

SE 3XA3: Software Requirements
Specification
Title of Project

Team #12, Team Name
Student 1 Eric Thai Thaie1
Student 2 Tianzheng Mai and mait6
Student 3 Linqi Jiang and jiangl21
Student 4 Junhong Chen and chenj297

February 12, 2021

Contents

List of Tables

List of Figures

This document describes the requirements for Asteroid Game Development. The template for the Software Requirements Specification (SRS) is a subset of the Volere template (?). If you make further modifications to the template, you should explicitly state what modifications were made.

1 Project Drivers

1.1 The Purpose of the Project

The goal of this project is to improve the user interface, functionality, and implementation of the old Asteroid Game. Improving the user interface of the original game can significantly enhance the user experience because the players will be able to play this game with better qualities. More importantly, this project also aims to improve the functionality and implementation by reproducing a two players mode and a rank system which provides a method for users to evaluate their game performance with others.

1.2 The Stakeholders

The stakeholders include the clients, customers, other stakeholders.

1.2.1 The Client

The clients are the professors, teaching assistants of Software Engineering 3XA3 who are responsible to supervise the project and keep track of the project process.

1.2.2 The Customers

The intended customers are the public people who have an intermediate understanding of the English language, interests in Asteroid, and the ability to use a computer device, mouse, keyboard, and web browser.

1.2.3 Other Stakeholders

Indirect Stakeholder: The indirect Stakeholders are all developers in team 3XA3 Lab3 Group12 who are responsible to develop, modify, maintain the Asteroid Game, and achieve all targets and requirements in SRS.

1.3 Mandated Constraints

1.3.1 Time Constraint

The entire project must be completed and submitted by April 12th, 2021.

1.3.2 Cost Constraint

There is no cost for this project because the required software and open-resource codes are free to use.

1.3.3 Product Constraint

This product requires users to have the access to an efficient internet connection and a web browser with a computer, keyboard, and mouse.

1.3.4 Knowledge Constraint

The users must have an intermediate understanding of the English language and the basic knowledge to browser a website.

1.4 Naming Conventions and Terminology

1.4.1 JavaScript

A high-level, object-oriented computer programming language conforms to the ECMAScript specification. It is commonly used to create interactive effects in a web browser.

1.4.2 HTML

Hypertext Markup Language (HTML) is the standard markup language for documents designed to be displayed in a web browser.

1.4.3 CSS

Cascading Style Sheets (CSS) is a style sheet language designed to enable the separation of presentation and content, including layout, colors, and fonts.

1.4.4 Node.js

an open-source, cross-platform, back-end JavaScript runtime environment built on Chrome's V8 JavaScript engine.

1.4.5 JSON

JavaScript Object Notation (JSON) is an open file and data interchange format to store and transmit data objects consisting of attribute-value pairs and array data types.

1.5 Relevant Facts and Assumptions

Asteroids will assume that the intended users must at least maintain an intermediate understanding of the English language and the ability to browse a website with using a computer, keyboard, and mouse. However, this application will not require the users to have any other technical background or education because a user manual will be provided on the main page of Asteroids.

2 Functional Requirements

2.1 The Scope of the Work and the Product

2.1.1 The Context of the Work

Asteroids is a browser game in which the user can control a spaceship and shoot asteroids while avoiding getting hit by them. The context of the work is to rebuild and modify Asteroids, including editing the existing code for better readability and maintainability, improving the interface of the game by adding some colors and images, adding some advanced features like score counting, difficulty level selection, leaderboard and two-player mode.

Here are the deadlines and deliverables of the project:

- Project Approval —Week of January 18
- Problem Statement —Week of January 25
- Development Plan —January 29

- Requirements Docs Revision 0 —February 12
- Proof of Concept Demonstration —Week of February 22
- Test Plan Revision 0 —March 5
- Design and Document Revision 0 —March 18
- Revision 0 Demonstration —Week of March 22
- Final Demonstration (Revision 1) —Week of April 5
- Final Demonstration (Revision 1) —Week of April 12
-

2.1.2 Work Partitioning

Event Number	Event Name	Input	Output
1	Asteroids Creation	developer code	web browser
2	User Interaction	developer code	web browser
3	Graphics Creation	developer code	web browser
4	Game Audio	audio file	sound effect
5	Asteroids Final Changes and Review	developer code	web browser

Table 2: Work Partitioning

2.1.3 Individual Product Use Cases

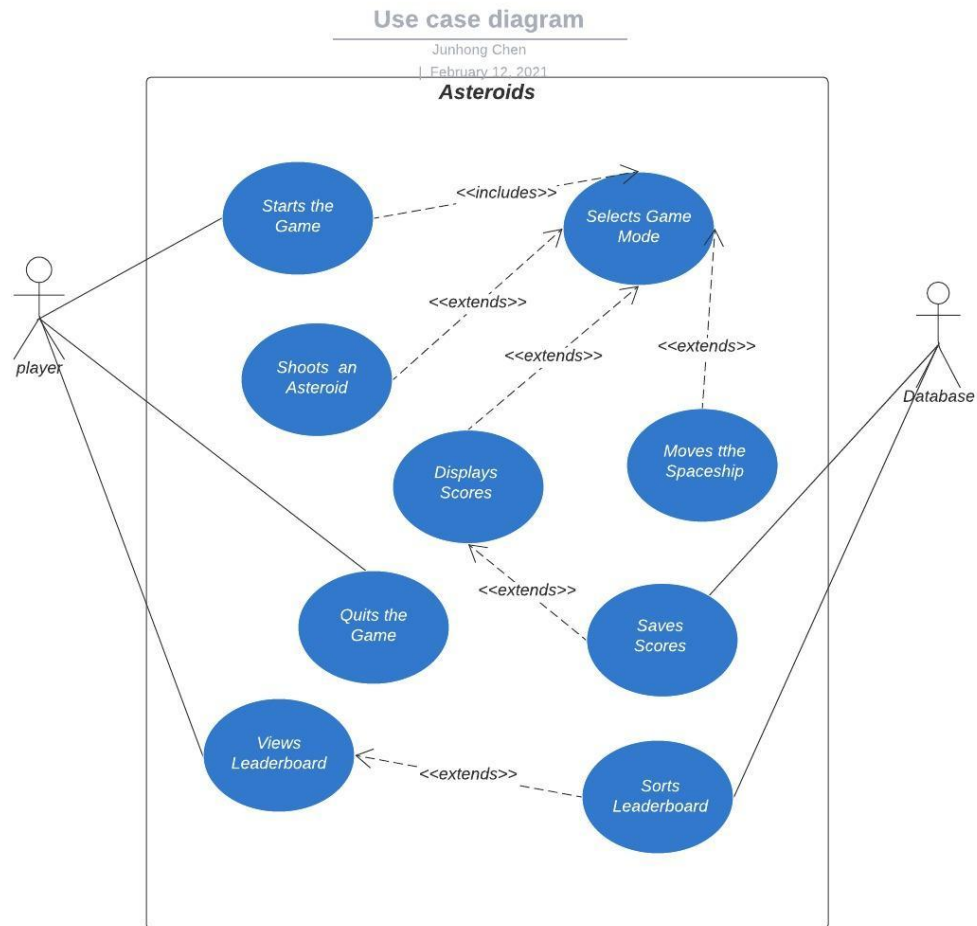


Figure 1: Use Case Diagram for Asteroids

2.2 Functional Requirements

FR1. The users must be able to access the website and play the game.

Fit Criterion: The user will be able to access the game by inputting

the link to the web browser and clicking the start button to start the game.

- FR2. Scores should be shown on the screen to the users during the game.
Fit Criterion: The game will display the scores that the player gains when playing the game.
- FR3. When the spaceship is hit by an asteroid, the lives of the spaceship will be reduced by 1.
Fit Criterion: The lives of spaceships will be reduced by 1 when collision of it and an asteroid occurs.
- FR4. When the lives of the spaceship is zero, the game will be over and the result will be shown.
Fit Criterion: The game will generate a result message and display the scores after the player loses three lives.
- FR5. The game will generate a battlefield for the player.
Fit Criterion: The website interface will generate a battlefield that allows the user to control the spaceship to fly around in a limited size.
- FR6. The game will generate a spaceship for the player.
Fit Criterion: The website interface will generate a spaceship that allows the user to control by arrow keys three lives and is able to shoot the asteroid to destroy them by pressing space.
- FR7. The main page of Asteroid will include a user manual for the player.
Fit Criterion: The users could view the instructions and controls of the game by viewing the user manual.
- FR8. The game will include a stop button on the side.
Fit Criterion: The users can stop the game by clicking on the stop button.
- FR9. The battlefield interface must display the information of the spaceship.
Fit Criterion: The battlefield will illustrate the spaceship information on the upper left corner which contains the spaceship lives and current scores.

3 Non-functional Requirements

3.1 Look and Feel Requirements

- The application must follow design standards and have a pleasant design
- The application should draw user's in from its look

3.2 Usability and Humanity Requirements

- The application must be familiar to other browser games such that the controls are easy to use and learn.
- The application must be able to be used by users with a standard QWERTY keyboard.
- The application must provide instructions that can be understood by users that can read basic English.
- The application must have a user interface that is friendly for those that are color blind
- The user interface must follow web accessibility standards

3.3 Performance Requirements

- The application responds with very little to no latency

3.4 Operational and Environmental Requirements

- The application shall be able to run by opening an HTML file on a browser
- The application will be usable on a standard web page size.

3.5 Maintainability and Support Requirements

- The application shall be able to run on the majority of the most popular web browsers
- The application shall be able to run on Windows and MacOS

3.6 Security Requirements

- The Application's source code shall be open source.
- Changes made to the application shall be traced back to the user who has made the changes

3.7 Cultural Requirements

- None

3.8 Legal Requirements

- The product shall comply with WCAG standards

3.9 Health and Safety Requirements

- The application must not have flashing lights which could cause an epileptic user to experience a seizure

4 Project Issues

4.1 Open Issues

The old game was initially implemented as a black and white rough interface. Our group will redesign the user interface and add some new features to the game so as to enhance the performance and quality of the game.

4.2 Off-the-Shelf Solutions

4.2.1 Existing Products

There are many existing products for that have created the game of Asteroid, They are listed below:

Asteroid online game:

<http://www.dougmcinnes.com/html-5-asteroids>

Asteroid online scoreboard:

<https://games.aarp.org/games/atari-asteroids>

Improved classic Asteroid game - webstore:

<https://chrome.google.com/webstore/detail/asteroids-classical-aster/pdjmcidindccobjfklphgnefglooeidl>

For this project, there are a lot of examples on the website, to create a better version of Asteroid, one of the important steps is to investigate these existing solutions, analyzing the advantages and drawbacks and improve the implementations and meet the project requirements.

4.3 New Problems

4.3.1 Problems in System

The project must be run on a web browser. The game will be interrupted and stop running if the system server crashes or the internet connection is lost.

4.3.2 Problems in Compiling Environment

The game interface must be compatible with diverse devices and browsers which the users can play this game on different sizes of screen.

4.4 Tasks

This is the link of Gantt Chart:

<https://gitlab.cas.mcmaster.ca/mait6/3xa3-lab3-group-12/-/blob/master/BlankProjectTemplate/ProjectSchedule/ganttchart.pdf>

4.5 Migration to the New Product

N/A

4.6 Risks

Testing is a risk on this project because it can not find out all potential errors which may be made in the development process.

4.7 Costs

There is no cost in this game because the codes are open-resource and the software is free.

4.8 User Documentation and Training

N/A

4.9 Waiting Room

N/A

4.10 Ideas for Solutions

HTML and CSS will be used to improve the user interface.

JavaScript will be used to develop some new features and improve the functionality of the game.

JSON will be used to store and update the game data.

5 Appendix

This section has been added to the Volere template. This is where you can place additional information.

5.1 Symbolic Parameters

The definition of the requirements will likely call for SYMBOLIC_CONSTANTS. Their values are defined in this section for easy maintenance.

Table 1: **Revision History**

Date	Version	Notes
2/12/2021	1.0 Projec Drivers	Tianzheng Mai
2/12/2021	2.0 Func- tional Require- ment	Junhong Chen
2/12/2021	3.0 Non- Functional Require- ment	Eric Thai
2/12/2021	4.0 Project Issues	Linqi Jiang