficientes para seu desenvolvimento. Diante disso, diagramas de classes são criados para representar tais informações. No entanto, o nível de abstração representado em um modelo de características e em um diagrama de classes é muito diferente, sendo necessária alguma etapa anterior para tornar essa transição menos brusca. Um método potencial que pode servir como um intermediário entre essas duas representações é um modelo genérico (MG).

Os MGs podem ser entendidos como uma abstração de alto nível da qual os modelos são derivados. Eles podem ser representados usando UML, sendo diagramados como um diagrama de classe ilustrando entidades, propriedades e associações, porém em um nível abstrato e não concreto. Após instanciar um modelo genérico, modelos são gerados, criando assim diagramas de classe concretos. Diante disso, uma série de regras foi construída visando possibilitar essa conversão de um modelo de características em um MG.

Para realizar a transformação de um modelo de características para um MG, um conjunto de regras deve ser analisado. A seguir, cada uma delas será descrita, no entanto, a primeira análise que deve ser feita é se cada uma das características é funcional ou conceitual.

- **Características funcionais**: representam comportamentos simples, únicos.
- **Características conceituais**: representam comportamentos mais complexos que abrangem diversas funcionalidades ou ações.

F.2 Elementos do modelo de características

Elemento	**Descrição**
	Característica obrigatória, indica que uma funcionalidade deve existir.
	Característica opcional, indica que uma funcionalidade pode ou não existir.
	Relação alternativa, indica uma relação entre um ponto de variação e suas variações, com cardinalidade igual a 1.
	Relação opcional, indica uma relação entre um ponto de variação e suas variações, com cardinalidade igual a 0 ou N.
	Relação simples, indica que uma característica está relacionada a outra.

	Relação de exclusão, indica que, se uma das características existir, a outra relacionada não pode existir.
	Relação de inclusão, indica que a característica inclusiva só pode existir se a outra característica incluída também existir.

F.3 Regras de Transformações

Table F.2: [Regra 1] Transformação de características em classes ou métodos.

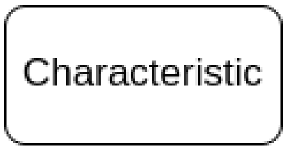
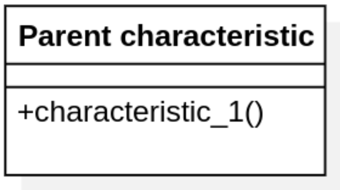
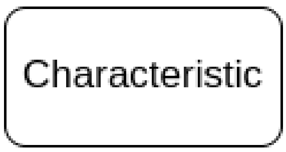
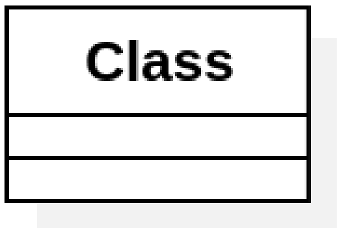
FODA	Generic Model
Transforme características funcionais em métodos dentro da caracterśitica pai que se tornará uma classe.	
	
Transforme características conceituais em classes.	
	

Table F.3: [Regra 2] Transformando relacionamentos simples em associações.

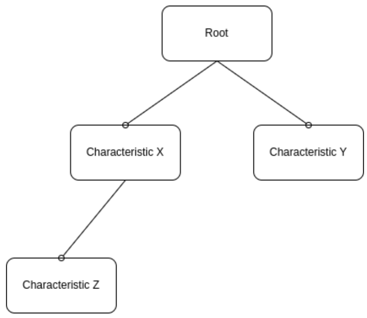
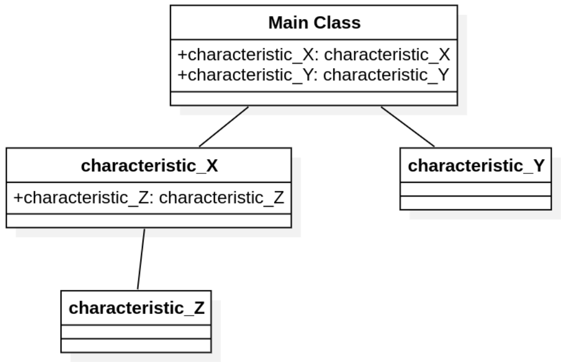
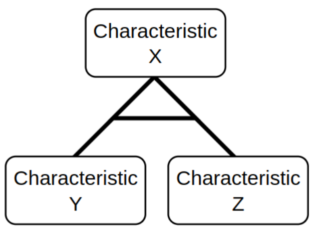
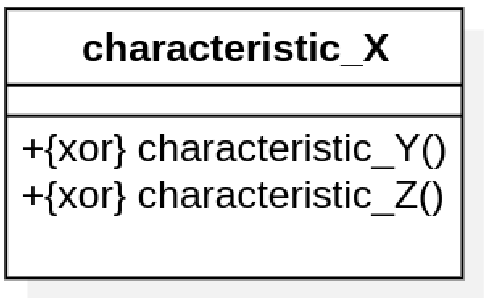
<table>
<tr><th>FODA</th><th>Generic Model</th></tr>
<tr><td>
</td><td>
</td></tr>
</table>

Table F.4: [Regra 3] Transformando pontos de variação alternativo em especialização/especialização.

<table>
<tr><th>FODA</th><th>Generic Model</th></tr>
<tr><td colspan="2">Transforme pontos de variação alternativos com características funcionais em classes com dois métodos.</td></tr>
<tr><td>
</td><td>
</td></tr>
<tr><td colspan="2">Transforme pontos de variação alternativos com características conceituais em classes com especialização.</td></tr>
</table>

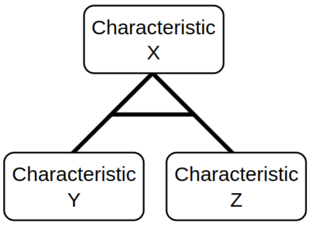
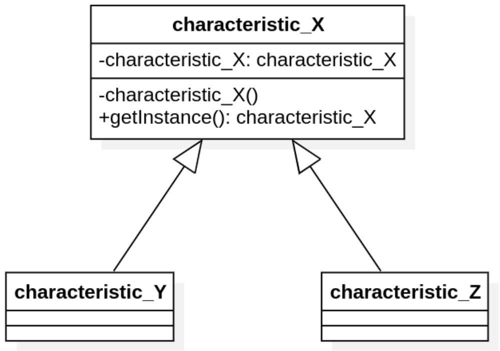
	

Table F.5: [Regra 4] Transformando pontos de variação opcionais em um relacionamento de composição.

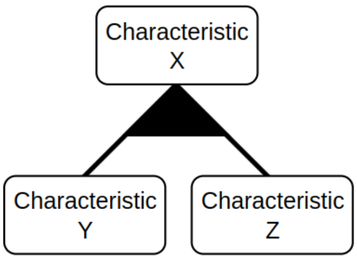
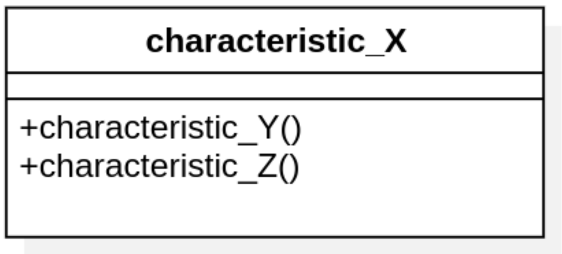
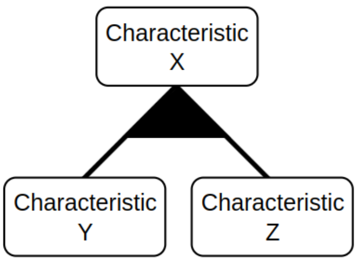
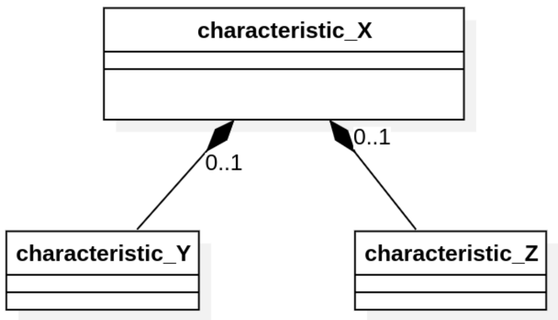
FODA	**Generic Model**
Transforme pontos de variação opcionais com características funcionais em classes com dois métodos.	
	
Transforme pontos de variação alternativos com características conceituais em classes com associação.	
	

Table F.6: [Regra 5] Transformando elementos obrigatórios em estereótipos.

FODA	Generic Model
	«Mandatory»

Table F.7: [Regra 6] Transformando uma relação de inclusão em uma relação de dependência.

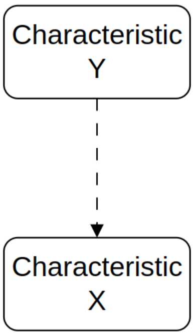
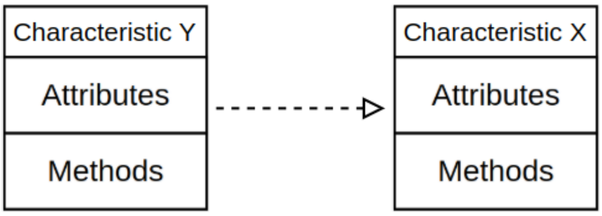
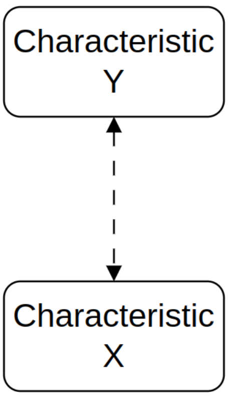
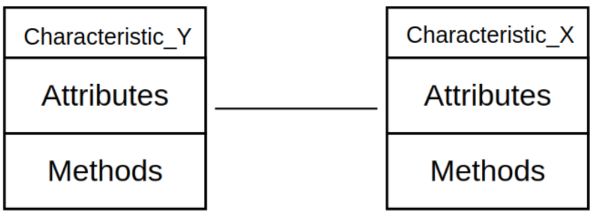
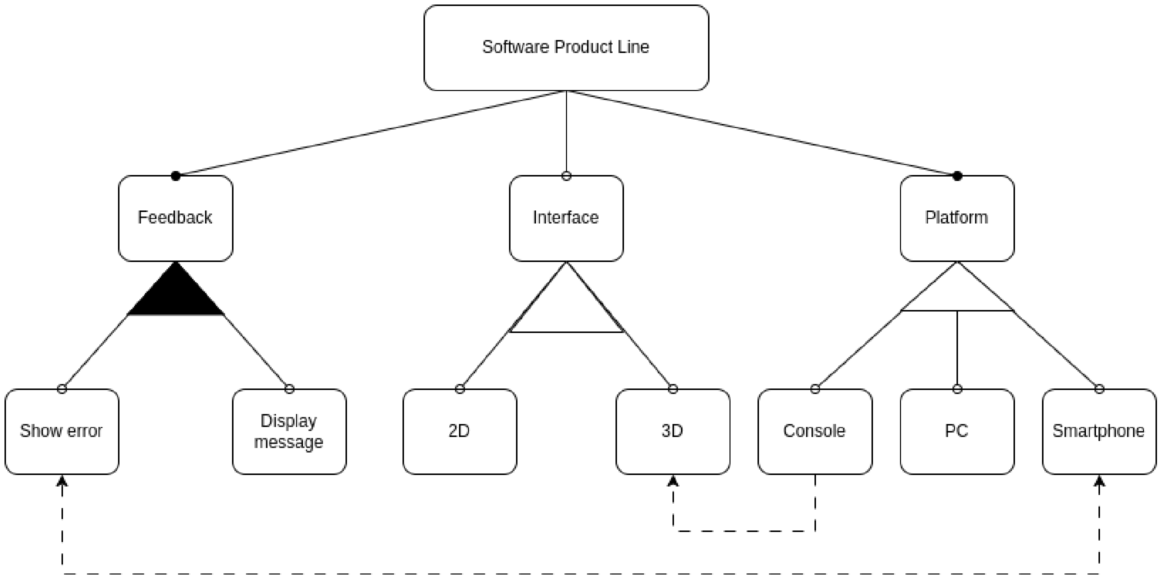
FODA	Generic Model
	

Table F.8: [Regra 7] Transformando uma relação de exclusão em uma associação com OCL.

FODA	Generic Model
	 **context** Characteristic_Y **inv Exclusion:** Characteristic_Y.allInstances()->size() = 0 or Characteristic_X.allInstances()->size() = 0

F.4 Atividades

Após contextualizar a necessidade do uso de MGs e estabelecer as orientações para a conversão dos modelos de características em MGs, execute a tarefa a seguir: converta a árvore de características apresentada a seguir em um MG, aplicando todas as normas previamente detalhadas. Em sequência, responda às questões a seguir.



F.5 Questionário de caracterização

Por favor, responda às questões abaixo com base na sua experiência. Todos os dados coletados são apenas para melhoria da pesquisa e serão publicados de forma totalmente anônima, não comprometendo o participante.

51. Formação Acadêmica:

- ☐ Doutorado concluído
- ☐ Doutorado em andamento
- ☐ Mestrado concluído
- ☐ Mestrado em andamento
- ☐ Graduação concluída
- ☐ Graduação em andamento
- ☐ Outro (Qual): ____________________

52. Faixa etária:

- ☐ Menos de 18 anos
- ☐ 18 a 28 anos

- ☐ 29 a 39 anos
- ☐ 40 a 50 anos
- ☐ Mais de 50 anos

53. Experiência

Conside: (1) Sem experiência com a atividade; (2) Possui conhecimento teórico, sem prática; (3) Pratiquei em projetos em pessoais ou em sala de aula; (4) Pratiquei em projetos na indústria; (5) Possui um vasto conhecimento

	1	2	3	4	5
Reutilização de Software	☐	☐	☐	☐	☐
Linha de produto de Software	☐	☐	☐	☐	☐
Modelo de características	☐	☐	☐	☐	☐
Feature-Oriented Domain Analysis	☐	☐	☐	☐	☐
GM	☐	☐	☐	☐	☐

54. Tempo de experiência

Por favor, detalhe sua resposta. Inclua o número de meses de experiência para cada uma das áreas de conhecimento.

Tecnologia	Meses
Reutilização de Software	
Linha de produto de Software	
Modelo de características	
Feature-Oriented Domain Analysis	
GM	

F.6 Questionário de avaliação

Conside: (1) Discordo totalmente; (2) Discordo; (3) Não concordo, nem discordo; (4) Concordo; (5) Concordo totalmente.

	1	2	3	4	5
Utilizar a transformação foi fácil para mim.	☐	☐	☐	☐	☐
Utilizar a transformação foi útil para me ajudar a compreender melhor o software a ser construído·	☐	☐	☐	☐	☐
Eu utilizaria a abordagem de transformação para desenvolver uma LPS	☐	☐	☐	☐	☐
As regras eram claras para mim	☐	☐	☐	☐	☐
Seguir cada uma das regras me ajudou a criar o GM de maneira simples	☐	☐	☐	☐	☐

55. Discorda de alguma transformação? Se sim, qual?

____

____

56. Acrescentaria alguma regra de transformação na abordagem? Se sim, qual?

____

____

57. Este espaço é reservado para quaisquer comentários adicionais (dificuldades, críticas e/ou sugestões) a respeito do estudo executado. Contamos com sua contribuição para que o trabalho seja aprimorado.

Obrigado pela sua colaboração!

Diego Cardoso Borda Castro
Cláudia Maria Lima Werner

Appendix G

Student Questionnaire (Portuguese version)

G.1 Descrição do trabalho

Por favor, responda às questões abaixo com base na experiência que obteve com a proposta de ideia do jogo. Todos os dados coletados são apenas para melhoria da pesquisa e serão publicados de forma totalmente anônima, não comprometendo o participante.

Comandos básicos e observações:

- Utilize as setas direcionais para movimentar o personagem;
- Utilize a letra "E" para interagir com os objetos e personagens;
- Utilize o mouse para direcionar o personagem e para selecionar as respostas das perguntas.

G.2 Questionário de caracterização

58. Formação Acadêmica:

☐ Doutorado concluído

☐ Doutorado em andamento

☐ Mestrado concluído

☐ Mestrado em andamento

☐ Graduação concluída

☐ Graduação em andamento

☐ Outro (Qual): ____________________

59. Faixa etária:

☐ Menos de 18 anos

☐ 18 a 28 anos

☐ 29 a 39 anos

☐ 40 a 50 anos

☐ Mais de 50 anos

60. Qual é a sua área de atuação?

____

____

61. Já teve aulas utilizando jogos? Conte sobre sua experiência?

____

____

62. Já desenvolveu algum jogo educacional? Qual foi sua maior dificuldade?

____

____

63. Utilidade

Considere: (1) Discordo totalmente; (2) Discordo; (3) Nem discordo, nem concordo; (4) Concordo; (5) Concordo totalmente.

	1	2	3	4	5
O design do jogo é atraente (tabuleiro, cartas, interfaces, gráficos, etc.).	☐	☐	☐	☐	☐
Os textos, cores e fontes combinam e são consistentes.	☐	☐	☐	☐	☐
Eu precisei aprender poucas coisas para poder começar a entender o jogo.	☐	☐	☐	☐	☐
Aprender a jogar este jogo parece fácil para mim.	☐	☐	☐	☐	☐
Eu acho que a maioria das pessoas aprenderia a jogar este jogo rapidamente.	☐	☐	☐	☐	☐
As regras do jogo são claras e compreensíveis.	☐	☐	☐	☐	☐
As fontes (tamanho e estilo) utilizadas no jogo são legíveis.	☐	☐	☐	☐	☐

As cores utilizadas no jogo são compreensíveis.	☐	☐	☐	☐	☐

64. Experiência

	1	2	3	4	5
A organização do conteúdo me ajudou a estar confiante de que eu iria aprender com este jogo.	☐	☐	☐	☐	☐
Este jogo é adequadamente desafiador para mim.	☐	☐	☐	☐	☐
O jogo oferece novos desafios (oferece novos obstáculos, situações ou variações) com um ritmo adequado.	☐	☐	☐	☐	☐
O jogo não se torna monótono nas suas tarefas (repetitivo ou com tarefas chatas).	☐	☐	☐	☐	☐
Completar as tarefas do jogo provoca um sentimento de realização.	☐	☐	☐	☐	☐
Me sinto satisfeito com as coisas que posso aprender no jogo.	☐	☐	☐	☐	☐
Eu recomendaria este jogo para meus colegas.	☐	☐	☐	☐	☐
Eu me diverti com o jogo.	☐	☐	☐	☐	☐
Houve algo interessante no início do jogo que capturou minha atenção.	☐	☐	☐	☐	☐
Eu estava tão envolvido no jogo que eu perdi a noção do tempo.	☐	☐	☐	☐	☐
Eu esqueci sobre o ambiente ao meu redor enquanto jogava este jogo.	☐	☐	☐	☐	☐
O conteúdo do jogo é relevante para os meus interesses.	☐	☐	☐	☐	☐
É claro para mim como o conteúdo do jogo está relacionado com o ensino.	☐	☐	☐	☐	☐
Eu prefiro aprender com este jogo do que de outra forma (outro método de ensino).	☐	☐	☐	☐	☐

65. De acordo com sua opinião, foi identificado algum aspecto positivo / negativo da utilização do jogo? Se sim, qual(ais)?

____

____

66. Você possui alguma sugestão para a melhoria do jogo? Em caso positivo, por favor, especifique-a.

67. Este espaço é reservado para quaisquer comentários adicionais (dificuldades, críticas e/ou sugestões) a respeito do estudo executado. Contamos com sua contribuição para que o trabalho seja aprimorado.

Obrigado pela sua colaboração!

Diego Cardoso Borda Castro
Cláudia Maria Lima Werner

Appendix H

Teacher Questionnaire (Portuguese version)

H.1 Descrição do trabalho

Por favor, responda às questões abaixo com base na experiência que obteve com a proposta de ideia do jogo. Todos os dados coletados são apenas para melhoria da pesquisa e serão publicados de forma totalmente anônima, não comprometendo o participante.

Comandos básicos e observações:

- Clique na área reservada para o professor e digite a senha;
- Escolha os tipos de questões desejadas para cada um dos andares;
- Formule as perguntas para um dos andares de desafio.

H.2 Questionário de caracterização

68. Formação Acadêmica:

☐ Doutorado concluído

☐ Doutorado em andamento

☐ Mestrado concluído

☐ Mestrado em andamento

☐ Graduação concluída

☐ Graduação em andamento

☐ Outro (Qual): ____________________

69. **Faixa etária:**
- ☐ Menos de 18 anos
- ☐ 18 a 28 anos
- ☐ 29 a 39 anos
- ☐ 40 a 50 anos
- ☐ Mais de 50 anos

70. **Qual é a sua área de atuação?**

____

____

71. **Quantos anos de experiência você possui com jogos educacionais?**
- ☐ Não posssuo experiência na área
- ☐ 1 a 3 anos
- ☐ mais de 3 a 5 anos
- ☐ mais de 5 a 10 anos
- ☐ Mais de 10 anos

72. **Quantas Linhas de produtos você já desenvolveu?**

0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**

73. **Quantas Linhas de produtos você já desenvolveu para jogos educacionais?**

0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**

74. **Já utilizou jogos para ensinar alguma disciplina? Se sim, quais?**

____

____

75. **Na sua opinião, qual a maior dificuldade de criar um LPS para jogos educacionais?**

____

____

76. **Na sua opinião, como poderíamos especializar essa LPS?**

____

____

H.3 Questionário de avaliação da proposta de ideia

77. Facilidade de Uso Percebida

Considere: (1) Discordo totalmente; (2) Discordo; (3) Nem discordo, nem concordo; (4) Concordo; (5) Concordo totalmente.

	1	2	3	4	5
O design do jogo é atraente (tabuleiro, cartas, interfaces, gráficos, etc.).	☐	☐	☐	☐	☐
A maneira de interação com a ferramenta é clara e facilmente compreendida.	☐	☐	☐	☐	☐
Eu sei facilmente como proceder com a ferramenta para fazer o que eu quero.	☐	☐	☐	☐	☐
A ferramenta é flexível para que eu possa usá-la da maneira que melhor me convier.	☐	☐	☐	☐	☐

78. Utilidade Percebida e Intenção de Uso

	1	2	3	4	5
Usando a ferramenta, é possível criar jogos educacionais mais rápidos.	☐	☐	☐	☐	☐
Usando a ferramenta, consigo expor meu conteúdo de ensino de forma lúdica e rápida.	☐	☐	☐	☐	☐
Eu usaria a ferramenta para criar jogos para as minhas aulas.	☐	☐	☐	☐	☐
A ferramenta é útil para que professores sem experiência consigam criar seus jogos.	☐	☐	☐	☐	☐
Minha tarefa de produzir conteúdo educacional de forma lúdica é mais efetiva com a ferramenta.	☐	☐	☐	☐	☐

79. De acordo com sua opinião, foi identificado algum aspecto positivo / negativo da utilização do ferramenta? Se sim, qual(ais)?

____

____

80. Você possui alguma sugestão para a melhoria do jogo? Em caso positivo, por favor, especifique-a.

____

____

H.4 Questionário de avaliação do jogo

81. Usabilidade

	1	2	3	4	5
O design do jogo é atraente (tabuleiro, cartas, interfaces, gráficos, etc.).	☐	☐	☐	☐	☐
Os textos, cores e fontes combinam e são consistentes.	☐	☐	☐	☐	☐
Eu precisei aprender poucas coisas para poder começar a entender o jogo.	☐	☐	☐	☐	☐
Aprender a jogar este jogo parece fácil para mim.	☐	☐	☐	☐	☐
Eu acho que a maioria das pessoas aprenderia a jogar este jogo rapidamente.	☐	☐	☐	☐	☐
As regras do jogo são claras e compreensíveis.	☐	☐	☐	☐	☐
As fontes (tamanho e estilo) utilizadas no jogo são legíveis.	☐	☐	☐	☐	☐
As cores utilizadas no jogo são compreensíveis.	☐	☐	☐	☐	☐

82. Experiência

	1	2	3	4	5
A organização do conteúdo me ajudou a estar confiante de que eu iria aprender com este jogo.	☐	☐	☐	☐	☐
Este jogo é adequadamente desafiador para mim.	☐	☐	☐	☐	☐
O jogo oferece novos desafios (oferece novos obstáculos, situações ou variações) com um ritmo adequado.	☐	☐	☐	☐	☐
O jogo não se torna monótono nas suas tarefas (repetitivo ou com tarefas chatas).	☐	☐	☐	☐	☐
Completar as tarefas do jogo provoca um sentimento de realização.	☐	☐	☐	☐	☐
Me sinto satisfeito com as coisas que posso aprender no jogo.	☐	☐	☐	☐	☐
Eu recomendaria este jogo para meus colegas.	☐	☐	☐	☐	☐
Eu me diverti com o jogo.	☐	☐	☐	☐	☐
Houve algo interessante no início do jogo que capturou minha atenção.	☐	☐	☐	☐	☐
Eu estava tão envolvido no jogo que eu perdi a noção do tempo.	☐	☐	☐	☐	☐
Eu esqueci sobre o ambiente ao meu redor enquanto jogava este jogo.	☐	☐	☐	☐	☐
O conteúdo do jogo é relevante para os meus interesses.	☐	☐	☐	☐	☐
É claro para mim como o conteúdo do jogo está relacionado com o ensino.	☐	☐	☐	☐	☐

Eu prefiro aprender com este jogo do que de outra forma (outro método de ensino).	☐	☐	☐	☐	☐

83. De acordo com sua opinião, foi identificado algum aspecto positivo / negativo da utilização e criação do jogo? Se sim, qual(ais)?

____

____

84. Você possui alguma sugestão para a melhoria do jogo ou no processo de criação do jogo? Em caso positivo, por favor, especifique-a.

____

____

85. Este espaço é reservado para quaisquer comentários adicionais (dificuldades, críticas e/ou sugestões) a respeito do estudo executado. Contamos com sua contribuição para que o trabalho seja aprimorado.

____

____

Obrigado pela sua colaboração!

Diego Cardoso Borda Castro
Cláudia Maria Lima Werner