Stellar Fighter

Software Requirements Specification

Human ICT Software Engineering

20162604 이병곤

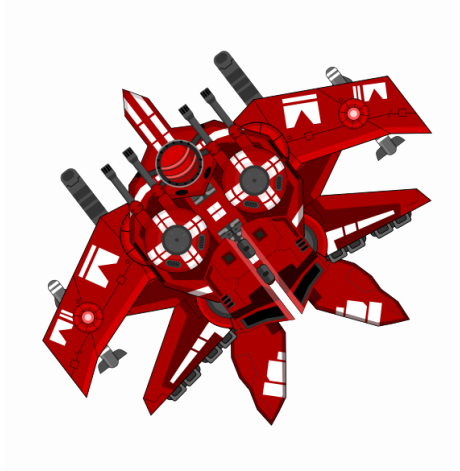20150798 이윤주

20166477 권우람

20153784 전호설

20171327 조민아

20153418 박채린

Table of Contents

1. Product Description

Stellar Fighter is a classic air shooter game in HTML5.

You may think this kind of classic arcade game may look too common.

However, given a little twist on graphics, enemies, weapons, items or a storyline, the game can be unique and attractive.

Also there's an advantage of these simple kind of games. We don't need to use a game engine because of its simplicity. Since we don't use an engine, we will have opportunity to study and implement a game loop, collision detection, scene graph, vector calculation algorithm.

Using HTML5, there's no need for users to install additional softwares; Only a web browser is all they need.

2. Functional requirements summary

- The user shall be able to go to settings, scores, credits, play page from main menu.
- The user shall be able to pause the game and go to other pages or resume.
- The user shall be able to move, shoot bullets, cast skills such as missiles.

3. Use Case Diagram

When users start game system, users can perform four actions on the game system.

First, users can confirm the game scores.

In addition, users can change settings such as sound by pressing the setting button.
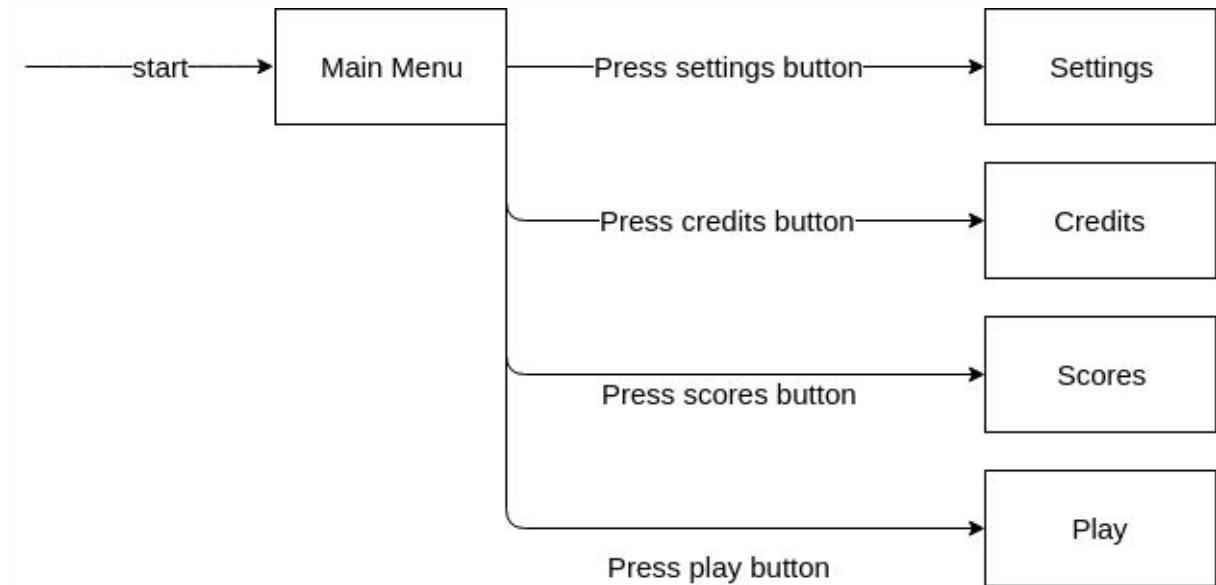
Users can also check out developers and designers through credit pages.

Finally, you can play the game with play button, which is the most important function.

In game play, you can improve your scores by moving a spaceship, shooting bullets, eating items, or avoiding enemies.
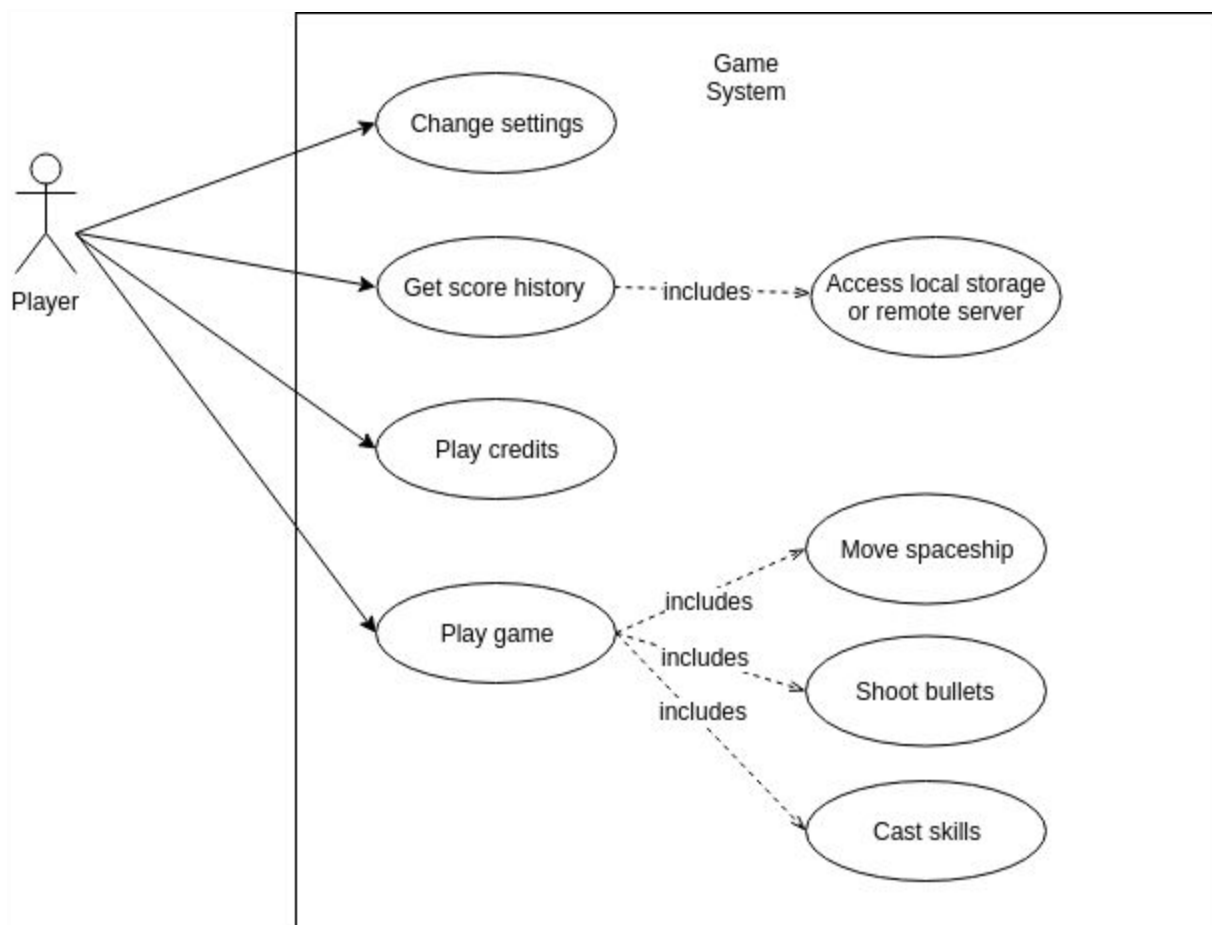
4. UI (User Interface) Diagram

When the user first accesses the game, he or she enters the main menu, where he or she can enter four different pages.

The first page is the setting page that you can enter by clicking the 'Setting button'.

The second page is a credits page that you can enter by clicking the 'Credits button'.

For each page, you can use the 'Back button' to navigate to the main menu.
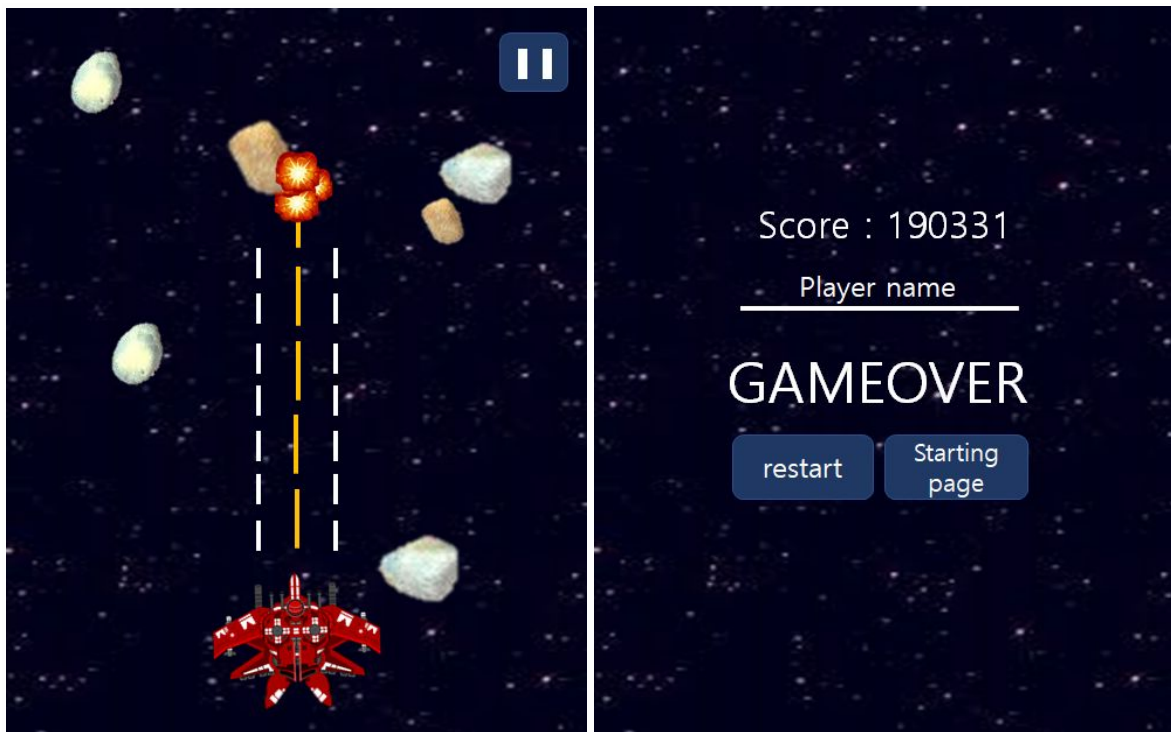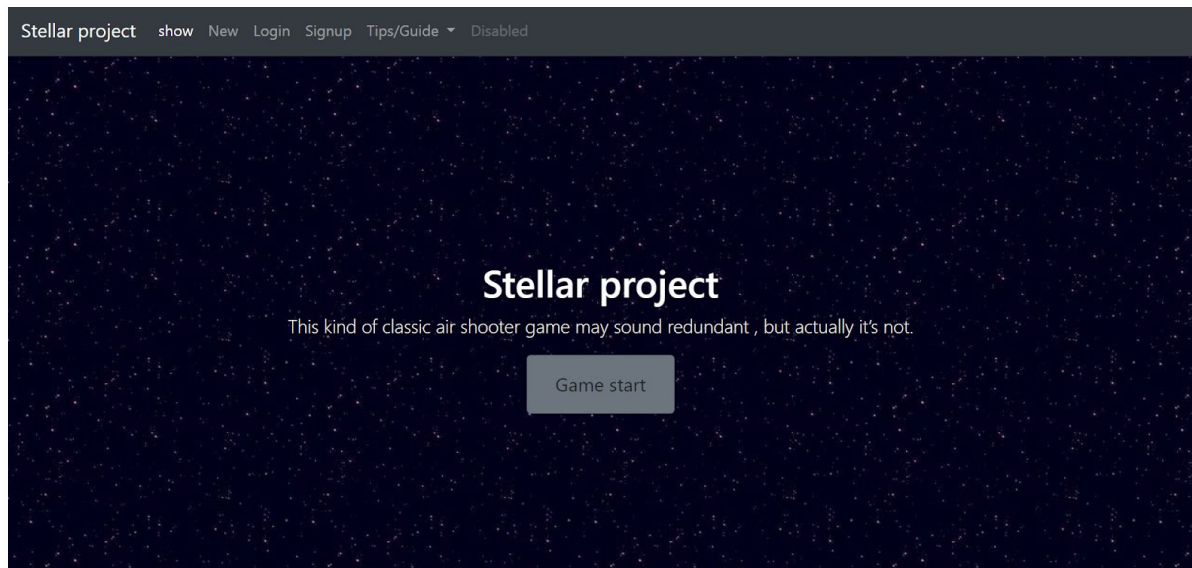
Finally, you can start the game by clicking the Play button.
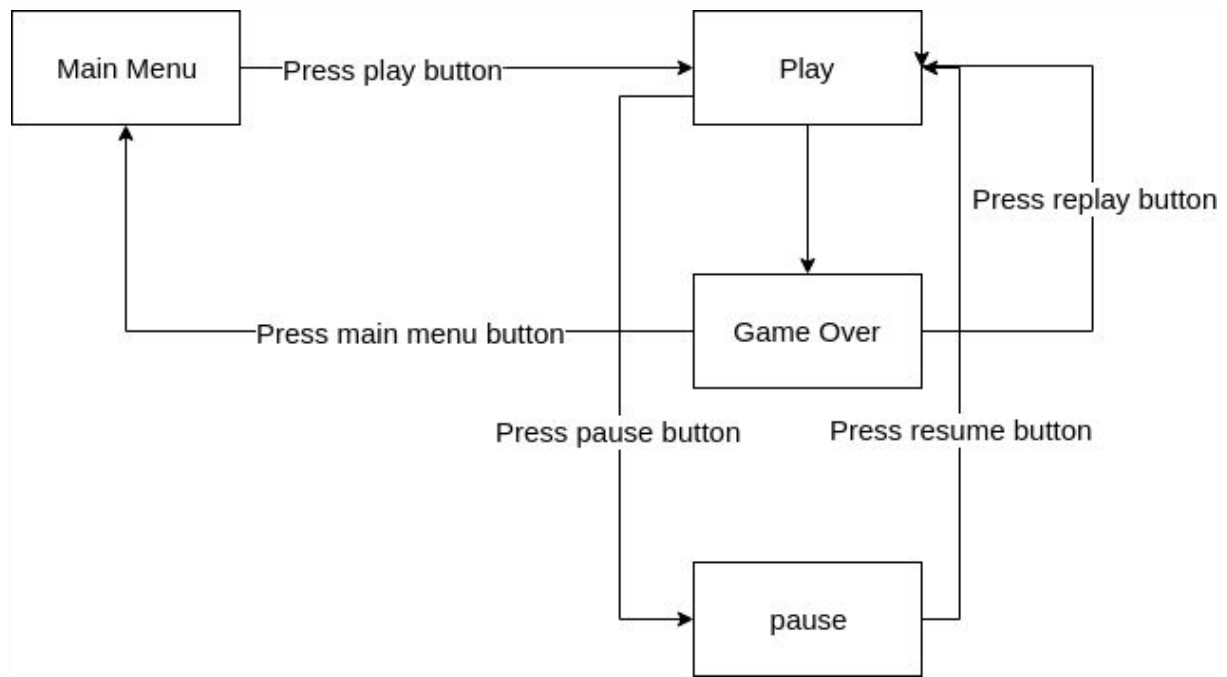
On the Play page, you can take a break from the game through the 'Pause button'.

Once the game is over, you can go to the Game Over page, go back to the Play page via the Restart button, or return to the main menu through the Restart button.

5. UI (User Interface) Examples

6. Non functional requirements

- This software shall be available in both desktop and mobile devices.
- This software shall be working in Chrome.
- Stable version shall be available on github pages.

7. Process Description

7.1 Software Tools

- any text editors

- node.js

- web browser

- git

7.2 Roles and Responsibilities

- **Main developers**: 이병곤, 이윤주

- **Designers**(디자인), **Sub-developers**: 권우람 조민아

- **Planners**(기획), **Sub-developers**: 박채린, 전호설

7.3 Schedule

Date	Activity
3.06 ~ 3.13	Project Proposal
3.14 ~ 3.20	Voting & Team set up
3.21 ~ 3.27	Planning Project & SRS
3.28 ~ 4.03	Writing SRS
4.04 ~ 4.10	Building The Development Environment Setting up UI Model
4.11 ~ 4.17	Setting Game Rules, Learning the Development Knowledge
4.22 ~ 4.26	Mid-term Exam
4.27 ~ 5.1	UI Design
5.2 ~ 5.8	Developing the Program (1)
5.9 ~ 5.15	Developing the Program (2)
5.16 ~ 5.22	Developing the Program (3)
5.23 ~ 5.29	Implementation of Additional Function
5.30 ~ 6.5	Test And Debugging
6.6 ~ 6.12	Project Presentation
6.13 ~ 6.19	Submit Final Project Document

7.4 Milestones

- Single player with same kind of enemies and bullets.
- If possible, we could implement following milestones.
- Use ECS design pattern.
- Single player and variant kinds of enemies.
- Add guided missiles.
- Add various items.
- Use particles.
- Use Quad Tree algorithm for collision detection.
- Use pixel-perfect collision detection.
- Use scene graph.
- Add maps, boss enemies and a scenario.
- Add another player.

7.5 Expected Risks

- Due to lack of experiences about web game development, our team needs some time to learn and get adapted.
- We have to draw or fetch design resources from internet for the project.

8. Team Website

- github organiztion: https://github.com/stellar-fighter
- github pages: https://stellar-fighter.github.io

9. License

Since the source code of this project is open, we decided to choose the license **MPL-2.0**