Educational Game – River Recycle

CSC8599 Dissertation Project

YouTube Demo Link: https://youtu.be/v9mYW8_TBC8

Bailey Gardner
School of Computing
Newcastle University
Newcastle Tyne & Wear England
b.gardner@ncl.ac.uk

ABSTRACT

This paper describes the process and evaluates the outcome of a project to teach people how to correctly recycle rubbish within the United Kingdom based on published government guidelines. This was carried out through the development of an educational game which informs the player how to correctly recycle a variety of different types of rubbish based on their material. The basic idea of the game entails a river in which rubbish will flow down; the player must place the rubbish into the correct recycle bin before it reaches the end of the river. The intention is to hopefully educate as many as possible on the many nuances of recycling rubbish correctly within the UK, as lack of recycling knowledge is currently a serious problem resulting in the inability to recycle and therefore, preventing the reuse of a large amount of material.

1 INTRODUCTION

Effectively recycling waste is one of the most difficult ongoing problems the world currently faces. As we move ever closer to material shortage it is becoming increasingly vital that we attempt to effectively recycle as much waste as possible, especially as the demand for many materials continues to increase.

There are several factors which determine how effectively waste can be recycled, such as the amount of energy required to recycle a particular material. For example, a plastic such as Polyethylene terephthalate (PET), commonly used for clear water bottles, are fairly low cost to recycle where as a plastic such as Polyvinyl chloride (PVC), commonly used for doors, windows and bank cards are much more difficult to recycle due to the high cost of the recycling process. [1] This creates discussions around the topic of whether it is worthwhile recycling materials with a high costing process since it is so inefficient economically in addition to the amount of energy required.

While the previously described problem is extremely important to address and should not be ignored, this project intends to focus on tackling another recycling-based issue. This issue is the fact that many people are misinformed or simply do not fully understand how they should be recycling different materials. For example, many people do not understand exactly which materials can be accepted for recycling and which cannot. This project will try to help tackle this problem by educating as many as possible so that they understand exactly how to correctly recycle various different materials.

To illustrate the problem this project is attempting to help solve, below is a list of plastic-based items that are currently not accepted by recycling plants. [2]

- Black plastic
- Crisp and sweet packets
- Plastic toys
- Expanded polystyrene

Since these materials are all plastic based, it would be understandable if the majority of people placed these types of items into their recycling bin. However, this simply results in more work for recycling facilities as

they must sort not only many varieties of recyclable materials but also filter out and dispose of any materials which cannot actually be recycled. This illustrates a case in which this project could help with this issue as informing people which materials can and cannot be recycled may help reduce a lot of the sorting work recycling plants currently have to carry out, especially if the item, at first glance, appears recyclable but in truth is actually not. For example, it would be fair to assume that most people do not realise that black plastic cannot be recycled. This is because recycling facilities sort plastics using optical sensors which utilise light beams. As black absorbs a large amount of light, it cannot be detected by the system and is therefore automatically sent to landfill or incineration. [3] Another example is food containers. Food containers made from recyclable materials such as plastic and cardboard are recyclable however, only if the food residue has been removed from the container otherwise, they will simply be rejected. [2] One final example is glassware. Glass jars and bottles are completely recyclable but items like drinking glasses and ceramics are not. This is due to the fact that the latter have a much higher melting point which means if they enter the glass recycling process, it can result in new containers being rejected which is, of course, a huge issue. [2]

This project will attempt to inform people of the many nuances similar to these which should be considered when attempting to recycle a variety of materials.

One of the major reasons the issues described above exist is because there has never been much of an attempt to educate people on how to correctly recycle in the most effective way possible. Realistically this problem is something that needs to be tackled on a much larger scale than what this project will be able to achieve. However, it intends to be a small step to illustrate the direction that should be taken on that much larger scale. An example of something that could help may be delivering a short and concise booklet of recycling guidelines to every household. However, if this booklet were too big, it would be likely that many people would simply discard it without reading it, which further illustrates the problem this project hopes to help tackle by offering people a fun and engaging way to learn.

2 RELATED WORK

To gain a better understanding of how the project could be approached it was important to research into similar related work. This was important to ensure the project did not simply regurgitate work already carried out by other sources but took this work, built upon it and presented it in a different manner. As the intent for the project was to develop an educational recycling game it was especially important to understand exactly what made a video game fun and engaging. This helps ensure those that play the game remain engaged for a long enough period of time to learn and take away the information the game is trying to convey. In addition, it was also beneficial to research the methods which are currently or have been used to provide some form of education on how to correctly recycle. This allowed the project to build upon previously attempted methods and the information they provide with the intent of hopefully creating a more effective and engaging form of education on the subject of recycling.

2.1 Video Game Design

When designing a video game, it is important to ensure the game is engaging for the player. This can be achieved in a variety of ways however, perhaps the simplest explanation is to design a video game with a fun and engaging game loop that rewards the player over time therefore, enticing them to continue to play the game.

2.1.1 Repetition. The difficulty in creating an engaging game loop is to design a game which is fundamentally repetitive while creating the illusion that it is not and therefore, does not feel so to the player. It is an almost impossible task to create a video game that is of a decent scale without some form of repetition as creating something that is entirely unique in every possible way throughout the entire experience would take a tremendous amount of time. Therefore, repetitive elements of video games should be designed with some thought. For example, the player could be doing largely the same thing throughout the entire game however, over time their character may grow such as gain new abilities that offer the player new options to tackle what is put against them. The main loop of the game remains the

same, but the player is given a new way to interact with it.

While repetition is commonly viewed in a negative way, it is not always as bad as it may appear. With repetition comes mastery. [4] Many players find it intriguing and fun to master the mechanics of a video game as it offers a sense of achievement. Whether this is to gain the highest score or beat the game in the shortest amount of time, it is important to remember that repetition is not always a negative and can, if implemented correctly, become a large factor in ensuring the player remains engaged with the game. As an example, a game which implements repetition extremely well is Rock Band developed by Harmonix. [5]

2.1.2 Effective Video Game Elements. Below is a list of common elements which help lead to an effective video game. [6]

- Interactivity

Ensure the player can interact with the game environment in a clear meaningful way. The player should have the ability to influence the game in some way.

- Clear goals

Ensure the player is aware of the reasoning behind what they are currently doing or working towards. This helps motivate players to continue playing the game.

- Well defined rules

Ensure the player understands the rules of the world they are interacting with. Outline what the player should do and how they should do this to achieve a particular goal.

- Narrative context

Narrative does not need to be complex however, the player should understand the reasoning behind the goals they are set to accomplish. Goals justified by narrative help engage the player within the world so that they continue to play.

- Control

The player should have the ability to influence the game world in some way.

- Ongoing feedback

Ensure the player receives some form of feedback when they carry out a particular action.

- Rewards

Ensure the player is rewarded in some way for completing a particular goal or carrying out a particular action.

Generally, incorporating these elements when designing a game will help to ensure it is, in some way, engaging for the player to interact with. This project will attempt to include these elements throughout development. However, when developing an educational game, some of the above points can be expanded upon.

- Interactivity

Players should be able to interact with the game in a way that engages and furthers their learning. For example, rehearsing memorable information.

- Clear goals

Goals should motivate the player to learn throughout the course of the game.

- Narrative context

Ensure the player understands what the game is attempting to teach them and why this information is important to absorb.

- Control

The player should have some control over the type of information they learn. For example, if the player does not understand a particular concept the game is trying to teach, allow the player to replay that information until they understand.

- Ongoing feedback

The game should provide the player with feedback on their performance throughout. If the player correctly absorbs the information the game is trying to teach them, then the game should relay this back to the player to boost confidence and encourage further learning.

- Rewards

The grand reward of educational games is the information the player learns however, they should be rewarded in some way within the game to keep the player engaged and encourage further learning.

- Problem solving

Encourage the player to think for themselves by allowing them to solve interactive problems and complete challenges presented by the game with the intent to further the learning of the player throughout the experience.

These elements are important to consider when building an educational game to ensure it remains engaging while also providing the player with the necessary information to effectively learn about a particular subject. It is extremely important in an educational game to motivate the player to learn throughout. The game should, of course, be fun and engaging but the ultimate goal is to ensure the player leaves the experience with a better understanding of the subject the game is trying to teach compared to what they previously understood prior to playing.

2.2 Recycling

Since the main goal of the project was to teach people how to recycle a variety of materials correctly, it was important to ensure that the information presented by the game was correct therefore, research into the UK recycling guidelines was vital.

In March 2017, a document was published by The Waste and Resources Action Programme (WRAP) which was later amended in August 2018. This document is an agreement from the recycling industry containing national recycling guidelines within the UK. [2] It outlines all of the necessary information needed to teach and inform people how to correctly recycle waste material, including all of the many nuances throughout the process.

WRAP works with governments, businesses and communities to deliver practical solutions to improve resource efficiency. [7] The recycling guidelines document was created in collaboration with the UK recycling industry, so it was extremely useful throughout the development of the project as it helped ensure the information delivered within the game was correct.

The document states the following: [2]

- 66% of households express uncertainty about how to correctly dispose of one or more items.
- 49% of UK households dispose of one or more items in the residual bin that are collected for recycling in their area.
- 68% of UK households add one or more items to their recycling collection that are not accepted locally.
- Only 12% of households do not put any items in the residual bin that could be recycled, nor do they put any items in the recycling that are not accepted.

The above statistics were collected from the WRAP Recycling Tracker survey carried out in 2016. [2] This helps highlight the serious issues faced leading to the goal of effectively recycling waste material within the UK. It shows that a lack of knowledge within households is one of the main issues as only 12% of households correctly sort their waste material into the correct bins while 66% are uncertain of how to dispose of particular items. Clearly, many households are willing and actively trying to recycle waste material which is of course great, however, the issue is the lack of recycling knowledge many of these households have as they do not fully understand which materials are or are not recyclable. This highlights the severe need for some accessible and engaging way to teach a large number of households how to correctly recycle their waste as many are happy to do so but have not been educated on the matter or do not know where to find the most reliable sources to learn.

2.2 Recycling Games

After researching into both game design concepts and recycling it was important to try to find some already existing educational recycling games to see how they attempted to teach the player with the intention that the game produced by this project would hopefully improve upon the concepts presented in these games.

2.2.1 Clean and Green Game. The first notable game was a recycling game called “Clean and Green” created by Cambridge English Online for a British Council children’s learning website. [8]

Fig. 1. “Clean and Green” game. [8]

The game entails a messy child’s room that the player must clear up by placing the rubbish into the correct recycle bins under the categories of “Paper”, “Plastic and Metal”, “Glass” and “Compost.” It is a very simple game clearly aimed at young children as there is not much depth to it. The player must simply click on an item and then click on the corresponding recycle bin for that item. If the player is correct a “Good Job” message appears whereas if they are incorrect a “Rubbish” message appears in which the player can then choose another recycling bin to place the item into. The different recycling bin categories was something that was incorporated into the River Recycle game, however, the game changed the categories slightly. The “Paper” and “Glass” categories remain while the “Plastic and Metal” category from the “Clean and Green” game was separated into two different categories for the River Recycle game with an additional “General Waste” bin for non-recyclable items. The compost bin was not included in the River Recycle game as it was decided that it would be more important to focus the game specifically on recycling rubbish rather than composting.

One of the main issues with the “Clean and Green” game, even though it is for children, is that there is no penalty for choosing the incorrect recycle bin which means the player can simply guess which recycling bin the item should be placed into until they are correct. This could be an issue because it could encourage the player to only guess which is the correct recycling bin rather than trying to learn the information the game is trying to teach. As teaching the player information is the main goal of an educational game, it is important to ensure the player is learning and understanding the content presented to them. In addition, the game does not directly give any recycling information to the player, instead it assumes the player will use their own prior knowledge or learn through trial and error.

Included on the website in which the game is hosted is a comment section filled with comments from the children who played through the game. In general, the children appeared to like the game however many described the game as a little bit boring and too easy. These comments were kept in mind throughout the process of this project as it was important to ensure the River Recycle game was much more engaging than this example by offering more of a challenging experience. This was achieved by adding the possibility of failure as the player must restart a level if they lose all of their lives. The addition of this simple mechanic allows the River Recycle game to not only offer a challenge to the player but also, it encourages the player to retain the information that the game presents to them so that they can complete levels successfully.

2.2.2 Recycling Waste Game. The next game found was a game called “Recycling Waste” created by Turtle Diary [9] which provide “Fun Educational and Online Games for Kids.” [10]

Fig. 2. “Recycling Waste” game. [9]

This game entails a conveyer belt of items which are presented to the player in groups of five. The player must then drag and drop the items into one of the three bins labelled “Compost Bin”, “Recycle Bin” and “Trash Bin.” Similar to the previous game, this game is very simple and does not provide much challenge to the player since there is no failure state, rather just a counter showing how many items the player attempted to place into the incorrect bin. Also, like the previously discussed game, this game does not

present the player with much educational information detailing which bins any of the items should be placed into but rather relies on prior knowledge and trial and error to succeed. Since this was a recurring issue throughout these educational games it was important that River Recycle, the game created for this project, presented the player with some form of tutorial that allowed them to learn which recycling bins items should be placed into. This meant that players would hopefully absorb the information given to them to allow them to successfully complete a level rather than succeeding through guessing or trial and error.

Another problem with the "Recycling Waste" game is that some items are categorised incorrectly. For example, the toilet rolls in the game should be placed into the recycle bin. However, only toilet roll holders can be recycled, the paper should be placed into a general waste bin. This indicated that it should be a top priority throughout the entirety of this project to ensure that all information presented within the game is completely correct to ensure misinformation is not being spread. This only further cemented the need to strictly adhere to the guidelines presented in the WRAP document discussed in the previous section.

The conveyer belt in "Recycling Waste" is similar to the river implemented in this project however, the conveyer belt only really serves an aesthetic purpose since it simply carries on the next set of items before coming to a halt. The river in River Recycle builds upon this by offering gameplay mechanics in which the items continue to move down the river until either the player correctly sorts them into a bin or the item reaches the end of the river in which the player then loses one of their lives. This should make River Recycle more fun and engaging than "Recycling Waste" as it effectively adds a timer so the player only has the time it takes for the item to reach the end of the river to sort it, otherwise they receive a penalty. Hopefully, this encourages the player to more effectively retain the information presented within the game so that they can sort items before they reach the end of the river.

2.2.3 Recycle Roundup Game. The final game found was called "Recycle Roundup" on the National Geographic Kids website. [11] The idea of the game is to clean up the park by sorting the various items people have thrown away.

Fig. 3. "Recycle Roundup" game. [11]

The player must move the character around the screen using the mouse and click to pick up rubbish falling from the sky or on the ground. The player must then move the character over to the corresponding bin for the item they have picked up and click on the bin to place the held item into it.

There are several issues with this game. The first being that the character is fairly difficult to control as it follows the mouse cursor. This may seem fine; however, the game is not two-dimensional so moving the mouse to the top of the screen moves the character further back into the distance while moving the mouse to the bottom brings the character closer. Additionally, the items that the player must sort fall from the sky at various distances along the z-axis, but it is extremely difficult to tell exactly where they will land. This is a real issue for an educational game as it must not be frustrating otherwise the player will simply stop playing. The controls should be simple and make sense according to the style of the game. In this game, it does not make sense to have the character follow the mouse. Perhaps it would be improved if the character were controlled by the mouse rather than following the cursor. This issue was considered throughout the development of the project to ensure the controls of the game were suitable and made sense. Because of this, River Recycle only uses the mouse and the mouse left click button to pick up and deposit items from the river which makes sense in the context of the game and is also very accessible and feels natural for the player.

Another issue with “Recycle Roundup” is that it only teaches the player the necessary information to succeed at the game through the use of a list of instructions accessed on the main menu. This may cause issues as many people may simply not read these instructions and therefore, not understand how to properly play the game which may have a negative impact on the player’s learning. Due to this, it was important to ensure that the information needed to complete a level in River Recycle was included in interactive tutorials within the game that play out in real time. This allows the player to learn while they play the game rather than reading an instruction manual which, hopefully, will help them retain the information more easily and for longer.

3 DESIGN AND IMPLEMENTATION

The design and implementation stage of the project was the most demanding section and took most of the time available. It was important to design and implement an effective recycling game that improved upon the already existing games described above in meaningful ways in order to give the project purpose. If the project copied the games already available, it would effectively be a waste of time, so it was vital that it built upon them. This section describes the design ideas and implementation stages decided upon and carried out throughout the development of the project.

3.1 Design

The main idea throughout the project was to ensure the game was educational but also fun and engaging so it was important to find a good balance between the amount of information presented to the player and the time they spend playing the game. This is because it is important that the player does not get bored of the game too quickly otherwise, they may not learn and retain the information the game is trying to teach them. Thus, the choice was made to only present the most important information within the game to the player. This means that the tutorials in the game, which offer the player useful recycling information, only contain the information most vital to further their learning. This was important not only to help prevent lack of interest, as throwing a huge amount of information at the player would quickly get boring, but also to ensure the game was as accessible as possible regardless of age or prior knowledge of recycling. However, for players that wish the learn even more on the subject, a link to the WRAP recycling guidelines booklet [2] is provided within the game.

3.1.1 Bin Categories and Colours. The recycling bins used in the game are categorised into “General Waste”, “Plastic”, “Paper”, “Metal” and “Glass.” These categories were chosen as they correspond to the most common categories used for recycling bins, at least within the United Kingdom. In addition to this, the bins in the game were given different colours as this should allow the player to distinguish between each bin much more easily without needing to read the label on the bin. Furthermore, the colours chosen for the recycling bins in the game also correspond with the commonly used real-life colours for those recycling bins in the United Kingdom. The “Recycling Bins” website states that general waste bins are usually black, plastic bins are usually red, paper bins are usually blue, metal bins are usually grey and glass bins are usually similar to teal. [12] This was an important design decision as the main point of the educational game is to teach players how they should be correctly recycling rubbish in their day to day lives. So, designing the bins to be of a familiar colour to their real-life counterparts hopefully allows players to bridge the gap between the game and real life so that they can more easily make use of the information they learn. The only deviation from this was that the general waste bin in the game is white rather than black as it better fits the aesthetic of the game since the bins all have a black outline.

3.1.2 Problem Solving. It was important to include problem solving elements within the game as this one of the game design elements described previously in the research section of this paper. Problem solving elements allow the player to think for themselves which, hopefully, results in more effective retention of the information presented to them. Luckily, the basic idea of the game is a problem-solving based game. The player must figure out which bin each item should be placed into otherwise they lose one of their five lives. This encourages the player to think carefully and recall the information the game has taught them so that their decision is correct, and they can complete the level successfully. In addition to this, at a certain point in the game the bin locations are switched around. The reason for this is to ensure

the player is not simply getting used to placing particular items into particular bins based on muscle memory but rather, they are actively thinking about where they are placing each item based on what they have learnt.

3.1.3 Recycle Items. The decision of which items to include in the game that the player must sort was an important one. The items needed to be easily distinguishable by the player so that they were not confused by what any particular item may be. In addition, some of the items had to be obscure enough to ensure that the player did not immediately know exactly which bin the item should be placed into since the game is trying to educate the player on recycling practices they may not know of.

It was decided that several items would be needed for each bin category to ensure there was a good balance of where the player was placing items. For example, the game would not be all that interesting if most of the of the items had to be placed into the "Paper" recycle bin. This is where the first major issue arose. It proved difficult finding assets for recyclable items that fit the intended aesthetic of the game. In addition, assets from different creators have different art styles and therefore, do not look like they belong together in the same game. After a good amount of time searching, an asset pack of recycle items was found. [13] These items fit the aesthetic of the game perfectly, provided a good balance of items for each category and were all created by a single author, so they had a consistent art style.

In addition to recycle items, it was decided that there should be items that do not fit into any of the bin categories and must therefore, be left in the river. Because of this, the game includes fish which swim along the bottom of the river. They can be picked up by the player but if placed into any of the bins they lose one of their lives. The only issue with the fish asset found [14] is that it is not of same art style so looks slightly out of place compared to the other recycle items, however, the added gameplay mechanic to challenge the player was more important.

3.1.4 The River and the Environments. It was decided that more than one environment would be included within the game. The park setting would be used for the first few levels before the setting would change to a city environment.

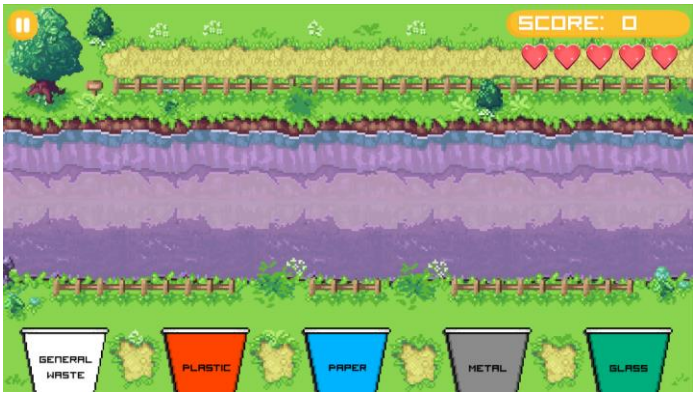

Fig. 4. The park setting in River Recycle

The park setting was created using a single nature-based asset pack. [15] This pack made it extremely easy to create an aesthetically pleasing park scene. It was important throughout the design process to ensure the game was not boring to look at as a bland and boring looking game could result in the player losing interest much more quickly than they otherwise would.

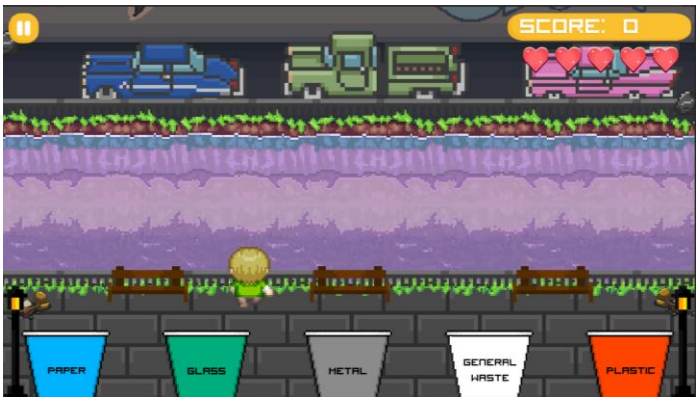

Fig. 5. The city setting in River Recycle

Figure 5 shows the city setting in River Recycle. While it may not be as aesthetically pleasing compared to the park setting, it is still interesting to look at. The city setting does not only differ from the park setting visually but also mechanically. This is because the player will notice a larger number "General Waste" based items indicating that the city is not very recycle friendly. This environmental world building will hopefully further engage the player in the game and encourage them to continue learning the information it presents. Figure 5 also shows the bin position changes compared to the park setting in figure 4 which, as described previously, will ensure the player is thinking carefully about the bin they are

placing each material rather than doing so by muscle memory.

3.1.5 The Game Over State. The way in which the game over state would occur in River Recycle was torn between two options for a large portion of the development time. The first option was that the items would pile up at the end of the river and once a specified number of items had piled up, the level would fail, and the player would need to restart. The second option was that the player would have a specified number of lives in which they would lose a life if they placed an item incorrectly into a bin or if an item reached the end of the river and moved off the screen. In the end, the lives option was taken forward and implemented into the game. The reason for this was because the pile up system would not penalise the player for choosing an incorrect bin and therefore, it would allow them to simply guess which bin the item belonged to rather than recalling the information the game taught them. This was an issue brought up previously in the research section and as the game for this project intended to improve upon the issues of already available educational recycling games it was important for the player to be penalised in some way for choosing incorrectly. Additionally, when using the pile up system the scene would become extremely crowded with items which could have become too much for some players to handle. As already stated, the game intends to be as accessible as possible so it made more sense to use the lives based system as when an item reaches the end of the river, the player loses a life and the item is removed from the scene which helps prevent the scene from becoming too cluttered.

3.2 Implementation

The idea at the implementation stage of the project was to implement most of the core mechanics as early as possible so that the remainder of the game could be built around them and additionally to ensure there would be enough time to remove any of the inevitable bugs associated with the core mechanics.

3.2.1 Technology. The project game was built using the Unity game engine [16] for a number of reasons. Perhaps most importantly, Unity allows projects to be built using WebGL. WebGL is a JavaScript API for rendering 2D and 3D graphics within any compatible web browser without the use of any plugins. [17] This has allowed the game to be hosted on a website which is accessible to most modern hardware and browsers compatible with HTML5. This is extremely useful as it results in almost no barrier to entry which makes the game accessible to almost anyone.

The game has been hosted through GitHub Pages. [18] GitHub Pages is a site hosting service that takes HTML, CSS and JavaScript files straight from a repository on GitHub. [19] This has made it extremely easy to make the game accessible online as the project was simply built in Unity on the WebGL platform then the build was pushed to a GitHub repository setup for GitHub pages.

Another reason the Unity engine was used to develop the game was because it is packed with useful features and functionality that allow developers to create a large portion of their games without writing much code. This makes development a lot faster and more straight forward, however, when a developer needs to add more specific or advanced functionality to a game that is not already available in Unity, it also offers scripting which allows developers to implement their own functionality using code written in C#. C# is an elegant and type-safe object-oriented language [20] that bares many similarities to languages like the more commonly recognised C++. Scripts in Unity are extremely useful as they can be as simple or as complex as the developer wishes. The developer simply writes the code for whatever functionality they wish to implement, from which they can then attach the script to any Game Object in the Unity editor. Furthermore, the developer can specify “public” variables inside a script which can then be accessed and changed directly from inside the Unity editor. These variables can even be assigned ranges if needed which is reflected in the Unity editor so that variables can only be set within the specified range. The scripting functionality available in Unity was used extensively throughout this project to implement much of the functionality described below.

Before detailing the implementation process it is important to mention the “prefabs” feature available to Unity. Unity’s prefab system allows the developer to create, configure and store a Game Object complete with all its components, values and children as a reusable asset. [21] This was an extremely useful feature throughout the development of this project as it allowed easy and quick duplication of assets between scenes. For example, for this project a

"LevelFoundation" prefab was created which included the river game object, the game objects for the bins and all of the in-game user interface. This meant that, when creating a new scene, all of this functionality was available by simply dragging and dropping the prefab from the Unity project explorer into the scene which meant that all of the core functionality of the game only had to be setup once.

3.2.2 Core Mechanics. The first step in the development process of the game was to implement the core mechanics needed for the game to correctly function. It was important to do this as early as possible in the implementation stage to ensure there was enough time to fix any potential unforeseen issues with the main game mechanics as, without these mechanics functioning correctly, the game would not meet the proposed specification. The core mechanics consisted of five main components which were:

- The player must be able to click and drag items using the mouse.
- The player must be able to deposit items into an appropriate bin.
- The items must flow down a river throughout the game.
- The game must have a "game over" state.
- The game must have a "level complete" state.

Implementing all of the above features would ensure the game was in some way playable. Of course, it would not be a very effective educational game, but this functionality was absolutely necessary to implement before any of the educational elements could be added.

3.2.2.1 The Click and Drag System. The core mechanic implementation began with the click and drag system which would allow the player to use their mouse to hold down the left mouse button over an item to pick it up and then move it around by moving the mouse while continuing to hold the left mouse button. Letting go of the button would result in the item being dropped.

This system at first, appeared fairly simple to implement however, it quickly became apparent that the complexity of this system had been underestimated. The initial implementation was straight forward as Unity offers functions such as OnMouseDown, OnMouseDrag and OnMouseUp which are fired when the left mouse button is clicked on an object, when the button is held down on an object and when the button is released on an object respectively. This made it very simple to create a script which allowed objects to be picked up, dragged around the scene and dropped. The complexity of the system arose when thinking about the kind of erroneous actions a player could potentially carry out while holding an item. For example, if the item could be dropped wherever the player released the left mouse button, the player could drop an item outside of the river or place items back at the beginning of the river which would trivialise the challenge the game presented. To fix this issue, an original position was stored for the object which fired the OnMouseDown method so that when the player releases the left mouse button the object can be reset back to its original position in the river, assuming it was not deposited into one of the bins. This ensured items were always in their intended positions and the player could not tamper with them erroneously.

There was another issue that became apparent with the click and drag system. The items had colliders associated with them to allow them to collide with each other in the river so that they would separate if a collision occurred. The issue with this was that if the player picked up an item, it would continue to collide with the other items in the river which could potentially allow the player to push items outside of the river or even outside of the scene. This was a serious issue as it effectively broke the game for example, the level complete state, which will be described later in this section, checks the number of recycle items in the scene and if the value becomes zero the level is complete. If the player were to push an item outside of the scene it would never be destroyed therefore, the number of recycle items would never be zero and the level could never be completed. To solve this issue, it made sense to prevent items held by the player from colliding with items in the river. To implement this, Unity offers a feature known as collision layers [22] which allows the developer to assign any objects a layer and then specify which layers can collide or cannot collide with each other. Using this, two layers were setup, one for river items and one for held items. As an item is picked up by the player it is assigned to the

“heldItem” layer and when it is dropped it is reassigned to the “riverItem” layer. These layers do not detect collisions between each other which prevents the player from moving the items in the river around using the item they are currently holding.

Another issue with the click and drag system was that items would occasionally jitter and rotate while being held. This was due to the fact that items had a RigidBody2D component associated with them which would cause gravity and forces to affect them as they moved through the scene. Even though, with this issue, the game would still function correctly, it would partially ruin the immersion as items should generally stay in place when being held. Luckily, this was a fairly simple fix as Unity offers functions which allow the developer to freeze and unfreeze the position and rotation of a RigidBody2D at any point. These method were easily implemented into the click and drag system so that items would be frozen on being picked up and then unfrozen on being dropped.

Easily the biggest problem throughout the development of the click and drag system was an issue that took a fair amount of time to solve as it was not apparent exactly what was causing it. This issue was that items would sometimes not be picked up when the left mouse button was held while moving the mouse over them. This was because, for some unknown reason, the OnMouseDown function was not firing. The reason this issue took so long to solve was because it was fairly difficult to consistently replicate as sometimes the system would work flawlessly and items would be picked up as intended while on other occasions it would not work at all. It was initially thought that the issue would only occur on items with small colliders as trying to pick up smaller items would more regularly cause the issue to occur. However, it was later noticed that it also occurred on some of the larger items, so this was not the problem. It was then noticed that the issue would only occur when the number of recycle items in the scene increased to above four as if there were four or less items in the scene, the OnMouseDown function would fire every time without error. This was, of course, an issue as only four items in the scene at a time would not result in a very interesting or challenging experience for the player. Finally, after further research into the OnMouseDown function it was noticed that the issue was due to the z-position of the items. This is because the OnMouseDown function works by firing the function for the first object detected under the mouse cursor calculated using a raycast into the scene. This issue was occurring because, even though the game was 2D, the Unity editor still builds the game in a 3D environment but simply prevents the z-axis value from affecting the position of objects. However, even when developing in 2D, the engine still relies on the z-axis position of an object when determining the closest object to the camera. This meant that when the items were not being picked up because the OnMouseDown function was not firing, it was due to some other object being nearer to the camera along the z-axis and therefore, the raycast was hitting this object first. Solving this problem was extremely simple now that the cause of it had been discovered. All that needed to be changed was changing the value of the z-axis for recycle items to be at -9. This was because the camera was positioned at -10 along the z-axis so simply moving the recycle items closer to the camera than any other object in the scene fixed the problem entirely.

3.2.2.2 The Deposit System. The system for depositing items was fairly simple to implement now that the click and drag system was implemented. The basic idea of the deposit system was, if a held item is dropped on top of one of the bins, if the item is of the correct category it will be removed, and the player will be rewarded points. However, if the item is incorrect, the item will be reset back to its position in the river and a life will be deducted from the player. This functionality was implemented easily using Unity’s tag system [23] which can be assigned to game objects. Each recycle item was given a tag that corresponded to the category of bin they belonged with, for example, the cardboard box was given the tag “Paper.” This allowed a simple script to be created which, on a collision between a recycle item and a bin, would check if the tag given to that item was equal to the tag of the bin. If true, then the item must be correct, and the player’s score increases.

3.2.2.3 The River. The river system was another of the core mechanics required for the game. This system was incredibly easy to implement as all of the functionality required for the river already existed in the Unity editor. Unity offers a 2D buoyancy system [24] which can be attached to a game object to define and simulate fluid behaviour such as floating, drag and flow. This buoyancy component was added to a

game object with the imported river sprite attached to it which then allowed the river parameters to be setup. The system acts as a simple collider that affects any physics-based component inside the collider such as the RigidBody2D component which is attached to all of the recycle items in the game. This meant that, upon setting the parameters for the buoyancy component like the surface level of the water, its density and speed of the flow, any recycle item that made contact with the buoyancy collider would automatically begin bobbing along in the direction specified therefore, simulating a river.

One issue with this system was caused if the items in the river had too high of a mass. This would result in the item falling through the river and out of the boundaries of the scene which, of course, was not the intended functionality. To solve this a collider was placed at the bottom of the river acting like an invisible wall which would prevent objects from falling through, effectively representing the riverbed.

It was also important that, once items reached the end of the river and moved off screen, they were deleted from the level and the player health bar was updated accordingly. To do this, a collider was added at the end of the river outside of the playable scene, not visible to the player. If any recycle items collide with this collider, they are deleted from the level and the player health is reduced by one. This was important for the game over state and level complete state, which will be described next, to function correctly.

3.2.2.4 The Game Over State. To allow the game to offer some form of challenge to the player, as described in the previous section, it was important to penalise the player if they placed an item into the incorrect bin which would eventually lead to a game over state in which the player would then need to restart the level. This state was implemented based on a health system in which the player is given five lives at the beginning of the level. The player can lose a life by incorrectly placing an item into a bin or if the item reaches the end of the river. Upon losing all of their lives, the game over state is triggered which presents the player with the option to return back to the main menu or restart the level. As already discussed, it was important to include this system in the game to prevent the player from guessing which bin each item should be placed into as, if the player were to do this, they would not be learning the information presented by the game which is, of course, the main goal of an educational game.

3.2.2.5 The Level Complete State. A state needed to be created that triggered when the player successfully sorted all items in the river, or when all items had reached the end of the river and the player still had some of their lives remaining. This state was quite simple to implement. At the start of the level the game counts how many recycle items are in that level. When the player deposits an item correctly the counter is decremented by one. Additionally, when an item reaches the end of the river and is removed from the scene, the counter is also decremented by one. Once the counter reached zero the level complete state is triggered which allows the player to either continue on to the next level or return to the main menu. It was important that this state was implemented into the game as otherwise, there would be no way to allow the player to continue the game. Additionally, it was implemented early in the implementation process to ensure there was enough time to get the state functioning correctly.

3.2.3 User Interface. Following the implementation of the core mechanics for the River Recycle game it was important to next create a user interface (UI) to allow the player to see their score and lives as well as other functionality such as pausing the game.

Unity makes user interface very simple to implement through the use of canvas game objects. [25] The developer creates a canvas and can then create various UI elements as children of that canvas. These elements could include text boxes, buttons, sliders and many more. The developer can then tweak and control the parameters of the UI elements to suit their needs.

The first section of UI that was created was the in-game elements which included a score tracker and a health bar. These were easy enough to implement using the Unity editor. For the score tracker a simple script was attached which updates the element whenever the player score changes so that the UI reflects the progress of the player. Additionally, for the health bar, a script was attached which updates the health bar every time the player loses a life with an aesthetically pleasing animation where one of the player's hearts on the health bar slowly drops down and out of the bottom of the screen. This animation helps ensure the player notices when they lose a life

so that they do not need to constantly glance at the top right of their screen.

Next, it was important to include a pause button, not only so that the player can pause and resume the game at any time, but also to allow the player to go back to the main menu at any time or access the settings menu. The pause menu is simple, on clicking the pause button the main game loop freezes and the player is presented with four options. Resume, settings, main menu and quit. The settings menu includes a volume slider which allows the player to adjust the master volume for the game. This volume slider was more difficult than predicted to implement as if would not save the current volume of the game so when the player clicked the settings menu, even if the master volume was at fifty percent, the volume slider would display at one-hundred percent. To fix this issue the PlayerPrefs class was used which stores and accesses player preferences between game sessions. [26] This is the same class that would be used to save player data in Unity as the data stored using the PlayerPrefs class is persistent so, is not deleted on quitting the game. The volume slider script uses this class to retrieve and display the current volume level whenever the settings menu is opened. In addition, it also saves the new volume level if the player chooses to adjust the slider. This was an important feature to include in the game as some players may have issues with their hearing and therefore, may want to increase the volume. On the other hand, many players may want to decrease the volume if the game is too loud and providing this option to do so for them is a nice quality of life feature.

3.2.4 Audio Manager. It was important to include audio within the game to further immerse the player in the world which would, hopefully, help encourage them to continue playing the game. Unity offers a system for playing audio by using audio source components which can be attached to game objects. This allows for specific objects to emit whatever sound the developer intends which can be extremely useful in 3D games as the object itself emits the played sound which gives spatial awareness to the player. Since River Recycle is a 2D game there was no need to worry about spatial awareness, so it did not matter which object was playing any particular sound. However, the audio implementation was still fairly difficult to manage as certain sounds had to be played in many situations. Due to this, it was decided to create an audio manager which would handle all of the game audio by allowing any game object to call the PlaySound function in the audio manager to play any sound available in the game. The audio manager simply stores an array of sound clips so that when the PlaySound function is called, the audio manager can search the array for the sound file based on the tag entered in the PlaySound function parameter. This initially took some time to figure out and setup, however, was worth the time investment as it made the implementation of audio into the game much easier in the long run. Additionally, using Unity's prefab feature, a prefab of the audio manager was made so that it could be easily imported into every level in the game which was extremely useful.

3.2.5 Tutorial Manager. The tutorials in the game were a very important section of development. This is because the tutorials are the way in which the game teaches the player the information they need to know about recycling. They are the most educational parts of the game, so it was extremely important that they were engaging to help ensure the player absorbed the information presented. In order to make the tutorials engaging it was decided that they must be interactive rather than presenting the player with a wall of text as, from experience researching the other recycling games described in the previous section, huge chunks of text are not engaging to the player and therefore, could result in the player not reading them.

To develop the tutorials so that they were interactive for the player but also quick and easy to integrate into the game, a tutorial manager was created. This tutorial manager would be imported into each level. In order to add a tutorial to the level, first, a UI text element would be created inside the tutorial manager and the text for the tutorial would be written. Then the tutorial text would be given a tag which associated it with the particular type of item that would trigger the tutorial. The tutorial would then begin once the corresponding item hit the tutorial manager collider at the start of the river. For example, the first item in level one is a cardboard box which is tagged as a "Paper" item. So, the tutorial was created for paper-based items in the same way as described above which meant that once the cardboard box triggered the tutorial manager collider, the paper-based tutorial would begin. As stated, the tutorials were developed to be interactive so once the tutorial has been triggered all of the items

in the river are paused. The player can then pick up the item and attempt to place the item into one of the available bins. Once the item has been placed into the correct bin, the tutorial will close, and the game will resume. These interactive tutorials are much more engaging to the player than a wall of text and their implementation was much easier due to the creation of the tutorial manager. Hopefully, they will encourage the player to learn and continue to play the game.

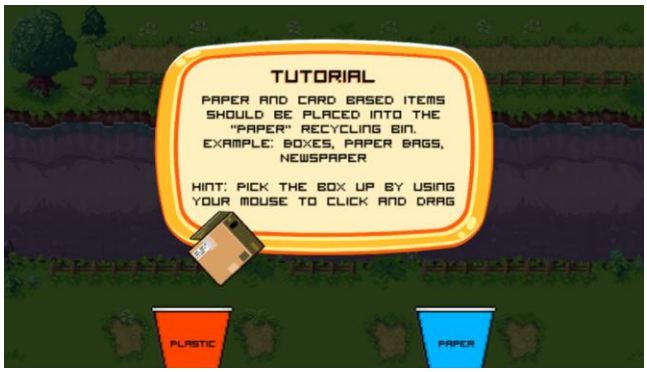

Fig. 6. The "Paper" tutorial in River Recycle, showing the item above the tutorial layer.

There were, of course, issues while creating the tutorial manager. One of the main issues was, since the tutorials were UI elements, they would always be placed above the items in the river. This meant that when the player picked up an item and dragged it over the tutorial text, it would disappear under the text panel. To solve this, the UI canvas needed to be changed from an overlay canvas to a camera-based canvas. This change places the canvas at a given distance in front of the specified camera [25] which allows the canvas to be given a sorting order which is lower than that of held recycle items, allowing them to be dragged over the top of the tutorial panel as displayed in figure 6.

3.2.6 Mischievous Characters. To make the game more interesting in the later levels a mechanic is introduced in which a character is standing near the edge of the riverbank and will occasionally throw items into the river that the player must then sort. Additionally, the player can click on the character with their left mouse button to stop him from throwing items for a short amount of time. This adds another layer of challenge to the game to test the player's knowledge on what they have learnt throughout the game by requiring them to think quickly but carefully.

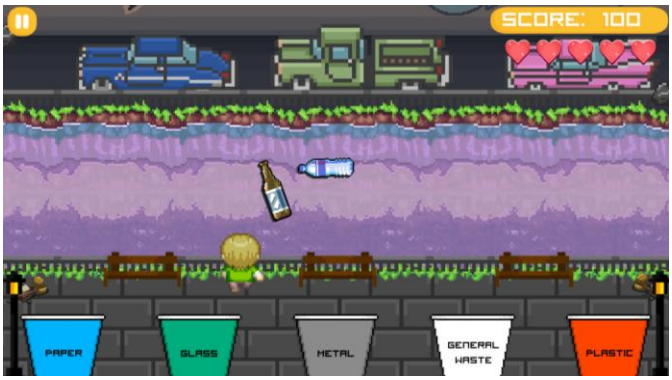

Fig. 7. The mischievous character throwing items into the river.

This functionality is implemented into the game through a script. The script, attached to the mischievous character, will cause the character to begin throwing items into the river after a set amount of time and at a frequency of a set amount. For example, the character can begin throwing items after ten seconds and throw an item every two seconds. Once the player clicks on the character to stop them from throwing items, the character will then not throw any items for ten seconds. This pressures the player into sorting the items as fast as possible however, the health bar system ensure that the player must still think about the bin in which they are placing each item to ensure the chosen bin is correct.

3.2.7 Endless Level. Once the player has completed all of the game levels, they are presented with an endless level. This level is intended to challenge them by making them use all of the information they have learnt throughout the game to sort as many items as they can for as long as possible. The level will continue forever, and it will get progressively more difficult over time.

The functionality in this level was created using a script which ensures at the beginning of the level there is a minimum of four items and will spawn a new item every time an item is deposited. Upon the player correctly depositing a specified number of items, the level difficulty will increase by increasing the minimum number of items allowed in the level as well as by slowly increasing the speed of the flow of water in the river so the items will begin to move faster and faster as the player continues to correctly deposit items. When the player fails the level by losing all of their lives, they will be shown their score

and, if their score is higher than all of their previous scores, they will be informed that they have achieved a new high score. This will, hopefully, engage the player further into the game so that they continue playing and continue to improve their knowledge on recycling which should help players retain the information they learn for longer.

There were issues when implementing this level. For example, it was difficult deciding how to balance the difficulty for the level. If the difficulty increased too quickly, the player may get bored of constantly losing however, if the difficulty increased too slowly, the player may also become bored as the game would not be engaging or challenging to them. In the end, I think there was a good balance achieved for the rate at which the difficulty increases.

3.2.8 Dropped Features. Throughout the development there were two main features that were dropped. These were fluid simulation and narrative.

Fluid simulation was dropped from the game for several reasons. Firstly, fluid simulation can be fairly performance intensive which, for a game intended to run in a browser and accessible to as many as possible regardless of their hardware, would not have been the best idea. Furthermore, the Unity buoyancy system described above gave the game all of the functionality needed which therefore, made a fluid simulation system almost redundant in this case. It would, of course, have been interesting and challenging to implement however, since the buoyancy system was so suitable for this project, it did not make sense to not make use of it.

The second dropped feature was a narrative. It was intended for the game to have some form of narrative throughout the levels to further engage the player and encourage them to continue playing. This feature, ultimately, was dropped due to time constraints. It would have been a great addition to the game to encourage the player to further their learning however, the features that were implemented were, arguably, more important which therefore, resulted in the narrative being dropped.

4 RESULTS AND EVALUATION

From the beginning of this project the general intention was to create an educational game to teach people about recycling a variety of items. It was specified that this would be achieved by creating a game in which the player would be tasked with sorting items as they flow down a river while making use of the knowledge they have gained from the information presented within the game.

4.1 Did the project meet the specification?

The core functionality is all present in the game to a fairly high standard. The main game loop revolves around items flowing down a river in which the player sorts by placing them into the recycling bin most appropriate to their material. Throughout the game the player is presented with a series of tutorials which give the player the necessary information to successfully sort all of the items available. This fits the specification of the educational game that was proposed at the beginning of this project so, in terms of the core fundamentals, the game absolutely meets the criteria.

There is also additional functionality that provides the player with a more fun and engaging experience to help encourage them to further their learning by continuing the play the game. This is achieved through item variety which provides a good balance of items for each recyclable category. Additionally, the change of level scenery offers the player something fresh to help ensure they remain interested in playing. Furthermore, the endless level offers a challenge to the player by slowly increasing the difficulty while tasking them to achieve the highest score possible. These added features meet the original specification by working to make the game a more fun and engaging experience for players so that, even though they are playing an educational game, they hopefully still have a lot of fun doing so.

The intention was to create the game with the goal of it being as accessible as possible. The project meets this specification by being available to run within a browser on a large number of hardware configurations. The game also performs well within the browser. The PC that was used to primarily develop the game which has an Nvidia GTX 1070 graphics card can easily run the game achieving over 144 frames per second in the browser. Of course, this is a fairly powerful graphics card however, upon testing the game using an old laptop with Intel HD graphics the game performed between 50 and 60 frames per second which is perfectly acceptable for a browser-based game. This shows that the project game has met the performance accessibility

specification that was detailed at the beginning of the project which enables the game to be played by a large number of people regardless of the hardware capabilities of their device.

In addition to accessibility through low hardware requirements, the specification also called for an accessible educational game suitable for almost all ages. The project has tried to meet this by being as easy to understand and simple to play as possible. The basic idea of the game is very simple and can be grasped easily by almost anyone while the interactive tutorial system in the game teaches the player how to understand the more complicated mechanics. This should allow anyone, regardless of age, to play the game and to find it fun and engaging while hopefully learning something about recycling in the process.

5 CONCLUSIONS

Overall, the project completes what it set out to do and there are several key features which have made this so. The game loop is simple to understand whilst also being fun and engaging. The interactive tutorial system keeps the player engaged during the slower sections of the game where they are expected to learn the information presented to them. The endless level provides replay value to the player which offers them a way to continuously improve. Finally, the accessibility ensures as many as possible can experience the game regardless of their age and hardware configurations. Because of these features the project has, for the most part, succeeded in creating the educational game that was proposed in the specification.

5.1 The Project

The project, as a whole, went fairly well. The research process was very interesting as it was eye opening when reading through the recycling guidelines document published by WRAP. [2] Learning about all of the nuances of what can and cannot be recycled was extremely useful information that can be carried forward and used in day to day life.

The implementation stage of the project was fun and interesting. Learning how to use a commonly used engine within the games industry was an exciting experience that will undoubtably come in handy in the future. In addition, the challenges presented while developing the game for the project were interesting and fun to solve and, luckily, never so difficult that they became stressful or frustrating.

Finally, the game that resulted from this project is indeed quite fun and engaging to play, especially the endless level which can become quite addictive. Overall, the project seems to be a success.

5.2 Future Work

Of course, not everything is perfect and there is still a lot that could be improved with River Recycle. If this project were to be worked on further, it would be possible to implement a narrative throughout the game. This would further engage the player within the game world by offering them a reason to play through all of the levels other than to further their own learning. Keeping the player engaged for as long as possible is extremely important for an educational video game and implementing a narrative would go a long way towards doing that.

Another way to improve the game would be to include more of the information present in the WRAP document [2] as a lot of the content in the document was not included in the game since there is a large amount of it. Including more of this content would only further educate any players of the game by providing them with more information on how to correctly recycle their materials.

Better rewards for the player could also be implemented in the future. This could include features such as allowing the player to collect medals or achievements for completing challenges within the game, for example, “successfully recycle 50 items.” Additionally, the player could be rewarded by unlocking new items to sort. Further rewards would be a great addition to implement into the game as rewards offer the player an incentive to continue playing and helps them feel as if their time is not being wasted.

Finally, level selection and difficulty selection would be good additions to the game. Level selection would allow the player to jump to any previously completed level if they wished to play that level again rather than needing to play through all of the previous levels first as it is currently. Difficulty selection would allow the player to fine tune their experience with the game which would increase the games’ accessibility by allowing those who may think it is too difficult to reduce the difficulty while those who want more of a

challenge could increase the difficulty. These features would be a great implementation for the future since they would offer the player more choice and overall, would make the game more fun and engaging.

5.3 To Conclude

To conclude, the end goal of the project has been met which was to create an accessible and engaging educational game teaching the player how to recycle a variety of items. The project was extremely fun, and a lot of useful experience was gained throughout that can be taken forward into the future. Hopefully, the result of this project, River Recycle, can help at least one person learn a little bit more about recycling.

REFERENCES

[1] Michael Marshall. How the UK is moving towards recycling most of its plastic waste. (March 2019). Retrieved June 10 2020 from https://www.telegraph.co.uk/business/reducing-plastic-waste/uk-recycling/#:~:text=Until%20now%20the%20UK,by%20the%20World%20Wildlife%20Fund.

[2] WRAP. Recycling Guidelines. (2017(Amended 2018)). Retrieved June 10, 2020 from http://www.wrap.org.uk/sites/files/wrap/Recycling%20guidelines%201.9.pdf

[3] David Gregory-Kumar. Big changes to recycling black plastic. (December 2018). Retrieved June 10 2020 from https://www.bbc.co.uk/news/uk-england-46571653#:~:text=Recycling%20black%20plastic%20is%20hard,off%20to%20landfill%20or%20incineration.

[4] Gamasutra. The Value of Repetition. (July 2012). Retrieved June 11 2020 from https://www.gamasutra.com/view/feature/174974/the_value_of_repetition.php?page=3

[5] Harmonix. Harmonix Music Systems. 2020. Retrieved June 11 2020 from https://www.harmonixmusic.com/

[6] Mifrah Ahmad. Categorizing Game Design Elements into Educational Game Design Fundamentals. November 2019. Retrieved June 11 2020 from https://www.researchgate.net/publication/337688105_Categorizing_Game_Design_Elements_into_Educational_Game_Design_Fundamentals

[7] WRAP. About Us. (2020). Retrieved June 12 2020 from http://www.wrap.org.uk/about-us/about

[8] British Council. Clean and Green. 2020. Retrieved August 10 2020 from https://learnenglishkids.britishcouncil.org/games/clean-and-green

[9] Turtle Diary. Recycling Waste. 2020. Retrieved August 10 2020 from https://www.turtlediary.com/game/recycling-waste.html

[10] Turtle Diary. Turtle Diary. 2020. Retrieved August 10 2020 from https://www.turtlediary.com/

[11] National Geographic Kids. Recycle Roundup. 2020. Retrieved August 10 2020 from https://kids.nationalgeographic.com/games/action-and-adventure/recycle-roundup-new/

[12] Recycling Bins. Recycling Bin Colours In The UK. September 2019. Retrieved August 10 2020 from https://www.recyclingbins.co.uk/blog/recycling-bin-colours-in-the-uk/

[13] Clint Bellanger. Recycle Items Set. August 2015. Retrieved August 10 2020 from https://opengameart.org/content/recycle-items-set

[14] RapidPunches. Vaquita Porpoise. February 2019. Retrieved August 10 2020 from https://rapidpunches.itch.io/vaquita

[15] GuttyKreum. Field of Green. June 2019. Retrieved August 10 2020 from https://guttykreum.itch.io/field-of-green

[16] Unity. Unity. 2020. Retrieved August 10 2020 from https://unity.com/

[17] Mozilla. WebGL: 2D and 3D graphics for the web. 2020. Retrieved August 10 2020 from https://developer.mozilla.org/en-US/docs/Web/API/WebGL_API

[18] GitHub Inc. GitHub Pages. 2020. Retrieved August 10 2020 from https://pages.github.com/

[19] GitHub Inc. About GitHub Pages. 2020. Retrieved August 10 2020 https://docs.github.com/en/github/working-with-github-pages/about-github-pages

[20] Microsoft. Introduction to the C# language and the .NET Framework. July 2015. Retrieved August 10 2020 from https://docs.microsoft.com/en-us/dotnet/csharp/getting-started/introduction-to-the-csharp-language-and-the-net-framework

[21] Unity. Prefabs. July 2018. Retrieved August 10 2020 from https://docs.unity3d.com/Manual/Prefabs.html
[22] Unity. Layer-based collision detection. 2019. Retrieved August 11 2020 from https://docs.unity3d.com/Manual/LayerBasedCollision.html

[23] Unity. Tags. 2019. Retrieved August 11 2020 from https://docs.unity3d.com/Manual/Tags.html

[24] Unity. Buoyancy Effector 2D. 2019. Retrieved August 11 2020 from https://docs.unity3d.com/Manual/class-BuoyancyEffector2D.html

[25] Unity. Canvas. 2020. Retrieved August 11 2020 from https://docs.unity3d.com/Packages/com.unity.ugui@1.0/manual/UICanvas.html

[26] Unity. PlayerPrefs. 2020. Retrieved August 11 2020 from https://docs.unity3d.com/ScriptReference/PlayerPrefs.html