Requirements

Cohort 4 Group 3

Team Members
Charles Fellows
Elif Gunes
Eve Davey
Hongrui Liu
Leo Owen-Burton
Tomi Olagunju
Will Bannerman

Introduction

The 3NG University Maze is a single player maze-like arcade game which is to be played by those visiting University open days. The game should be interactive and fun, and is limited to a 5 minute play time.

The game is focused towards a casual audience. Within the project, those involved are:

- Developers
- Project managers
- Customer
- Testers
- Players
- Hand-off team

We are limited by the time we have to create the project, and the skills of those involved. We must create the game using Java, but have access to external resources such as assets and libraries. This allows us to expand the scope of our project into areas such as sound effects and game art, which should give 3NG's game a fuller, friendlier feel.

To create this requirements analysis, we:

- Researched industry methodologies on requirements capturing
- Met with Tommy Yuan, the customer, to define exactly what was expected from our game
- Held multiple meetings to discuss requirements with internal project managers and developers to decide the quality threshold, and the depth of the requirements

Due to this project being the first game development to our group, it was important to us to systematically and thoroughly set out requirements to reference throughout, to give a guide to everyone who is participating. In order to achieve this, we give each requirement (both user and system) an ID which can be referenced back to, along with easy-to-read criteria for success. We have followed the MoSCoW method to assign priorities to each requirement.

Websites used for research:

- https://www.modernrequirements.com/blogs/good-software-requirements/
- https://www.genestial.com/how-to-write-clear-and-effective-software-specifications/

User Requirements

ID	Description	Priority
UR_OS	Game should load on Windows, Linux and Mac	SHOULD
UR_THEME	The theme of the game should be that of an arcade	SHOULD
UR_LOSE	The player will lose the game when the 5m timer runs out	MUST
UR_SOUND_SETTINGS	The user should be able to turn off/on the sound/music	SHOULD
UR_PLAYABILITY	The game should be fully playable, intuitive, and enjoyable for families, children and anyone impaired	SHOULD
UR_RESTART	It should be easy to restart the game when a new player arrives.	COULD
UR_MAP	A map must show the maze layout with limited visibility of where the player can go	MUST
UR_MAZE_DESIGN	The maze must reflect a university-like environment, built from scratch with clear paths and boundaries.	MUST
UR_OBSTACLES	At least five visible events that hinder the player’s progress	MUST
UR_BOOSTS	At least three visible events that help the player	MUST
UR_SCORING_SYSTEM	Escaping quickly improves the base score. Interacting with certain events can increase or reduce the score depending on difficulty or choices.	MUST
UR_WIN	The player wins by achieving the best score through fast completion and smart decisions.	MUST
UR_PAUSE_FUNCTIONALITY	The player must be able to pause the game at any time, especially on easier difficulty levels.	MUST
UR_LEADERBOARDS	The players name will be taken, and their score will be shown on a leaderboard compared to other players	COULD

System Requirements

ID	Description	Priority	Linked User Requirement

FR_PAUSE_MENU	The player should be able to pause the game at ANY time while playing.	MUST	UR_PAUSE_FUNCTIONALITY
FR_TIME_TRACKING	The system must track the total time elapsed from game start until the player escapes or the 5-minute limit expires	MUST	UR_WIN,UR_LOSE
FR_DETECT_ESCAPE	The system must detect when the player reaches the maze exit and trigger the end-of-game state	MUST	UR_WIN,UR_LOSE
FR_CALCULATE_SCORE	The system must compute final score from time + event modifiers (positive/negative/hidden)	MUST	UR_WIN,UR_LOSE
FR_DISPLAY_SCORE	The system must display the final score and result (win/loss) at the end of the game	MUST	UR_WIN,UR_LOSE
FR_LOAD	The system must load on Mac, Windows and Linux within 20s from clicking the executable.	SHOULD	UR_PLAYABLE
FR_RESTART	The system should allow the user to restart the game from the pause menu	SHOULD	UR_RESTART
FR_INPUT_LOCK	The system should disable player input once the loss state is triggered.	SHOULD	UR_LOSE
FR_GAME_OVER	The system shall automatically end the game and display a “Game Over” message when the timer reaches zero (or s	SHOULD	UR_LOSE
FR_QUICK_RESTART	The system shall include a restart option accessible from pause and end screens.	SHOULD	UR_RESTART
FR_STATE_RESET	The system shall reset the timer, score, and maze state to default values on restart.	SHOULD	UR_RESTARt
FR_BOUNDARY_CONTROL	The system shall prevent the player from moving beyond defined maze limits.	SHOULD	UR_MAP
FR_MAP_UPDATE	The system shall dynamically reveal new areas as the player explores.	SHOULD	UR_MAP
FR_MAP_DISPLAY	The system shall display a map showing explored areas only.	SHOULD	UR_MAP

FR_OBSTACLE_EFFECT	The system shall apply the corresponding hindrance effect upon player interaction.	SHOULD	UR_OBSTACLE

ID	Description	Priority	Linked User Requirement	Fit Criterion
NFR_GAME_TIME	Game must contain a 5m timer.	MUST	UR_LOSE,UR_WIN	The timer must measure time accurately within ±0.1 seconds
NFR_GAME_OBSTACLES	The game must contain barriers which hinder player progress.	MUST	UR_OBSTACLES	The game should have atleast 5 obstacles.
NFR_UX_USABILITY	The game interface shall be intuitive for first-time players	SHOULD	UR_PLAYABILITY	Children above 5 years old should be able to use the game.
NFR_SETTINGS	The system should allow the player to change settings of the game at any time (e.g. sound settings)	SHOULD	UR_SOUND_SETTINGS	Sound should stop/start throughout the game when the action is taken.
NFR_LEADERBOARD	The system should log the players score and update a leaderboard (Player name - Score) in order.	COULD	USER_LEADERBOARDS	Players' scores are ordered on the leaderboard screen.