Super Sea Lion

A 2D Web Game Made for CS537 Software Engineering

#### Credits

**Lead Developer:** [Yinchu Xia](http://github.com/warycat)

**Animation Developer:** [Yuanlin Xu](http://github.com/yxun)

**AI Developer:** [Mark Sargent](http://github.com/detroitteatime)

**Level Developer:** [Chootiraka Niraporn](http://github.com/too87)

**Document Change Log**

|  |  |  |
| --- | --- | --- |
| **Update** | **Date Released** | **Changes** |
| Draft #1 | 4/17/2014 | Introduction and weekly progress report |
|  |  |  |
|  |  |  |

**Table of Contents**

**1.0 INTRODUCTION**

1.1 General Description 1-1

1.2 Story 1-2

**2.0 Weekly Progress 2-0**

**A. DATA DICTIONARY A**-1

**B. ACRONYMS** B-1

**List of Figures**

None.

**List of Tables**

None.

**1. INTRODUCTION**

**1.1 General Description**  1-1

Our project is a 2D web game. It is based on Pixi.js rendering engine and node,js platform. We will develop an action type game including 2D map rendering, skeleton animation, multiple levels with a story board and Boss challenges. The game will have a 2D map which contains half above water images and half under the water images. When the game starts, the player will control a running or swimming character SSL(“SuperSeaLion”) to jump, across barriers, fight with enemies or fight with the boss. We decided to develop our animation by using Spine skeleton animation tool. In the nearest future, we may implement it into android and IOS mobile device platform.

[Project Home Page](http://warycat.github.io/SuperSeaLion)

**1.2 Story** 1-2

Sea lions have been disappeared from the bay. One day our hero, Super Sea Lion (SSL), sees a boat capture a sea lion and sail away. SSL is determined to find out what has been happening to his friends and relatives, so he follows the boat. Along the way, SSL overcomes various enemies and bosses. At the end of the game SSL discovers that the sea lions have been enslaved in an evil sea creature show. SSL must defeat the evil show master and his henchmen to free the sea lions, dolphins, and killer whales locked up in the tanks.

The game is comprised of levels with a story board between each level indicating SSL's progress in the story. SSL travels through the levels, eating fish, squid, lobster, clams, and crab for points. SSL must either avoid or defeat enemies in his way. At the end of each level he defeats a boss. In the final level SSL defeats the Big Boss and frees the enslaved sea creatures.

**2.0 Weekly Progress 2-0**

Last we had established our project webpage, [Project Home Page](http://warycat.github.io/SuperSeaLion)

And a wiki page, <https://github.com/warycat/SuperSeaLion/wiki>,

The majority of our work has been reported on the issue page on github.

<https://github.com/warycat/SuperSeaLion/issues?state=open>

1. We have designed the basic idea of three layer map
2. We have named the bosses as follows: Evil Aquarium Master, Kraken, Sharky and Killer Lobster. There are more detail story descriptions in our wiki page, <https://github.com/warycat/SuperSeaLion/wiki/Bosses>
3. We have committed bosses and enemies json files.

<https://github.com/warycat/SuperSeaLion/commit/fd5478424611b546d5dae23a69ce1eb0d12a64cd>

1. The first skeleton animation has been committed. And we also provided several candidate SSL character images in the issue, <https://github.com/warycat/SuperSeaLion/issues/11>

**A. DATA DICTIONARY** A-1

SSL (SuperSeaLion) Refers to our project name and the main character in our game. We are discussing its image and developing its animations in the third week.

Pixi.js This is a devoted rendering engine. It is also a host of other engines covering game, sound and physics etc. Details, <http://www.pixijs.com/>

node.js is a platform built on Chrome's JavaScript runtime for easily building fast, scalable network applications. Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient, perfect for data-intensive real-time applications that run across distributed devices. <http://nodejs.org/>

**B. ACRONYMS** B-1

**SSL** Super Sea Lion