Detailed Design Document

<Group name>

Oct xx, 2020

Version [0]

**<Logo>**

**Presented To:**

**<client>**

**Submitted By:**

<Team Members>

**Table of Contents**

[**Revision History**](#_vqzfpkca7atq) **2**

[**1.0 Introduction**](#_qltqo1cy8iw1) **3**

[**1.2 Scope**](#_wmvt2ngdgf9j) **3**

[**2.0 Related Documents**](#_5p2mp07s7mtu) **3**

[**3.0 Design**](#_1q88ieyn5hfc) **3**

[**3.1 Class Diagram**](#_a8l7bx9idz2a) **3**

[**3.2 Database Interface (Schema)**](#_69sqt72e9rqe) **4**

[**3.2.1 Database setup information**](#_f4ge63yhmorc) **5**

[**3.2.2 How to utilize the methods**](#_9ldr4fjmzxyt) **6**

[**3.3 Major Use Case**](#_axpy8df2w1jd) **6**

[**3.3.1 Example Story**](#_73e9ha8q1ljl) **6**

[**3.4 Next Use Case**](#_wyn8m7vvoy7e) **7**

[**3.4.1 Next User Story**](#_1kajo7ix7gcw) **7**

[**4.0 Systems**](#_ol9uamhj4yuk) **8**

[**4.1. Architecture**](#_x15ith2bmk4r) **8**

[**5.0 Non Functional Requirements**](#_sqw06bahpxg) **9**

[**5.1 SSL certification?**](#_mqq25r9sb9j1) **9**

[**5.1 Performance Criteria?**](#_vdc9p3kbbt3w) **9**

### 

### 

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Author** | **Distributed to** | **Version** |
| Oct 19, 2022 | D DeMarco | class | template |
|  |  |  |  |
|  |  |  |  |

*NOTE: For your sections for each sprint:*

*This document should be consistent with the SRS. You may have to go back and update SRS, or add more details in this document.*

*Add the detailed design document sections to the appropriate columns in the RTM.*

* *Review the requirements*
* *Add user stories for this sprint*
  + *Identify pre-conditions*
  + *Identify post conditions*
  + ***List design steps - add specific details that need to be remembered***
  + *Any constraints or assumptions*
* *Add sequence diagram*

# **1.0 Introduction**

*<updated from SRS>*

# **1.2 Scope**

This document includes the detailed design in the form of user stories and sequence diagrams for the *<application name>*.

The actors include:

*<name of each actor and brief description of their role in application>*

# **2.0 Related Documents**

*<links to all related Documents>*

*RTM, SRS, Test Plan, diagrams in this document so you can easily update them*

# **3.0 Design**

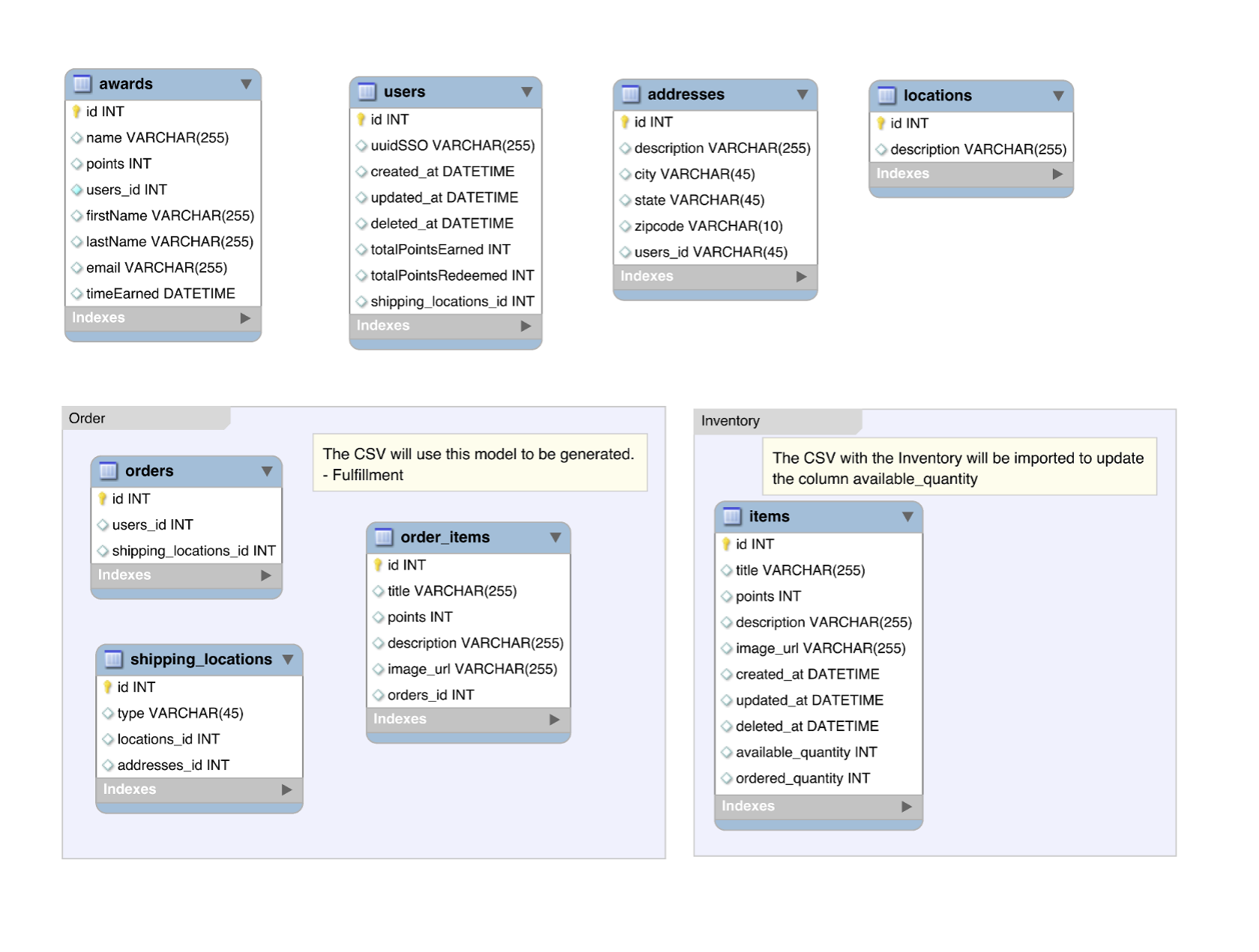
## 3.1 Context Diagram

*<Insert context diagram>*

## 3.2 Database Interface (Schema)

*<list table and exact field names and type****; include schema****. Schema can be* [*auto generated by MySQL Workbench*](https://youtu.be/RbKEYDtkAJI?t=30)*. This section will be added/* ***updated every sprint as you add functionality****>*

Example schema:



### 3.2.1 Database setup information

*<This section explains how you create and setup the database>*

*(The information below is if you are using AWS, otherwise you can delete if N/A):*

If the database is going to run on AWS, below is a starting point for creating a database on AWS. this should be reviewed & modified as it applies to your project

3.2.1.1: Follow these tutorials

[Getting Started with AWS](https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/CHAP_SettingUp.html)

[Creating a MySQL DB Instance](https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/CHAP_GettingStarted.CreatingConnecting.MySQL.html)

[Create a Budget Alert](https://aws.amazon.com/aws-cost-management/aws-budgets/)

3.2.1.2: Make sure that you have enable access from All IP addresses in the security settings in AWS

3.2.1.3: Remember the username and password that you set during the creation of the db, it’ll be useful later

3.2.1.4: In MYSQL Workbench, for the hostname, enter the endpoint for the AWS (It’s listed as such in the control panel), then enter any other relevant information when making the connection.

3.5.1.5: Create, and populate tables as needed.

### 3.2.2 How to utilize the methods

< library / path / includes to access DB>

<methods to access DB >

Example:Be sure to download and include config.php, it is there to make sure that the database information is at minimum 1 file removed from whatever file is being used. From there, the function files will work as normal functions in your php code as long as you include the specific file that contains the calls.

## 3.3 Major Use Case

*< From the SRS,create a section for each use case (3.x) >*

*< add a sub- section for each user story (3.x.1…) >*

### 3.3.1 Example Story

**User Story:** As a general user of this website, I want to be able to view any upcoming events in a nice compact list so I can know exactly where to be and how to attend an event.

**Description:** Display the events for upcoming AITP events. These events will be displayed on a tab of the portal where users can log in to view. Visitors to the site will be able to see the upcoming events on the Events tab of Weebly.

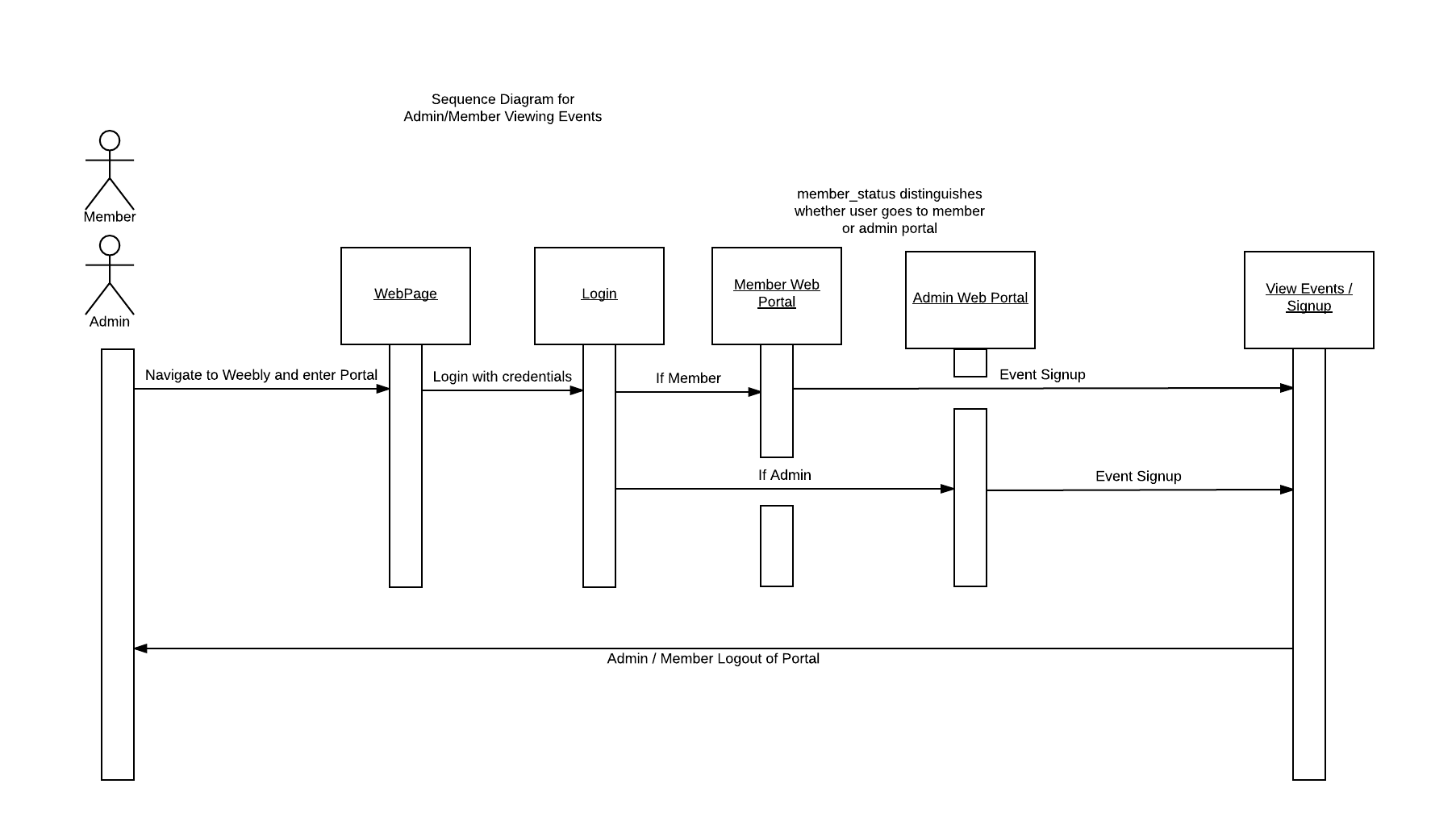
**Pre-Conditions**: none

**Post-Conditions:** Events are displayed

**List design steps - add specific details that need to be remembered**

**Constraints**: Members can see additional columns that visitors can’t view

**Assumptions:** none



## 3.4 Next Use Case

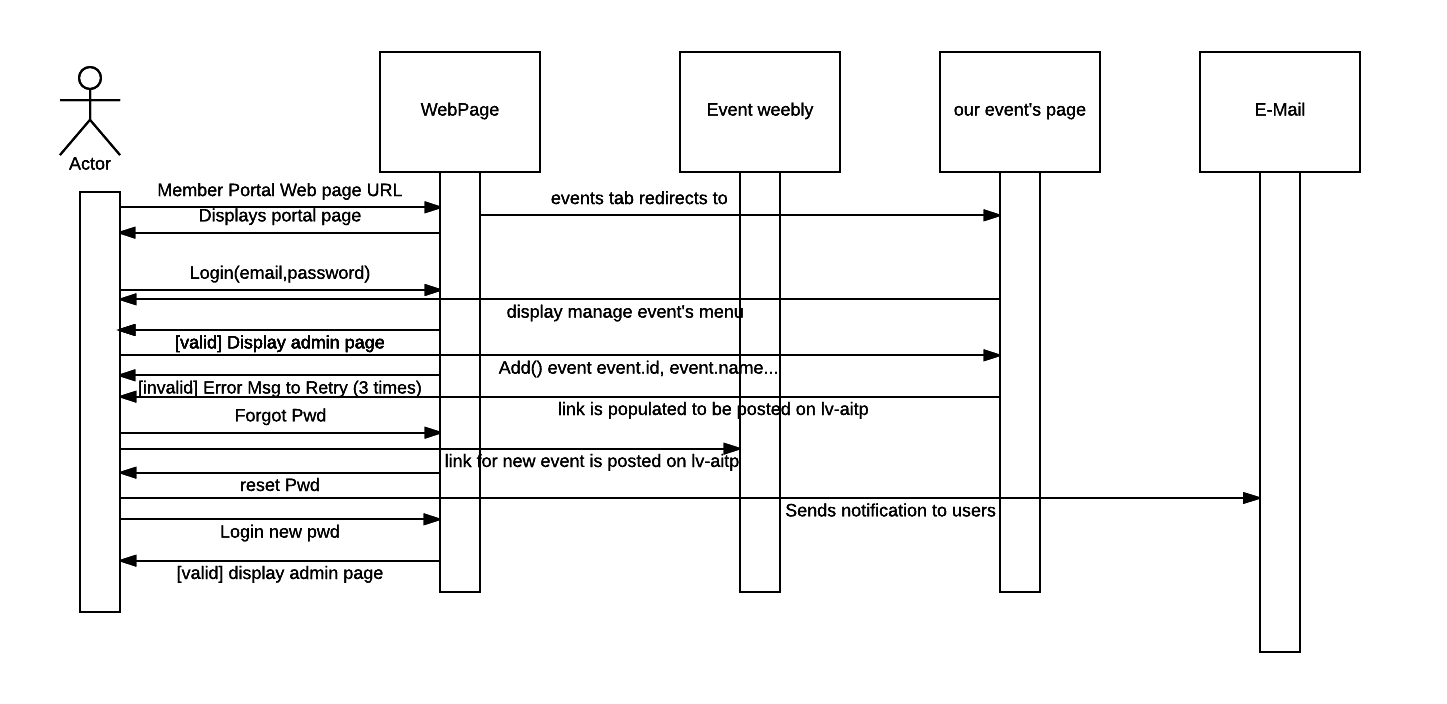
Next use case

### 3.4.1 Next User Story

**User Story:** As a general user of this website, I want to be able to view any upcoming events in a nice compact list so I can know exactly where to be and how to attend an event.

3.4.1.1 More detailed event

For larger use cases, you may need to break them into smaller use cases.

****

**Description:**

Admin will login in Weebly. He would hit event’s tab and be redirected to http://Eventeam.x10.mx/events.php Admin enters event. After event is added he sends email to users.

**Pre-Conditions**: none

**Post-Conditions:** emails sent

**Constraints**: Only admin can send emails

**Assumptions:** none

# **4.0 Systems**

## 4.1. Architecture

***<diagram of architecture>***

*<identify development, staging, and production environments>*

*<****scripts to deploy to staging & production****>*

*<description of overall architecture and what is running on each component>*

# **5.0 Non Functional Requirements**

Below is a list of non-functional requirements implemented.

*<Pull from SRS and add Specific details to prove project can be tested, measured, and complies>*

## 5.1 SSL certification?

*< specific cert; expiration; how / where to renew>*

## 5.1 Performance Criteria?

*< how many users? Load testing?>*