UNIVERSITY OF TARTU
Institute of Computer Science
Computer Science Curriculum

Jan Markus Rokka

Perpetua - A Turn-Based Role-Playing Game With Runes

Bachelor's Thesis (9 ECTS)

Supervisor: Daniel Nael, MSc

Tartu 2025

Perpetua - A Turn-Based Role-Playing Game with Runes

Abstract:

The thesis describes the design and implementation of *Perpetua*, a turn-based role-playing game. The purpose of this game is to provide variety in its corresponding genre through the combination of equipment customization and survival strategy. The thesis gives an overview of the design and implementation of the game's most important systems. The usability of the game was also evaluated through playtesting on potential players. The thesis provides an analysis of the testing results. Future plans for development were made based on the given feedback.

Keywords:

Computer game, role-playing game, game design, game development, Unity, software development, playtesting

CERCS: P170 Computer science, numerical analysis, systems, control

Perpetua - Ruunidega käigupõhine rollimäng

Lühikokkuvõte:

Lõputöö kirjeldab käigupõhise rollimängu „Perpetua" disaini ja arendusprotsessi. Mängu eesmärk on pakkuda vaheldust vastavas žanris, kombineerides varustuse kohandamist ja ellujäämisstrateegiat. Lõputöö annab ülevaate mängu olulisemate süsteemide disainist ja teostusest. Viidi läbi kasutatavusele suunatud testimine potentsiaalsete mängijatega. Lõputöö sisaldab testimisest saadud tulemuste analüüsi. Saadud tagasiside põhjal on plaanis mängu edasi arendada.

Võtmesõnad:

Arvutimäng, rollimäng, mängudisain, mänguarendus, Unity, tarkvaraarendus, mängu testimine

CERCS: P170 Arvutiteadus, arvutusmeetodid, süsteemid, juhtimine (automaat-juhtimisteooria)

Table of Contents

1. Introduction

The video game industry has rapidly evolved over the past few decades, becoming a dominant form of entertainment that blends art, technology, and storytelling. This process has influenced many other fields and boosted Estonia's economy[1]. Role-playing games (RPG) stand out among the diverse genres that have emerged due to their deep narrative structures, complex character development, and immersive worlds[2].

Perpetua is a turn-based RPG. It seeks to introduce a dynamic rune-based equipment system that goes beyond traditional stat boosts. This entails offering players customization and strategic flexibility. Runes not only enhance abilities but also alter combat roles, enabling players to adapt their strategy to evolving challenges. Survival is central to the game, requiring careful resource management and strategic decision-making. The death of a party member intensifies the importance of runes, as players must reconfigure their party's roles to overcome setbacks, adding tension and depth to every battle.

The next chapter provides a basis for RPG game design concepts. Chapter 3 explores various role-playing games that inspired the creation of *Perpetua*. It also highlights similarities and differences in design between *Perpetua* and these referenced games. Chapter 4 focuses on the game design of *Perpetua*. Additionally, the intentions behind *Perpetua* and its reliance on fundamental game design principles are discussed. Chapter 5 explains the implementation of the game's mechanics. This includes mentions of the technologies used throughout the game's development. Chapter 6 provides an overview of the testing process along with an analysis of the testing results. Possible future improvements to the game are also briefly discussed.

Text generated by *ChatGPT 4*[3] was used as a starting point for various parts of this thesis. Before being added to the thesis, paragraphs produced by AI were inspected and modified. On some occasions, *ChatGPT 4* was also used to improve the phrasing of sentences.

Some specialized vocabulary is explained in the glossary (Appendix I). A short guide to playing the game is in Appendix II. The assets, source code and the game itself can be accessed in the GitHub repository (Appendix III). Links to videos of the testing sessions are available in Appendix IV.

---

[1] https://www.aripaev.ee/sisuturundus/2024/06/17/toostusharud-mida-mojutab-mangude-kasv-eestis

[2] https://www.kwalee.com/blog/gaming/the-evolution-of-rpg

[3] OpenAI ChatGPT 4 website https://openai.com/index/gpt-4/

2. Design of Role-Playing Video Games

To provide a basis for role-playing game design, this chapter contains descriptions of core game design concepts and the role-playing genre. Subchapter 2.1 explains core game design concepts, 2.2 aims to provide an overview of the role-playing genre and 2.3 describes the main characteristics of role-playing games.

2.1 Game Design Theory

The main elements that form a game are technology, story, aesthetics, and mechanics [2]. Sicart [3] defines game *mechanics* as rules that define how player actions affect the state of the game. In the same way, RPG games also contain mechanics that allow players to overcome challenges and objectives.

Adams [1] claims that in a user-centered approach to design, the content of a game is derived primarily from two components: the *core mechanics* and the user interface. Ernest [1] describes the core mechanics as the rules that define the game's challenges. The core mechanics also imply corresponding behaviors that a player can exhibit to overcome these challenges. These concepts were taken into consideration in the design of *Perpetua*.

The role of the user interface (UI) is to mediate the interaction between the user and the game's core mechanics. Unlike the UI of a conventional application, the main goal of a game UI is not efficiency but entertainment [1]. Since it plays a crucial in making a game accessible, *Perpetua*'s UI was also evaluated based on the testing results.

Games can take place in discrete or continuous time. Most action games and sports take place in continuous time. The players are able to take actions independently and in real time. Discrete differs from continuous in that time is divided into units called turns. In such a system, the time between turns does not affect the outcome of the game. Games that incorporate discrete time into their design are referred to as *turn-based* [2]. For example, Scrabble[4] is a turn-based game, during which moves/turns can be recorded without taking time into account. *Perpetua*'s combat system is also turn-based.

[4] Scrabble website https://shop.hasbro.com/en-us/scrabble-words

2.2 Role-Playing Games

Adams [1] claims that the RPG (*role-playing game*) genre allows the player to take on the role of a character and experience adventures in a fictional world. Throughout the game, the player's character becomes stronger and gains more skills. In tabletop games such as Dungeons & Dragons[5], a designated person, the *game master*, designs goals for players and controls the flow of the adventure (Figure 1. Dungeons & Dragons game in progress). In video games, however, this role is programmed into the game via the use of various automatic game

Figure 1. Dungeons & Dragons game in progress

components.

According to Wolf [4], player characters in role-playing games are represented by various statistics, possibly including a developed persona. Some of these statistics, such as strength or dexterity, are represented numerically. This genre is identifiable by its focus on identity and characters being represented statistically.

[5] Description of Dungeons & Dragons https://dnd.wizards.com/what-is-dnd

3. Similar Video Games

Various role-playing games and other works of fiction in the fantasy genre have inspired the design of *Perpetua*. The design process was mainly influenced by the games *Fear & Hunger*[6], *Fear & Hunger 2: Termina*[7], *Pillars of Eternity*[8], *Octopath Traveler II*[9] and *Path of Exile*[10].

Figure 2. Screenshot from Fear & Hunger

Fear & Hunger (Figure 2. Screenshot from Fear & Hunger) is a 2018 horror *dungeon crawler* game developed by Miro Haverinen[4]. It also has a sequel – *Fear & Hunger 2: Termina*[5]. The story of *Fear & Hunger* takes place in the Middle Ages, and contains fantastic elements and a gloomy yet richly described world. The game's mechanics are designed to punish the player for even the smallest mistakes. Thanks to this design decision, survival is a separate mechanic in and of itself: the player must constantly consider the well-being of their characters and carefully assess the possible consequences of decisions. In addition, new mechanics have been added to the usual turn-based combat system, making the system more complex. In designing

[6] *Fear & Hunger* wiki https://fearandhunger.wiki.gg/wiki/Fear_%26_Hunger

[7] *Fear & Hunger 2: Termina* wiki https://fearandhunger.wiki.gg/wiki/Fear_%26_Hunger_2:_Termina

[8] *Pillars of Eternity* home page https://www.paradoxinteractive.com/games/pillars-of-eternity/about

[9] *Octopath Traveler II* website https://octopathtraveler2.square-enix-games.com/en-us/about/

[10] *Path of Exile* website https://www.pathofexile.com/game

Perpetua, inspiration was taken from the fictional world-building of *Fear & Hunger* as well as its survival mechanics. *Perpetua* has similar survival elements to *Fear & Hunger* because both emphasize the survival of player-controlled characters. Both also feature a turn-based combat system.

Pillars of Eternity is a *Dungeons & Dragons*-inspired digital role-playing game developed by Obsidian Entertainment. Sawyer explains (2016) that the game was designed in a way to maintain a linear progression throughout the game – the player's character is expected to grow stronger at a static rate over the course of the game. This was done to ensure the viability of wildly different strategies and make the game easier to learn for new players. In designing Perpetua, achieving a similar progression for the game was intended.

Perpetua incorporates a separate type of item into its systems. This is the rune, which shares similar aspects with the gem in *Path of Exile*[11].

Figure 3. Screenshot from *Path of Exile*

Path of Exile (Figure 3) is an RPG released by Grinding Gear Games in 2013[9]. In addition to a main story campaign, the game features an in-game item trade market that allows players to customize characters and their abilities. *Path of Exile* also includes a gem system: by attaching different gems to a character's equipment, the character will gain various skills or attributes that are crucial in the game's real-time combat. This adds depth and customizability. This

[11] *Path of Exile* website https://www.pathofexile.com/game

system is somewhat similar to the rune system in *Perpetua* in that both have mechanics based on attaching items to the player character's equipment. However, unlike the gems in *Path of Exile*, the *Perpetua* runes are designed for a turn-based combat system. Additionally, the usage of gems in *Path of Exile* is limited by the player's equipment. This is not the case in *Perpetua*.

Figure 4. Screenshot from *Octopath Traveler II*

Octopath Traveler II[12] (Figure 4) is a 2023 RPG game developed by Square Enix and Acquire. Both the first *Octopath Traveler* and its sequel utilize a 2.5D graphical style, officially dubbed HD-2D. The *Octopath Traveler* franchise also features a turn-based combat system. In creating *Perpetua*, similar computer graphics techniques were used. These include placing 2-dimensional characters and objects in a 3-dimensional space and using pixelated textures on all in-game entities. Both games also feature a turn-based combat system.

[12] *Octopath Traveler II* website https://octopathtraveler2.square-enix-games.com/en-us/about/

4. The Design

The current version of *Perpetua* was designed to function as a prototype. It was designed through an iterative process. This is characterized by repeated playtesting, tuning, and modifications [1]. Several fundamental game design principles have been proposed by various experts in the field [1, 2]. Throughout the design and development, some game design principles were pursued. This was done to improve *Perpetua*'s gameplay. Subchapter 3.1 delves into the narrative foundation of the game, exploring the world, characters, and central conflicts that drive the player's experience. Subchapter 3.2 focuses on the core mechanics, detailing the interactive systems and player controls that shape gameplay.

4.1 The Story

Perpetua takes place in a fantasy world where a small minority known as the "perpetua" can only die of old age and mystical forces are tangible: gods with physical bodies and magical runes that influence the environment are integral aspects of the world. This has led to some wildlife adapting and evolving into mythical creatures that now roam the countryside. The powerful nation of Vegge'Don is ruled by the perpetua. It is currently at war with Tüüga, a purely capitalist nation with a strong economy, which shares a long border with Vegge'don along its eastern side. Vegge'don is marked by discrimination against regular humans. This prompted the rise of a secret organization determined to overthrow the government. In the demo, the player assumes the role of Märta, a revolutionary tasked with recovering an ancient artifact rumored to have been found in the small village of Fleken. This is located in the far-northern regions of Vegge'Don. Throughout the journey, Märta and her companions encounter and fight wildlife and bandits, who are after the same goal. Upon reaching the village, Märta finds it ruined along with a suspicious-looking ancient temple. After approaching the temple, she is greeted by a mysterious voice. It tells her to find and defeat 3 sinners, alleged to have committed unforgivable acts toward the gods. Märta and her comrades carry out the given objective, with each fight seeming more grueling and puzzling than the last. They get through it and obtain valuable runes from their targets along the way. After returning to the temple's entrance, they are ambushed by a monstrous amalgamation of animals with a cold rune embedded within it. After somehow defeating it, the party advances into the temple, where they become acquainted with the rumored artifact. The Cage of Knowledge is a trickster god of knowledge capable of accessing the collective human consciousness, now confined to an object. While conversing with it, Märta discovers that her organization is backed by Tüügan officials. This marks the end of the game.

The challenges and objectives described put the player in strategic combat and survival situations, both of which are core gameplay elements of *Perpetua*.

4.2 The Mechanics

To overcome challenges and reach the game's goal, a player can learn *Perpetua*'s game mechanics. *Perpetua*'s core mechanics revolve around resource management, turn-based combat, and a bit of exploration.

4.2.2 Resource Management

The player controls a party of 4 characters (Figure 5. Party members), all with numbers representing their physical well-being and combat capabilities. These will be referred to as a character's stats. The player also has access to an inventory, which contains items.

Figure 5. Party members

Items gained from lootable objects or combat encounters are divided into consumables and equipment. Consumables can only be used once, after which they improve a character's stats. Generally, consumables raise a character's health points or willpower. Examples are bread, which mainly raises a character's health, and meat (Figure 6), which mainly raises a character's willpower.

Figure 6. Consumable - meat

Equipment improves a character's stats only while it is equipped. Equipment is further divided into 4 categories: armor, weapons, accessories, and runes. Each party member can only have 1 item of each type equipped, except for runes, which each party member can equip 2 of.

Equipping a set of armor (Figure 7. The equipping of an armor) improves the wearer's defensive capabilities. Equipping a weapon enhances the character's offensive capabilities and modifies their accuracy. Accessories provide slight general improvements to various stats. Equipping runes modifies a character's role in combat more significantly. For example, the stun rune makes a character's attacks stun the enemy, and the extra turn rune gives a character an extra turn in combat. It is up to the player to decide when and how to use these items. Because of this, the usage of equipment provides the player with the opportunity to customize their approach to combat.

Figure 7. The equipping of an armor set

4.2.3 Combat

While wandering around the player can be chased by enemies. When the player is sufficiently close to an enemy, they will enter combat. It is possible to avoid some encounters by outrunning the enemies or fleeing from combat (Figure 8. The player being chased by an enemy).

Figure 8. The player being chased by an enemy

Perpetua's combat system is turn-based, meaning that each combat participant gets a turn to commit an action. The turn order is decided by sorting each participant's attack speed value in descending order.

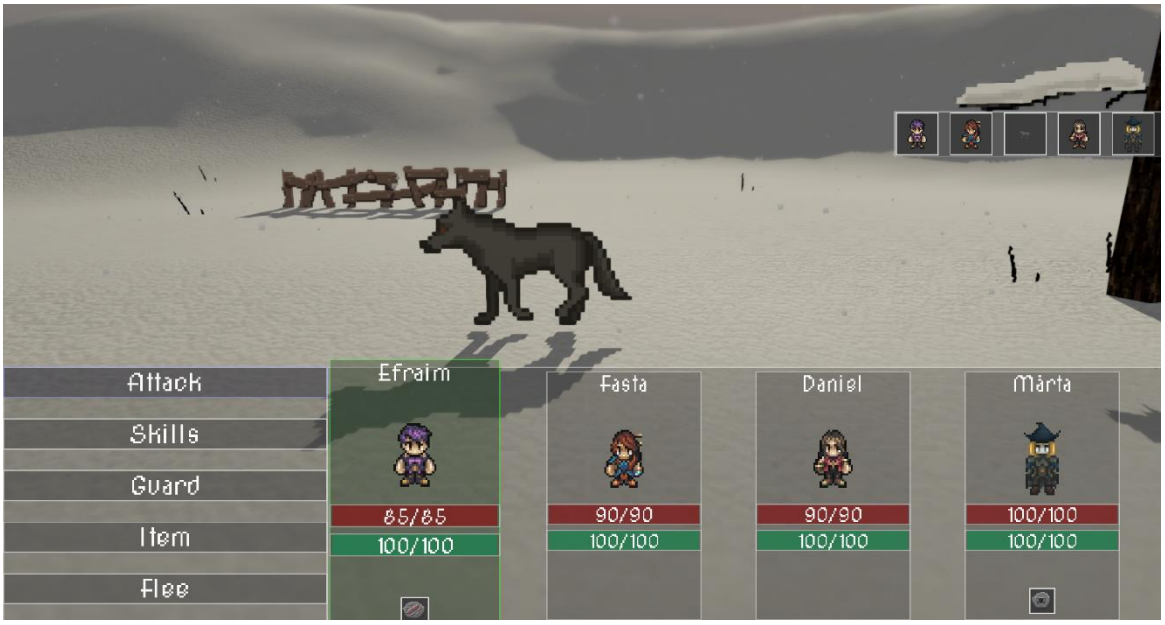

Figure 9. The battle menu

In combat, the player selects actions for each of their party members (Figure 9). During a turn, a party member can either attack, defend, use an item or perform a special skill. Special skills include offensive and weakening spells as well as damage- or defense-boosting actions. Skills also drain a character's willpower and are only usable while the character has enough willpower. After the player has selected actions for all party members, the turn is played out and every action's consequences are shown. This constitutes the turn-based battle gameplay loop, which is quite similarly executed in *Fear & Hunger* and *Octopath Traveler II*.

The combat encounter ends when either the player's party or their enemies are no longer able to fight. Additionally, the player's party will also lose if the main character is out of combat. This was decided because the game's narrative is scripted and revolves around the main character.

After a combat encounter, the player is back in the scene in which the battle started. They can also gain items from enemies after defeating them, as was previously stated in the resource management subchapter.

After a battle, the player's party will likely have lost some health and willpower. The player can restore their entire party's willpower at a save point on the map. Health points are only

replenishable by using consumables, rather than by using a save point. This makes the game more punishing and forces the player to think critically before losing a teammate, making gameplay more engaging. The same design decision is also likely to encourage the player to reconsider the management of their equipment after losing a teammate. This could similarly make the game more engaging.

5. The Implementation

This chapter outlines the process of realizing the previously described mechanics and ideas. Subchapter 4.1 introduces the technologies employed. Subchapter 4.2 details the implementation of the game's core systems.

5.1 Technologies

Unity[13] is a popular game engine. Unity was selected because of previous experience with it from University of Tartu courses. It also has many community support forum threads and good inbuilt packages.

The Universal Render Pipeline[14] was used for more customization and advanced features relative to the In-Built Render Pipeline within Unity. Other packages used include Odin, an interface expansion package, ProGrids and ProBuilder for 3D modeling inside of Unity, Unity User Interface for easier development of user interfaces, AI Navigation for navigation of various characters in the overworld, and Scriptable Objects for data structures.

C# was the chosen programming language. It is the main language used within Unity and provides object-oriented features that fit game development.

Blender[15] is a free and open-source 3D modeling software. It was used to make some simple 3D models including trees.

Paint.net[16], Aseprite[17], and Clip Studio Paint[18] are raster graphics software, which were used to draw sprites and textures.

The game includes music composed in Fruity Loops[19] Studio 2023. Some sound design was also done with Audacity[20], a free open-source digital audio editor.

[13] https://unity.com/

[14] https://docs.unity3d.com/Manual/universal-render-pipeline.html

[15] https://www.blender.org/

[16] https://www.getpaint.net/

[17] https://www.aseprite.org/

[18] https://www.clipstudio.net/en/

[19] https://www.image-line.com/

[20] https://www.audacityteam.org/

5.2 Implementation of Systems

One of the project's challenges was the implementation of a character data class system. The system needed to include both the data of party members and enemies, as they both needed to partake in battle. `CharacterData` (Figure 10) serves as the foundational class housing shared properties and methods among both enemies and party members. These include basic information alongside skills, status effects, and stats. Stats, skills, and status effects are stored using separate classes, utilizing the component pattern[21]. The classes `EnemyData` and `PartyCharacterData` derive from CharacterData and include methods and variables specific to each character type. Enemy types with specific scripted attack patterns derive from `EnemyData`. These classes override the `SelectTurn` method within `EnemyData` and sometimes include extra variables.

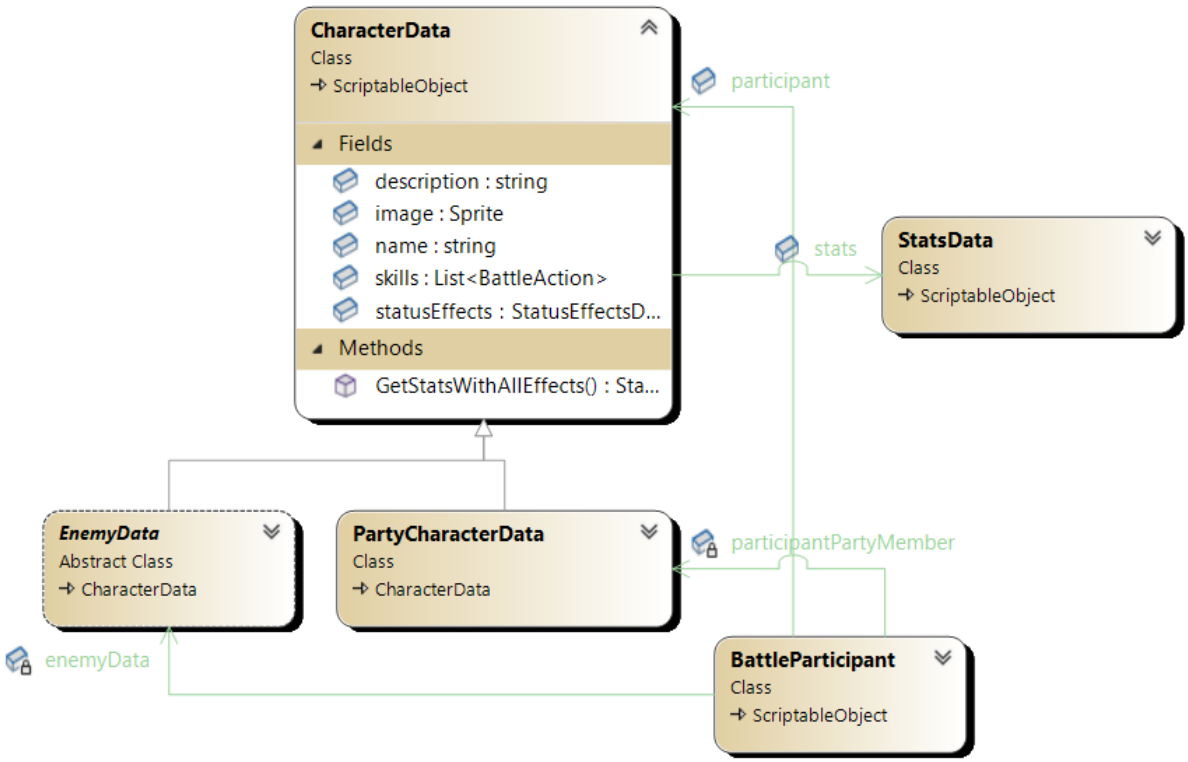

Figure 10. Class diagram of characters

[21] https://gameprogrammingpatterns.com/component.html

`ItemData` (Figure 11) defines all items in the game. It references class instances of `ArmorVariables`, `WeaponVariables`, `RuneVariables`, and `ConsumableVariables`. These contain values that affect character stats such as health, defense, damage, accuracy, and dodging ability. `RuneVariables` also includes fields for an accuracy multiplier, extra turn count, and a list of status effects to give to the enemy after a successful attack.

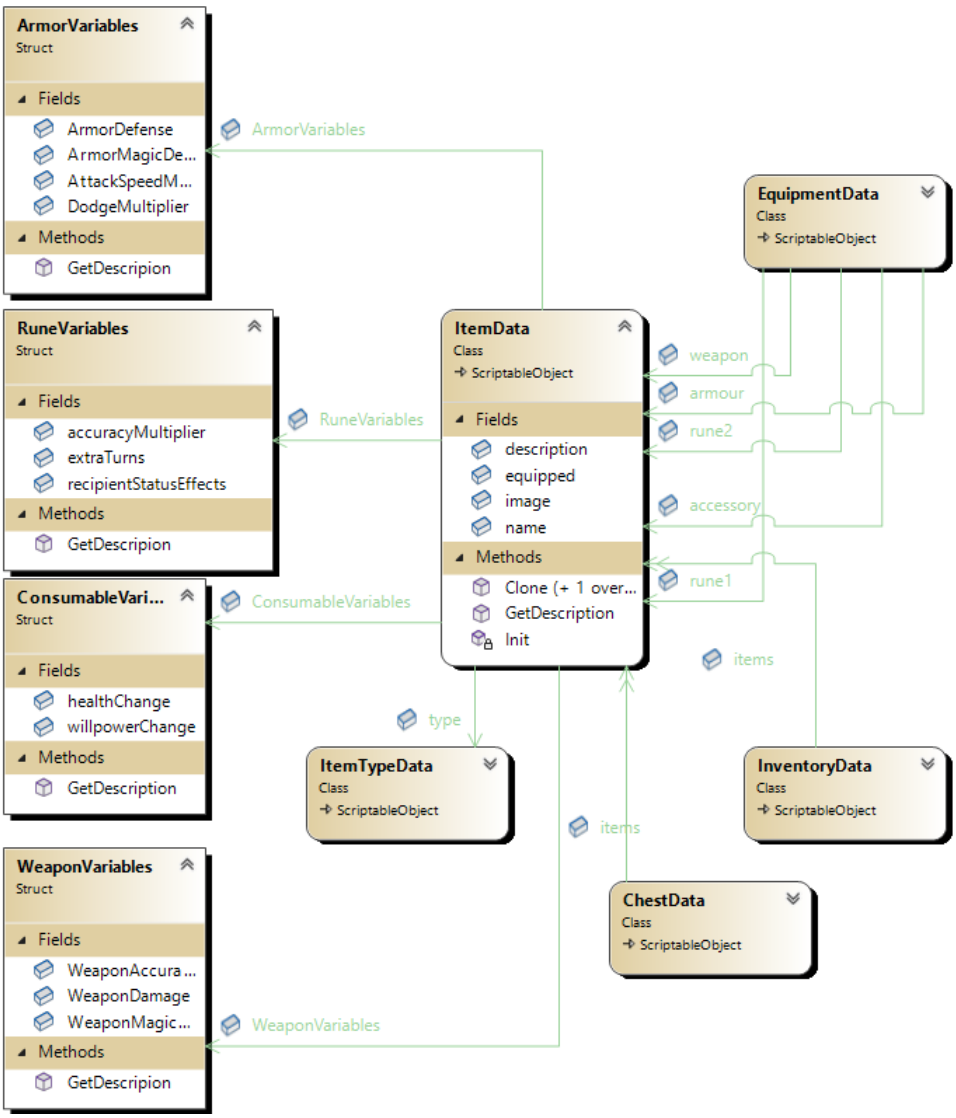

Figure 11. Class diagram of items

`EquipmentData` represents a character's equipped items. It is referenced within `PartyCharacterData` and thus follows the component pattern. The effects of worn equipment are applied within the overridden method `GetStatsWithAllEffects` in `PartyCharacterData`. This method returns a new Stats instance with the given equipment and status effects applied.

`BattleParticipant` is a wrapper class that mediates the usage of different character data types in battle. It includes a reference to an enemy or party member and various methods that behave differently depending on the type of character it references. At the start of a battle, the participating `EnemyData` and `PartyCharacterData` instances are referenced in new BattleParticipant instances.

The battle loop is taken care of in the class `BattleManager`. First, it sorts all the battle participants in a descending order based on their attack speed. This determines the turn order. After that, actions are gathered from the player and enemies in the turn order. Then the actions are carried out in the same order.

Most other features and smaller systems were relatively straightforward to implement, thus they were not described in this thesis. Most of these were implented using game programming patterns such as singleton, state, and component. The singleton pattern ensures that a class has only a single instance and provides global access to it [8]. This was used for classes that manage scene setup, menus, and the player's inventory, party members. The state pattern allows objects to change their behavior when its state changes [8]. The selected game engine's animation system already works according to this logic[22]. Most classes that deal with animations, take advantage of this system. The component pattern divides the different domains a class spans into individual decoupled components. These components can exchange information without being connected [8]. Objects within Unity already work using a modular component system[23]. Using this, classes were divided into more specialized components for better versality and reused in various places. To evaluate how well these implemented systems and features are received, playtesting is required. Thus *Perpetua* was tested on potential players.

[22] https://docs.unity3d.com/Manual/AnimationStateMachines.html
[23] https://docs.unity3d.com/Manual/Components.html

6. Testing

Playtesting was conducted on potential users to evaluate *Perpetua*'s game design and implementation. Subchapter 5.1 describes how the testing was conducted and why. Next, subchapter 5.2 features an analysis of the results. Possible future improvements based on the results are discussed in subchapter 5.3.

6.1 Methodology

Playtesting is an important aspect of game development. It helps the developer identify potential problems and strong points within their game [2]. Perpetua was tested on 5 people since this is enough to find most usability issues [7].

Playtesters completed *Perpetua* in a single session without significant prior knowledge of the game's systems. Throughout the session, they received no external assistance to guide them through the gameplay. The sessions were observed and recorded, capturing both gameplay and the testers' vocal commentary. The footage was later analyzed. After completing the game, the testers filled out a questionnaire about their experience with the game. The questionnaire covered their overall impression of the game, game mechanics, difficulty, user interface, and visuals. This feedback offered valuable insights into the strengths and weaknesses of *Perpetua*'s design and execution.

6.2 Results

This chapter provides an analysis of the testing results. Subchapter 5.2.1 examines the players' experience with video games and their overall impressions of the game. In subchapter 5.2.2, the focus shifts to the testers' opinions on the implementation and design of *Perpetua*'s mechanics, exploring how well the gameplay elements were executed. Subchapter 5.2.3 discusses how balanced the game is, addressing the testers' experiences with difficulty and fairness. Subchapter 5.2.4 reviews the user interface, analyzing its intuitiveness and usability. Next, subchapter 5.2.5 describes evaluating the visuals, considering the aesthetic appeal and graphical quality of the game. Finally, subchapter 5.2.6 concludes the testing results with an overview of the game's potential for an audience.

6.2.1 Overall Impressions

In the first section, the survey collected data on the testers' previous experience with video games, including their frequency of play and the types of games they typically engage with. This information is important in contextualizing their feedback on the game and understanding how their background may influence their perceptions of the game's mechanics and overall experience.

Firstly, the respondents were asked how often they play video games. The majority of participants reported playing video games "Almost every day", indicating a high level of engagement with gaming. Smaller groups reported playing "One or more times per week" and "Maybe a few times per month". This distribution suggests that the survey sample predominantly consists of frequent gamers, potentially leading to a more critical and informed evaluation of *Perpetua*.

Next, the participants were asked to identify the types of games they commonly play. The results revealed a diverse range of gaming preferences among the respondents. The most commonly mentioned genres were shooter, role-playing, adventure, strategy and racing. A single respondent also indicated a preference for survival games. The prevalence of RPG and adventure games among the participants is particularly noteworthy, as elements of these genres are a large part of *Perpetua*.

For respondents who indicated that they play RPGs, additional data was collected on the specific titles they play. Frequently mentioned games include Baldur's Gate, Dragon's Dogma, Stardew Valley, Cyberpunk 2077, and Horizon Zero Dawn. The inclusion of these titles suggests that the respondents are familiar with both classic and contemporary RPGs, which often feature intricate gameplay systems, expansive worlds, and detailed storytelling.

This background likely equips these players with a set of expectations regarding game mechanics, narrative structure, and overall experience. This could influence their evaluation of *Perpetua*.

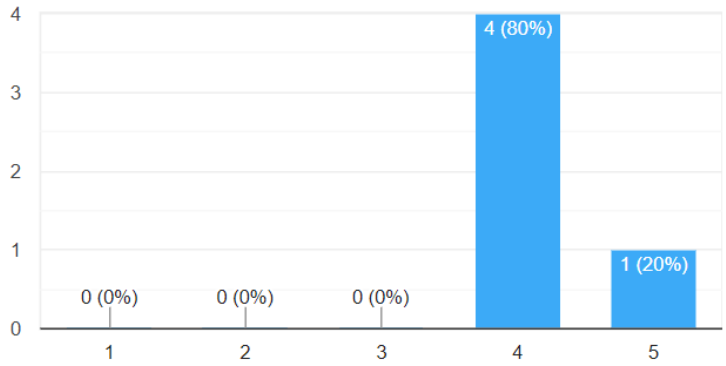

Figure 12. The overall impressions of *Perpetua*

The next part of the survey focused on capturing the respondents' overall impressions of playing *Perpetua*. Participants were asked to rate how much they enjoyed playing Perpetua on a scale from 1 to 5 (Figure 12),. An answer of 1 indicates the player enjoying the game "Not at all" and 5 - "A lot". The average score was 4.2, indicating a generally high level of enjoyment among the players. This positive reception suggests that *Perpetua* succeeded in delivering an enjoyable experience that resonated with most of its audience.

Testers were also asked to describe their favorite and least favorite moments in the game. Commonly cited favorite moments included boss fights, equipment management, and fighting strategy mechanics. Conversely, the least favorite moments centered around frustration with navigation within the game. However, these negative experiences did not appear to impact the overall positive perception of the game significantly. The aforementioned positive opinions seemed heavily influenced by the testers' interactions with *Perpetua*'s mechanics.

6.2.2 The Mechanics

The questionnaire included ratings for how several key mechanics affected the players' gameplay. The ratings were on a scale from 1 to 5, where 1 represents a poor experience, and 5 represents an excellent experience. The mechanics evaluated include moving around, equipping weapons and armor, equipping runes, attacking and guarding, performing skills, using consumables, fleeing, looting chests, and completing objectives.

Movement rating averaged at 4 (Figure 13) indicating that players generally found movement to be smooth and functional, though not without minor issues.

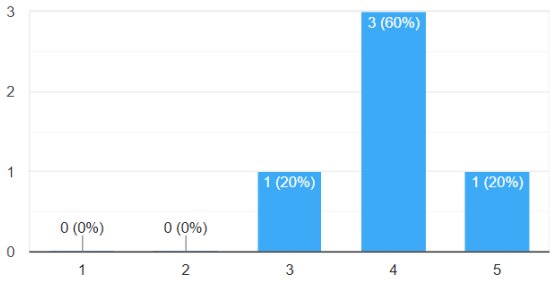

Figure 13. Ratings of movement mechanics

The mechanics around equipping weapons and armor received mixed ratings (Figure 14), with an average score of 3.4. This suggests that while it was functional, it might have lacked some intuitiveness or depth.

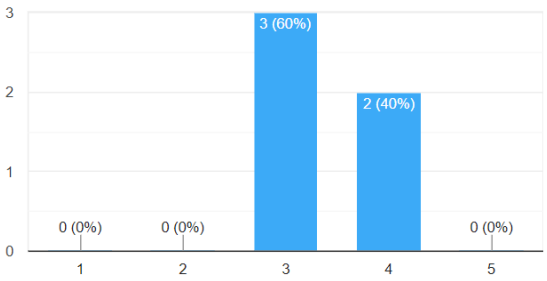

Figure 14. Ratings of equipping weapons and armor mechanics

The mechanics around equipping runes got an average rating of 4 (Figure 15), indicating a generally positive experience, with some players finding it particularly satisfying. Since runes are a major part of this thesis, these results signal that the mechanics around them have been successful.

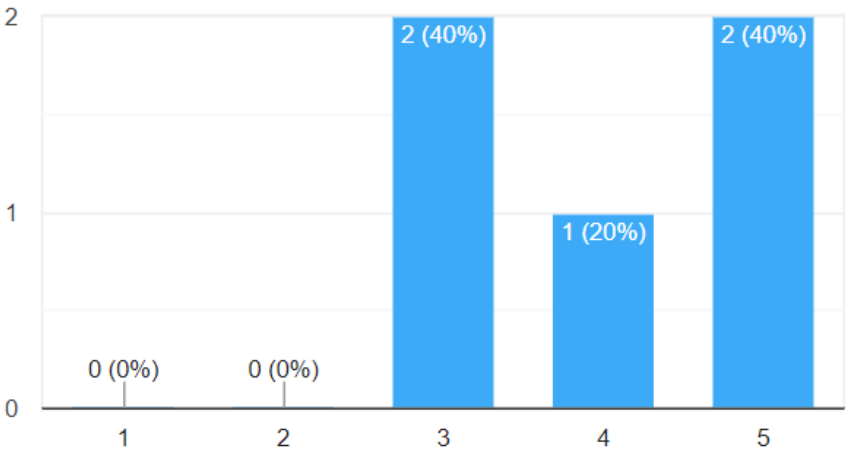

Figure 15. Ratings of rune equipping mechanics

Moving onto more direct combat mechanics: attacking and guarding received an average rating of 4.2 (Figure 16). These mechanics being well received may indicate that the combat felt engaging and responsive.

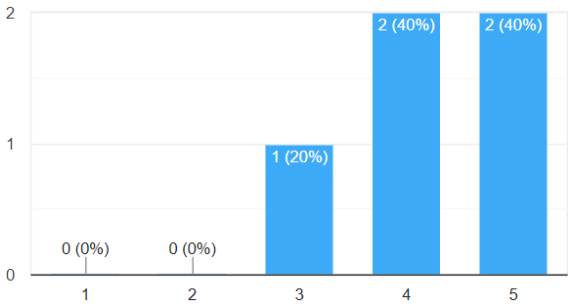

Figure 16. Ratings of attacking and guarding mechanics

Similarly, performing skills in combat also received consistently high ratings with an average of 3.8 (Figure 17). This could indicate satisfaction with the variety and execution of skills in the game.

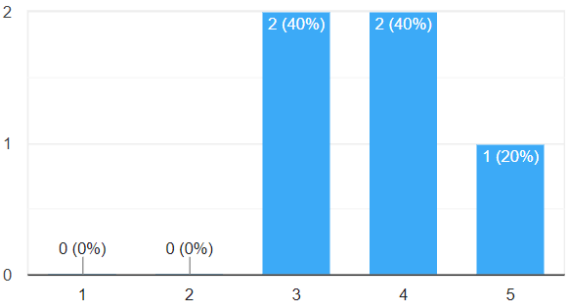

Figure 17. Ratings of skill performing mechanics

Using consumables (Figure 19) received more varied ratings, with some lower scores suggesting that some players may have found the system limiting or inconvenient. This could be because consumables can only be used while in combat. During testing, most players were observed trying to use consumables outside of battle, which may have contributed to the rating of this mechanic.

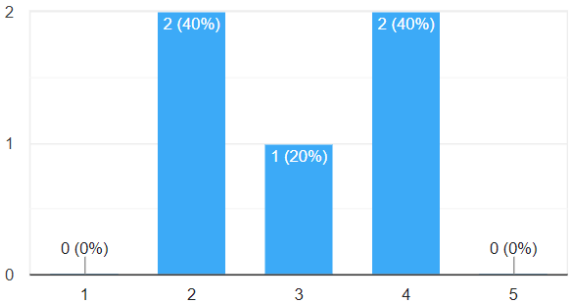

Figure 19. Ratings of consumable mechanics

Fleeing from combat encounters received an average score of 3 (Figure 18). This came along with mixed feedback possibly indicating that while the fleeing mechanic was functional, it may not have been fully integrated into the gameplay experience.

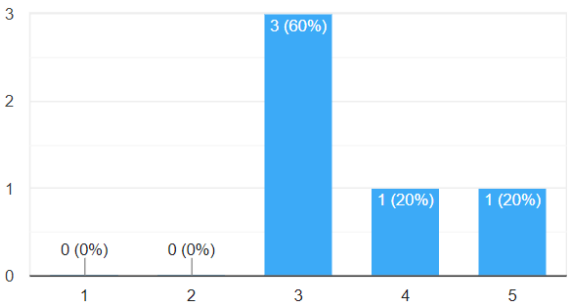

Figure 18. Ratings of the fleeing mechanic

Looting chests was rated highly, with an average rating of 4.4 (Figure 20). This shows, that players enjoyed the looting mechanic.

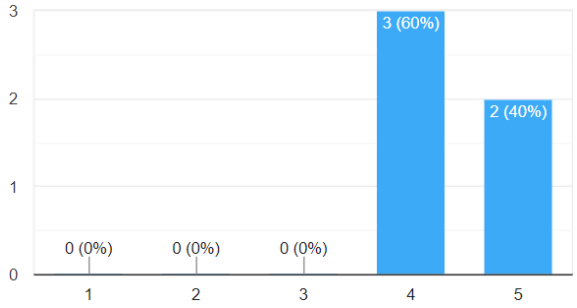

Figure 20. Ratings of looting mechanic

When it comes to completing objectives, the ratings were generally positive (Figure 21). However, there was an outlier, suggesting that while objectives were clear, someone felt they could have been more engaging or challenging.

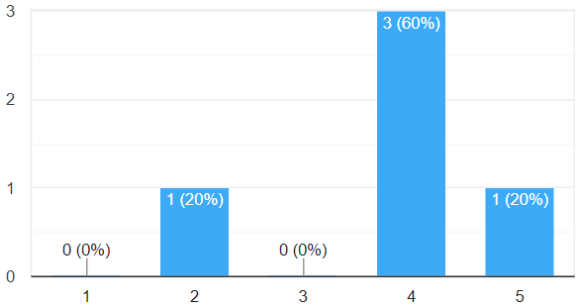

Figure 21. Ratings of objectives and their completion

When it came to the testers' favorite mechanics, several players highlighted attacking, guarding, performing skills, exploring, and the overall turn-based combat experience. On the other hand, the consumable system and movement were commonly cited as areas needing improvement. Players found the inability to use consumables out of combat and certain movement restrictions to be limiting.

The testers were also asked for improvement suggestions. Several areas were identified based on respondent feedback. A recurring suggestion was the enhancement of consumable usage, with many participants advocating for the ability to use consumables outside of combat. Adding more interactability to explorable areas was also suggested multiple times. Some improvements mentioned revolved around having more visual indicators for equipped items and skills. Adding more unique enemy mechanics along with mechanics that affect the turn order was also suggested. These changes would likely change the game's difficulty dramatically and thus require balancing of various enemy data.

6.2.3 The Balance

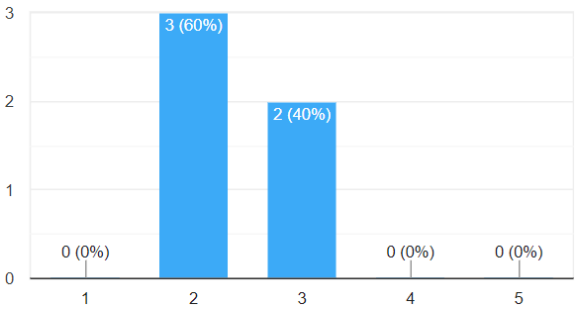

Figure 22. Ratings of gameplay difficulty

Achieving the right level of difficulty is important in making a game fun. To assess potential issues, the testers were asked to rate the perceived difficulty of *Perpetua* (Figure 22). This was rated on a scale from 1 to 5, ranging from "very easy" to "extremely difficult". The average rating was 2.4, indicating moderate difficulty. Explanations for these ratings highlighted the game mechanics being relatively simple and the game becoming easier after learning its mechanics.

Afterward, the testers were asked whether the game felt too easy at any point. Several playtesters noted instances where the game felt too easy, primarily when they had equipped themselves with proper gear. For example, one participant mentioned that "with proper gear the game was too easy," indicating that the difficulty scaling did not match the player's progression in terms of equipment. Another tester pointed out that „all the non-boss enemies felt like they could spam basic attacks and win", suggesting that regular encounters lacked challenge, which could diminish engagement over time. This feedback indicates that while the game's overall difficulty was balanced for some, there may be a need for more adaptive difficulty mechanisms, particularly for players who quickly optimize their gear and strategies.

In contrast, the testers were also asked whether any moments of the game felt too unforgiving. Some playtesters did. One participant specifically mentioned the inability to revive party members, which made the game feel too punishing. Another player suggested that health regeneration outside of combat should be included with saving at a save point. These comments point to a potential need for adjustments that provide more opportunities for recovery.

When asked whether any runes seemed too powerful, a playtester identified that using an attack buff rune with an extra turn rune was too powerful. Other participants did not find any runes to be too powerful.

The participants were also asked, whether any enemies seemed too strong. All playtesters did not find any enemies to be overwhelmingly powerful, indicating that the difficulty curve related to enemy strength was generally well-received. However, the absence of complaints could also suggest that some players found the enemies to be too easily manageable.

When asked which battles were the most enjoyable, the necromancer and the final boss stood out among the rest (Figure 23). These enemies likely provided a satisfying level of challenge and engagement, with one player explicitly mentioning the final boss as a highlight of the combat experience.

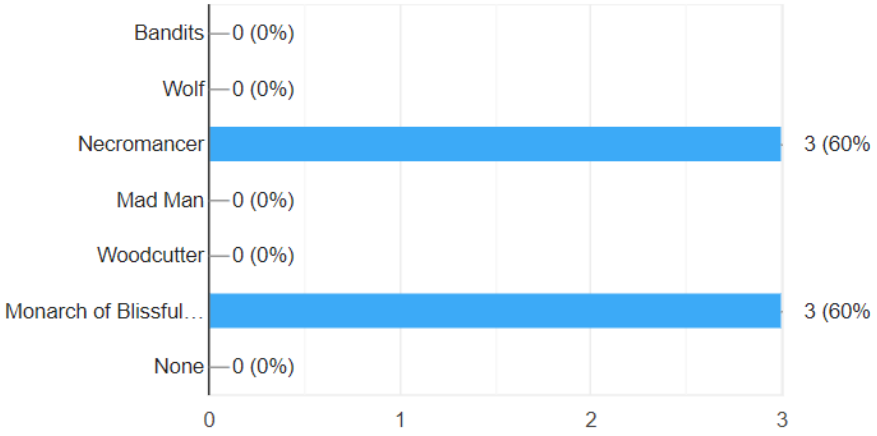

Figure 23. Enemies who were the most enjoyable to fight

When discussing what they liked or disliked about the battles, playtesters provided a mix of feedback. On the positive side, some players appreciated the unique character skills, which likely added variety and strategic depth to the combat. However, there were also critiques, such as one player noting that there were “lots of missing the attacks”. This contributed to the outcome of the battle feeling a bit randomized. Another player mentioned that the enemy sprites did not change after being defeated, which may have detracted from the immersion and visual feedback during battles.

Playtesters also provided constructive suggestions for improving the overall balance of the game. One player suggested enhancing accuracy for both enemies and party members. This would reduce the amount of turns that end up not changing the outcome of the battle. Another recommendation involved introducing a place where players can permanently heal, thus alleviating the harshness of ongoing survival. A general suggestion to make fights more difficult was also mentioned, indicating that some players desired a higher level of challenge. These suggestions highlight areas where incremental adjustments could lead to a more balanced and engaging gameplay experience, catering to a range of player skill levels. For a game to provide a challenge, the player needs to have some level of control.

6.2.4 The User Interface

The user interface of a game should provide the player with a sense of power and control in the game world [2]. To achieve this, a game's controls need to feel intuitive [2]. To assess these factors, the testers were asked to judge the intuitiveness and communication of *Perpetua*'s controls.

Playtesters generally found the controls of the game to be intuitive (Figure 24), with an average rating of 4.2. The controls were described as "standard and easy to pick up". This suggests that players did not struggle to understand or use them. However, there was some variation in feedback, with one player rating the controls as a 3, indicating room for improvement in making them more intuitive or user-friendly.

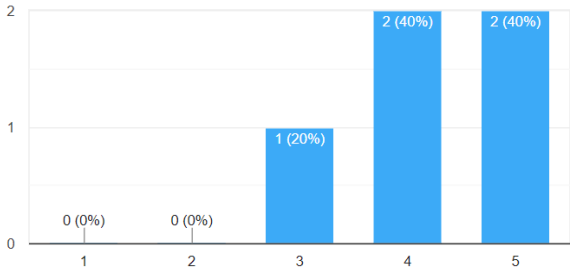

Figure 24. Ratings on the intuitiveness of controls

The written feedback on the controls was mixed. Positive comments highlighted that the controls were mapped in an expected way, which aligns with common gaming conventions. On the other hand, some players expressed dissatisfaction with the layout, indicating that there may have been elements that felt cumbersome or less streamlined.

The ratings for how well the game communicated its controls (Figure 25) averaged at 3.6, suggesting variability in player experiences. While some players felt the communication was effective, others thought it could be improved. For example, a rating rating of 2 indicates that at least one player struggled to understand the controls, which could be due to a lack of clear guidance.

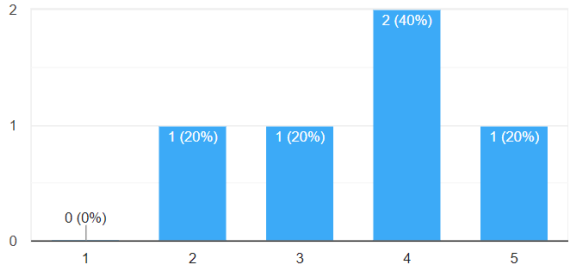

Figure 25. Ratings on the communication of controls

Playtesters provided several suggestions for how the game could better communicate its controls. One player suggested introducing game mechanics more gradually through in-game narratives or characters rather than relying solely on upfront tutorials. Another suggestion was to include a dedicated "Controls" page in the settings menu, which could be useful for players

who need a quick reference. Adding more button prompts to interactables was also suggested, to make the controls more obvious for new players.

The graphical user interface (GUI) also plays a crucial role in giving the player control. Usability and comprehension are important aspects of good GUI design [6]. Because of this, the playtesters were asked to rate *Perpetua*'s GUI based on these factors. When it comes to usability, the ratings varied (Figure 26). The average rating was 3.8. This suggests that while the interface was generally functional, some aspects may have been less user-friendly or intuitive for certain players.

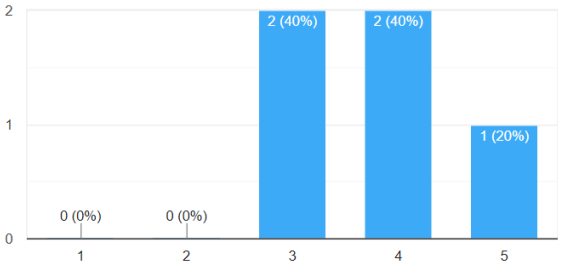

Figure 26. Ratings on the usability of the GUI

Similarly to the usability ratings, the readability and comprehension of the GUI received mixed feedback (Figure 27). The average score was 3.6. These ratings imply that while the majority of players found the GUI readable, there were instances where comprehension may have been hindered by design choices such as font type.

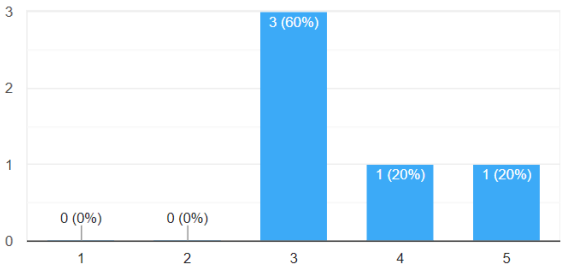

Figure 27. Ratings on the readability and comprehension of the GUI

When asked about ways to improve the GUI, one player noted that there were issues with reading the font, especially when text overlapped. Another participant mentioned that in some areas characters could not be seen behind buildings. This made controlling the characters more difficult. It was also suggested that hovering over buffs and debuffs should display a more detailed description of the given status effect.

The feedback on both the controls and graphical user interface indicates that while they are both generally functional, there are several areas where improvements could enhance usability. This also applies to readability and comprehension in terms of the GUI. Video games generally rely a lot on visuals, thus it is also important to judge *Perpetua*'s graphics and aesthetics.

6.2.5 The Visuals

The visuals of *Perpetua* consist mainly of 2D sprites set within a 3D environment, showcasing a dark fantasy aesthetic. To judge how beneficial these visuals are to the game, playtesters were asked to rate *Perpetua*'s overall appearance on a scale from 1 to 5 with 1 being „Terrible“ and 5 being „Amazing“ (Figure 28). The average score was 4.4, indicating a generally positive reception.

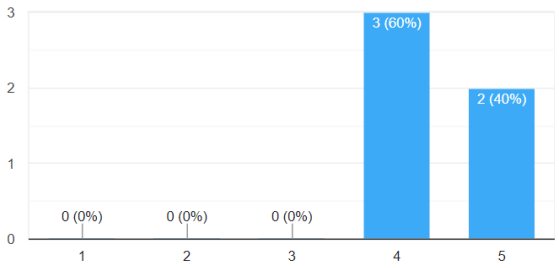

Figure 28. Ratings on the game's overall appearance

The visual design was appreciated by most players, with comments highlighting the unique aesthetic. One player specifically noted that they “loved the shadows and the 2D in a 3D world feel,” suggesting that the blend of visual styles was effective and visually appealing. However, the feedback also featured critique. One playtester mentioned that the environment felt “a bit too barren,” indicating that while the core visual design was appreciated, there might be room for adding more detail or variety to the environment.

Participants also rated how the game’s graphics affected its functionality, with 1 standing for „Worsened“ and 5 for „Improved“. The average rating was 4.2 (Figure 29). This indicates that, for the most part, the graphics supported the gameplay effectively. When explaining their ratings, some players emphasized the readability and simplicity of the graphics, which contributed to the overall ease of gameplay. However, there were some issues that could be addressed. For instance, one player mentioned that “sometimes the 2D interactable notification above a 3D world seemed jarring”.

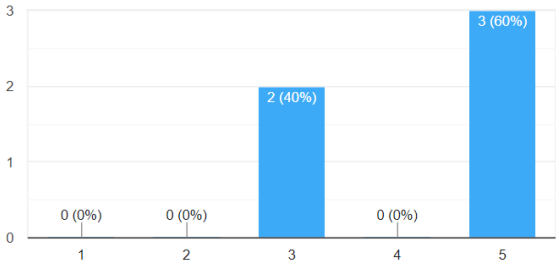

Figure 29. Ratings on how the game's graphics affected its functionality

Overall, the visuals of *Perpetua* were received positively. Still, some areas of improvement were discovered. This wraps up the testing of *Perpetua* from various perspectives. The testing results are concluded in the next subchapter.

6.2.6 Conclusion

Games are made to be played. For this an audience is necessary. To assess the existence of a potential audience for Perpetua, the testers were asked 2 questions. The first asked how likely they were to play a longer version of Perpetua (Figure 30). The answers were on a scale of 1 ("Not at all") to 5 ("Very"). Responses were largely positive (Figure 30), with a significant majority indicating a strong likelihood of continuing to engage with the game.

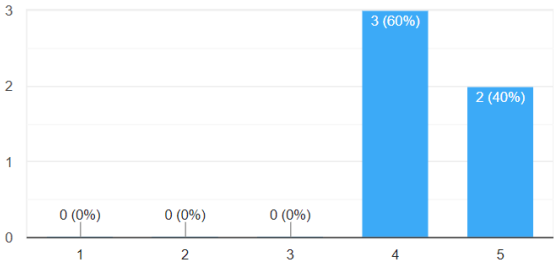

Figure 30. The likelihood of playing a longer version of *Perpetua*

The second question focused on the respondent's willingness to recommend *Perpetua* to friends (Figure 31). This question is crucial as it gauges the game's overall appeal and likelihood of organic growth through word-of-mouth. An average score of 3.8 out of 5 indicates that testers might recommend *Perpetua* to their friends. This could imply the existence of a potential audience for the game. However, answers to both of these questions were likely made on the presumption that most of Perpetua's current problems would be fixed.

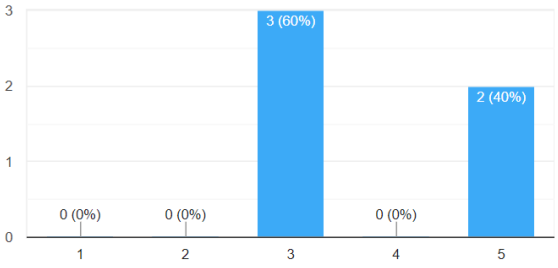

Figure 31. The likelihood of recommending *Perpetua* to a friend

6.3 Future Improvements

In the previous analysis chapters, many issues with the game were identified. One tester encountered a game-breaking bug, when they tried to flee from the final boss. This has since been addressed. Addtionally, the ability to use consumables outside of combat was also implemented. Most fixes and improvements however are scheduled for future implementation. These include:

- creating more objectives and different ways to complete them;
- adding more interactable characters and objects throughout the game's scenery;
- better balancing focusing on increasing accuracy for all battle participants;
- fixing issues with text clipping and the game's chosen font;
- adding more unique enemy mechanics;
- replacing royalty-free artwork with self-produced materials;
- expanding the rune mechanics further, thus adding more depth;
- adding more visual indicators to enemies and making tooltips more descriptive;
- adding more ways for the player to recover health and party members;
- making the controls mappable within a separate tab;
- adding graphics settings.

7. Conclusion

As a result of the work conducted in this bachelor's thesis, *Perpetua* was developed as a turn-based role-playing game. The game was designed to integrate the core principles of RPG design while offering a fresh take on equipment customization and survival. The most important elements designed and implemented were party members, enemies, equipment and runes, and a turn-based battle system. The mechanics around survival and runes were implemented to create critical moments during which players are forced to reassess their strategy. Inspiration was taken from similar turn-based RPGs that utilize survival mechanics or deep equipment customization. Although similar games were identified, an example that fused these elements through a specific item type was not found.

The game was realized using Unity and C#. In addition to the previously mentioned systems, more features tied to the game's functionality, audio, and visuals were realized. As these mostly followed standard patterns and were rather straightforward to implement, they were not described in detail within the thesis.

A few scenes containing enemies, loot, and cutscenes were created along with objectives for the playtesters to follow. Individual testing sessions were recorded and a questionnaire was administered to participants to assess usability, enjoyment, and the presence of a potential audience for the game. The topics discussed included the playtesters' experience with video games, overall impressions, game mechanics, balancing, user interface, and visuals. All testers enjoyed the game but still found numerous flaws with *Perpetua*'s game design and implementation. The future plan is to resolve the issues identified during testing, significantly expand the content, increase the depth, enhance the polish, and ultimately transform *Perpetua* into a role-playing experience spanning several hours of gameplay.

I express my gratitude to my supervisor Daniel Nael, whose guidance and course Programming Patterns in Video Games helped improve the quality of this thesis and game immensely. Many thanks also go to the playtesters of *Perpetua* for providing feedback, and everyone from The Computer Graphics and Virtual Reality Study Lab, whose courses I studied, for providing me with the basic knowledge and skills necessary to create video games.

References

[1] Adams, E. (2013). *Fundamentals of Game Design.* USA: New Riders.

[2] Schell, J. (2019). *The Art of Game Design: A Book of Lenses*. Massachusetts: CRC Press/A K Peters.

[3] Sicart, M. (2008, 12). Defining Game Mechanics. *Game Studies, 8.* (08.01.2024).

URL https://gamestudies.org/0802/articles/sicart

[4] Wolf, M. (2001). Genre and the Video Game. *The medium of the video game,* (28.02.2024).

URL https://www.academia.edu/435740/Genre_and_the_Video_Game

[5] Sawyer, J. (2016). 'Pillars of Eternity' and Proper Attribute Tuning. (11.03.2024).

URL https://www.youtube.com/watch?v=fvyrEhAMUPo

[6] Chow, S. (2020). Immersing a Creative World into a Usable UI. (10.08.2024).

URL https://www.youtube.com/watch?v=ONYmBBZiBj8

[7] Nielsen, J. (2000). Why You Only Need to Test with 5 Users. (10.08.2024).

URL https://www.nngroup.com/articles/why-you-only-need-to-test-with-5-use

[8] Nystrom, R. (2014). *Game Programming Patterns.* [s.l.]: Genever Benning.

URL https://gameprogrammingpatterns.com/contents.html

Appendix

I. Glossary

Core mechanics — methods invoked by agents for interacting with the game world [3].

Game engine — a specialised software framework primarily designed for the development of video games, enabling developers to create and manage a game's various aspects efficiently[24].

Game programming patterns — a set of reusable solutions to common problems that game developers encounter when creating games. These patterns are designed to improve code quality, reduce complexity, and increase the flexibility and maintainability of game projects[25].

Render pipeline — a series of operations that take the contents of a Scene, and displays them on a screen[26].

User interface — the user interface lies between the player and the internals of the game. The UI knows all about any supported input and output hardware. It translates the player's input — the button-presses (or other actions) in the real world — into actions in the game world [1].

[24] https://drawandcode.com/learning-zone/what-is-a-game-engine/
[25] https://medium.com/@apoorvmishra05/game-programming-patterns-9e4a877dc942
[26] https://docs.unity3d.com/6000.0/Documentation/Manual/Glossary.html#Renderpipeline

II. Launch Guide

The playable build of *Perpetua* can be accessed in the Accompanying Files in the folder "/Build". After the publishing of this thesis, new versions of the game are planned to also be released on Github. After extracting the specified folder, the game can be launched by running the file "Perpetua.exe".

Minimum recommended system requirements:

- 8th generation i3 CPU
- GTX 950
- 8GB of RAM
- Windows 10 or newer operating system

All objectives and controls are shown in the game.

III. Source code

The source code and assets of *Perpetua* can be accessed in the GitHub repository here (https://github.com/JanMarkusRokka/Perpetua).

IV. Videos of Test Sessions

Video 1	https://drive.google.com/file/d/1O0Tjc3tf_Fk8OQ4bEq3c4-Ut3EVrRT_6/view?usp=sharing
Video 2	https://drive.google.com/file/d/15Oo9HiS-yC-UR24HspKYzix6pn2ImCyw/view?usp=sharing
Video 3	https://drive.google.com/file/d/1E2WYqtaaudeGfIEu0pOMSZFkTcM9hxZl/view?usp=sharing
Video 4	https://drive.google.com/file/d/1iQJR40EYAi9kARkk_YQqtzWn4gndIc5b/view?usp=sharing
Video 5	https://drive.google.com/file/d/10roxIy49UeBNrDvpiORpfhllh-3xtgMQ/view?usp=sharing

V. Accompanying Files

The accompanying files are stored in a compressed file with the following structure:

- /SourceCode – contains the Unity project files of *Perpetua*
- /ThesisBuild – contains the executable and other necessary files to run *Perpetua*
- /Test – contains the tester questionnaire along with the playtesters' answers

VI. License

Non-exclusive licence to reproduce the thesis and make the thesis public

I, **Jan Markus Rokka**,

1. grant the University of Tartu a free permit (non-exclusive licence) to reproduce, for the purpose of preservation, including for adding to the DSpace digital archives until the expiry of the term of copyright, my thesis

   Perpetua - A Turn-Based Role-Playing Game With Runes

   supervised by **Daniel Nael, MSc**.

2. I grant the University of Tartu a permit to make the thesis specified in point 1 available to the public via the web environment of the University of Tartu, including via the DSpace digital archives, under the Creative Commons licence CC BY NC ND 4.0, which allows, by giving appropriate credit to the author, to reproduce, distribute the work and communicate it to the public, and prohibits the creation of derivative works and any commercial use of the work until the expiry of the term of copyright.

3. I am aware of the fact that the author retains the rights specified in points 1 and 2.

4. I confirm that granting the non-exclusive licence does not infringe other persons' intellectual property rights or rights arising from the personal data protection legislation.

Jan Markus Rokka

Tartu, **09.01.2025**