The Legend of Zelda Mean Stack App
Software Requirements Specification
Version1.5
6/13/2017

Evan Kenney, Jason Costi, Camren Prater

Application Developer

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Evan	4/20	Updated template/ added more material	1.1
Evan	6/13	Updated for final	1.5

1. Introduction

This software requirements specification (SRS) contains all of the information needed to allow a software engineer to implement the requirements within. The SRS contains functional, non-functional requirements, scope, description, and analysis of the project.

1.1 Purpose

The purpose of the SRS is to provide a specification for the Mean Stack web app project.

1.2 Document Conventions

When going into detail on what the product is for, the font is italicised, all other font remains the same size and font-type.

1.3 Intended Audience and Reading Suggestions

The intended audience for this SRS are any users interested in the app. The SRS has been laid out in basic format, to understand how the app is laid out, see the case diagrams and wireframes for reference.

1.4 Product Scope

(1) This application will be the final project for Software Engineering 2.
(2) The application will be able to provide an in game map with clickable cities to provide descriptions and in game items that the towns have.
(3) There will be a counter system that allows the user to see what weapon are used in an advantage over the enemy.
(4) This application will be able to consolidate several Legend of Zelda games and provide an effective k-nn algorithm to show how connected each game is to each other all in one page. (TBD)

1.5 References

The Legend of Zelda series is produced by Nintendo all rights reserved.
The guidebook used to list all the items and towns was published by; Piggyback; Har/Pstr C edition (March 3, 2017)

2. Overall Description

The Legend of Zelda single page application is meant to demonstrate our knowledge of the MEAN stack. The application will act as a guide to help users succeed in the Legend of Zelda games. The application will be able to provide an in game map with clickable cities to provide descriptions and in game items that the towns have. There will be a counter system that allows the user to see what weapon are used in an advantage over the enemy. The application will be able to display all current Zelda games, provide descriptions of each game, display a timeline showing when all the games take place, and use the k-nn algorithm to find similarities. (TBD)

2.1 Product Perspective

This product well be coinciding with the guidebook published by Piggyback, allowing for an interactable guide while not taking the place of its predecessor. Where the book goes into much greater detail, the app will only touch on certain things for a quick and easy to use “look-up”.

2.2 Product Functions

Utilizes the Mean stack
Able to provide a counter system for in game items

2.3 User Classes and Characteristics

There is only one user for the LZ-MEAN. Primary user: Web user. Web users will have the ability to go through all the content provided.

2.4 Operating Environment

Any device capable of browsing the internet at reasonable speeds is required to view the app. All pieces of the app will live on a server where it can be updated and monitored.

2.5 Design and Implementation Constraints

Being a Nintendo Product, we may not in any way sell this product. Even keeping it live for an extended period of time leaves us viable to be sued by the Nintendo Corporation. Because of this, local testing and private repos are the only option until final delivery.

2.6 User Documentation

This Document will be the only artifact available. The app is very simple, with only being a point and click.

2.7 Assumptions and Dependencies

It is assumed that all users will have internet access
It is assumed that all users will be able to navigate to the site via URL or google search
It is assumed that the developers will be maintaining the application to a certain extent

3. External Interface Requirements

3.1 User Interfaces

The app will include the following requirements:

- Ability to zoom in and out on the overworld map
- Matching system working properly
- Ability to click on key locations and villages

.

3.2 Hardware Interfaces

Any device that has access to the internet and can browse at reasonable speeds can interact with the app. Clicking items will bring up new pages and interactions. No user data will be saved.

3.3 Software Interfaces

The product will use MongoDB for it's counter system and be running on the MEAN stack.

3.4 Communications Interfaces

This product does not have any form of communication with the clients, it will merely be a guide for users to refer to.

4. System Features

4.1 Single Page Application Design for the Interactive Timeline

4.1.1 Description and Priority

The SPA design allows for smooth access to the pages in the app. This is of high priority. See img-6.3.5 and img-6.3.6 for screens and img-6.2.2 and img-6.2.3 for use cases.

4.1.2 Stimulus/Response Sequences

Stimulus: User interacts with the timeline

Response: The page changes to the game that the user clicks on without changing url path

4.1.3 Functional Requirements

REQ-1: User must have a device capable of Web-Browsing
REQ-2: User must be able to interact via mouse or touch screen
ERR-1: If User clicks area that does not have any information, no information will appear

4.2 MongoDB Item Matcher

4.2.1 Description and Priority

The webpages built in item matcher allows the user to select an item and see the best match against an enemy. This is of medium to high priority. See img-6.3.10 and img-6.3.11 for screens and img-6.2.5 for use case diagram.

4.2.2 Stimulus/Response Sequences

Stimulus: User selects weapon

Response: The page calls out to the database and generates a list of enemies that are weak against the item.

4.2.3 Functional Requirements

The requirements are a AWS server instance that hosts a Mongo database
REQ-1:Mongoj database
REQ-2:AWS server
ERR-1: If User inputs Enemy/Weapon that does not exist within the Database, a prompt will appear letting the User know that they have entered an invalid Enemy/Weapon

4.3 Interactive Map

4.3.1 Description and Priority

The map of the game will be displayed as the main feature of the home page. A user will be able to interact with key places on the map. When a location is clicked on information about it will be displayed. This is of high priority. See img-6.3.7, img-6.3.8 and img-6.3.9 for screens and img-6.2.4 for use case diagram.

4.3.2 Stimulus/Response Sequences

Stimulus: User clickes on map location
Response: Information about the location (location description, available items, etc)

4.3.3 Functional Requirements

The Requirements are simply that the user has the ability to view the map and interact with the map via touch/mouse.

REQ-1: User must have a device capable of Web-Browsing
REQ-2: User must be able to interact via mouse or touch screen
ERR-1: If User clicks area that does not have any information, no information will appear

5. Other Nonfunctional Requirements

5.1 Performance Requirements

Internet speeds should be around 5mb down.
Browser support has been enabled for:

- Microsoft Edge
- Firefox
- Google Chrome

5.2 Safety Requirements

As with all use of technology, the product may induce a stroke. Long usage may harm the eyes.

5.3 Security Requirements

Currently, a user will be anonymous while using the product.

5.4 Software Quality Attributes

This product is going off of the main guidebook, so all data is correct. As the game will not change, it's maintainability will be very efficient. Being mobile friendly, it will be portable as well as reliable.

5.5 Business Rules

Users may only read the app and interact with it.

6. Other Requirements

6.1 Use Case

Diagram 1

User:U
System:S

U:Clicks on intro page
S: Catches “click” moves to the timeline landing page.

Diagram 3

User:U
System:S

U: Accesses Computer
U: continues with read privileges only

Diagram 4

User:U
System:S

U:Accesses Computer
U: continues with read privileges only

Diagram 5

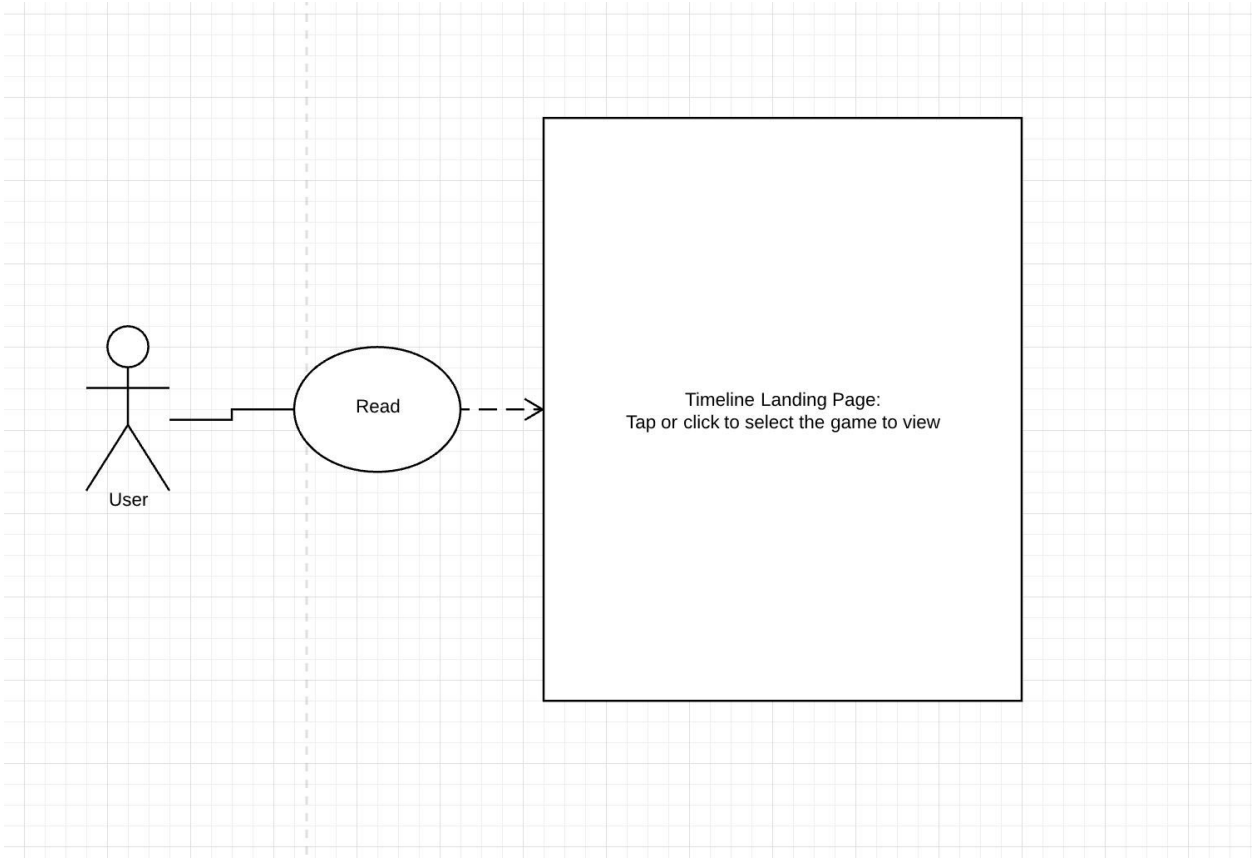
User:U
System:S

U:Accesses Computer
U: continues with read privileges only

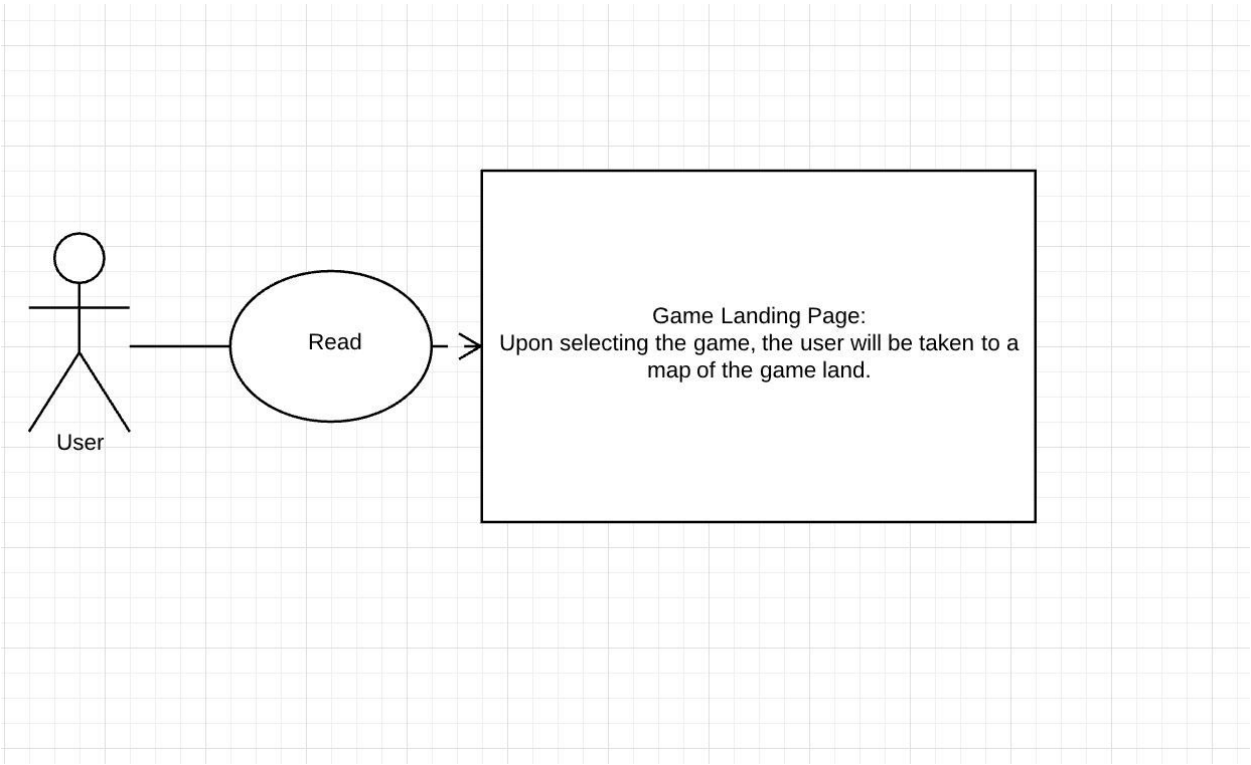
6.2 Use Case Diagrams

img-6.2.1

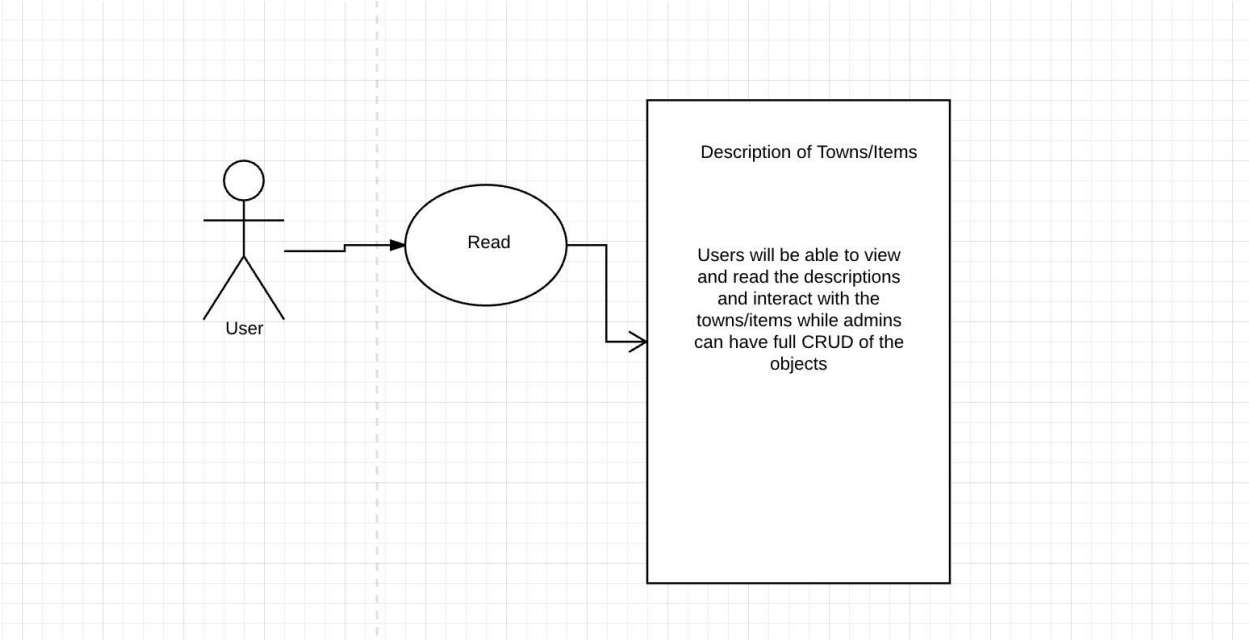
TimeLine



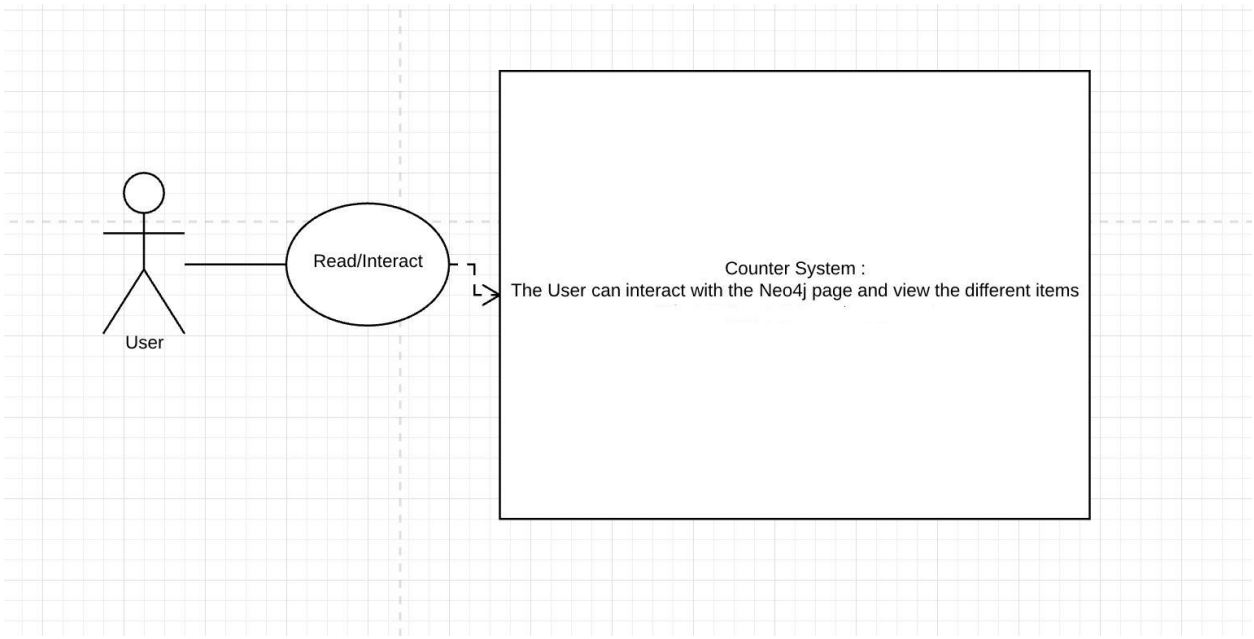
img-6.2.2
Game Map



img-6.2.3
Town Page



img-6.2.4
Counter System

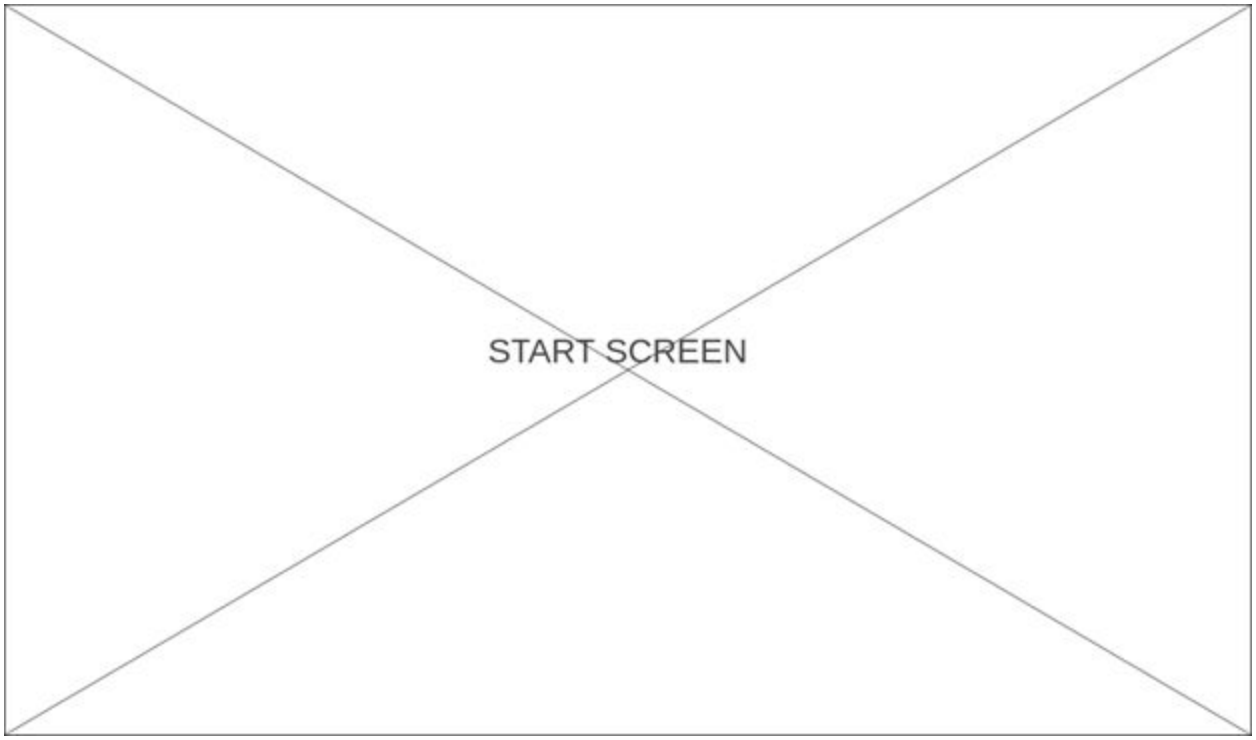


The Neo4j page has been swapped for a MongoDB backend that feeds the data to an angular page.

6.3 WireFrames

img-6.3.1

Intro Page

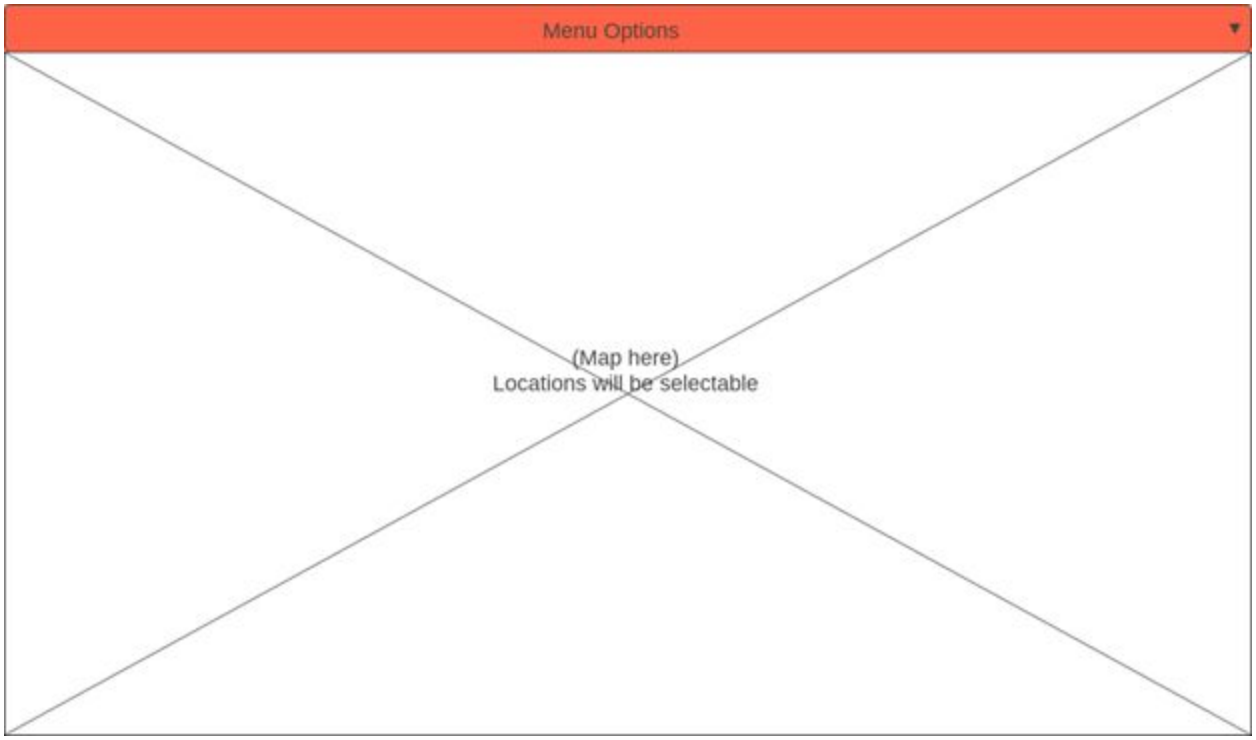


img-6.3.2
Timeline/Landing page

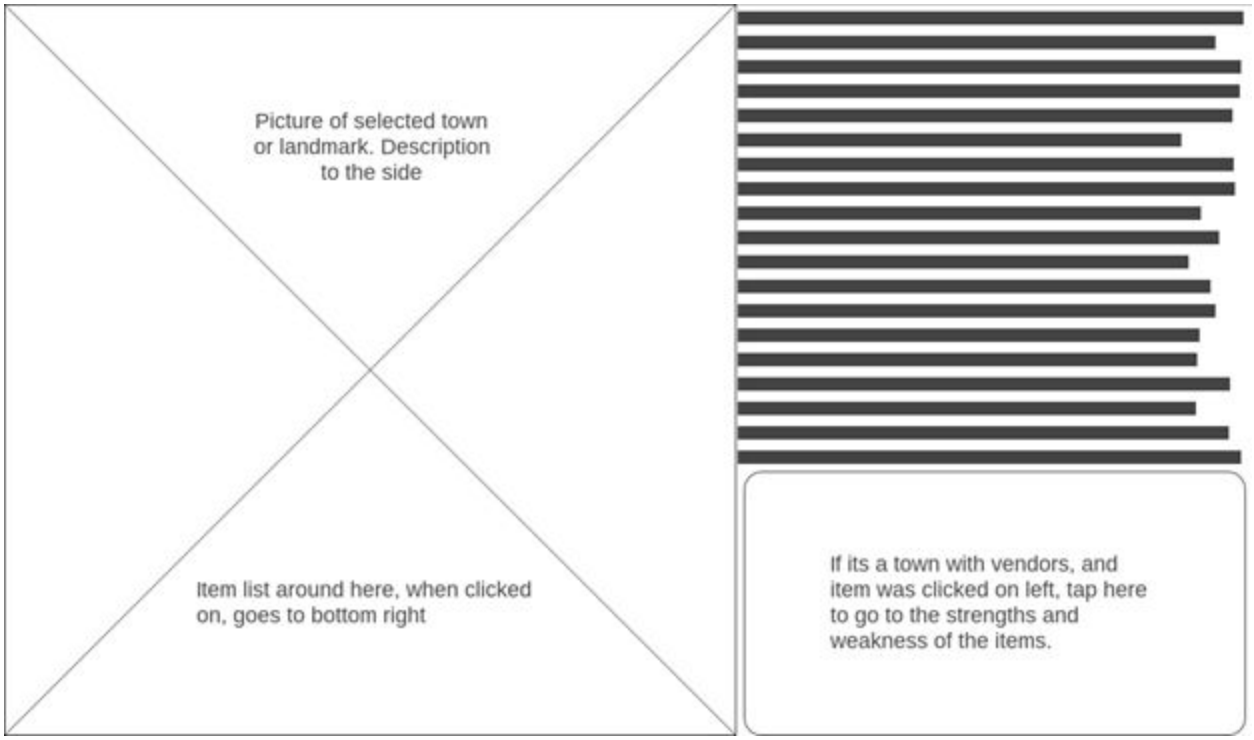
(Currently the Timeline is only setup for The Legend of Zelda Breath of the Wild, other games can be implemented in given more time)

img-6.3.3

Map



img-6.3.4
Town/Location Description Web



img-6.3.5
Town/Location Description Mobile

img-6.3.6
Counter System Web

The Neo4j page has been swapped for a MongoDB backend that feeds the data to an angular page.

img-6.3.7
Counter System Mobile