Req1

Cohort 3 Group 5

Members:

Amity Van Rooyen
Cassian Kanhukamwe
Dhruv Madan
Gilda Grimes
Jerry Anish
Matt Ritchie
Oakley Fiddler
Ruby Brown

In order to elicit our requirements for the Greenfield Development stage of the assessment, we engaged in client interviews with our customer, in which we asked them details relating to the different features and requirements of the maze game as detailed in the project brief. This interview followed up with questions both in person and over email, all of which added to and further developed the requirements over time.

The interview identified core gameplay expectations (e.g., platform, control scheme, duration, and objective) and clarified both functional needs (what the system must do) and non-functional qualities (how it should perform).

We opted for a table format with ID, requirement number and a brief description of each requirement as it's easy to understand and display user requirements. It's also easy to understand and track which requirements have been met and which are an ongoing process as the project develops. Moreover, towards the end of the project we are able to see which user requirements may not be met in time for the submission.

The hidden, negative and positive events to choose listed in the project brief were selected based on what ideas we came up with would be simplest to implement and what would make the game most fun to play.

The system shall enable users to play a game for a maximum of 5 minutes playing against various obstacles along the way in a university-maze like environment. The system is a Java-based 2D top-down (bird's-eye) escape game where the player navigates a university map, interacts with NPCs and objects, collects keys, and unlocks doors to reach the final goal within a timed session.

User Requirements Table

Requirement Number	**ID**	**Description**	**Priority**
1	UR_CREDITS	At the end of the game, players shall see a short credits screen, showing which tools or libraries were used and who created the game.	SHOULD
2	UR_PERSPECTIVE	Game shall be shown from above, in a 2D/birdseye view, so that the environment, character and interactions are seen from a top down view.	SHALL
3	UR_MAP_LENGTH	It should be short enough for players to complete the game within the provided time.	SHALL
4	UR_FINAL_GOAL	The player's main goal in the game is to escape from the university; the primary objective is to make the game fun while including interactions with the environment and having a bonus/reward system.	SHALL
5	UR_SPEED_BOOST	Power Up: There shall be collectable items that give the player a temporary boost in speed for a short time.	SHALL
6	UR_IMMUNITY	Power Up: There shall be collectable items that if passed through, the player is protected from negative events such as being caught by professor.	SHOULD
7	UR_TELEPORT	Power Up: There shall be collectable items that allows the player to move to a helpful location or allow them to bypass an obstacle or a surprise event etc.	SHOULD
8	UR_DOOR	Obstacle: A Door blocking progress in the game until the player finds a key to unlock the door.	SHOULD
9	UR_KEY	Players shall be able to find a key that unlock a specific door, a key is a collectable item.	SHOULD
10	UR_PROFESSOR1	Obstacle: Some professors shall block the path of a player and ask a question, the user must interact and answer correctly to pass. Surprise Event: If answered incorrecly, the professor chases the player; if caught, the game is over.	SHALL
11	UR_PROFESSOR2	Obstacle: A Professor who slowly follows the player. If caught, the player is frozen in space and the game ends.	SHOULD
12	UR_FRESHERS_FLU	Obstacle: If the player touches a moving 'Freshers Flu' object, their movement is temporarily slowed down.	SHALL
13	UR_WETFLOOR_SIGN	Obstacle: If the player passes over a Wet floor sign	SHOULD

		their movement is temporarily halted for a given time before continuing.	
14	UR_DUCK	Surprise Event: If the player walks into a duck-chase area, duck shall chase them, if the player is caught by the ducks, the game ends.	SHOULD
15	UR_LAKE_TOKEN	Surprise Event: If the player goes into the lake and collects it, the player will be able to evade a duck chase and instead chase ducks without being caught.	SHOULD
16	UR_TIMER	The game shall include a visible clock showing how long the game has been in progress, but will allow the player to pause and unpause the game at any time.	SHALL
17	UR_EVENT_COUNTER	The game shall include a simple counter showing how many of each event have been interacted with so far during the current game.	SHALL

Functional Requirements

ID	Description	User Requirements
FR_DISPLAY_CREDITS	The system shall display an end game screen showing credits, including required libraries and tools used.	UR_CREDITS
FR_TOP_VIEW	The system shall render the game in a birdseye view using Swing graphics in Java.	UR_PERSPECTIVE
FR_SESSION_LIMIT	The system shall end the game session automatically after 5mins of play.	UR_MAP_LENGTH, UR_TIMER
FR_TIMER_DISPLAY	The system shall display a in-game live countdown clock showing remaining time, which can be paused and resumed.	UR_TIMER
FR_SCORE_TRACKER	The system shall include a score tracker that starts fron 500 and reduces by 50 points for every 30s passing.	UR_TIMER
FR_END_GAME	The system shall end the game if a negative event is interacted with, and go to the end game screen.	UR_PROFESSOR1, UR_PROFESSOR2
FR_COLLISION	Blocks player from entering restricted restricted tiles on the map using Map.isCollliding().	UR_MAP_LENGTH
FR_UNLOCK_DOOR	Unlocks door when matching key is collected.	UR_DOOR, UR_KEY
FR_POINTS_UPDATE	Decrements 30 points every 30s while game is not over.	UR_TIMER

FR_SUCCESS_SCREEN	Success screen when player successfully completes the maze, showing final score and final time aswell.	UR_FINAL_GOAL
FR_FAIL_SCREEN	Failure screen showing final time and points along with message stating failure.	UR_FINAL_GOAL

Non-Functional Requirements

ID	Description	Linked user requirements
NFR_GAME_PERFORMANCE	Smooth animation of players or characters.	UR_PERSPECTIVE, UR_MAP_LENGTH
NFR_OPERABILITY	The game will be operable without any training, using only the keyboard.	UR_FINAL_GOAL
NFR_RESPONSIVENESS	Player input shall take effect soon after key press, ensuring smooth game.	UR_FINAL_GOAL, UR_SPEED_BOOST
NFR_VISUAL_CLARITY	All game elements shall use clear colours and readable text.	UR_PERSPECTIVE, UR_TIMER
NFR_RELIABILITY	The game shall not crash or lag during game play and shall finish cleanly on game completion or timeout.	UR_FINAL_GOAL, UR_TIMER
NFR_ACCESSIBILITY	Any on-screen messages and prompts shall be clear and free from any jargon users may not understand.	UR_FINAL_GOAL
NFR_CODE_MAINTAINABILITY	Should be able to add new obstacles, elements or items without changing main program.	UR_SPEED_BOOST, UR_IMMUNITY, UR_TELEPORT
NFR_EVENT_BALANCE	Temporary effects such as Speed Boost, Immunity, Freshers flu, Wet floor sign etc shall last short time so that theyre not in excess to keep the game fun to play for users.	UR_SPEED_BOOST, UR_IMMUNITY, UR_FRESHERS_FLU