UNIVERSITY OF YORK

DEPARTMENT OF COMPUTER SCIENCE

Requirements
Cohort 2 - Group 16 (Skloch)

Group Members:

Charlotte MacDonald

Hollie Shackley

Luis Benito

Kaustav Das

Sam Hartley

Owen Gilmore

Introduction from previous group:

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 set 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. Throughout the project, we also liaise with the client to gain clarification on any requirements that seemed ambiguous.

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, for example appropriate use of 'Shall' and 'Should' in the User Requirements table- following the 'Requirements Construct' rules in Section 5.2.4 of IEEE 29148-2018 [1] . 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.

User Requirements

ID	Description	Priority
UR-DEVICE	The game shall be playable on a desktop/laptop.	Shall
UR-MENU	The game shall provide the user 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.	Shall
UR-INTERACT	The user's character shall interact with objects/buildings within the world to complete tasks.	Shall
UR-TIMED	The game shall be timed so that the user plays through a sped up version of a week's worth 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 game may have music and sound effects, controllable by the player. The sound controls should be visible and easy to navigate.	May
UR-SETTINGS	The user shall access certain settings.	Shall
UR-SLEEP	The user shall replenish their character's energy levels by sleeping.	Shall
UR-ACCESSIBLE	The avatar and buildings in the game should be distinguishable by shape and colour. The text throughout the game should be legible.	Shall
UR-DESIGN	The game shall have a happy/positive aesthetic and vibe.	Shall
UR-STUDY-TASK	The player must study at least once per day, with the option to study twice in one day if they missed studying on another day. Studying twice is allowed only once per game.	Shall

UR-COMPLETE-GAME-SCREEN	The player shall automatically return to the main menu once they finish the game.	Should
UR-MAP-LOCATIONS	The map will represent at least some of the Heslington East campus with one location for sleeping, eating, studying, and three for relaxing.	Shall
UR-RECREATIONAL-ACTIVITY-TASKS	At least three recreational activities must be available for interaction in the map, this cannot be sleeping.	Shall
UR-PAUSE-MENU	The user should be able to access the pause menu throughout the game.	Should
UR-ACTIVITY-COUNTER	This counter shall display the number of times the avatar has performed each activity so far and display the final count at the end of the game.	Shall not
UR-RECEIVE-FEEDBACK	The player should receive text feedback after completing some activities.	Should
UR-POINTS-SYSTEM	The game shall reward the player with points for maintaining a healthy work-life balance. Points will be hidden until the end of the game.	Shall
UR-POINTS-PENALTY	Contralily, the game shall penalise the player with points for not maintaining a healthy work-life balance.	Should
UR-ACHIEVEMENTS	The game will have a number of hidden achievements (at least 3) that users will unlock for additional points by meeting specific requirements in game	Shall
UR-LEADERBOARD	The game will have a leaderboard with the 10 people who have achieved the highest in game score.	Shall

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	Manual
FR-START	Starting the game shall allow the user to choose from a range of avatars	UR-CUSTOMISE	testSelectAvatar1 testSelectAvatar2
FR-ANIMATIONS	The game must have some level of animation	UR-DESIGN, UR-ACCESSIBLE	Manual

FR-INTERACT-ANIMATION	When a player interacts with a building, they shall stay outside the building	UR-INTERACT	Manual
FR-INTERACT-TEXT	When a player starts to interact with a building, there shall be a pop-up with text and choices	UR-INTERACT	canStudyAtCompSciBuilding canEatAtRonCookeHub canGoToSleep canEatAtRonCookeHub
FR-MAIN-MENU	Going to the main menu shall give the user the choice of credits, start game, exit game and audio settings	UR-MENU	Manual
FR-SAVE	Games shall not be saved	UR-MENU	Implicit (Never implemented)
FR-PAUSE-MENU	Pressing esc shall pause the game and navigate to a pop-up menu with options to resume, navigate to settings or exit	UR-PAUSE-MENU	Manual
FR-NAVIGATE	The user using the arrow keys shall cause the player to navigate around the map	UR-WORLD	PlayerTests
FR-SLEEP-TIME	Reaching the end of the day (16 hours) shall lock all other activities other than sleeping	UR-INTERACT	attemptActivityTooLate
FR-ENERGY-INTERACTION	A player completing an interaction shall deplete their energy by a set amount	UR-INTERACT	canEatAtRonCookeHub
FR-WEEK	The game shall end after a week	UR-TIMED	GameOverTests
FR-ACTIVITY-TIME	A player completing an interaction shall jump the time along by a set amount	UR-TIMED	All tests that test events
FR-GAME-TIME	Time will increment automatically while player is not completing interactions	UR-TIMED	Manual
FR-DEVICE	An attempt to play the game on a system other than a Windows Desktop PC or laptop shall result in an error and the game being unavailable to play	UR-DEVICE	CHANGED - The game now runs on Windows, Mac and Linux (Manual)

FR-GAME-PLAY-SLEEP	The player shall interact with one sleep location	UR-INTERACT	canGoToSleep
FR-GAME-PLAY-STUDY	The player shall interact with one study location and make choices at this location	UR-INTERACT	canChooseToCatchUpInStudies canChooseNotToCatchUpInStudies canStudyAtCompSciBuilding
FR-GAME-PLAY-EAT	The player shall interact with one eating location and make choices at this location	UR-INTERACT	canEatAtRonCookeHub
FR-GAME-PLAY-RECREATIONAL	The player shall interact with up to three leisure location and make choices at this location	UR-INTERACT	canGoFishing ,canFeedDucks, canMeetFriendsAtPiazzaAndTalkAboutCats

canClassifyAsBestFisherWhenEnoughFishHaveBeenCaught canClassifyAsDuckDuckGoWhenEnoughDucksHaveBeenFed |
FR-SLEEP-ENERGY	The player shall be locked out of all activities other than sleeping if they run out of energy	UR-INTERACT	attemptActivityTooTired
FR-MENU	The menu shall provide the player with a list of options	UR-MENU	Manual
FR-COUNTER	The amount of each activity performed shall be counted.	UR-INFO	canClassifyAsBookWormWhenEnoughHoursHaveBeenStudied cannotClassifyAsBookWormWhenNotEnoughHoursHaveBeenStudied
FR_INTRODUCTION	The user is introduced to the game upon starting	UR-RECEIVE-FEEDBACK	correctTutorialString
FR-RANDOM-EVENTS	When performing an interaction, a random event with a positive/negative effect can occur	UR-INTERACT	piazzaRandomEnergy
FR-RESOLUTION	The game should be in 1080p	UR-DEVICE	correctResolution
FR-NPCS	The game has NPCs that can be	UR-INTERACT	canInteractWithNPC1 canInteractWithNPC2

	interacted with to make the game more lively		
FR-POINTS-REWARD	Performing activities in an order indicative of a healthy work life balance (e.g. eating three times a day) will reward the player with the most points	UR-POINTS-SYSTEM	eatingGivesBonusScore recGivesBonusScore
FR-ACHIEVEMENTS	Meeting hidden criteria (e.g. studying at least 8 times a week) will unlock achievements and reward the player with more points	UR-ACHIEVEMENTS	canClassifyAsBookWormWhenEnoughHoursHaveBeenStudied
FR_POSTGAME_LEADERBOARD	At the end of the game, a leaderboard will be displayed with the 10 players who have achieved the highest scores. The player will be added to the leaderboard if they are among the top 10.	UR-LEADERBOARD	leaderboardFileExists tenOrLessEntries addToLeaderboardLowScore addToLeaderboardHighScore

Non-Functional System Requirements

ID	Description	User Requirements	Fit Criteria
NFR-ARCH-DOCUMENTATION	The game shall be accompanied by detailed architecture documentation	UR-DESIGN	6 pages of architecture documentation containing diagrammatic representations and justifications shall be produced
NFR-CODE-DOCUMENTATION	The game code shall be commented and documented	UR-DESIGN	>95% of code should either be self-explanatory or well-documented
NFR-RESILIENCE-LOCATIONS	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-GAMEPLAY	The game shall be playable by users with no prior experience of it	UR-INTERACT	>95% of users will find the game easy to understand even if they previously played for 0 hours
NFR-OPERABILITY-SETUP	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-VISUAL	All game items 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-ERRORS	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-STARTUP	The game shall be reliable and start as expected without being unavailable	UR-WORLD	>98% of game starts will be successful
NFR-USABILITY-INSTRUCTIONS	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-CONTEXT	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	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-CURRENT-TEAM	Team members not involved in implementation shall understand what is happening in the code	UR-DESIGN	All team members will be able to understand the code within 1 hour
NFR-MAINTAINABILITY-NEW-TEAM	A new team shall be able to understand and change the code	UR-DESIGN	>90% of comments and code documentation must be understandable to new teams immediately on reading
NFR-USABILITY-PRESENTATION	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

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