

SW Engineering CSC648/848 Fall 2015

TableMe

Team: 11 (Local Team)

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

Adrian Lee

Haichuan Duan

Sabreen Michael

Michael Lee

SeungKeun Kim

Milestone 4

December 11, 2015

Revision history

Revision number	Date	Comments
Revision 1	11 th Dec 2015	Initial Draft

Table of Contents

Product Summary.....	3
Usability Test Plan.....	4
QA Test Plan.....	6
Code Review.....	9

1. Product summary

Our product is a restaurant reservation website called TableMe. It allows users to search for restaurants by name, location, and type of food. Once a user searches, they see a list of results containing number of restaurants found, the restaurant's thumbnail picture, name of the restaurant, and a button that allows them to make a reservation at the certain restaurant. When clicking the name of the restaurant, the user can see the restaurant details page which includes the restaurants photos, address, phone number, hours of operation, customers' rating and reviews, Google map displaying location of restaurant, upcoming special events, and a button that allows them to make a reservation. When making a reservation, users will enter the date and time of the reservation, the number of people, and their name and contact information. A user can also register to TableMe by following a link and providing their personal information, email address, and creating a username and password. A registered user can login to their account in order to cancel their reservation(s), view reservation history, and to change their personal contact information. A registered user can also post ratings and reviews of restaurants they have dined in.

TableMe accommodates restaurant owners by allowing them to register for service by following a link to fill out their information on the restaurant registration form. In order to register, for restaurant owners as well as users, one must agree to the privacy policy which can be seen by clicking on the privacy policy link on the registration pages. Once a restaurant owner is registered, they can login to their account in order to edit their information, upload additional restaurant images, add/remove special events of the restaurant, and add/remove hostesses. TableMe accommodates the host/hostesses of a restaurant by allowing them to log in and out of the host's view which displays upcoming reservations including customers' names, their reservation time, and the button to cancel a reservation. The host/hostesses can also make a reservation for walk-in customers. To keep everything in order, TableMe will have a site administrator that can login to their account in order to see/remove registered restaurants and users from the database. The site administrator will also be able to see pending restaurant registration requests and check the information provided by the restaurant owner in order to approve or deny them.

TableMe is unique in that it allows restaurant owners to post special events that are taking place in their restaurant. The restaurant owner can enter the information of the special event by logging on to their account page. Users will be able to see these special events on the certain restaurant page. Information on a special event will include the event date, the event name, and details of the event. Another way TableMe is unique is that it allows registered users to post ratings and reviews of restaurants they have dined in. Users can access this by logging on to their account page. This allows all users to see the ratings and reviews of a restaurant by going to the restaurant details page.

TableMe can be accessed by going to <http://sfsuswe.com/~f15g11/>.

2. Usability Test Plan

Usability Test Plan on Search Function

Test objective:

To get user feedback on how easy and usable is the search restaurant function of our website.

To uncover any potential non-user-friendly defects on the search restaurant function, so that they can be remedied before final product release.

Test plan:

System setup & starting point: Laptop computers in Windows or Mac OS operating systems using latest Chrome, Firefox, Internet Explorer or Safari browsers. TableMe restaurant home page (<http://sfsuswe.com/~fl5g11/>) is opened in the browsers.

Users: 4~8 potential users of TableMe website. Users have basic computer operation and web browsing skills but are not computer system experts, ages 18~50.

Tasks to be accomplished: Users are asked to complete the following tasks on their own:

1. Find the restaurant named “The Crystal Chicken”
2. Find all restaurants in San Francisco
3. Find all Italian restaurants
4. Find all restaurants located in Florida that serve seafood

Completion criteria: users are able to correctly complete all searches within 2 minutes with no help

Evaluation and Questionnaires:

Users are asked to grade their search experience on these statements:

1. It was easy to find the restaurants that I need

☐ Strongly agree ☐ Agree ☐ Neither agree nor disagree ☐ Disagree ☐ Strongly disagree

2. The search result display was easy to understand and correct

☐ Strongly agree ☐ Agree ☐ Neither agree nor disagree ☐ Disagree ☐ Strongly disagree

3. Overall the search process was user-friendly and intuitive

☐ Strongly agree ☐ Agree ☐ Neither agree nor disagree ☐ Disagree ☐ Strongly disagree

4. Do you have any comments regarding the search process?

--

3. QA Test Plan

Test objectives: To test “search restaurant” functionality.

Setup:

Hardware setup:

ASUS Laptop

System type: Windows 8.1, 64- bit operating system, x64- based processor.

Processor: Quad core processor: Intel® Core™ i5-337U CPU @1.80 GHz

Software setup:

Chrome browser version 47.0.2526.73 m

Open sfsuswe.com/~f15g11/ in the browser.

Feature to be tested:

Search restaurant feature of our website allows its users to search registered restaurant by name, city or type of food. Search box is provided to enter search text. The results are displayed on the same page. The search results are paginated to limit number of results to be displayed on single page. and Five results are displayed per page. If the search results are more than five, first five results are displayed and navigation is provided at the bottom to see other search results. The search results include number of restaurants found, list of restaurants containing thumbnail, name, address, description and reservation button for each restaurant. Reservation button opens a popup to allow users to make reservation by entering minimal details. Users can also see details of each restaurant by clicking on thumbnail or name of the restaurant.

Actual test cases:

	Description	Test Input	Expected Output	PASS/ FAIL
1	To test search by name	Enter ‘little tokyo’ in search box and press ‘Search’ button	<p>The search page should display message’ Your search found 1 restaurant’ and display restaurant with the thumbnail and following information:</p> <p>Thumbnail of the restaurant</p> <p>Name: Little Tokyo</p> <p>Details: 1861 Taylor St, San Francisco,Ca-94133</p> <p>Upscale Japanese restaurant in San Francisco downtown with an extensive sushi menu and unique udon soup.</p> <p>Click on thumbnail and name to make sure that a page containing detailed information about the restaurant ‘little tokyo’ is displayed.</p>	PASS

2	To search by food category	Enter 'thai' in the search box and press 'Search button'	<p>The search page should display message' Your search found 2 restaurants' and display the restaurants with thumbnails and following information:</p> <p>1. Name: The bamboo Hook</p> <p>Details: 6427 Heritage Drive Phoenixville PA 19460</p> <p>Once you get hooked you'll never go!</p> <p>2. Name: The Jade Tree</p> <p>Details: 540 Heritage Drive Little Rock, AR 72209</p> <p>Sit in the shade of the legendary jade tree!</p> <p>Click on each name/thumbnail to see the details of the restaurant is displayed.</p>	PASS
3	To search by city	Enter 'Madison Kitchen' in the search box and press 'Search' button.	<p>The search page should display message' Your search found 1 restaurant' and display restaurant with the thumbnail and following information:</p> <p>Thumbnail of the restaurant</p> <p>Name: Madison Kitchen</p> <p>Details: 4370 Washington Blvd, St Louis,MO 63108</p> <p>Casual Midwestern dining.</p> <p>Click on thumbnail and name to make sure page contains detailed information about the restaurant.</p>	PASS
4	To search by name and city	Enter 'Peking Dragon san francisco' in search box and press 'Search' button.	<p>The search page should display message' Your search found 1 restaurant' and display the restaurants with thumbnails and following information:</p> <p>Name: Peking Dragon</p> <p>Details:718 Grand Ave, South San Francisco, CA 94080.</p> <p>Come in and enjoy: Delicious Sichuan specialties, tasty Dim sum made fresh daily in-house.</p> <p>Click on name/thumbnail of the restaurant to see the details of the restaurant is displayed.</p>	PASS

5	To search by food category and city	Enter 'italian Pacific Grove' in search box and press 'Search' button.	<p>The search page should display message 'Your search found 1 restaurant' and display the restaurants with thumbnails and following information:</p> <p>Name: Lombardi's Café</p> <p>Details: 414 19th St, Pacific Grove, CA 93950.</p> <p>Inspired by Italy's casual neighborhood, Lombardi's café features honest Italian food and a distractive wine list from Italian producers.</p> <p>Click on name/thumbnail of the restaurant to see the details of the restaurant is displayed.</p>	PASS
6	To search pagination	Enter 'CA' in search box and press 'Search' button.	The search results should display 'Your search found 14 restaurants' with 5 restaurants and navigation displaying three buttons 1, 2 and 3. Click on each button to make sure the restaurants are displayed and navigation works correctly.	PASS
7	To search some non-existing search restaurant as a negative test case	Enter 'Non-existing restaurant' CA' in search box and press 'Search' button.	The search should display 'Your search found 0 restaurants'	PASS

4. Code Review

Coding style:

1. PHP: camelCase case for methods, properties, functions and variables
2. Javascript: camelCase for variables and functions
3. HTML: lowercase for all html tags and attributes

 Close all elements
4. Indentation: 4 space indentation (as per NetBeans standards)

Email conversation and code review comments:

code for review
Inbox x

Haichuan Duan
Dec 6 (5 days ago)

to me

Hi Pooja,

The search restaurant code are in these two files.

Thanks,
Haichuan

2 Attachments

Restaurant_mod...

restaurant.php

pooja kanchan <pooj.kanchan@gmail.com>
Dec 7 (4 days ago) ☆ ↶ ▼

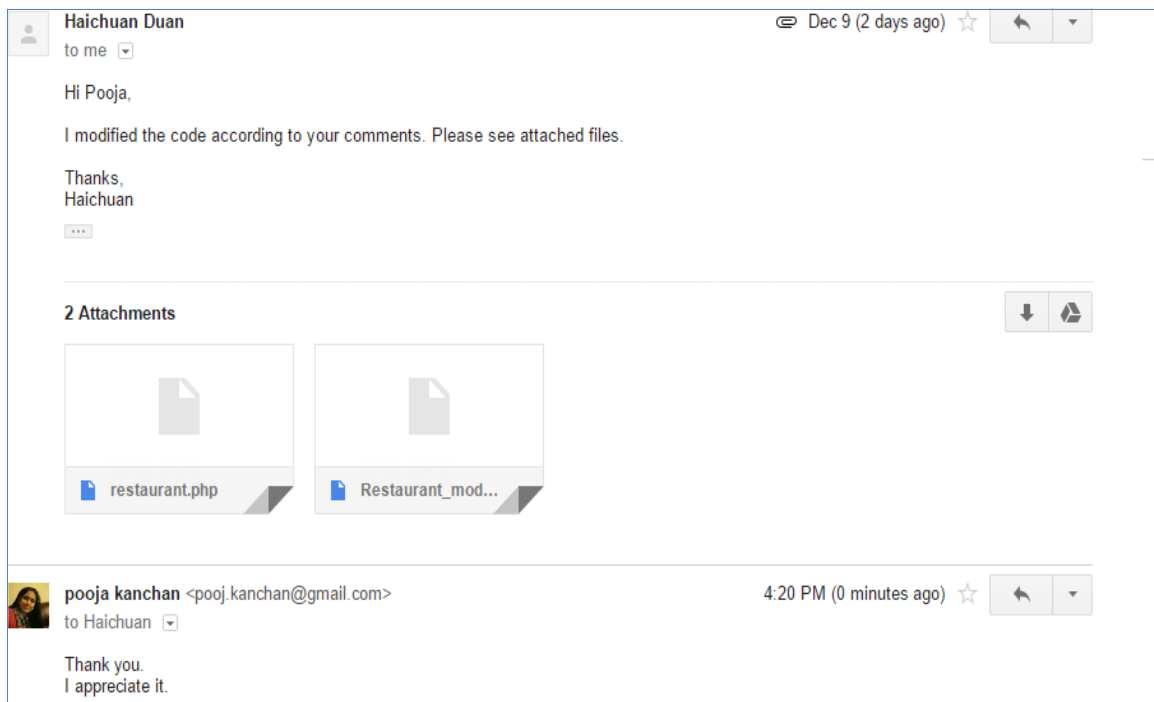
to Haichuan

Hi HaiChuan,

I reviewed the code, The code is organized well , modular and separates HTTP handling and database handling as per standards. But I have some comments as follows:

1. For some functions code comments are missing. Please add them.
2. For PHP variables, please use camelCase as per naming conventions.
3. In Restaurant_Controller.php, instead of accessing superglobal variable \$_POST directly, use `filter_input_array(INPUT_POST)`.

xxx



Sample code before change:

```
public function updateMultimedia($multimedia_id, $img, $restaurant_id) {
    if (intval($multimedia_id) > 0) {
        $sql = "UPDATE multimedia SET media=:media WHERE multimedia_id=:multimediaId";
        $stmt = $this->dbh->prepare($sql);
        $stmt->bindParam(':multimediaId', $multimedia_id, PDO::PARAM_INT);
        $stmt->bindParam(':media', $img, PDO::PARAM_LOB);
    }
    else {
        $sql = "INSERT INTO multimedia(restaurant_id, type, media) "
            . " VALUES(:restaurantId, 'image', :media)";
        $stmt = $this->dbh->prepare($sql);
        $stmt->bindParam(':restaurantId', $restaurant_id, PDO::PARAM_INT);
        $stmt->bindParam(':media', $img, PDO::PARAM_LOB);
    }

    if (!$stmt->execute()) {
        print_r($stmt->errorInfo());
        return false;
    }
    return true;
}
```

Sample code after change:

```
/*
 * If provided multimediaId > 0, update restaurant's image in multimedia table based on
 * restaurant id and multimedia id.
 * If multimediaId < 0, insert new image into multimedia table
 */
public function updateMultimedia($multimediaId, $img, $restaurantId) {
    if (intval($multimediaId) > 0) {
        $sql = "UPDATE multimedia SET media=:media WHERE multimedia_id=:multimediaId";
        $stmt = $this->dbh->prepare($sql);
        $stmt->bindParam(':multimediaId', $multimediaId, PDO::PARAM_INT);
        $stmt->bindParam(':media', $img, PDO::PARAM_LOB);
    }
```

```
}
else {
    $sql = "INSERT INTO multimedia(restaurant_id, type, media) "
        . " VALUES(:restaurantId, 'image', :media)";
    $stmt = $this->dbh->prepare($sql);
    $stmt->bindParam(':restaurantId', $restaurantId, PDO::PARAM_INT);
    $stmt->bindParam(':media', $img, PDO::PARAM_LOB);
}

if (!$stmt->execute()) {
    print_r($stmt->errorInfo());
    return false;
}
return true;
}
```