SEPR 2019/20 Assessment 2

Team CheatCodez

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

Updates on Assessment 1 Deliverables

Updated Requirements

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”. ***Justification*** *: an SSON was added as it can act as a long term indicator of how well the requirements process is being completed. We can refer to this statement when amending or adding new requirements to measure their relevance to the project.*

A colour coding scheme has also been attached to the priorities of the requirements. Requirements which must be implemented in the system (“Shall”) have been colour coded with red, requirements that should be implemented have been assigned orange, and requirements which may or may not be included in the system has been assigned green. ***Justification*** *: The colour scheme was added with the intention of providing a distinct distribution in relation to the steps that need to be taken in the future.*

Some requirements were removed due to a vague description or because the requirement was redundant. For example : UR_PRODUCT BRIEF. ***Justification*** *: This requirement statement was redundant, as it is already insinuated that all of the requirements for the system conform to the product brief.*

After re-analysing the product brief and cross examining with the stakeholder specifications, the user requirements section has been expanded to ensure that all requirements reflect these two documents. Other user requirements were added with the intention of expanding their definition and adding specific details in the functional requirements section in order to make them testable. For example : the user requirement UR_ET_PATROLS:

UR_ET_PATROLS	There will be ET patrols who will attack fire engines	Shall

The user requirement fully detailed in the expanded corresponding functional requirement FR_PATROLS:

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**

Each functional and non-functional requirement also now explicitly references a relevant user requirement in order to facilitate traceability throughout each iteration of the requirement documents.

Updates to Methods and Plans

For assessment 2 there are additional tools we are using for development that we did not anticipate when planning the previous assessment. These include:

Program Name:	**Explanation:**	**Justification:**
Photoshop [https://www.photoshop.com/]	This program is used to design and edit images.	Most of the images used for the game, such as menus and sprites, can be designed well using this program.
Tiled [https://www.mapeditor.org/]	A map/level editor.	As our game is based on locations in York, this program is used to design what our version will look like. Maps can be exported from here into the game.
IntelliJ [https://www.jetbrains.com/idea/]	An IDE with integrated GitHub support.	This editor will be used to write code for the game. The integrated GitHub support facilitates pushing commits directly to the repository
MySQL Workbench [https://www.mysql.com/products/workbench/]	Allows for designing, generating and maintaining databases.	To ensure the database is doing what we want it to do, we can use this tool.
Swagger [https://editor.swagger.io/]	Allows for describing the structure of an API so a machine can read it.	We are using this to show how the API for the database works. This shows GET and POST methods with required parameters and authentication.
LibGDX [https://libgdx.badlogicgames.com/]	A Java game development framework which provides an API	Cross platform compatible and has lots of documentation available.
JUnit [https://junit.org/junit4]/	A framework for developer side testing	Common testing framework, lots of documentation available.
Box2D [https://box2d.org/]	A 2D physics engine	This handles the collisions and movement found within the game.
Mockito [https://site.mockito.org/]	A mocking framework that allows tests to be created with an API	Allows us to create a headless application so that we can test the mechanics of the game.

Updates to Risk Assessment

Assessment 1’s risks remained unchanged due to effective mitigation.

In terms of general amendments to our risk analysis approach, all risks have been reviewed by the relevant ownership, resulting in the review dates and statuses for each risk being updated. In order to provide a more coherent overview of our risk analysis, the risks have now been organised into categories as follows:
Scheduling/planning risks : essential time-related risks which directly affect the delivery of the project - **format : SR_RiskNumber**. **Operational/procedural risks** : risks that affect the day-to-day activities operational activities involved in the development of the system - **format : OR_RiskNumber. Technical/functional risks** : risks associated with the functionality of our software/with the performance of the software - **format : TR_RiskNumber**. **Unavoidable risks** : risks associated with the whole project that are difficult to anticipate - **format : UNR_RiskNumber.**

New risks were also added:

Scaling errors (TR_6) - Incorrect scaling of the map could result in overlap, rendering certain buttons inaccessible or causing two events to be triggered when one button is pressed. ***Justification*** *: added after experiencing various problems with scaling throughout the development of the game. The mitigation and contingency action for this risk will ensure that scaling problems encountered during future development of the game can be solved quickly and with ease.*

Corruption of files affecting development of the game (UNR_3) - This risk involves the potential corruption of files during processes such as importation. ***Justification*** *: added as the development team experienced the corruption of an important file while it was being imported. Our approach with unavoidable risks such as this is to focus on prevention strategies to significantly lower the likelihood of the risk occurring in the first place.*

Large number of entities (TR_7) - If number of entities is too great FPS will drop, therefore negatively impacting the players engagement. ***Justification*** *: added as it concerns both user satisfaction and engagement, which are both mentioned in the requirements specification (UR_USER_EXPERIENCE, UR_ENGAGING).*

Documentation of the new project is difficult to understand (UNR_4) - documentation of the new project chosen is difficult to understand or it is incomplete. ***Justification*** *: commencing work on the project of another group involves dealing with unpredictable risks that may arise, therefore we have included documentation to mitigate these.*