

SEPR 2019/20 Assessment 1

Team CheatCodez

**Jonathan Grout, George Lesbirel, Cheuk Wang
Wu, Lauren Quarshie, Muaz Atif, Lillian Coultas**

Requirements

Introduction - changes as part of assessment 2 are highlighted in blue

In order to gain domain knowledge about the game and its requirements, the two main methods of requirements elicitation used were brainstorming sessions and interviews [1] with the stakeholders.

A whole group brainstorming session took place before the interview with the stakeholders, in which our primary objective was to prepare an agenda of relatively open-ended questions. At the end of the session, a document consisting of questions about requirements given in the product brief, as well as potential new requirements, was produced. An interview was then conducted with Dimitris Kolovos, our stakeholder and liaison for communication with the York Communications Office, in which we recorded the responses given to the questions and assumptions we had produced.

After the interview a second brainstorming session was held in which the requirements that were left up to the development team were discussed. The main objective of this session was to decide which specifications would be possible to implement within the timeframe and on budget, while also producing a game which best represented the specifications given by product brief and our stakeholders (i.e. the most impressive finished product, and the most engaging game.)

Single Statement of User Need - "To design and implement, cost effectively and on budget, an engaging and age suitable game that accurately represents the specifications discussed with stakeholders"

In accordance with IEEE standard 729 (IEEE standard glossary of software engineering terminology), a requirement is defined as :

- (1) *"A condition or capability needed by a user to solve a problem or achieve an objective"*
- (2) *"A condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification or other formally imposed documents"*
- (3) *"A documented representation of a condition or capability as in 1 and 2" [2]*

Using these definitions as a guide, we have chosen to present the requirements in the form of three linked tables which are sorted into user requirements, functional requirements, and non- functional requirements. This method of presenting the requirements allows relationships between requirements to be recognised easily by both the development team and the stakeholders. Each requirement can be identified with a unique meaningful ID to ensure that the requirements can be referred to easily in different contexts. Each user requirement has a priority attached to it:

- Shall: the finished product will conform to this requirement.
- May: the product might conform to the product, as time and other factors allow.

The non-functional requirements have fit criteria attached to them, which can be used to quantify the intentions of the stakeholder, and the success of the product.

Requirements are presented in three linked tables, with each identified by a unique ID (the format of which is shown for each table).

List of Requirements

User Requirements Table - UR_NameOfUserRequirement

ID	Description	Priority
UR_USER_EXPERIENCE	The system shall offer a pleasant user experience	Shall
UR_ENGAGING	The system shall be engaging for the user, for the entirety of the gameplay.	Shall
UR_SUITABLE	The system shall be suitable for use at open days by the target users of prospective students. The game itself will follow the guidelines of a PEGI 3 rated video game; the game will contain a very mild form of violence, no profanity and no frightening sounds.	Shall
UR_EASY	The system shall be easy for players to pick up with no instructions necessary	Shall
UR_ACCESSIBLE	The system shall be made accessible to those with cognitive, visual, and hearing impairments.	Should
UR_GAME_LENGTH	The duration of the game will be no more than 15 minutes.	Shall
UR_MOBILE_ADAPTABILITY	The game should be able to be configured to run on mobile.	May
UR_ENGINE_CONTROL	The game should have fire engines that are controlled by the player	Shall
UR_GAME_OBJECTIVE	The objective of the game will be to eliminate ET fortresses and ET patrols before the player's fire trucks are destroyed.	Shall
UR_REPAIR	The system shall allow fire engines to repair at the fire stations.	Shall
UR_REFILL	The system shall allow fire engines to refill at the fire stations.	Shall
UR_SPECS-ENGINE	The system shall have unique specs for each engine.	Shall
UR_GAME_SETTINGS	The system should provide a range of game settings that the user can adjust to their preference	Should
UR_FORTRESS_IMPROV	The system shall allow the ET fortresses to improve	Shall

EMENT	over time, and become more difficult to flood.	
UR_SPECS-FORTRESS	The system shall have unique specs for each fortress.	Shall
UR_WEAPONS	The ET patrols and fortresses will be destroyed using water based weapons	Shall
UR_COMPETITIVE	The system should provide a means of measuring competition between users.	May
UR_GAME_CONTROL	The user will be able to control the fire engines via a keyboard, mouse, or a combination of both	Shall
UR_ENGINE_NO	The system should have at least 4 fire engines in play.	Should
UR_FORTRESS_NO	The system should have at least 6 ET fortresses based on locations in York	Should
UR_GAME_WIN	The player shall win once all fortresses have been flooded, and the game shall end.	Shall
UR_GAME_LOSS	The player shall lose once all the fire engines have been destroyed and the game shall end.	Shall
UR_MINI_GAME	The system should provide an embedded mini game for the user, which shall be in a different style to the main game but conforms to the same theme.	Should
UR_DESTROY_STATION	After a fixed amount of time following the first attack to an ET fortress, the system shall cause ETs to figure out where the Fire Engines are coming from and destroy the Fire Station	Shall
UR_NO_REPAIRS	The system shall not allow fire engines to be repaired or refilled after the fire station has been destroyed	Shall
UR_ET_PATROLS	There will be ET patrols who will attack fire engines on sight	Shall
UR_DIFFICULTIES	The system may provide the user with a range of different game difficulties.	May
UR_ENGINE_WEAPONS	The fortresses and ET patrols will be eliminated with water-based weapons	Shall
UR_ET_WEAPONS	The fire trucks will be attacked by the ET patrols and the ET fortresses	Shall

Functional Requirements Table - FR_NameOfFunctionalRequirement

ID	Descriptions & User Requirements ID
FR_ENGINE_SPECS	Unique values for the fire engines will include the volume of water it

	can carry, its maximum speed, the range of its cannon, delivery rate of its water canon, and the amount of damage it can take before it is completely destroyed. The values of these specs will be randomly assigned to each fire engine. - UR_SPECS-ENGINE
FR_REPAIR_REFILL	The system shall allow fire engines to repair and refill at the fire station ;repairing and refilling will restore each spec of the engine to its maximum value that it was originally initialised to.- UR_REPAIR, UR_REFILL
FR_FORTRESS_IMPROVEMENT	Fortress improvement will be implemented in the form of the fire engines having a decreased fire rate and the amount of damage the fortress can endure before being destroyed increasing. - UR_FORTRESS_IMPROVEMENT
FR_FORTRESS_SPECS	Unique values for each fortress will include the range of its defensive weapons, the amount of damage these weapons can deal to Fire Engines over a period of time, and the volume of water it takes to flood the fortress. The values of these specs will be randomly assigned to each fortress individually. - UR_SPECS-FORTRESS
FR_MINI_GAME	The mini game will affect some features of the main game; such as an increase or decrease in the value of an attribute of the fire engines, depending on the result of the mini game. - UR_MINI_GAME
FR_GAME_SOUND	Any sounds involved with the game will be appropriate for the intended audience of the game. - UR_SUITABLE
FR_STATION_DESTROY	The system must follow the requirement NFR_PLAYTIME which puts an upper boundary on the game's play time. In order to ensure the playtime is within the time boundaries, the system will make ETs destroy a fire station 3 minutes after a fortress attack.
FR_APPROPRIATE	The system shall have no inappropriate violence, and should be relaxing and enjoyable to play - UR_SUITABLE
FR_GAME_DIFFICULTIES	The user will be able to choose from three different difficulties (such as easy, medium, hard) from the main menu before commencing gameplay. - UR_DIFFICULTIES, UR_ENGAGING
FR_PLAYER_NAME	The system shall not require a player to log in or set up an account, just to enter a name for a leaderboard - UR_COMPETITIVE
FR_SETTINGS	The system should provide settings such as brightness control, mouse sensitivity, difficulty and sound on/off functionalities that can be adjusted by the user. - UR_GAME_SETTINGS, UR_ACCESSIBLE
FR_PATROLS	There will be two types of alien patrols (which are made up of individual aliens) - attack or defend. Attacking aliens will pursue the fire engines. Defending aliens will loiter near the fortresses and attack the fire engines that pass within a certain range. - UR_ET_PATROLS

Non-Functional Requirements Table - NFR_NameOfNonFunctionalRequirement

ID	Description	User Requirement ID(s)	Fit Criteria
NFR_PLAYTIME	The playtime of the game will be between 10-15 minutes	UR_ENGAGING	90% of the games will have a playtime of 10-15 minutes
NFR_CONTROLS	Game controls should be explained to the player before the game begins	UR_EASY	Game controls are easily understandable by first time player after explanation.
NFR_FUNCTIONS	Functions of game (e.g. showing the map) require clear instructions to inform the user of their purpose, i.e. in a user manual users can access	UR_EASY	The game functions will be available to the user within 5 seconds from when the user requests them.
NFR_SIMPLE	All user instructions or written statements should be in plain english and easily understood i.e. no abbreviations or colloquial language	UR_EASY UR_ACCESSIBLE	Eight out of ten new users should be able to understand what the written statements are trying to convey upon first read through.
NFR_READING_TIME	All user instructions or written statements should allow sufficient time for users to read through, i.e. the user dismisses the text and there's no time limit on reading	UR_ACCESSIBLE UR_EASY	All instructions will be shown until the user dismisses them; the instructions should be able to be dismissed within 3 seconds of the user action.
NFR_COLOURS	The game should not rely on colour coding alone to distinguish things, or should have a colourblind assist mode.	UR_ACCESSIBLE	95% of users who are colour blind should be able to play the game as well as a regular user.

Assumptions were made about future development - more on Appendix 1.4

Assumption	Requirement ID
------------	----------------

We assume that our users will have an age range of 16-19.	<ul style="list-style-type: none"> • UR_SUITABLE
We assume that data we fetch from online will remain the same	<ul style="list-style-type: none"> • All~
We assume users will be given sufficient time to experience the whole game.	<ul style="list-style-type: none"> • UR_ENGAGING • NFR_PLAYTIME
We assume that the code will be maintained	
We assume that the costs of development is negligible	
We assume that the input data from the computer (e.g. keyboard and mouse) will not change standard (e.g. ASCII)	<ul style="list-style-type: none"> • All~ • NFR_CONTROLS • UR_ACCESSIBLE

Risk Association - please refer to Risk1.pdf for corresponding risks

Requirement ID + exp.	Risk ID	Requirement ID + exp.	Risk ID
All If function names do not match, this could hinder maintenance and future development.	2	FR_MINI_GAME/NFR_PLAYTIME As a small part of the game, time consumed could easily be unbalanced.	11
All Productivity issues (personal/team) could affect any and all parts of the project.	10	NFR_COLOURS / ACCESSIBLE Lack of understanding towards colour schemes could lead to poor system design and implementation.	14 -> 5 -> 2
NFR_SIMPLE If we rush the development then the user manual could get delayed and then be rushed.	11	UR_ACCESSIBLE We are likely to underestimate the complexity for accessibility, hence causing unexpected delays.	4