**Team 52 - PartyFreaks – PartyGuard Web Application**

**Client:** Jacob Stansbury ([S516098@mail.nwmissouri.edu](mailto:S516098@mail.nwmissouri.edu))

**Roles and Responsibilities:**

1. **Primary Contact:**Prem Abishai Parise
2. **Client management:**Nihitha Reddy Bhimireddy
3. **Requirements management:**Lokesh Reddy Pothapu
4. **Data management:**Srichaitanya Peddinti
5. **Issues management:**Sindhuja Pannala
6. **Quality and testing management:**Spandana Velaga
7. **Communications and documentation management:**Gopinadh Cherupally

**Artifacts:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Artifact** | **Evaluation/Notes** | **% Complete** |
| Prem Abishai Parise | Project Folder | Project folder has been redesigned so that its compatible with Angularjs/expressjs/nodejs | 80% |
| Login page Navigation | Navigation to different screens based on the User type | 80% |
| Events page | Added time field and populated dummy data from json object | 70% |
| Nihitha Reddy Bhimireddy | Login page Navigation | Navigation to different screens based on the User type | 80% |
| Events page | Added time field and populated dummy data from json object | 70% |
| Client feedback | Communicated with client to get feedback and suggestions. | 100% |
| Spandana Velaga | HowitWorks.html | Updated HowitWorks.html page | 60% |
| FAQ.html | Updated Faq page | 50% |
| Srichaitanya Peddinti | Members.html | populated dummy data from Json object .Should include api functionality once the service gets available | 60% |
| Events page | Populated dummy data from json object. Should include api functionality once the service gets available | 60% |
| Lokesh Pothapu | GuardProfile.html  BasicProfie.html | Updated code to enable editing option in both pages. | 60% |
| Gopinadh Cherupally | Test cases | Created test cases for few UI pages. | 50% |
| Sindhuja Pannala | integratedMaps into the addCustomer page | obtained the latitude and longitudes | 80% |

**Accomplishments:**

* **Project Folder Structure:**Almost finished restructuring project folder to integrate Node.js, Express. Js and Angular Js. Installed all the required software and made project running in the server

     Repo Link: [https://github.com/s525080/PartyGuardWeb (Links to an external site.)](https://github.com/s525080/PartyGuardWeb)

     Members involved: Prem Abishai Parise, Nihitha Reddy and Srichaitanya

     Hours: 7 hrs.

     Challenges faced: As both express.js and angular.js have own routing functionality it was a challenge for us to separate email functionality and rest of the page routing functionalities. Did research for many hours and finally found a solution to separate both `of them

* **Login Page Navigation:**As each user has different navigation flow, we modified the functionality to navigate to respective screens based on the user.as of now we are using dummy data from controller. Once we get the web service running. we will use that to control the user navigation

 Repo Link:

* [https://github.com/s525080/PartyGuardWeb/blob/master/app/view/Navbar/Login.html (Links to an external site.)](https://github.com/s525080/PartyGuardWeb/blob/master/app/view/Navbar/Login.html)
* [https://github.com/s525080/PartyGuardWeb/blob/master/app/config/config.js (Links to an external site.)](https://github.com/s525080/PartyGuardWeb/blob/master/app/config/config.js)
* [https://github.com/s525080/PartyGuardWeb/tree/master/app/controller/authentication (Links to an external site.)](https://github.com/s525080/PartyGuardWeb/tree/master/app/controller/authentication)

      Members involved: Prem Abishai Parise, Nihitha Reddy, and Spandana Velaga

      Hours: 4 hrs.

* **Events page:**Added time field and populated dynamic data in the table.as of now we are using dummy data from controller. Once we get the web service running. we will use that to control the user navigation

 Repo Link: [https://github.com/s525080/PartyGuardWeb/blob/master/app/view/HostUser/Events.html (Links to an external site.)](https://github.com/s525080/PartyGuardWeb/blob/master/app/view/HostUser/Events.html)

      Members involved: Sri Chaitanya, Prem Abishai Parise, and Nihitha Reddy

      Hours: 3 hrs.

* **HowitWorks and FAQ page:**modified both pages

 Repo Link: [https://github.com/s525080/PartyGuardWeb/blob/master/app/view/Navbar/Howitworks.html (Links to an external site.)](https://github.com/s525080/PartyGuardWeb/blob/master/app/view/Navbar/Howitworks.html)

[https://github.com/s525080/PartyGuardWeb/blob/master/app/view/Navbar/faq.html (Links to an external site.)](https://github.com/s525080/PartyGuardWeb/blob/master/app/view/Navbar/faq.html)

      Members involved: Spandana velaga

      Hours: 2 hrs.

* **Data broadcast functionality:**implemented broadcast function using Angularjs to pass data from one controller to other controller when we click on button.

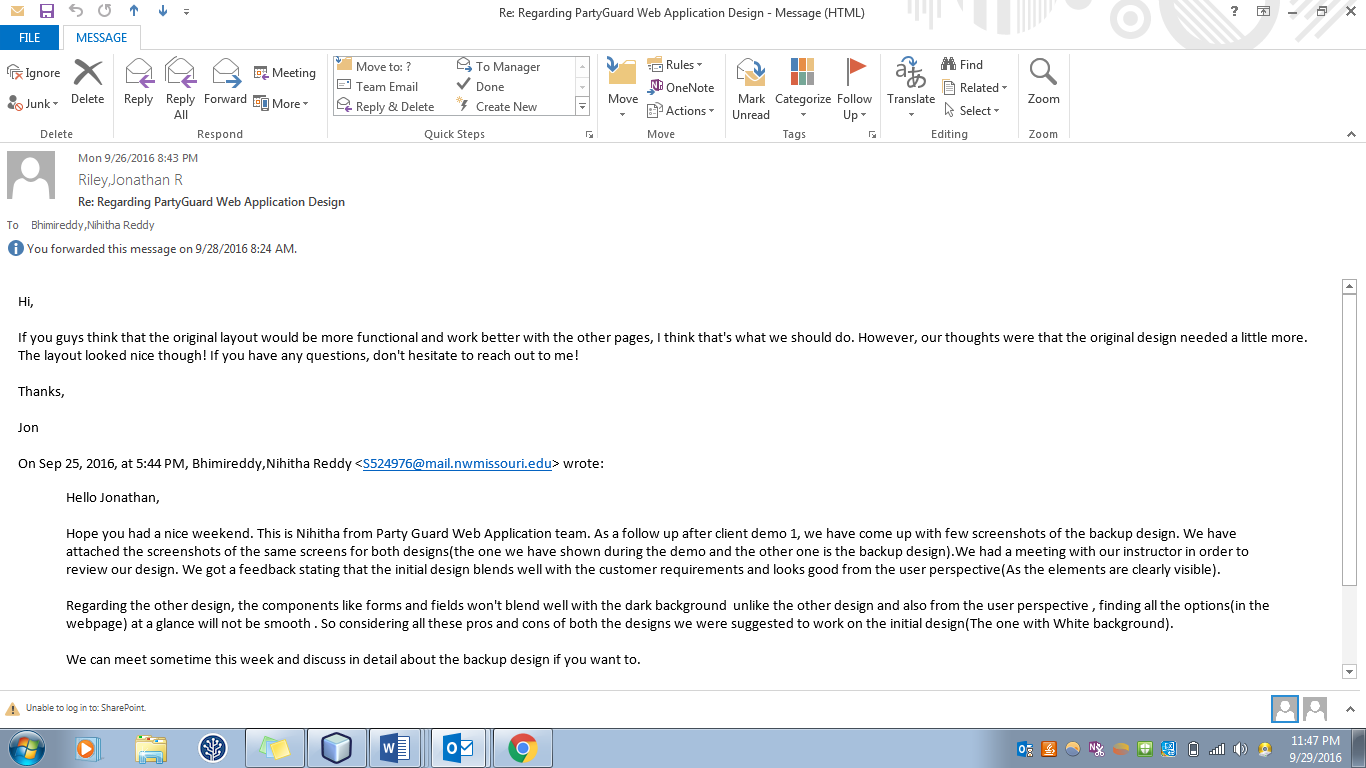
 Repo Link: [https://github.com/s525080/PartyGuardWebApp\_PartyFreaks/blob/018b7682f19ce6ceabcdf7ee8f0fdd2ba87ed365/loveforangular.zip (Links to an external site.)](https://github.com/s525080/PartyGuardWebApp_PartyFreaks/blob/018b7682f19ce6ceabcdf7ee8f0fdd2ba87ed365/loveforangular.zip)

      Members involved: Lokesh pothapu

      Hours: 3 hrs.

* **Feedback:**Communicated with client about the design webpages.

     Screenshot:



     Members involved: Prem Abishai Parise, Nihitha Reddy

     Hours: 30 mins

* **Test cases:**Prepared test cases for few UI pages

      Repo link: [https://github.com/s525080/PartyGuardWebApp\_PartyFreaks/blob/PartyGuard\_Working/testcases.zip (Links to an external site.)](https://github.com/s525080/PartyGuardWebApp_PartyFreaks/blob/PartyGuard_Working/testcases.zip)

      Member involved: Gopinadh Cherupally

      Hours – 4 hrs

* **Map Functionality:**obtained the latitude and longitudes

      Repo link: [https://github.com/S525079/partyguardwebapp\_partyfreaks/commit/2092f2a63208b6830a18a747e7781f48377789ef (Links to an external site.)](https://github.com/S525079/partyguardwebapp_partyfreaks/commit/2092f2a63208b6830a18a747e7781f48377789ef)

      Member involved: Sindhuja Pannala

      Hours – 3 hrs

* **Client Demo:**Prepared for Client demo. Added various functionalities overall to give demonstration to client

      Repo link: https://github.com/s525080/PartyGuardWeb

      Member involved: Entire team

      Hours – 5 hrs

**Planned Tasks:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Task** | **Hours** | **Deliverable** |
| Prem Abishai Parise | Complete routing of webpages in config file | 3 | Working webpages. |
| Implement Api functionality in Login page,Add Customer,Customer acknowledgement pages | 5 | Working webpage and screenshots |
| Nihitha Reddy Bhimireddy | Implement Api functionality in Hostprofile page,House location page | 5 | Working webpage and screenshots |
| Spandana Velaga | Start implementing Angularjs functionality  in Contactus and Basic user profile page | 3 | Working webpage and screenshots |
| Srichaitanya Peddinti | Implement Api functionality on evetns page, Members page..  Create GuardUser events page. | 5 | Working webpage and screenshots |
| Lokesh Pothapu | implementing functionality Basicuser nad Guard user profile pages | 4 | Working webpage and screenshots |
| Gopinadh Cherupally | Testing already developed webpages.  Learning selenium | 5 | Test cases |
| Sindhuja Pannala | Create Subscription pages.  Will be working on parsing JSON and populating the data onto those pages | 6 | Working webpage and screenshots |

**Challenges:** Inorder to implement API functionality we need to have Web service up and running. So as long as the webservice is delayed,our progress will be slow. anyhow we are writing the functionality by using dummy data.So once the webservice is up, we will integrate it.