



Bilkent University  
Department of Computer Engineering

# Object Oriented Software Engineering Project Report

*Project short-name: Curve Fever*

## Analysis Report

Zeynep Delal Mutlu, Yunus Ölez, Barış Poyraz

Course Instructor: Bora Güngören

Jury Members: Jury Member Name and Jury Member Name

Analysis  
October 15, 2016

## **Table Of Contents**

<b>1.Introduction</b>	<b>2</b>
<b>2.Overview and Gameplay</b>	<b>2</b>
<b>2.1.Power-ups</b>	<b>4</b>
<b>2.2.Player</b>	<b>4</b>
<b>2.3.Key Configurations</b>	<b>5</b>
<b>3.Requirement Specification</b>	<b>5</b>
<b>3.1.Functional Requirements</b>	<b>5</b>
3.1.1.Play Game	5
3.1.2.Pause Game	6
3.1.3.View Help	6
<b>3.2.Nonfunctional Requirements</b>	<b>6</b>
3.2.1.Performance	6
3.2.2.Arcade& Last One Standing Themed	6
3.2.3.Offline Multiplayer	7
3.2.4.User-Friendly Interface	7
3.2.5.Color Selection	7
<b>3.3.Constraints</b>	<b>7</b>
3.3.1.English Language	7
3.3.2.Java	7
3.3.3.Game Start	8
<b>4.System Models</b>	<b>8</b>
<b>4.1.Scenarios</b>	<b>8</b>
4.1.1.Play Game Success Scenario	8
4.1.2.Help Success Scenario	8
4.1.3.View Credits Success Scenario	8
<b>4.2.Use Case Model</b>	<b>9</b>
4.2.1.Play Game	9
4.2.2.View Help	10
4.2.3.View Credits	11
4.2.4.Pause	11
<b>5.Glossary &amp; References</b>	<b>12</b>

## ***1.Introduction***

The game we decided to develop is Curve Fever which is an arcade game. Curve Fever is a game which can be played with at least 2 people. It is an offline multiplayer type game which can be played in the same computer.

Information about the game is in Overview and Gameplay part but, mainly, there are players, which are points in a map, where these points leaves a solid pattern, a pattern of their color.

The objective of the game is not to cross each others patterns and the boundaries of the map. The winner is also be the last player standing in the game. Also power ups might appear in randomly, which may either do harm or boost the player. The game we do replica is following:

“Curve Fever 2”

<http://forum.curvefever.com/play2.php>

The game will be a desktop application and will be played using keyboard.

## ***2.Overview and Gameplay***

As stated earlier, Curve Fever is an offline-multiplayer arcade game. The game is easy and fun to play. The entertainment you get in this game is directly proportional with the number of players. As the number of players increases, the game becomes more enjoyable.

In this game, the players are considered as points. The players are allowed to choose their color. The important thing here is that, each player shall have a different color than the others. When the game starts, the players are dropped in a map which has a black background. As the game starts, each point, players, starts to move to forward automatically, which gives players control over their left and right movement, and leave a pattern behind themselves while giving some gaps randomly. These gaps can be used for advantage during game. The players can turn left or right by using keyboard buttons.

Another description for the game is as follows:

“To briefly explain, “Curve Fever”, is a free, online multiplayer game where players have the ability to turn left or right leaving a permanent, solid line behind them in their colour. If a player collides with another player’s line or the outside boundary, the player loses. Gaps in the line are randomly generated, which players can use to their advantage. The goal of the game is to be “the last player standing” and have the most points!”(Curve Fever Wikia).

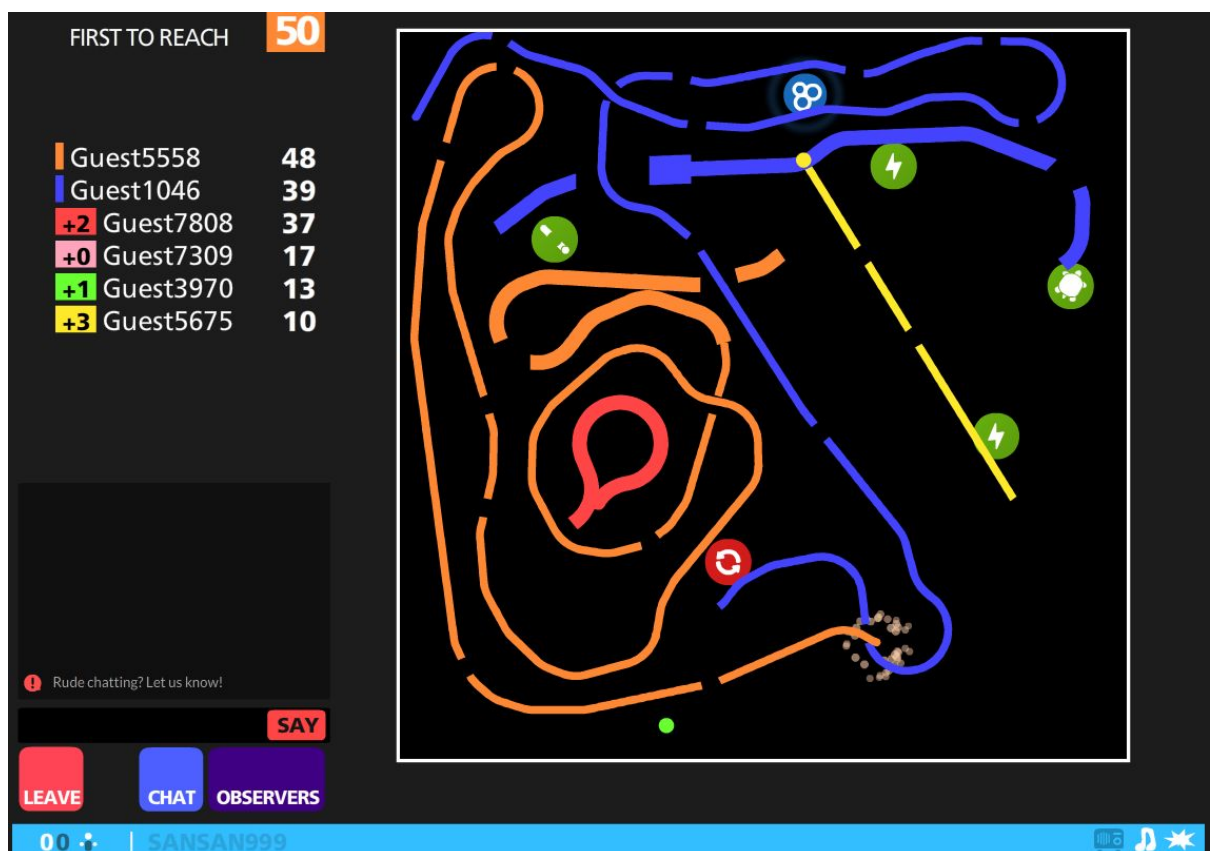


Figure 2.i

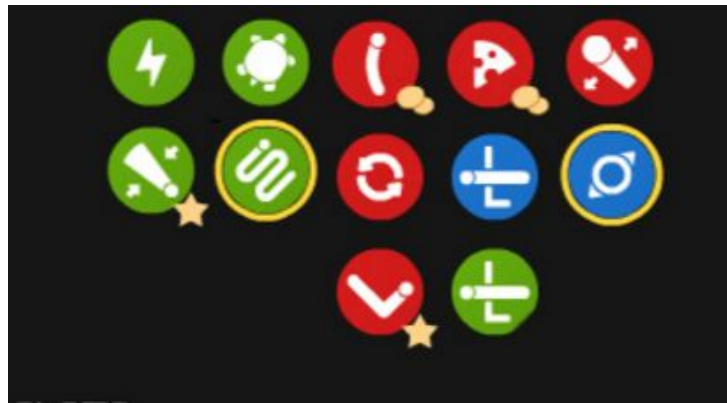
An image from the original Curve Fever found in Google Images. The original game is actually online which requires internet connection but in our version, it will be offline.

As seen from the figure above, there are 6 players playing the game. Each of the player has names on the left side of the figure with their colors. The players which can also be told as players leaves a pattern, a solid line behind themselves. From the left part of the figure, where we see number of points gained from the round, we can understand that, during that part of gameplay, orange and blue points continue playing. For example, the yellow point which leaves a yellow line behind it got 3

points, which means that it collided to one of the colors, blue in that case, after third elimination. The power-ups and players are discussed deeper in the subsections 2.1 & 2.2.

## ***2.1.Power-Ups***

During the gameplay, power-ups are very crucial. The power-ups pop up during the game within the boundaries from the set of power-ups that are created earlier. The power-ups are very crucial because, they can change win situation in any game. There are two types of power-ups. These differ in terms of their background color. Green power-ups have an effect on the player who touches to it whereas red power-ups have an effect on the other players and it doesn't not affect the player who takes it. There will be different types of power-ups, which will be taken as an example from the original game and implement. Examples of power-ups can be seen from the figure 2.i and from the image below



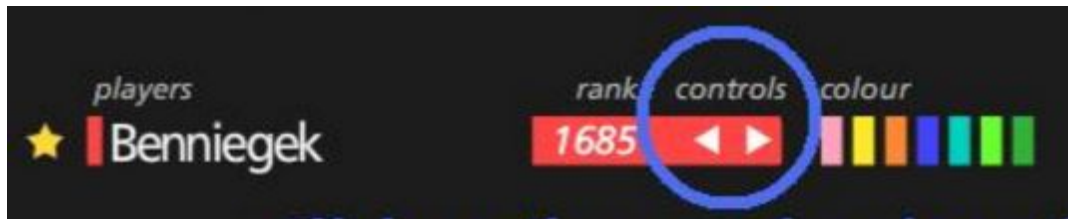
## ***2.2.Player***

Each player or points has a unique name, color and key configurations to play the game with in order to participate and play the game. During the player selection part, each person will have the opportunity to create a player with these specific requirements. Meanwhile, in the game, the players start in an area, which has boundaries colored with color white. Each user can control it's point by using the key

configurations they decided in player selection stage. Forward movement will be done automatically. Players can give left and right directions to their point.

### ***2.3.Key Configurations***

Key configurations allow control on points. These configurations will be selected in player selection stage. For the sake of offline multiplayer, only keyboard keys are allowed for the key configuration selection. The players will have to choose two keys. One key for giving left movement, the other for right movement in the



game.

## ***3.Requirement Specification***

### ***3.1 Functional Requirements***

#### ***3.1.1 Play Game***

- In order to move the curve some certain functions must be added to the game to get inputs from the keyboard. With these inputs the players should be allowed to move their curve to the left or right direction.
- To detect collisions between two curves or with the borders, some functions must be added. We are planning to detect collisions with checking the color of the position right in front of the curves head. If the color of that position matches with another curve's color, that would mean a collision.
- In order to make the game more exciting and challenging power-ups must be added. Therefore, some functions should spawn power-up's randomly and at random locations.

### ***3.1.2 Pause Game***

- During the game, players might want to pause or exit the game. In order to add such a capability to the game, a pause button which directs the players to a pause menu must be added. This button should be a button from the keyboard, for example the escape or 'p' button.
- In the pause menu, the players should be capable of pausing or exiting the game with two different buttons. These buttons should be GUI elements.
- When a player hits the pause button, the game should stop and no further progress should happen. The curves must stop their movement and power-ups should stop spawning.

### ***3.1.3 View Help***

- To make the game user-friendly and useful a help menu must be added to the game in order to guide the players. This menu should be reachable from the main menu of the program via a GUI button.
- In the help system, there must be some guidelines about the GUI elements, how to play the game, and overall the program.

## ***3.2 Non-Functional Requirements***

### ***3.2.1 Performance***

Since, the game will have some dynamic movements, to make the players feel more flawless and satisfactory experience when playing the game, the performance of the game should be pleasant. In order to do so, there should be no issues in the code.

### ***3.2.2 Arcade & Last One Standing Themed***

In order to make the game more entertaining the GUI elements of the game and the designs of the curves must be matching. Since the players who cross the line or boundaries lose the game, the player that remains in the map wins the round and gets (# of players - 1 ) points.

### ***3.2.3 Offline Multiplayer***

Today most of the multiplayer games are online and thus, do not have the side-by-side challenge and entertainment. With our project game, it is desired to bring these pleasures back. To do so, this game is planned to be an offline multiplayer game which players can play the game using only one keyboard and one computer.

### ***3.2.4 User-Friendly Interface***

To make the game more understandable the GUI elements should be designed and positioned accurately to create an user-friendly interface. However this is not an essential process, it is very important to create the scene correctly in order to make the game enjoyable.

### ***3.2.5 Color Selection***

The players should be able to select the color of their curves in order to distinguish theirs from others. Therefore, a color selection option must be added to the game whenever a new player is added to the game.

## ***3.3. Constraints***

### ***3.3.1. English Language***

In terms of game development and marketing, developers should choose a language which is widely used. This way, accessibility of a game increases. For the sake of this purpose, we are constraining ourselves to use English language as our game's user interface.

### ***3.3.2. Java***

Java is our second constraint in this project. For the project developers and programmers, Java is a language which has big familiarity in the world of software. Java programs don't require to be installed and Java is famous for being portable. Also, encountering some errors such as buffering, overflow and segmentation fault in C/C++ are not included to Java. Because of these reasons, we are forcing ourselves to use Java programming language.



### ***3.3.3. Game Start***

There is no point for a player to play the game on your own. Therefore, the game can be played only if there are 2 or more players. A practice feature may be implemented if the time allows it.

## ***4. System Models***

### ***4.1. Scenarios***

#### ***4.1.1. Play Game Success Scenario***

In the “Main Menu, one of the players clicks on “Play Game”. It results with a “Player Selection” stage in which the players can add or remove players. Since in order to play a game at least two players are needed, the players need to add two players at minimum. Players go through name, color and key configuration stages. After each user adds his player to the game, they click on the “Play” button. Within a countdown by three, the first round starts.

#### ***4.1.2. Help Success Scenario***

User clicks on the “Help” button in “Main Menu” and reads the manual. After finishing the manual, user clicks on “Back” button to return to the “Main Menu”.

#### ***4.1.3. View Credits Scenario***

User clicks on the “Credits” button in “Main Menu”. System displays contact information of the developers of the game. User clicks on “Back” button to return to the “Main Menu”.

The diagram illustrates the use cases for the 'Curve Fever' system. A stick figure actor labeled 'Player' is connected to four use cases: 'Help', 'Settings', 'View Credit', and 'Player Selection'. The 'Player Selection' use case includes 'Player Name' and 'Player Color' extension points. The 'Player Name' use case includes 'Player Color' extension points. The 'Player Color' use case includes 'Key Configuration' extension points. The 'Key Configuration' use case includes 'Key Configuration' extension points. The 'Play Game' use case includes 'Pause' and 'End' extension points. The 'Pause' use case includes 'End' extension points.

```

graph TD
    Player((Player)) --- Help((Help))
    Player --- Settings((Settings))
    Player --- ViewCredit((View Credit))
    Player --- PlayerSelection((Player Selection))
    PlayerSelection -.->|<<Include>>| PlayerName((Player Name))
    PlayerName -.->|<<Include>>| PlayerColor((Player Color))
    PlayerColor -.->|<<Include>>| KeyConfiguration((Key Configuration))
    KeyConfiguration -.->|<<Include>>| KeyConfiguration2((Key Configuration))
    PlayerSelection -.->|<<Include>>| PlayGame((Play Game))
    PlayGame -.->|<<Include>>| Pause((Pause))
    PlayGame -.->|<<Include>>| End((End))
    Pause -.->|<<Include>>| End2((End))
  
```

<b>Use Case Name:</b>	Play Game
<b>Primary Actor:</b>	Player
<b>Entry Condition:</b>	Players are selected in appropriate way in terms of name, color and key configuration
	Player selects “Play Game” button from Main Menu.
<b>Exit Condition:</b>	Player selects “Exit” button from the main menu
	Player selects “Return to Main Menu” from Pause Menu.

**Flow of event:**

Player selects Player Name.

Player selects Color.

Player selects Key Configuration.

After identification, Player click “Play” button.

Each player try to keep going on playing field without touching to other player’s/players’ pattern.

The Player who is first eliminated gets “0” point. The other player/players win one more point than previous player in each time until one player wins the round. The player who gets the 20 points in total means the one wins the “Play”.

For each round, loser player/players wait for completion of round.

After completion the play, just winner gets one point and the other player/players get zero.

Scores are recorded to players’ total score list.

System, returns to Main Menu.

#### 4.2.2. View Help

**Use case name:** ViewHelp

**Participating actors:** Users

**Entry condition(s):** Users can select this option from the main menu

**Flow of events:** The gameplay guide will be appeared.  
Back button provides to close “Help” window and turn back main manu

**Exit condition(s):** Click to back button

**Postcondition(s):** To be in main menu

### ***4.2.3. View Credit***

<b>Use case name:</b>	View Credit
<b>Participating actors:</b>	Users
<b>Entry condition(s):</b>	Users can select this option from the main menu
<b>Flow of events:</b>	User selects the View Credit button. There are two options: <ol style="list-style-type: none"><li>1. Load: Load the suggestions and complaints</li><li>2. Back: User turns back to Main Menu without saving any suggestions and complaints.</li></ol>
<b>Exit condition(s):</b>	Click to back button
<b>Postcondition(s):</b>	To be in Main Menu

### ***4.2.4. Pause***

<b>Use case name:</b>	Pause
<b>Participating actors:</b>	Player
<b>Entry condition(s):</b>	Users click to Pause button while the play is continuing.
<b>Flow of event:</b>	When the play is continuing, player selects Pause button. There are 2 options: <ol style="list-style-type: none"><li>1.Continue: Continue to Play</li><li>2.Exit: Return the Main Menu</li></ol>
<b>Exit condition(s):</b>	User click to “Continue” button to continue to Play.
<b>Postcondition(s):</b>	Play is continuing.

## ***5. Glossary & References***

Curve Fever Wikia | Fandom powered by Wikia

[http://curvefever.wikia.com/wiki/Curve\\_Fever\\_Wikia](http://curvefever.wikia.com/wiki/Curve_Fever_Wikia)