**Incremental and Regression Testing - Team 4**

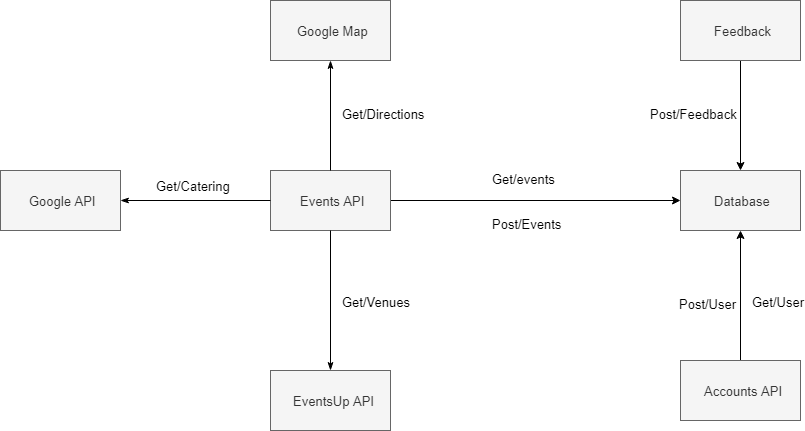
February 9, 2019

**Team members:** Ethan Niu, Sripath Mishra, Siddharth Dhar, Shivangi Chand, Pooja Tewari

**Project Coordinator:** Adam Johnson

1. **Classification of Components**

1.1 -



|  |  |  |  |
| --- | --- | --- | --- |
| **Component** | **Input** | **Output** | **Dependent** |
| Database | Event information, profile information, feedback information | Trip information, profile information | Accounts API, Events API, FeedBack API |
| Accounts API | User information | User information | Database |
| Events API | Event information | Event information | EventUp API, Google Map, Amazon API,  Google API |
| Google Map | Location coordinates (lat, long) | Directions to the place | Events API |
| Feedback API | Feedback from users | Feedback to Developers | Database |
| Google API | Location | Catering options | Events API |
| EventUp API | Location | Hosting location options | Events API |

1.2 - We first started by building the backend and frontend of our web app simultaneously. This allowed us to do a bottom-up testing approach by following the control flow of our software and system. After which we accrued all the components and conducted incremental testing.

1. **Incremental and Regression Testing**

**Incremental Testing Defect Log**

|  |  |  |  |
| --- | --- | --- | --- |
| **Defect No.** | **Description** | **Severity** | **How to correct** |
| 1 | Fixed all non-bootstrap CSS pages to be viewed correctly | 2 | Limited the global scope of the CSS pages |
| 2 | Route of forgot password page would not take security question as a parameter | 1 | Changed the backend route to accept an additional query parameter |
| 3 | CSS was not showing up on event invitation email | 2 | Used a different npm package for displaying the CSS in the emails |
| 4 | Backend for event hosting location broke | 1 | Changed the POST request to the EVENTup API |
| 5 | CORS was disabled for the backend | 1 | Enabled CORS by using the CORS npm package |
| 6 | The frontend and the backend was not able to be deployed on the same Heroku instance | 1 | Created two separate instances for the backend and the frontend |
| 7 | The data type used to store the date was incorrect | 2 | Changed the data type from String to date-time |
| 8 | SQL injection was possible for the feedback route | 1 | Used parameters in place of string manipulations to create the SQL statement |
| 9 | False positives of an invalid email | 3 | Used a different npm package to validate the emails |
| 10 | Deletion of guests from the guest list does not occur | 1 | Updated the SQL statement to allow deletion of the guest list |
| 11 | Reused docker image did not copy the updated code to the working directory of the docker container | 2 | Updated the dockerfile to copy the new code upon rebuild |
| 12 | Incorrect HTTP API protocols used for 2 forget password routes | 1 | The HTTP API protocols were corrected for the incorrect routes |
| 13 | Parameters passed to an SQL statement were in incorrect order for creating events route | 1 | Fixed the order of the parameters |

**Regression Testing Defect Log**

|  |  |  |  |
| --- | --- | --- | --- |
| **Defect No.** | **Description** | **Severity** | **How to correct** |
| 1 | While fixing non-bootstrap CSS pages, the pages with bootstrap broke because the scope of the global coverage was limited to the div | 2 | The whole code was covered in an outer div which gave the bootstrap pages the whole scope required |
| 2 | While fixing the forget password route to take the security question as a parameter, the frontend AXIOS broke because it did not pass the security question | 1 | Updated the AXIOS of the frontend to pass the security question |
| 3 | When the backend was fixed for the EVENTup API issue, the frontend had to pass an additional parameter, which it did not | 1 | Fixed the frontend AXIOS to pass an extra parameter of range |
| 4 | When CORS was enabled in the backend the frontend broke as it had to pass the extra parameter in the header named (ie. access-cross-origin) | 1 | Enabled CORS in the frontend by using a CORS npm package |
| 5 | Upon splitting the backend and the frontend codebase, the package.json file was taking unnecessary dependencies of the frontend for the backend and vice-versa | 2 | Having independent package.json files for both the codebases |
| 6 | Upon passing a date-time datatype for event date, the backend route had to be fixed to not send a string. | 1 | Converted the type of the date parameter received from the front end to date-time from string |
| 7 | Upon changing the SQL statement to take in parameters, the create event backend route broke. | 1 | Updated the backend route to follow the correct procedures. |
| 8 | Upon deletion of guest from the guest list, entire guest list gets deleted | 1 | Fixed the SQL statement to delete only one guest at a time. |
| 9 | Upon correcting the order in which the parameters were passed to the create event route, the backend route of the same broke. | 1 | The order of the parameters passed to the database package was fixed. |
| 10 | Upon correcting the HTTP API protocols for the forget password route, the frontend of the same broke. | 1 | The front-end code base was changed to use the correct HTTP API protocols. |
| 11 | Upon changing the npm package that validates emails, extra configurations had to be done to check for domains | 3 | Added the domain testing configuration. |

1. **Update Product Backlog**

Newest version updated and attached to the email.