

SE 3XA3: Software Requirements  
Specification  
DinoDodger

Team 39, S.R.A Squad  
Shrey Mittal, mittas1  
Razan Abujarad, abujarar  
Zhiwen Yang, yangz18

6 October, 2017

# Contents

<b>1</b>	<b>Project Drivers</b>	<b>1</b>
1.1	The Purpose of the Project . . . . .	1
1.2	The Stakeholders . . . . .	1
1.2.1	The Client . . . . .	1
1.2.2	The Customers . . . . .	1
1.2.3	Other Stakeholders . . . . .	2
1.3	Mandated Constraints . . . . .	2
1.4	Naming Conventions and Terminology . . . . .	2
1.5	Relevant Facts and Assumptions . . . . .	3
<b>2</b>	<b>Functional Requirements</b>	<b>4</b>
2.1	The Scope of the Work and the Product . . . . .	4
2.1.1	The Context of the Work . . . . .	4
2.1.2	Work Partitioning . . . . .	4
2.1.3	Individual Product Use Cases . . . . .	5
2.2	Functional Requirements . . . . .	5
<b>3</b>	<b>Non-functional Requirements</b>	<b>7</b>
3.1	Look and Feel Requirements . . . . .	7
3.2	Usability and Humanity Requirements . . . . .	7
3.3	Performance Requirements . . . . .	8
3.4	Operational and Environmental Requirements . . . . .	9
3.5	Maintainability and Support Requirements . . . . .	9
3.6	Security Requirements . . . . .	9
3.7	Cultural Requirements . . . . .	10
3.8	Legal Requirements . . . . .	10
3.9	Health and Safety Requirements . . . . .	10
<b>4</b>	<b>Project Issues</b>	<b>10</b>
4.1	Open Issues . . . . .	10
4.2	Off-the-Shelf Solutions . . . . .	11
4.3	New Problems . . . . .	11
4.4	Tasks . . . . .	11
4.5	Migration to the New Product . . . . .	11
4.6	Risks . . . . .	11
4.7	Costs . . . . .	11

4.8	User Documentation and Training . . . . .	11
4.9	Waiting Room . . . . .	12
4.10	Ideas for Solutions . . . . .	12

## List of Tables

1	Revision History . . . . .	ii
---	----------------------------	----

## List of Figures

1	Use Cases Diagram . . . . .	5
---	-----------------------------	---

Table 1: **Revision History**

Date	Version	Notes
2017/10/6	1.0	created SRS
2017/11/9	1.1	updated Functional Requirements, enumerated non-Functional requirements and Mandated Constraints
2017/12/6	1.2	Final Revision (rev1) of SRS. New edits in red ink; discarded text is struck out.

# 1 Project Drivers

## 1.1 The Purpose of the Project

The purpose of DinoDodger is for the entertainment of the user. As a redevelopment of Chromium's T-Rex Runner, the game involves a virtual character that is continuously running in one direction within a particular landscape. It encounters obstacles in its path that must be overcome by jumping over them, which the character, under the control of the user, can do by pressing a key ~~the Spacebar~~ on the keyboard. ~~Each step the character takes unharmed by the obstacles increments the number of points accrued, with the intention of the game to score as high as possible as the game progressively becomes more difficult with the speed of the character and the number of obstacles being encountered. The game only terminates once the character comes into contact with an obstacle. For this project, a variety of characters and landscapes each with their intrinsic obstacles and difficulty levels will be provided to the user to choose.~~ The objective of this game is to accrue as many points as possible, which, initially at zero increment continuously at a particular time interval as long as the character has not intercepted an obstacle in its path. The game progressively gets difficult with the speed of the obstacles coming towards the character marginally increasing at a certain time interval. For the game's implementation of this project, there will be a simple UI provided that would allow a user choose a combination of character and landscape containing the obstacles. The game window will be larger than before, taking up at least 50% of the screen size.

## 1.2 The Stakeholders

### 1.2.1 The Client

The client of this project is Dr. Asghar Bokhari, Professor of SFWRENG 3XA3 in McMaster University

### 1.2.2 The Customers

This project is intended for a single user of an age group 5 and older.

### 1.2.3 Other Stakeholders

Other stakeholders for this project include the developers of the game, and any other persons who take interest in its future development.

## 1.3 Mandated Constraints

1. There must be only a single user interacting with the User Interface of this project at one time
2. The User Interface must take up at least 50 percent of the screen of the device.
3. There must be at least three characters and three landscapes available for the user to choose.
4. Project must be completed by December 2017.

## 1.4 Naming Conventions and Terminology

**User Interface:** The User Interface is a window that will display the game, acting as the visual medium of communication between the program and the user, taking keyboard and cursor inputs wherever necessary and delivering correct output.

**Character:** A character is the virtual player controlled by the user taking the form of various kinds of dinosaurs.

**Landscape:** The landscape is the virtual environment in which the character runs. Examples include desert, tundra, forest etc.

**Obstacle:** An obstacle is any stationary or mobile object within the landscape that inhibits the characters path or may come into contact with it. These objects are to be overcome.

**Gameplay:** The stage that displays the game with the character and landscape selected and animated, with Points and Highscore being shown. This stage takes user input to get the character to jump.

Points: A numerical representation of the points accrued during gameplay.

Highscore: A numerical representation of the maximum number of points accrued with the current character and landscape settings (i.e.) the user has only chosen to 'Play again'

Game Over: The state following Gameplay upon collision of the Character with an incoming obstacle.

Play Again: Option given in 'Game Over' state allowing user to re-enter Gameplay scene with current settings.

Main Menu: Option given in 'Game Over' state allowing user to return to the main menu of the UI to choose a different combination of character and landscape.

Cactus: An obstacle on the ground

Pteradactyl: An obstacle flying towards the Character at two different heights.

## 1.5 Relevant Facts and Assumptions

~~User has already played an equivalent of this game hence there is no necessity to include instructions~~

- In case the user has not played an equivalent of this game, simple instructions are provided on what to do. This includes information on which key to press to get the character to jump and instructions to choose a character and landscape.
- Assuming game is easy to learn, instructions on where and why to use the jump key are not provided.

## 2 Functional Requirements

### 2.1 The Scope of the Work and the Product

There is minimal scope for this project since there is only a single user intended to use the product at one time.

#### 2.1.1 The Context of the Work

The game motive is to score as high as possible by overcoming obstacles. There is minimal context apart from this.

#### 2.1.2 Work Partitioning

- Razan: User Interface
- Shrey: Main Module that interacts with User Interface to produce output
- Allen: Creation and animation of characters and background

Razan

- UI Module

Shrey

- Cactus Module
- Pteradactyl Module
- SpriteAnimation Module
- PointCounter Module
- Character Module
- part of PlayScene Module

Zhiwen

- PlayScene Module

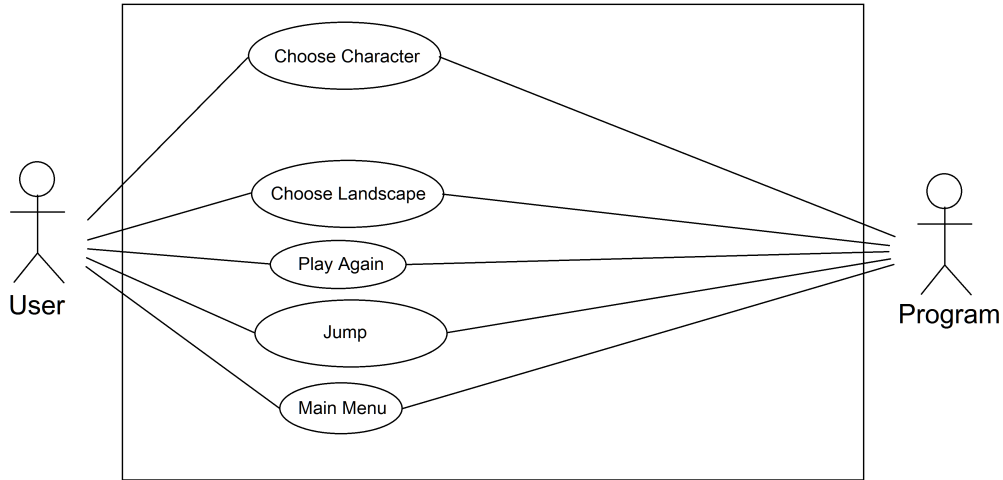


Figure 1: Use Cases Diagram

### 2.1.3 Individual Product Use Cases

## 2.2 Functional Requirements

1. Program must prompt user to select character
2. Program must prompt user to select landscape
3. Program must begin game when user provides key input or uses cursor to click initiatory button. presses the 'Play' button. Program will transition to the Gameplay stage.
4. The user interface must display the animation of a running character from a characters perspective, meaning the landscape must be animated to be moving relative to the character.
5. Every time the user presses the jump Space key, the program responds by showing the character jump from the designated ground level, such that the jump is high enough to overcome the maximum height of obstacles positioned on the ground level, and then land back on the ground. This process should take less than one second to complete responds by transitioning the Character from its initial y-position along



the y-axis to a certain y-position such that cactus obstacles can be overcome and back down to its initial y-position.

6. ~~Game shall terminate when the character comes into contact with an obstacle, meaning there is intersection between the pixels of the character and obstacle.~~ Upon intersection between character and an obstacle, there is a transition from 'Gameplay' to the 'Game Over' state
7. Game shall prompt user to play again or exit, restarting the program or terminating the program based on selection.
8. During gameplay, program will display the current score and overall highscore on the top right corner of the window.
9. When the Space key is pressed and held, the jump transition animation occurs repetitively until the user releases the key.
10. The program will randomly pick a cactus from a set of cactus images to release them.
11. The program will randomly choose intervals to release Pteradactyls at two different heights
12. The program will randomly choose intervals to release cacti
13. The program will increase the speed of the cacti and the pteradactyls at certain time intervals
14. The highscore is initially 0, and updated only if the value of points exceeds. This happens within the chosen character and landscape settings only
15. Points is always reset to 0 whenever Gameplay stage is entered.
16. In the Game Over stage, the program displays the updated highscore.
17. In the Game Over stage, two options are shown, to 'Play Again', in which current character and landscape choices are maintained and the highscore preserved, and a 'Main Menu' option which brings user back to main menu to choose characters.

18. When program is initially compiled and run, the default character and landscape choices are both grey.
19. Highscore is reset to zero if the user decides to select 'main menu' and select a new character and landscape combination.
20. If user chooses to go back to Main Menu, default character and landscape option does not apply. Same choices are maintained if 'Play' is pressed.
21. The points accrued are updated (incremented) every 200 milliseconds.
22. Pteradactyl must be shown animated.

### 3 Non-functional Requirements

The following sections describe the non-functional requirements of DinoDodger.

#### 3.1 Look and Feel Requirements

1. The product shall be comfortable for the user to look at i.e no harsh colour schemes shall be used in the design of the game. The characters and landscapes will have the following color options: grey(original), red, blue.
2. ~~The product shall fit into the screen of the device without the user having to readjust any settings of the product.~~ The UI of the game will use at least 50% of the screen of the PC. This screen will not be resizable.
3. Elements of the game shall look cohesive i.e the elements of the game (characters, obstacles) shall have appropriate sizes such that the final product looks well-integrated.

#### 3.2 Usability and Humanity Requirements

1. The product shall be easy to use by users of a broad age group mostly ages 5+.
2. ~~The product shall be understandable by users with all language backgrounds~~

3. The product shall be used by people without prior training.
4. The product shall be memorable to users i.e users can remember how to use the product at all times.
5. ~~The product shall retain the highest score obtained by the user.~~
6. ~~The product shall provide the user a feedback of their current score as well as their highest score at the end of each round~~
7. Users with hearing disabilities shall be able to use the product with ease.

### **3.3 Performance Requirements**

1. ~~The product's interface shall have a maximum startup time of 2 seconds after an internet connection becomes unavailable.~~
2. ~~The character element of the product shall response to a screen tap within 0.5 seconds i.e character must jump within 0.5 seconds of user tapping the screen of the phone.~~
3. The character element of the product must jump in a time interval equal to the time interval taken to land.
4. The product shall display the score of the user accurately within 1 point.
5. ~~The product shall be available 24 hours per day, 365 days per year given an internet connection is absent.~~
6. The product shall operate for a period of at least 3 months with minimal maintenance.
7. The product shall be able to operate continuously for at least 1 hour without crashing or producing an error message to the user.

### 3.4 Operational and Environmental Requirements

1. ~~The product shall be used at any location that lacks an internet connection.~~
2. The product shall be used in any weather conditions given the device used to use the product is safe from damage.
3. The product shall be used with any level of light or sound that the device used permits i.e maximum and/or minimum brightness and volume level respectively.
4. ~~The product must be of a size that is compatible with the device used.~~  
Product will be able to run on Windows and Mac OS systems. This program is not designed for mobiles.
5. ~~The product must be installed with the supporting browser and/or application used for browsing the internet.~~ This program should be easily downloaded and compiled using an executable .jar file or makefile
6. ~~The product must be available for download by the user on any web browser.~~

### 3.5 Maintainability and Support Requirements

1. Additional test cases shall be made for any new features added.
2. The product shall run on any personal computers supporting Java.

### 3.6 Security Requirements

1. Feedback of the user's score shall be displayed only on the users device.
2. The user's score shall not be distributed without the consent of the user.
3. The product shall not ask the user for personal information, including name, email, date of birth or banking information.

### **3.7 Cultural Requirements**

1. The product shall not contain any inappropriate symbols.
2. The product shall not contain any material offensive to religious or ethnic groups.

### **3.8 Legal Requirements**

1. The product shall be ethical and comply with all cyber security laws.

### **3.9 Health and Safety Requirements**

1. ~~The product shall produce a warning to the user who uses the product continuously for more than 90 minutes.~~
2. ~~The warning shall inform the user of risks of cramped fingers, eyesight problems and the risk of driving while using the device.~~

## **4 Project Issues**

### **4.1 Open Issues**

1. The characters jump action performed by pressing spacebar or arrow key
2. How complex the animation of the characters will be (simple pixelated or smooth animation)
3. What characters and landscapes to create
4. Whether this game will include different elevation levels for the character to run on
5. If extra features like temporary power-ups for the character will be available
6. Will audio be included each time an obstacle is overcome or point milestone met
7. How points will be calculated, (using time passed or steps taken by the character)

## **4.2 Off-the-Shelf Solutions**

- Whether the redevelopment of this project will incorporate ideas from other games like Mario Bros such as different levels and power-ups for the characters
- How to evaluate points accrued (number of steps taken, obstacles overcome, time survived)

## **4.3 New Problems**

- How to enhance difficulty of the game - What new features to add to the game to improve existing game

## **4.4 Tasks**

See use cases diagram in Figure 1.

## **4.5 Migration to the New Product**

In order for the intended product-to-be to work, required files must be downloaded and executed. No other steps must be taken.

## **4.6 Risks**

- Game being hacked and source code being changed.

## **4.7 Costs**

In order for the intended product-to-be to work, required files must be downloaded and executed. No other steps must be taken.

## **4.8 User Documentation and Training**

No user licence and prior training is required for this project since it is assumed the project is simple enough for the targeted age group or the user

has already been exposed to a game of equivalent difficulty.

## **4.9 Waiting Room**

As of now all requirements functional and non-functional are to be met by the deadline.

## **4.10 Ideas for Solutions**

- Designing different characters using MS Paint or obtaining existing images from the Internet
- Obtaining designs for landscape and obstacles found within from existing sources or creating new designs using MS Paint or equivalent software