UNIVERSITY OF YORK

DEPARTMENT OF COMPUTER SCIENCE

Requirements
Cohort 2 - Group 18 - Octodecimal

Group Members:

Izz Abd Aziz

Phil Suwanpimolkul

Tom Loomes

Owen Kilpatrick

Zachary Vickers

Michael Ballantyne

Introduction

Requirements were elicited and negotiated through the product brief and an initial client meeting. A product brief document was provided at the start of the project. This sets out the story of the game and the interactions that the user must complete. It specified the control scheme of the game and the timescale for the play. Primary objectives for the users were laid out which were used to create user requirements. Items the game must include were set out which created a basis for requirements. How to successfully win the game was also defined which gave an initial idea of player goals which was important as players of the game will be stakeholders. Other stakeholders identified in the document were the customer (the main stakeholder) and the remainder of the cohort (who will be deciding whether to continue the project). This was used as a starting point to prepare a list of questions to take to the client and ask in order to get a better understanding of their aims and preferences for the project. The list of questions asked in the meeting can be found on the project website. Questions were split into topics to allow for in-depth discussion and follow-up questions were asked as they were thought of in the meeting. The client meeting crucially gave an insight into who the project was targeted towards and what the aim of the project was. It also allowed for features to be assigned priorities and made clear exactly what was and wasn't wanted within the project. The final question asked for any additional requirements that hadn't been discussed already to ensure that nothing had been missed.

This allowed a single statement of need to be formed: "The system shall enable users to play a game based on the life of a university student in which they have interactions that influence their score". User requirements and functional and non-functional requirements are discussed later in this document. They are presented through three tables - user requirements, functional system requirements and non-functional system requirements. Requirements were specified and presented by adapting the guidance given in IEEE 29148-2018 [1]. First, the stakeholder needs and goals as established in the customer meeting were refined to create user requirements. Following this, functional and non-functional requirements were established. To ensure they were well-formed requirements each functional requirement was a requirement that shall be met or possessed by the system to solve the problem and each non-functional requirement is possible to qualify by specific measurable conditions. Wording used followed the conventions specified. A referencing system was used to maintain traceability of all types of requirements. Requirements did not include design decisions or implementation ideas or suggestions.

In addition to these requirements, there were also a few constraint requirements. One was the project constraint of the timeline with the deadline for the first part of this project being non-negotiable and due by 21st March. The other was the design constraint of the game needing to run on Windows Desktop PCs.

Following the release of the update product brief in Assessment 2, we reevaluated our requirements. Many new requirements had to be added to the requirements table, some requirements were amended or removed, and we also reviewed if these additions changed our product statement. We concluded that this did not change, and decided not to hold a second client interview as the additional requirements did not fundamentally change the project or original requirements, although over the course of assessment 2, we were in contact with our client through eMail to receive clarification when needed.

User Requirements

ID	Description	Priority
UR-DEVICE	The user is able to play on all Windows devices with Windows 10 or above.	Should
UR-MENU	The user has a main menu which they can use to navigate to different features of the application.	Shall
UR-CUSTOMISE	The user should be able to personalise their in game character.	Should
UR-WORLD	The user shall move their character around a 2D map, appropriately representative of Heslington - The user shall recognise that the map represents Heslington at full size.	Shall
UR-INTERACT	The user's character shall interact with objects/buildings within the world to complete tasks.	Shall
UR-TIMED	The user shall be timed so that they play through a week of university life of the in-game student character. The game shall inform the user of the current in-game time.	Shall
UR-INFO	The user shall be informed of their character's energy levels	Shall
UR-SOUND	The user may experience music and sound effects.	May
UR-SETTINGS	The user shall access settings to change their Music and SFX volume	Shall
UR-SLEEP	The user shall replenish their character's energy levels by sleeping.	Shall
UR-ACCESSIBLE	The user with colour blind should be able to play the game. Buildings are distinguishable from the surrounding as they have different shapes and each one of them is separated.	Should
UR-DESIGN	The user shall experience a happy/positive theme when playing the game.	Shall
UR-SIMPLICITY	The game shall be easily played by an inexperienced user.	Shall
UR-LEADERBOARD	The user shall see the leaderboard of top 10 scores after completing the game or from the main menu.	Shall
UR-SCORE	The user shall see the player's final score at the end of the game.	Shall
UR-ACHIEVEMENT	The user shall see the player's achievement(s) at the end of the game.	Shall
UR-USERNAME	The user shall enter their name before starting the game	Shall
UR-CREDITS	The user should able to see credits in the main menu	Should
UR-NPCS	The game should have some stationary NPCs (non playable characters) that the user can interact with to receive hints and game strategies.	Should
UR-CITY	The user should be able to visit another map (city) that they can explore during gameplay with unique activities.	Should
UR-AVATAR	The user should select an avatar to play with during gameplay and this avatar collides with objects.	Should

UR-SCORE-BREAKDOWN	The user should be able to view the score breakdown after completing the game.	Should

Functional System Requirements

ID	Description	User Requirement
FR-VIEW	The system shall always use a top-down view point in the third-person	UR-WORLD
FR-NAVIGATE	The game can be played by using the arrow or WASD keys for the player to navigate around the map.	UR-WORLD
FR-SPRINT	The user is able to sprint by pushing the shift key.	UR-WORLD
FR-MENU-OPTIONS	The menu shall provide the user with a list of options including credits, start game, exit game, settings and leaderboard.	UR-MENU
FR-MENU-SAVES	The gamestate cannot be saved until you reach the end, when your score is saved.	UR-MENU
FR-MENU-PAUSE	Pressing escape shall pause the game and navigate to a pop-up menu with options to resume, navigate to settings or exit	UR-MENU
FR-START	Starting the game shall allow the user to choose from a range of avatars	UR-AVATAR
FR-SLEEP	Reaching the end of the day (16 hours) shall lock all other activities other than sleeping	UR-INTERACT
FR-ENERGY-RESTORE	A game shows the depletion of a player's energy when completing an interaction by a set amount.	UR-INTERACT
FR-INTERACT-PLACING	When a player interacts with a building, they shall stay outside the building	UR-INTERACT
FR-INTERACT-MESSAGE	When a player starts to interact with a building, there shall be a pop-up with text and choices	UR-INTERACT
FR-ENERGY-LIMIT	The game shall never allow players to continue with activities other than sleeping once they have no energy left	UR-INTERACT
FR-WEEK	The game shall end after an in-game week	UR-TIMED
FR-TIME	A player completing an interaction shall progress the time by a set amount	UR-TIMED
FR-NO-TICKING-TIMER	Time should not move without completing an activity	UR-TIMED
FR-DEVICE	The game should be runnable on all Windows desktop and laptop with Windows 10 or above.	UR-DEVICE
FR-GAME-SLEEP	The player shall interact with a sleeping location to progress to the next day.	UR-INTERACT
FR-GAME-STUDY-PLACE	The player shall interact with study locations (town, CS building) and make choices at these locations.	UR-INTERACT
FR-GAME-EATING	The player shall interact with eating locations(Piazza, restaurant) and make choices at these locations.	UR-INTERACT
FR-GAME-LEISURE	The player shall interact with leisure locations(walking trail, duck feeding, talking to friends, bar) and make choices at these locations	UR-INTERACT
FR-GAME-WALKING	The player shall interact with the walking trail as a recreational activity	UR-INTERACT
FR-GAME-DUCKS	The player shall interact with the duck feeding platform as a recreational activity	UR-INTERACT
FR-GAME-FRIENDS	The player shall interact with a group of NPCs as a	UR-INTERACT

	recreational activity	
FR-GAME-BAR	The player shall interact with the duck feeding platform as a recreational activity	UR-INTERACT
FR-GAME-STUDY-SESSION	The player should have 1 core study session per day or one day of the week as “catch-up” where player perform 2 core sessions if they missed core study	UR-INTERACT
FR-SCORING-STUDY	The game calculates scoring for studying, with additional points given for studying at different locations each in-game day, fewer points for catch-up days, and 7 maximum study sessions a week (including catch-up)	UR-SCORE
FR-SCORING-EAT	The game calculates scoring when eating. Graph sets to minimum points and rises to maximum X hours after eating to give maximum score for eating at regular intervals.	UR-SCORE
FR-SCORING-RECREATIONAL	The game calculates scoring when performing recreational activities. X points for doing activity on a day but X/n for each repetition of activity to show less satisfaction doing the same activity repetitively	UR-SCORE
FR-SCORING-STREAKS	For every achievement gain for doing streaks, the game will give extra X points.	UR-SCORE
FR-SCORING-SLEEP	The game calculates a lower score if the player had a late sleep.	UR-SCORE
FR-LEADERBOARD-RANK	The game will calculate players’ score and place them in the leaderboard from highest score to lowest score. However the system will only show the top 10 only.	UR-LEADERBOARD
FR-STATS	The game should keep the achievement list a secret until they achieve them. List of achievements includes “Tree?”, “Bookworm”, ”Duck duck go!”, “Jogger”.	UR-ACHIEVEMENT
FR-TIRED	The game will reduce energy given the next day if the player sleeps late.	UR-SLEEP
FR-USERNAME-LIMIT	The game will only allow players to enter names with a character limit of 16, only allowing alphabetical characters and cannot be left blank.	UR_USERNAME

Non-Functional System Requirements

ID	Description	User requirements	Fit criteria
NFR-DOCUMENTATION-ARCHITECTURE	The game shall be accompanied by detailed architecture documentation	UR-INTERACT	6 pages of architecture documentation containing diagrammatic representations and justifications shall be produced
NFR-DOCUMENTATION-CODE	The game code shall be commented and documented	UR-SIMPLICITY	>95% of code should either be self-explanatory or well-documented
NFR-RESILIENCE–MAP	A problem with one map location shall not impact other map locations	UR-INTERACT	In >95% of game plays that experience an issue with one location, all others will not be affected
NFR-SCALABILITY	The game shall support a single player at a time	UR-INTERACT	No more than 1 person will play in 1 game
NFR-OPERABILITY-EXPERIENCE	The game shall be playable by users with no prior experience	UR-INTERACT	>95% of users will find the game easy to understand even if they previously played for 0 hours
NFR-OPERABILITY-TRAIN	Users shall set up the game without needing training	UR-INTERACT	>95% of users will find the set up easy despite having 0 hours of training
NFR-ACCESSIBILITY-ASSET	All game assets shall always be distinguishable by shape as well as colour	UR-ACCESSIBLE	>95% of colour-blind users will be able to access the game
NFR-USABILITY-MESSAGES	Any technical error messages shall be hidden from the user and a user-friendly, plain English message shall be presented instead	UR-DESIGN	<1% of users will see a technical error message when playing the game
NFR-RESILIENCE-GAME-START	The game shall be reliable and start as expected without being unavailable	UR-WORLD	>98% of game starts will be successful
NFR-USABILITY-INSTRUCTION	All game instructions shall be provided in plain English and avoid technical and university jargon	UR-DESIGN	100% of game instructions will be in plain English with no jargon
NFR-ACCESSIBILITY-SOUND	No elements or instructions of the game shall only be indicated by sound	UR-ACCESSIBLE	100% of sounds and music will be supplemental and not necessary
NFR-OPERABILITY-PLAYABLE	The game shall be playable by users who have had no experience of the game features in real life	UR-WORLD	>95% of players shall report that they found it easy to play the game even with 0 hours of university experience
NFR-TIMING-SLEEP	After 16 game hours, the player shall be unable to do anything other than sleep	UR-SLEEP	After 16 hours of game time, players must be forced to sleep in 100% of cases
NFR-TIMING-GAMEPLAY	The game shall last between 5-10 minutes for an average player	UR-TIMED	>90% of players will play for a minimum of 5 minutes and a maximum of 10 minutes

NFR-MAINTAINABILITY-TEAM-CODE	Team members not involved in implementation shall understand what is happening in the code	UR-SIMPLICITY	All team members will be able to understand the code within 1 hour
NFR-USABILITY-POSITIVITY	The game shall be appealing and present the university in a happy and positive way	UR-DESIGN	>90% of users should report that the design of the game was appealing

References

[1] IEEE Systems and software engineering - Life cycle processes - Requirements engineering, IEEE Standard 29148 Second edition, 2018