Requirements

Group Number: Cohort 2 Team 7

Group Name: pickNmix

Group Member Names:

Sameer Minhas

Phyo Lin

David Lun

Alex McBride

Harry Muir

Introduction

The process of eliciting and negotiating the requirement for the game began with a comprehensive examination of the product brief that was provided by the customer (lecturer) as it outlined the main objectives of the game and expectations of the customer from the game. In order to get a deeper insight into the requirements of the game development, our group created a series of questions that can assist us to clarify the specific features and solve the potential ambiguities that can arise during the development. After preparing the questions, our team conducted an interview with the customer with the purpose of validating the necessary requirements and discussing the additional things related to the game development. Based on the answers that we got from the customer, we have discovered the additional requirements that are essential for the game dynamics and usability but are not clearly mentioned in the provided product brief. After thorough negotiation with the customer and engaging discussions between the team members, we finalised the requirements of the game which satisfied the customer's preferences.

In this report, the requirements will be presented by dividing them into two different types: user requirements and system requirements. For the system requirements, we will present them in two categories called functional requirements, and non-functional requirements.

The users requirements are presented in a table which has three different columns called ID, Description and Priority. The first column ID represents the unique identification for each user requirement and it makes it easier to reference them easily throughout the project. Secondly, the Description column is the explanation for each user requirement for clearer understanding. The last column Priority is included to point out the level of the importance of each user requirement in our system. This format of presentation makes us have clearer vision on the user requirements.

For the functional requirement, the columns called ID, Description, User Requirement are included in the presented table. Similar to the user requirements, ID is the unique representation for each. The Description will emphasise on the core functionalities that the system must have to fulfil each user requirement. The final column called the User Requirement will express which user requirements will be handled by these system requirements. This method of presenting functional requirements ensures a clear path from user requirements to implementation by breaking down user goals into system actions.

The table with four columns named ID, Description, User Requirement and Fit Criteria will be used for presenting the non functional requirements. This format can reflect the qualitative nature of the non-functional requirements by emphasising system quality, performance, and usability. The purpose of using the first two columns is similar to the previous requirements' table. The final column Fit Criteria can be used as the measurable standards in order to check how each non-functional requirement is met for the user requirements during development and testing

User Requirements

ID	Description	Priority
UR_BUILDINGS	The system shall allow the user to be able to place buildings on the map	Shall
UR_MAP	The system shall provide a map to the user	Shall
UR_STUDENT_SATISFACTION	The system shall provide a student satisfaction metric to the user	Shall
UR_EVENTS	The system shall provide at least 3 events to the user that impact gameplay	Shall
UR_CONSTRAINTS	The system shall be constrained for the user	Shall
UR_DIFFICULTY	The system may provide more than one difficulty game mode to the user	May
UR_UI	The system shall be intuitive for the user	Shall
UR_UX	The system shall offer a pleasant user experience to the player.	Shall

System Functional Requirements

ID	Description	User Requirements
FR_BUILDING_PLACEMENT	The system shall allow the user to be able to place buildings anywhere on the campus map, subject to constraints such as terrain and available space	UR_BUILDINGS
FR_BUILDING_COUNTER	The system shall let the user be able to track how many of each building have been placed by them	UR_BUILDINGS
FR_BUILDING_STUDENT_SATISFACTION	The placement of buildings by user considers multiple factors such as location and interaction with other nearby buildings to calculate the student satisfaction	UR_BUILDINGS
FR_BUILDING_TOOLS	The system shall provide intuitive tools to the user for placing and moving buildings	UR_BUILDINGS
FR_BUILDING_TYPES	The system should make available various types of buildings (e.g lecture hall, CS building, university library) with their own function/s (e.g sleep, eat, study)	UR_BUILDINGS
FR_MAP_INFO	The system shall provide a static/fixed top-down grid based map to the user	UR_MAP
FR_MAP_SAME	The system shall provide the same map to all users	UR_MAP
FR_MAP_PATHS	The map shall have predetermined paths and connectivity	UR_MAP
FR_STUDENT_SATISFACTION_E	The system should provide a satisfaction metric to the user at the end of the game (e.g numerical score, grading system,visual indicator)	UR_STUDENT_SATISF

ND		ACTION
FR_STUDENT_S ATISFACTION_F ACTORS	The student satisfaction system shall be calculated based on factors that contribute to student wellbeing	UR_STUDE NT_SATISF ACTION
FR_EVENTS_TY PES	The system shall have different event types (e.g random, scripted and player-triggered events)	UR_EVENT S
FR_EVENTS_RE ACTION	Based on user's reaction to events, the system shall positively/negatively impact the student satisfaction	UR_EVENT S
FR_GAME_DUR ATION	The system shall be a 5 minute timer before the user's game ends	UR_CONT RAINTS
FR_DIFFICULTY _BASELINE	The system shall have at least a single baseline difficulty game mode	UR_DIFFIC ULTY
FR_DIFFICULTY _GAME_MODE	The system may have multiple difficulty game modes	UR_DIFFIC ULTY
FR_DIFFICULTY _HARD_IMPACT	The system shall make hard mode affect multiple factors to increase difficulty e.g: have multiple events at same time, more emphasis on random events, larger impact of each event on student satisfaction, different scaling for student satisfaction	UR_DIFFIC ULTY
FR_DIFFICULTY _EASY_IMPACT	The system shall make the normal/easier difficulty modes affect multiple factors to decrease difficulty e.g: have less random events, more predictable events, a maximum of one event happening at any given time, smaller impact of each event on student satisfaction, different scaling for student satisfaction	UR_DIFFIC ULTY
FR_UI_HUD	The system shall have a HUD that provides gauges and previews. Gauges help players understand what is currently happening in the game. Previews let players know the immediate effect of their actions	UR_UI
FR_UI_TUTORIA L	The system shall provide a tutorial to the user	UR_UI
FR_UI_BUILD_M ENU	The system shall have a building selection menu where the user can choose a building of their choice at a cost	UR_UI
FR_UI_PAUSE_ MENU	The system shall have a pause menu for the user in case they want to take a break or they get interrupted	UR_UI
FR_UI_NAVIGATI ON	The system shall be easy to navigate for the user from the map to the menu design	UR_UI

System Non-functional Requirements

ID	Description	User Requirements	Fit Criteria
NFR_PLATFORMS	The system shall operate as a computer application on multiple platforms such as (Windows, MacOS, Linux)	UR_CONSTRAINTS	Game can run on multiple computers with different platforms
NFR_INPUT	The system shall support mouse input as the primary interaction device; other inputs (keyboard, controllers) are optional.	UR_CONSTRAINTS	Game can be fully controlled with a mouse; other devices are not required
NFR_BUILD	The system shall restrict building placement on certain map areas (e.g., lakes, hills, forests)	UR_CONSTRAINTS	Players cannot place buildings on terrain marked with placement restrictions.
NFR_FULLSCREEN	The system shall be optimised specifically for fullscreen	UR_CONSTRAINTS	Game operates smoothly in full-screen on all standard screen sizes.
NFR_NAVIGATE	The system shall allow map navigation exclusively through mouse dragging, with no scroll or zoom functions.	UR_CONSTRAINTS	Players can navigate the map only by dragging with the mouse; scroll and zoom functions are disabled.
NFR_USABILITY	The system will have a simple, intuitive UI to able to easily execute all the functionalities in the game	UR_UX	Over 90% of the players can complete the game within that 5 minutes.
NFR_TUTORIAL_EFFECTIVENESS	The system will be easy to learn how to play.	UR_UI	90% of new players who complete the tutorial should be able to perform basic tasks
NFR_PERFORMANCE	The system will be able to be run smoothly without any delay.	UR_UX	The longest delay that can happen in game will be less than 2s.
NFR_STABILITY	The system will be stable and consistent even when the multiple tasks are happening at the same time.	UR_UX	99% of games will not crash
NFR_DESIGN	The design of the game, animation, should be appealing to the target audience.	UR_MAP	99% of players will actively engage during gameplay.