

---

# EDUCATIONAL CYCLING GAME FOR YOUNGER AUDIENCE

---

Final Year Project



APRIL 16, 2018

KRISTINA BLINOVA

P4076882

## ABSTARCT

More than a half of UK primary school children regularly participate in playing video games, which is commonly believed to have a negative impact on their health and school marks. The aim of this project was to design a game which would achieve the opposite effects.

As part of the study a literature review in the fields of educational games, exercising games and games for children was conducted. The findings were used as a basis for the serious exergame prototype, which utilises cycling interface coupled with school curriculum maths questions. This preliminary study suggests that gaming, learning and exercising components can be combined to create an engaging video game for 7-9-year-old children.

## ACKNOWLEDGMENTS

I would like to thank my research supervisor Dr Julie Porteous, for her enthusiastic encouragement, professional guidance and invaluable support. I would also like to thank Sarah O'Brien, Julien Cordry and Alison Innerd, who inspired me to pick this project as well as provided useful and constructive recommendations on this research. I am particularly grateful for the assistance given by Dmytro Stratiychuk-Dear for his advice and help in keeping my project on schedule.

Finally, I would like to offer my special thanks to the University of Teesside and the School of Computing.

## Table of Contents

ABSTARCT.....	1
ACKNOWLEDGMENTS.....	2
1. INTRODUCTION.....	5
1.1. Problem Definition.....	5
1.2. Scope.....	5
2. METHODOLOGY.....	6
2.1. Tools.....	6
2.1.1. Hardware.....	6
2.1.2. Software.....	6
2.2. Process.....	6
3. RESEARCH.....	8
3.1. Educational Aspect.....	8
3.1.1. Subject choice.....	8
3.1.2. National Curriculum.....	8
3.1.3. Student book review.....	9
3.1.4. Types of educational games.....	9
3.1.5. Motivation to play educational games.....	9
3.2. Exercising Aspect.....	10
3.2.1. Effects of active video games.....	10
3.2.2. Feasibility and best practices.....	10
3.3. Games for Children.....	10
3.3.1. Rating.....	10
3.3.2. Engaging elements and human values.....	11
4. DESIGN.....	12
4.1. Goals.....	12
4.2. Concept.....	12
4.3. Progression.....	12
4.3.1. Rounds.....	12
4.3.2. Levels.....	13
4.4. Main Menu.....	13
4.5. Avatar Creation Component.....	14
4.5.1. Layout and mechanics.....	15
4.5.2. Items and pricing.....	15
4.6. Maths Quiz Component.....	16

4.6.1.	Layout and mechanics.....	17
4.6.2.	Bonus challenge .....	18
5.	IMPLEMENTATION .....	20
5.1.	Asset management at the avatar creation scene .....	20
5.1.1.	Storing assets .....	20
5.1.2.	Uploading assets .....	21
5.2.	Maths questions.....	23
5.3.	Audio elements .....	24
6.	EVALUATION .....	26
6.1.	Subjective evaluation .....	27
6.2.	Objective evaluation .....	28
7.	CONCLUSION.....	29
8.	FUTURE WORK .....	30
8.1.	Persistence .....	30
8.2.	Live testing .....	30
8.3.	Custom installer .....	30
9.	REFLECTION.....	31
10.	BIBLIOGRAPHY .....	32

# 1. INTRODUCTION

## 1.1. Problem Definition

According to statistics, more than 32 million of the United Kingdom residents regularly played video games in 2017, making UK the 5<sup>th</sup> largest game market in the world (Ukie, 2018). Children form a significant part of this phenomena. Sixty six percent of 8-11-year-olds play console games at home on weekly basis (Statista, 2018) and even more participate in PC and mobile gaming (Prigg, 2014).

It is debatable whether playing video games has a positive (Chuang & Chen, 2009) or a negative (Skoric, et al., 2009) impact on children, however criticism often stresses that gaming prevents children from getting “enough active play and exercise” and negatively affects “how well kid does in school” (Kids Health, 2014).

The aim of this project is to combat these stereotypes and use gamification in order to improve both activity level and school results of children. A successful prototype must be a fun to play video game demo, which provides exercising opportunities and learning outcomes.

## 1.2. Scope

The project will not focus on maximising physical benefits, but rather use exercising aspect to increase engagement and concentration among players. Cycling hardware will be used as the main interface due to its availability at the university.

The success of the project will be determined experimentally and at this stage will not involve testing with the focus group. The assessment criteria will be obtained via a literature review. The game will be aimed at 7-9-year-old school children.

## 2. METHODOLOGY

### 2.1. Tools

Both software and hardware ‘tools’ were utilised within this research to facilitate the creation of the prototype game. Further details on the specifications are provided below.

#### 2.1.1. Hardware

The set of hardware equipment consists of a mini cycle (Oypla, 2018) and a sensor with appropriate SDKs (Cyberbiking, 2016). Available output includes pedalling information of going forward, but not backwards. Controller, mouse or keyboard can be used in parallel with the sensor.

The mini cycle has a display which shows key workout data, such as distance travelled, session time and calories burned. This data cannot be accessed by software due to the sensor’s limitation but can be used during life testing to obtain information about physical activity level during the play. The mini cycle allows the resistance to be modified manually thus making cycling harder or easier. The resistance cannot be changed via software.

This set of hardware was chosen due to its availability at university.

#### 2.1.2. Software

The game demo is developed in Unity3d (Unity 3d, 2018) and utilises the 2D framework (Unity 3d , 2015). All the scripts are written in C#. Visual Studio 2017 (Visual Studio, 2017) was used as an IDE.

In addition to the programming software, Adobe Photoshop (Adobe, 2018) was used to create some of the visual assets for the game. Throughout the development GitKraken (Axosoft, 2018) was used as a source control.

### 2.2. Process

As shown on Figure 1 there are six main elements of the project development, which can be divided into three stages: research, planning and implementation.

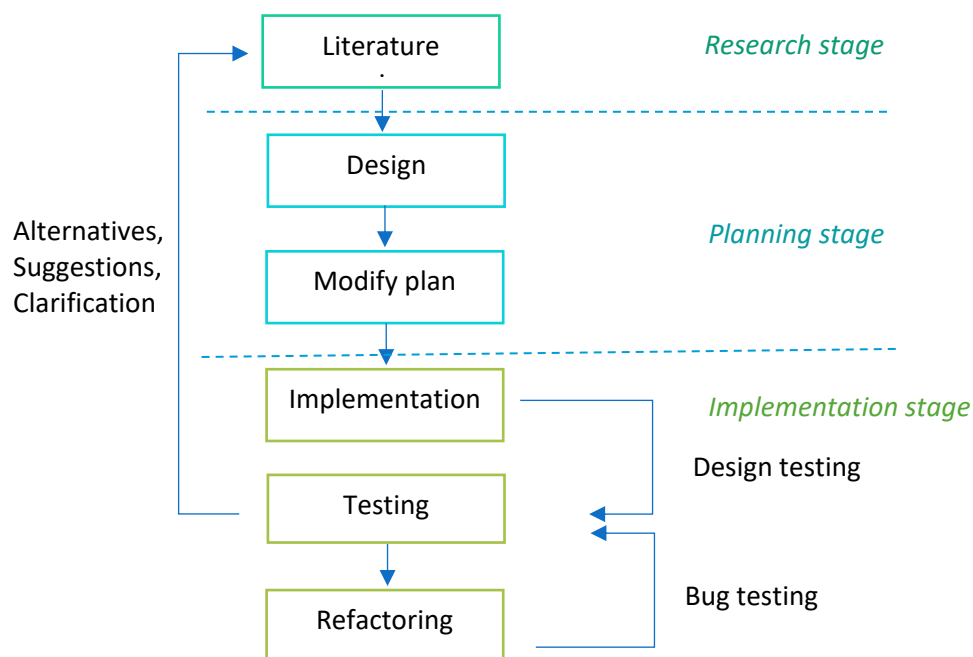


Figure 1 Development process

The development starts at the research stage where the literature review is conducted to help identify the best practices and approaches of creating an educational exercise game for children.

The planning stage focuses on reviewing the literature findings. Any changes to the original game concept, mechanics, goals, and other components are decided. Necessary adjustments to the schedule are made with regards to deadlines and available resources.

Once requirements are clearly defined the implementation stage starts. The initial goal is to produce working code as quickly as possible, even if it means efficiency compromise or hardcoding. This approach allows testing to be initiated relatively fast to examine whether the theoretical design solution works in practice. After the initial test the refactoring process starts to optimise the code. It is followed by further testing with the main focus being errors and bugs. At any point during the testing an extra literature review can be required. In this case the cycle starts over.



### 3. RESEARCH

Due to the lack of video games on the market which equally utilise exercising and learning aspects, the comprehensive research could not be conducted on this genre as a whole (Mellecker, et al., 2013). Instead, it was decided to focus on investigating more specific subject areas. The initial research targeted educational games and exercising games. In the later stage of the development the research also included games for children as a separate field.

#### 3.1. Educational Aspect

##### 3.1.1. Subject choice

The educational program of the Key Stage 2 students can differ depending on the type of school that they attend (Department of Education, 2015). The only compulsory events across all the educational institutions are the National tests in English and maths. Figure 2 shows the pass rate of the National test exams among the Key Stage 2 students.

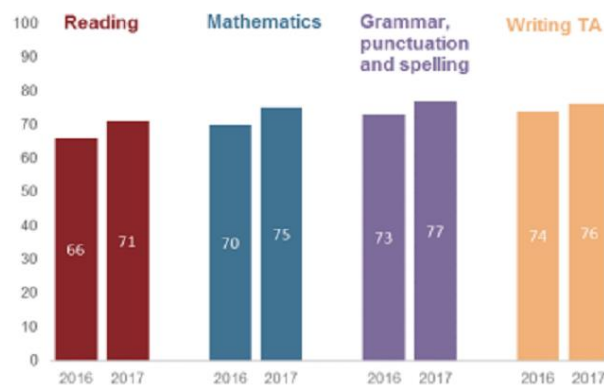


Figure 2 Percentage reaching the expected standard by subject among the Key Stage 2 students (Department of Education, 2017)

In 2016 and 2017 the lowest results were achieved in reading and mathematics, which indicates that the learning environment for these subjects can be improved. Since maths questions can be procedurally generated as opposed to reading tasks, it was decided to base the game around maths curriculum.

##### 3.1.2. National Curriculum

Mathematics programme of study for the key stage 2 in England is defined by the National curriculum (Department for Education, 2014). Among others, the statutory requirements for the Year 2 and Year 3 students include the following:

1. Ability to add and subtract mentally, including:
  - a two-digit number and ones
  - a two-digit number and tens
  - two two-digit numbers
2. Ability to recognise and use symbols for pounds (£) and pence (p); combine amounts to make a value
3. Ability to solve simple problems in a practical context involving addition and subtraction of money of the same unit, including giving change

### 3.1.3. Student book review

In order to decide on the best approach to introduce those tasks, three student books were consulted: “Let’s do Mental Maths” (Brodie, 2013), “Mental Maths” (Collins, 2017) and “Mental Arithmetic 3” (Schofield & Sims, 2017). All of these books were published in London in the recent years and support the national curriculum. Table 1 shows a comparison analysis of the books.

Firstly, it can be noted that the number of tasks per session varies from book to book, however those always include questions where the use of language is kept to a minimum and questions using number vocabulary. Secondly, the concept of currency adds an extra layer of complexity to the arithmetic question e.g.  $25 + 9$  and  $25p + 9p$  follow the same numerical pattern ( $A + B$ ) but are considered different types of questions. Finally, each book has a reward system which provides visually clear feedback about the success of the student and a non-learning activity e.g. colouring.

Table 1 Student book comparative analysis

	Let’s Do Mental Maths	Mental Maths	Mental Arithmetic 3
<b>Number of questions per test</b>	20 questions	15 questions	3 parts, 12 questions each
<b>Age group</b>	8-9	7-9	7-11
<b>Addition/Subtraction question examples</b>	<ul style="list-style-type: none"> <li>• <math>58p + 22p</math></li> <li>• What is the total of 18 and 23?</li> <li>• What is my change from £1 when I spend 56p?</li> </ul>	<ul style="list-style-type: none"> <li>• <math>100 - 5</math></li> <li>• <math>25p + 9p</math></li> <li>• Work out the change from £1:               <ul style="list-style-type: none"> <li>a) 50p</li> <li>b) 45p</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• <math>43 + 47</math></li> <li>• Find the total of 29 and 81</li> <li>• What must be added to 75p to make £1.50?</li> </ul>
<b>Reward and feedback</b>	Reward stickers + numerical score value	Colouring objects to represent score. Objects are a part of an illustration. Illustrations differ for each test.	Colouring cells in the grid to represent score.

### 3.1.4. Types of educational games

Educational games can be divided into teaching games or testing games (Kapp, et al., 2014). Testing games require the learner to know the information to be successful. Games such as Trivia and Jeopardy are classified as testing games. The aim of the teaching games, on the other hand, is to help the learner accumulate new knowledge or new skills. For example, it is not enough to know how to move pieces to win a chess game. Instead, a player is required to come up with new tactics, plan, adapt and adjust.

Testing games are often criticised for not being engaging with children, however it can be an excellent way of learning things that require lots of repetitive practice (Prensky, 2007). As arithmetic falls under that category, it was decided to use testing game approach for this project.

### 3.1.5. Motivation to play educational games

Research shows that children’s motivation to learn maths concepts is increased when the problem is presented in a game context. Interestingly, this remains true even if the game is perceived as “full of maths” (Kamran, 2008). Similar results were achieved by studying educational games focused on

other learning areas, such as nutrition (Mellecker, et al., 2013). Users, children aged 7-11, positively reacted to a game which “help you learn while having fun”. In other words, users can consciously work towards the curriculum goals, but still find the process enjoyable.

The elements of games that provide motivation include engaging graphics, onscreen rewards and progress feedback (Prensky, 2007). As long as the learning context does not suppress those game features, it can be expected to keep players engaged.

### 3.2. Exercising Aspect

With introduction of Microsoft Kinect, Wii, and Sony Eye Camera, exercising video games are becoming popular among gamers. Monique et al., demonstrate that 40 % of adolescents play active and non-active video games, and 3 % play active games exclusively (Monique, et al., 2014). This section of the research summarises relevant information about exercise games obtained via literature review.

#### 3.2.1. Effects of active video games

Research suggests that regularly playing active video games can result in positive effects on children’s overall physical activity level (Cliona Ni Mhurchu, et al., 2008). Health benefits of exercising games are comparable to those of gym work outs. Studies show that the effect of active video gaming on the heart rate did not differ significantly from traditional physical activities (Peng, et al., 2011). It is also proven that active video games can be used to fight against children’s obesity, reduce body fat and result in positive change in waist circumference (Maddison, et al., 2011). Interactive cycling games in particular can result in a higher exercise intensity and provide significantly greater enjoyment than conventional exercise (Monedero, et al., 2015).

Another area, which is affected by exercising games is non-active gaming. Research consistently shows that the daily time spent in non-active video games is decreased, if users have access to active games (Cliona Ni Mhurchu, et al., 2008), (Maddison, et al., 2011), (Simons, et al., 2015).

#### 3.2.2. Feasibility and best practices

Research shows that the level of enjoyment from playing active video game steadily decreases over time. Within a session of a basic cycling game, a significant drop in interest can be noticed within 20-30 minutes of play time (Monedero, et al., 2015). In the long-perspective, the boredom becomes noticeable within 7-10 weeks (Mellecker, et al., 2013). Both observations can be explained with the general tendencies towards engagement in sports overtime (Monedero, et al., 2015) and a poor game design, which becomes evident once the novelty factor of the active interface disappears (Mellecker, et al., 2013). This can be overcome by investing more resources into the “gaming aspect” of the application. For example, a study suggests that active video games with narrative show greater physical activity than non-narrative games among 8-11-year-old children (Lu, et al., 2016).

### 3.3. Games for Children

#### 3.3.1. Rating

In order for the game to be suitable to primary school children, it should not contain any elements inappropriate for their age (PEGI, 2017). There are five PEGI rating categories recognised in the UK. The successful prototype of this research project can be labelled as 3+ or 7+ however mustn’t have any features of the 12+ rating or higher. The breakdown of the requirements is shown in table 2.

Table 2 PEGI rating description

3+	7+	12+
May contain some violence in a comical context (e.g. very mild cartoon slapstick)	May contain occasional violence to non-realistic fantasy characters, pictures or sounds likely to be scary or frightening to young children, and nudity in a non-sexual context.	May contain graphic violence towards fantasy characters, non-graphic violence towards humans or animals, explicit sexual descriptions or images (nude people in a sexual context, although not necessarily explicit in content), and mild swearing.

### 3.3.2. Engaging elements and human values

It is hard to objectively evaluate the level of players engagement or make conclusions about how different game elements contribute towards it (Whitton, 2010). What is known, is that a game event must trigger some human value to provoke emotion (Tynan, 2013). For example, losing a piece at the late stage of the chess game provokes great emotional response because the victory/defeat human value is shifted. The same action in the early game results in smaller engagement as there are many other pieces left on the board and the impact on the human value is smaller.

It is essential not to discourage children in their learning and exercising efforts (Prensky, 2007). Therefore, the literature review was focused on “less dramatic human values” such as [unique, personalised vs standard], [skilled vs unskilled], [knowledge vs ignorance] as opposed to [life vs death] human value, which is commonly used in the games for adults. Table 3 summarises the common ways of incorporating human values in games.

Table 3 Incorporation of the human value triggers in children games

Value	Incorporation	Game examples
<b>Unique, Personalised vs Standard</b>	Avatar creation/modification	<ul style="list-style-type: none"> <li>• The Sims series</li> <li>• Pokémon Go</li> <li>• Hogwarts Mystery</li> </ul>
	Building creation/modification for non-functional purposes	<ul style="list-style-type: none"> <li>• Tycoon Zoo series</li> <li>• Fantasy Life</li> <li>• Sims Mobile</li> </ul>
<b>Skilled vs Unskilled</b>	Challenge, which requires player to develop a set of skills in real life e.g. reflexes, abstract thinking	<ul style="list-style-type: none"> <li>• Mario</li> <li>• Crash Bandicoot series</li> <li>• Civilization series</li> </ul>
<b>Knowledge vs Ignorance</b>	As player progresses through the game, the narrative is unravelled, and mysteries are revealed	<ul style="list-style-type: none"> <li>• Professor Layton series</li> <li>• Stardew Valley</li> <li>• To the Moon</li> </ul>
	Exploration, discovery of new territories or secret pathways	<ul style="list-style-type: none"> <li>• Rayman Origins</li> <li>• Minecraft</li> <li>• The Legend of Zelda series</li> </ul>

## 4. DESIGN

### 4.1. Goals

Based on the literature review, the following list of requirements for the successful prototype was developed:

1. The game must utilise learning and exercising aspects in similar proportion (see section 1.1).
2. The learning focus of the game must be addition and subtraction of money, (see section 3.1.2).
3. The game must have narrative (see section 3.2.2)
4. The game must provide on-screen rewards (see sections 3.1.3, 3.1.5, 3.2.2 and 3.3.2)
5. The game must be engaging and therefore trigger a shift in human values (see section 3.3.2)
6. The game content must be suitable for 7-9-year-old children according to UK game rating system (see section 3.3.1)

### 4.2. Concept

The game starts at the avatar creation scene, where the player is offered to personalise his in-game representation. However most of the items must be purchased with golden coins, which the player does not possess. To obtain the coins player must go on adventure and complete a series of challenges.

All the challenges take place in the maths scene. The player is presented with three treasure chests and a maths question. If the question is answered correctly, then one of the chests would open revealing several coins and a quest item related to the narrative. Next the new maths question appears.

The game has a progression system, which includes increasing difficulty of the maths questions, greater rewards and a mini game between levels.

The prototype is designed to be a single player video game, which can be played by 7-9-year-old children without teachers' supervision.

### 4.3. Progression

In game progression happens on multiple layers. Currently, the prototype has three levels where each level has three rounds.

#### 4.3.1. Rounds

Each round has different coloured chest, which marks player's progress within the level. Table 4 shows that each round also requires more patience and long-term concentration to open the chest and get a reward. At the end of each round the game is saved and the player can return to the avatar creation without losing any progress. Once the last round was complete, the player is offered to play a mini game to progress to the next level.

*Table 4 Round Progression in game*

	Number of chests	Questions per chest	Chest colour
Round 1	3	1	Brown
Round 2	3	2	Blue
Round 3	1	3	Purple

#### 4.3.2. Levels

Table 5 shows the level progression. The difficulty of the game increases with each level, starting with one arithmetic operation and finishing with a combination of two. The number of coins per opened chest also increases, which leads to more avatar items to be unlocked.

*Table 5 Level progression in game*

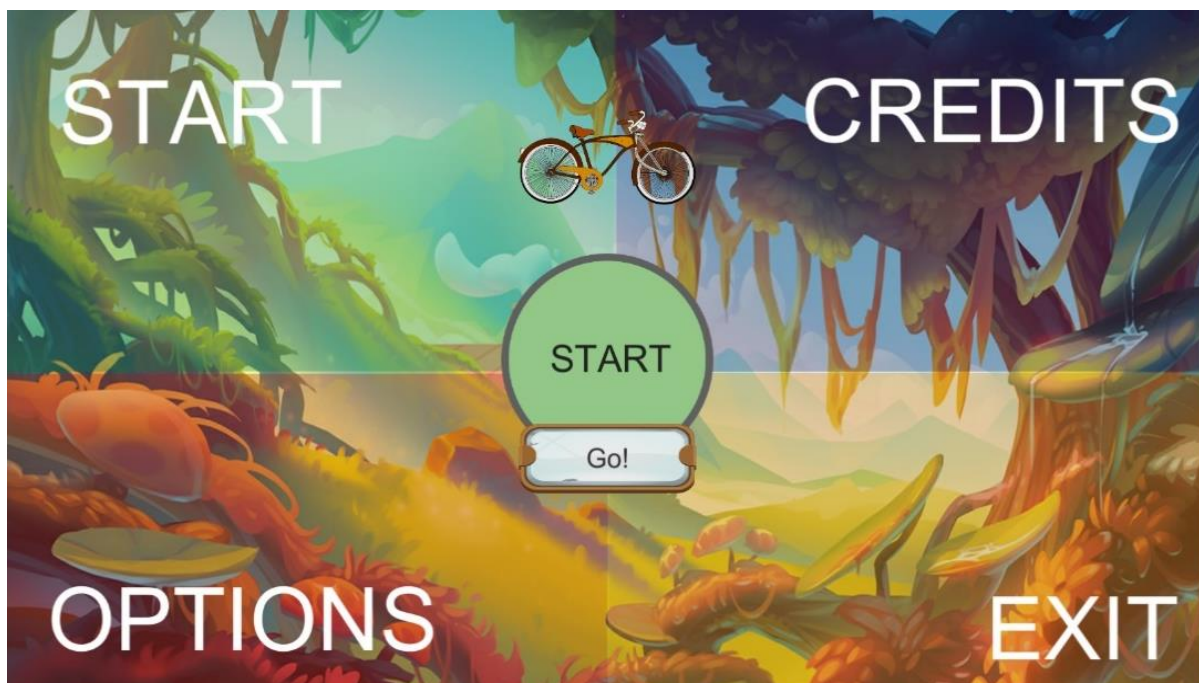
	Questions	Coins	Quest	Quest form
Level 1	Addition	10 x chest	Restore the map	Jigsaw puzzle
Level 2	Subtraction	100 x chest	Restore the lighthouse	Colouring book
Level 3	Addition and subtraction	1000 x chest	Restore the ship	Connect the dots

Levels are visually different. Each level has its own background picture, background music and visual progress feedback (quest). See section 4.6 for figures and details.

#### 4.4. Main Menu

One of the issues, which became apparent during the implementation stage, is the necessity to adjust the equipment before the game session. It is essential to ensure that the position of the cycle, its resistance and the height of the chair are appropriate for the user, to avoid potential injuries. The design challenge was to seamlessly incorporate this functionality in the prototype.

The problem was solved by creating the main menu screen, which is navigated by pedalling. The bicycle starts on the “PLAY” option, which allows user to start the game immediately or take a full circle around other options to test the hardware in the “safe” environment.



*Figure 3 Initial phase of the main menu*



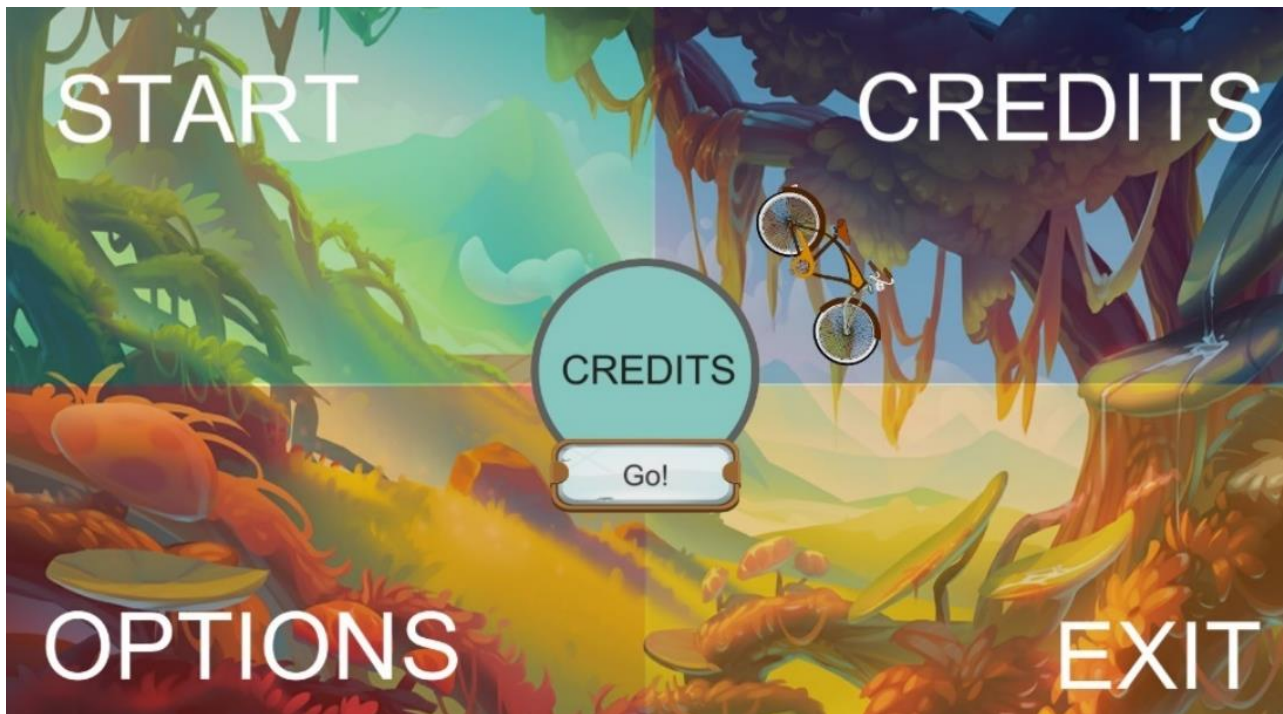


Figure 4 Credits is selected

#### 4.5. Avatar Creation Component

Figure 5 demonstrates the avatar creation component. The design rational behind this scene is discussed below.



Figure 5 Avatar creation scene in game

#### 4.5.1. Layout and mechanics

Figure 6 shows the layout of the avatar creation scene. The screen is divided into three areas. UI elements, which are present in all the game play scenes are marked blue.

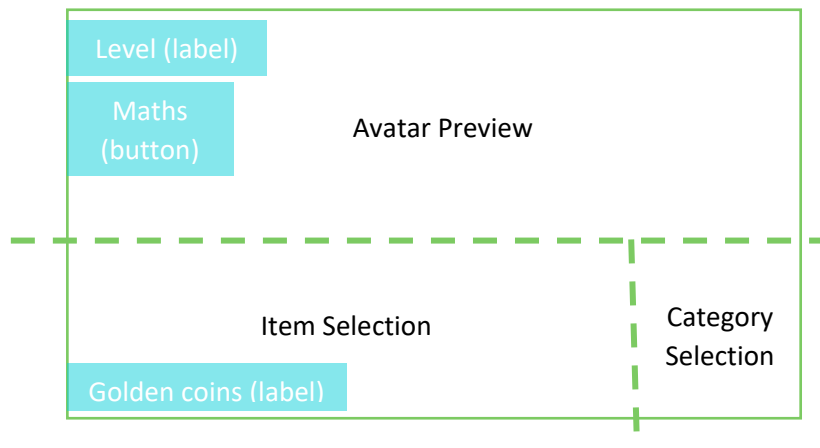


Figure 6 Layout of the avatar scene

The following actions can be performed by the player:

- **Select a category.** Player clicks on one of the icons in the *category selection area*, which immediately loads a collection of avatar parts into the *item selection area*.
- **Select an item.** Player uses arrow keys on the keyboard to navigate between avatar parts in the *item selection area*. Alternatively, arrow buttons in the *category selection area* can be used. Selected item is highlighted yellow, if it is owned by the player, or green if it is not.
- **Buy an item.** Player must select an item first, then press space on the keyboard or click the price button. Player can only buy an avatar part if it is not owned and there are enough golden coins to do so.
- **Apply an item.** Player must select an owned avatar part first, then pedal. The bicycle sprite will move towards the avatar. Item will be applied, once the bicycle reaches the avatar.
- **Move to the maths scene.** Player clicks “On Adventure” button at the top of the screen.

#### 4.5.2. Items and pricing

Avatar items are priced differently depending on the item category. As the player progresses through the game, the rewards become greater and therefore more types of avatar parts can be afforded. Table 6 shows the pricing breakdown.

Table 6 Pricing of the avatar parts

Category	Price
Skin colour	Free
Eyes	$10 + 5 * \text{item id}$
Nose	$10 + 5 * \text{item id}$
Mouth	$10 + 5 * \text{item id}$
Face shape	$100 + 50 * \text{item id}$
Hair dye	500
Hair up (fringe)	$700 + 350 * \text{item id}$
Hair down	$700 + 350 * \text{item id}$
Body	$1000 * \text{item id}$



*Item id* stands for the item's sequence number, starting with 0. For example, if there are 3 types of eyes available, the first item will cost 10 coins ( $10 + 5 * 0$ ), second item 15 coins ( $10 + 5 * 1$ ) and third item 20 coins ( $10 + 5 * 2$ ). Skin colours are available for free for ethical reasons.

#### 4.6. Maths Quiz Component

Figures 7, 8 and 9 demonstrate different rounds and levels of the maths scene.



Figure 8 Level 1 Round 2

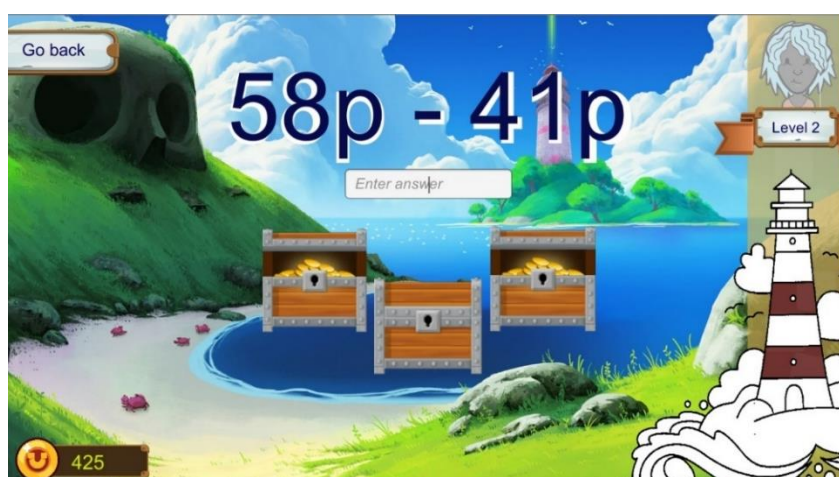


Figure 9 Level 2 Round 1



Figure 7 Level 3 Round 3

#### 4.6.1. Layout and mechanics

All the maths levels follow the same layout presented in Figure 10. The screen is divided into 2 areas and has 3 static UI elements, marked blue.

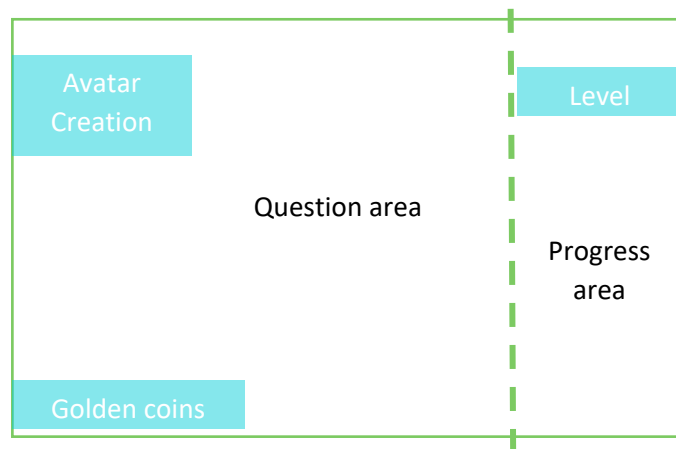


Figure 10 Layout of the maths scene

The following actions can be performed by the player:

- **Reveal a maths question.** As player cycles, the question appears in the *question area*. If the cycling is stopped, the question will fade away.
- **Answer a question.** Player enters an answer in the input field in the *question area*. If the answer is incorrect, then it is erased from the input field, a notification appears, and a sound effect is played. Correct answers contribute towards the progression of the game (see section 4.3).
- **See progress feedback.** There are various components, which indicate player's progress:
  - Level label located in the *progress area* shows current level number (e.g. "Level 2").
  - The colour of the chests in the *question area* indicates current round number (see section 4.3.1).
  - Quest image in the *progress area* indicates current level progress as a percentage towards completion.
  - Golden coins label shows current number of coins owned.
  - Avatar image in the *progress area* shows currently equipped avatar parts as a reminder of how much progress was made so far.
  - Level up label in the *question area* reminds player to level up once available.
- **Level up.** Once player is eligible move to the next level (see section 4.3.2), a level up button in the *progress area* appears. When player clicks on it a mini-game starts (see section 4.6.2).
- **Move to the avatar creation scene.** Player clicks "Go back" button at the top of the screen.

#### 4.6.2. Bonus challenge

The bonus challenge becomes available at the end of each level. Player has an option of levelling up immediately or completing more rounds before taking the challenge.

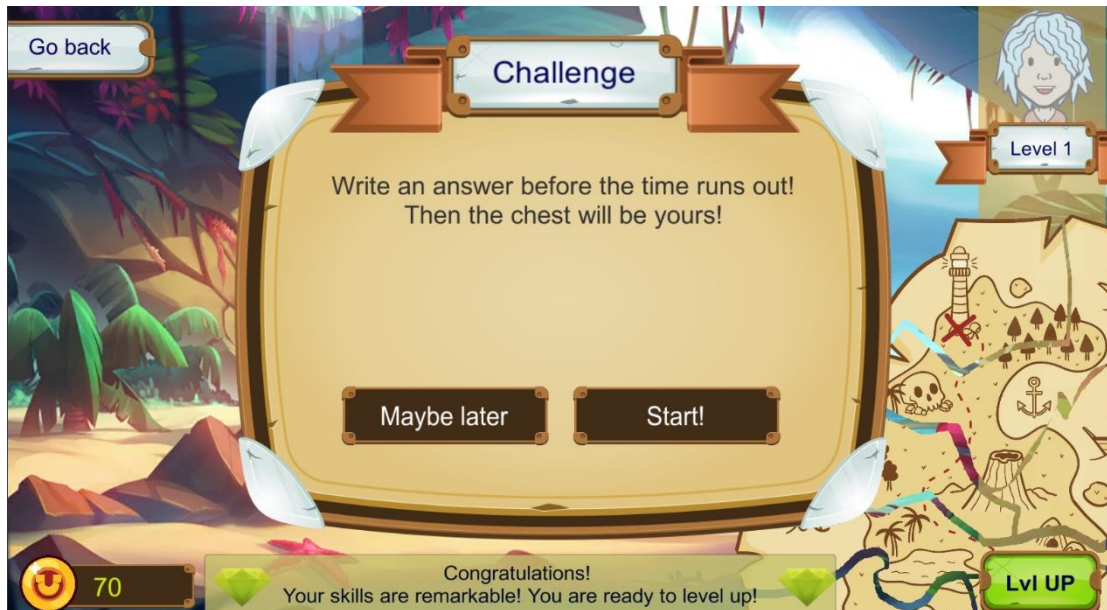


Figure 11 Bonus challenge becomes available

The bonus challenge is won if the player correctly answers one maths question within the time limit. The bonus question is on the same topic as regular level question, but it is formulated differently. For example, regular level 1 questions are presented as [Number A p + Number B p] and the bonus question as [Find the total of Number A p and Number B p] (see section 3.1.3).



Figure 12 Bonus challenge question



Player can have an unlimited number of tries, however if the time runs out then the challenge is lost. Correct answer will be shown. Player can choose to try the challenge again with the different question.



Figure 13 Bonus challenge is lost

If the bonus challenge is won, player receives a number of golden coins, which is 50 times greater than the regular chest reward on that level. A story snippet related to the quest is revealed and the player immediately moves on to the next level. Level labels are updated.



Figure 14 Bonus challenge is won

## 5. IMPLEMENTATION

### 5.1. Asset management at the avatar creation scene

Avatar creation scene contains 95 different avatar sprites, which can be grouped together in different ways as shown in the Table 7. Both the way of uploading and storing the elements were considered.

Table 7 Avatar parts categorisation

Category	Colour	Changes colour	Number of variation
<b>Skin colour</b>	Skin	Yes	3
<b>Face shape</b>	Skin	-	5 * 3 colours
<b>Body</b>	Skin	-	3 * 3 colours
<b>Hair Colour</b>	Hair Colour	Yes	5
<b>Hair up (fringe)</b>	Hair Colour	-	5 * 5 colours
<b>Hair down</b>	Hair Colour	-	4 * 5 colours
<b>Eyes</b>	-	-	7
<b>Mouth</b>	-	-	7
<b>Nose</b>	-	-	4

#### 5.1.1. Storing assets

Figure 15 shows the key classes used to store asset information.

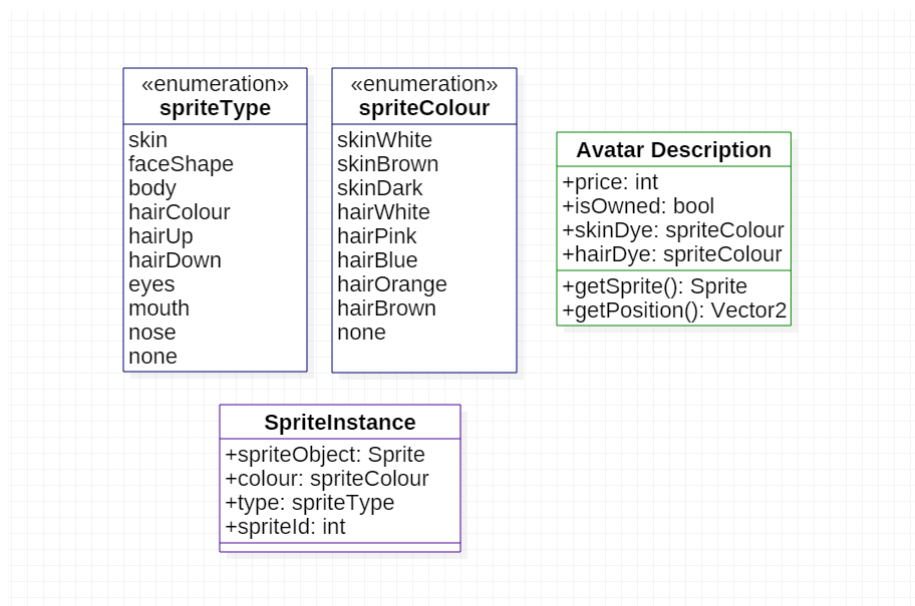


Figure 15 Classes used to store avatar assets

At the beginning of the game all the sprites are loaded into the lists of *SpriteInstances* divided by sprite type e.g.

```
public List<SpriteInstance> faceShapes
```

Sprite instance does not inherit from Mono Behaviour class and therefore does not require a *GameObject* representation. This approach allows saving memory space and computational power, which is not crucial at the scale of this prototype but can make a performance difference if the game is to be expanded.

*GameObjects* are instantiated when the player selects a category and needs to interact with the item. Information from the relevant *SpriteInstances* is converted into Unity components. An additional *AvatarDescription* script is attached. Figure 16 shows the code of the conversion.

```
//Obtain current collection from avatar collection script
List<SpriteInstance> currentCollection = ObtainCurrentCollection();

foreach (var item in currentCollection)
{
    //Check if type and colour match
    if(ItemIsValid(item))
    {
        //-----CREATE NEW ITEM-----
        GameObject newItem = new GameObject();

        //Sprite Renderer
        newItem.AddComponent<SpriteRenderer>();
        newItem.GetComponent<SpriteRenderer>().sprite = item.spriteObject;

        //Script
        newItem.AddComponent<AvatarDescription>();

        //ID
        newItem.GetComponent<AvatarDescription>().spriteId = item.spriteId;

        //----POPULATE AVATAR DESCRIPTION----
        //Set price and colours
        #region PopulateAvatarDescription

        //Instantiate
        selectableItems.Add(newItem);
    }
}
```

Figure 16 Code that converts *SpriteInstance* variables into *GameObjects*

### 5.1.2. Uploading assets

The naïve approach of uploading assets would be to manually drag and drop each sprite from the asset folder to the relative lists and then manually set *spriteColour*, *spriteType* and *spriteId* variables. This approach, however, is very time consuming and error prone. Therefore, a script for automated asset upload was developed.

```
[MenuItem("AssetDatabase/2DPrototype")]

void Start ()
{
    //Array of strings to store paths to the asset folders
    string[] folders = new string[9];

    //Assigning paths
    folders[0] = "Assets/VisualAssets/AvatarElements/FaceShape";
    folders[1] = "Assets/VisualAssets/AvatarElements/SkinColour";
    [...]
    folders[8] = "Assets/VisualAssets/AvatarElements/Nose";

    //Finds assets of the type Sprite in given folder paths
    var paths = AssetDatabase.FindAssets("t:Sprite", folders);
    foreach (string guid in paths)
    {
        //Load sprite from path to a list
        LoadSprite(guid);
    }
}
```

Figure 17 Extracting assets from the prefab folder via code

Figure 17 shows the way to extract all the assets of the type *Sprite* from a given number of folders. Then, folder by folder, the results are processed and converted into *SpriteInstances* which are immediately placed in the corresponding lists. All the sprites are named using the following convention: [type id colour] e.g. HairDown 1 Blue. This allows all the key information such as type, id and colour to be extracted as shown in Figure 18.

```
void LoadSprite(string path)
{
    //Debug
    Debug.Log(AssetDatabase.GUIDToAssetPath(path));

    //Split name to components
    string[] fileName = AssetDatabase.GUIDToAssetPath(path).Split('/',
    '.');

    //Split file name to components
    string[] components = fileName[fileName.Length-2].Split(' ');

    //Sprite instance
    SpriteInstance newInstance = new SpriteInstance();

    //ID
    newInstance.spriteId = int.Parse(components[1]);

    //Assign sprite object
    newInstance.spriteObject =
    (Sprite)AssetDatabase.LoadAssetAtPath(AssetDatabase.GUIDToAssetPath(path),
    typeof(Sprite));

    //Switch depending on the avatar part
    AssignToListByType(components[0], components[2], newInstance);
}
```

Figure 18 Processing assets

## 5.2. Maths questions

All the maths questions in game are procedurally generated. Figure 19 shows the key stages of the process. Simplified class diagrams are used for easier understanding.

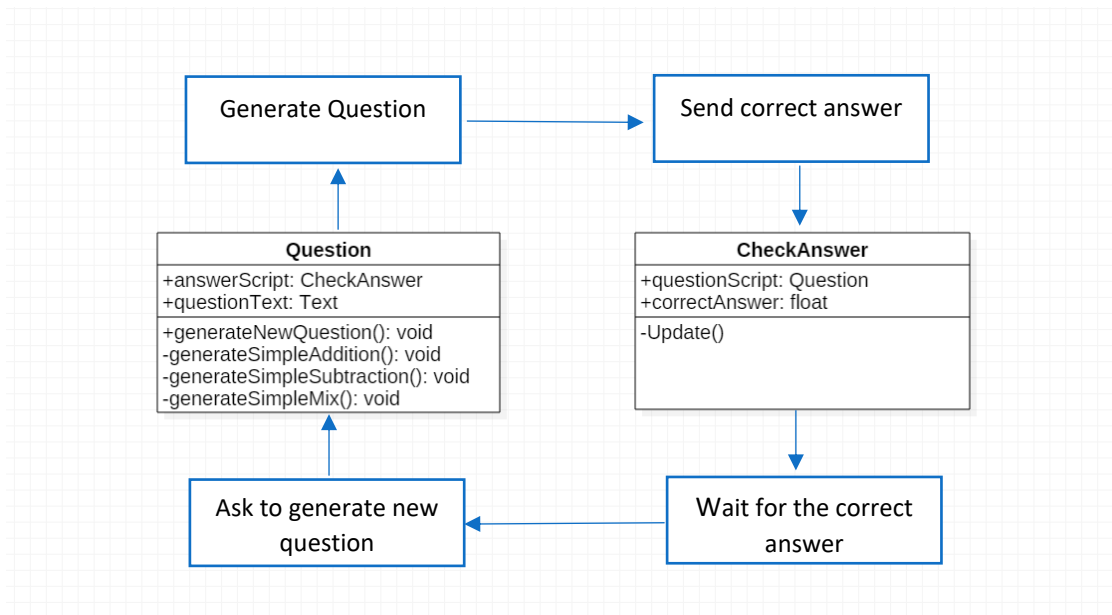


Figure 19 Class diagram for generating maths questions

At the start of the scene a *Question* class calls its public function “`public void generateNewQuestion()`”, which decides what type of question should be generated based on the player’s level. Next an appropriate private function is called. Figure 20 shows an example of addition question generation.

```

//Script that checks user answers
public CheckAnswer answerScript;

//Example: 10p + 34p
void generateSimpleAddition()
{
    //Get random int between 1-90
    int numberA = Random.Range(7, 90);

    //Get the maximum value of the second int
    int difference = 100 - numberA;

    //Randomly generate second int
    int numberB = Random.Range(7, difference);

    //Display the question
    questionText.text = numberA.ToString() + "p + " + numberB + "p";
    questionShadowText.text = questionText.text;

    //Set answer in the answer script
    answerScript.setCorrectAnswer(numberA + numberB);

    //Display answer in the Debugger
    Debug.Log(numberA + numberB);
}

```

Figure 20 Addition question generation



After the question is generated, the correct answer is passed to the *CheckAnswer* class. On update the class checks if the player entered any answer in the input field, if this answer is longer than two digits and if it is correct. If all three are true, then the *Check Answer* class calls the “`public void generateNewQuestion()`” function of the *Question* class and the cycle starts over.

If the answer is incorrect, then the call to the *Question* class does not happen. Player’s answer is deleted from the input field. One of the four feedback phrases is randomly chosen and displayed.

```
//Generate a random number between 1 and 4
int randomPhrase = Random.Range(1, 5);

//Choose a feedback phrase based on the random number
switch (randomPhrase)
{
    case 1:
        inputField.placeholder.GetComponent<Text>().text = "NOT EXACTLY...";
        break;

    case 2:
        inputField.placeholder.GetComponent<Text>().text = "TRY AGAIN";
        break;

    case 3:
        inputField.placeholder.GetComponent<Text>().text = "MM, NOT REALLY";
        break;

    case 4:
        inputField.placeholder.GetComponent<Text>().text = "ARE YOU SURE?..";
        break;

    default:
        inputField.placeholder.GetComponent<Text>().text = "NOT EXACTLY...";
        break;
}
```

Figure 21 Handling incorrect answers

### 5.3. Audio elements

The audio in game is managed in the same way across all scenes. In the root of the hierarchy an empty game object is created tagged as “AudioManager”. It has an audio script and three children with one audio source component each. The script contains references to all the music clips used in the scene and public functions to play them. Figure 22 shows an example.

```
//Btn click
public void PlayBtnClick()
{
    soundEffectsSource.clip = btnClickClip;
    soundEffectsSource.Play();
}
```

Figure 22 Function to play an audio clip

This approach allows easily swapping audio assets during the development. Three audio sources are required to play up to three music clips at a time e.g. background music, chest opening sound effect and bonus challenge ready sound effect.

The following music and sound effects were implemented in the game:

**Background Music:**

- Main menu
- Credits
- Options
- Avatar Creation
- Maths level (different for level 1, level 2, level 3)

**Sound Effects**

- Button clicks
- Selection changed on avatar item
- Avatar part was bought
- Avatar part cannot be bought
- Avatar part is applied
- Chest opened
- Chest half-way open
- Incorrect answer
- Bonus challenge ready
- Bonus challenge victory

## 6. EVALUATION

The final product was analysed using an Activity Theory-based Model of Serious Games (ATMSG) to provide a comprehensive view of the structure of the game, including both its high-level purpose and concrete implementation (Carvalho, et al., 2015). The original model was modified by adding an “exercise” category to match project’s goals. Figure 23 represents the game sequence and implemented components. Table 8 shows the way gaming, learning and exercising are integrated into the key stages of the game in a greater detail. The ATMSG diagram can be used to decide of whether the goals listed in the design part of the report are met (see section 4.1).

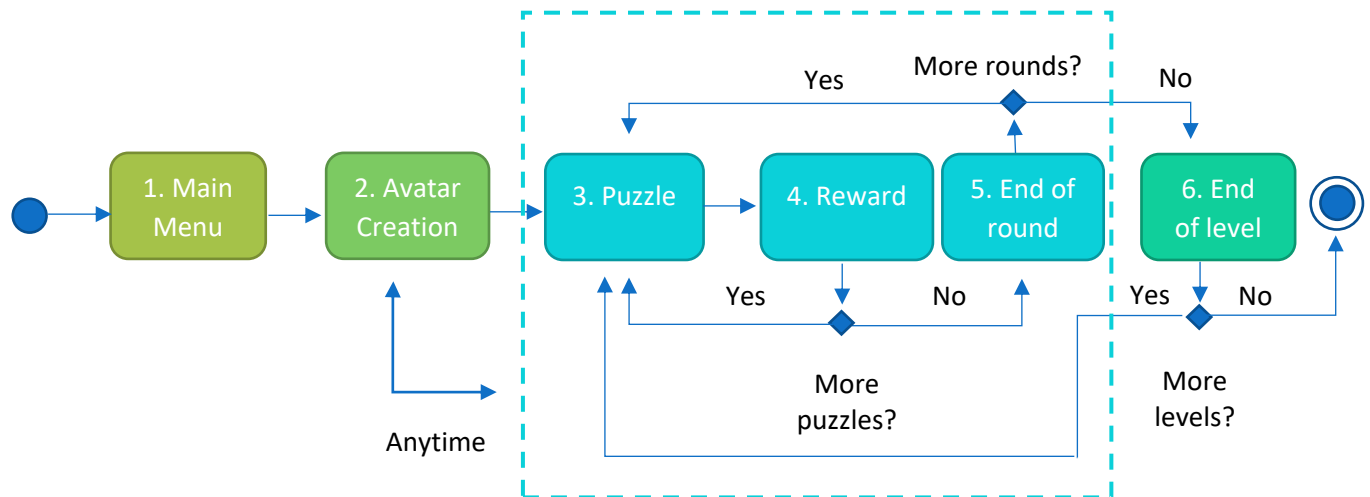


Figure 23 Game sequence represented as ATMSG

Table 8 Game events and stages according to ATMSG guidelines

Game Node	Gaming	Learning	Exercise	Comments
1. Main menu	Standard game element. Provides access to options, credits, start of the game and exit from the application.	-	Player navigates through the menu by cycling from one option to another. This provides a warm-up prior to any gameplay.	Main menu is a safe place to adjust the equipment and ensure that player's position is comfortable.
2. Avatar Creation	Player creates a character by choosing elements from given categories, such as hair colour or face shape. Unlocking new elements requires spending in-game currency. Avatar is visible in later stages of the game.	Game simulates real-life shopping experience. Player has a limited amount of money and can choose how to spend it. Interface prevents players from purchasing items unless they have enough coins.	To apply a new element to the avatar, player must deliver it by bicycle. The bicycle moves as the player cycles forward.	Interest to avatar creation among children and teenagers is outlined by multiple researches (Kafai, 2009).

3. <i>Puzzle</i>	Player answers a series of quiz questions. If the answer is incorrect, the player has to try again.	Puzzle after puzzle, player's arithmetic skills are tested. Questions gradually become harder with each level.	The question is only visible if the player is cycling. If cycling is stopped, then the questions fades away.	Questions are procedurally generated, which aids replay ability.
4. <i>Reward</i>	Correct answers open treasure chests which contain a certain number of coins and a quest item e.g. part of the map. Sound and visual effects empathise the importance.	-	-	Rewards vary depending on the player's level.
5. <i>End of round</i>	Chests change colour and become harder to open, more correct answers are required. Player's progress is saved.	Player is pushed to solve multiple questions at a time before taking a break. This is similar to test conditions.	Player is pushed to cycle consistently for a prolonged period of time.	Difficulty is increased via gaming rather than learning element.
6. <i>End of level</i>	At the end of the level player is offered to level up by completing a special challenge. Player can choose when to do it. Successful completion grants great rewards and story progression. Background picture, music and the quest change.	After answering ten questions player is offered to complete a test. The same skill is tested as during normal puzzles, but the wording of the question is different. There is a 30 second time limit. If unsuccessful, the player can try again.	-	Wording of the questions is taken from the student book. (see section 3.1.3).

### 6.1. Subjective evaluation

Design requirements 1 and 5 cannot be entirely objectively evaluated. The conclusions are based on the literature research and subjective opinion.

1. As shown in Table 8, most of the game stages provide outcomes in all three areas. There is no clear distinction between “fun”, “learning” and “exercising” in the prototype, which is especially evident at stages 1 and 2. The main menu is a non-entertaining, structural, part of the game, which still contributes towards exercising. Avatar creation was developed as a “break from maths”, but it

provides a learning outcome regardless. There is a potential for improvement at stage 4, but overall the prototype shows even distribution between all the areas.

5. The prototype incorporates game elements, which are aimed at triggering human values and therefore be engaging for children (see section 3.3.2).

Stage 2 enables player to shift between *unique*, *personalised* and *standardised* state by creating an avatar. Stage 3 challenges player to answer similar maths question and therefore move from *unskilled* to *skilled* state. The shift is reinforced by the visual feedback of the current level and current round. Finally, stages 4 and 6 trigger a *knowledge vs ignorance* human value by introducing a narrative, which unravels as the player progresses through the game.

Similar ways of incorporating human values in games can be met in successful existing products on the market (see section 3.3.2). Therefore, it is believed that the goal 5 of the design document was met.

### 6.2. Objective evaluation

Design requirements 2-4 and 6 can be objectively evaluated:

2. The puzzle questions on stages 3 and 6 focus solely on the addition and subtraction of money and are presented in the way synonymous with modern student books. There is an additional learning layer on stage 2, which allows testing acquired skills in the real-world situation.
3. The game has narrative elements, which are revealed in a format of quest items on stage 4. More conventional story snippets appear on stage 6.
4. On-screen rewards are presented in a variety of formats including in-game currency, quest items, levelling system and avatar parts.
6. The game meets PEGI 3+ requirements and therefore is suitable for 7-9-year-old children (see section 3.3.1).

## 7. CONCLUSION

The research presented here shows that gaming, learning and exercising components can be combined in equal proportions to create an engaging video game for 7-9-year-old children. Previous studies show that this approach to the gaming industry can result in significant improvements of children's cognitive skills and physical activity level (Cliona Ni Mhurchu, et al., 2008). As a part of this project a list of the design goals was formulated based on the review of the existing literature of the serious exergames. Following the identification of requirements, a prototype was developed, which utilises a cycling interface and incorporates school curriculum maths questions. Evaluating children's development in the aforementioned areas is outside the scope of this research.

## 8. FUTURE WORK

### 8.1. Persistence

The application can be expanded by introducing a database to save player's progress and achieve persistence between gaming sessions. This functionality was a part of the project proposal, however after careful consideration, it was decided that implementation of this feature should become a separate independent research. Successful design must consider various computing aspects including security, authentication algorithms and server creation, which are beyond the scope of this project.

In order to make future work simpler, all the information which must be persistent was separated into four classes. The result was tested by successfully keeping track of player's progress after scene transitions.

### 8.2. Live testing

A testing with a group of children is required to obtain the results on:

- a) Educational benefits of the short-term and long-term use of the game
- b) Impact of the prototype on the physical activity level
- c) The level of engagement which users show towards the game (e.g. average play time per session)

### 8.3. Custom installer

To handle the huge amount of sprite assets outside the editor, a custom installer is required. This would allow pointing the game to a pre-set folder location rather than to a prefab folder, which would make the game playable outside Unity Engine as a standalone executable.

## 9. REFLECTION

Overall, I believe this project to be a success. The final product meets the key requirements defined in the proposal, it is delivered on time and does not contain any major bugs or issues.

The literature review was very challenging at first due to my little knowledge of designing serious games or exergames. The more I read, the more questions I got, however once the first design ideas were put into places, the research became more focused and productive. Instead of reading around the subject, I could search for approval or disapproval of my theories and alternative approaches. This had great impact on the development process. I will use this approach more in the future.

The programming part was more familiar. Throughout university coursework I learned various ways of solving code related issues and was happy to see how gained skills and knowledge could be utilised in this research. A few unexpected problems appeared at the late stage of the development, mainly associated with finding appropriate art and audio assets. I greatly underestimated how time-consuming this part can be, however providing a uniform look to the prototype made a massive positive impact on perception of the game at the end. Another challenge was refactoring and variable naming. Some parts of the game, such as bonus challenge, were heavily modified on multiple occasions, which made some of the class and game object naming obsolete and confusing.

The avatar creation scene is something I am particularly proud of. Managing a large number of assets was a new programming challenge for me and I am happy about the way implementation is handled.



## 10. BIBLIOGRAPHY

Adobe, 2018. *Adobe Photoshop CC*. [Online]

Available at: <https://www.adobe.com/ca/products/photoshop.html>

[Accessed 19 April 2018].

Axosoft, 2018. *GitKraken*. [Online]

Available at: <https://www.gitkraken.com/>

[Accessed 19 April 2018].

Brodie, A., 2013. *Let's So mental Maths*. London: Bloomsbury.

Carvalho, M. B., Francesco, B. & Riccardo, B., 2015. An activity theory-based model for serious games analysis and conceptual design. *Computers and education*, Volume 87, pp. 166-181.

Chuang, T.-Y. & Chen, W.-F., 2009. Effect of Computer-Based Video Games on Children: An Experimental Study. *Journal of Educational Technology & Society*, 12(2), pp. 1-10.

Cliona Ni Mhurchu, R. M. et al., 2008. Couch potatoes to jumping beans: A pilot study of the effect of active video games on physical activity in children. *International Journal of Behavioral Nutrition and Physical Activity*, 5(1).

Collins, 2017. *Mental Maths*. 1st ed. London: Collins.

Cyberbiking, 2016. *Cyber ExerCycle*. [Online]

Available at: [http://www.cyberbiking.com/buy\\_now.html](http://www.cyberbiking.com/buy_now.html)

[Accessed 27 January 2018].

Department for Education, 2014. *National curriculum in England: mathematics programmes of study*. [Online]

Available at: <https://www.gov.uk/government/publications/national-curriculum-in-england-mathematics-programmes-of-study>

[Accessed 20 April 2018].

Department of Education, 2015. *National curriculum in England: primary curriculum*. [Online]

Available at: <https://www.gov.uk/government/publications/national-curriculum-in-england-primary-curriculum>

[Accessed 29 April 2018].

Department of Education, 2017. *National curriculum assessments at key stage 2 in England, 2017 (interim)*. [Online]

Available at:

[https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment\\_data/file/624576/SFR30\\_2017\\_Text.pdf](https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/624576/SFR30_2017_Text.pdf)

[Accessed 29 April 2018].

Kafai, Y. B., 2009. World of Whyville: An Introduction to Tween Virtual Life. *Games and Culture*, V(1), pp. 3-22.

Kamran, S., 2008. From Play to Thoughtful Learning: A Design Strategy to Engage Children With Mathematical Representations. *Journal of Computers in Mathematics & Science Teaching*, 25(1), pp. 65-103.

Kapp, K. M., Blair, L. & Mesch, R., 2014. *The Gamification of Learning and Instruction Fieldbook*. 1st ed. San Francisco: Wiley.

Kids Health, 2014. *Are Video Games Bad for Me?*. [Online]  
Available at: <https://kidshealth.org/en/kids/video-gaming.html>  
[Accessed 18 April 2018].

Lu, A. S. et al., 2016. The Narrative Impact of Active Video Games on Physical Activity Among Children: A Feasibility Study. *Journal of Medical Internet Research*, 18(10).

Maddison, R. et al., 2011. Effects of active video games on body composition: a randomized controlled trial. *The American Journal of Clinical Nutrition*, 94(1), pp. 156-163.

Mellecker, R. R., Lisa, W. & Tom, W., 2013. Active Learning: Educational Experiences Enhanced Through Technology-Driven Active Game Play. *Journal of Educational Research*, pp. 352-360.

Mellecker, R. R., Witherspoon, L. & Watterson, T., 2013. Active Learning: Educational Experiences Enhanced Through Technology-Driven Active Game Play. *The Journal of Educational Research*, pp. 352-359.

Monedero, J., Lyons, E. J. & O’Gorman, D. J., 2015. Interactive Video Game Cycling Leads to Higher Energy Expenditure and Is More Enjoyable than Conventional Exercise in Adults. *Plos One*, 10(3).

Monique, S. et al., 2014. Active and non-active video gaming among Dutch adolescents: Who plays and how much?. *Journal of Science and Medicine in Sport*, 17(6), pp. 597-601.

Oypla, 2018. *Arm/Leg Mini Cycle Pedal Exercise Resistance Bike Fitness Gym*. [Online]  
Available at: [https://oypla.com/sports-leisure/indoor-sports/arm-leg-mini-cycle-pedal-exercise-resistance-bike-fitness-gym?gclid=Cj0KCQiAgZTRBRDmARIsAJvVWAtT\\_wgCcWoOd6NEFGbHXO0w9VIIeAuSzxZuGxKAoFV1rXVZEktWGQUaAvPZEALw\\_wcB](https://oypla.com/sports-leisure/indoor-sports/arm-leg-mini-cycle-pedal-exercise-resistance-bike-fitness-gym?gclid=Cj0KCQiAgZTRBRDmARIsAJvVWAtT_wgCcWoOd6NEFGbHXO0w9VIIeAuSzxZuGxKAoFV1rXVZEktWGQUaAvPZEALw_wcB)  
[Accessed 27 January 2018].

PEGI, 2017. *Pan European Game Information*. [Online]  
Available at: <https://pegi.info/>  
[Accessed 29 April 2018].

Peng, W., Lin, J.-H. & Crouse, J., 2011. Is Playing Exergames Really Exercising? A Meta-Analysis. *CYBERPSYCHOLOGY, BEHAVIOR, AND SOCIAL NETWORKING*, 14(11), pp. 681-689.

Prensky, M., 2007. Digital Game-Based Learning. In: *How Game Teach and How they Work*. s.l.:Paragon House, pp. 145-180.

Prensky, M., 2007. Digital Game-Based Learning for Kids. In: *Digital Game-Based Learning*. s.l.:Paragon House, pp. 180-200.

Prensky, M., 2007. Fun, Play and Games. In: *Digital Game-Based Learning*. s.l.:Paragon House, pp. 106-145.

- Prigg, M., 2014. *How the iPad replaced the toy chest: Researchers find children play with touchscreens more than traditional toys*. [Online]  
Available at: <http://www.dailymail.co.uk/sciencetech/article-2565061/How-iPad-replaced-toy-chest-Researchers-children-play-touchscreens-traditional-toys.html>  
[Accessed 18 April 2018].
- Schofield & Sims, 2017. *Mental Arithmetics 3*. 6 ed. London: Page Browns (Norwich) Ltd.
- Simons, M. et al., 2015. Replacing Non-Active Video Gaming by Active Video Gaming to Prevent Excessive Weight Gain in Adolescents. *PLoS ONE*, 10(1).
- Skoric, M. M., Teo, L. L. C. & Neo, R. L., 2009. Children and Video Games: Addiction, Engagement,. *CyberPsychology & Behavior*, 12(5), pp. 567-572.
- Statista, 2018. *Use of games consoles and players among children to consume media at home in the United Kingdom (UK) from 2013 to 2017, by age group*. [Online]  
Available at: <https://www.statista.com/statistics/274418/use-of-games-consoles-and-players-among-children-in-the-uk-by-age/>  
[Accessed 18 April 2018].
- Tynan, S., 2013. *Designing Games: a guide to engineering experiences*. 1st ed. Sebastopol: O'Reilly.
- Ukie, 2018. *The games industry in numbers*. [Online]  
Available at: <http://ukie.org.uk/research>  
[Accessed 18 April 2018].
- Unity 3d , 2015. *Unity 2D*. [Online]  
Available at: <https://docs.unity3d.com/ru/current/Manual/Unity2D.html>  
[Accessed 19 April 2018].
- Unity 3d, 2018. *Unity 3d*. [Online]  
Available at: <https://unity3d.com/ru>  
[Accessed 19 April 2018].
- Visual Studio, 2017. *Visual Studio IDE*. [Online]  
Available at: <https://www.visualstudio.com/ru/vs/>  
[Accessed 19 April 2018].
- Whitton, N., 2010. *Learning with digital games*. 2010 ed. New York: Routledge.