Software Requirements Specification

for

Sticky Car - 2D Game

Version 1.0.9

Prepared by Ben Sopher, Sagi Weizmann

25/05/20

**Table of Contents**

Table of Contents......................................................................................................... 2-3

Revision History........................................................................................................... 3

1. Introduction............................................................................................................ 4-5

1.1 Purpose................................................................................................................ 4

1.2 Document Conventions......................................................................................... 4

1.3 Intended Audience and Reading Suggestions.......................................................... 4

1.4 Product Scope....................................................................................................... 4-5

1.5 References........................................................................................................... 6

2. Overall Description................................................................................................ 7-10

2.1 Product Perspective............................................................................................... 7

2.2 Product Functions................................................................................................. 9

2.3 User Classes and Characteristics............................................................................ 9

2.4 Operating Environment......................................................................................... 9

2.5 Design and Implementation Constraints................................................................. 9-10

2.6 User Documentation............................................................................................. 10

2.7 Assumptions and Dependencies............................................................................. 10

3. External Interface Requirements.........................................................................11-15

3.1 User Interfaces..................................................................................................... 11-14

3.2 Hardware Interfaces.............................................................................................. 15

3.3 Software Interfaces............................................................................................... 15

3.4 Communications Interfaces................................................................................... 15

4. System Features..................................................................................................... 16-26

4.1 Game registration & login system............................................................................... 16

4.2 Game menus.................................................................................................... 17-18

4.3 In-game store....................................................................................................19-20

4.4 High score table ...........................................................................................21-22

4.5 coins feature ......................................................................................................23

4.6 levels feature ....................................................................................................24-25

4.7 in-game sound effects ........................................................................................26-27

5. Other Nonfunctional Requirements..................................................................... 28

5.1 Performance Requirements.................................................................................... 28

5.2 Safety Requirements............................................................................................. 28

5.3 Security Requirements.......................................................................................... 28

5.4 Software Quality Attributes................................................................................... 28

5.5 Business Rules..................................................................................................... 28

6. Other Requirements.............................................................................................. 28

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Sagi Weizmann Ben Sopher | 24/11/19 | Initial Document | 1.0.0 |
| Sagi Weizmann | 25/11/19 | Added Purpose, Scope, References, and some abbreviations. | 1.0.1 |
| Ben Sopher, Sagi Weizmann | 01/12/19 | An added interface from in-game and website description | 1.0.2 |
| Ben Sopher, Sagi Weizmann | 04/12/19 | Added all the features of the game and detailed information | 1.0.3 |
| Ben Sopher, Sagi Weizmann | 05/12/19 | Revised for the final version added some non-functional requirements. | 1.0.4 |
| Sagi Weizmann | 06/12/19 | Revised again, checked grammar mistakes and updated features | 1.0.5 |
| Ben Sopher | 07/12/19 | Fixed English mistakes, added current features from the GitHub project | 1.0.6 |
| Sagi Weizmann | 08/12/19 | Added more detailed features. | 1.0.7 |
| Ben Sopher, Sagi Weizmann | 09/12/19 | Fixed grammar. | 1.0.8 |
| Ben Sopher , Sagi Weizmann | 25/05/20 | Achievement system got removed and new game feature added: in-game sound effects . | 1.0.9 |

# **1.** **Introduction**

## **1.1** **Purpose**

This document is a software requirement specification for the Sticky Car game project which is a 2D Arcade game. We will give a complete description and overview of the project. we will also list the requirements which meet the needs of the users.

This document aims to explain the requirements of our game which consists of designing a 2D game that will be built on the Unity game engine.

## **1.2** **Document Conventions**

The entire doc is written with an Arial font in size 11.

Some sentences are written in red to highlight important notes.

## **1.3** **Intended Audience and Reading Suggestions**

The intended audience of this document is the members of the project group

and developers who are going to implement the application explained in this

document. The document will guide the developers through the implementation

phase. This document is arranged according to the IEEE format.

## **1.4** **Product Scope**

The project name is Sticky Car. The name “Sticky Car” was chosen because the car staying “sticks” to the ground.

You need to navigate your way through a variety of difficult hilly trial courses to reach the finish line making sure you don’t run out of fuel in the process. The game contains few different levels and the levels contain different maps, also the map itself contains coins that the player can collect them and it will be added to their current balance in-game.

The players will use our in-game store and spend cash points(an in-game currency) in-game, We also wish to provide a quality gaming experience that includes a progression system.

## **1.5** **References, Definitions, Acronyms and Abbreviations**

[Unity (game engine) - Wikipedia](https://en.wikipedia.org/wiki/Unity_(game_engine))

[Brackeys - Balance Racing Game](https://github.com/Brackeys/Balance-Racing-Game)

SRS: Software Requirement Specification

Unity: Unity is a game development ecosystem: a powerful rendering engine fully integrated with a complete set of intuitive tools and rapid workflows to create interactive 3D/2D content.

**2.** **Overall Description**

## **2.1** **Product Perspective**

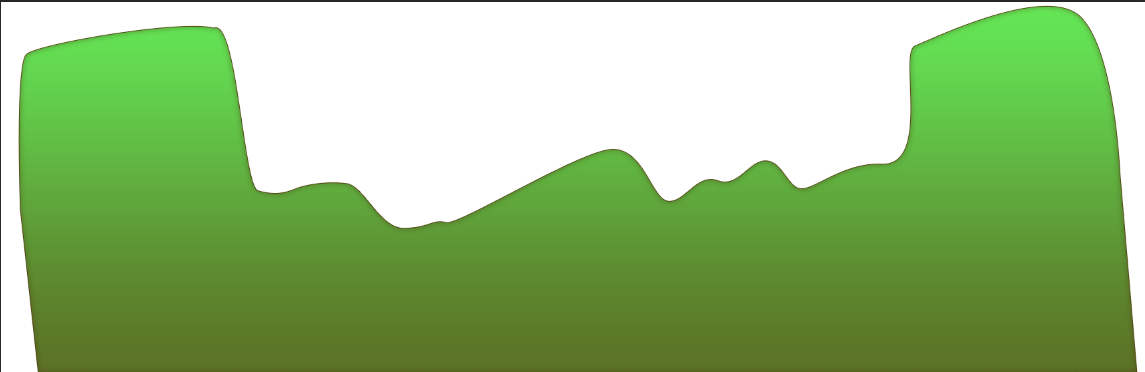
Sticky Car is a totally independent system that is based on a self-made game from GitHub called: “Balance Racing Game” by Brackeys and it’s actually an extension to that game. This program has only one type of user (the gamers themselves).

Brackeyshas already implemented a few features such as:

* Car with two wheels(moving with arrows)



* Dynamic 2D System
* Physics between the map to the car
* Basic Map



* Dynamic Moving Sky (\*subject to change)



* Custom Cartes (Obstacles)



* End Game Flag



## **2.2** **Product Functions\*\*\* updated list**

Sticky Car main function will include:

* Create a user account.
* Login system.
* Game Interface(Main Menu, Shop, Settings, etc…)
* ~~Player Achievements system.~~ //not in final project
* Player Customization section(Changing Skins).
* High Score Table.
* Coins System.
* Different levels(including completely new terrain)
* Mobile Support.
* Game settings(sound effect toggle).
* In game sound effect(including background music)//new features.

## **2.3** **User Classes and Characteristics**

Mobile Gamers is the users we anticipated to use our game product:

1. New player completely without any experience in arcade 2D Gaming.

* A casual player that plays around 45min - 1 hour weekly.
* No security privilege.
* Educational levels may be varied.

These players are the main audience that is likely to use our game product.

## **2.4** **Operating Environment**

Software Requirement:

Our game product “Sticky Car” is based on Unity Game Engine and this allows us to run on multiple platforms(Windows, macOS, Android, Etc.…). But we choose to focus on developing for Mobile (Android Operating System), there are no additional libraries dependencies that needed to run the game.

The game will Run on the Android Operating system from Google.

Hardware Requirement:

Generally, content developed with Unity can run pretty much everywhere.

## **2.5** **Design and Implementation Constraints**

Specific Technologies:

* Unity 3D Game Engine (Unity 2019.2.17f1 (64-bit))
* MySQL (For Achievements System).
* Xampp (For MySql)

Tools:

* Adobe Photoshop CC 2019

Database:

* MySQL Database - Includes the user’s tables etc..

Language Requirements:

* Visual Studio 2019 + C# (already integrated with unity)
* Php 7 - (For the backend)

## **2.6** **User Documentation**

Not Relevant.

## **2.7** **Assumptions and Dependencies**

* The Unity game engine well exceeds the personal user’s cap and will require a commercial license.
* Adobe photoshop license is required.
* Visual studio 2019 may require a paid edition for certain functionality.
* License OS such as Windows 7 or above macOS 10.11 or above or Linux OS.
* A valid internet connection and GitHub account.

# **3.** **External Interface Requirements**

## **3.1** **User Interfaces**

## **3.1.1 Sticky Car - Main Interface (Visual )**[Not exists in current project prototype\*]

\*The UI is subject to change

****

Figure 1. A prototype of the main interface

The main interface will mainly include the play button, achievements button, shop in-game, information medals, and settings.

## **3.1.2 Levels Screen - Choosing Levels**

## [Not exists in current project prototype\*]

\*The UI is subject to change



Will contains the current level and locked levels,

Home button - for returning to the main menu.

Shop Button - for shop in-game.

Settings -Changing settings such as volume and sound effects.

Play button - entering the game.

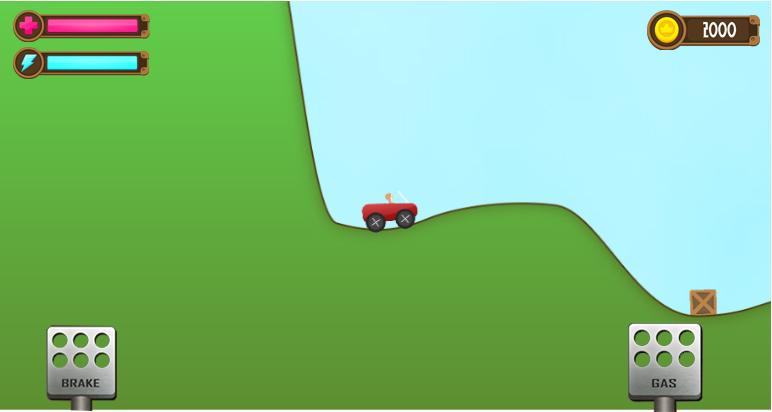
## **3.1.3 GamePlay (in-game)** [current and future features]

Current Game:

****

Figure 1. Playing the game.

Future Features that will be added:



**3.1.4 In-game Menu (Pause Menu)** [Not Exists]



The in-game menu will include the following buttons:

1. Resume - this button will resume the current paused game.

2. Restart - this button will restart the current level.

3. Options - Settings of the game.

4. Exit - Exit to the main menu.

## 

## **3.2** **Hardware Interfaces**

Not relevant.

## **3.3** **Software Interfaces**

Specific Software Components:

Integration between the database, website and the game itself:

We store in the database the user information (password, username, progression in-game), the user has been registered in game and the information will be stored in the database.

C# and Visual Studio 2019

The application will run on version 8.0 of the C# language. This is the most recent

version of the C# programming language. The C# programming language is used for

all the backend programming for the application. Visual Studio 2019 is used to

compile all the code that is developed in C#.

Unity

The application will use the Unity game engine for the user interface of the

Sticky Car game. None of the components of the professional version should be needed for the application. Unity version 2019.2.17f1 is used for the development of the project.

Android SDK

The Android SDK will be needed to compile the game itself , as we dedicate it to Android supported phones.

MySQL

We use MySQL as our database in our game product, MySQL is an open-source relational database management system(RDBMS).

Apache & PHP

We will use the web server, the web server will the backend of the game. will be responsible for communication between the user and the game database.

The Apache HTTP Server is free and open-source cross-platform web server software.

PHP is a general-purpose programming language originally designed for web development.

## **3.4** **Communications Interfaces**

Electronic Forms:

Registration page that includes:

Username, Password and Email.

Login page:

Username , Password.

# **4.** **System Features**

[These features are not contained in the Balance Racing Game by Brackeys !]

## **4.1** **Game registration & login system**

In order to play the game and track progression, the user must register via the game itself and then login with his credentials in-game.

**4.1.1** Description and Priority

The game registration & login system is a critical part of our game hence it’s considered a high priority. The registration process will require the user to choose a username + password and fill his email address.

In login, the user will need to fill username + password in-game to enter the game.

The benefit that we get for this feature is track progression,achievements,current mission,coins and other necessary information for developers and allow user account security.

**4.1.2** Stimulus/Response Sequences

A sequence of actions:

* Player enter game
* The player clicks the register button
* The player is directed to fill his information
* The player will be asked to enter his desired user name and password and his email
* After the player will fill all the details necessary he will hit the register button
* The player will be able to resume the game and enter with his credentials

**4.1.3** Functional Requirements

1. The game server must be online.

* If the game server is not online than the user won’t be able to log in and will get a proper error message.

1. The game database must online.

* If the game database isn't working for some reason the shop will become unavailable and the user will get a proper error message.

1. The game database must be up to date.

* Or the game will not respond to user login information.

**4.2** **Game menus**

The game will require menus in order to present its features and game modes to the players in a reasonable manner.

**4.2.1** Description and Priority

The game menus have high priority in our game because in the current state there is no real menu. The game will contain the main menu and a few other windows branching out of it.

The main menu will contain things like the Play button, Store button, etc...

The benefit that we get for this feature is the accessibility and ease of use for players.

**4.2.2** Stimulus/Response Sequences

A sequence of actions:

* Player enter game
* Player fills his username + password
* Player is directed to the game main menu

**4.2.3** Functional Requirements

1. The game server must be online.

* If the game server is not online than the user won’t be able to log in and will get a proper error message.

2. The game database must online.

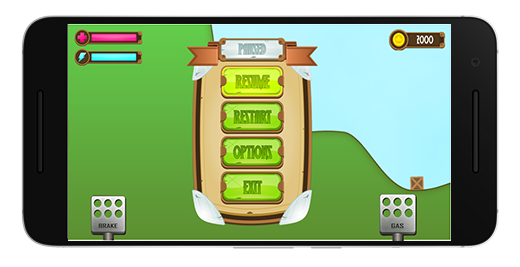
* If the game database isn't working for some reason the shop will become unavailable and the user will get a proper error message.

**4.2.4 Prototypes of the 2 menus**

Main Menu:



In-game menu:



## 

## **4.3** **In-game store**

The game will contain an in-game store that will enable the players to purchase cosmetic items.

**4.3.1** Description and Priority

The game store will contain in-game cosmetic items such as character skins or car skins.

The players will be able to buy things from the store using in-game currency.

This feature is considered a high priority feature for us to implement in-game.

**4.3.2** Stimulus/Response Sequences

A sequence of actions:

* Player enter game
* The player fills his username + password
* The player is directed to the game main menu
* Player hits the store button
* The player is directed to store window
* The player will select a specific item
* The player will be asked if sure he wants to purchase the item
* The item will appear in player inventory and the right amount of currency will be subtracted from his balance

**4.3.3** Functional Requirements

1. The game server must be online.

* If the game server is not online than the user won’t be able to log in and will get a proper error message.

2. The game database must online.

* If the game database isn't working for some reason the shop will become unavailable and the user will get a proper error message.

1. The game database must be up to date.

* If the database won’t be up to date with the user's most recent in-game currency balance and shopping request the store might be exploited.

**4.3.4** Prototype & Details about the store

There will be 5 skins of cars to choose from, and it will be shown on the game store as well:

Skin Store

The game will contain also a skin store. The store will contain several skins to choose from, and the player will have to buy some cash points to buy from the skin stores.

We have future plans to add more skins to the skin store.



Figure 2. Store Prototype

The Current Game Contains:

* **Default Player:** a stick man with a regular car



## **4.4** **High Score Table**

The game will contain a high score table feature that will allow the players to compete with other players.

**4.4.1** Description and Priority

The game will contain a feature that will allow players to “compete” with each other and beat other players’ scores.

This feature is considered a low priority feature for us to implement in-game.

**4.4.2** Stimulus/Response Sequences

A sequence of actions:

* Player enter game
* The player fills his username + password
* The player is directed to the game main menu
* Player hits the Play button
* The player is directed the game
* Player is earning points
* Player finish a certain level
* Points of the player recorded automatically into the table
* Other players earn points
* The players with the most points at summary shown on the high score table.

**4.4.3** Functional Requirements

1. The game server must be online.

* If the game server is not online than the user won’t be able to log in and will get a proper error message.

1. The game database must online.

* If the game database isn't working for some reason the shop will become unavailable and the user will get a proper error message.

**4.4.4** Prototype and detailed image



**4.5** **Coins Feature**

The in-game will contain earnable coins while you play.

**4.5.1** Description and Priority

The game will allow players to earn coins during the gameplay.

This feature is considered a high priority feature for us to implement in-game.

The benefit that we get for this feature is mainly versatility and increase the playability options for players.

**4.5.2** Stimulus/Response Sequences

A sequence of actions- not relevant

**4.5.3** Functional Requirements

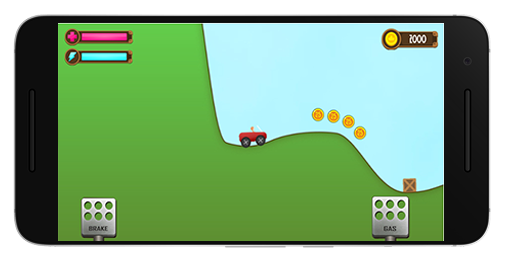
1. The game server must be online.

* If the game server is not online than the user won’t be able to log in and will get a proper error message.

1. The game database must online.

* If the game database isn't working for some reason the game won't be able to track the player progression.

**4.5.4** Prototype & Details about the mode



Earnable coins will appear during the gameplay , the player will be able to collect them.

**4.6** **Levels Feature**

The in-game will contains a variety of levels with different difficulties .

**4.6.1** Description and Priority

Levels is an important measure to determine your in-game progress, that is why we want to implement it to our game.

This feature is considered a medium priority feature for us to implement in-game.

The benefit that we get for this feature is mainly versatility and increase the playability options for players.

**4.6.2** Stimulus/Response Sequences

A sequence of actions:

* Player enter game
* The player fills his username + password
* The player is directed to the game main menu
* Player hits the Play button
* The player is directed to a windows with different levels
* The Player click on certain level
* The Player enter the game with the selected level.

**4.6.3** Functional Requirements

1. The game server must be online.

* If the game server is not online than the user won’t be able to log in and will get a proper error message.

1. The game database must online.

* If the game database isn't working for some reason the game won't be able to track the player progression.

**4.6.4** Prototype & Details about the mode



Figure 1. Different Levels at the Level Select screen.

As you can see in the picture above , the player can select different levels , each level indicates different difficulty.

**4.7** **In-game sound effects\*\*\* new feature added**

The game will feature sound effects including: background music and coins pickup sound and car engine sound .

**4.7.1** Description and Priority

We chose to add sound effect to the game to increase game immersion.

This feature is considered a low priority feature for us to implement in-game.

The benefit that we get for this feature is mainly quality of life and improved gameplay experience.

**4.7.2** Stimulus/Response Sequences

A sequence of actions:

* Player enter game
* The player fills his username + password
* The player is directed to the game main menu
* The game music automatically start
  + 1. Functional Requirements

The player device must have some speaker to represent the sound from the game.

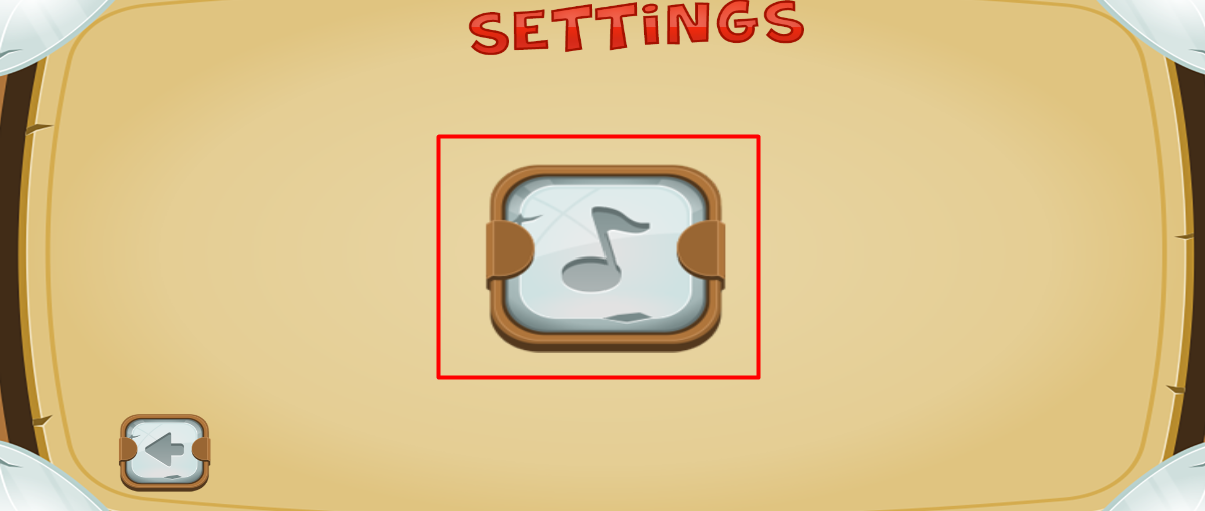
**4.7.4** Prototype & Details about the mode

To enable or disable in-game sound effects you need to go to setting in main menu



After pressing menu button you are presented with sound effects toggle which can enable/ disable that option.

\*By default the sound effects are turned on.



# **5.** **Other Nonfunctional Requirements**

## **5.1** **Performance Requirements**

Not relevant

## **5.2** **Safety Requirements**

Some players might get psychologically addicted to the game.

## **5.3** **Security Requirements**

In order to play the game players must register online and fill their credential and these details are stored on a secure database.

## **5.4** **Software Quality Attributes**

The game must be optimized to run smoothly on lower-tier mobile phone hardware, robustness.

The game UI must be arranged in a reasonable manner- usability.

## **5.5** **Business Rules**

Not relevant

# **6.** **Other Requirements**

Not relevant