

# **ENG 1 - Requirements**

Jacob Tangye

Joseph Wipat

Isaac Williams-Mayo

Hasan Syed

Huzaifa Thakur

Alex Worsley

Charlie Taylor

## **Introduction to Requirements Elicitation and Negotiation**

Requirements Elicitation and Negotiation are critical phases in the development of any software project, creating the foundations necessary for a system that meets all of the needs of the customer. Requirements elicitation was conducted using techniques such as conducting customer interviews and analysing existing games similar to the one we are building. This process was designed to identify the important features that the game has to have, interactions the player will have with the game and system limitations.

After gathering the initial requirements, negotiation was used to create a balance between different stakeholder priorities. For example making gameplay complex enough to be fun and engaging for the end user but not too complicated that it would take the average player a long time to understand. Negotiation helped with setting clear boundaries for what the game needed to include and what should or may be included (“shall”, “should” and “may” labels for requirement priority).

## **Presentation of Requirements**

The requirements for the game are presented in a structured format to improve their clarity and traceability. They are split into three categories:

### **User requirements (UR)**

These are statements that outline the functions that a user should be able to perform using the proposed system. An example of this would be placing down or moving a building.

### **Functional requirements (FR)**

These specify the specific functions or features the system must perform to fulfil user requirements. Each functional requirement relates to a user requirement.

### **Non functional requirements (NFR)**

Non functional requirements are physical qualities that the system needs to have in order to meet the user requirements. These each have a fit criteria which measurable benchmarks that the system must meet in order for the requirement to be satisfied, such as performance, usability, or reliability. Similarly to the functional requirements, each non-functional requirement references a user requirement.

All of the requirements are formatted with unique IDs to allow for easy referencing.

Over the course of the development of the project, some requirements needed to be altered and some needed to be added. This was due to new stakeholder insights and evolving project needs. An example of this was sound effects. In the initial gathering of requirements, sound effects weren't initially considered. However after feedback user tests were analysed, it became clear that sound effects would significantly improve user immersion. This led us to adding the new requirement.

SSON:

The system will be a single-player game that allows players to design and build their own university campus from scratch, strategically placing buildings to create a fun and intuitive environment that maximises student satisfaction. The game should provide a cohesive and immersive experience to the user.

User requirements:

ID	Description	Priority
UR1_CAMPUS_CONSTRUCTION	The player shall be able to construct a university campus from scratch by placing various buildings and activities.	Shall
UR2_EDUCATIONAL_BUILDINGS	The player should be able to place at least one type of building where students can attend classes, such as a lecture hall or library.	Shall
UR3_RESIDENTIAL_BUILDINGS	The player should be able to place at least one type of building where students can live and sleep, like student accommodation.	Shall
UR4_RECREATIONAL_BUILDINGS	The player should be able to place at least two types of buildings or areas for student recreational activities.	Shall
UR5_DINING_BUILDINGS	The player should be able to place at least one type of dining building where students can eat, such as a cafeteria building.	Shall
UR6_BUILDING_COUNT	The player should be able to see the total number of buildings that have been placed down.	Shall
UR7_PAUSE_FUNCTIONALITY	The player should be able to pause the game at any time for convenience.	Shall
UR8_MOVE_BUILDINGS	The player should be able to move buildings after placing them to improve campus layout or respond to events.	Should
UR9_TIME_ELAPSED	The player should be able to see how much time has passed within the game session.	Shall
UR10_MUTE	The player shall be able to mute the volume of the game at any time.	May
UR11_SOUND_EFFECTS	Sounds shall play that will give the player auditory feedback on their actions.	May
UR12_BUILDING_RESTRICTIONS	The user will not be able to place buildings outside the bounds of the map, and will have limits to the number they can place.	Shall
UR13_REMOVE_BUILDINGS	The player will be able to remove buildings that they place, should they wish to restructure their university campus.	Should

Functional Requirements

ID	Description	User Requirements
FR1_DISPLAYED_BUILDINGS	The game will display the different building options the player can choose from. It should provide a description for each building type and what they do.	UR_CAMPUS_CONSTRUCTION
FR2_EDUCATIONAL_BUILDING	The system will allow the user to place down a	UR_EDUCATIONAL_BUILDINGS

S	lecture hall. This will increase student satisfaction the closer they are to the residential buildings. But student satisfaction will decrease if they are near recreational buildings.	
FR3_RESIDENTIAL_BUILDINGS	The system will allow the user to place down a student halls building. Student satisfaction will be calculated based on how close this building is to the other types of building. Having more than three residential buildings within a certain distance of each other will cause student satisfaction to reduce.	UR_RESIDENTIAL_BUILDINGS
FR4_RECREATIONAL_BUILDINGS	The system will allow the user to place down a gym and a pub. Student satisfaction is based on how close these buildings are to residential buildings, but too close and the noise will reduce student satisfaction.	UR_RECREATIONAL_BUILDINGS
FR5_DINING_BUILDINGS	The system will allow the user to place down a restaurant. This is where students will be able to eat. Student satisfaction is based on how close these buildings are to residential buildings.	UR_DINING_BUILDINGS
FR6_BUILDING_SELECTION	The game will allow the player to select which building type they want to place down.	UR_CAMPUS_CONSTRUCTION
FR7_BUILDING_PLACE	The game will allow the player to place a building they have selected onto the map.	UR_CAMPUS_CONSTRUCTION
FR8_BUILDING_COUNT	The game will display a counter that increments every time a building is placed down. This counter will be decremented if a building is moved.	UR_BUILDING_COUNT
FR9_PAUSE	The game will allow the player to pause the game at their convenience, the player will still be able to interact with their campus but time will not pass.	UR_PAUSE_FUNCTIONALITY
FR10_MOVE_BUILDINGS	The game will allow the player to move buildings from their original location to another, appropriate space to respond to events.	UR_MOVE_BUILDINGS
FR11_BUILDING_RESTRICTIONS	The game will not allow the player to place buildings in invalid locations, e.g. on other buildings, on map features, outside the bounds of the map.	UR_BUILDING_RESTRICTIONS
FR12_MUTE_FUNCTIONALITY	The system will allow the player to mute the game at any time, eliminating all sound from the game.	UR_MUTE
FR13_DISPLAY_TIME	The system shall display the amount of time that is left before the game ends.	UR_TIME_ELAPSED
FR14_SOUND_EFFECTS	The system may play sound effects to provide auditory feedback to the player when they perform actions.	UR_SOUND_EFFECTS
FR15_REMOVE_BUILDINGS	The system shall allow the player to remove or demolish buildings they have previously placed.	UR_REMOVE_BUILDINGS

## Non Functional requirements

ID	Description	User Requirements	Fit Criteria
NFR1_USABLE_UI	The game shall have an intuitive user interface that allows players to easily navigate and perform actions.	UR_CAMPUS_CONSTRUCTION UR_MOVE_BUILDINGS	The interface shall allow >90% of new players to locate and use primary functions, within <5 minutes of starting the game.
NFR2_AUDIO_CONTROL	The mute function shall work immediately when the player chooses to mute/unmute the game.	UR_MUTE	The game shall stop/resume all audio output <0.1 seconds after the player activates the mute function.
NFR3_PAUSE_RELIABILITY	The pause functionality shall be responsive and not cause the game to freeze or crash.	UR_PAUSE_FUNCTIONALITY	The game shall pause or resume <0.1 seconds of the player's command, without causing errors, in 100% of test cases.
NFR4_IMMEDIATE_RESPONSIVENESS	The game shall provide immediate feedback to player actions, such as placing, moving, or removing buildings.	UR_CAMPUS_CONSTRUCTION UR_MOVE_BUILDINGS UR_REMOVE_BUILDINGS	The system shall respond to player inputs <0.1 seconds, ensuring seamless interaction.
NFR5_MINIMAL_LOADING	The game shall have minimal loading times when performing actions like placing or moving buildings.	UR_CAMPUS_CONSTRUCTION UR_MOVE_BUILDINGS	Loading times for such actions shall be <0.5 seconds on systems meeting the minimum hardware requirements.
NFR6_PERFORMANCE_SCALABILITY	The game shall maintain performance standards even as the number of buildings increases.	UR_CAMPUS_CONSTRUCTION UR_BUILDING_COUNT	The game shall maintain >30 FPS when up to 100 buildings are placed, on systems that meet minimum specifications.
NFR7_STABILITY_RELIABILITY	The game shall operate reliably without crashes during building placement, movement, or removal.	UR_CAMPUS_CONSTRUCTION UR_MOVE_BUILDINGS UR_REMOVE_BUILDINGS	The game shall not crash during >95% of 5-minute game sessions in testing over 5 hours of gameplay.
NFR8_DISPLAY_READABILITY	The time elapsed and building count displays shall be easily readable and will update in real time.	UR_TIME_ELAPSED UR_BUILDING_COUNT	Displays shall update within <0.1 seconds of changes, with a readable text size on a standard 1920x1080 resolution.
NFR9_BUILDING_RESTRICTION_ENFORCEMENT	The game shall prevent players from placing buildings outside allowed areas or exceeding placement limits.	UR_BUILDING_RESTRICTIONS	Attempts to place buildings in restricted areas or beyond limits shall be blocked 100% of the time during testing.
NFR10_PLATFORM_COMPATIBILITY	The game shall be compatible with commonly used operating systems	UR_CAMPUS_CONSTRUCTION	Full functionality on Windows 10 and above, macOS 10.13 and above, and Ubuntu 18.04 LTS and above.
NFR11_TRANSITION_LOAD_TIMES	The game shall have minimal load times when starting up or transitioning between screens.	UR_CAMPUS_CONSTRUCTION	Loading screens shall not exceed 5 seconds on systems that meet minimum hardware requirements.