Module	SEPR
Year	2019/20
Assessment	3
Team	Krojan Horse Expanding on NP Studios' Assessment 2
Members	Abel Kent, Toprak Onat Sen, Vinu Jey, Tom Emmerson, Diana Udrescu, Toby Popov, Jamie Todd
Deliverable	Implementation Report

Implementation report

Introduction

For this stage of the assessment we continued development of another group given the same product brief as our own. Our aim was to finish implementing the features that the original project outline specified. We selected the work of NP studios to finish.

New Features

Over the course of this stage of the project, we implemented new features to better suit the project outline as given by the client.

Tetris Minigame

As elicited in the product brief, “There should be an embedded mini-game, completely different in style from the main game, but aligned to the theme of the main game” **(UR_ minigame)**.

Within the previous requirements for NP Studios’ Assessment 2, their minigame requirement stated about including an event when refuelling fire trucks. This sounded tedious and due to the pre-existing architecture of level selecting, we decided on implementing a bonus level with a new concept as the implemented minigame within the level-select screen **(UR_select_ level)**. In order to enhance user experience we decided to incorporate the idea of Tetris as the minigame. This was done in agreement with the client who thought it was a fitting idea. The blocks in the game are made up of firetrucks and once they hit the floor they turn into alien geese; this ensures the theme of the mini game is still in line with the main game. We also included that the game can be paused to comply with **(UR_pause)**

More levels

NP’s work, contrary to our own, had its fortresses spread out over a level system, with one fortress per level, as opposed to a system of them spread around a large world. The assets we were given contained 3 levels, and we developed it to have 3 additional levels **(FR_6_levels)**, each centered around landmarks in York, as specified in the project outline provided by the client.

We also centred the maps, opposed to a screen fill that was implemented before, so as to make the information presented as part of **(UR_seeHUD)** clearer to the user. We ensured fire trucks regained full health at the start of each level **(UR_fresh_ health**). Progress is saved as well for these further levels. **(FR_auto_save)**

Patrols

One feature that was specified in the project outline that had not been implemented at this stage of the project was the need for alien “patrols” that would wander the levels and provide challenge to the player. There existed in the assets of the game sprites and behaviour for aliens who would remain in place near each level’s fortress, however due to their lack of movement we determined they did not fit the definition of a patrol. We utilised the sprites provided to create patrol entities who move between points on the map and attack the player if they encounter them.

Fortress attack patterns

We implemented fortresses attack patterns such that they now cycle between a set of predetermined patterns, thus making the game more enjoyable and easier to play as their attack patterns can be recognised and memorised by the Player. This is an improvement over the previous method of the fortresses not attacking the player, therefore being in violation of the product outline given to us by the client for each fortress having defensive weaponry, albeit not being in violation of any of the previous teams requirements, we thought it to be necessary.

Fire truck death and selection visualisation

When a fire truck is destroyed, it now leaves a skull on the map in the place of where it was at the moment of it's death, as well as being marked as dead at the bottom of the screen where the information for them is displayed to the user. This was done to better fit **(NFR_artwork)** as well as assist in the information provided to the user as part of **(UR_seeHUD).** We also marked the firetruck at the bottom that the user currently has selected in blue text for the same reason.

Changes

In addition to adding new features to their game, we had to make changes to their existing code to better fit the project outline and requirements.

Tilemap (colour)

We altered their tilemap to create our own levels and redesigned them accordingly, as their levels were designed with their own assets, but these assets were not provided to us initially when we adopted their work. We also felt that the tilemaps colour palette and style was potentially in violation of requirements **(NFR_artwork)** and **(UR_interest)** We also altered the sprites for the firetrucks and fortresses to better fit the aesthetic of the new tilemap, with fortresses and the fire station adopting a less birds eye view of the buildings to make them recognizable.

Collision map

We altered their collision such that they were no longer using their method, that involved manually adding each and every object that desired collision in individual lines of code **(UR_collisions, FR_deny_collisions)**. We altered their code so that each level has an asset of a mask, which is consulted in regards to the tiles of the map to see if the fire trucks can enter the tile. This allowed us to make easier changes to their levels and assisted in developing our own. We felt that implementing this improved the overall quality of the game and fit the requirement of **(UR_interest)** as it made the gameplay more enjoyable.

Music and sounds

We changed the music in the game to some of our own creation, as the previous music was developed by a third party and we didn't have the rights to use it in our game (**FR_play_music**). This might have been in against the project outlined, which specified the need to use it at open days for the department of computer science at the university of york,

and such we felt the need to change the music. The click sounds on menus also sounded unfitting for the game, so these also have been updated.

Diagonal movement

The movement controls have been altered to allow for diagonal movement, rather than just horizontal and vertical axis **(UR_instruct_ engines)**, to improve the user experience, as we found it abrasive, confusing and tedious to use while playing. This potentially put the game in violation of **(UR_interest)**.

Swap between fire trucks

We altered the game such that the firetruck that the player is controlling can be swapped to another by means of the numkeys on the keyboard, allowing the user to swap between them without having to click on them with the mouse. This was done to allow for easier transitions between trucks due to their small placement on screen making it difficult to isolate and click them, thus making the game difficult to play and violating **(UR_interest)**.

Fire trucks: Manual - Array

We changed the referencing of the fire trucks from a manual system to an array, this allowed us to implement features more easily with the fire trucks as we continuously developed the code. In particular it made it easier to identify when the lose condition for the game was fulfilled.**(UR_end_game_ screen)**

Shoot with mouse

We altered the shooting method of the game so that it can now be activated by means of pressing the spacebar and by pressing the left mouse button. This made the game more intuitive to the player, hence reducing the possibility of violating **(UR_interest)**

Patrol targeting

We rewrote the targeting system of the patrol units to use Euclidean geometry as opposed to conditional statements, allowing us to develop patrols to have more consistent behaviour as well as making code easier to read and edit.

Projectile disposal

Projectiles generated are now removed when the exit the bounds of the screen, preventing them from escaping to cover the UI **(UR_seeHUD)** or distract the user, which may violate **(UR_interest)**.

Errors encountered

As we developed the work by NP we encountered and corrected several errors that already existed in their work.

The collision hitbox for fire trucks encapsulated a larger area than the sprite itself, making the game confusing to play as firetrucks would take damage even when they appeared to avoid the projectiles. This may violate **(UR_interest)** as players may become frustrated at their seemingly meaningless damage.

The sprites used for the bullets in the game were measured at 500x500 pixels each, just then compressed to fit on the screen. This caused the game to use up more memory than necessary and in situations with many bullets on screen the game would cause the FPS of the game to drop, once again potentially frustrating the player and violating **(UR_interest)**.

Certain buttons clicked in the menu screen would cause the game to crash as the relevant functions were not present in the memory, such as pressing the “Back” button on the options menu, and then clicking the same spot on the next page would crash the game. This posed a major threat to the ability of the player to play the game.

Bullets generated by the player would not be removed upon having dealt damage to the enemy fortress, and so would continue to travel through them, continuously dealing damage each frame they travel through a fortress. This meant that fortresses health would drain rapidly and inconsistently, especially between varying frame rates, leading to easy levels, violating **(UR_interest)** as the player may have felt unchallenged.

Errors created

Whilst developing our implementation, we created, before subsequently fixed, errors as we made changes to the code. Some of the more significant are summarised below.

As we have changed the referencing of the fire trucks, it led to the error that when the firetruck the player had selected was destroyed, if the truck was the 0th in the array, the would automatically fail the level, even if the user had other available trucks.

The implementation of bullets for the attack method of fortresses led to a number of errors in implementing them with the rest of the game, notably the fact that the bullets can escape the confines of the game window and potentially obscure the HUD from the player. We considered this a major issue as it was in violation of **(UR_seeHUD)**. This has since been fixed.

As we added the ability to swap between firetrucks using numpad keys, it gave rise to the error that when the player does click on a firetruck to switch between them, both of these firetrucks will briefly fire. This occurs as each firetruck simultaneously detects the click as the command to fire, as well as swapping vehicles. This is due to an overlap in keybinds and from time constraints over the short development period, is the only remaining known bug, but has little impact on gameplay and only occurs as a secondary keybind for firetruck swapping to number key swapping.

Unimplemented elements

As it stands, for this stage of the assessment we have implemented all major features as specified by the product brief.