

Zombie Run

By Rob Wakefield

Single player fun and interactive game which aims to aid learning of Computer Science insioutside de and of the classroom.

Contents

Analysis

Introduction To The Program	3
Computational Features Of The Program	3
Suitable Stakeholders	4
Researching existing Solutions	6
Essential Features	14
Requirements List	15
Limitations	16
Hardware and Software Requirements	17
Success Criteria	17
Bibliography	19

Analysis

Introduction To The Program

My project will be a 2D game where the user can control a player using the device's peripherals. The game will involve differing types enemies which the user will have to avoid in order to progress further in the game. As the user defeats the enemies, the game will progressively become harder, leading to a more challenging game. Certain algorithms will be available to the user in order to help them to become better at the game and spark interest in computer science. Upgrades can be gained and purchased using in-game currency which will allow the user to progress through this casual style and educational game.

Computational Features Of The Program

Feature

Collision Detection

The location of borders of objects must be tracked and many calculations must be made each second to check if these borders overlap. The program should then react to this information immediately - faster than a human could and will always be accurate.

High Score / Leaderboard

The high scores of users will be stored in the computer's memory and be displayed to the user. This will be able to be reordered and altered quickly and easily. Progress may also be tracked over a time period which could not be done efficiently with many users without using a computer system.

User Input and Control

The program will detect and respond to user input from peripheral devices in order for the user to control the character on screen instantly and accurately. The user will not have to physically move their character and so can perform a more varied amount of motion.

Experience Gain / Levels

As the user progresses through the game the difficulty will increase which can be changed easily by using a computer – this would be difficult to do in a non-computational way as keeping track of levels

and which different enemies should be created would not enhance the game's experience.

In-Game Money / Upgrades

The user will be able to store and use virtual currency in order to upgrade certain aspects of the game. This money can be generated and tracked easily using a computational method. This also allows for an unlimited amount of money to be theoretically gained, which would no be possible using physical money.

Enemy movement algorithms

The way that enemies move around the screen during gameplay will be shown to users in algorithm/pseudocode form - encouraging and interest in programming while letting some players who understand the code gain a competitive edge by knowing how the enemies move. This obviously would be suited for a computational approach as the movement algorithms are carried out by computer system.

Suitable Stakeholders

Stakeholder 1

Caleb Tham

Age: 17

Occupation: Computer Science Student

Organisation: Bournemouth School

Caleb is a bright and interested student who enjoys playing casual games which he can compete against others in. He also likes to improve his programming abilities in fun ways and playing the programming project game will allow him to combine these two activities. Caleb is a curious student and would be interested in finding out how the game works.

How Caleb will use the Solution

- Relaxing and enjoying a fun to play game
- Developing his interest in computer science
- He will compete against his friends to reach highscores in the game

Why it is appropriate to Caleb's needs

- Resembles games which Caleb likes
- Engages competition between Caleb and his friends
- Further his interest in computer science

Features Caleb would like

- Includes programming features to interest him in computer science outside of the classroom
- Fun to play
- Appealing user interface

Stakeholder 2

Mr Albanozzo

Age: 52

Occupation: Computer Science Teacher

Organisation: Bournemouth School

Mr Albanozzo teaches computer science at Bournemouth School to students from years 7 to 13. In order to interest his younger students and to increase the number of students taking computer science at GCSE level and higher, he uses non-traditional methods which appeal to them.

How Mr Albanozzo will use the Solution

- Peak his students interests by allowing them to play the game
- Show the students how computer science can be used to create interesting projects
- Allow the students to modify the source code and change the game

Why it is appropriate to Mr Albanozzo's needs

- Will interest younger students in computer science through its exciting and challenging gameplay
- Certain algorithms used in the game will be easily accessible to the students so that they can see how the game works and how it could be changed.
- The source code will be available and well documented so the students could easily recreate the game.

Features Mr Albanozzo would like

- Shows algorithms to help him teach the class and show the students how it was made
- Engages the students so that they will be interested
- Varied enemy strengths to show how flexible programming can be

Stakeholder 3

Student

Age: 16

Occupation: GCSE student

Organisation: School

This student is currently taking part in his GCSE courses and has exams coming up. He struggles to revise for these exams without taking a break and balancing his work and leisure time. He enjoys playing casual games as he finds that more intense games cause him to only focus on the game and lose track of time - often playing these types of games for too long. He enjoys a challenge so harder games appeal to him.

How the student will use the Solution

- Take a break from GCSE revision and play the game for short periods
- Enjoy and have fun, keeping his morale up during revision
- Compete against his friends in a friendly way

Why it is appropriate to the student's needs

- The game is suited to being played at short amounts of time, meaning the student does not need to spend too long on the game
- The game offers a relaxing whilst entertaining experience
- A leaderboard allows him to keep track of his progress as well as his playing time so that he does not spend too much time on the game and less time revising.

Features the student would like

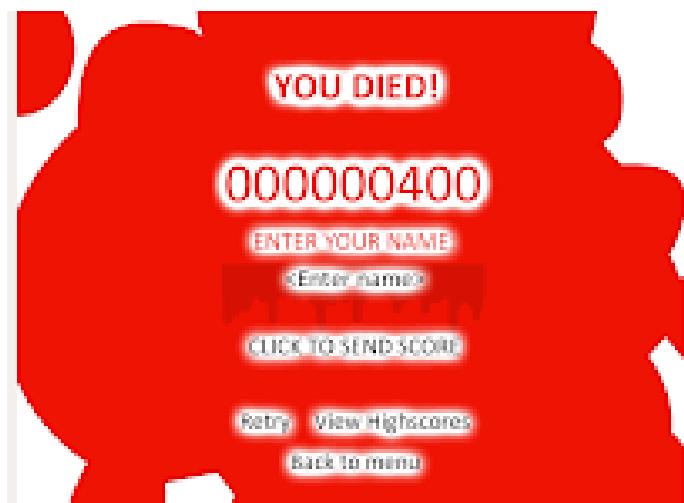
- Enjoyable and exciting gameplay that will entertain him over long periods of time
- Contains a leaderboard so he can compete against friends
- Has a way to progress through the game's levels over time

Researching existing Solutions

Boxhead

Overview

This game is an Adobe Flash Player web game hosted at <http://www.crazymonkeygames.com/Boxhead.html>. When the game is initially loaded, the user is presented with options such as to view highscores of previous attempts and other users and also to play the game.



After a user clicks play, the main gameplay begins - the aim of this game is to defeat timed 'waves' of zombies by killing them, whilst saving the 5 'hostages' which follow the user's character. The user controls the player with the arrow keys and spacebar to shoot which are easy to use and simple controls to learn when you first start playing the game. In order for the player to aim where they shoot they must carefully use the arrows to rotate their player. I found this feature very tedious and annoying because I could not get the angle that I wanted most of the time in order to shoot the enemies - in my solution an easier and more fluid way to rotate your character will be used.

As the game progresses, the amount of zombies and other enemies spawned increases and the difficulty of these enemies also increases with some of them being able to attack the player from a longer range which makes it harder to defeat the enemies without dying. The game, however, starts off quite slowly as hardly any enemies are created in the first few levels which I would change in my implementation, but as the levels progress, the game does become more challenging.

During the game, the player can pick up new weapons and upgrades as their score increases, which helps them to defeat the enemies more easily. However, if you do not deal damage to the enemies after a while your score starts decreasing and you lose some upgrades. This means that you cannot just hide but you must engage and concentrate on the game. You are able to pause the game and check your score/upgrades or take a break which I think is an important feature. When the user dies, the end screen shows which displays the score the user gained and allows them to save this score with their name so that others can view it.

Although the game is simply about defeating enemies, the upgrades that you can acquire and the increasing difficulty make this an entertaining game to play. I think that the 'save the hostages' part of the game distracts from the otherwise fun to play game. I think that the leaderboard is a great way to encourage people to continue playing the game - but, unlike my project, this game lacks any educational interest apart from displaying what computer science can be used to create. I think that a better understanding of the game

mechanics would lead to a much more enjoyable game experience.

Applying knowledge gained to this project

Positive approaches

Feature	Approach to solution	Justification
Highscore menu	I plan on including a leaderboard in my solution, so a way to keep track of highscores would be useful in my solution.	This would allow users of the game to track their progress and compete against other players of the game. A graph to show a user's progression might help to appeal to the user.
Increasing difficulty levels / waves	My solution will constantly become harder as the user reaches higher levels.	This makes the game exciting to play as the game progressively gets more challenging as the user gains a higher score.
Ability to gain upgrades	My solution will include a way to upgrade your character so that it becomes easier to reach higher scores.	This also adds another way in which the game is more exciting as the player is able to increase how good their character becomes at the game, letting them reach new higher scores.

Negative approaches

Feature	Approach to solution	Justification
'Hostages'	My solution will not include any other characters which the user must protect	These are distracting to the main gameplay as the user is not focussing on the main focus of the game. I think that the player should only need to protect themselves in a game like this as protecting other characters makes the game too challenging

Approaches to improve

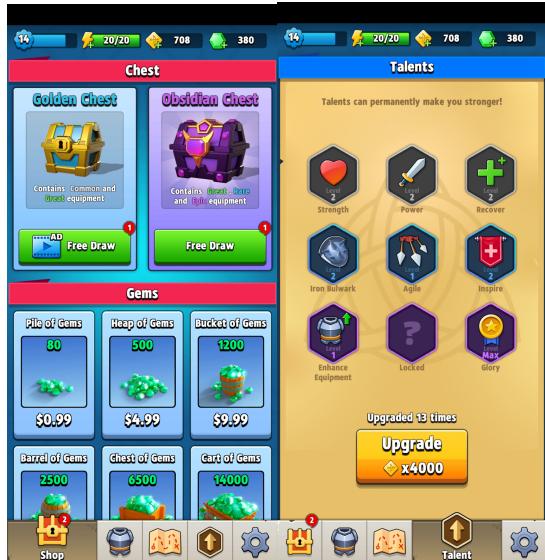
Feature	Approach to solution	Justification
Pace of the game	My solution will be more difficult at the beginning than the game I am researching and then become progressively more difficult	Although during later levels the game becomes challenging as more and more enemies are created, the first few levels are too slow and sometimes boring. In my project I would start the initial levels with a quicker amount of enemies being produced.

Archero

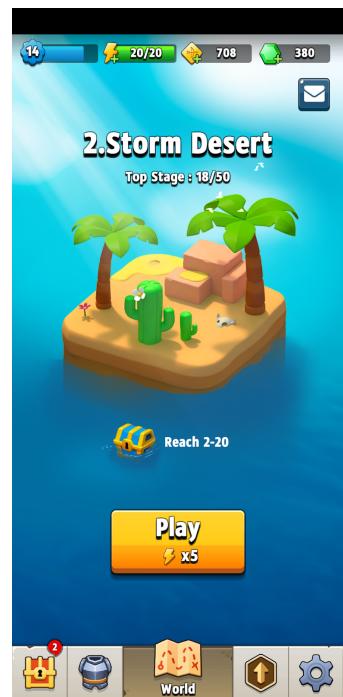
Overview

Archero is a mobile game which is played in portrait mode. The aim of the game is to complete levels by running from the bottom of the mobile screen to the top, but in order to access the next level, all the enemies on the current level must be killed. Each level is quite small and only takes about 30-60 seconds to complete, but the game becomes difficult as your health does not get recharged as you pass a level and so you must be careful not to get shot or injured by the enemies. There are a wide variety of enemies on this game, each with different abilities such as being able to shoot arrows or jumping onto your character, dealing damage. The enemies in this game are seemingly randomly spread out throughout the game and the levels are not the same each time you play. This leads to an exciting gameplay as each level is unexpected, but does not lead to a consistent play through of the game and that

might be why there is no leaderboard in this game.



In this game, you have the opportunity to upgrade your character. You can upgrade different aspects and abilities such as max health, damage per attack, defence/shield and how much you can recover health by. These upgrades, called Talents, can be obtained by using the in-game currency of tokens. After each upgrade, the cost to upgrade increases. These tokens can be purchased using real money in the shop section and can also be unlocked in gameplay and as special rewards. The game includes 'cheats' which can be opened after a certain period of time. The chests can include extra upgrades that are not available when



upgrading normally using tokens. These special upgrades include armour and weapons which alter not only how your character looks but also gives them extra abilities such as having the random chance to dodge an attack that hits you and also decreasing the damage you take.

Another element of this game is the experience levels. After every level you complete, depending on how far you get, you can gain experience points which go towards your experience level. After the experience bar is full, you reach the next experience level and you gain rewards for doing this. These rewards include tokens and sometimes items which help you to upgrade your character further. I think that this is a good way of increasing the amount of time spent by a user playing your game - and in this case how much money can be spent on the game. There is, however a low limit to how many games you can play in a certain period of time, but this can be bypassed by the user paying to unlock this feature.



Applying knowledge gained to this project

Positive approaches

Feature	Approach to solution	Justification
Series of small levels	I will include many smaller levels, rather than few bigger levels.	This ensures that the user is less likely to get bored when playing the game and will engage my target audience with the game.
Variety of enemies	My solution will include different enemies with different abilities such as more strength/damage per hit.	In order to create a fun and exciting game the different enemies will engage the audience and make sure that the game does not become too repetitive.
Experience levels	My solution should include a reward as the user progresses and plays the game more.	This is another way to make the game more exciting and give the user a reason to play more.

Negative approaches

Feature	Approach to solution	Justification
Upgrades	I will allow the user to upgrade their character in different aspects of the game such as health and damage per hit.	Unlike the game I will only be implementing features to upgrade that are permanent. I think that this will give the user a sense of progression as they play my solution over time.
In-Game currency	The user will be able to gain in-game currency by playing the game and reaching further levels. This will then be able to be spent on upgrades	This could add a revenue source for my game as in the future the user could be able to purchase in-game currency using real money. Also, as the user gains the currency they will likely feel accomplishment when buying upgrades.

Approaches to improve

Feature	Approach to solution	Justification
Randomly generated levels	My solution will include randomness to the levels, however it should not be so random that the difficulty of the levels is very different between two users of the game so that a leaderboard can be implemented.	Although during later levels the game becomes challenging as more and more enemies are created, the first few levels are too slow and sometimes boring. In my project I would start the initial levels with a quicker amount of enemies being produced.

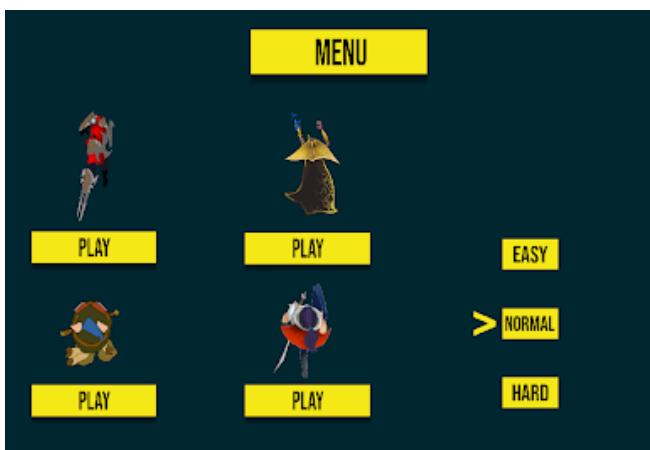
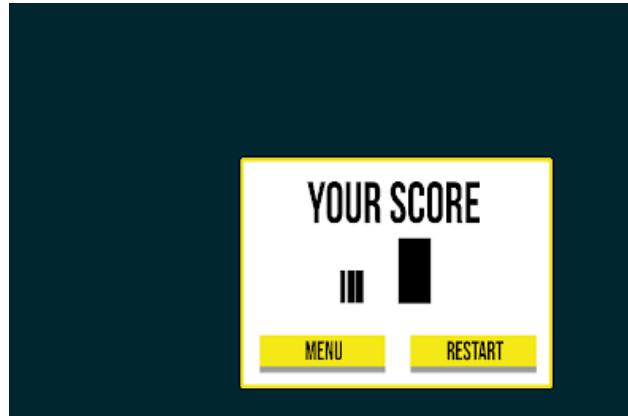
League of Dodging

Overview



League of Dodging is a browser game hosted at <https://loldodgegame.com/play/>. It is a simple but challenging game that involves moving a character around the screen by clicking with the mouse where you want the character to walk towards. The aim of the game is to dodge the incoming enemies which fly towards you and survive as long as you can. When you first play the game, you are able to choose one of four different characters and a difficulty setting - Easy, Medium or Hard. Once you press the play button the game starts instantly, which can be hard sometimes if you are not expecting to need to control the character so soon and you are not accustomed to the surrounding that your character is in yet. To move the character you right-click in order to walk slowly towards the cursor. If you are walking towards a destination and you click in a different location, a new target is set and the character moves towards the new cursor location.

A feature that this solution integrates is an ability to ‘teleport’ to the current mouse position instantly by pressing the keys ‘F’ or ‘D’. This feature is very useful for when you seem trapped and need to dodge an enemy quickly - it acts as a last resort to stay alive a bit longer. However you need to be careful where you mouse cursor is when you use this feature as you may teleport in to the path of an oncoming enemy, and if you do this you cannot just teleport away as the teleport feature has a cooldown period of about 10 seconds before you can use it again. I think that the benefits of this feature are great for the user as an extra help, but the negatives, such as the cooldown help to balance the feature so that it cannot be abused by the player and break the game.



Although the game does keep track of a highscore locally, there is no leaderboard to compare with previous attempts and other players. When you refresh the page the highscore also resets which does not suit having long periods of time in between plays of the game. In my solution I would include a leaderboard which tracks highscores and stores them even when the program is closed.

Applying knowledge gained to this project

Positive approaches

Feature	Approach to solution	Justification
'Dash' ability	I will allow the user to instantly travel to the current location of the mouse cursor using an input by the user. There will be a cooldown period after its use.	This adds a feature which when used well can be very effective, but is also not so good as it affects how enjoyable and challenging the game is.

Negative approaches

Feature	Approach to solution	Justification
Difficulty increase	My solution will have a gradual increase in difficulty, unlike this solution which starts off very hard and continues to get harder - however I could not last more than about 30 seconds	In order to make the game less frustrating and more enjoyable to play as if you die every 30 seconds you soon become bored of the game.

Approaches to improve

Feature	Approach to solution	Justification
Movement by mouse	I will use the mouse position on the screen as a constant target for the player's character instead of needing to click the screen to move.	I think that this will lead to a more fluid gameplay which is more enjoyable as the position of the character on screen can be more precisely controlled.
Simple motion of enemies	In my solution the enemies will travel towards and use the player as a target - unlike the random, unchanging movement in this solution	This will lead to a more challenging game and allow me to include the algorithms for movement which aids the computer science aspect.

Essential Features

Feature	Explanation of Feature
Features which appeal to users who like computer science	<ul style="list-style-type: none"> → The game should include aspects such as explanations of algorithms used in the gameplay in order to interest users who study computer science. → These should be easily accessible in the main menu of the game
Engaging gameplay with users	<ul style="list-style-type: none"> → The gameplay should be made fun and enjoyable by interacting with the user and allowing them to progress through accomplishments in the game. → The users should want to continue playing the game after they have played it once.
Varied ability of enemies	<ul style="list-style-type: none"> → The enemies health and attack power will vary by level, increasing in the higher levels. → The movement of the enemies will be different depending on the type of enemy they are.
Leaderboard / Progression tracker	<ul style="list-style-type: none"> → The solution should include a leaderboard or a progression tracker which shows the player's ranking/how the player's statistics have changed whilst playing the game.
In-Game upgrades	<ul style="list-style-type: none"> → There will be an option for the user to upgrade their character in order to help them to beat the higher levels more easily. → These upgrades can be purchased using in-game currency or through completing levels.
Multiple levels	<ul style="list-style-type: none"> → As the user plays the game more, the levels will become more challenging which should continue to engage the user as they become better at the game and their character is upgrading to have higher abilities.
Mouse and keyboard controlled input	<ul style="list-style-type: none"> → Responsive and intuitive control of the character on-screen will be required by the user in order for them to enjoy and learn the controls of the game.
Special abilities	<ul style="list-style-type: none"> → There must be a way for the player to gain special abilities in the game which help them defeat the enemies.
Score display throughout levels	<ul style="list-style-type: none"> → The user should be able to view their current score when they are playing the game.

Pause function

- The game can be paused in order for the player to take a break or rest.

Main Menu

- Before the user starts playing the game there will be a main menu which allows them to access different parts of the program.

Requirement List

Feature**Explanation of Feature****Features which appeal to users who like computer science**

- There will be a section in the game where the player can view the algorithms used in the program.
- There will be a tab to switch between native source code and pseudocode.
- This section will be accessible via the main menu.

Engaging gameplay with users

- There will be levels which will become harder and more challenging which will engage the user.

Varied ability of enemies

- The enemies will increase their health by 5% each increment in level.
- The enemies will increase their attack damage by 2% each increment in level.
- The enemies will increase their speed by 2% each increment in level.

Leaderboard / Progression tracker

- The leaderboard will show the top 10 players stored on the computer and their scores.
- The current user will be able to check how their overall score has changed over time.
- The player's data will be stored in a csv file on the computer.

In-Game upgrades

- There will be an upgrade tab in the main menu screen.
- The upgrades will be purchased by in-game currency.

Multiple levels

- Each level will have a wave of enemies that need to be defeated.
- After each level the user can check their score and continue playing.
- After each level the abilities of the enemies increase.

Mouse and keyboard controlled input

- The player will click on buttons with the mouse and use the keyboard to enter text values.
- The mouse will be used to control the player's movement

during the game.

Special abilities

- There will be the option to buy/pick up a special ability during the game.
- This ability will enhance a certain attribute of the player.

Score display throughout levels

- The score will be displayed at the top of the screen.
- The score will increase as enemies are defeated.
- The score will be displayed at 'game-over'.

Pause function

- Using the 'P' key on the keyboard during the game will cause the game to pause and the score and options to quit the game will be shown.

Main Menu

- There will be buttons for playing the game, viewing leaderboards, viewing algorithms.
- Statistics such as score and in-game currency amount will be shown at the top of the screen.

Limitations

Limitation

Explanation of Limitation

Leaderboard

- As communication over the internet is above my programming scope, the program will not include an online leaderboard as in some of the products researched.
- The leaderboard will be locally stored and only players who play on the computer system the program is installed on.

Online gameplay

- There will not be multiplayer support as communicating over the internet in real time is above the scope of this project.
- Local multiplayer will also not be implemented as the mouse is used to control the player on the screen.

Encryption of leaderboard/data

- The game is not intended to be secure, but is meant to be fun to play.
- Therefore the leaderboard and player data will not be encrypted when it is stored on the machine.
- The csv file will still be checked to see if it is in a valid format before reading it.

Hardware and Software Requirements

Component	Requirement
CPU	3GHz Dual-Core
GPU	Integrated graphics or above
RAM	2GB or above
OS	Windows XP - Windows 10 Mac OSX Linux
Other	Display, Mouse, Keyboard Must be able to run Java Runtime Environment

Requirements are based on the reported requirements of Processing from <https://github.com/processing/processing/wiki/Supported-Platforms>.

Success Criteria

Criteria	Explanation of Criteria
Does the program have an element of Computer Science?	→ Is there a section which shows the user code/pseudocode of certain game algorithms?
Do the enemies change in ability as the game progresses?	→ The enemies should increase their health by 5% each increment in level. → The enemies should increase their attack damage by 2% each increment in level. → The enemies should increase their speed by 2% each increment in level.
Does the leaderboard show the correct standings and information?	→ The leader board should show the top 10 player's scores in descending order from the top. → The correct score should be displayed with the correct user.

Does the progression tracker store and display player stats correctly?

- The graph should scale correctly.
- The data for each user should be accessible.

Can the in-game upgrades be purchased?

- Can the user purchase upgrades using the in-game currency?
- Can multiple upgrades be purchased?

Are there multiple levels accessible by the user?

- Does the level increment after a level has been cleared?
- Does the level remain when the level was failed?

Does the mouse control the character movement?

- Does the player's character follow the mouse during gameplay?

Are the special abilities available to be picked up and do they aid the user?

- Do special abilities have an effect on the user?
- Is the effect positive?

Does the score at the top of the screen update correctly?

- Does the score continuously show the correct and updates score?

Can the user pause the game?

- When the user presses the 'P' key the game should show a pause screen.

Is the main menu functional?

- Do all the buttons function and produce a response?
- Are there options to go to the leaderboard/progression and code view screens?

Bibliography

Analysis

<http://www.crazymonkeygames.com/Boxhead.html>

→ Accessed on 19/06/2019

<https://loldodgegame.com/play>

→ Accessed on 19/06/2019

<https://github.com/processing/processing/wiki/Supported-Platforms>

→ Accessed on 08/07/2019