

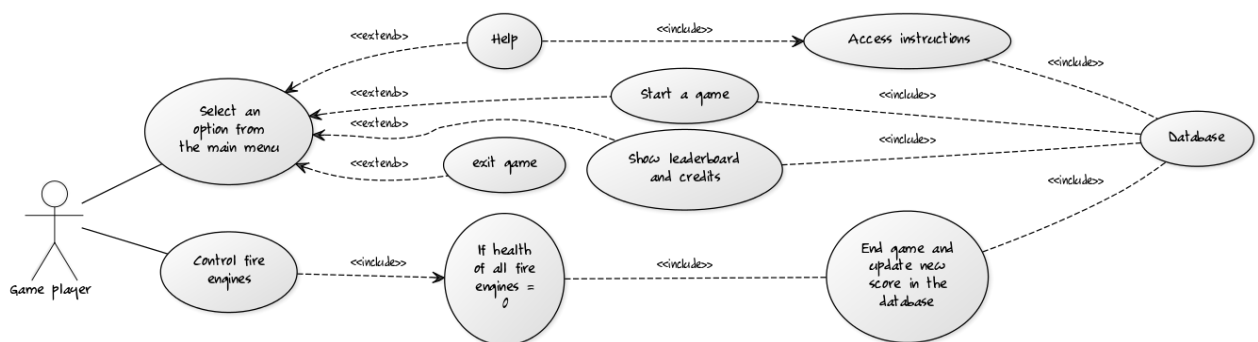
Appendix

Appendix 1 - Requirements

1.1 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”

1.2 Functional Requirements Use Case



1.3 Constraint Requirements Table

Format of IDs : CR_NameOfConstraintRequirement

ID	Description	User requirements ID	Priority
CR_STANDARDS	We shall use IEEE 29148-2018 to elicit requirements, and for guidance throughout the requirements engineering process	UR_PRODUCT_BRIEF	Shall
CR_DESKTOP	The game shall be suitable for a desktop implementation.	UR_PROUDCT_BRIEF	Shall

1.4 Different categories of assumptions

Control assumptions are for monitoring the expected control flow. Including these allows for better maintenance because there is a clear description on what needs to be called or invoked before a certain method is called.

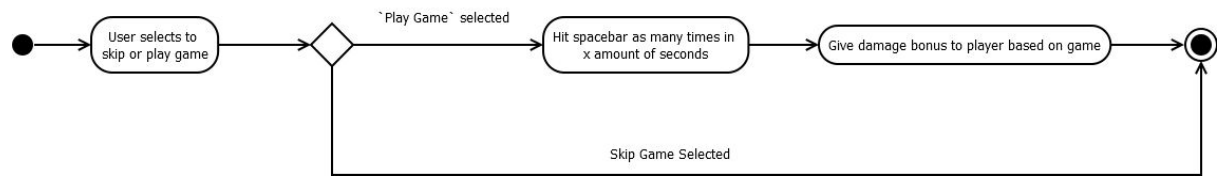
Environmental assumptions encompasses what is needed from the environment where the application will operate. For example, database variables.

Data assumptions are concerned with how the data needs to be structured prior to being processed and how it will look once it is outputted.

Convention assumptions are concerned with standards and conventions associated with the project. For website data, JSON and XML would be two examples.

Appendix 2 - Architecture

2.1 UML Diagram of the main game.



Appendix 3 Method Selection & Planning

3.1 Belbin roles <https://www.belbin.com/about/belbin-team-roles/>

3.2 Gantt chart for the software development



3.3 Plans for assessment 2, 3 and 4

Assessment 2

Task Title	Sub Task	Description
Website	Link to projects-related assessment 2 deliverables (along with assessment 1 deliverables)	There should be a page on the website that contains all final deliverables from Assessment 1
	Link to executable to game so far	There should be a downloads page that links to all downloads for the game, including all operating systems.
	User manual	The user manual should be available to download (PDF) and view online.
	Executable test plan and testing results	A testing and analytics place should be included. This will contain performance reports and unit test reports. There will also be a detailed plan of how we tested each part of the game.
Architecture Report	Concrete architecture showing the structure of the code	We should include a detailed outline of the structure of the code, justification of languages used for different elements of the project and any pieces of software used to create the concrete architecture representation
	Systematic justification for the concrete architecture	There must be a justification of the concrete architecture, clearly stating how it represents the abstract model (UML) from assessment 1. It should also mention any changes that needed to be made from the abstract model. The concrete architecture must link to the requirements laid out in assessment 1.
Implementation	Provide documented code for working implementation	We must provide well documented code for the working version of the game that meets all original outlined requirements and the concrete architecture. The code should be made available via a link to download or a repository. It should be clearly labelled with a date prior to the hand in. All dependencies must be included. Consistent use of names and constructs should be used to make it easy to trace.
	State explicit features	We should state explicitly any of the features planned to be included that were not fully implemented. Consistent use of names and constructs should be used to make it easy to trace.
Software Testing Report	Summary of testing methods and approaches	We must include testing reports and how we approached testing our software. Explaining why these methods and approaches are suitable.
	Report of actual tests including stats	We should include a report of the tests that were made, including what the tests were and what

		results they achieved. Any failed tests, should be accompanied by a brief statement explaining why these are failing and comment on what is needed to make them pass. There should also be a comment on the overall success, completeness and correctness of the tests made
	Provide precise urls for testing material on the website	There should be a URL which leads to the testing materials. This will include the testing design and evidence.
Updates on Assessment 1	Updated requirements	The website should include an updated statement of requirements, showing what has been changed, added and removed. The deliverable should include an explanation of the changes made and include the url of the update statement.
	Methods and plan update	The website should include an updated plan, showing any remaining tasks that need to be completed and detailed tasks for assessment 3. There should also be a statement on the changes. The deliverable should include a summary of any changes made to the methods and tools, with a short explanation for each. This should be accompanied by a url for the updated section and any relevant methods.
	Risk assessment update	The website should include updated risks and the respective mitigation strategy. If the approach to risk has changed since assessment 1, this must be clear. The web pages must be updated and a new link should be provided so a comparison can be made. The deliverable should include an explanation for any changes made for the whole project, each element must be justified. If nothing changes, explain why the methods are still suitable.

Assessment 3

Task Title	Sub Task	Description
Website	All the project-related Assessment 3 deliverables, as well as the Assessment 1 and 2 versions	The deliverables should be uploaded to the website, available to view and download, alongside the documents for 1 and 2.
	The website must contain the download link to the executable for the game;	The website must contain the download link to the executable for the game
	Must contain test plan and testing results	There must be a web page displaying our, updated, testing report and test plan. This should include statistics and a clear representation of the results.
	Link/Display of user manual	There should be an editable source user manual and a PDF that can be viewed online and downloaded.

Change report	Briefly summarise the team's formal approach(es)	We should outline any changes that have happened in the deliverables, documentation, code and overall changes to management. This should be clearly presented and written about.
	The testing report: methods and approaches, materials or presentation of tests and testing statistics	There should be a clear explanation and justification of changes made. There should be urls of the pages where the updated material can be found. If there are no significant changes, we must comment on why this is so. The changes to the test report must contain a clearly present the tests run and the statistics. The methods and planning section must show any changes made to development methods/tools, management and a plan for assessment 4 (the updated plan must be linked to on the site)
Implementation & Report	Provider documented code for a working implementation of the game	We must provide well documented code the the latest implementation of the game that meets the requirements, remit and the concrete architecture for assessment 3. We should highlight the new pieces of code. This can be submitted via a zip file or link to repo on the site.
	Explain how the code implements the architecture and requirements outlined for Assessment 3.	We must explain how the code implements the architecture and requirements we had planned out. We must cover any new features, such as data types, structures and algorithms. A systematic report explaining each change is also required. These explanations must link to the requirements and architecture. There should be a clear statement on what is now left for assessment 3 that aren't complete.

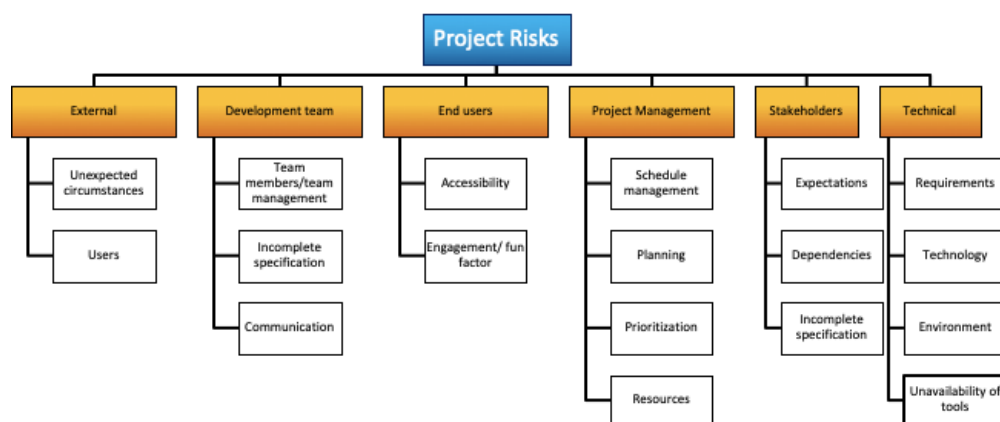
Assessment 4

Task Title	Sub Task	Description
Environment and Testing Report	Explain and justify the approach that the team took for evaluating and testing the final product	We must explain and justify the teams approach to evaluating and testing the final product. We must clearly state how well we met the brief and requirements. We must comment on code quality, state and justify what is the appropriate quality. There must be a clear note of what has changed the testing approach or materials. We should include urls for any relevant material on the website (e.g. test plans, testing materials, test statistics, previous versions of testing, previous justifications, etc.).
	Comment on how the product meets/does not meet different	Comment on how the product meets/does not meet different requirements. Include links to where requirements can be found

	requirements. Include links to where requirements can be found	
Implementation and report	Provide documented code for a working implementation of the game that meets the remit and requirements	We should provide documented code for the latest working implementation of the game that meets the requirements for assessment 3. This should also include the concrete architecture and changes introduced in assessment 4. Comments should highlight new or extended sections. This should be uploaded to the site/linked to on the site as a new version so a comparison can be made.
	Summarise how the software (code and GUI) was <i>modified</i> to incorporate required changes, and any other changes	There should be a clear summary of how the code and gui was modified to suit any new changes. Each change should be related to a revised final requirements and architecture, and clearly justifying each change. Explain how and why we made changes to pieces of the software and GUI that we inherited. We should also explain and justify and extra features that have been included in the software.
Project review report	Summarise, with appropriate citation of literature and online sources, your teams approach to team management and the structure at the end of the project.	We should briefly summarise, with the correct citation and resources, our teams approach to team management and structure by the end of the project. Commenting on how the team layout changed. We should make reference to the changing needs are risks of the project and our understanding of the members in our team.
	Briefly summarise, with appropriate citations of literature and online sources, the software engineering development methods and tools that your team chose to use.	We should briefly summarise the SE development methods and tools that we chose to use. We should provide citations and online sources that were used. We should make reference to the changing needs and risks of the project.

Appendix 4 - Risk Assessment and Mitigation

4.1 Risk Breakdown Structure (RiBS)



4.2 Definition of Risk

“Risk, the probability that an asset will suffer an event of a given negative impact, is determined from various factors: the ease of executing an attack, the attacker’s motivation and resources, a system’s existing vulnerabilities, and the cost or impact in a particular business context.” [3]

4.3 Risk Rating Bands

LOW (1-3)	MODERATE (4-7)	HIGH (8-14)	URGENT (15-25)

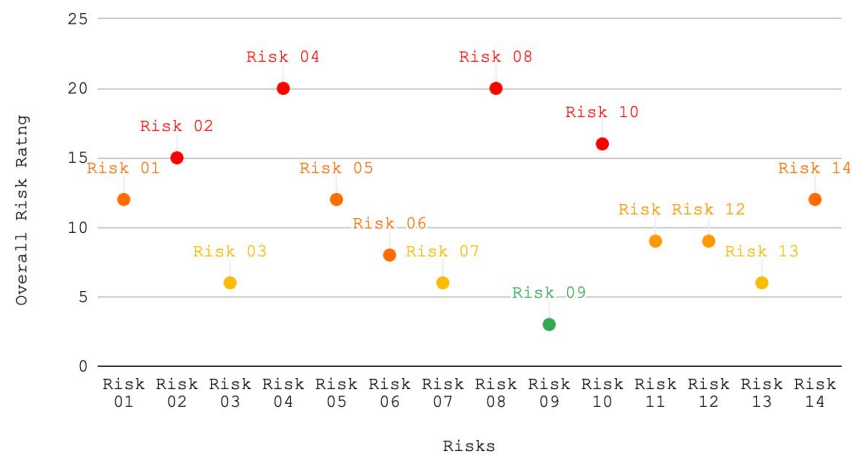
4.4 Risk Matrix Keys

Risk Severity	Description
(1) Negligible	<ul style="list-style-type: none">Minimal or no impact on achieving outcome objectives.Objectives will reach its set expectations.
(2) Minor	<ul style="list-style-type: none">Minor impact on achieving desired results.Some objectives might not reach its set goal, but will remain above acceptable standards.
(3) Moderate	<ul style="list-style-type: none">Moderate impact on achieving desired results.Some objectives might not reach its set goal, but will remain at acceptable standards.
(4) Significant	<ul style="list-style-type: none">Significant impact on achieving desired results.Some objectives might fall below acceptable levels.
(5) Critical	<ul style="list-style-type: none">Severe impact on achieving desired results.Some objectives might not be achieved.

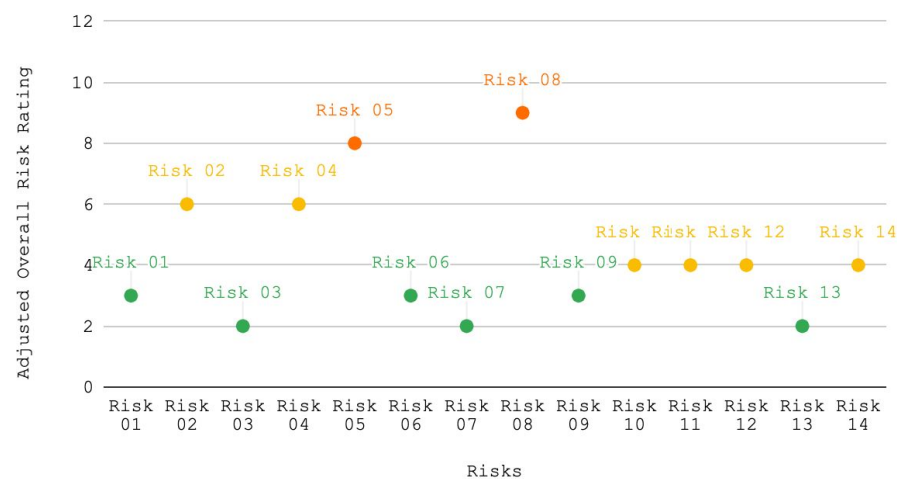
Likelihood of Occurrence	Description
(1) Rare (0-20%)	<ul style="list-style-type: none">The occurrence is close to never.
(2) Unlikely (21-40%)	<ul style="list-style-type: none">It will occur rarely.
(3) Possible (41-60%)	<ul style="list-style-type: none">The occurrence is uncommon.
(4) Likely (61-80%)	<ul style="list-style-type: none">It will occur from time to time, however not frequently.
(5) Very Likely (81-100%)	<ul style="list-style-type: none">It will occur frequently.

4.5 Risk Analysis Visual Aids

Overall Risk Ratings



Overall Risk Ratings After Adjustment



4.6 Risk Matrix

RISK MATRIX	1 Rare	2 Unlikely	3 Possible	4 Likely	5 Very Likely
5 Critical	5 Moderate	10 High	15 Urgent	20 Urgent	25 Urgent
4 Significant	4 Moderate	8 High	12 High	16 Urgent	20 Urgent
3 Moderate	3 Low	6 Moderate	9 High	12 High	15 Urgent
2 Minor	2 Low	4 Moderate	6 Moderate	8 High	10 High
1 Negligible	1 Low	2 Low	3 Low	4 Moderate	5 Moderate