



LAMPLIGHT: DESIGN DOCUMENTATION

600090: Commercial Games Development ACW 2



By:

Damon Moore

Stephan Zaneboni

Eleanor Woodhouse

Olumide Osikomaiya

Efren Krauss

Contents

Game Design and Aesthetical Decisions	4
1. Initial Design Concept	4
2. Character Design	4
a. Hunters.....	4
b. Ghosts	4
3. Game Mechanics.....	4
a. Perspective and Control Systems.....	4
b. Level Environment and Interaction.....	5
c. Lamp Capturing.....	5
4. Aesthetics.....	5
a. Art Style and Music	5
b. UI	6
Roles and Tasks	7
1. Game Design (Olumide/Eleanor/Efren/Damon).....	7
2. UI Design (Stephan, Damon)	7
3. Lighting (Damon).....	8
4. 3D Modelling and Animation (Efren)	8
5. Report (All)	9
Project Management Evaluation	10
References	11
Appendices.....	12
1. Instructions Manual	12
2. Marketing Plan	13
3. Ethical and Legal Considerations	14
4. Time Plan.....	15
5. Time Sheets.....	16
6. Individual Statements, Contributions and Portfolio	18
a. Damon.....	18
b. Efren.....	18
c. Eleanor	18
d. Olumide.....	18
e. Stephan	18
7. UML Diagrams and Concept Sketches	19
c. Concept Sketches (Figures 7.A).....	19
d. Use Case Diagrams (Figures 7.B).....	23

e.	Class Diagrams (Figures 7.C)	25
f.	Activity Diagrams (Figures 7.D).....	26
8.	Additional Items (Including Game Screenshots).....	29

Game Design and Aesthetical Decisions

1. Initial Design Concept

'*Lamplight*' is designed as a team-based, local-multiplayer casual party game set in a haunted building, where players are divided into two teams, *Hunters* and *Ghosts*. The *Ghosts* inhabit the building and have the goal to prevent all the lamps in the rooms from being lit without being seen and eliminated by the hunters. Whereas the goal of the *Hunters* is to keep the lamps ignited, whilst also chasing and eliminating the *Ghosts*. The *Hunters* can win by either eliminating all the *ghosts* or by lighting all the lamps. The *Ghosts* can only win if all the light sources within the game have been extinguished.

2. Character Design

a. Hunters

Hunters have been designed to be the main antagonist of the game, featuring a lamp to light the light sources with. The aim of the hunters within the game is to eliminate ghosts whilst also ensuring that all light sources are lit throughout the building. In context of capturing the ghosts, this will be achievable through the hunter pressing the 'Interact' button on the controller, launching a net. This net will be launched from the hunter player model in the same direction that they are facing. Once colliding with a ghost player, it will lock them in place and cause a capturable point to spawn on top of them, which fellow ghost players can capture and free their teammates. In terms of victory conditions, the hunter players will achieve victory when all light sources have been found around the map and lit, causing a scene transition to indicate that their team has won (Figure 7.B.1).

b. Ghosts

Ghosts have been designed to be almost completely invisible to the hunters and must navigate around the map through physical interaction with the game world. Ghosts will have the ability to interact and collide with 'Small' Objects such as books or small tables in order to create a distraction for the hunters. In addition, ghosts will also feature the ability to 'Boo' other hunters, causing temporary disability of their movement, allowing ghosts to move freely without risk for a short period of time. Friendly ghosts may also free their teammates through standing near a netted ghost, gradually filling a progress bar identically to capturing a light source; which in turn will gradually decrease should they move away. The main objective for ghost players is to extinguish all light sources around the map through standing within the lights area of influence, decreasing it gradually until the progress bar reaches '0' (Figure 7.B.2).

3. Game Mechanics

a. Perspective and Control Systems

'*Lamplight*' applies a 3D top-down perspective, displaying the game map to all players at the same given time to conform to the concept of a 'Party Game'. The game is designed to work with mobile phones acting as controllers, allowing movement and abilities to be activated using a simple UI outlined below. This will be implemented by using 'AirConsole' technologies to allow hot joining within the game, directly appealing to the 'party game' playstyle of being able to pick up and play with relative ease (AirConsole.com, 2018).

Within this interface, players, either hunter or ghost, will be able to navigate using an analogue control stick alongside two interaction buttons. The first button will allow ghosts to spook a hunter to temporarily incapacitate them, likewise with a hunter they will be able to capture ghosts with a net device that will immobilize the ghost player until they are freed by friendly teammates. The second button initially is to provide both parties with the ability to 'fling' objects out of their way,

causing a distraction for ghosts or removing irritants as hunters. Both forms of players will also be able to interact with lamps by standing within an area of influence, gradually increasing the capture rate to whomever has the most weight within the radius. This 'capture' rate will be displayed on a progress bar to indicate the percentage remaining for either side to capture the lamp. Depending on the state of the lamp, if either party leaves the circle of influence it will begin to transfer back to its original state. For example, if the lamp originally was unlit and both parties left the collision area, then it will begin to extinguish the light source automatically back to '0' (Figure 7.C.1; Figure 7.C.2; Figure 7.C.3).

b. Level Environment and Interaction

The environment consists of a single floor haunted house plan based around would could be considered a 'Pre-War Mansion. Featuring only the ground floor of this building, it will consist of many uniquely thematic rooms connected through varied passageways. To each end of the passageway and rooms, a door will be featured that can be interacted with through pushing it with the player model. Outside the parameter of the playable environment, a decorative garden area will be utilized to allow better ambience and immersion within the game. Household and environmental objects will be present throughout the map, with some remaining static in place, such as the piano, and others becoming interactable as found with boxes. Interactable objects will allow basic physics to occur to allow players to push them around the game world and utilize them to their advantage. Whilst originally planned but cut due to time constraints, the environment would have contained random timer-based lighting visuals to provide a brief glimpse of where players are around the map. This was evolved out of the project as stated, and instead the luminosity of the lamps and lightning visuals found around the game world were significantly enhanced (Figure 7.A.1; Figure 7.A.6).

c. Lamp Capturing

The main fundamental mechanic and purpose behind the game relies on the concept of capturing or un-capturing these light sources to achieve victory. Both parties will have to find these lamps and prevent the opposite team from capturing them. Whilst every room will contain a lamp, not all will be activated, with a small set being activated and hidden during game generation. Whilst not invisible, these lamps will show no indication to both teams as to which is active, and thus both teams will have to navigate around the map to try and identify the lamp sources. Once either team stands inside, it will present a small UI featured above the lamp, symbolizing its current progress of being lit through a progress bar. Should the hunters be more numerous within the lamp's vicinity, the progress bar will increase until a maximum of 100. Likewise, if more ghosts are present and the lamp progress is higher than 0, then it will gradually decrease. Once reached, the lamp will remain lit or unlit and revert to this state in the event any members of the opposing team attempt to capture and leave the lamps vicinity (Figure 7.C.1; Figure 7.D.1).

4. Aesthetics

a. Art Style and Music

The aesthetics will be encompassing a family-oriented environment in order to become more accessible to a larger market audience. As a result of this, the graphical aspects of the game will remain suitable, simple and attractive to all audiences, with textures and lighting following suite to further boost the immersion and appeal of the game. All models produced will be low polygon count and textured, with various objects such as hunters and ghosts retaining a 'quirky', but comical animation to them. Lighting, sound and music will also be implemented to create an atmosphere of mystery and immersion, with the mansion layout being incredibly dimly lit through a variety of light

sources. External lighting such as lightning strikes or sounds such as creaking floorboards depending on areas which players walk on top of will also form an extra level of immersion.

b. UI

The UI aims to also follow this minimalistic approach, aiming to remain simple to read yet highly informative. Due to the game being situated on a single screen, the information present will have to be kept to a minimum to maximise the screen space available. As a result, many UI elements will either be featured on the user's controller, or within the game world itself. Formerly, this will include the analogue stick and ability buttons which will communicate within the games underlying code to provide minimal UI interaction. Likewise, the UI found for lamps and freeing ghosts will also be presented in game through a small progress bar above the object it's representing (Figure 7.A.4, Figure 8.1).

Roles and Tasks

Due to the nature of the game, various roles have been combined to achieve maximum potential within the given time frame rather than delegating them to individual members specifically. This ensures that all members will remain flexible and be able adjust quickly to achieve milestones should delays occur when meeting the deliverable targets. Each role type however was easily categorized depending on the task specifically requested to ensure all the deliverable milestones were completed quickly and efficiently.

1. Game Design (Olumide/Eleanor/Efren/Damon)

This role entails the creation and implementation of the game world through applying the back-end mechanics with the produced art assets and sound. Being highly diversified in content, this role aims to complete the game world once all the fundamental resources are readily available, such as textures and models. For this reason, it will also be arguably the most substantial in terms of time planning and expenditure. As such, the tasks involved are:

- All produced assets as shown with art and music files are imported into the project without any issues arising; ensuring the arbitrary scale value is adjusted to unity's scale system for consistency
- Construct the game world using the art assets providing by following strictly to the proposed house layout design as shown in Figure 8.1. Ensure that the garden and objects are placed in their respectively uniquely attributed locations to ensure that the rooms feel aesthetically fitting
- Construct gameplay mechanics as found with the refinement of the light capturing system, the connection of the victory conditions. In addition, secondary features should be implemented such as spawning, 'Boo' / 'Net' ability for ghost and hunter respectively and producing the output from the input responses.

All the proposed concept features outlined above and shown within the tasks were completed fully, with no tangible differences made to implementation to affect the achieved status. All gameplay elements and aesthetic elements were implemented, with exception for the removal of sound and the 'fling' ability for both player types from the game due to time constraints. However, the sound nor fling ability did not play detrimental parts to the game, and as such can still be classed as a success despite this.

2. UI Design (Stephan, Damon)

This role involves both the implementation and design of the UI Content that the players will utilize throughout playing the game, alongside creating and managing the scene changes and the 'AirConsole' implementation systems. This also includes the code to establish the inputs from the user, alongside the complimentary systems to change factors within the game world. Therefore, the several tasks involved within this category to be carried out into the project are:

- The creation of 'AirConsole' controller UI elements that will sit on top of the mobile phone users' interface. This controller UI needs to include the analogue stick for movement alongside the two ability buttons and a disconnect button.
- The handling of in-game UI elements as found within the lighting system implementation and the ghost freeing system. These elements should be positioned above their reflective objects, and gradually increase/decrease as specified within the original concept.

- Handling and creating the back-end systems to utilize 'AirConsole' as the controller, namely handling the inputs provided and applying the respective control responses
- Creation and application of the Main menu, two victory scenes and the game scene, alongside their transition conditions. In context of the victory scenes, this should link to the victory conditions specified within the gameplay mechanics.

Each task listed within this category was completed to the highest potential, with each UI and element being featured within the game and scenes successfully. No additional problems were found, and thus no changes had to be made to the initial task list to ensure completion of the final product.

3. Lighting (Damon)

This role, whilst near identical the game design in structure, will focus more-so on the implementation of the back-end systems required rather than the design itself; creating the fundamental systems for interaction and environmental immersion. The lighting systems will be used in conjunction alongside the gameplay mechanic implementations to allow better communication as found with the light capturing and player rotation of the lamp source. As such, the tasks involved are:

- The creation and implementation of lighting based back-end systems to allow game design members to apply modular interactions with the game environment. This will be found through the lamps application to automatically adjust the light source according to rotation and context
- To apply and implement the light capturing system which should work through a triggerable area of influence and number of player types within the circumference.
- To Create the framework to allow victory conditions to be checked with each light source being lit or unlit

Each task listed within this category was completed to the highest potential, with each light source working independently and capturable as expected. The light sources are determined at game creation as expected, with the relevant victory condition framework established.

4. 3D Modelling and Animation (Efren)

This role will involve the production of assets to form the game world, with tasks ranging from modelling to texturing and rigging / animation of the models themselves. This will be achieved through utilizing the 'Blender' application to construct the meshes themselves and apply the textures to match. In addition, the 'Mixamo' studio solution will be utilized to apply high quality animation skeleton rigs to the models (License – blender.org, n.d.; Mixamo FAQ, 2019). As such, the tasks involved are:

- The creation of the assets for the game world utilizing the full design pipeline regarding modelling and texturing. The main objects of significance will be the house, outside aesthetic elements, small objects such as piano/boxes and player models.
- To Apply detailed animations to the player models where relevant and ensure the state transitions work without problems when requested

Majority of models were produced for the game to provide an aesthetically pleasing world as found within the original concept design. All elements were added to rooms within the game world to allow unique identification, despite several elements still retaining primitive shapes as found with 'Beds' and 'Piano'. However, all other objects were made as shown with the house and the smaller objects littered around the rooms (Figure 8.1).

5. Report (All)

The report itself consists of the conceptual designs introduced, all the way to the implementation and evaluation of the fundamental architecture and designs. All members will contribute to the report, with the most apparent contributions arriving from their respective main roles within the team. The report aims to include various factors of conceptual design all the way to evaluation of production management to provide an insight into the design process implemented. In addition, the report hopes to highlight the costing, ethical and lawful considerations involved as shown within the appendices regarding costing plan and any additional copyrighted content.

Project Management Evaluation

The project was initially planned to be managed through a weekly agile scrum system, resulting in a weekly meeting to determine the following weeks goals. Alongside this, each member was also allocated to specific tasks, whilst remaining flexible should they complete their tasks early to ensure the team met deliverables for that week (Figure 4.1; Figure 4.2). All meeting content was recorded and documented in a summarized version found on canvas pages which contained the outlining of what each member should be completing each week, alongside what work got completed the previous week (Figure 8.2). The deliverables for the project were managed through a repository system known as 'TortoiseSVN', housing each commit with a descriptive message alongside the author's ID (Apache Subversion, n.d.; Figure 8.3). Every SVN commit was aimed to be completed with a simple but intuitive bullet pointed comment list outlining each change made by each member. This is especially so regarding the programming where multiple changes occurred within the same class object, resulting in many changes having to be tracked throughout iterations (Figure 8.3).

The initial planning was split into three distinct stages: planning, implementation and testing. Each stage aptly named for its specific contribution to the project, was carried out linearly and sequentially; with the final milestone for each section indicating success and the ability to start onto the next. The planning stage sought to draft both the report and the game side by side weekly through a draft documentation deliverable containing both respective types. This stage also sought the inclusion of many preliminary concepts to be implemented within the game project that were required for the implementation stage such as the 'AirConsole' integration and light sources. Finally, the documentation features were finalized containing the UML, Time plan, concept sketches and market research, ready to move onto the implementation stage. This stage instead brought the implementation and integration of all produced concepts and finalized the game by completion date. This also includes a small buffer period that could be absorbed if ahead of schedule or remain in place in the event of delays to the project. Having the game implemented and finalized successfully, the final 'testing' phase could begin. This phase sought to test the software against the initial designs to ensure that it worked against specification, alongside ensuring the documentation was completely finalized. Once this stage was complete regardless of whether the buffer period set out under the 'Completion' section was used, both deliverables were complete and finalized (Figure 4.1; Figure 4.2).

In hindsight, two major factors would be taken into consideration should this project be established again. Firstly, the report would be completed throughout the duration of the project instead of just within the planning phase. This was down to the issues experienced where UML diagrams had to be changed, or concept sketches simply did not work as intended. As a result, this meant the project ran into the non-critical buffer alongside delays to the project to ensure the changes were met. Secondly, the tasks themselves specified to group members could potentially be uncondensed, allowing a better visualization of what remains to be done, and better management should features have to be cut. Whilst this was easily applied for the music and sound files initially planned, various nested elements such as the lobby screen had to be removed in favour of quicker and less intuitive solutions such as the hot joining.

References

Airconsole.com. 2018. *Airconsole*. [online] Available at: <<https://www.airconsole.com/>> [Accessed 26 April 2020].

blender.org. n.d. *License — Blender.Org*. [online] Available at: <<https://www.blender.org/about/license/>> [Accessed 26 April 2020].

Helpx.adobe.com. 2019. *Mixamo FAQ*. [online] Available at: <<https://helpx.adobe.com/uk/creative-cloud/faq/mixamo-faq.html>> [Accessed 26 April 2020].

Horti, S. (2020). Some Epic Store games are pushing back or scrapping Steam releases. [online] pcgamer. Available at: <https://www.pcgamer.com/some-epic-store-games-are-pushing-back-or-scraping-steam-releases/> [Accessed 19 Feb. 2020].

Langley, R. (2012). Xbox Live Arcade by the numbers - the 2011 year in review. [online] Gamasutra.com. Available at: https://www.gamasutra.com/view/news/39713/Xbox_Live_Arcde_by_the_numbers__the_2011_year_in_review.php [Accessed 26 April. 2020].

MyFreeTextures, n.d. *Wooden Texture*. [image] Available at: <<https://www.myfreetextures.com/seamless-wood-texture/>> [Accessed 27 April 2020].

Reddit.com. n.d. *Reddit - /R/Gamedev*. [online] Available at: <<https://www.reddit.com/r/gamedev/>> [Accessed 26 April 2020].

Statista. 2016. *Video Games Played In 2015, By Game Type | Statista*. [online] Available at: <<https://www.statista.com/statistics/527986/uk-survey-video-games-played-by-game-type/>> [Accessed 26 March 2020].

Statista. 2019. *U.S. Most Popular Video Game Genres 2018 | Statista*. [online] Available at: <<https://www.statista.com/statistics/189592/breakdown-of-us-video-game-sales-2009-by-genre/>> [Accessed 26 March 2020].

Steam Database. n.d. *Steam Database*. [online] Available at: <<https://steamdb.info/>> [Accessed 26 March 2020].

Steam Database. n.d. *The Jackbox Party Pack*. [online] Available at: <<https://steamdb.info/app/331670/>> [Accessed 26 April 2020].

Steam Database. n.d. *When Is The Next Steam Sale?*. [online] Available at: <<https://steamdb.info/sales/history/>> [Accessed 26 April 2020].

Steamcharts.com. n.d. *Steam Charts - Tracking What's Played*. [online] Available at: <<https://steamcharts.com/>> [Accessed 26 March 2020].

Subversion.apache.org. n.d. *Apache Subversion*. [online] Available at: <<https://subversion.apache.org/>> [Accessed 26 April 2020].

Unity Store. (2020). Compare Unity plans: Pro vs Plus vs Free. Choose the best 2D - 3D engine for your project!. [online] Available at: <https://store.unity.com/compare-plans> [Accessed 19 Feb. 2020].

Vgchartz.com. 2020. *Video Game Charts, Game Sales, Top Sellers, Game Data - Vgchartz*. [online] Available at: <https://www.vgchartz.com/#embed_link> [Accessed 26 March 2020].

Xellena, 2018. *Wallpapertexture*. [image] Available at: <<https://www.goodfon.com/wallpaper/tekstura-fon-ornament-vintazh-victorian-style-pattern-seamle.html>> [Accessed 27 April 2020].

Appendices

1. Instructions Manual

Once the user has first acquired a copy of the game, they should be able to download it and navigate inside the folder. Once instead, they should notice an aptly named 'LampLight.exe' file, of which once opened will launch the game as place the user onto the main menu.

Within the main menu screen for the game, the user will have three available options presented to them: 'Play', 'Map Select' and 'Quit'.



Figure 1. Main Menu scene as described when the user first launches the game.

The user should first press on 'Map Select' to choose an available map they wish to play. This should be achieved through left clicking on the map, which will select and take them back to the main menu. From where, they will be able to press 'Play', which will start the game and produce the AirConsole link ready for people to start connecting using their mobile phones

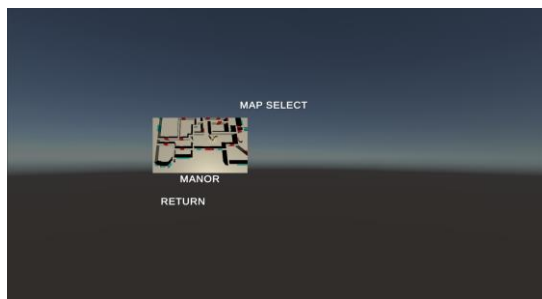


Figure 2. Map selection screen showing the available options that the user can select to begin playing the game.

In order to connect users, must navigate to the relevant app store and search for the 'AirConsole' app available to them. Once downloaded, they should be able to input the same code available on the screen into the phones, which in turn will spawn their character within the game world. When they have connected successfully, the control UI will appear on their phone, and their character within the game world as shown below

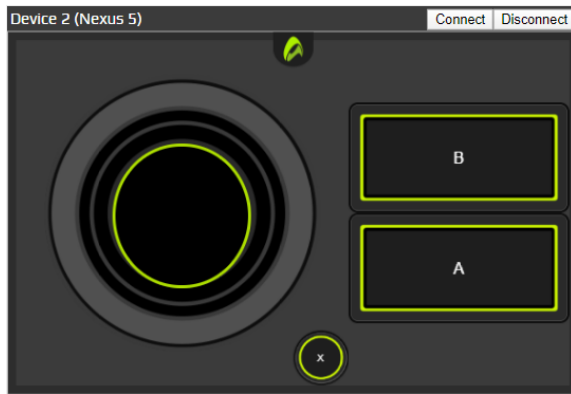


Figure 3. UI Controller that will appear once a user has successfully connected to the game.

Player character movement can be controlled through using the virtual joystick shown on the air console user interface. The hunter specific controls include net capture (button B) which can be used to throw a ghost net in the direction the player character is looking. Whereas the ghost's specific ability button is a spook button, that can be used to scare the hunter character. Both player types have a fling button (button A) that can be used to toss in-game objects. Players can also leave the game by pressing the leave button (button x).

There are two forms of player groups that the player will join depending on the order they have joined the game. Even players will be a hunter, whilst odd numbered players will be ghosts.

As a hunter, the objective of the game is for players to navigate through the map in search of oil lamps to capture through standing in the nearby vicinity. When all the lamps are lit, the hunters win. By lighting lamps, nearby ghosts become visible, giving the hunters a change to capture them with their net gun to prevent them from moving. Ghosts can extinguish these lamps, and thus make the hunters lose the game should they capture them through standing within the nearby vicinity.

Ghost players on the other hand keep the level in darkness by extinguishing all light sources in the level, whilst avoiding light sources near enemy hunters. Ghosts can fear hunters for a short period to give them an advantage in any situation – use this wisely! Captured ghosts can be freed with the help of an ally. An ally can stand close for a few seconds to free the captured ghosts, as shown with a progress bar to indicate their progression.

2. Marketing Plan

The proposed target audience for 'Lamplight' is aimed towards a 'Casual' and 'Party Game' audience, suiting all ages and thus the largest audience possible. This genre aims to appeal to people socially through providing a means of either family or friends to join together completing short and simple scenarios either together or against one another. Published by Statista in 2016, it was shown that casual games were played the most, with 27% of all surveyed UK population having played them regularly. In addition, 18% of this study have also played the 'Party Game' genre, sufficiently proving the target audience's suitability.

Despite this however, when looking multi-nationally the US has found that both genre types have not made significant impact and thus don't warrant their own categories. As a result, both genres must be presumed considered as shown within the category of 'Other' (Statista, 2019). Although in context of the UK alone, these games can also be considered successful despite this. The Jackbox Party Pack series is the most dominant series in the party games market. While it doesn't climb above the number 5 spot, multiple sequels take 50% of the top 10 showing people are committed to this franchise. With a total of 1.2 million units sold and the series being currently ongoing, it is a consistent front in the party game genre. One of the standout titles on the list is Castle Crashers.

While this game was released in 2012, is still one of the highest local multiplayer games on the platform with an estimated 5 million copies sold on Steam it continues to hold its strong position (Vgchartz.com, 2020). Stick Fight: The Game is another example of a game that was able to break through the competitive barrier to sit on top of the bestselling games; released in 2017, the title with marketing boosts from content creators on platforms, currently now sits in at over 2 million titles sold and growing (Vgchartz.com, 2020).

In context of economic evaluation and potential proposal for revenue, a few major factors ranging from game pricing to secondary revenue streams need to be considered. It was found that 'Castle Crashers' achieved 5 million copies on steam, with 2 million copies on the xbox 360 within 2 years costing only £10 (Vgchartz.com, 2020; Langley 2012). Released in 2014, Jackbox Party Pack also followed suite by keeping to a consistent price bracket fluctuating depending on sales between £18.99 and £6.64 (The Jackbox Party Pack, n.d.). Therefore, a safe assumption can be made to sell the game based on £10 due to the success found with both games, and fluctuate the game alongside sales as found with steam to ensure maximum potential revenue (When is the next steam sale?, n.d.). In addition, given the sale rate for these games of 2M and 5M over 6 years respectively, another safe assumption can be made that providing the game successfully applies itself to the market, around 550K sales can be expected minimum as deduced from both comparative game figures.

In addition to this, several cost factors must be evaluated beforehand with the inclusion of the software used to produce this game. The software 'Blender' and 'Mixamo' used for the 3D modelling and animation respectively do not contribute any expenditure by conforming with the GNU license and remaining completely free to use respectively (License – blender.org, n.d.; Mixamo FAQ, 2019). The main expenditure during the production of the product will be that of the engine itself and the publisher platform, with Unity taking a flat \$150 rate annually per team member and 'Steam' taking a 30% contribution cut (Horti, 2020; Unity Store, 2020).

Likewise, to this, secondary revenue sources such as 'Kickstarter' cannot be directly counted upon and planned around due to them arguably being more heavily subjective to public awareness and developer reputation (Kickstarter.com). Instead, more direct application of cost should be spent towards advertising through the means of free social media sites. Sites like 'Reddit' can be used to keep daily developer contact with the community alongside post content and generate social awareness about the product (Reddit -/r/GameDev, n.d.). In addition, advertisement through banners and various ads on websites can be considered, as found with 'Google Ads' that can be adjusted and applied at an easy rate to identically reflect the amount of revenue the game produces within a specified time (Ads.google.com, n.d.).

3. Ethical and Legal Considerations

Nearly all graphical assets including UI design were created by our own team with a few minor exceptions from accredited open-source sources. As such, in terms of legal commitment of the project, no material was infringed, and all licenses / content has been accounted for and legally used. The modelling software used to create our assets was Blender utilizing the mixamo studio package, with the former being open source under the GNU license, and the latter free to use commercially (License – blender.org, n.d.; Mixamo FAQ, 2019). In order to build this project, we used Unity; as we were developing this for a university project, we did not need to purchase a licensing fee but if we planned to release this as a full game we would have needed a business licence in order to develop and release the game (Unity Store, 2020). TortoiseSVN, which we used for version control, is free for commercial use, and as such did not need to take into consideration any legal implications from utilizing it (Apache Subversion, n.d.). AirConsole was also used for implementing the multiplayer game access; this is a subscription-based service for users, but developers do not need to pay a fee to host a game or release commercial products using it (AirConsole, 2018). In terms of legal issues, no parts of this game or product can or will deem themselves to be detrimental

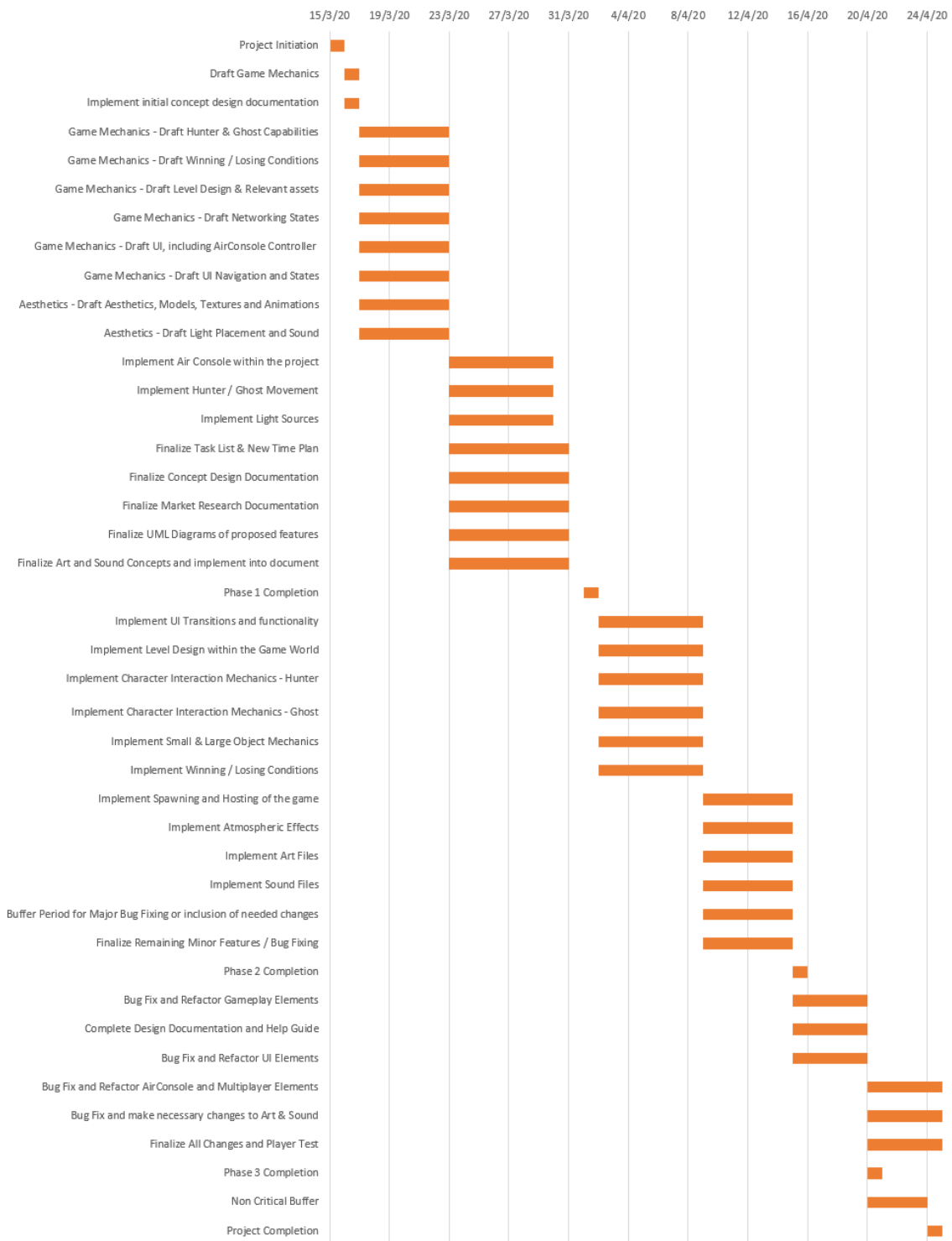
to any persons upon playing. No material or content is in breach or finds itself to be explicit, ensuring that any content produced can be used for the intended family audience.

4. Time Plan

Figure 4.1. Time plan and development cycle shown through the expected start date, end date, duration and team member allocation

	Task Name	Task Layout				
		Duration	Start	Finish	Allocated Members	Type
Phase 1 - Planning	Project Initiation	1d	16/03/2020	16/03/2020	All	Milestone
	Draft Game Mechanics	1d	16/03/2020	17/03/2020	All	Task
	Implement initial concept design documentation	1d	16/03/2020	17/03/2020	All	Task
	Game Mechanics - Draft Hunter & Ghost Capabilities	6d	17/03/2020	23/03/2020	All	Task
	Game Mechanics - Draft Winning / Losing Conditions	6d	17/03/2020	23/03/2020	All	Task
	Game Mechanics - Draft Level Design & Relevant assets	6d	17/03/2020	23/03/2020	Eleanor	Task
	Game Mechanics - Draft Networking States	6d	17/03/2020	23/03/2020	All	Task
	Game Mechanics - Draft UI, including AirConsole Controller	6d	17/03/2020	23/03/2020	All	Task
	Game Mechanics - Draft UI Navigation and States	6d	17/03/2020	23/03/2020	All	Task
	Aesthetics - Draft Aesthetics, Models, Textures and Animations	6d	17/03/2020	23/03/2020	Efran	Task
	Aesthetics - Draft Light Placement and Sound	6d	17/03/2020	23/03/2020	All	Task
	Implement Air Console within the project	7d	23/03/2020	31/03/2020	Damon, Olumide, Stephan	Task
	Implement Hunter / Ghost Movement	7d	23/03/2020	31/03/2020	Olumide	Task
	Implement Light Sources	7d	23/03/2020	31/03/2020	Damon	Task
	Finalize Task List & New Time Plan	8d	23/03/2020	01/04/2020	Damon	Task
	Finalize Concept Design Documentation	8d	23/03/2020	01/04/2020	Eleanor, Efran, Stephan	Task
	Finalize Market Research Documentation	8d	23/03/2020	01/04/2020	Olumide	Task
	Finalize UML Diagrams of proposed features	8d	23/03/2020	01/04/2020	Stephan	Task
	Finalize Art and Sound Concepts and implement into document	8d	23/03/2020	01/04/2020	Efran	Task
Phase 2 - Implementation	Phase 1 Completion	1d	01/04/2020	01/04/2020	All	Milestone
	Implement UI Transitions and functionality	7d	02/04/2020	09/04/2020	Stephan	Task
	Implement Level Design within the Game World	7d	02/04/2020	09/04/2020	Eleanor, Efran	Task
	Implement Character Interaction Mechanics - Hunter	7d	02/04/2020	09/04/2020	Olumide, Damon	Task
	Implement Character Interaction Mechanics - Ghost	7d	02/04/2020	09/04/2020	Olumide, Damon	Task
	Implement Small & Large Object Mechanics	7d	02/04/2020	09/04/2020	Eleanor, Efran	Task
	Implement Winning / Losing Conditions	7d	02/04/2020	09/04/2020	Stepan	Task
	Implement Spawning and Hosting of the game	6d	09/04/2020	15/04/2020	Olumide, Damon	Task
	Implement Atmospheric Effects	6d	09/04/2020	15/04/2020	Efran	Task
	Implement Art Files	6d	09/04/2020	15/04/2020	Efran	Task
	Implement Sound Files	6d	09/04/2020	15/04/2020	Efran, Eleanor	Task
	Buffer Period for Major Bug Fixing or inclusion of needed changes	6d	09/04/2020	15/04/2020	All	Task
	Finalize Remaining Minor Features / Bug Fixing	6d	09/04/2020	15/04/2020	All	Task
	Phase 2 Completion	1d	15/04/2020	15/04/2020	All	Milestone
Phase 3 - Testing	Bug Fix and Refactor Gameplay Elements	5d	15/04/2020	20/04/2020	Olumide, Damon	Task
	Complete Design Documentation and Help Guide	5d	15/04/2020	20/04/2020	All	Task
	Bug Fix and Refactor UI Elements	5d	15/04/2020	20/04/2020	Stephan	Task
	Bug Fix and Refactor AirConsole and Multiplayer Elements	5d	20/04/2020	24/04/2020	Olumide, Damon	Task
	Bug Fix and make necessary changes to Art & Sound	5d	20/04/2020	24/04/2020	Efran, Eleanor	Task
	Finalize All Changes and Player Test	5d	20/04/2020	24/04/2020	All	Task
	Phase 3 Completion	1d	20/04/2020	20/04/2020	All	Milestone
Complete	Non Critical Buffer	4d	20/04/2020	24/04/2020	All	Non Critical Task
	Project Completion	1d	24/04/2020	24/04/2020	All	Milestone

Figure 4.2. Gantt chart showing a visualized chart of the tasks produced within the task list, alongside the date of estimated completion



5. Time Sheets

Expenditure and projected income per year

<u>Item</u>	<u>Quantity</u>	<u>Cost</u>	<u>Total</u>
Salaries			
Developer	5	£28000	£140000
Licensing			
Unity	5	~£325*	£1625
TortoiseSVN	5	£0	£0
Blender	5	£0	£0
AirConsole	5	£0	£0
Marketing			
Advertising (Google Ads, Facebook)	1	£9000	£9000
Social Media	1	£0	£0
Other Expenditures			
Computers (already acquired)	5	£0	£0
Development location (already acquired)	1	£0	£0
Total expenditure			
<u>Income per year</u>			
<u>Item</u>	<u>Quantity</u>	<u>Cost</u>	<u>Total</u>
Crowdfunding			
Kickstarter	1	£0	£0
Unit Sales			
Release units	55000	£10	£550000
Airconsole	1	£0	£0
Total income			£550000
Yearly Profit			£399375
Seven Week development timeframe			£53762

* = Subject to change due to dollar to GBP conversion rate as of 27/04/2020.

Figure 5.1. *Table detailing the cost and projected sales against the market research conducted.*

6. Individual Statements, Contributions and Portfolio

a. Damon

During this project I completed and contributed towards mostly the report writing and the programming. Majority of the report was produced solely alongside all main body elements completed minus that of the concept sketches and UI diagrams. Majority of the code was produced by me except for player animation, UI handling and ghost 'Boo' ability. In addition, I contributed towards the team through organizing all the meetings and performing the type ups prior and after the meetings.

Portfolio Link: <https://mythious.github.io/>

b. Efren

During the project I helped contribute to the project planning and designing, as for my role in the project I was tasked to do majority of the game's assets such as the 3D models, textures and animations. I further helped with minor debugging and testing where needed. Overall, I think the project was successful as majority of the initial ideas and aims of the project were implemented. Everyone did their part. It was very enjoyable.

Portfolio Link: <https://krausse1711.wixsite.com/website>

c. Eleanor

During this project, my role was mainly working on parts of the report, such as the legal and ethical issues we had faced, and working on timesheets and costings. I also helped with aesthetic choices for the game; for example, whilst music and sound were not able to be implemented in the final game, I had the task of selecting music to meet the aesthetic choices we had already made in the game (light-hearted, low-poly). I also worked on sourcing sound effects for the game. Another contribution I made was defining and filling out the map with placeholder models, so some idea could be had of what the map looked like and testing and coding of movement and abilities could be done before the map was populated with the final models.

Portfolio Link: <https://github.com/BasicGoth/University-Portfolio>

d. Olumide

I created the initial concept for Lamplight which was voted by the rest of the group as the idea to develop. During planning I worked on the market research on audience, sales and potential competition and also worked on the sequence diagrams for the interactions of the game and its systems. During development I was an assistant programmer, checking, touching up and finishing aspects of the game including the abilities system and general cleaning of some code.

Portfolio Link: <https://github.com/DrunkenHyena>

e. Stephan

Over the course of this group project, I have contributed largely towards game design choices and personally worked on developing and implementing the in-game user interface. This has involved developing and refining the core gameplay loop while communicating with team members to ensure that the project's design direction was fully understood and achievable within the team.

Portfolio Link: <https://github.com/Shteb/The-Code-Cove>

7. UML Diagrams and Concept Sketches

c. Concept Sketches (Figures 7.A)

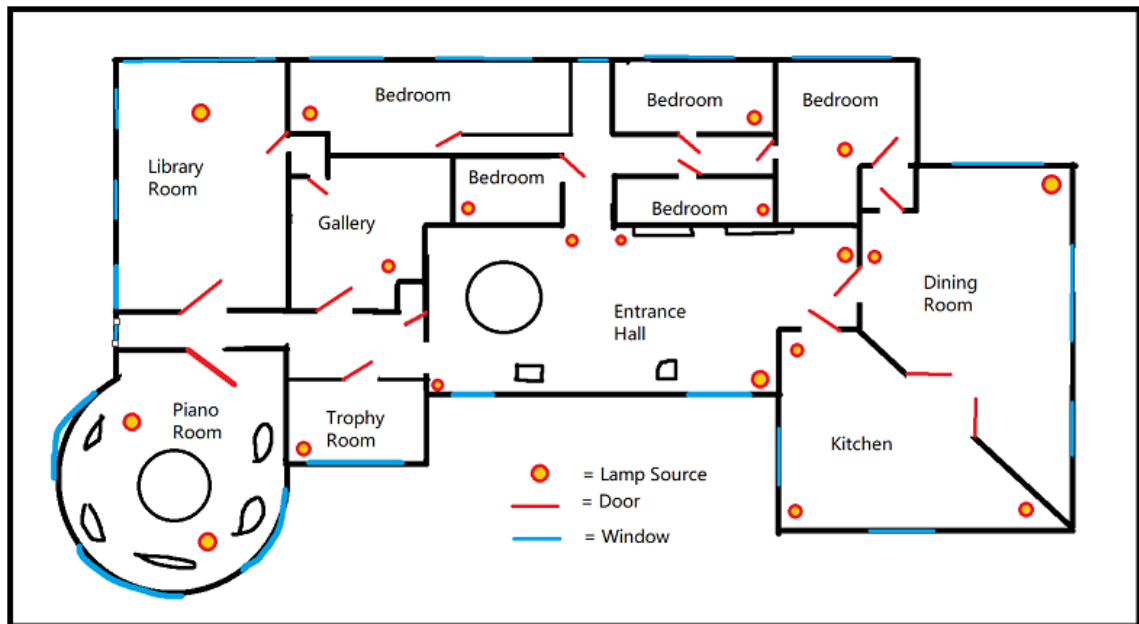


Figure 7.A.1: Sketch of the level design floor plan used throughout the game, including the potential lamp source locations and the doorways



Figure 7.A.2: Main menu UI showing the availability to the host user when launching the game.



Figure 7.A.3: Host game menu detailing the current connections within the lobby for both hunters and ghost lists which occur through passing an activation code to users through the 'AirConsole' interface.

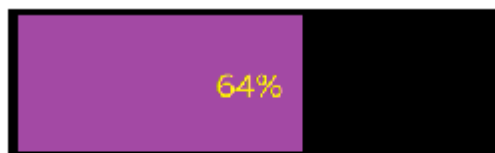


Figure 7.A.4: In-game UI which will be present for all users to see, displaying the current percentage status of a capture point, whether be a light source or freeing a ghost. This will appear directly above the object in question).

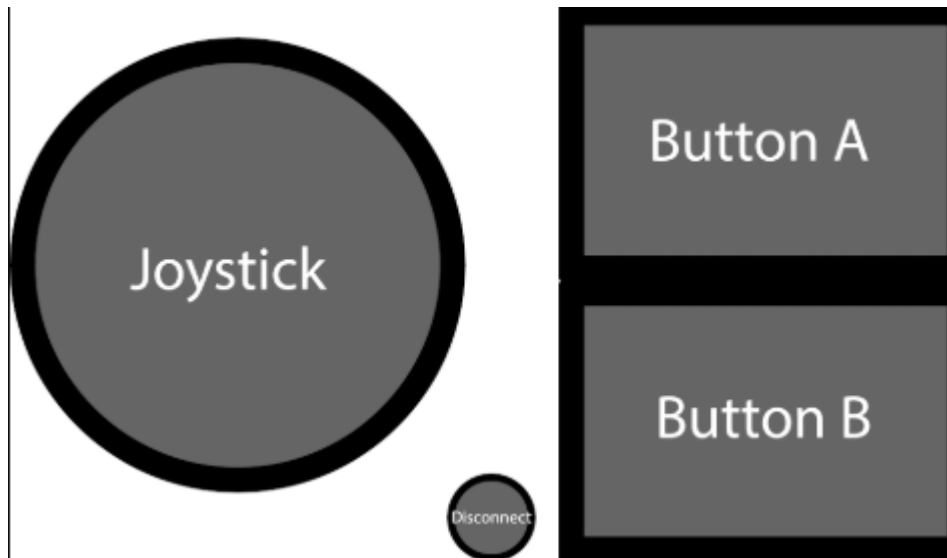


Figure 7.A.5: UI display showing the availability to the user that uses AirConsole to act as the controller, with a joystick for movement, disconnect button and two interactable buttons for navigating and interacting.



Figure 7.A.6: Collection of concept sketches that form the base model designs for the level content found within the game world

d. Use Case Diagrams (Figures 7.B)

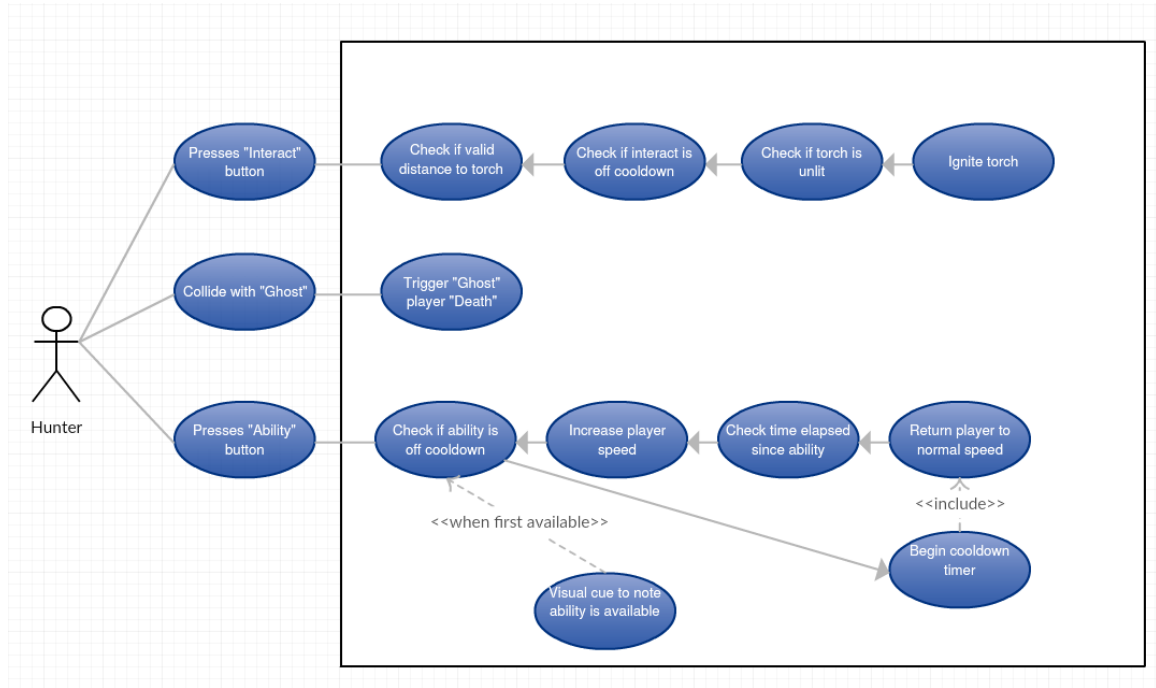


Figure 7.B.1: Use Case Diagram detailing the interactions between hunters and game play mechanics



Figure 7.B.2: Use Case Diagram detailing the interactions between Ghosts and gameplay mechanics

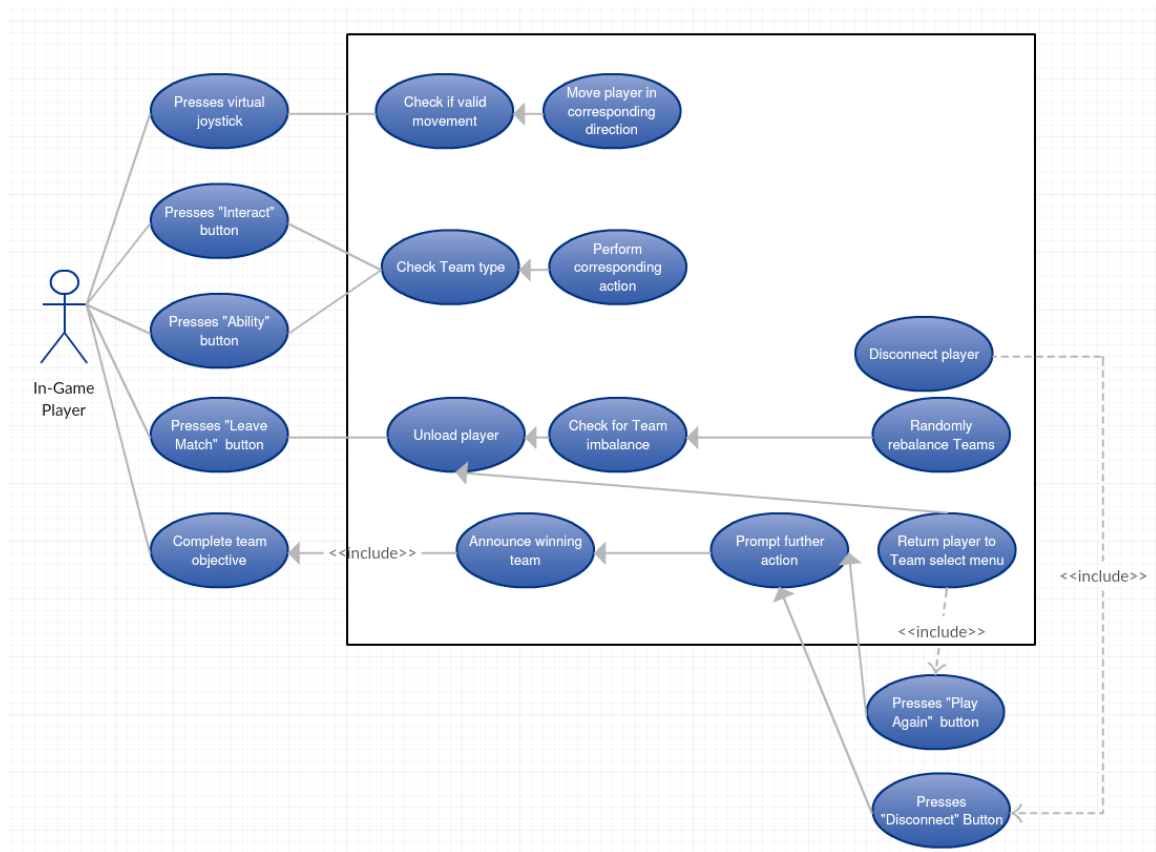


Figure 7.B.3: Use Case Diagram detailing the shared logic between both the ghosts and hunter players within the game world

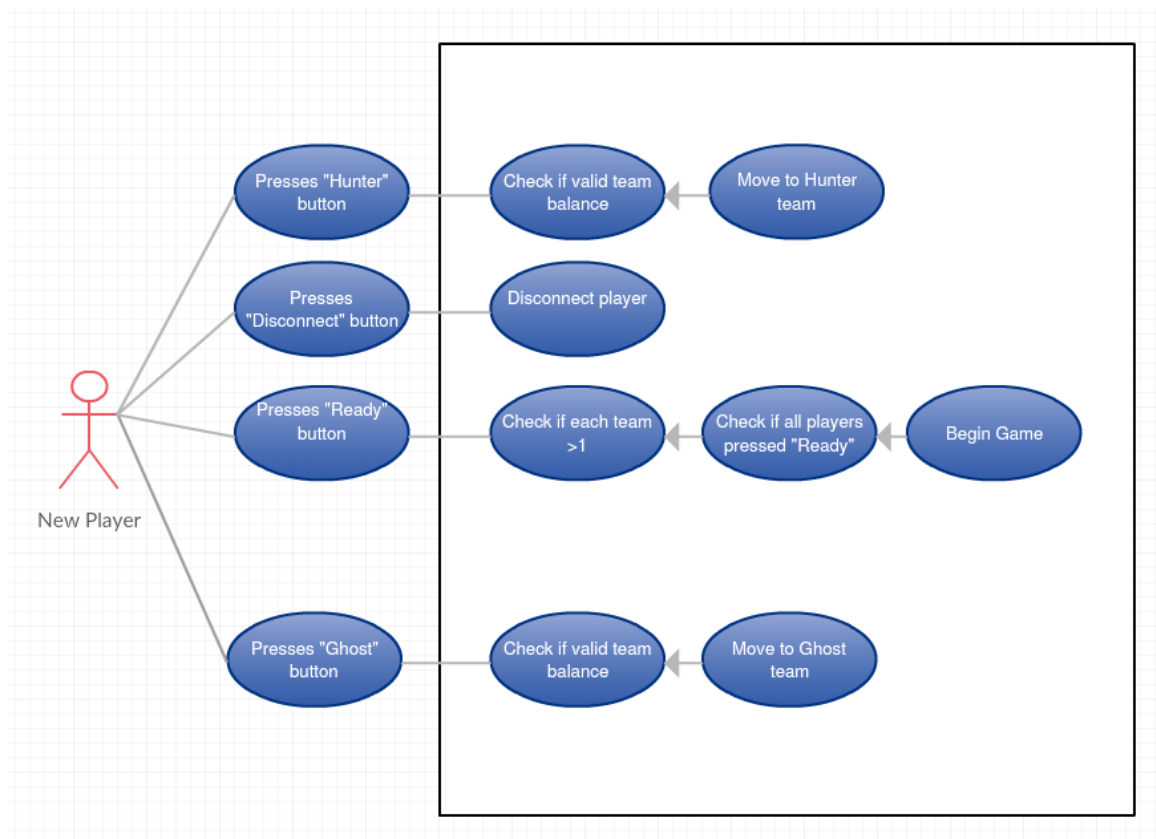


Figure 7.B.4: Use case diagram detailing the behavior's and mechanics that a new player would navigate through in order to enter the game world and begin playing

e. Class Diagrams (Figures 7.C)

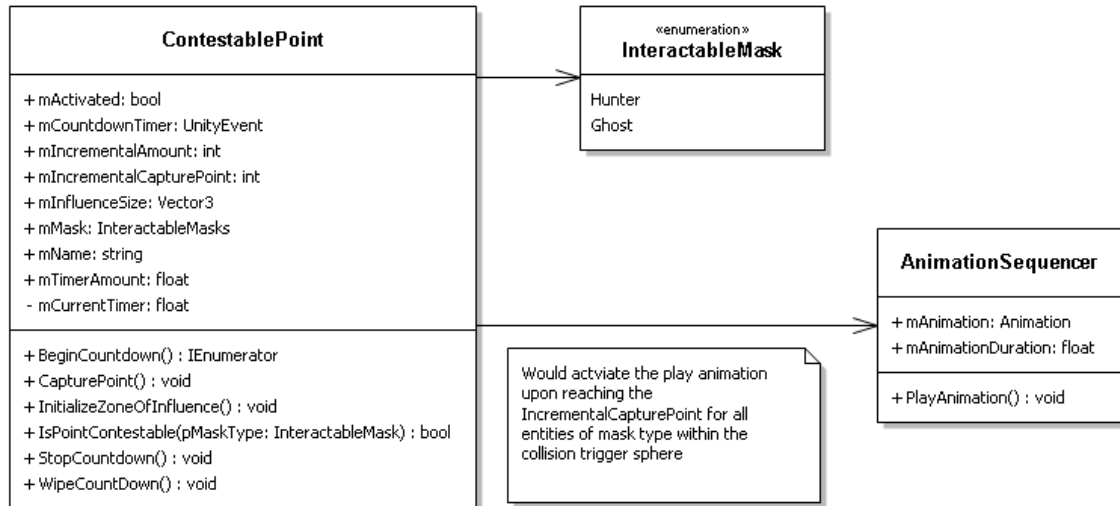


Figure 7.C.1: Class Diagram outlining the methods and data transfer shared between contestable points and the animation sequences of entities within the points zone of influence

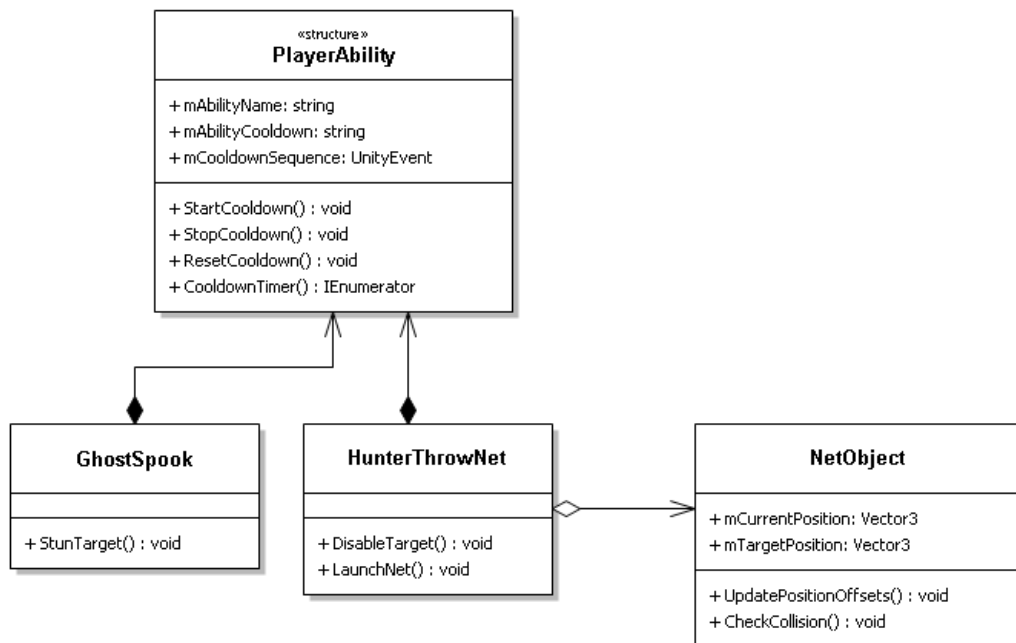


Figure 7.C.2: Class Diagram outlining the behaviours and data transfer between abilities for both the hunter and the ghost with their respective abilities

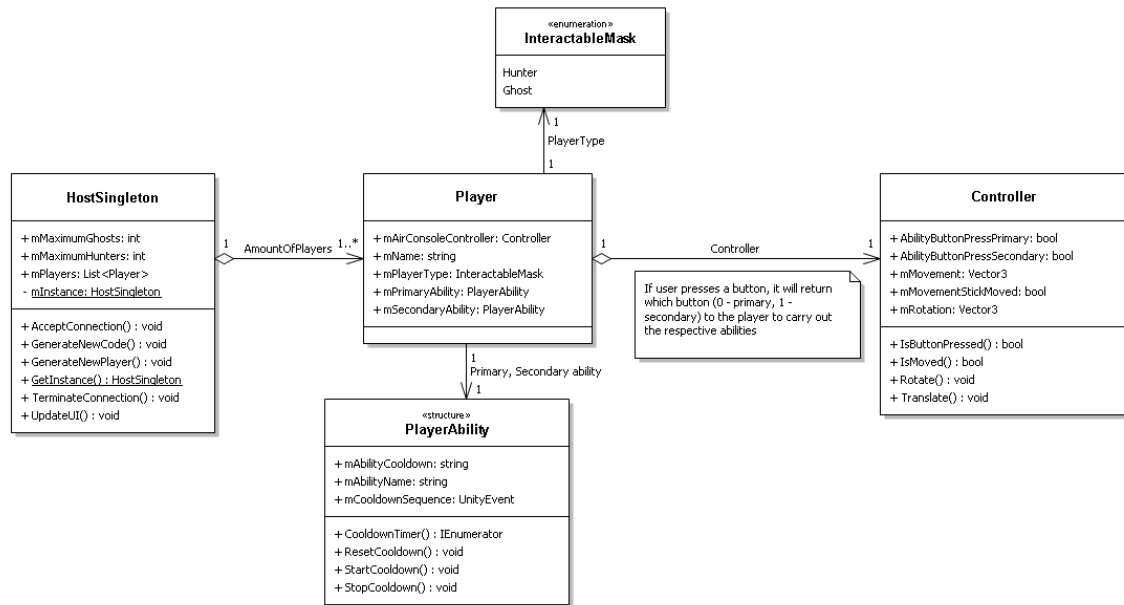


Figure 7.C.3: Diagram detailing the connection classes involved for a new player joining the server lobby

f. Activity Diagrams (Figures 7.D)

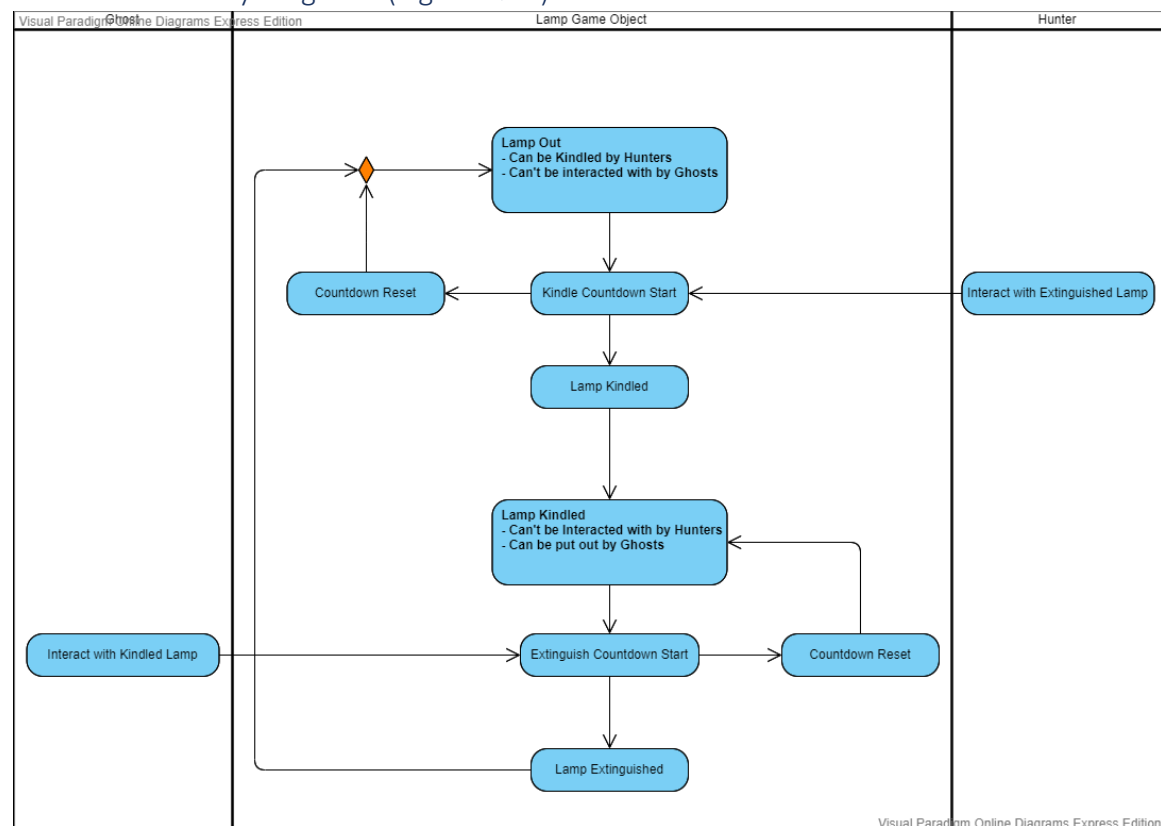


Figure 7.D.1: Contestable point activity diagram outlining the steps involved for either types of player to capture the point under their control once stepping in the 'sphere of influence'

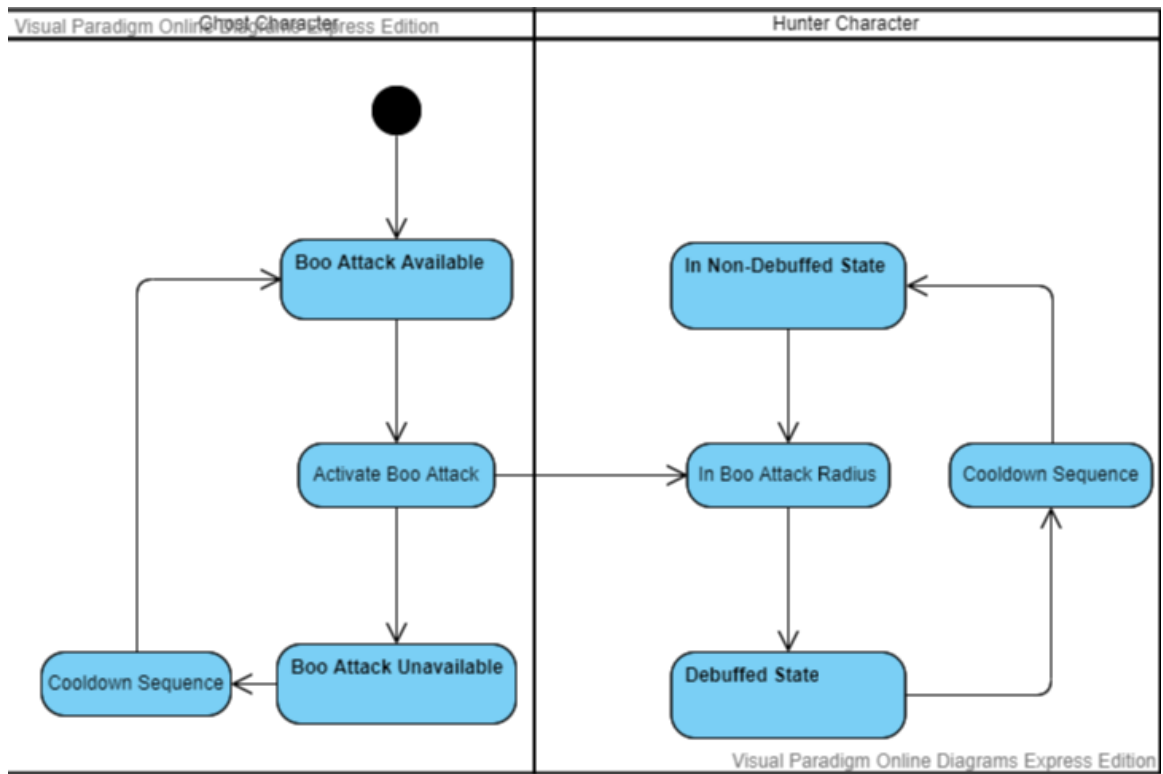


Figure 7.D.2: Activity diagram outlining the steps involved for the 'Boo' interaction from the ghost players

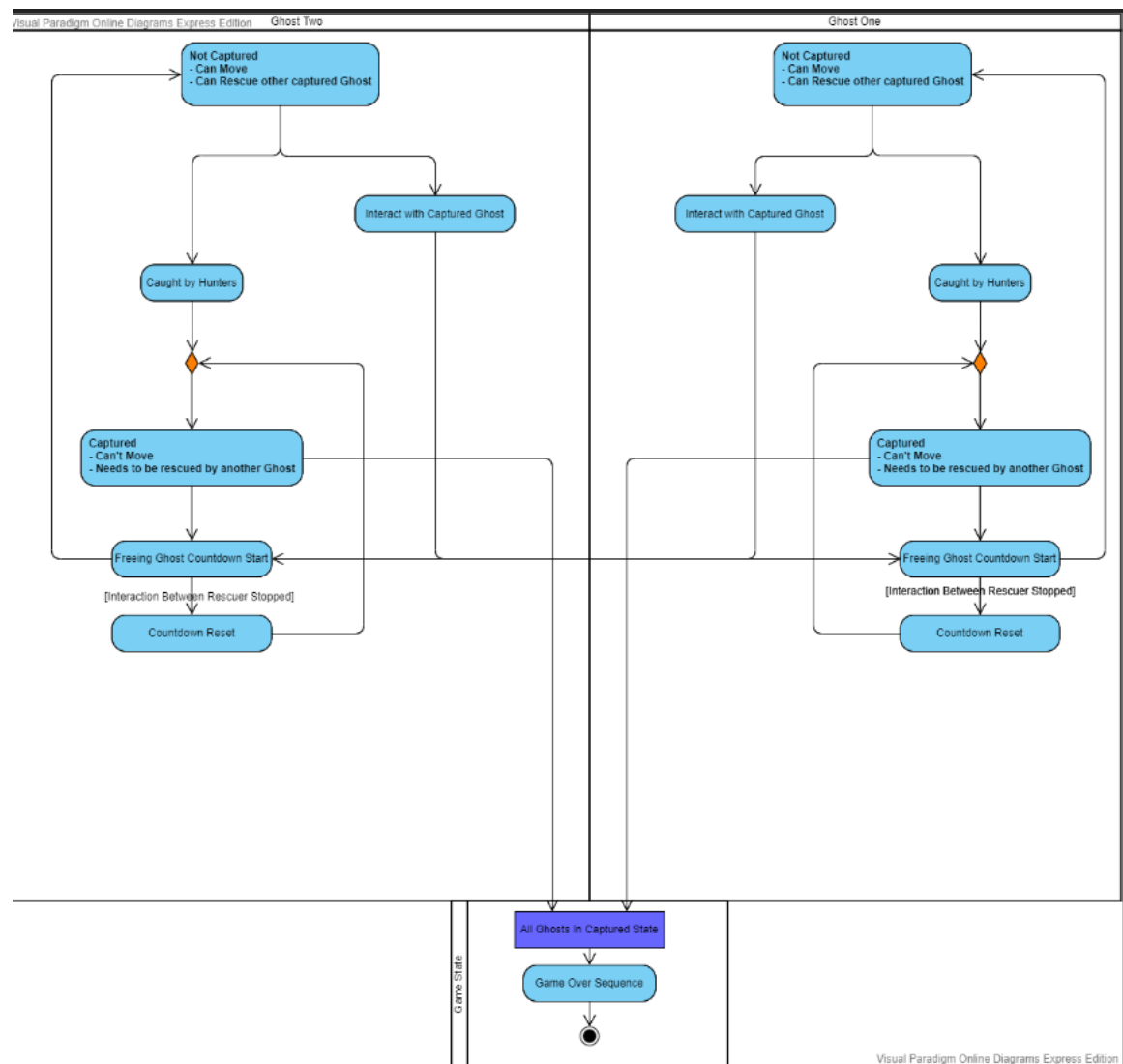


Figure 7.D.3: Interaction between ghost players after a hunter has captured a ghost and the behaviours involved for freeing them

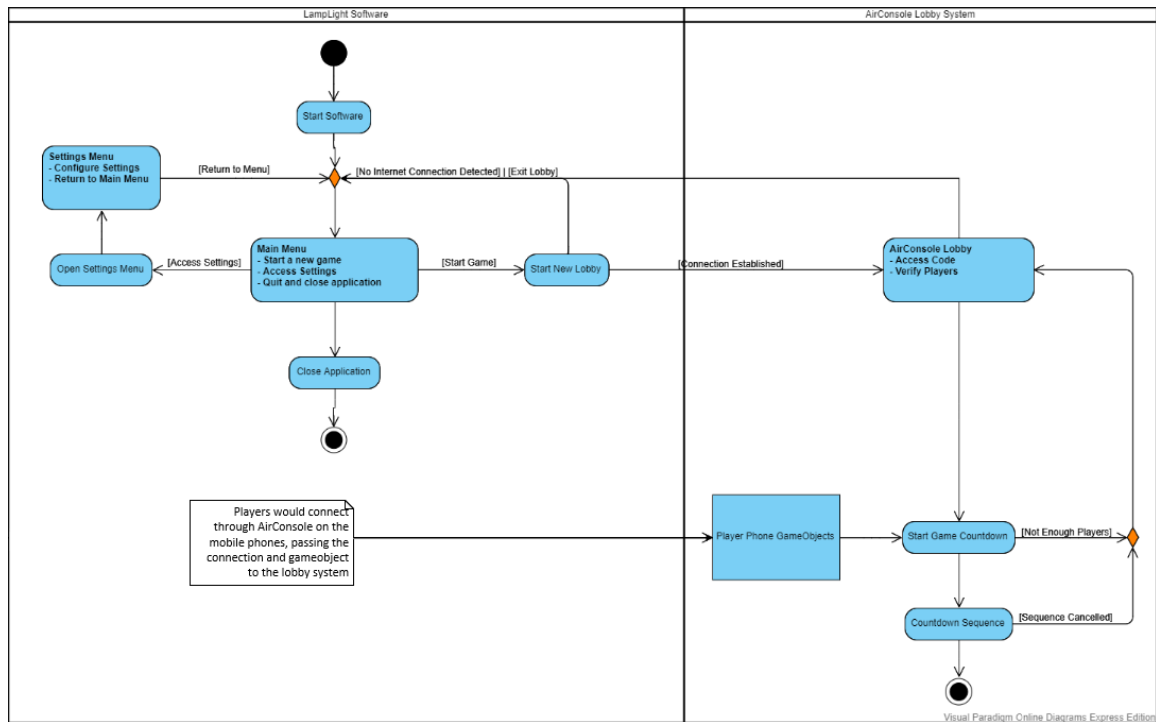


Figure 7.D.4: Activity diagram outlining the behaviours and steps involved for utilizing AirConsole within the 'Lobby' system within the game prior to starting the game world

8. Additional Items (Including Game Screenshots)

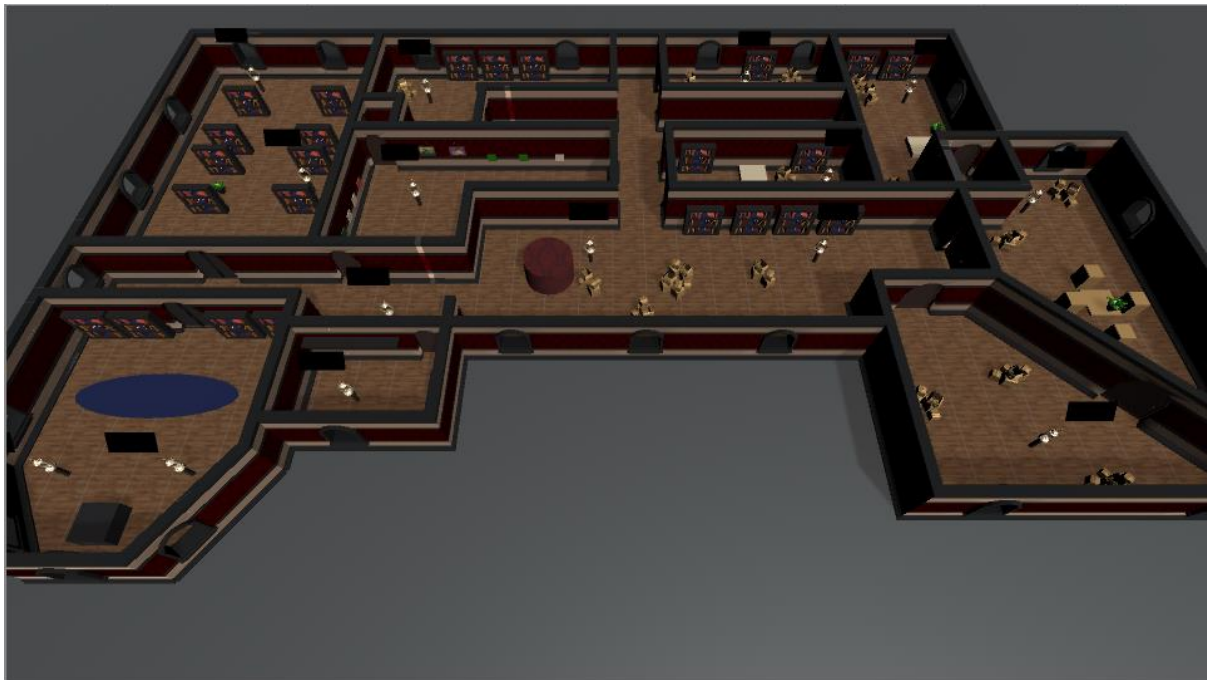


Figure 8.1. Screenshot detailing the pre-made game world with all elements introduced, including the small objects, art and light capture points. Foliage and background assets are disabled to provide clearer view.

Attendance

All Members Attended.

Run time: 2 Hours

Meeting Purpose

The purpose of this meeting was to finalise the team member roles, alongside prepare documentation and initial concepts in order to meet the following deliverables. In addition, technologies and relevant libraries were also finalised alongside the first functional implementations of the project ready for full development

Conclusion

The rough game concept stands as previously mentioned, finalised with the following key principles:

- The game will involve two parties, the 'hunters' and the 'hunted'
 - Hunters aim is to ignite all of the lamps by chasing and eliminating the ghosts from play. They will have to ignite the lamps by standing close to them
 - Ghosts aim is to put lamps out, but in the act of doing so, they reveal themselves to the hunter and risk getting caught
- The game finishes when either all ghosts have been eliminated or all lamps have been lit in context of the hunters, whilst the ghosts achieve victory through eliminating all light sources within the game

Technologies / Initial mechanical concepts are derived from the following:

- Air console will be implemented in order to act as a controller to move around for all parties on the same screen
- The environment will be simple, family oriented 3D with atmospheric textures, lights and models
- Utilizing Unity's built in light sources and physics engine due to it not being a requirement in the game and thus not suitable to redesign a different solution
- GitHub will be utilized for working and developmental branches, with builds being published on the SVN on a bi-weekly basis and/or milestone deliverable targets have been reached
- **Unity 2019.3.5f1 will be deployed as the active game engine build**

Aims for next meeting on Thursday 19th March 2020:

- Implement a basic handshake code for AirConsole within a new unity project
- Generate light sources for both static and dynamic integration to models at a later date
- Implement WASD Movement for initial debug, with the aim to have movement handled by normalized analog inputs through AirConsole
- Implement basic UI to handle joining and handling of AirConsole
- Develop the concept brief deliverable with complimentary time plan and Gantt Chart

Confirmed Team Member Roles:

The initial team roles have been designated with the initial concept of being able to move between roles as of when required to aid existing team members to meet deliverable targets. The roles given are more guidelines as to the departments each member should be working within initially for managing workloads and handling discrepancies

- Damon - Physics / Lighting + Meetings and Git-Master
- Eleanor - Game / Level Design + Report Contribution
- Olu - Game / Level Design
- Stephan - UI Design + Game/Level Design
- Efran - Modelling / Animation + Report Contribution

Figure 8.2. Screenshot showing the notes produced during a weekly scrum meeting and the layout used.

The screenshot displays the SVN revision history for a project. The top section shows a filter bar with options for Messages, Paths, Authors, Revisions, Bug-IDs, Date, and Date Range. The main table lists revisions from 173 to 188, with columns for Revision, Actions, Author, Date, and Message. Revision 188 is selected, and its details are shown in the bottom section, including the commit message and a list of paths that were modified or added.

Revision	Actions	Author	Date	Message
188		551834	26 April 2020 11:05:24	- Report body completed minus smaller reference changes
187		551834	26 April 2020 10:11:59	Project evaluation complete
186		551834	26 April 2020 08:51:30	- Implemented first half of the report, containing the layout and initial concept designs within the first section.
185		552721	25 April 2020 23:22:55	Added Portraits of Efran and Olu
184		551834	25 April 2020 22:51:21	- revision fix
183		551834	25 April 2020 22:09:58	Version Fix
182		554068	25 April 2020 21:27:34	
181		554068	25 April 2020 21:06:40	
180		554068	25 April 2020 20:46:20	
179		551834	25 April 2020 20:43:04	- Implemented hunter net - Implemented capturing ghosts and freeing behaviours
178		551834	25 April 2020 16:14:47	- Changed AirConsole to load utilizing only mobile phones, removing virtuals
177		551834	25 April 2020 16:13:13	- Implemented Opacity for the ghosts - Rebalanced the light intensity for the hunters
176		551834	25 April 2020 16:06:50	- Implemented numerous bug fixes relating to lighting, multithreading, character interaction and lamp intera...
175		551834	25 April 2020 15:41:07	- Implemented victory condition checks
174		551834	25 April 2020 15:03:54	- Implemented connector fix for the Controller - Implemented animation fix for the controller. Models should...
173		551834	25 April 2020 15:02:42	- Removal of broken version

Showing 83 revision(s), from revision 43 to revision 188 - 1 revision(s) selected, showing 4 changed paths

Figure 8.3. Screenshot showing the SVN usage alongside how commit messages are used, and the version control implementations provided.

Two textures were used in context of the game production, with the 'Wooden Floor' texture being utilized for the building floor, alongside the wallpaper for the house walls (MyFreeTextures, n.d.; Xellena, 2018).

Figure 8.4. Statement dictating the resources used within the game, with all associated resources free to used and acquire from open source resources.