Software Requirements Specification

Super Tetris

Group#: 38

Team Name: Binary

Members:

Tongfei Wang : wangt62

Bowen Yuan : yuanb1

Tim Zhang : zhangj14

December 6, 2017

SFWR ENG 3XA3

McMaster University

Contents

List of Tables

List of Figures

Table 1: Revision History

Date	Version	Notes
Oct 6 2017	1.0	Project Drivers Uploaded
Oct 6 2017	1.1	Functional Requirements Uploaded
Oct 6 2017	1.2	Non-Functional Requirements Uploaded
Oct 6 2017	1.3	Project Issues Uploaded
Dec 6 2017	2.1	Rev_1 Updated
Dec 6 2017	2.2	Rev_1 Updated

1 Project Drivers

1.1 The Purpose of the Project

Tetris is a tile-matching puzzle game. "Tetrominoes" are differently shaped blocks and they drop from the top of the screen one by one. The player can move or rotate the Tetriminos as they fall. The goal of this game is to create a horizontal line without gaps. Once the line is created, it will disappear and be replaced by the blocks above it.The purpose of the game is to bring the joy back to the players who love it.
Game occupies a great part of many people's leisure time, and lately, it becomes much more so with a rising use of gaming devices. However, as there are numerous games to choose, Tetris is no longer a good choice for people. Once upon a time, this classical game earned its fans with the unique play style. Currently, with the burst of technology, many new games come to the world with their fascinating theme and attractive scene, which pushes Tetris out of the age. As a result, it is essential to modify the game to make it more competitive.

1.2 The Stakeholders

1.2.1 The Client

Dr. Asghar A Bokhari is the client. He is the professor of Software Engineering Department at McMaster University. The project is developed under his requirement.

1.2.2 The Customers

Anyone who has a laptop or a computer connected to the Internet are the customers. Although the game is developed for the people who interested in Tetris, anyone can get access to the game.

1.2.3 Other Stakeholders

There will be some testers to test the game if it is playable or even fun. As the project will be uploaded to the Gitlab as open source, any developer who is interested in modifying it would be welcome. Shucai Yao, the teaching assistant of Dr. Asghar A Bokhari, will help the team to manage the project and develop the program.

1.3 Mandated Constraints

1.3.1 Solution Constrains

The game shall playable on the Internet.
The system shall meet every functional and non-functional requirements.

1.3.2 Implementation Environment of the Current System

The game is playable if the devices connect to the Internet. The system shall work on website. It shall be opened on every major browser such as IE, Chrome, etc. The system shall be implemented in Javascript and HTML.

1.3.3 Schedule Constraints

The project must be finished before December 6th.

1.4 Naming Conventions and Terminology

1.5 Relevant Facts and Assumptions

We assume that the players have the basic awareness of the original Tetris. It is also essential that the Internet connection will stay stable and the device will not be turned off by accident. The device for playing the game should either be a laptop or a computer which has a keyboard and a mouse. We assume that the players have the basic skills about using a computer or laptop.

2 Functional Requirements

2.1 The Scope of the Work and the Product

2.1.1 The Context of the Work

The main task is to create a game called ”SuperTetris” based on the classic Tetris game that everyone can play this game on the Internet. We

Table 2: List of Terminology

Term	Definition
Tetrominoes	Geometric shape composed of four squares
Item	Four different items to help player get higher scores
Line Clear	The only way to obtain score
Score	Measure of Player's performance
up	arrow key in the keyboard
left	arrow key in the keyboard
right	arrow key in the keyboard
down	arrow key in the keyboard
enter	enter key in the keyboard
r	character key in the keyboard
1,2,3,4	number key in the keyboard
experience	Measure of Player's performance
money	economy system for the items system.

need a ~~elastic cloud~~ server on which we can put all our project files to host the game. And this will also require a domain linked to our public IP address to make everyone can have access to this game. Last but not least, we plan to make this game downloadable so users can play this game without the Internet.

2.1.2 Work Partitioning

~~The development of this game should be shared evenly between three team members. Tim is responsible for designing the game logic. The specific coding work is mainly going to be done by Micho and Bowen. And Bowen is also responsible for dealing with the server's issues.~~
Work Partitioning Event List:
1. User inputs Website address on a browser.
2. User starts the games.
3. User moves the Tetriminos.
4. User rotates the Tetriminos.
5. User uses an item
6. A horizontal line is formed
7. Game ends

2.1.3 List of Events

Table 3: List of Event

Label & Event Name	Inputs/Outputs	Summary
E1 :New Game	User runs the game(in)	A new game was started
E2 :Pause	Press ”enter”(in)	Game paused
E3:Resume	Press ”enter”(in)	Game resumed
E4:Restart	Press ”r”(in)	Game restarted
E5:Rotate tetromino	Press ”up”(in) game state(out)	The current tetromino was rotated clockwise by 90 degrees
E6:Left move	Press ”left”(in) game state(out)	Move the current tetromino left by one grid
E7:Right move	Press ”right”(in) game state(out)	Move the current tetromino right by one grid
E8:Let it fall	Press ”down”(in) game state(out)	Let the current tetromino fall to the bottom
E9:Use item(bomb)	Press ”1”(in) game state(out)	Buy an item(bomb) and use it/Transaction fails and a reminder appears
E10:Use item(slow)	Press ”2”(in) game state(out)	Buy an item(slow) and use it/Transaction fails and a reminder appears
E11:Use item(skip)	Press ”3”(in) game state(out)	Buy an item(skip) and use it/Transaction fails and a reminder appears
E12:Use item(clear)	Press ”4”(in) game state(out)	Buy an item(clear) and use it/Transaction fails and a reminder appears
E13:Clear up row(s)	game state(out)	One row or multiple rows are cleared up and everything above shall fall Experience and score shall increase
E14:Bomb falls	game state(out)	Clear up an area of 5*5 around the bomb/Experience and score shall increase
E15:Game over	Game Over(out)	The game is over and no more tetromino falling

2.1.4 Use case

Use Case #: 1

Name:	New Game
Trigger:	As the user enters our website
Precondition:	None
Postcondition:	A new game starts

Use Case #: 2

Name:	Pause the game
Trigger:	As the user press ”enter”
Precondition:	The game is in progress
Postcondition:	Game pauses

Use Case #: 3

Name:	Resume the game
Trigger:	As the user press ”enter”
Precondition:	The game is paused
Postcondition:	Game resumes

Use Case #: 4

Name:	Restart the game
Trigger:	As the user press ”r”
Precondition:	The game is in progress
Postcondition:	Game restarts

Use Case #: 5	
Name:	Rotate the tetromino
Trigger:	As the user press ”up”
Precondition:	The game is in progress
Postcondition:	The current tetromino is rotated clockwise by 90 degrees

Use Case #: 6	
Name:	Tetromino Movement
Trigger:	As the user press ”left” or ”right”
Precondition:	The game is in progress
Postcondition:	The current tetromino is moved left/right by one grid

Use Case #: 7	
Name:	Tetromino Movement
Trigger:	As the user press ”down”
Precondition:	The game is in progress
Postcondition:	The current tetromino falls to the bottom

Use Case #: 8	
Name:	Clear up a row(s)
Trigger:	RoClearroww(s) filled with ”squares”
Precondition:	The game is in progress
Postcondition:	One row or multiple rows are cleared up and everything above shall fall

Use Case #: 9

Name: Player uses items
Trigger: When the user press ”1” or ”2” or ”3” or ”4”
Precondition: The game is in progress & Player has enough experience to ”buy” the corresponding item
Postcondition: Transaction is successful and item is used at once

Use Case #: 10

Name: ~~Player uses items~~
Trigger: ~~When the user press ”1” or ”2” or ”3” or ”4”~~
Precondition: ~~The game is in progress & Player has enough experience to ”buy” the corresponding item~~
Postcondition: ~~Remind the user ’you don’t have enough experience’~~

Use Case #: 11

Name: Game over
Trigger: Part of a new tetromino(or the whole tetromino) cannot be displayed in the game playfield
Precondition: The game is in progress
Postcondition: The game is over

2.2 Functional Requirements

FR1: Interface
Description: ~~We bought a cloud server and a public IP address, So the project's URL shall be accessed by any user with the Internet and any component of this game shall be displayed successfully.~~ The user shall have access to the website.
Rationale: User shall be able to open the website
Fit Criterion: the Super Tetris website is opened after inputting the right website address

FR2: Playfield
Description: ~~The height and the width of the playfield shall be 20 and 10.~~ The playfield shall be stable and unchangeable ~~throughout the game and any part of Tetrimino shall not exceed the playfield's board.~~
Rationale: throughout the game and any part of Tetrimino shall be in the right place
Fit Criterion: No Tetriminos exceeds the playfield in entire time.

FR3: Tetriminoes
Description: ~~There are 7 Tetriminoes with different shapes in the game. Every Tetrimino is made up of four 1*1 squares. Any Tetrimino shall be displayed androtated correctly~~. Any Tetrimino shall be able to be rotated
Rationale: User clicks the rotate button.
Fit Criterion: The Tetrimono is rotated correctly.

FR4: Tetriminoes' color
Description: ~~Every Tetrimino shall be painted in 7 different colors to make it easier for users to classify them.~~ Users shall easily classify each kind of Tetriminos. **Rationale:** Each kind of Tetriminos has different color
Fit Criterion: Tetrimino colors: Cyan, Orange, Red, Blue, Green, Purple, Yellow for each kind of Tetriminos.

FR5: ~~How to~~ Steer the Tetrimino Use case: #5,#6,#7
Description: ~~The cursor keys shall steer the Tetriminoes correctly(Up arrow is to rotate the Tetrimino, down arrow is to let the current Tetrimino fall, left and right arrow is to move the Tetrimino left or right by one grid)~~User

is able to steer the Tetriminos.
Rationale: User clicks the steer button.
Fit Criterion: The Tetrimono is steered correctly.

FR6: Game control Use case: #2,#3,#4
Description: ~~Press r shall restart the game and Press enter can pause the game.(When the game is paused, press enter again shall resume the game)~~User is able to restart the game.
Rationale: User shall be able to click the a button to restart the game.
Fit Criterion: The game is restarted when restart button is clicked.

FR7: Next Tetrimino
Description: There is an individual field to display the next Tetrimino. ~~The next Tetrimino is random and the Tetrimino displayed in this field shall appear in the playfield at once.~~
Rationale: The next Tetrimino is random and the Tetrimino displayed in this field shall appear in the playfield at once.
Fit Criterion: the Tetriminos are correctly displayed

FR8: Collision
~~When a Tetrimino's any part's bottom touches another Tetrimino or the bottom of the playfield, the Tetrimino shall stop falling immediately. Every two Tetriminoes cannot take up common grids and any part of Tetrimino can't exceed the playfield.~~
Description: The Tetriminos shall stop dropping at a cuter case.
Rationale: The Tetriminos shall stop at the bottom of the playfield.
Fit Criterion: Tetriminos stops at the bottom of the playfield.

FR9: Clear up one line Use case: #8
~~After a Tetrimino falling, the game shall check if any row is filled. If so, rows filled with Tetriminoes shall be cleared up and everything above shall fall.~~
Description: A horizontal line shall be cleared
Rationale: When Tetriminos form a horizontal line the line shall be cleared.
Fit Criterion: When Tetriminos form a horizontal the line shall be removed from the screen.

FR10: Game over Use case: #11
Description: The game shall be ended

Rationale: The game shall be ended on a certain circumstance
Fit Criterion: When part of a new Tetrimino(or the whole part) cannot be displayed in the game playfield because of collision with other Tetriminoes, the game shall be over.

FR11: ~~Speed of~~ falling
~~When the game starts, the original speed of falling shall be 3 grids per second. And the speed of falling shall accelerate as more and more rows are cleared up.~~
Description: The Tetriminos shall fall.
Rationale: Tetriminos shall fall from the top to the bottom when appearing.
Fit Criterion: The Tetriminos move from the top top the bottom at a certain speed.

FR12: Score ~~and Experience~~ system
~~The game shall have a scoring system and an Experience system. How many rows a user has cleared up and how much time a user stays alive shall determine their score. But only how many rows a user has cleared up determines their exp. And if users clear up two or more rows at the same time, they can have extra experience. is for buying items that shall help them clear up more rows.~~
Description: User shall score
Rationale: When Tetriminos form a horizontal line the user gain scores.
Fit Criterion: When Tetriminos form a horizontal the score shall increase.

FR13: Item Use case: #9,#10
~~There are four different items listing to the left of the main playfield. Each item has a price and users can buy them by experience you have earned. Press 1 to 4 to use the different item. If users do not have enough experience, the transaction shall not be successful.~~
Description: The system shall tell what items it has
Rationale: User shall know what items the system has and can be used.
Fit Criterion: The items are displayed on the screen

FR14: Use Items Use case: #9,#10
~~There are four different items listing to the left of the main playfield. Each item has a price and users can buy them by experience you have earned.~~

~~Press 1 to 4 to use the different item. If users do not have enough experience, the transaction shall not be successful.~~
Description: User shall use the items
Rationale: User shall use the items
Fit Criterion: The items shall displayed and have its own function when clicking the items buttons

FR15:~~Slow(item2)~~ Use case: #9,#10
~~When the user presses 2, the falling speed of tetrominoes shall be halved and this effect shall last for 30 seconds. After that, the falling speed shall return to original speed.~~

FR16: ~~Skip(item3)~~ Use case: #9,#10
~~When the user presses 3, the next tetromino showed at the next tetromino field shall be replaced with another random tetromino.~~

FR17:~~Clear(item4)~~ Use case: #9,#10
~~When the user presses 4, the top row which contains any testimonies shall be cleared up immediately.~~

3 Non-functional Requirements

3.1 Look and Feel Requirements

LF1: The falling of the tetrimino and line clear require animation. (Animation is required when the tetrimonos are falling or when a row is eliminated.)

3.2 Usability and Humanity Requirements

UHR1: Super Tetris shall follow the same rule in the Classic Tetris.

UHR2: Control commands, item bar, stop button and other event trigger button are visible and easy to learn. see list of event.

3.3 Performance Requirements

PR-Speed1: Super Tetris shall run at 30 fps, and it shall catch the input within 100 milliseconds.

PR-Safety1: Super Tetris shall not trigger any physical damage to the users.

PR-Precision1:The event shall happen at the correct time. The boom shall explode when it stops. The ice shall slow the falling time when it is triggered. The line shall be cleared when a complete line is made. see list of event.

PR-Reliability1:The program should be usable 24 hours per day, 365 days per year or 366 on leap years as long as the server is not expired.

PR-Capacity1:Super Tetris needs less than 10MB in memory only if the browser want store the websites data to increase the loading speed.

3.4 Operational and Environmental Requirements

OER1: Super Tetris shall run on all major web browsers (Chrome/Firefox /Safari /IE)

3.5 Maintainability and Support Requirements

MR1: The developers should be in charge of maintenance, so the contact information should show in the end of the website page for the players to report bugs and performance issues.

3.6 Security Requirements

SR2:Super Tetris shall not collect users data such as email, username and personal information.

3.7 Cultural Requirements

CR1: Super Tetris shall instruct player in English.

CR2: Super Tetris shall keep the original Tetriss Playing Method which is to obtain the score from line clearing
CR3: The game shall not contain any violence and pornographic element.

3.8 Legal Requirements

LR1: Super Tetris is based on an open source project. Super Tetris shall abidance by law.

3.9 Health and Safety Requirements

HSR1: ~~The game shall not contain any violence and pornographic element.~~
HSR1: The game shall not cause finger strains, we separate the major control key and it will forces user use two hand to play the game.

4 Project Issues

4.1 Open Issues

When playing the game, The direction buttons may cause the screen moving instead of playing functions to the game. Also, as the game will be played on the Internet, a different browser may be used to play the game. As a result, the interface of the game may not fit the screen in some versions of the browsers.

4.2 Off-the-Shelf Solutions

There are many versions of Tetris on the Internet such as Welltris, Harris, Tetris Plus and so on. The Tetris Plus has some items added, such as bombs. Some others added level to the game. Many components including items, level systems are already made in some games. These elements are same or similar to the Super Tetris we are supposed to make. As result, there are many redeveloped Tetris game we can follow or copy.

4.3 New Problems

Finding right animation associated with the game will be a problem. Numerous tests will be needed to check an ~~anime~~ animation is played at the right time with right function and does not affect the gameplay. the ~~anime~~ animation may have a larger size than it is supposed to, which, may result in lack of memory or lag. Every ~~anime~~ animation needs to be adjusted frequently or changed in order to make the game look fantastic without affecting the gaming function.

4.4 Tasks

The check date of tasks below will show in the Gantt Chart.

4.4.1 Step

1. Know the logic by studying the code in the original project.

2. Learn how to build the new interface by CSS.

3. Build the interface for score and experience and make sure they increase by some certain algorithms we mentioned in requirements.

4. Build the interface for game items and create the animation and sound when using game items.

5. Make sure every game item's function does work as we mentioned in requirements

6. Put all the codes in our cloud server and buy a domain for this game to make sure everyone can play our game by our domain

4.4.2 Phase

Table 4: List of Phase

Phase Name	**Summary**
1.User Interfaces	Upgrade to include Score, Experience, and game items
2.Animation	about clearing up rows and bomb explosion
3.Sound	about clearing up rows and using items
4.Item functionality	Items' functions shall work as requirements
5.Testing	Test the game

4.5 Migration to the New Product

The new product shall add some more elements(Score, Experience, items) by upgrading the interface, but it won't interfere the original interface. Game items shall work as requirements, but it won't influence the original game mechanics or cause bugs. After transferring everything to the server, we shall test the project carefully on different devices.

4.6 Risks

There are several risks while proceeding the project. One of the risks is that graphics, ~~anime~~ animation , and buttons will be added to the game. These features are supposed to connect each other which makes the program complex to implement.

To overcome the program is very complex to implement, the project will be tested more frequently. After each element added to the program, numerous tests will be needed. If the first element is added successfully, then other elements are allowed to be added. In term of connecting each element, one team member will keep researching the way to make all the elements effectively connected. Also, each member will be assigned to research a certain type of elements so as to let them functional necessary.

During the-the Proof of Concept Demonstrations, all the elements should be tested. The goal is to make all the elements work excellently without bugs. However, if a bug appears during playing the game, the game will shut down, and a line says Please contact the developer. will pop up.

4.7 Costs

As the game runs on an Internet server, an annual fee of 70 Canadian Dollars will be needed. However, the whole project will be developed using our own laptop, there will be no other fee needed. For this project, it will be finished within a year. As a result, the total cost of the project is 70 Canadian Dollars.

4.8 User Documentation and Training

As Super Tetris is similar to the original Tetris, only a little User Documentation will be needed. There are less than ten buttons needed. A simple instruction will be shown at the beginning of the game. While playing, a small picture will appear on the right-top corner of the screen which indicates how to use the items. There is no training for the players.

4.9 Waiting Room

After the Super Tetris is finished, numerous new features can be added. We can add PvP system(People vs People System) to the game. People will be able to play with a stranger on the server. While playing with each other, some debuff items shall be added, which makes the game more competitive. Additionally, a friend system can also be added. Thus, the users are able to play with their friends using their own account on the server.

4.10 Ideas for Solutions

For programming, all elements should be added one by one and after an element is added, hundreds of test will be implied to the certain element to make sure it works fine. Moreover, there are numerous redeveloped Tetris similar to our program. Thus, if issues occur when programming, there are many samples we can consult or learn from.