CS 510 Full Stack Web Development
Web Form Project Report
Haomin He

Overview

This project is aimed at creating an event registration form that helps event host knows more about his/her guests. For instance, how many people are coming to the event and how much vegetarian food should be ordered. I build a Wedding Registration Form that includes a front-end web interface in jQuery and Bootstrap, and a back-end server that tracks the form data which is written in JavaScript.

Structure

In this project, there are four websites display useful information. They are: index.html, confirmation.html, guests.html, and localhost:8080/registered. The first two websites are used by guests for registering the event. The last two websites are used by host to display information of guests.

In index.html, there is a main form for family head. Here we assume each family has only one family head; we use the family head's email as a primary key to identify the entire family; the family head is an adult, so he/she doesn't have 'Sit At Kids Table' option; the head may bring other family members to attend the event as well; both family head and members could be vegetarian, so they all have 'Vegetarian' option; other family members' email address are not needed; members might be kids, therefore they have 'Sit At Kids Table' option.

If the family head wants to bring more family members to the event. He/she can simply click on 'Additional Guest' button. Guests can check corresponding options to indicate their dietary and sitting table preferences. They can click on 'Submit' button after filling out the form. The page jumps to confirmation.html which shows guests have registered the event successfully.

On the other side, host can see guests' information on guests.html, and localhost:8080/registered. From here you can see how I choose to parse the data. Overall, data is a collection that is identified by family head's email address. Then it is followed by information of family head. Each family head contains a family members collection where stores information of the rest of family.

In order to display information nicely on a table, I count how many people in each family; so that I can use 'rowspan' on email attribute. This display method helps host easily identify who the guests are under each different email address. Host can contact guests through emailing. Total number of guests is calculated at the end of guest.html page. If there are newly added guests, refreshing page guests.html can reflect new changes.

All guest information is stored into a file called 'guests. json'. The very first thing my program does is reading data from this file; parse and display data on guests.html, and localhost:8080/registered. If there is no such a file exists in the directory, the program will create one automatically. Newly added guest information is written to this file. As a result, if I close my current running program session, all my data is reserved. Next time I run my program again, my old guest data is displayed as well.