**TicketVault**

**Design Documents**

**Group**: TicketVault (Team 8)

**Members:** Hanxun Huang, Kai Sun, Chengkang Xu, Yijie Wu, Longding Zhang

**Table of Contents**

**Purpose**

**General Priorities**

**Design Outline**

**Design Issues**

**Design Details**

**1.Purpose**

Current Organization website, such as: BoilerLink.com does not provide tickets-reserving service for organizations and students. Our goal is to develop a platform (i.e. a website with well-organized UI) that provides a convenient way for students to reserve tickets, and organizations to promote their events.

**2.Design Outline**

Our system will be client-server model. The server will communicate with website UI(User) and the database.

**A ) Components**

***Server:*** The server will handle the requests sent from the Website UI (Clients), store or fetch necessary data from database. Such as: login and search function. The databases will reply the information back to the server and the server will repley to the user. Due to security reason, we will not let the server have direct admin access to the database.

***Database:*** Store the users’ and event information. Such as username, password, event date and ticket seating information. We will be using Mysql database.

***Website UI(client):*** For user to use send request to the server and see the result from the server. It will handle all the user’s request.

**b) Interactions**

**Server and Website UI**

The user uss button and text field in the website UI system. The UI system will send these information and request to the server. Such as　login and search. The server will get the reply from the database and send to the website UI. The website UI system will display these feedback in corresponding page.

**Server and Database**

The server will send the request from the user to the database with corresponding method such as login and signup. The database will recieve these information and send feedback to the server.

**c)UML diagram**

**3.Design Issues**

***(a) Ensure you spend enough time thinking about the design issues. Only one or two design issues will not be sufficient to get full credit.***

***(b) Each design issue requires descriptive title, solution options for the issue, and justification of your choice.***

***(c) You may divide this section into two subsections, Functional Issues and Non-Functional Issues.***

**Issues 1:** Login field or login page

**Option1:**Create a login field on the top of every page.

**Option2:**Create a single page for login.

**Decision:**Option 1

Option 1 is more fit for our design style, which is very simple. It would be waste of user’s time to link to another login page. We could create a base webpage with the login field and other page inherit from the base.

**Issues 2:**

**Option1:**

**Option2:**

**Decision:**

**4.Design Details**

***(a) Include class level design of the system (i.e. class diagrams) and be as detailed as you can.***

***(b) Describe the classes and interactions between the classes.***

***(c) Add sequence diagrams for different activities in the system, which will be helpful at the later stages of your project.***

***(d) If necessary, try to also include activity diagrams (or state diagrams) and UI mockups.***

**a)Class level:**

**Student user:**

Username: Set by the user.

Password:  Set by the user.

Email:  Set by the user.

Ticket booking information: It will contain the event name that the student has booked and seating information. (For example Eventname:Trip to Chicago Seating: group2 )

**Organization User:**

Username: Set by the person in the organization.

Password: Set by the person in the organization.

Email: Set by the person in the organization.

Organization name: We will store all the Purdue student organization name in the database.

Event information: Event name is same with the one in the Event class.

**Event:**

Event name: Set by the organization user who own this event.

Date: Set by the organization user who own this event.

Hosting organization name: Corresponding to the organization name in Organization user class

Seating information:  We will store some seating plan of Purdue classroom in the database. Organization user can also create on their own.

Booked user: A list of username corresponding to the one in student user class.

**b)Class interactions:**

**c)Sequence diagrams and activity diagrams**