**SE112 Introduction to Software Engineering**

**2023-2024 (II)**

**Assignment 2**

**İclal NAR**

**Emir Kutluhan Arslan**

**Saruhan Mert Ağaya**

* **(13 points)** For your project, gather functional user requirements (FR) – at least 10, and non-functional user requirements (NFR) – at least 3, and classify them according to MOSCOW scheme. Show all the requirements in a table similar to the one below:

|  |  |  |
| --- | --- | --- |
| Functional Reqs ID | Explanation | MOSCOW priority (M, S, C, W) |
| FR1 | **Multiple Endings**: The game should offer multiple endings based on player choices, encouraging replayability | **S** |
| FR2 | **Mythological Quests**: Include quests related to Greek mythology, where players interact with gods, monsters, and legendary figures | **C** |
| FR3 | **Inventory System**: Implement an inventory system for collecting items, artifacts, and magical objects | **M** |
| FR4 | **Puzzle Challenges**: Integrate myth-themed puzzles or riddles that players must solve to progress | **M** |
| FR5 | **Dynamic Dialogue**: Create dynamic dialogues with branching options, allowing players to shape their character’s personality | **M** |
| FR6 | **Epic Boss Battles**: Design challenging boss fights against mythological creatures like Cerberus or Medusa | **C** |
| FR7 | **Underworld Exploration**: Let players explore different realms within the underworld, each with unique environments and challenges | **S** |
| FR8 | **Character Customization**: Provide options for customizing the player character’s appearance, skills, and abilities | **M** |
| FR9 | **Hidden Lore**: Scatter hidden lore, inscriptions, or murals throughout the game world, revealing deeper mythological secrets | **C** |
| FR10 | **Emotional Impact**: Craft emotional moments that resonate with players, such as Charon’s internal struggle or encounters with lost souls | **C** |
| Non-Functional Reqs ID | Explanation | MOSCOW priortiy  (M, S, C, W) |
| NFR1 | **Smooth Performance**: Optimize game performance to maintain consistent frame rates and prevent lag | **M** |
| NFR2 | **Responsive Controls**: Ensure precise and responsive controls for fluid gameplay | **S** |
| NFR3 | **Localization**: Accurate translations for subtitles in various languages to cater to a global audience | **C** |

* **(7 points)** Choose a group of related Functional Requirements and express it in more detail as a user-story.
* **User story:**
* Our game called Styx, The Underground River is a game created to teach mythology to the player while having fun. For example, players can face **mythological guests** such as Zeus, Hades and Ares on the game screens. Moreover, they can not only encounter mythological legends but also engage in **dynamic dialogues** with them through options. At the same time, they can learn **hidden lore** about what the characters have done in the past, thanks to the inscriptions they encounter in the game.
* **(15 points)** For the user-story in Question 2 above, using **PlantUML**, draw a **use-case diagram** representing different components of your software by identifying the actors and tasks in each group of related FRs (components). Note that, your diagrams should include at least one <<include>> and one <<extend>> use-cases. PlantUML scripts should be added to your answer next to the diagram.



@startuml

left to right direction

skinparam packageStyle rectangle

actor Player

rectangle "Styx, The Underground River" {

usecase "Encounter Mythological Characters" as EMC

usecase "Engage in Dialogues" as ED

usecase "Learn Hidden Lore" as LHL

usecase "View Game Screens" as VGS

usecase "Read Inscriptions" as RI

EMC <|-- VGS : <<include>>

ED .u.|> EMC : <<extend>>

LHL .u.|> RI : <<extend>>

}

Player --> EMC

Player --> ED

Player --> LHL

@enduml

* **(20 points)** From the use-case diagrams developed in Question 3, using **PlantUML** draw a class diagram. Show the attributes, and methods (together with visibility symbols -/+) of each class. Relationships (association, aggregation, composition, inheritance) and cardinalities (e.g., 0..1, 1..\*) between classes should also be shown. Don't forget to include PlantUML scrips along with the class diagram.



@startuml

skinparam classAttributeIconSize 0

class Game {

-name:

-version:

+start():

+stop():

}

class Player {

-playerId:

-playerName:

+play():

+pause():

}

class Character {

-characterId:

-name:

-mythology:

+speak():

}

class Inscription {

-inscriptionId:

-text:

+read():

}

class Dialogue {

-dialogueId:

-text: String

+showOptions():

}

Game "1" -- "1..\*" Player : has >

Player "1" -- "0..\*" Character : encounters >

Character "1" -- "1..\*" Dialogue : engages >

Inscription "0..1" -- "1" Character : reveals >

Dialogue "1" -- "1..\*" Inscription : includes >

@enduml

**DEPENDING ON YOUR PROJECT ONLY ONE OF THE QUESTIONS 5 AND 6 (20 points) SHOULD BE ANSWERED.**

* Using **PlantUML**, prepare a detailed sequence diagram to show the interaction between the objects of the classes you’ve used in the class diagram you’ve developed in Question 4. Explain the steps depicted by the sequence diagram.



**Steps:**

1.Player starts the game: The player sends a start message to the game .

2.Game instantiates player: The game responds by instantiating the player object.

3.Player encounters a character: The player sends an encounter message to a character.

4.Character engages in dialogue: The character sends an engage message to the dialogue.

5.Dialogue shows options to the player: The dialogue sends a show Options message to the player ,who then selects an option.

6.Character speaks: The dialogue sends a speak message to the character (c), which then displays text to the player.

7.Player finds an inscription: The player sends a findInscription message to the inscription.

8.Player reads the inscription: The inscription sends a read message back to the player.

9.Player stops the game: Finally, the player sends a stop message to the game ,ending the interaction.

* **(25 points)** Design two user interfaces (mock-ups) for your system using FIGMA ([http://figma.com](http://figma.com/)).

https://www.figma.com/file/K4Kx5xFGs1q4QDwQv1AgzN/SE112Project2?type=design&node-id=0%3A1&mode=design&t=vtMuZH4aSBvfDZHI-1