Fall 2021 Principles of Software Engineering

FAU CEN 4010

**The Vaccine Strikes Back**

Group 23

Milestone 3

October 25, 2021

**Team Members**

Robert Johnson (Product Owner)

[rjohnson2018@fau.edu](mailto:rjohnson2018@fau.edu)

-

Tony Rojas (Scrum Master)

[trojas2019@fau.edu](mailto:trojas2019@fau.edu)

-

Renan Quiroz (Backend)

[rqueiroz2019@fau.edu](mailto:rqueiroz2019@fau.edu)

-

Ajay Seedoo (Backend and Frontend)

[aseedoo2020@fau.edu](mailto:aseedoo2020@fau.edu)

-

Mykyta Palamarchuk (Frontend)

[mpalamarchuk2019@fau.edu](mailto:mpalamarchuk2019@fau.edu)

**Table of Contents**

1. [Revision History](#RevisionHistory) … (2)
2. [Executive Summary](#ExecutiveSummary) … (3)
3. [Competitive Analysis](#CompetitiveAnalysis) … (3-4)
4. [Data Definition](#DataDefinition) … (5-6)
5. [Overview, scenarios and use cases](#OverviewScenariosUseCases) … (7-12)
6. [List of High-level Functional Requirements](#FunctionalRequirements) … (13)
7. [List of Non-Functional Requirements](#NoneFunctionalRequirements) … (14)
8. [High-level System Architecture (UML)](#HighLevelArchitecture) … (15)
9. [Key Risks](#KeyRisks) … (16)
10. [Vertical Demo Testing](#VerticalArchitectureTesting) … (17)

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Revision | Description | Performed by |
| 10/24/2021 | 1 | Made changes that happened during development process | Mykyta Palamarchuk |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Click [here](#TableOfContents) to return to the Table of Contents

**Executive Summary:**

The impact that COVID-19 has brought upon the world is undeniably horrendous. There have been approximately 4.55 million deaths, and the death toll is still rising. Multiple vaccines have been put out, and as of September 1st, 2021 only 39.6% of the world's population has received a vaccination. Currently, 3 major vaccines are available: Pfizer, Moderna, and J&J.

We came up with the idea of our game “The Vaccine Strikes Back” not only to have a bit of fun in these tough times, but to try and help spread awareness of how the vaccine can potentially help stop the spread of the virus. Our game will be similar to how old arcade games function, but with a COVID-19 theme and some twists here and there. Basically, you will control a vaccine needle(i.e. the spaceship in “Space Invaders”) with the vaccine inside of it. This will be used as ammunition to destroy moving and falling harmful covid cells inside the body. If a harmful covid cell reaches the bottom of the screen or touches the vaccine needle, you will lose points. There will also be some cells that are too strong to destroy, which is where the in-game shop will come in. You will be able to buy power-ups and other items to aid the fight through the in-game shop. As you play the game, you will also be able to replenish ammo from vaccine droplets that will also fall from the top of the screen periodically. Our plan for the in-game shop is for the power-up items to be things that are known to be effective in fighting covid . There will be in-game descriptions for each item to help educate people about what they actually do. In conclusion, we came up with the design of “The Vaccine Strikes Back” to not only have a fun experience, but to also learn more about COVID-19 as well.

Click [here](#TableOfContents) to return to the Table of Contents

**Competitive Analysis:**

There are a few different competitor games which have been released with similar gameplay to the one in which we are creating. One of these is Space Invaders, a top down arcade classic released in 1978 where you played as a spaceship defending from an alien attack. Another example is Galaga which was released a few years after Space Invaders in 1981 with added features such as being able to take over an enemy spaceship and play as them. Another game like these which introduced a new mechanic was Asteroids. In Asteroids, you controlled a spaceship which had free movement across the screen instead of a fixed position and shot asteroids out of the sky. The table below summarizes the features included in other games and the differences between those and the one we are creating:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Features | Space Invaders | Asteroids | Galaga | The Vaccine Strikes Back |
| Fixed Spaceship | X |  | X | X |
| Currency/Shop |  |  |  | X |
| Endless | X | X | X | X |
| Varying enemies | X |  | X | X |
| Powerups |  |  | X | X |
| UI/Menu |  |  |  | X |

**Space Invaders:** Space Invaders involves controlling a spaceship located at the bottom of the screen. The spaceship can only move horizontally and shoots vertically based on its position at the bottom. A key difference between “Space Invaders” and “The Vaccine Strikes Back” is the ability to access a shop and UI to buy power ups within the game.

**Asteroids:** Asteroids use a unique control mechanism to control the spaceship to blow up asteroids. Unlike our game, Asteroids allows control of the spaceship on any part of the screen, losing the restriction of staying at the bottom of the screen. This opens up the field that the player interacts with.

**Galaga:** Galaga offers many of the same features found in our game including a powerup and different types of enemies. In Galaga, if you destroy a boss ship you can gain the boss ship as an upgrade to be attached to your ship, doubling your damage against enemies. A key difference here is instead of dropping upgrades from enemies, enemies will drop ammunition for our ship to use. The shop will be the primary place to find upgrades for our ship.

**The Vaccine Strikes Back:** The Vaccine Strikes Back is meant to fulfill all of the features of the previous competitors and add even more to this classic game. It is also meant to be a fun and educational game which can provide a learning experience unlike the previous games. The item descriptions will allow people to learn about real life parts in the creation of the vaccine and the menu and shop added into the game allows more control and freedom over the functions of the ship. We choose to keep a fixed spaceship design to allow a finer area where we can control what happens in the player space. This allows for a higher quality game which requires more skill than luck and controls which are simple for anyone to learn.

Click [here](#TableOfContents) to return to the Table of Contents

**Data Definition:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Meaning** | **Usage** | **Comment** |
| Player | Actor | Use Case Scenarios | A user who is registered with the game and playing it |
| Score | Data | Activity Type | Player score which changes depending on their performance |
| Enemies | Data | Use Case Scenarios | Interacts with the player by being their target, and also causing damage to player |
| Account | Data | Use Case Scenarios | Store users information such as email, password, username and scores |
| Username | Data | Use Case Scenarios | User’s identifier that is seen by other players on a leaderboard |
| Email | Data | Use Case Scenarios | User’s email address used to access a website |
| Password | Data | Use Case Scenarios | User’s password used to access a website |
| Ammunition | Data | Activity Type | Ammunition will change depending on how many shots were taken and how many refills were picked up |
| System | Platform Hardware and Services | Use Case Scenarios | The mySQL database,  all code, front end  design and back end  supporting services |
| https://lamp.cse.fau.edu | Production Server | Use Case Scenarios | The server that stores all of our data |
| Login System | Service | Site user service | Allows users to login and keeps track of their score and currency |
| Register | Service | Site user service | User will create an account with their email and password to access a game |
| Web Site | User Interface | User Interface | Front end display for user interaction |
| Pause Menu | User Interface | User Interface | Menu displayed when user pauses the game |
| Settings Page | User Interface | User Interface | Allows users to change game settings |
| Home screen | User Interface | User Interface | First page seen by a user |
| Leaderboard | User Interface | User Interface | Display scores of the highest scoring players in a system and compares them to the current player |
| Shop | User Interface | User Interface,  Use Case | Displays possible powerups that can be purchased by a player |

Click [here](#TableOfContents) to return to the Table of Contents

**Overview, scenarios, and use cases:**

**Use Case - Start**

User wants to start a game. User will come to a home page where he would use a “Play” button in order to be taken to a game screen to begin a game.

1. **Description:**

Use case will describe a process of how the user will start a game.

2. **Actors:**

2.1 User

2.2 System

3. **Preconditions:**

3.1 System is available

3.2 User has an internet connection

4. **Primary flow of events:**

4.1 User opens the application and arrives on a home page

4.2 User Performs Use Case - Login

4.3 User selects a play button

4.4 System displays a quick tutorial

4.5 User closes a tutorial and starts a game

4.6 Terminate Use Case: Start

**Use Case - Settings**

User wants to adjust game settings such as volume and account settings. User will come to a home page where he would use a “Settings” button to get to a settings screen.

1. **Description:**

Use Case will describe a process of how the user gets to the settings screen to adjust settings.

2. **Actors:**

2.1 User

2.2 System

3. **Preconditions:**

3.1 System is available

3.2 User has an internet connection

4. **Primary flow of events:**

4.1 User arrives at a home page

4.2 User selects “Settings” button

4.3 System takes user to a Settings page

4.4 User adjusts Settings according to their needs:

4.4.1 User may turn music and sounds on or off.

4.4.2 User may log in or log out of their account (perform Use Case: Login)

4.5 System saves users adjustments

4.5 User presses “close” button

4.6 System takes user back to home page

4.7 Terminate Use Case: Settings

**Use Case - Log in**

The user wants to login into their account to save progress and view the leaderboard. User would come to the “settings” menu and click the “Login” button.

1. **Description:**

Use Case will describe a process of user logging into an existing account

2. **Actors:**

2.1 User

2.2 System

3. **Preconditions:**

3.1 User has an internet connection

3.2 System is available

3.3 User has an account

3.4 User is in the settings menu

4. **Primary flow of events:**

4.1 If user arrives at a game screen for the first time

4.1.1 Go to step 4.4

4.2 User arrives at “Settings” menu

4.3 User clicks on a login button

4.4 System prompts user to enter their Login and Password information

4.5 User enters their information and clicks “Login”

4.5.1 If user doesn’t have an existing account, go to alternate flow 5.1

4.6 System logged user into a system

4.7 Terminate Use Case: Login

5. **Alternate Flows:**

5.1 User Does not have an existing account:

5.1.1 Perform Use Case: Create Account

5.1.2 Move to step 4.1

**Use Case - Create Account**

The user wants to create an account to save progress and view the leaderboard. User will go to the settings menu and click on “create account”.

1. **Description:**

Use Case will describe a process of a user creating a new account.

2. **Actors:**

2.1 User

2.2 System

3. **Preconditions:**

3.1 User has an internet connection

3.2 System is available

3.3 User is in the settings menu

4. **Primary flow of events:**

4.1 If the user opens a game for the first time

4.1.1 Go to step 4.4

4.2 User arrives at “Settings” menu

4.3 User clicks on Create Account button

4.4 System Prompts user to Enter a new Login and Password

4.5 User enters login and password information and clicks create button

4.6 System Checks for the availability of the Login

4.6.1 If login isn’t available go to alternate flow 5.1

4.7 System saves Account information

4.8 Terminate Use Case: Create Account

5. **Alternate Flows:**

5.1 If Login is take by another user

5.1.1 System will notify a user that login isn’t available

5.1.2 Go back to 4.3

**Use Case - Leaderboard**

The user wants to view their score on a leaderboard and compare it to other players.

1. **Description:**

Use case will describe a process of user opening and tracking a leaderboard

2. **Actors:**

2.1 User

2.2 System

3. **Preconditions:**

3.1 User has an internet connection

3.2 System is available

3.3 User has an active account

3.4 User is logged into their account

4. **Primary flow of events:**

4.1 User arrives at a home menu

4.2 User clicks on a leaderboard button

4.3 System takes user to a Leaderboard menu

4.3.1 If user isn’t logged in, go to alternate flow 5.1

4.4 User can scroll and see their position on a leaderboard

4.5 User clicks on “X” to go back to the home menu

4.6 Terminate Use Case: Leaderboard

5. **Alternate Flows:**

5.1 User isn’t logged in:

5.1.1 Perform Use Case: Login

5.1.2 Go back to 4.3

**Use Case - Shop**

The user wants to purchase improvements with an in-game currency to better progress in the game.

1. **Description:**

Use Case will describe a process of how the user will interact with a shop

2. **Actors:**

2.1 User

2.2 System

3. **Preconditions:**

3.1 System is available

3.2 User has an internet connection

4. **Primary flow of events:**

4.1 User arrives at a home menu

4.2 User clicks on a shop button

4.3 System takes user to a shop menu

4.4 User clicks on the item he wants to buy/upgrade

4.4.1 If funds are insufficient, go to alternate flow 5.1

4.5 System processes users selection

4.6 User can go back to step 4.4 if necessary

4.7 User can click “X” to go back to the home menu

4.8 System takes user to the home menu

4.9 Terminate Use Case: Shop

5. **Alternate Flow:**

5.1 If funds are insufficient

5.1.1 Notify user that funds are insufficient

5.1.2 Take user back to step 4.4

**Use Case - Shoot**

The user wants to shoot the vaccine in order to destroy Covid particles.

1. **Description:**

Use Case will describe how user will use the shooting function of the game

2. **Actors:**

2.1 User

2.2 System

3. **Preconditions:**

3.1 System is available

3.2 User has an internet connection

3.3 User is in the game menu

3.4 User has ammunition

4. **Primary flow of events:**

4.1 User arrives to a game screen

4.2 Covid Particles appear

4.3 System will shoot automatically

4.3.1 If there is no ammo, go to alternate flow 5.1

4.4 System will register the action and display shooting animation and shows ammo particle

4.5 System will decrease ammo count

4.6 Terminate Use Case: Shoot

5. **Alternate Flow:**

5.1 If there is no ammunition

5.1.2 Go to step 4.6

**Use Case - Collect Ammunition**

The user wants to collect vaccine particles in order to replenish ammunition.

1. **Description:**

Use case will describe a process of a player collecting the ammunition

2. **Actors:**

2.1 User

2.2 System

3. **Preconditions:**

3.1 System is available

3.2 User has an internet connection

3.3 User is in the game menu

3.4 Ammunition is on the game canvas

4. **Primary flow of events:**

4.1 User arrives on a game screen

4.2 System creates and displays ammunition particle on the canvas

4.3 User performs Use Case: Move

4.4 User hovers over the ammunition particle

4.5 System registers the interaction of the ammunition and a player

4.6 System increases the amount of ammunition

4.7 Terminate Use Case: Collect ammunition

**Use Case - Destroy**

The user wants to destroy Covid Particles in order to get points.

1. **Description:**

Use Case will describe the process of destruction of covid particles

2. **Actors:**

2.1 User

2.2 System

3. **Preconditions:**

3.1 System is available

3.2 User has an internet connection

3.3 User is in the game screen

3.4 System has created Covid particles

3.5 User has ammunition

4. **Primary flow of events:**

4.1 User arrives to a game screen

4.2 System created Covid particles

4.3 User performs Use Case: Move to line up with Covid Particle

4.4 User performs Use Case: Shoot

4.5 If ammunition particle touched Covid particle

4.5.1 System registers a hit

4.5.2 Covid particle life points will decrease

4.5.2.1 If life points reach 0 go to step 4.7

4.6 If life points are above 0, go back to step 4.3

4.7 System registers a destruction of a particle and hides it from the game screen

4.8 System registers points for destroying a Covid Particle

4.9 Terminate Use Case: Destroy

**Use Case - Move**

The user wants to move around the canvas in order to shoot and destroy covid particles, as well as collect ammunition.

1. **Description:**

Use case will describe all cases in which the user can move across the board to complete certain actions.

2. **Actors:**

2.1 User

2.2 System

3. **Preconditions:**

3.1 User is stationary on the canvas.

3.2 User has an internet connection

3.3 User wants to move to shoot the vaccine at a covid cell.

3.4 User wants to move to collect a piece of the vaccine to replenish ammunition.

4. **Primary flow of events:**

4.1 Covid cell appears on the user's screen.

4.2 User uses arrow keys on the keyboard to move the vaccine to shoot said covid cell mentioned in 4.1.

4.3 User uses arrow keys on the keyboard to collect a potential spawn of the vaccine to replenish ammunition.

Click [here](#TableOfContents) to return to the Table of Contents

**High-Level Functional Requirements:**

1. **Creating Account** 
   1. User should be able to sign up using their email and password that will be stored in a system.
   2. Creating an account must be the first action user performs (if they haven’t created account before) before they are able to enter the game.
2. **Log in**
   1. User should be able to log in to their account using an email and password, given that they are already stored in a system.
   2. This should be the first action performed by a user (if they have an existing account) before they are able to entre the game.
   3. User should be able to change their account in Settings menu.
3. **Report**
   1. User should be able to report a bug in an application in a form of a ticket that will allow developers to review and fix the code.
4. **Move**
   1. User should be able to move around a canvas during a playtime.
   2. Moving should allow user to avoid enemies.
   3. Moving should allow user to collect ammunition.
   4. Moving should allow user to line up the shot.
5. **Shoot**
   1. Shooting should be done automatically if ammunition is present.
   2. Shooting an enemy should be able to destroy it.
6. **Change Username**
   1. User should be able to create/change their username in a Settings menu
   2. Username will be displayed on a Leaderboard.
7. **Collect Ammunition**
   1. User should be able to collect and replenish ammunition to destroy enemies and improve their score.
   2. User will have to hover over ammunition to collect it.
8. **Score**
   1. Destroying an enemy should increase user score during playtime.
   2. Score should be saved and later displayed on a Leaderboard.

Click [here](#TableOfContents) to return to the Table of Contents

**List of Non-Functional Requirements:**

1. **Performance -** The performance of this product will not be demanding on hardware as it will be used in the web itself. There should also be no delay in the response time in regards to inputs for the controls.
2. **Reliability -** Since the product is web based, the main thing that is reliant on is that the user has access to the internet/wifi/service, and of course a computer or some other electronic device that can access it. We also plan to keep the product running non-stop unless there is an issue that needs to be addressed.
3. **Security -** For this product we will be using Firebase , which is a Google created software that helps with security and creating web applications such as this. Furthermore the security regarding the user data (passwords, ect) will be handled by Google through Firebase.
4. **Availability -** The product will be available in a short matter of weeks once we are able to launch it. Once it is released on the web , we will closely manage how it progresses and will decide if we will work on a non-web version through an app store. The product should be available 24/7 on the web once released unless it requires some maintenance.
5. **Usability -** For this web product the user is required to have some sort of internet access as well as a phone/computer/android device. Furthermore, if the user plans to use it via a computer it is also required to have access to a keyboard , as the product will not be usable with a mouse alone.
6. **Services and Manageability -** For this product we plan on using a MySQL Database in order to store the user’s login information as well as other data through actions on the website. This data will be managed by us the developers, as well as other services in case the user may have found a bug/exploit that they would like to report.

Click [here](#TableOfContents) to return to the Table of Contents

**High-Level System Architecture and Database Organization:**

1. **Browser Compatibility** - Our game will be a web-based application that will work on all major browsers.
2. **Visual Studio Code(IDE)** - Visual Studio Code is a code editor that we will be using to create the code for our web game application. The following languages will be used in our project:
   1. **Javascript** - Javascript will be the language used for the backend of the game to make it very interactive and enjoyable for the user.
   2. **HTML (HyperText Mark-Up Language)** - HTML will be the language used for the frontend to ensure that the UI(User Interface) is visually appealing and enjoyable for the user. HTML will allow the browser to display the game.
3. **fau.edu Lamp Server** -The FAU lamp server is what we will be using to host our project.
4. **P5.Play** - P5.Play is a Javascript library for the creation of games. This is what we will be using for all of our animations, interactions, collision detections, etc.
5. **Google Firebase** - We will be using Firebase to handle user authentication and storing information for our project. We will not be able to see user passwords, and encryption is handled on Google’s end. This seemed like the best option for our type of project.

**High Level UML Diagram:**

Click [here](#TableOfContents) to return to the Table of Contents

**Key Risks for Project At This Time:**

**Skill Risk/Actions:**

Not all members of our team are equally knowledgeable when it comes to coding and are often unaware of what is behind every change that happens to a program in a course of development. This risk is being mitigated by frequent online meetings in which everyone is sharing what they did and how they did it.

**Schedule Risk/Actions:**

Multiple members of our group are full time workers and students and are often overloaded with the amount of work they have which makes it quite difficult to perform their assigned tasks. This is mitigated by team members who have more free time taking on some extra tasks to complete the milestones in time.

**Technical Risk/Actions:**

Although some of our team members have made some simple games before, none of us had use Data bases to store user information which makes it difficult for us to work with any kind of Data base. This is mitigated by our Product Owner doing research on the subject and explaining basics to everyone during team meetings.

**Teamwork Risk/Actions:**

None of the team members have previously worked together on such big programming projects in a team which sometimes makes it difficult to properly divide the work and deadlines among people. This is mitigated by a Product Owner and a Scrum Master who keep track of everything that is happening and assist those in need.

**Legal-Content Risk/Action:**

We do not have any legal issues as all our code and assets are to be made specifically for our project.

Click [here](#TableOfContents) to return to the Table of Contents

**Vertical Architecture Testing:**

Working version of a code can be found here: <https://renanqueiroz.com/projects/vsb/>

The following functionalities can already be tested:

1. **Creating New Account**
2. **Log in to existing account**
3. **Move around a canvas to avoid enemies**
4. **Collision with enemies**
5. **Shooting function**
6. **Replenishing Ammunition**
7. **Reviewing a Score**

Click [here](#TableOfContents) to return to the Table of Contents