**COMP 3020**

**Human Computer Interaction 1**

**Milestone 3**

**High-Fidelity Prototyping**

**Group Number: 36**

**Group Members:**

**Kevin Kim /** [**kimh3427@myumanitoba.ca**](mailto:kimh3427@myumanitoba.ca)

**Stuart Campbell/** [**campb128@myumanitoba.ca**](mailto:campb128@myumanitoba.ca)

**Date of Submission: Nov 23, 2018**

Please note that we have split up as group 36 we are Group 36a

**Project Description**

* The system (Restaurant Search Application) was built by using HTML, CSS, JavaScript and jQuery. The system allows the users login or register a new account to the system. The user with the account will have an access to profile, which the user can check the user information, favourite restaurants, reviews written by the commonly used tags by the user. Also, the user can search the restaurant by typing in a keyword to the search bar.

**Register and Login functions**

* It is not necessary for the user to login to use the system. However, if the user has an account, the user can use that account to login and get an access to more features of the system (e.g. Profile, favourites, history, etc.). If the user does not have an account, it is no big deal. The user can create an account if not have one.

**Register (Horizontal High-Fidelity Prototype)**

* Register can be started by clicking the button ‘REGISTER’ on the top-right of the navigation bar. A window will pop up on the center of the screen and the main page will become shadowy to indicate that the user has to interact with the window just popped up. There are four sections to that needs to be completed in order to create an account, which includes username, email, password and confirm password. The system will check if all sections are filled in before creating an account. If any section is left blank, the system will not register an account and prompt the user to fill in the empty section. When all sections are filled in, the system will check if the user has violated any of two conditions; duplicate username and disparity between password and confirm password. The user cannot register an account with the name that is already taken by other user. Also, the system will check if the inputs of password and confirm password sections are identical. If the user has not violated any restrictions, the system will print out the message (Welcome! ‘Username’, Your ID is ‘sample’, etc. ) and prompt the user to manually login with the account just created (We do not want the system to automatically login).
* Note: When registering, the system will only store username and password. the system does not store profiles. The system will only store username and password

**Login (Vertical High-Fidelity Prototype)**

* If the user has an account, the user can click the button ‘LOGIN’ on the top-right of the navigation bar. Similar to the register function, a new window will pop up and the background will become darker, too. The system will prompt the user to fill in two sections, which are username and password. The system will print out an error message if any section is empty or the user has typed in wrong username or password. If the user filled in the correct username and password, the system will greet the user (ex. “Welcome Kevin!”) and ‘LOGIN’ button will be updated to display ‘LOGOUT’ to indicate that the user has logged in to the system. Once the user has logged in, the user can access to the profile by clicking the button ‘PROFILE’ on the top right corner.
* Note: At the bottom-right of the login window, there are two hyperlinks for the user to use in the cases if the user needs to register or forgot the password of an account. These hyperlinks do not work.
* Note: There are four premade accounts to test the login function. The user may login using any of the account below. Also, the user may use register function to create a new account and use that account as well to login to the system.

|  |  |
| --- | --- |
| Username | Password |
| Kevin | pw1234 |
| Test1 | pw1111 |
| Test2 | pw2222 |
| Test3 | pw3333 |

**Profile (Horizontal High-Fidelity Prototype)**

* Profile is available for the user that has an account. The system will prevent the user to access to the profile if the user has not logged in to the system. Once the user has logged in, the user can access to the profile by clicking the button ‘PROFILE’ on the top right corner. A new window will pop up on the center of the screen. A picture and a brief information about the user will be displayed on the left of the window. On the middle of the window provides following buttons that the user can interact with; Information (displayed as Info), Favourites, Reviews and Tags.
* Information is set to default when a window is opened.
* Favourites will display the user’s favourite restaurants.
* Reviews will display the reviews that he user has written.
* Tags will display the commonly used tags by the user
* Note: The profile contains hardcoded data.

**Search  (Vertical High-Fidelity Prototype)**

* The system provides a search function, where the user can search restaurants based on the keyword provided on the search bar. The system will then display all of the restaurants with the corresponding tag, the search getting more and more precise the more tags you give it. In our prototype, if you give the search bar 3 or more correct tags it will return a small amount of restaurant items because your scope is narrow. If you give the search bar 2 or less correct tags, it will return a large amount of restaurant items because the scope of your search is quite large. The search organizes the restaurant items by descending rating. A ubiquitous scale of 1 to 5 stars. Though in our case the stars were bugged and we used a character “O” instead of a star.
* The tag system is quite robust as well. The tags ignore capitalization so “Grill” and “grill” are read the same. It does not count the same tag twice and can handle plurals, such that “bar” and “bars” are read the same. For a full list of working tags see HTML file around line 1500. Example tags that work for testing convenience are grill, wing, greek, pizza, cheap and expensive.

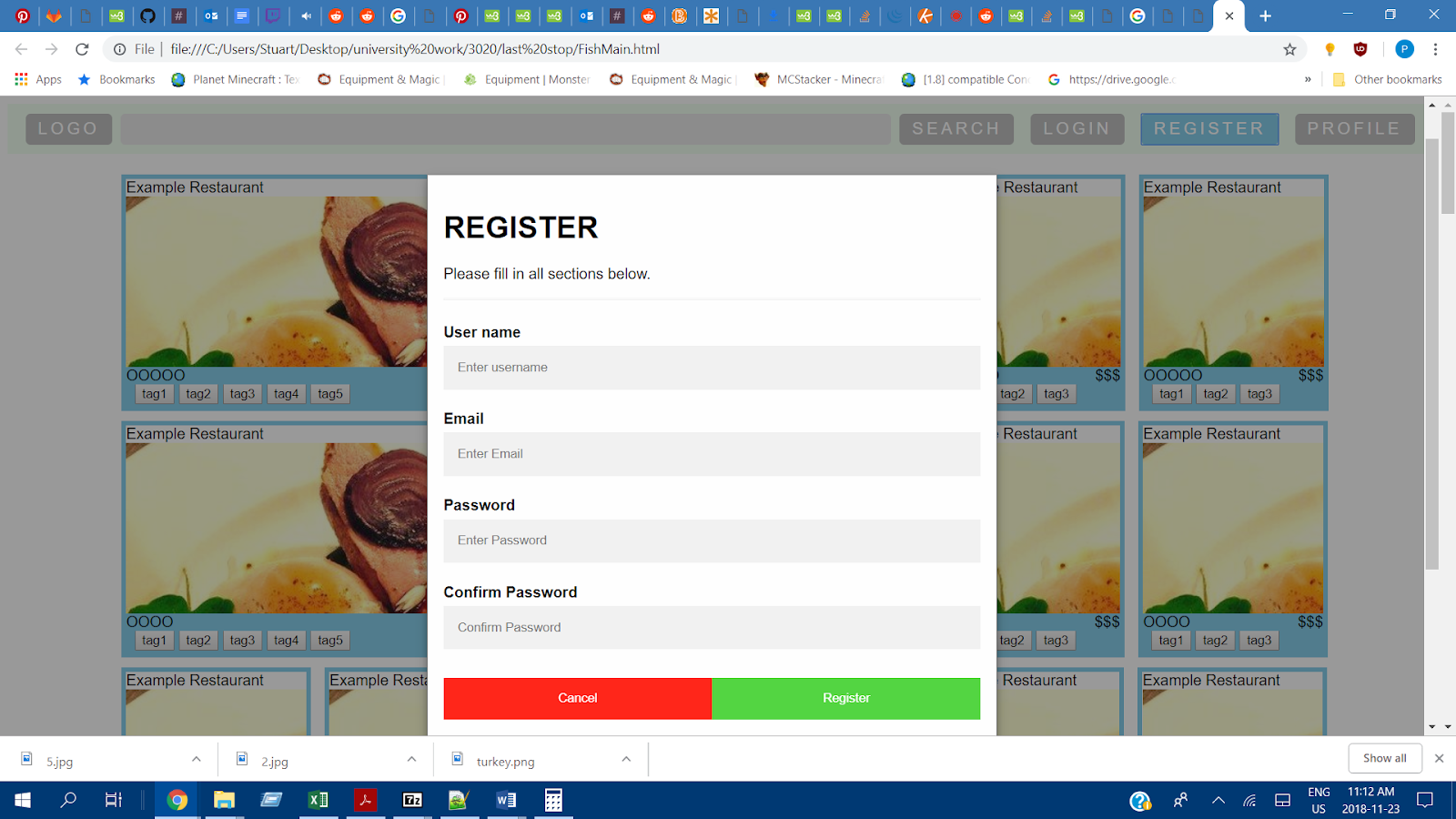
**Strengths**

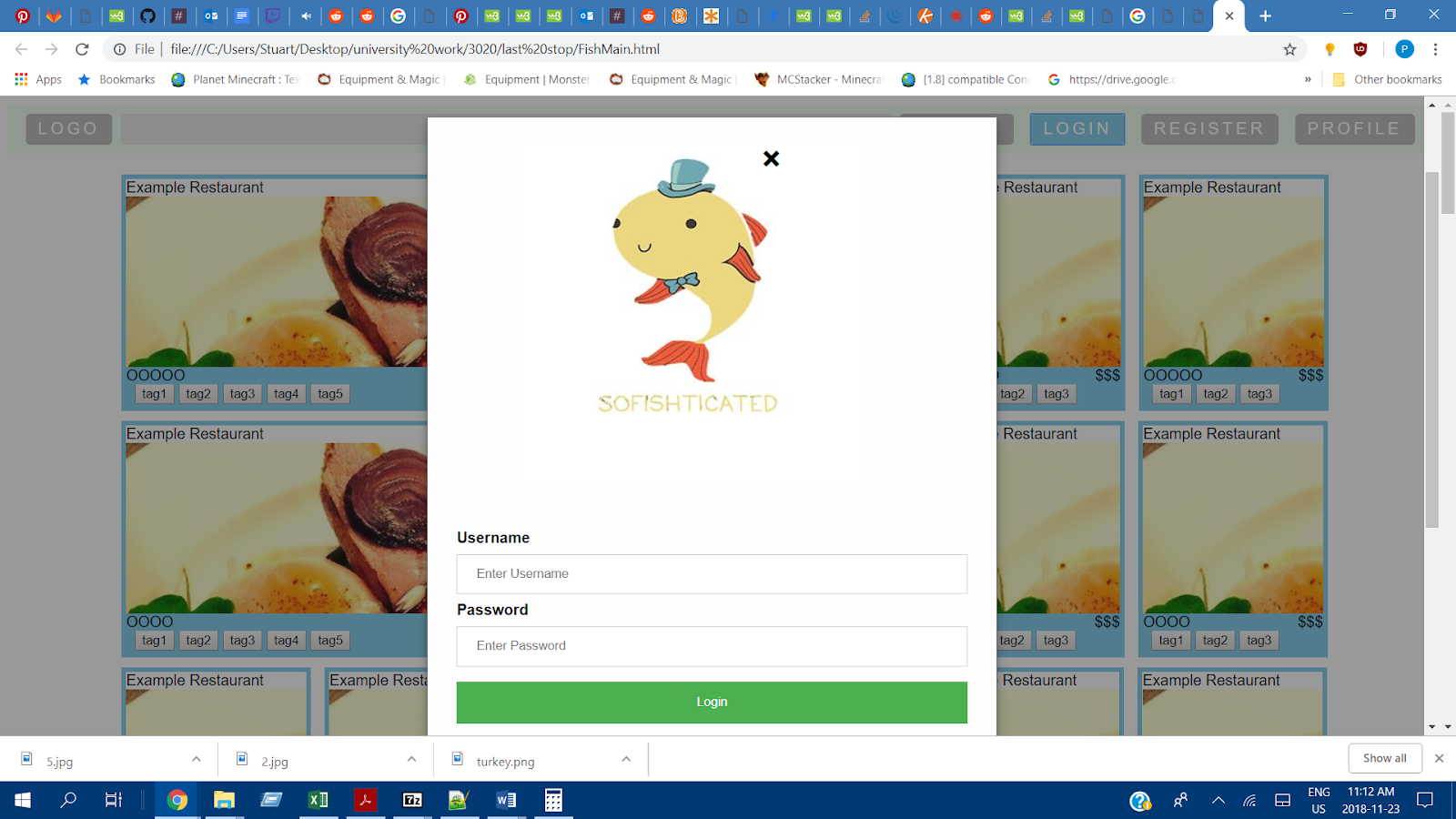
* The strength of the system is that the system is easy to use and learn. The major features can be accessed by simply clicking the buttons on the navigation bar (ex. Click ‘REGISTER’ button to register and Click ‘LOGIN’ button to login).
* The system also provides intuitive interface and interaction so that the user will only take few moments to identity what functions are available and understand what is happening in the system. (ex. The navigation bar on the top and
* The colour of the border of a box will turn into red when the user hovers a mouse on it. The promoted restaurant has larger size and different border colour.)
* The system provides safety to the user. The system will prevent the user from creating an error by displaying the message to the user wherever appropriate (ex. Registering duplicate username, password and confirm password are different). The user may undo the action that was unintentional (ex. Exit login and register window by clicking ‘Cancel’ button).
* An robust tag system the compensates for small discrepancies in the users typing.

**Particular problems**

* Resizing the window into certain range is fine. However, if the window size is very small, the positions of the elements become awkward.
* The system should allow the user to exit the register window by clicking cancel button or ‘X’ mark on top right corner. The problem is that ‘X’ mark is located outside the register window, though it functions properly. The same problem is present in login window. The user only can exit the window by clicking ‘Cancel’ button.
* When registering for a new account, the system is not able to recognize whether the user has filled in email section correctly (we replaced <form> tag to <div> and set the button into “type=button” to prevent the page to refresh). Also, there is no particular restriction on the format of the username and password (ex. Username should be n characters long. Password should contain at least one letter and number).
* When you search for tags, the first row of returned items is slightly off center.

**Screenshots of register and log**





**Appendix**

**References**

Third-party plugins

<https://www.w3schools.com/>

Modal Login Form

<https://www.w3schools.com/howto/howto_css_login_form.asp>

Modal Sign up Form

<https://www.w3schools.com/howto/howto_css_signup_form.asp>

jQuery

<https://jquery.com/download/>

Sofishticated Picture

<https://redbubble.com>

The basis of our header nav bar

<https://codepen.io/bllnsr/pen/yaLxBX>

Basis of our tabs design

<https://www.w3schools.com/howto/howto_js_vertical_tabs.asp>

Referenced for flex box heavily

<https://css-tricks.com/snippets/css/a-guide-to-flexbox/>

