Requirements

Windows 007
Cohort 3 Team 7

Stephen Lavender
Sam Leach
Daron Lepejian
Ding Lim
Hasan Majid
Joshau Marshall-Law

0.1 Requirements Elicitation and Presentation

We began by individually familiarising ourselves with the product brief and then organised a small scale meeting where we discussed the product brief further and clarified any queries we had as a group. The conclusion from this meeting was that the product brief was very open and that it was important to set up an interview with our client, Dr. Tommy Yuan.

In preparation for this interview we created a shared client interview script to clarify the requirements of the brief. This interview was very important to the development of UniSim as it meant decisions about how the final product would turn out were based on the clients needs and not our needs.

Following the interview, we conducted a requirements discussion meeting where we formalised a list of requirements based solely on the product brief and the client interview.

In order to present the user requirements and system requirements, we chose to separate the system requirements into two sections: functional and non-functional. The user requirements section focuses on how the users' experience and the system requirements section focuses on the game logic. This structure was chosen as we can distinguish between the needs of the user and the functionality of the game. We also gave each requirement success criteria so we can objectively measure whether we have met a specific requirement and can focus development towards meeting the requirements and therefore the product brief.

0.2 User Requirements

ID	Description	Priority
UR_PLACE_BUILDING	The player should be able to place different types of buildings on the map (learning center, student dorm, diner, and recreation center)	Shall
UR_TIME_LIMIT	There should be a maximum time limit of five minutes	Shall
UR_GRAPHICS	The game should have high-quality, visually distinct graphics that allow the player to differentiate between the different types of obstacles, buildings, and other in-game features (roads). This ensures that the game is clear to the player and helps them quickly recognize the different aspects of the game	Shall
UR_EMERGENCY_STOP	The player shall be able to restart or exit the game at any time without experiencing a crash	Shall
UR_STABLE_PERFORMANCE	Players expect smooth performance with minimal crashes or bugs	Shall
UR_SCALABILITY	The game should be scalable across different platform dimensions (e.g., large screens, and projectors)	Shall

0.3 System Requirements

ID	Description	User Requirements
FR_POSITION_BUILDING	The system shall allow the players to place the different types of buildings on the game grid by clicking on the building and dragging it to the place of their choice	UR_PLACE_BUILDING
FR_BUILDING_TYPE	The system should allow the player to place at least one of each building type on the map grid	UR_NUMBER_OF_BUILDINGS
FR_FRAME_RATE	Ensure that the game runs at a consistent frame rate to increase interactivity and user satisfaction	UR_STABLE_PERFORMANCE

0.4 Non-Functional Requirements

ID	Description	Req. ID	Fit Criteria
NFR_AVAILABILITY	The game should be available at all times	UR_STABLE	Uptime: Over 99% at all times
NFR_SYS_REQUIREMENTS	The system should run on all PCs requiring minimal hardware resources	UR_SCALABILITY	Compatible with all basic PC hardware and software
NFR_OPTIMISATION	The system shall be optimized to run without crashes	UR_STABLE	The game should load within 5 seconds of launching
NFR_UI	The UI shall be user-friendly and make the player intuitively understand the functionality of the game	UR_GRAPHICS	UI should be intuitively understood and easily comprehended, making the different objects in the game visually distinguishable
NFR_PAUSE	The game shall have functionality that allows the player to pause the game at any time	UR_EMERGENCY_STOP	The game should be pausable at the request of the player
NFR_FAST_RESPONSE	The game shall have fast responsiveness to the actions of the player	UR_RESPONSIVENESS	Each action the player makes should be represented in the game in < 10ms