

Requirements

Setting out requirements is essential for successful game development. Establishing requirements ensures all team members have clarity of the aims and a strong baseline for development guidance. This is key for good communication and satisfying our stakeholders (customers).

In order to design our requirements we began by scoping out the brief and designed a set of questions for our customer. We scheduled a meeting where he addressed our questions and provided additional guidance on the goals and requirements of the product.

Below is a snapshot of our questions and answers from the customer

Should we implement various game modes
NO
Should we implement multiple playable sprites
Yes- Very Important
Should we implement multiple stages
NOT YET - customer stated in assessment 2 this can be used

After our discussion we gained a clear understanding of what User requirements, non-functional requirements and functional requirements we needed in order to please our stakeholder. We noticed an emphasis on the principle of KISS(KEEP IS SIMPLE STUPID) in the briefing, so we ensured to prioritise simplicity. To facilitate the software development process, we then formalised these requirements and applied fit criteria to allow us to easily check if a non-functional requirement has been met sufficiently well.

For user requirements we additionally applied a priority, either "Should", "Could" or "Shall". These were set after the meeting in line with the expectation of our customer. "Shall" was used where the client expressed a hard requirement. "Should" was used where the client wants a feature but doesn't have a make or break need for it. "Could" was selected where the inclusion of a feature is considered optional, typically this applies to aspirational features that may be implemented depending on the progress of the project. As can be seen in the [Project Planning document](#) time is allotted for this as "Nice to have features" in the project timeline.

User requirements

<u>ID</u>	<u>Description</u>	<u>Priority</u>
UR_Student	The user has a sprite they can move to perform actions.	Shall
UR_Map	The game has a world with objectives within and on the edges of the map.	Shall
UR_Scoring_System	The player is given a score at the end of the week for their performance.	Shall
UR_UX	The system will provide clean, distinguishable graphics with clearly labelled buildings and collision points. Menus and buttons will be also clearly labelled	Should
UR_Student_Interact	The student will be able to interact with a range of collisionable items	Shall
UR_Lose_Game	The user will lose the game if they do not study enough, rest enough or have enough leisure time.	Shall
UR_Tutorial	On first start, the user will receive a tutorial explaining how to play.	Could
UR_Win_Game	When the Student has completed his study quota while managing his leisure and energy they win.	Shall
UR_Student_Move	The user has controls to move their student..	Shall
UR_Rest_Area	The user must interact with this area to rest and replenish energy.	Shall
UR_Eating_Area	The user must eat to continue to live, maybe cannot progress until eats.	Shall
UR_Sound	The player hears sound corresponding to actions in the game.	Could
UR_Study_Area	The user may interact with one or more locations to study.	Shall
UR_Rec_Area	The user may interact with one or more locations to complete a recreation activity.	Shall

UR_Choose_Avatar	Users must be able to choose an avatar to represent them in the game.	Shall
UR_Sleep	Character must sleep for 8 hours each night.	Shall
UR_Choose_Action	If a user is not currently occupied they should be able to choose an action to complete,	Shall
UR_Mute	Users should be able to mute the game.	Shall

Functional Requirements

<u>ID</u>	<u>Description</u>	<u>User Requirements</u>
FR_Main_Menu	On start the user can choose to start a new game ,change their avatar or mute the game.	UR_UX UR_Mut UR_Choose_Avatar
FR_Mute	There will be a mute button or volume slider on the main menu.	UR_Mute
FR_Controls	The control scheme will be simple for the user to move and interact with the environment.	UR_Student_Move UR_Student_Interact
FR_Choose_Action	When clicking on a building users will be shown one or more actions to select.	UR_Choose_Action UR_Rec_Area UR_Study_Area
FR_Complete_Action	When the user selects an action the relevant amount of time should pass and the event should be counted.	UR_Rec_Area UR_Study_Area UR_Rest_Area
FR_Sleep	Users should be forced to sleep after a certain number of hours of actions have been taken.	UR_Sleep
FR_Track_Stats	Keep track of how many times each action has been	ALL User Requirements
FR_Energy	The student starts with 100% energy, this depletes as actions are carried out	UR_Student UR_sleep

Non functional requirements

<u>ID</u>	<u>Description</u>	<u>User Requirements</u>	<u>Fit Criteria</u>
NFR_Instructions	Provide the user with a set of instructions which are clear and understandable.	UR_UX	The objectives of the system will be clear and an instruction set will accompany it.

NFR_Bugfixing	Bugs will be fixed and updated	ALL User Requirements	The system will be repaired by developers when game breaking bugs are run into
NFR_Smooth_Gameplay	The game shouldn't crash.If a crash occurs the game should be restartable without any issues.	All User Requirements	The system is operational and available for use approximately 95% of the time.
NFR_Delay	The game should have minimal or no lag	ALL User requirements	The system should not be affected by lag
NFR_Controls	All controls should be similar to other games in the same industry	UR_UX	The system will be operated by a set of recognizable controls
NFR_options	Hotkeys ,sound and window settings will be changeable for the user.	UR_UX	The system will allow users to change settings to their preference
NFR_Map	The map's entities will be distinguishable and we will have affordance in place for all interactables	UR_Student UR_UX UR_student_interact	Users will receive a visual
NFR_Scalability	The game can be updated by other developers and previous code shouldnt break(within reason of changes)	ALL user requirements	The system will be able to accommodate updates to the code whether it be a bigger map or other implementations
NFR_Desktop	The game will have compatibility support for desktop	ALL User Requirements	The system will be playable on a desktop device
NFR_Graphic_clarity	The game will have high quality, consistent graphics which can render across all devices (within reason)	All User Requirements	The system will have high details and textures
NFR_Age_Relative_compliance	The game will abide by PEGI 12 industry ratings and content regulations	UR_UX UR_Student UR_Student_interact	The system won't contain any vulgar language, violent themes or innuendos.