

## **SW Engineering CSC648/848 Fall 2015**

### **EZ Restaurant**

#### **Team: 11**

Pooja Kanchan (email: pkanchan@mail.sfsu.edu)

Adrian Lee

Haichuan Duan

Sabreen Michael

Michael Lee

SeungKeun Kim

Milestone 2

October 23, 2015

#### **History Table**

Date	Revision Number	Comments
3 Oct 2015	Revision #1	Initial Draft
23 Oct 2015	Revision #2	Final Draft

## **Table of Contents**

<b>1. Use Cases.....</b>	<b>3</b>
<b>2. Data Glossary.....</b>	<b>5</b>
<b>3. Functional Specifications.....</b>	<b>9</b>
<b>4. Non-Functional Specifications.....</b>	<b>11</b>
<b>5. UI-Mockups and Storyboards.....</b>	<b>12</b>
<b>6. High Level Architecture and Database Organization.....</b>	<b>23</b>
<b>7. High Level UML Diagrams.....</b>	<b>27</b>
<b>8. Key Risks.....</b>	<b>28</b>
<b>9. Team Organization.....</b>	<b>29</b>

## **1. Use Cases**

### **1. User**

- a. Dave needs to make a reservation for the anniversary dinner for tomorrow night. He really wants it to be perfect but he doesn't have enough time and skills to do a good research. So he visited EZ Restaurant. On the home page, he sees a prominent search bar. He puts in his preferred city and his wife's favorite cuisine in the search bar. Scrolling down the search result list, he checks the photo, rating, and the short description of the restaurant. After he finds the restaurant he likes, he further checks the details by going to the restaurant page. On the restaurant page, Dave starts to look at the photos, video, menu and the event notices of the restaurant to make sure there are no unexpected activities during the dinner. After that, he goes to the reservation page. There he was given options to log in as a registered user, to register as a new user, or to continue reservation as a guest. Dave does not want to register at this point, so he just picks the date and time of his reservation and provides his name and contact number to make a reservation.
  
- b. Diana is a registered user of EZ Restaurant. She has used the website several times to make reservations but has never left reviews or ratings for the restaurants she has visited. Today she wants to use EZ Restaurant to make another reservation. She goes to the homepage, logs into her user account, finds the restaurant through the search bar, and makes a reservation. When she finishes the reservation, she decides to leave ratings and reviews for the restaurant she has dined in. She opens the ratings & reviews page. There she sees a list of the restaurant she has booked and visited through EZ Restaurant, with a 1-5 stars score bar and a link to add her review besides each restaurant. She leaves ratings for all the restaurants and leaves reviews for a few of them that she particularly likes.

### **2. Restaurant owner**

- a. Scott recently opened a restaurant. However, he doesn't have any technical skill to build up a webpage for his restaurant, so he decides to register his restaurant at EZ Restaurant. He goes to the home page, follows the Owner Registration form to register his restaurant. There he provides his restaurant's name, profile picture, description, location, menu, keywords, extra photos of the restaurant along with his own name and contact information. He decides he wants his restaurant to allow reservations, so he fills out how many reservations he allows per time slot, such as ten reservations per half an hour. Then he chooses a username and the password to make sure no other restaurant owners have access to the restaurant page once it's approved. After the submission, all he has to do is to wait for the administrator to review the registration form and approve.
  
- b. Sarah is a registered owner of a fine dining cafe on EZ Restaurant. This morning she does a routine check and update of her owner's account. She goes to the homepage, logs into her account. There she sees all information about her restaurant, including its profile, pictures, videos, upcoming events and reservations, approved hosts/hostesses and users' ratings and reviews. She adds a new host that she just hired to her approved hosts account so that the new host can log into his host view. Then she reads new reviews on her

restaurant. She finds no inappropriate reviews this time, but she knows that when that happens, she will be able to report it to the site administrator to moderate.

### **3. Website administrator**

Sally is the administrator of EZ Restaurant. She is a well-trained developer who has worked with the website for a few years. Today she does a routine check of the website. She follows the “Administrator” link to log in to the website, which directs her to the admin view. There on the new restaurant list she finds there are two new restaurants waiting for approval. She checks the contents of the registration form for each restaurant. After she finds no missing or inappropriate information, she approves them. From then on the two restaurants are added to the database and are searchable through the key words. Finally, she goes to the review section to check the new reviews for the restaurants. There she finds one restaurant owner has reported a user review that purportedly contains derogatory remarks. She reads through the review carefully and concludes that the review is against the appropriate policy of the website. She deletes the review so that it will not show up in the restaurant’s review section from then on.

### **4. Host/Hostess**

Sue is the hostess of a registered restaurant on EZ Restaurant and has an account created by the owner. After arriving for work today, she follows the “Host Log in” link to log in to the website, which directs her to the host view. There she has a view of a calendar that shows upcoming reservations for the restaurant. It contains information about customers’ names, party size, and contact information and reservation time. When a customer arrives, she clicks the list item, which will removes the customer from the list and also indicates that he/she has dined in at the restaurant. She can also add the call-in or walk-in reservations through the web page. When she has time, she can also read the reviews that customers left for their restaurant. If she finds any negative and abusive reviews, she can always report and ask the administrator to review it.

## 2. Data Glossary

**User:** can access restaurant information and make reservations. No login/registration required.

**Registered user:** retains user privileges and gains extra usability features in exchange for providing personal information. Must login/register a user account to access extra features.

**Registered user account:** account for registered users. Stores the user's username, real name, contact information, and a short list of visited restaurants. Accessing a listing allows posting of reviews/ratings that will be shown on the restaurant's page. The username, real name, and contact number will be stored in the user table in the database. The list of restaurants will be stored separate.

**Contact information:** all registered accounts for EZ Restaurant must have an email address associated with the account. Contact number optional. Shall be inputted during the registration form.

**Restaurant/owner:** client for our website, either the owner of a restaurant or a representative of the restaurant. Responsible for registering the restaurant, providing all relevant info, and being the primary contact for any issues with tech support. Must create an "owner" account and use account to access registered user comments and restaurant information changes.

**Owner account:** account for the owner/representative of a restaurant. Allows access to the restaurant's information. Changes to the data will be updates after the first submission and admin approval of the restaurant. Can also access all the other features of the website available to a registered user. Account will store the owner's username, real name, email, contact number, personal address, and the id of his restaurant in an owner's table in the database.

**Host/Hostess:** employee of a restaurant owner. Must register/login with an employee account linked to an owner account. Will use host view to manage current day's list of reservations.

**Website administrator/admin:** can access all data in the database. Responsible for approving new restaurant registration, responding to owner complaints against users, and modifying list of registered users and owners/restaurants. Requires a master account.

**Master account:** account that allows full access to all aspects of the website and the ability to modify data in the database and the website itself. Only given to admins. Shall be created during website creation. Subsequent accounts can then be made through the first. Shall contain a username, real name, and contact info.

**Reservation:** securing a specific time slot at a selected restaurant for a number of people. Can be made using the website or by calling into the restaurant and having the host manually entering into the list. Can only be canceled by calling into the hostess for manual removal or re entering information at website. Registered users have a pre-filled interface due to information having been stored. Requires a calendar date, a start time, name of the user,

number of people, and contact information. All information will then be stored in the reservation table appended with the user's username if the user is registered and the id of the restaurant.

**Extra usability features:** these include being able to make/change/cancel reservations without having to re enter information, post ratings/reviews/comments for the restaurants the registered user has visited, and gaining advance notice of upcoming special events from restaurants.

**Special events:** planned occasions restaurant's can add to their listings that can be advertised to registered users as promotions. Owners will have an option on the restaurant view to add an event to their display page. Requires a short description of their event, a time, date, and an optional display photo. All info is then stored in an events table appended with the restaurant's id. When a restaurant's page is clicked from the search results, any events associated with that restaurant's id shall be retrieved and displayed in the events notice.

**Reviews:** after a confirmed visit, registered users can write about their experience for public viewing. Subject to moderation from admin and responses by the restaurant. Reviews will be stored in a reviews table along with the user's user id, the restaurant's id, a 1-5 number rating, the date it was posted, and a flag field used to notify admin if a review requires moderation.

**Rating:** a 1-5 number rating entered by registered users during their review. An average aggregate of all posted ratings shall be displayed alongside search results as well as on the restaurant page itself.

**Host view:** employee screen that lists all the day's reservations as well as provisions for adding/modifying/deleting reservations. Shall also be allowed to confirm reservations when people arrive, storing their information if a username is associated with the reservation and allowing the user to post a review. If they do not show up, the user will not be flagged as having visited the restaurant.

**Owner/Restaurant view:** Owner's screen that allows owner to add or modify the restaurant's name, description, location, profile picture, photos, videos, reservation time slots, create employee accounts, as well as view and respond to registered user reviews.

**Reservation time slots:** the specific time intervals the owner wishes to allow reservations to his restaurant from the website. These shall be displayed to the user as the only choices to select from when making a reservation. Shall be stored in an operating hours table with specific times for each day of the week.

**Employee/Host account:** account made by the owner for host/hostesses to use. Shall be stored in a host table with the username of the account tagged with the restaurant id along with the host's name and optionally their contact information.

**Restaurant:** Establishment the website stores reservations for. Must be registered by the restaurant's owner or a confirmed representative. During the owner's account registration,

places to fill out the restaurant's info shall be provided. A restaurant must contain the restaurant's name, description, address, categorical keywords (cuisine, etc), capacity for reservations, and contact info at the minimum. Optional multimedia content can be uploaded at the owner's prerogative. All info provided will be stored in a restaurant table along with a profile photo in a blob if applicable. All other multimedia content will be stored elsewhere. Upon entry into the table, an ID will automatically be created and appended to the restaurant for easier referencing.

**Photos/videos:** Optional photos or videos showcasing the restaurant the owner can upload from the owner view. Shall be stored in a blob in a separate multimedia table from the restaurant. Each media file shall be tagged with the restaurant's id and the type of media and retrieved when a user wishes to see more media on the restaurant page.

**Registered user view:** registered user screen that allows registered users to access their extra usability features as well as opt out of any options they do not like. Also contains the user's restaurant history for posting reviews and current reservations.

**Restaurant history:** a short listing of the registered user's previous confirmed visits. Each listing has a link to leave a review and a 1-5 score bar. Each restaurant is only listed once. This history is used to ensure registered user has actually visited the restaurant and allows the user to post a review of his experience. After the hostess confirms the reservation, the restaurant will be added to the user's history that will be checked when the user writes a review.

**Pull down menu:** a collapsible tab next to the search bar that will allow searching of categories in conjunction with the search bar. IE: if a user wants to dine Japanese, "Japanese" can be selected in the menu and a location or particular restaurant can be input into the search field to find Japanese restaurants matching either the name or location.

**Search bar:** a box that the user can type in the name and/or address of the restaurant in order to browse the list of restaurants in the database.

**Keywords:** details about the restaurant that will be used to categorize which restaurants are returned in a search. Examples include type of cuisine, a propensity toward a type of event (anniversary, birthday, concert, etc) and the like. Keywords will be stored in a category field in the restaurant table.

**Search result:** a list of restaurants in our database that closely match either the keywords or are nearby and offer similar categories as alternatives. Includes a short description from the owner and aggregated rating from registered users along with a default profile photo if uploaded.

**Restaurant page:** specific restaurant page where the information the owner inputted from his account page is displayed for the user. Includes the menu, photos and videos, a form to make a reservation, a visible "Make a Reservation" button, reviews from registered users about

their experiences and navigation to the restaurant. If there are any upcoming events the restaurant wants to advertise, it will be displayed in the events notice.

**Events notice:** an optional bulletin a restaurant can add to advertise their restaurant's upcoming occasions i.e. live music appearance, comedy, Valentine's Day, holidays, etc. Retrieves all upcoming events associated with the restaurant from the events page.

**Username:** each account made on the website must have a unique name associated with it for identification purposes. User usernames will be stored in the user table, owner usernames in the owner table.

**Report:** a general form for the users, owners and his employees can use to bring the admin's attention to bugs, technical problems, or reviews that need moderation. Shall be sent to the admin's contact info.

### **3. Functional specifications**

#### **1. User**

##### **Priority 1:**

1. All users shall be able to search for restaurants on the home page by name, location and type of food in the free text field
2. All users shall be able to see a list of search results after the search is completed. Each list item includes the restaurant's thumbnail picture, name, location, rating and a button to make reservation. Clicking on the restaurant's name opens the details page, which displays the restaurant's photos and video, address, phone number, hours of operation, customers' ratings and reviews, restaurant's time availability and upcoming special events. There the user can click on a button to enter the reservation page, where the user can enter date, time, number of persons, and name, contact information to make a reservation.
3. All users shall be able to register on the website by following a link and entering minimal required personal information, including name, phone number (optional) and email address.
6. Registered users shall be able to log into their accounts from the home page to make, change and cancel reservations. The reservation procedure for registered users is the same with unregistered users except personal and contact information are not needed.
8. Registered users shall be able to change personal contact information, including phone number, email and home address if available.

##### **Priority 2:**

- 1a. All users shall be able to search for restaurants by category in the pull down menu.
4. All users shall be able to clearly read the website's privacy policy during registration, shown inside a textbox in the registration page. The text box requires user to scroll to the bottom in order to continue registration.
7. Registered users shall be able to post and modify ratings and reviews of restaurants that the registered user has dined in from the registered user view.

##### **Priority 3:**

5. All users shall be able to report any bugs to the administrator by following a link.
- 1c. All users shall be able to search for restaurants by special events.
- 1d. Users will get an email after registration.
- 1e. Users will be able to see Google map to view the restaurant location when they visit restaurant details page.

#### **2. Restaurant owners**

##### **Priority 1:**

9. Restaurant owners shall be able to register for service from EZ Restaurant following link from the home page. The required information for owner registration includes business name, address, phone number, business hours, restaurant category, and reservation time information.

10. Restaurant owners shall be able to upload additional information such as restaurant menus and video files.
12. Restaurant owners shall be able to edit the restaurant's information after registration.
11. Restaurant owners shall be able to specify table numbers for reservations allowed per specified time in his or her restaurant.
13. Restaurant owners shall be able to log into restaurant owner's view once registered, where the owner can add/remove hostesses/hosts who are approved to log into the hostess/host view.
14. Restaurant owners shall be able to add, edit and delete information of the special events of the restaurant such as date, time and description of the event from the restaurant owner's view.

**Priority 2:**

- 12a. The owner's view shall display ratings and reviews from customers
- 13a. Restaurant owners shall be able to report inappropriate reviews to site administrator in the owner's view.

**Priority 3:**

- 13c. Restaurant owners shall be able to respond to customer's reviews in the owner's view.
- 11a. Restaurant owners shall be able to specify the blackout days (the dates on which reservation is not possible)
- 5c. Restaurant owners shall be able to report any bugs to the administrator by following a link.

### **3. Host/Hostess**

**Priority 1:**

15. Hosts/hostesses shall be able to log into and out of the restaurant host's view.
16. Hosts/hostesses shall be able to see display of upcoming reservations, including customers' names, reservation times and special instructions from the host's view.
18. Hosts/hostesses shall be able to add walk-in customers to waiting list from the host's view.

**Priority 2:**

17. Hosts/hostesses shall be able to check in reserved customers to remove them from the upcoming reservations list, which also serves as indication the customer has dined in here.
19. Hosts/hostesses shall be able to see customers' ratings and reviews and report inappropriate reviews to site administrator.

**Priority 3:**

- 5d. All hosts/hostesses shall be able to report any bugs to the administrator by following a link.

## **4. Site Administrator**

### **Priority1:**

20. Site administrators shall be able to log into the site administrator's view.
22. Site administrators shall be able to see list of registered restaurants and users from the database and search through different fields.
24. Site administrators shall be able to modify and delete information in the database, such as users, restaurants and reviews, which requires the master key.

### **Priority 2:**

21. Site administrator shall be able to see pending restaurant registration requests in the administrator view, check the information and files provided during registration and approve or deny them.
23. Site administrators shall be able to see restaurant owners' and hosts' reports of inappropriate reviews and arbitrate.

### **Priority 3:**

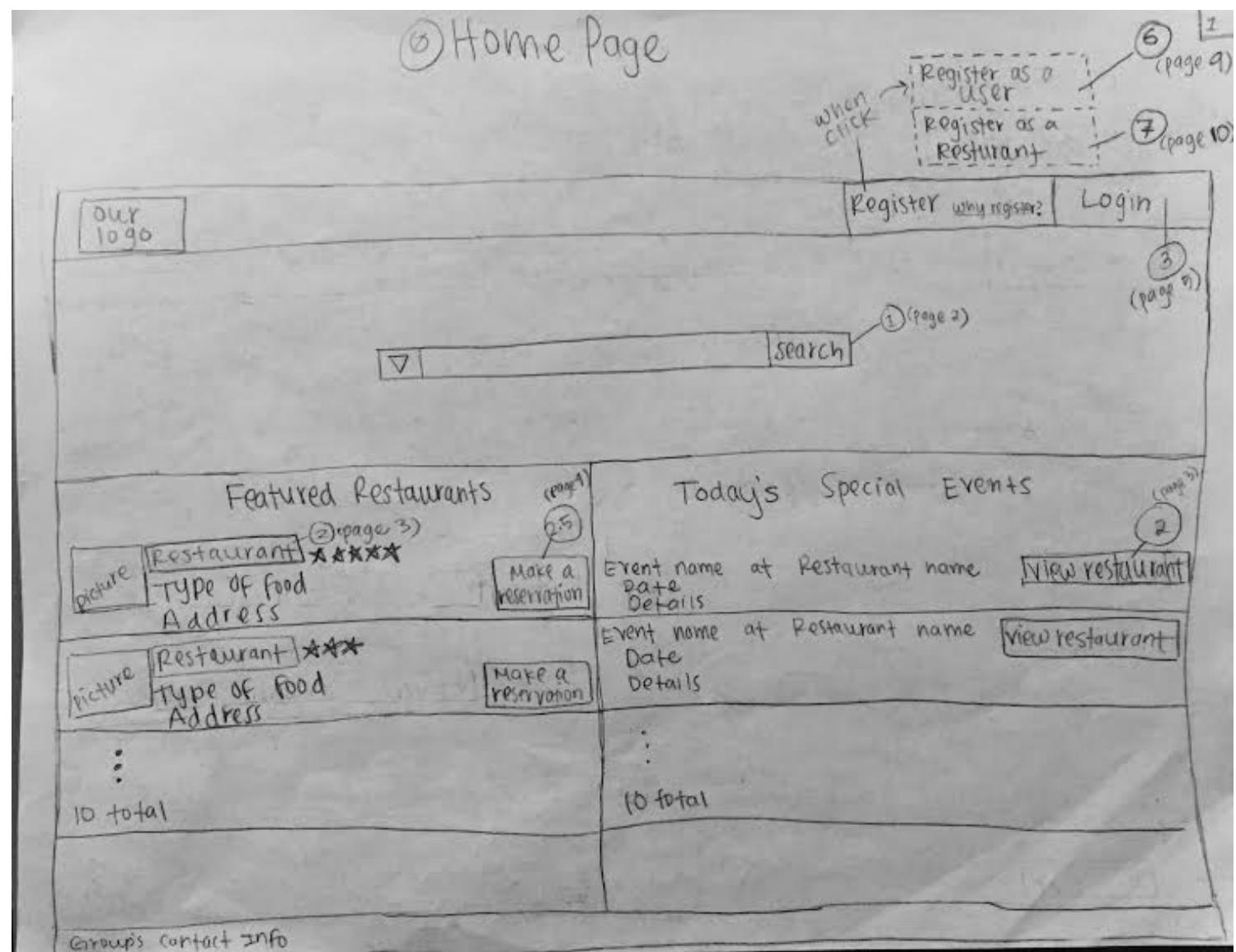
- 5e. Site administrator shall be able to view the reported bugs by following a link.

## **4. Non-Functional Specifications**

1. The website's pages shall load fast (< 5 seconds) and be responsive with high-speed internet connection common in US households;
2. Any personal/sensitive information shall be displayed and can be edited in https pages;
3. Application shall be developed using class provided LAMP stack
4. Application shall be developed using pre-approved set of SW development and collaborative tools provided in the class.
5. Application shall be hosted and deployed on Amazon Web Services as specified in the class.
6. Application be viewable in a standard desktop/laptop/mobile browsers, and shall render correctly on the two latest versions of all major browsers: Mozilla, Safari, Chrome and IE.
7. Data shall be stored in the database on the class server in the team's account.
8. Application shall be served from the team's account.
9. No more than 50 concurrent users shall be accessing the application at any time.
10. Privacy of users shall be protected and all privacy policies will be appropriately communicated to the users.
11. The language used shall be English.
12. Application shall be very easy to use and intuitive. No prior training shall be required to use the website.
13. Google analytics shall be added for major site functions.
14. The website shall prominently display the following text on all pages "SFSU/FAU/Fulda Software Engineering Project, Fall 2015. For Demonstration Only".
15. Modern SE processes and practices must be used as specified in the class, including collaborative and continuous SW development, using the tools approved by instructors.

## 5. UI Mockups and Storyboards

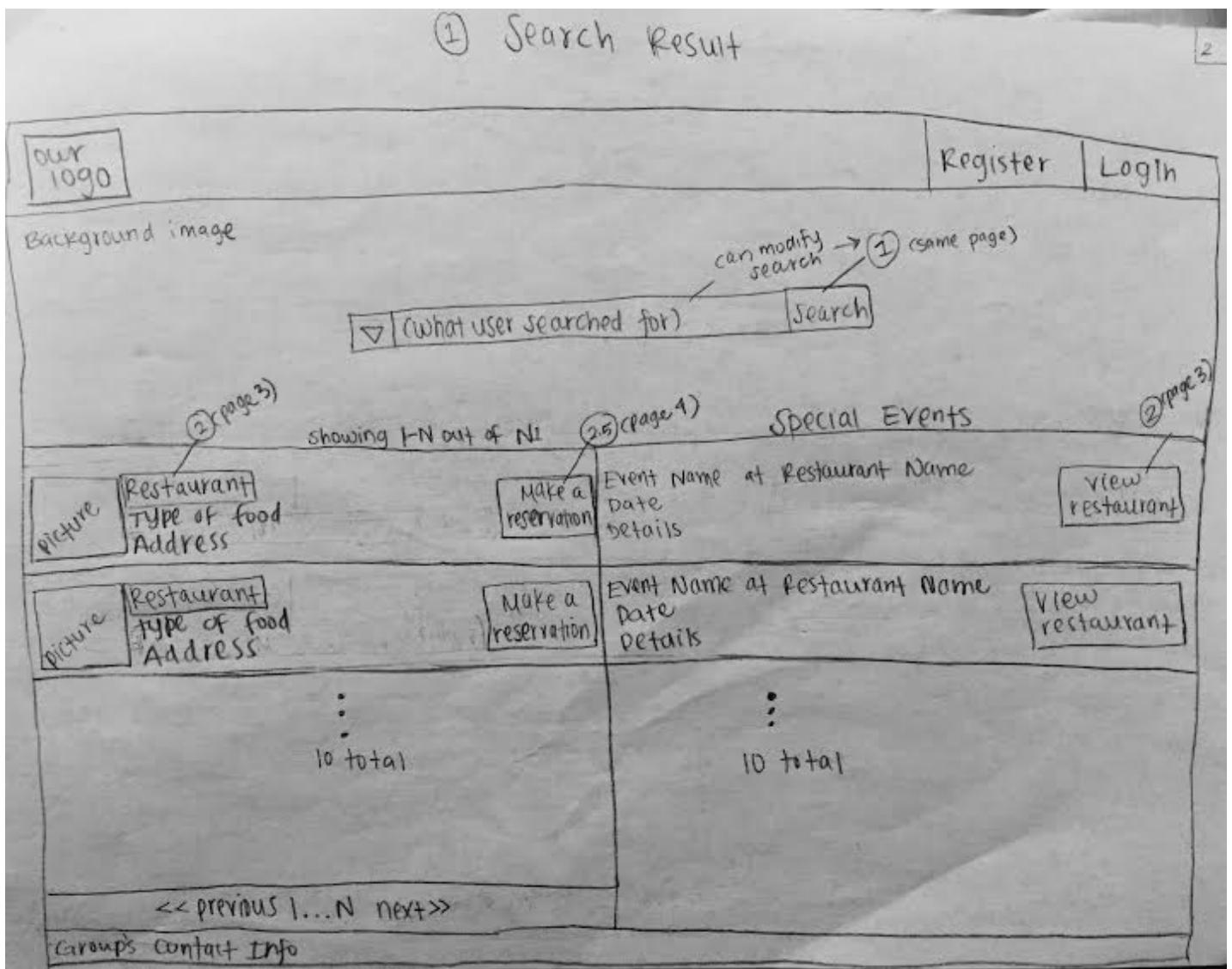
### 1. Home Page



#### Storyboard:

- When clicked on 1: Redirect to page 2
- When clicked on 2: Redirect to page 3
- When clicked on 2.5: Redirect to page 4
- When clicked on 3: Redirect to page 5
- When hovered over Register: Display two options: Register as a user and register as a restaurant.
- When clicked on 6: Redirect to page 11 (User registration)
- When clicked on 7: Redirect to page 12 (Restaurant registration)

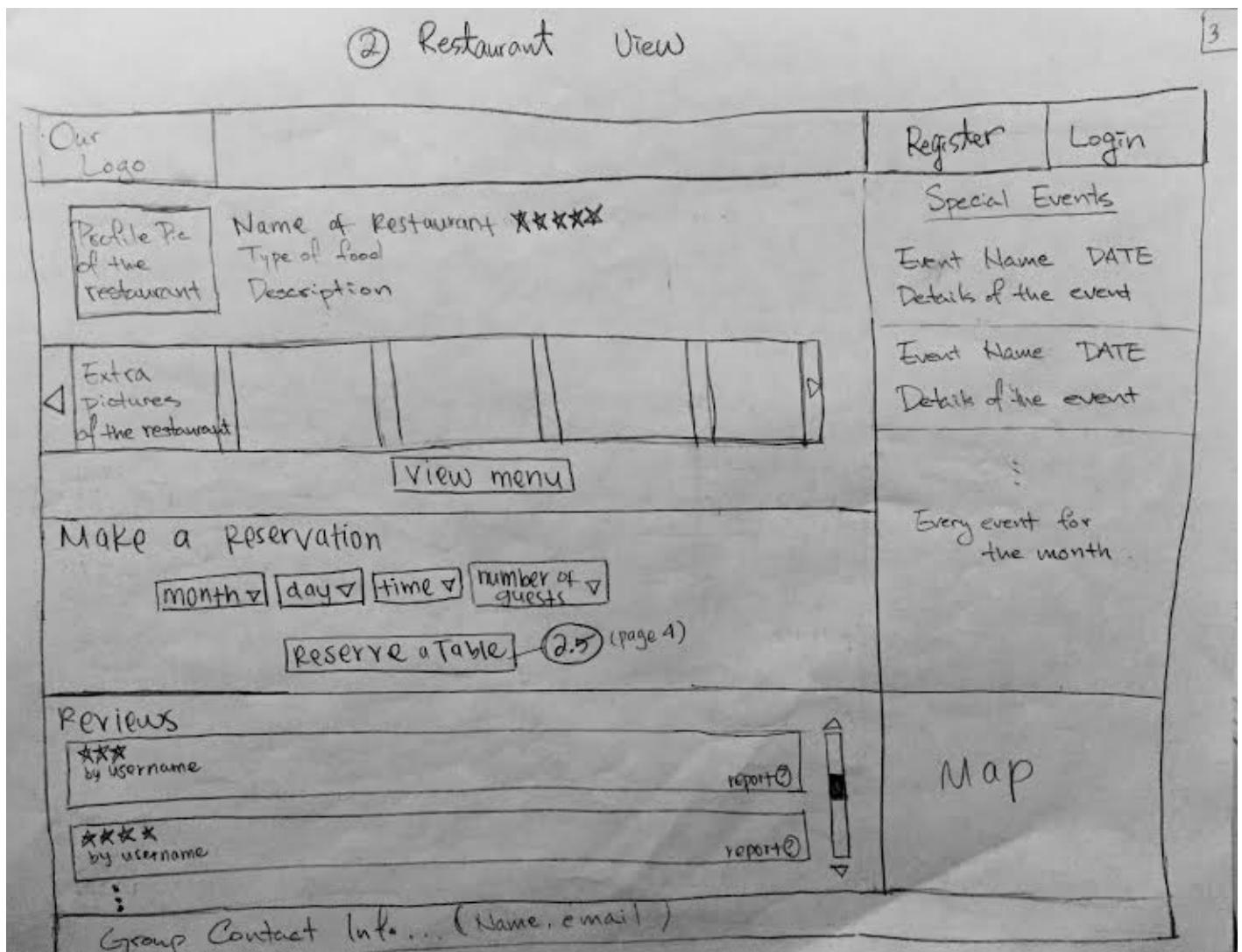
## 2. Search Results Page



### Storyboard:

- When clicked on 2: redirect to page 3
- When clicked on 2.5: Redirect to page 4

### 3. Restaurant View Page



#### Storyboard:

When clicked on 2.5: Redirect to page 4

When clicked on menu: Menu is opened as a pop up.

#### 4. Reservation Details Page

2.5 Reservation  
(this will be a pop up)

[4]

our logo | Register | Login

Number of people ▾

Month ▾ Day ▾ Year ▾ Time ▾

First Name

Last Name

Email

Phone Number

Accommodations

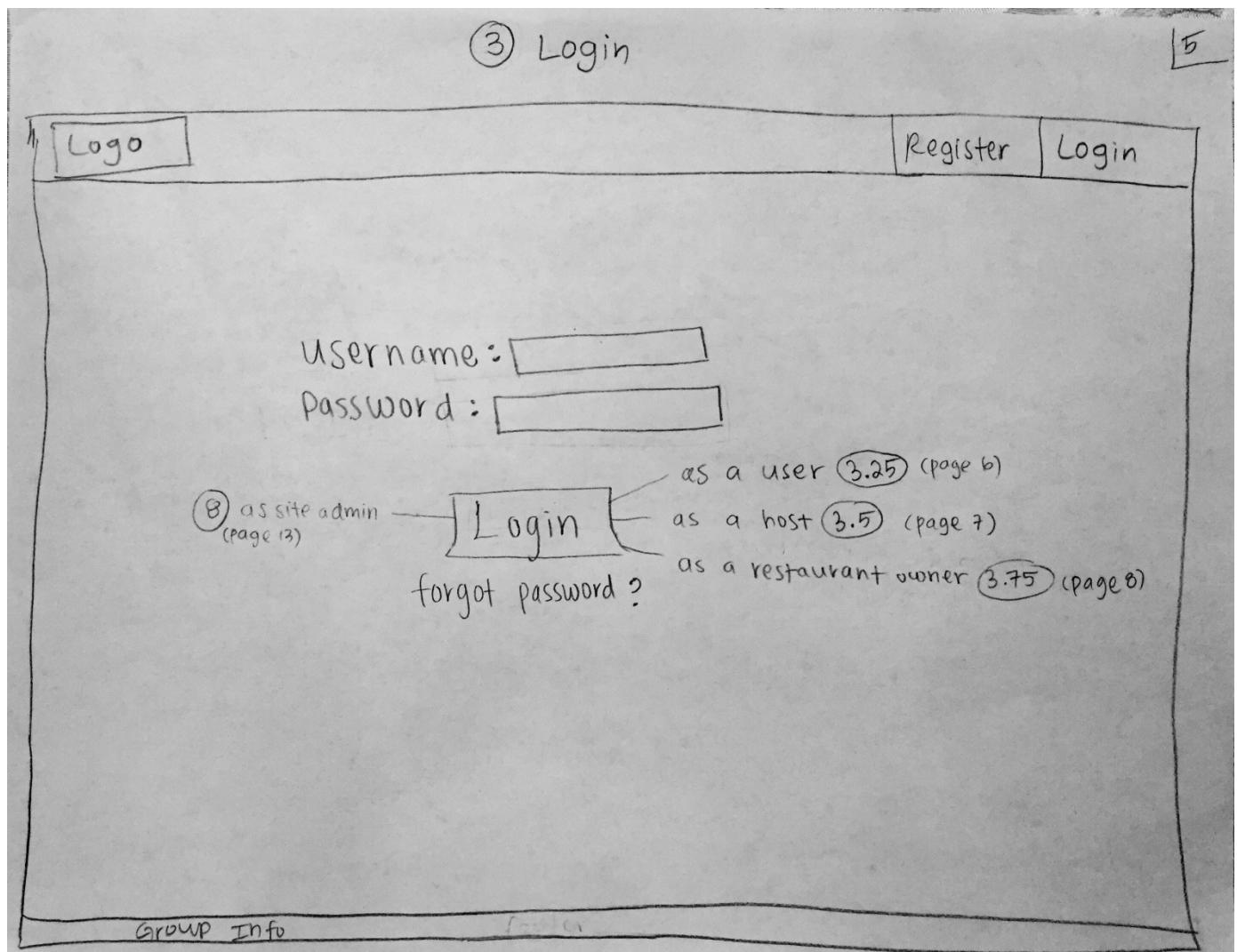
Groups contact Info

close | make reservation

#### Storyboard:

When clicked on make reservation: Availability of the restaurant is checked and reservation of table is made if the restaurant is available. If not, error message is displayed on the bottom.

## 5. Login Page



### Storyboard:

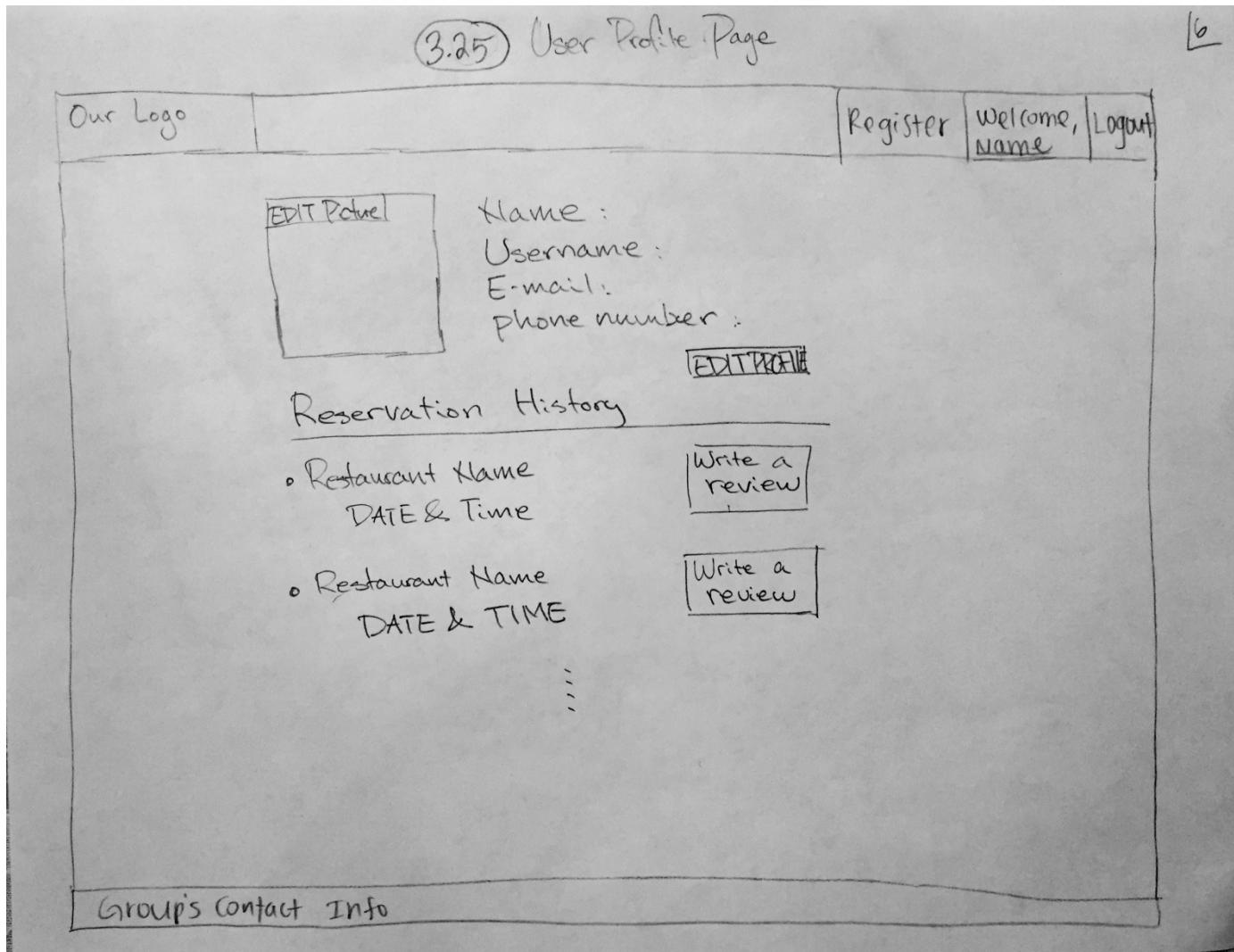
When clicked on 3.25 with user login credentials: Redirect to page 6, if the login is successful, otherwise error message is displayed.

When clicked on 3.5 with host credentials: Redirect to page 7, if the login is successful, otherwise error message is displayed.

When clicked on 3.75 with restaurant owner credentials: Redirect to page 8, if the login is successful, otherwise error message is displayed.

When clicked on 8 with site administrator credentials: Redirect to page 13, if the login is successful, otherwise error message is displayed.

## 6. User Profile Page



### Storyboard:

When clicked on “Edit Profile”: User can edit his profile information

When clicked on “Write a review”: Allows user to give rating and write a review about the restaurant.

## 7. Restaurant Host View

(3.5) Restaurant Host view [7]

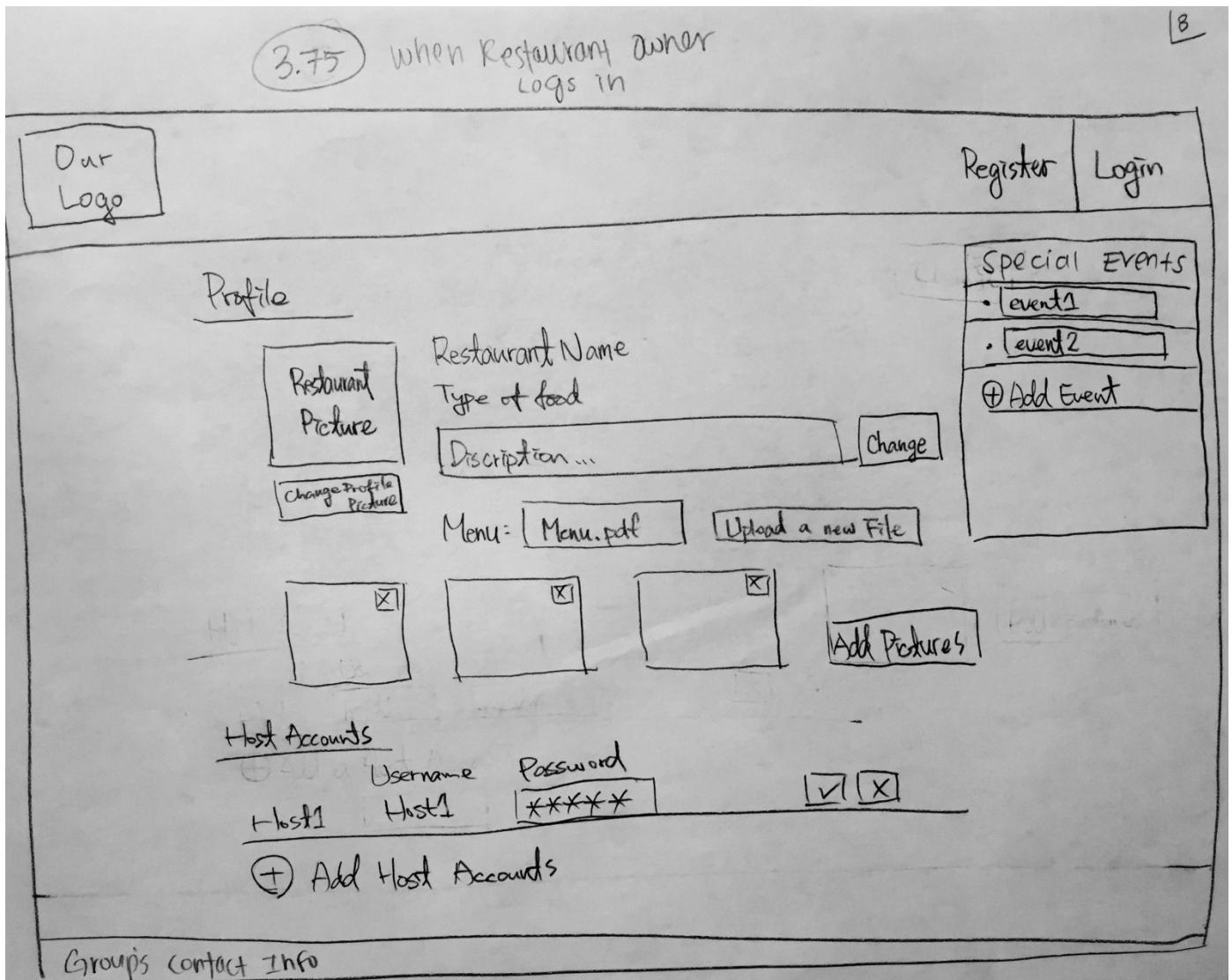
The diagram shows a wireframe of a web application. At the top left is a placeholder for 'Our Logo'. On the right are 'Register' and 'Login' buttons. Below the header is a section titled 'Reservations' with a 'Today' button. A 'Make a reservation' button is located on the right. A vertical scroll bar is on the right side. The main content area displays reservations for different times:

- 5:00 PM
  - John Smith 2 people Accommodations
  - Jenny Lin 4 people Accommodations
- 5:30 PM
  - Joyce 1 person Accommodations

Below the scroll bar is a 'confirm arrival' checkbox with a checked state. At the bottom left is a 'Groups Contact Info' section.

When clicked on 2.5: Redirect to page 4.

## 8. Restaurant Owner Home page



### Storyboard:

Add host account: allows a restaurant owner to add hostess

Add Event: Allows a restaurant owner to add events held at his/her restaurant.

## 9. User Registration

⑥ User Registration [9]

The form consists of several input fields and buttons:

- Top left: "our Logo" (with a small icon)
- Top right: "Register" and "Login" buttons
- Left side:
  - "First Name" input field
  - "Username" input field
  - "Password" input field
  - "Phone Number" input field
- Middle section:
  - "Last Name" input field
  - "Confirm Password" input field
  - "Email Address" input field
- Bottom left:
  - A checked checkbox next to the text "I agree to the Privacy Policy.  
If you do not agree please cancel." followed by a "cancel registration" button.
  - "Reset Form" button
  - "Group's Contact Info" button
- Bottom right: "Register Me" button

When clicked on register: The entered information is validated. If the validation is successful, then the user is registered and success message is shown. Else, error message is displayed.

## 10. Restaurant Registration

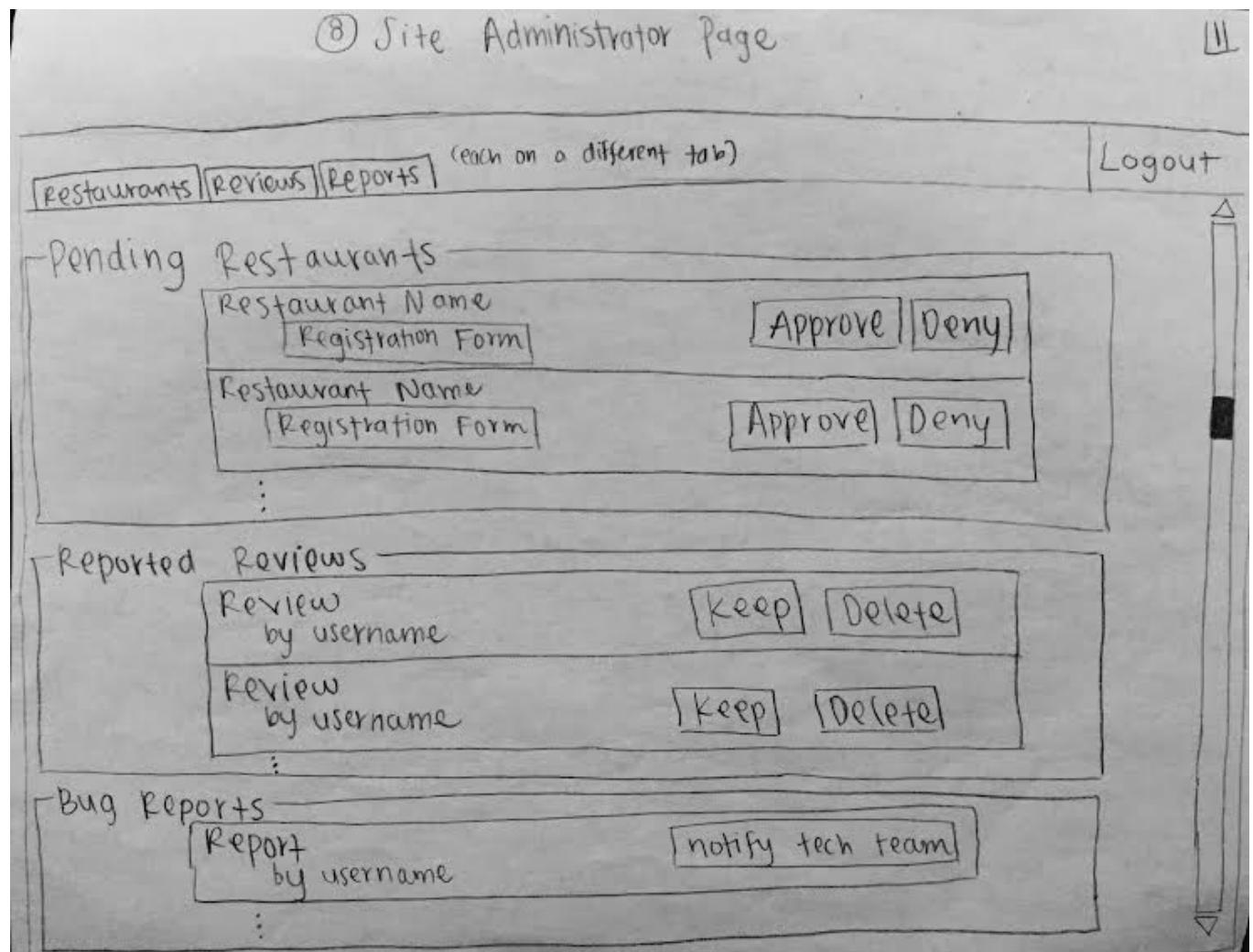
20

⑦ Restaurant Registration (owner)

<input type="text" value="Our Logo"/>	<input type="button" value="Register"/>	<input type="button" value="Login"/>
<b>Login Information</b>		
<input type="text" value="username"/>	<input type="text" value="confirm password"/>	
<input type="text" value="password"/>		
<b>Personal Information</b>		
<input type="text" value="First Name"/>	<input type="text" value="Last Name"/>	
<input type="text" value="Phone Number"/>	<input type="text" value="Email"/>	
<b>Restaurant Information</b>		
<input type="text" value="Restaurant Name"/>	<input type="text" value="Phone Number"/>	<input type="text" value="Address"/>
<input type="text" value="Description"/>	<input type="text" value="Type of Food"/>	<input type="text" value="Restaurant Capacity"/>
<b>Hours of Operation:</b>		
<input type="text" value="Number of People Every 1/2 hr"/>	<input type="text" value="Monday"/>	<input type="text" value="Tuesday"/>
<input type="text" value=""/>	<input type="text" value="Wednesday"/>	<input type="text" value="Thursday"/>
<input text"="" type="text" value="Friday"/>	<input type="text" value="Saturday"/>	
<input text"="" type="text" value="Sunday"/>		
<input checked="" type="checkbox"/> I agree to the <a href="#">Privacy Policy</a>		
<input type="button" value="Cancel"/> <input type="button" value="Register"/>		

When clicked on register: The information entered by restaurant owner is validated. If the validation is successful, the success message is shown. Else, error message is displayed.

## 11. Site Administrator Page



### Storyboard:

When clicked on Approve: The restaurant is approved and restaurant can be searched by the user.

When clicked on Deny: The restaurant is removed from the database.

When clicked on keep: The review will be retained.

When clicked on delete: The review will be deleted.

## 6. High level Architecture and Database Organization

In order to conform to industry standards, we will be hosting our website in the cloud.

### Architecture of the Project

Item	Details
Architecture	<p>Three-Tier Architecture using LAMP stack</p> <p><b>Server:</b></p> <p>Type: Virtual Ubuntu Server 14.01.1 LTS x64</p> <p>Location: <a href="http://sfsuswe.com">http://sfsuswe.com</a>.</p> <p>Running in the AWS (Amazon Web Services) cloud.</p> <p><b>Database:</b></p> <p>type: MySQL with version: 5.5.44-0ubuntu0.12.04.1-log</p> <p>Web Access: <a href="http://sfsuswe.com/phpmyadmin">http://sfsuswe.com/phpmyadmin</a></p> <p>Database Name: student_f15g11</p> <p><b>Web page:</b> <a href="http://sfsuswe.com/~f15g11">http://sfsuswe.com/~f15g11</a></p>
Supported Browsers	<p>Standard desktop/laptop/mobile browsers</p> <p><b>Windows 8.1 6.39600:</b></p> <p>Google Chrome v46.0.2490.71 and v46.0.2490.80</p> <p>Mozilla Firefox 41.0.1 and 41.0.2</p> <p><b>MAC OS 10.11.1:</b></p> <p>Safari: 8.0.8 and 9.0.1</p> <p>Google Chrome v46.0.2490.71 and v46.0.2490.80</p>
Version Control	EC2-hosted SVN repository
Tools	NetBeans 8.0.2 IDE with PHP and SVN Plugins
Frameworks	Twitter Bootstrap v3.3.5 jQuery v2.1.4
Languages	HTML5 CSS3 JavaScript 1.8 PHP 5.6
Analytics	Google Analytics

## **Classes and Components:**

1. **Restaurant:** This class stores the information about the registered restaurant such as name, description, address, and menu. It has a method getRestaurantInfo () method which returns the information of restaurant.
2. **Reservation:** This class holds reservation information such as date, time, number of people and restaurant info. The information of the reservation can be accessed by getReservationInfo method.
3. **Review:** This class holds review details of the restaurant such as ratings (out of 5 stars) and the actual review given by a registered user.
4. **Special Event:** This class stores special event information: description, date and time.
5. **User:** This class holds user information, when he makes a reservation like name and contact number. The method serachRestaurant provides a user to browse through the registered restaurants according to his preferences entered through search bar. The methods makeReservation and cancelReservation allow a user to make use of reservation facility. readReview method returns the reviews of the restaurants and viewSpecialEvent returns special events hosted by restaurant.
6. **Registered\_user:** This class is a subclass of user class and it holds information of registered user such as email ID, username and password. It has postReview method, which allows the user to post review of the restaurant he has dined in the past.
7. **Site\_administrator:** This class stores the username and password of site administrator and provides methods like approveRestaurant, viewRestaurants, and deleteRestaurant for handling restaurant specific actions and viewReview and arbitrateReview methods for handling review specific actions.
8. **Restaurant\_owner:** This class stores restaurant owner's information like name, contact number, email id, username and password. It has addSepcialEvent, modifySpecialEvent methods for handling special events of the restaurant. addHostess and deleteHostess to add and remove hostess respectively. Also, registerRestaurant method allows an owner to register his restaurant to EZ restaurant website and unregisterRestaurant allows him to unregister the restaurant. The owner can access review information and report the inappropriate reviews to site administrator through viewReviews and reportReviews methods respectively.
9. **Hostess:** This class holds hostess information such as name, contact number, username, and password and email id. A hostess can make a reservation using

makeReservation method. Similarly she can also view the reservations and cancel a reservation with viewReservation and cancelReservation methods. The method readReview allows to get the restaurant's reviews. She can also report a review she found inappropriate to site administrator with reportReview method.

### **Search Algorithm:**

Our search algorithm will contain a drop down menu and a search bar. The drop down menu allows user to search for a particular cuisine like Italian, Mexican etc. The user can also provide additional filters: name and city through search text box. The searching algorithm scans the database for the relevant information and the matches are displayed to the user. The algorithm works by taking the keywords the user provides in the text box and storing them into a string. That string is then compared to a combination of the restaurant name and address from the restaurant table. This way, the user can search "little Tokyo" to receive results for all restaurants named "little Tokyo", "san Francisco" to get all restaurants located in or have san Francisco in their name, or "little Tokyo san Francisco" to get the specific restaurant in a specific city.

### **Reservation algorithm:**

The registration page of restaurant allows the owner to enter number of reservations available per half an hour. The reservation algorithm checks the availability by comparing capacity of the restaurant available at the time slot entered by user. If the time slot is available then reservation is done and success message is displayed. Otherwise error message is displayed.

### **Order of displaying restaurants on home page:**

The restaurants are displayed according to their popularity. The reviews and ratings given by registered users are taken into account while calculating the popularity.

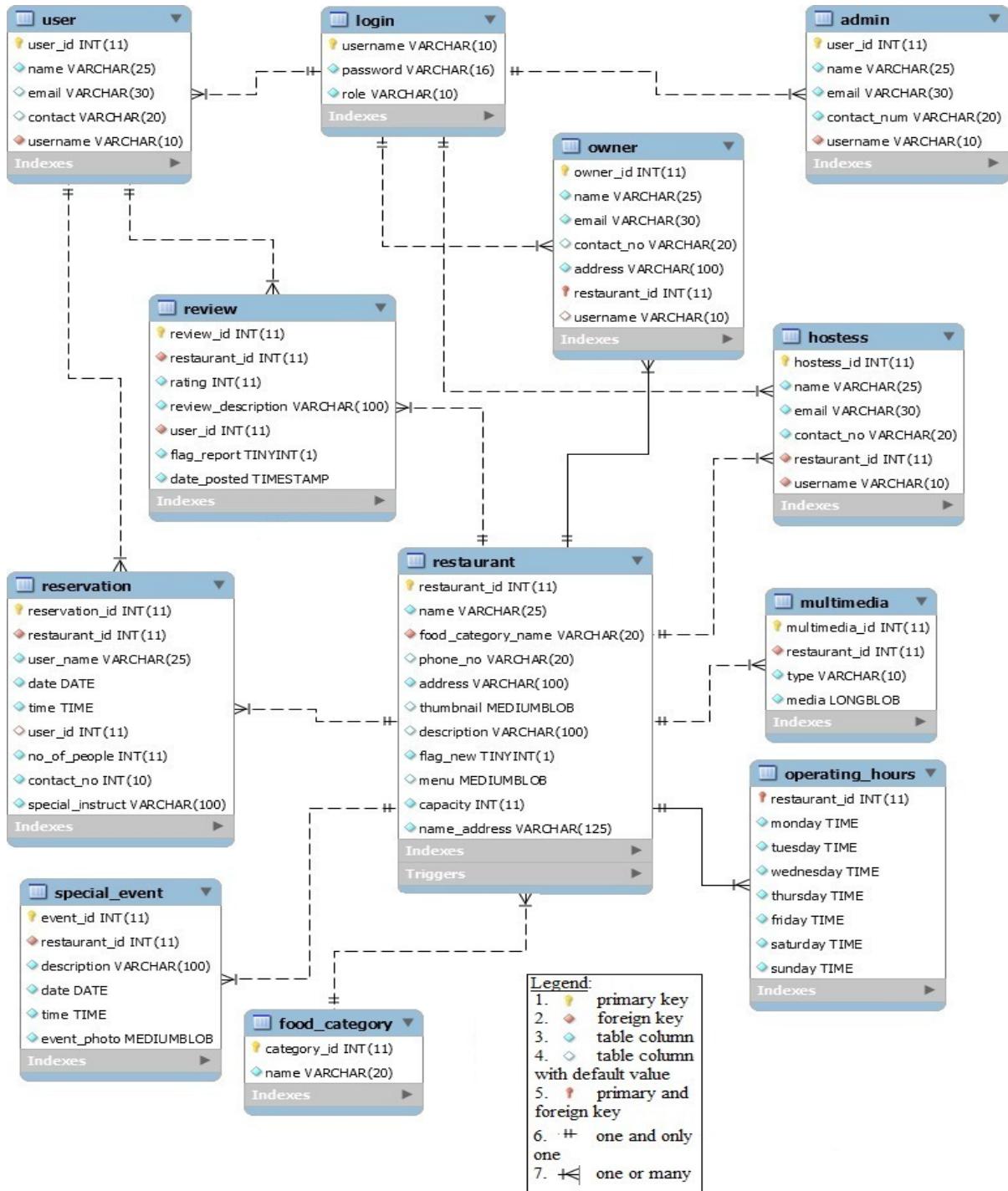
### **Order of special events displayed on home page:**

The upcoming events are displayed according to their dates. (Earliest event first).

### **Order of displaying reviews of a restaurant:**

Ratings are ordered by the rating with the highest ratings first. The reviews with the same ratings are ordered based on their posting dates.

## DB Schema

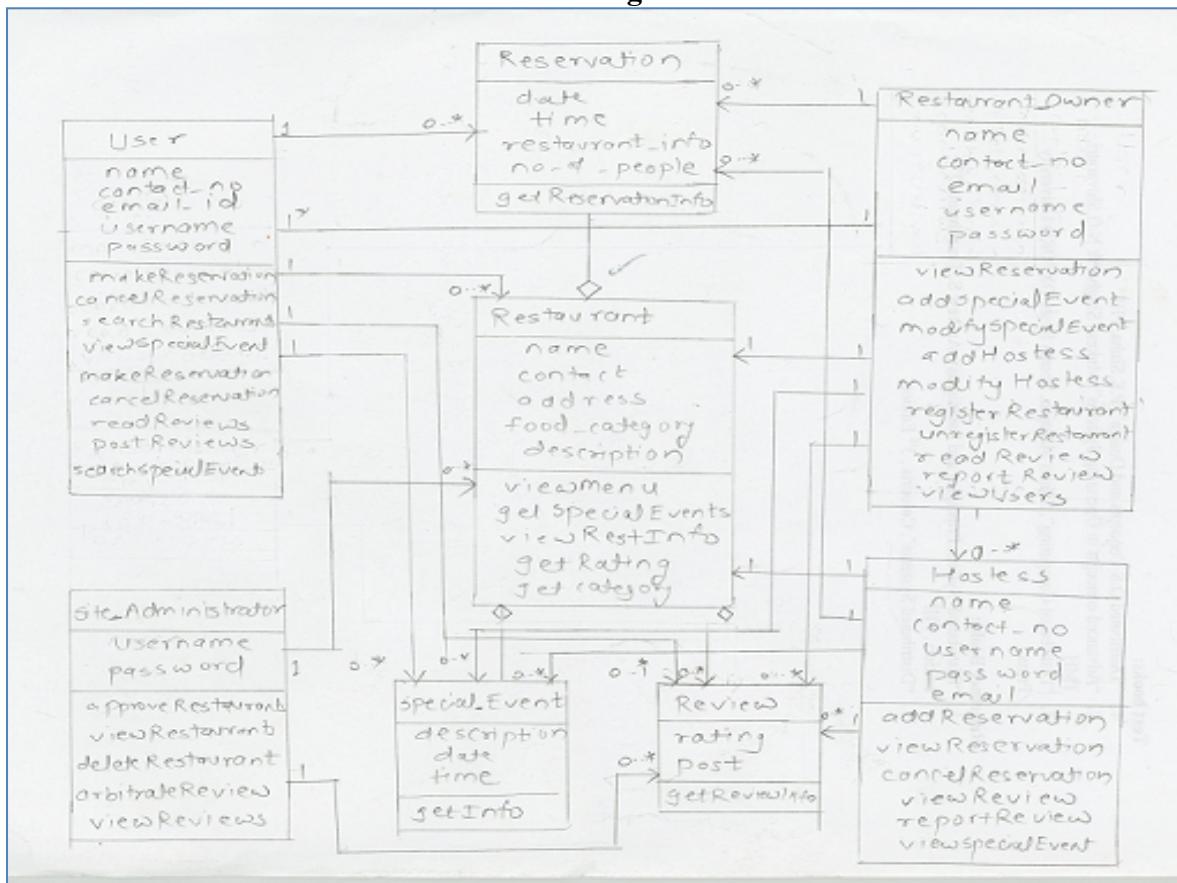


To store restaurant images and videos and menu, we are using MEDIUM BLOB data type MYSQL for following reasons:

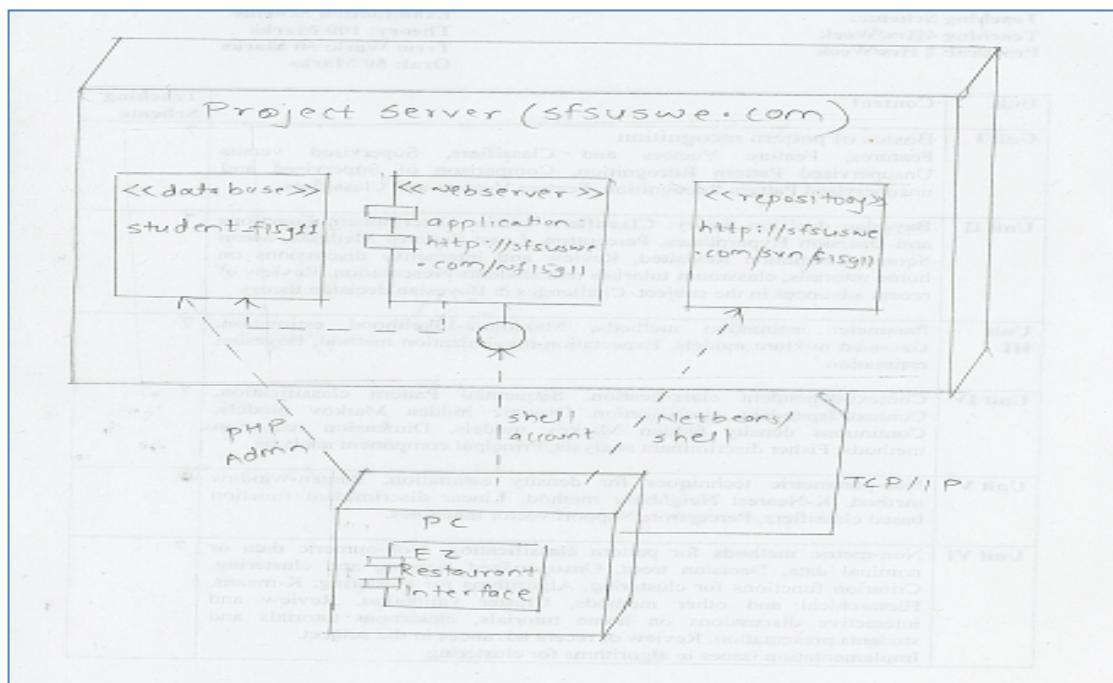
1. Easy to use and secure.
2. Medium Blob data type of MYSQL allows maximum size of image/video up to 16MB, which is sufficient for the website.
3. We are planning to limit the size of image a user can upload. Therefore, there won't be significant delay.

## 7. High Level UML Diagrams

**Class Diagram**



**Deployment Diagram**



## **8. Key Risks**

### **1. Skill Risks**

- Many of us have little to no experience with PHP.
- Many of us have little to no experience with databases.
- Many of us have little to no experience with HTML.

We plan to resolve these skill risks by learning as much as we can on our own. And by having a couple people be more focused on a certain skill, so that each skill will have a couple people that know it well. This will allow others to help each other and our group together will be knowledgeable of all the skills.

### **2. Schedule Risks**

- Many of us have a busy schedule and aren't able to meet as often as we would like.
- Some of us live far from campus.

We plan to resolve these schedule risks by having two different days where everyone can meet and by planning on having Google hangouts whenever we find them necessary.

### **3. Technical Risks**

- Some of us have Windows while others have Macs, so this can be an issue because we are using different software.
- We have a little experience with subversion.

We plan to resolve these issues by learning all the necessary technical components and by working together.

## **9. Team Organization**

Pooja Kanchan: Team Lead, SVN manager

Adrian Lee: Technical lead, Database manager

Sabreen Michael: Documentation lead

SeungKeun Kim: Front end lead

Haichuan Duan: Backend lead

Michael Lee: QA lead, Performance analyst