Implementation Report

SEPRet Studios

Anthony Nel, Ben Silverman, Jake Billington, Leah Moss and Shouyi Yuan

Inherited Software Modifications:

The inherited software had the same base game as we used for Assessment 3, as such the minigame and 3 fortress assets were designed differently to our previous modifications. We decided to modify the fortress assets due to the fact the design and positioning could cause confusion to the player and impact intuitive playing of the game, in addition to simply not staying within style with the three pre-existing fortress assets.

Before the game map loads an information screen is displayed providing some instructions to the user [UR_CONTROLS]. This intermediary information screen is closed and the game is loaded by clicking a cross icon in the upper right corner of this screen. The graphical cross icon was not mapped precisely to the area of the screen that when clicked closes the screen. This was a preexisting bug that we overcame in our assessment 3 modifications of the template game, however this inherited game still contained this issue.

The patrol assets that the inherited software used, contained shadows within the asset design. Given that the patrols follow paths that are not limited to the paths on the map, and patrols may travel over buildings and water so that the shadow looks out of place when the assets move off of the paths. The decision to remove the shadows from the asset was made in regards to maintaining consistency among assets and also uphold satisfying graphics. Patrols have been given a hovering movement implemented by varying the movement in the axis opposite to the axis of propagation by some value proportional to the speed of movement of that patrol.

As before in assessment 3, and also following “Team 3”, who we inherited the software from, we decided to add our team’s logo to the splash screen sequence. We decided this is an important and necessary change inorder to show the groups of teams that have contributed to this iteration of the development of KROY.

The minigame in the inherited software consists of a “dance battle” where the user presses an arrow key on the keyboard in response to the corresponding arrow icon reaching a red box on the screen. The margin for error was decided to be too small after trialling the game, so the allowable margin for the icon to be from the red box was expanded so that the minigame is not as difficult. The minigame simplification was introduced to maintain the time constraints of the game. If the minigame remained as difficult it was difficult to proceed and finally finish the game, particularly in the 15 minute time suggestion within the requirements [UR_TIME], the fact that the user needs little practice to be passable at the minigame also satisfies the requirement for the user being able to learn the controls quickly [NFR_MINIGAME_CONTROLS].

In addition to adjusting assets and fine-tuning existing aspects of the game, we have also included a range of additional aspects. The required updates included the change in difficulty [UR_DIFFICULTY]. This change allows the user to select the difficulty of the game upon loading the game and on the title screen of the game the user will be able to choose a difficulty level, upon which the game object specifications will change accordingly, altering the relative difficulty of the game. All other requirements are maintained regardless of difficulty level, including the time constraints, that despite difficulty potentially increasing the game will be concluded at approximately 15 minutes [UR_TIME].

A completely new addition is the inclusion of a save and load game feature that allows the game state to be saved at any point during game main gameplay and loaded at a later date to the same point [UR_SAVE_GAME]. This load/save addition greatly adds to the user experience [UR_UX], given the relative shortness of the game a user may be able to

complete the entirety of the game in a single sitting, however if the user does not finish the game in its entirety and is forced to stop without saving, the user may not wish to revisit the game at all if they are unable to save, however if saving is possible the user may be more inclined to replay the game, and restart the game multiple times, this is relative to the user of course but allowing this mechanism opens more options for the user and allows for an increase in possible experience.

Power-Ups [UR_POWER_UPS] are also added which take the form of icons in paths on the map, that when a fire-engine collides with a power-up the power-up type collided with, will have a particular effect on that fire-engine. The stated power-up types include temporary immunity for 5 seconds, speed boost for 5 seconds, damage boost for 5 seconds, clones a fire engine for a brief period and instantly repair and refill the current fire truck. The inclusion of the power-ups adds an additional level of activity to the game. By including a range of power-ups allows the user to be able to play the game with multiple strategies, aiming for a particular or multiple power-ups to assist the user at various points in the game, such as attacking a particular fortress. This allows for a heightened user experience [UR_UX] and as such it may result in a user returning to play the game multiple times as the experience can change over multiple games. The power-ups do not add or reduce difficulty, [UR_DIFFICULTY], or alter the approximate time taken [UR_TIME] to complete the game, but do add to the experience a user can have.