Personal Portfolio 2

Course: IFB299 – Application Design and Development

Name: Ibrahim Alharbi

Student Id: 08653097

I am Ibrahim Alharbi, a student from Information Systems School. I was highly involved with the development of Release 2. I did the whole works related to managing the duty roster for all of the volunteers. Volunteers can view their duties in their portal and send request to manager to change the duty (if required). On the other hand, managers are able to assign, view, and accept or reject requests from all volunteers. Both manager and volunteers can search duties using an easy search panel.

My main focuses in this release includes 1) Designing user-friendly GUI, 2) creating of database table for storing related data in efficient way 3) developing the pages where manager can create roster, and both volunteers and manager can view duties, 4) developing the options where volunteers can send change request to manager, and 5) developing the facility for manager so that they can accept or reject request for volunteers. Testing the functionality of the code was performed time to time. In following, I described each of these artefacts briefly.

Artefact 1 – I put strong effort to make the GUI simple so that both volunteers and managers can easily use it. I designed the 'volunteer_my_roster.php' where volunteer can view their duties with a simple but effective search option and also they can send change requests for each entry. 'volunteer_roster.php' page was designed for manager which had many functionalities including creating roster, deleting, accepting or rejecting requests, but these can be done in a very simple way, GUI.

Tools – *pen and paper, html, css.*

Artefact 2 – I created a relational table named 'scheduling' in the MYSQL database. This table stores date of duty with corresponding volunteer id. It has two more attributes 'request' and 'reqdate' to store and handle the requests from the volunteers. Value of request can be 0 to 3 where 0 means no request for that entry and 1 or 2 represent date change and delete request respectively. When it is a date change request then 'reqdate' attribute holds the new requested date otherwise it's always NULL. For each entry the information of volunteer can be obtained by joining this table with 'logins' table.

```
SELECT logins . * , scheduling . * FROM scheduling INNER JOIN logins ON logins.id =
scheduling.volunteer_id
```

Tools – *mysql.*

***Artefact 3* –** I developed the two web pages for manager (volunteer_roster.php) and volunteers (volunteer_my_roster.php). Manger can create or delete roster for a volunteer and also view/search the duty roster for all volunteers. Creation of roster and searching are done when it is submitted by two php pages: create_roster.php and show_roster.php respectively. Manager can also delete a duty roster which was handled by delete_roster.php page.

On the other hand volunteers can view only their rosters and able to use simple search option. It was done with the show_my_roster.php page. All of these webpages validated during implementation.

***Tools* –** *php, mysql, html.*

***Artefact 4* –** After implementing simple duty scheduling, I aimed to provide more facility to the volunteers so that they can send request to manager to change a duty. I developed the changedelete_my_roster.php file for this purpose. When volunteer clicks change date or cancel request submit-button then this page identifies all necessary information and update the database table 'scheduling' accordingly.

***Tools* –** *php, mysql, html.*

***Artefact 5* –** I built the panel for the manager so that they can accept or reject the requests of volunteers. I update the delete_roster.php page to make it enable to handle accept or reject actions. Following algorithm shows how this web page updates 'scheduling' relational table and handles accept and reject commands.

```
if it's date change request:
    if accept:
        change dateofduty to reqdate
        request = 0
    else if reject:
        request = 0
else if it's delete request:
    if accept:
        delete that entry
        request = 0
    else if reject:
        request = 0
```

***Tools* –** *php, mysql, html.*

Necessary files are attached.