Requirements

Requirements documentation from Assessment 1:
https://engteam14.github.io/website2/pdfs/Req1.pdf

A) Elicitation of requirements

1. The provided product brief indicated the overall goals and intentions of the finished product and contained general descriptions of its desired functionality
2. Group brainstorming session was held to compare our interpretations of the brief and to raise a list of questions to address to the customer
3. In a group meeting with the customer, answers to the questions and any other customer comments were detailed in informal meeting notes.
4. Recorded info was formalised as a set of user requirements.
5. User requirements were distilled down into more specific functional and non-functional requirements.
    1. Functional requirements detailed concrete, specific functionality and capabilities of the product as related to its software implementation.
    2. Non-functional requirements captured the performative characteristics of the completed product as a whole which could be perceived by the user or tester.
6. Resulting functional and non-functional requirements were closely evaluated for possible risks to their implementation; these were detailed within the risk register.

Research into requirement specification and presentation

- IEEE requirements engineering document[1]:
    - Provided comprehensive information on all aspects of requirement elicitation and presentation, although sections 5.1-5.2.8 and 6.1-6.6.3 were most helpful
    - Contained robust justification for the need for requirements and their role in the overall software lifecycle process
    - Informed our choice of specific language, priority language(low/medium/high), the choice of imperative tone, and justifications for these
    - Ultimately, aimed at larger, more critical projects than ours
- ENG1 lecture on requirements engineering:
    - Provided an excellent overview of the motivations for requirements engineering and a lucid overview of this process
    - Introduced the user/functional/non-functional requirements methodology which proved an excellent fit for our scope of project (versus lifecycle-based requirement methodology in the IEEE document)
    - Demonstrated requirements tables as a tool for writing down and detailing requirements

Informed by these resources, we chose a tabular format for implementing for the requirements register, allowing us to easily add additional metadata to individual rows as extra columns and permitting a quick, comprehensive overview of the entire register. All rows were labeled with unique identifiers, permitting cross-referencing between user and

(non-)functional requirements, as well as with other sections of project documentation, such as the risk register and architecture specs.

[1] *Systems and software engineering -- Life cycle processes -- Requirements engineering*, ISO/IEC/IEEE 29148:2018(E), 2018.

B) Requirements Tables

ID	Description	Priority
UR_PLATFORM	The user shall use a standard laptop PC to play the game	High
UR_GAME_INIT	The user shall begin a new game from an initial state	High
UR_SHIP_CONTROL	The user shall control a ship sailing across the great Lake of York	High
UR_COMPETING_COLLEGES	The user shall encounter at least 3 other colleges	Medium
UR_LEARNING_CURVE	The user shall play the game without training	High
UR_GAME_DURATION	The user shall be able to complete the game within a ~5 minute timespan	High
UR_GAME_OBSERVABILITY	The game shall accomodate onlookers in the PCs surroundings	Low
UR_FRIENDLY_SHIP_ENCOUNTER	The user shall encounter friendly NPC ships	Low
UR_HOSTILE_SHIP_ENCOUNTER	The user shall encounter hostile NPC ships	Low
UR_FIRE_WEAPONS	The user shall be able to fire weapons from the ship	High
UR_BULLET_DODGE	The user shall be able to maneuver their ship to dodge fired munitions	Medium
UR_HOSTILE_BUILDING_COMBAT	The user shall engage in combat with hostile buildings	Medium
UR_HOSTILE_COLLEGE_CAPTURE	The user shall capture other colleges via combat	High
UR_EARN_PLUNDER	The user shall earn plunder	High
UR_EARN_XP	The user shall earn XP	High

ID	Description	Priority
UR_QUEST_PROGRESS	The user shall progress through a series of quests	Medium
UR_GAME_WIN	The user shall win the game through achieving an ultimate objective unlocked by the fulfilment of preceding requirements/quests	Medium
UR_GAME_LOSE	The user shall lose the game through being defeated in combat	High
UR_SHIP_COMBAT	The user should be able to engage in combat with other ships	High
UR_OBSTACLE_ENCOUNTER	The user may encounter obstacles while sailing in game	High
UR_WEATHER_ENCOUNTER	The user may encounter bad weather while sailing	High
UR_SPEND_PLUNDER	The user should be able spend the plunder earned	High
UR_POWER_UP	The user should be able to obtain power ups through either the shop or at random locations on the map.	High
UR_DFCLTY_LVL	The user should be able to choose from 3 difficulty levels (e.g. easy, normal, hard)	High
UR_GAME_SAVE	The user should be able to save the state of the game at any time and be able to resume it at a later point.	High

Functional Requirements

ID	Description	User requirement	Risks	Priority
FR_MENU_KB_INPUT	The game shall accept keyboard input for menu navigation	UR_PLATFORM		Low
FR_VIEWPORT_SCALING	The game shall render on a 13"-27" monitor	UR_PLATFORM	R5	High
FR_MIN_FPS	The game shall render at a minimum of 30 FPS	UR_PLATFORM	R6	High
FR_CROSS_PLATFORM_WIN	The game shall be playable on Windows	UR_PLATFORM		High
FR_CROSS_PLATFORM_MAC	The game shall be playable on Mac OS	UR_PLATFORM		Low
FR_CROSS_PLATFORM_GNU_LINUX	The game shall be playable on GNU/Linux	UR_PLATFORM		Low
FR_GAME_RESET	The game shall allow restarting play from an initial configuration	UR_GAME_INIT		High
FR_SHIP_KB_INPUT	The game shall accept keyboard input for ship control	UR_SHIP_CONTROL		High
FR_COLLEGE_ENTITY_TRACKING	The game shall keep track of ships and buildings for a minimum of 3 distinct factions	UR_COMPETING_COLLEGES		High
FR_FRIENDLY_AI	The game shall control the actions of friendly ships	UR_FRIENDLY_SHIP_ENCOUNTER		Low
FR_HOSTILE_AI	The game shall control the actions of enemy ships	UR_HOSTILE_SHIP_ENCOUNTER	R1, R3	Medium
FR_PLAYER_FIRE	The game shall enable the user to fire ship weapons	UR_FIRE_WEAPONS		High
FR_PLAYER_HEALTH	The game shall mantain the state of the users' health points	UR_SHIP_COMBAT		Medium
FR_PLAYER_AMMO	The game shall maintain the state of the user's ship armament and ammunition	UR_FIRE_WEAPONS		High

ID	Description	User requirement	Risks	Priority
FR_BULLET_TRAVEL	The game shall render the travel of a ship's fired munition	UR_BULLET_DODGE	R2	High
FR_PLUNDER_TRACKING	The game shall keep track of a player's plunder	UR_EARN_PLUNDER, UR_SPEND_PLUNDER		High
FR_PLUNDER_UPDATE	The game shall reward plunder on success in quests and encounters	UR_EARN_PLUNDER		High
FR_XP_UPDATE	The game shall give XP with time survived and obstacles navigated	UR_EARN_XP		Medium
FR_XP_TRACKING	The game shall keep track of a player's XP	UR_EARN_XP		High
FR_XP_UPDATE	The game shall give XP on successful combat encounters completed	UR_EARN_XP		Medium
FR_QUEST_TRACKING	The game shall maintain the state of the user's progress through multiple objectives	UR_QUEST_PROGRESS		High
FR_QUEST_RANDOMISE	The game shall randomise user's objectives between different playthroughs	UR_QUEST_PROGRESS		Medium
FR_QUEST_OBJECTIVE	The game shall associate quest objectives with game entities	UR_QUEST_PROGRESS		Medium
FR_PLAYER_DEFEAT	The game shall display game stats upon player defeat	UR_GAME_LOSE		Low
FR_SCENARIO_FAIL	The game shall display game stats upon game over scenario completion	UR_GAME_LOSE		Low
FR_PLAYER_DAMAGE	The game shall track damage that the player takes and adjust the displayed health accordingly	UR_HOSTILE_BUILDING_COMBAT UR_SHIP_COMBAT		Medium
FR_ENEMY_SHIP_DAMAGE	The game shall track damage dealt to enemy ships	UR_SHIP_COMBAT		Medium
FR_OBSTACLE_SHOW	The game shall render obstacles for the player to encounter	UR_OBSTACLE_ENCOUNTER		Medium
FR_OBSTACLE_HIT	The game shall deal damage to the user when the user hits an obstacle	UR_OBSTACLE_ENCOUNTER		Medium
FR_WEATHER_SHOW	The game shall render weather events for the player to encounter	UR_WEATHER_ENCOUNTER		Medium

ID	Description	User requirement	Risks	Priority
FR_WEATHER_HIT	The game shall deal damage to the user when the user enters a weather event	UR_WEATHER_ENCOUNTER		Medium
FR_PLUNDER_SPEND	The game shall allow users to spend plunder in exchange for benefits to the player	UR_SPEND_PLUNDER		Low
FR_POWER_UP	The game shall boost certain elements of the gameplay such as speed, weapons and damage when the user collects a power-up	UR_POWER_UP		Medium
FR_DFCLTY_CHANGE	The game shall allow users to change the difficulty setting before the game starts	UR_DFCLTY_LVL		Low
FR_DFCLTY_DFRNCE	The game shall adjust elements of the gameplay (speed of enemies, health, damage dealing etc.) according to the difficulty level	UR_DFCLTY_LVL Low		
FR_SAVE_GAME_STATE	The game shall save the state of the game at any point when the user clicks 'save'	UR_GAME_SAVE	R9	Medium
FR_LOAD_GAME	The game shall be able to load the game from the same point at which it was saved	UR_GAME_SAVE	R9	Medium

Non-Functional Requirements

ID	Description	User requirement	Fit criteria	Risks	Priority
NFR_SHIP_COLLISIONS	The game shall detect collisions between different ships	UR_HOSTILE_SHIP_ENCOUNTER	Distance between drawn assets <5px	R4	High
NFR_WORLD_COLLISIONS	The game shall detect collisions between ships and world objects	UR_COMPETING_COLLEGES	Distance between drawn assets <5px		High
NFR_BULLET_COLLISIONS	The game shall detect collisions between game entities and fired munitions	UR_BULLET_DODGE	Distance between drawn assets <5px		High
NFR_USER_INPUT_LAG	The game shall be responsive to user input	UR_SHIP_CONTROL	Input lag <200ms		High

ID	Description	User requirement	Fit criteria	Risks	Priority
NFR_AI_LAG	NPC actions' responsiveness shall approximate that of player actions	UR_HOSTILE_SHIP_ENCOUNTER	AI response time <200ms	R3	Low
NFR_RENDER_SMOOTHNESS	The game world shall render smoothly during player movement	UR_SHIP_CONTROL	Visual render lag <200ms	R8	Medium
NFR_COLOURBLINDNESS	Game map and assets should be distinguishable by a colourblind person	UR_PLATFORM	Subjective screenshot test via colourblind accessibility evaluation app		Medium
NFR_EASE_OF_USE	The game shall be self-explainable and feature obvious controls	UR_LEARNING_CURVE	Tester must be able to pick up and play with no prior instruction		High
NFR_GAME_DURATION	The game shall finish within ~5 mins in a win or loss for the player	UR_GAME_DURATION	Tester must reach the game stats screen within 4-6 mins		Medium
NFR_LARGE_ASSETS	The game assets shall be large enough to observe from several metre's distance away on a standard laptop PC screen	UR_GAME_OBSERVABILITY	Observer standing 2m away should be able to answer questions about gameplay state		Medium