PROJECT

**CSCI 5409 – CLOUD COMPUTING**

**GROUP 25**

**MEMBERS**

B00836612 – Siddhant Ashutosh

B00832226 – Praneeth Nuthalapati

B00841521 – Jatin Partap Rana

B00822138 – Tomi Lechpammer

B00830791 – Moni Shah

B00836700 – Harshit Trivedi

Table of Contents

[Scenario for Business Context 3](#_Toc47805979)

[High-Level Scenario 3](#_Toc47805980)

[Functional requirements 3](#_Toc47805981)

[Non-functional requirements 3](#_Toc47805982)

[Deficiencies 3](#_Toc47805983)

[Meritable features/properties 3](#_Toc47805984)

[Software Overview 3](#_Toc47805985)

[Collaboration Summary 4](#_Toc47805986)

[Minutes of Meetings 4](#_Toc47805987)

[Microsoft Teams 4](#_Toc47805988)

[Trello 5](#_Toc47805989)

[Gitlab 6](#_Toc47805990)

[Resources-Related Information 7](#_Toc47805991)

[References 8](#_Toc47805992)

# Scenario for Business Context

## High-Level Scenario

There are three companies which work in collaboration with each other. Customers visit the website of Events 4U when they are looking to book an event. Events 4U allows customers select a venue for whatever event they desire and also required corresponding equipment. The company, however, does not own the facilities or equipment, but rather communicates with their conglomerates to support the service. One such supplier is Valiant Venues, a company that specializes in property rentals. Valiant Venues owns several multi-purpose properties which are great locations for any sort of indoor event. These locations include a ballroom, public house, community hall, and recreation zone. Although having a spot to hold an event is essential, an open space is not enough to host any function – additional equipment is required. This is the role of Lights N Stuff, a company that specializes in technical equipment. They offer to provide whatever may be lacking in a venue to make it ready for use. Not enough light in the hall? They will bring tripod commercial lighting to illuminate the place. No speakers in the ballroom? Lights N Stuff will supply cabinets of speakers to liven up the spot. Together these three companies work to make any event happen for their customers.

## Functional requirements

We followed three-tier architecture for developing this project that includes frontend, backend, and database. One of the companies will act as a middle layer (coordinator) to handle the transaction. We have used Node JS and Express JS to develop out backend functionality. We have used MVC framework and MERN (MySQL, Express JS, React JS, Node JS) stack architecture. Each company has its own database to store data. We have used Sequelize ORM framework [1] to develop all three companies’ backend. All the backends for all the companies had been containerized. We have used AWS and Heroku cloud platforms to develop and deploy our applications. We have deployed frontend of a company in S3 bucket to add serverless module in our application.

We are using MYSQL transactions to store the data in the database. We are creating the transaction in auto commit mode as false using Sequelize ORM. Event 4 U company is acting as the coordinator and it waits to commit the transaction until event company and equipment company gives the success message which means that both the companies APIs have completed their task. After getting the success message from both the companies Event 4U company commits the transaction. In case anything error comes in any of the company then the transaction gets rolled back.

## Non-functional requirements

Users are not allowed to book the same venue twice. Application will be available to users 24/7 unless the cloud instances are terminated. Communication amongst three companies is over secure channels. All three companies’ website are mobile-friendly and responsive. All the input forms have input validations and show appropriate error messages. Each request will be processed under 5 seconds. All three websites will have consistent design and color choices.

# Deficiencies

There are no known deficiencies in the submission. As to the team’s knowledge, all developed software adheres to the requirements.

# Meritable features/properties

In addition to the functional requirements we have implemented a home page for the project with a navigation bar linking to each company’s webpage. Valiant Venues and Lights N Stuff also include links back to the homepage from their navigation bars. This configuration allows the entire assignment to be traversed via the user interface without reentering any endpoint information. Another component developed in excess of requirements is the session management feature for Events 4U. Once a user successfully logs in to complete an order, they are not prompted to log in for subsequent orders unless they have logged out beforehand.

# Software Overview

Valiant Venues, Lights N Stuff, and Events 4U have all been implemented using the ReactJS framework [2]. The data for all venues and equipment was stored on a MySQL server, which was deployed on an AWS EC2 instance. Although equipment and venue information shared a server, they had their information stored in separate databases. The database for Events 4U, storing data on orders and users, was configured on a different server and deployed on an AWS instance separate from the other two databases. The backends of Valiant Venues and Lights N Stuff follow the MVC pattern, connecting models to their corresponding databases and using controllers to update the models for create, update, read operations. The backend of each company was deployed in a container on a separate AWS instance. The front ends of the three companies were deployed independently and make API calls using Axios [3] to their respective backends to facilitate interaction with users. Lights N Stuff has its front end deployed in a Heroku [4] app. Similarly, the front end for Events 4U has its front end deployed in another Heroku app. In order to satisfy the serverless computing requirement, we deployed the front end of Valiant Venues in an AWS S3 [5] bucket. Overall, the architecture of our web applications includes EC2 instances, two Heroku applications, and an AWS S3 bucket.

# Collaboration Summary

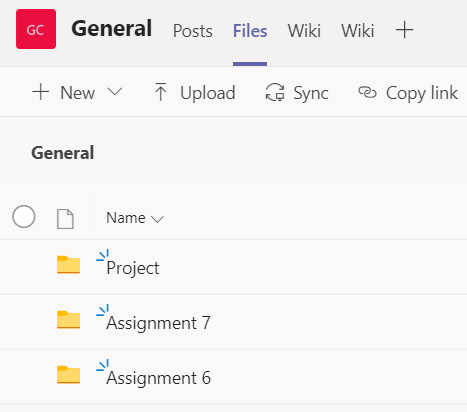
## Minutes of Meetings

*Table 1.* Summaries of team meetings.

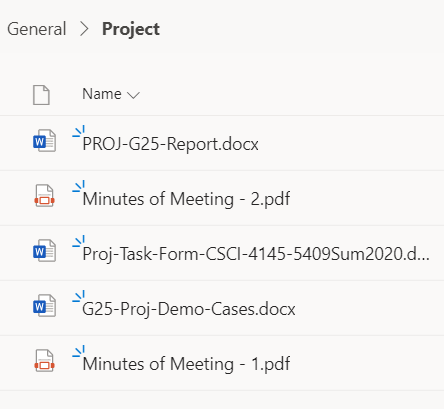
|  |  |  |
| --- | --- | --- |
| **Date** | **Attendance** | **Summary** |
| August 6, 2020 | Siddhant Ashutosh, Praneeth Nuthalapati, Jatin Rana, Moni Shah, Harshit Trivedi, Tomi Lechpammer | We met to discuss project requirements and expectations. We discussed a business scenario wherein one company is responsible for renting out venues and they coordinate with another company that rents out equipment. A third company allows customers to rent places for various events, using the venue of the first company and equipment of the second company. We also discussed how to implement the managed transactions while keeping a serverless component. We decided to all do research on XA transactions and meet the following day with our findings to finalize an approach. |
| August 7, 2020 | Siddhant Ashutosh, Praneeth Nuthalapati, Jatin Rana, Moni Shah, Tomi Lechpammer, Harshit Trivedi | We discussed our findings for how to implement the project. We decided that we will use XA transactions in the MySQL database for transaction management, with the company that offers events to clients acting as the controller of the transactions. This will be implemented by Siddhant, Praneeth, and Jatin. For serverless, we decided that Tomi will implement the frontend of company X in an S3 bucket. Moni and Harshit are to implement UI modifications to coincide with new scenario and investigate if 2FA is feasible to implement. |

## Microsoft Teams

MS Teams [6] was the primary communication platform used in the project. We carried out two main meetings where everyone attended and we planned our approach to the project. After we split into smaller groups to work on specific parts, we would have calls via Teams to collaborate and update each other on our progress. Teams was also used to share all of the documentation. We created a folder for the project (see Figure 1) and added each meetings minute, the task form, testing and demo documents, and the report so that everyone could easily contribute (see Figure 2).



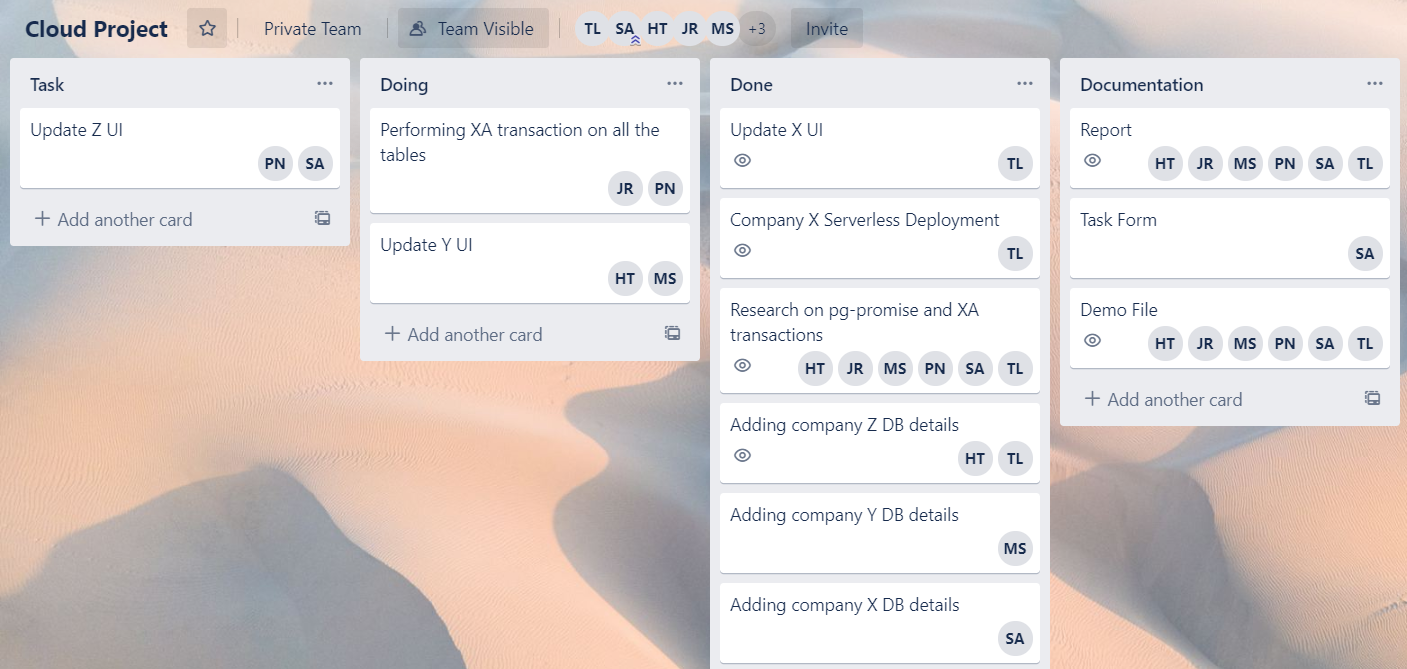
*Figure 1. Screenshot of MS Teams general directory structure.*



*Figure 2. Screenshot of MS Teams project directory structure*

## Trello

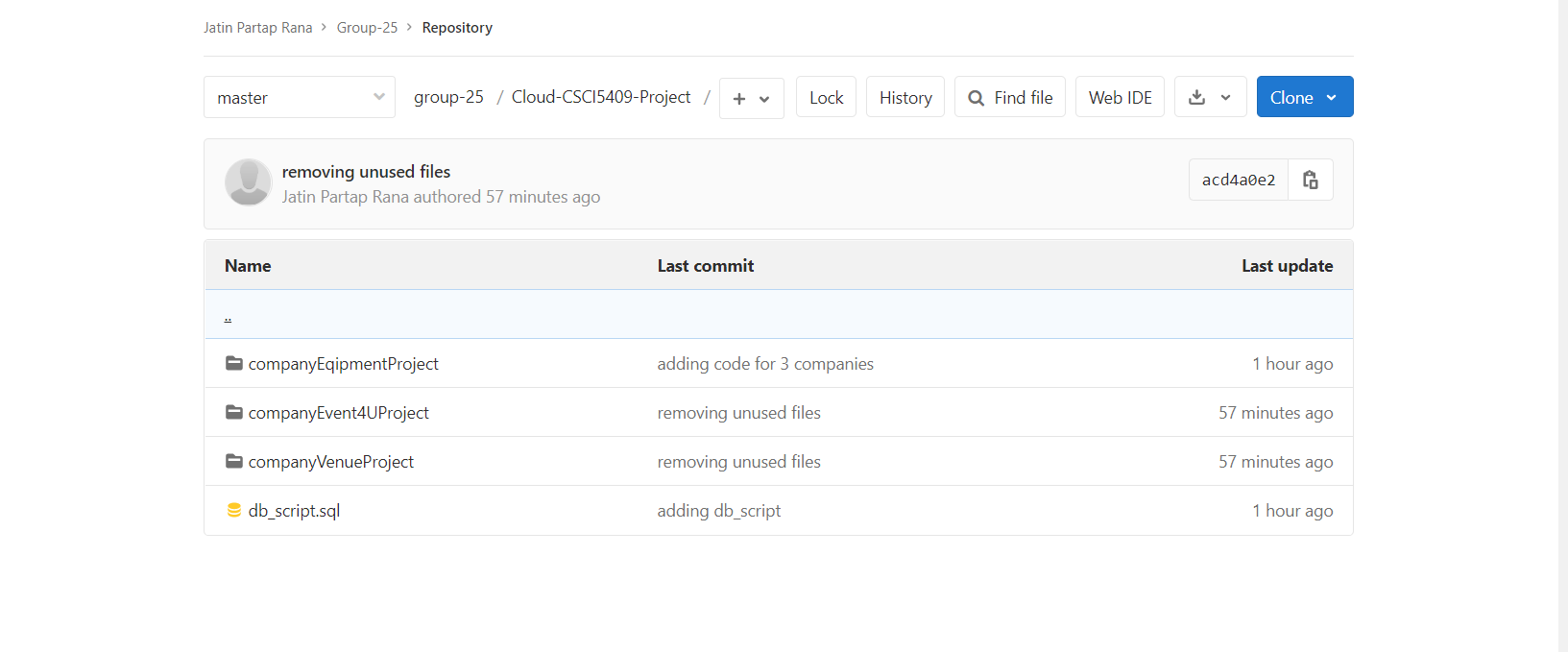
Trello [7] was implemented to track and distribute relevant tasks, as with Assignments 6 and 7. All tasks were agreed upon by the team during meetings and were then added to Trello. Three categories were utilized during development: “Tasks,” “Doing,” and “Done.” These tasks included research that was needed to complete the work in addition to development tasks. Particularly, we needed to research how we could implement the XA transactions and synchronize them effectively. We also investigated how to deploy to the AWS S3 serverless environment, and how we could implement two-factor authentication for login. An additional category, “Documentation” was used as a checklist to follow up on all documentation after primary development concluded; tasks were created and assigned for each deliverable. Trello allowed us to effectively organize and track all required work throughout development. Figure 3 shows what the Trello board looked like towards the end of development.



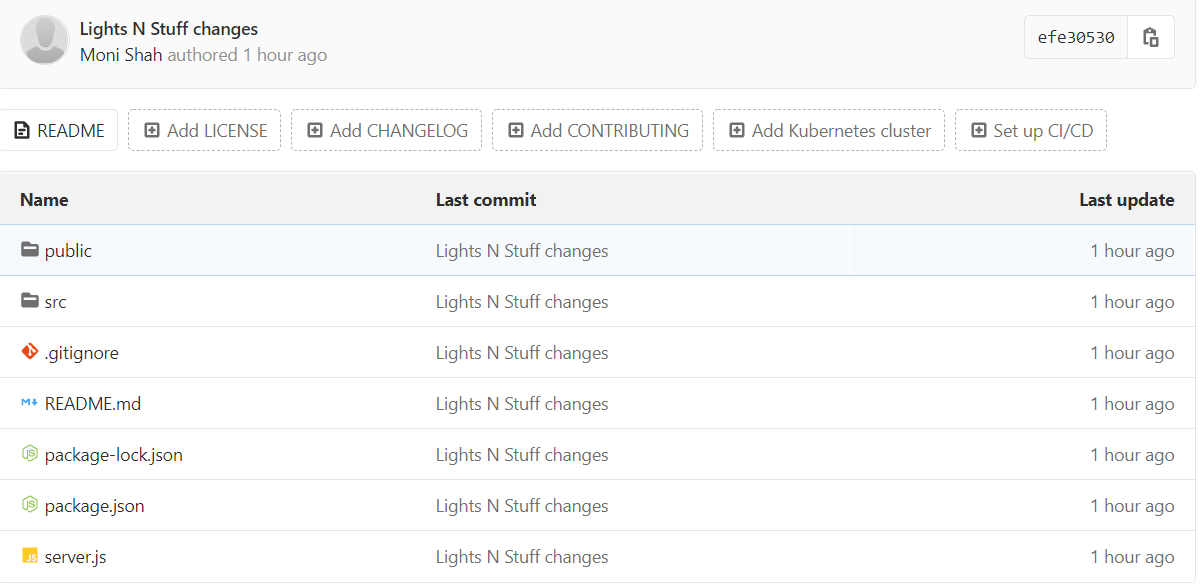
*Figure 3.* Screenshot of the Trello board structure.

## Gitlab

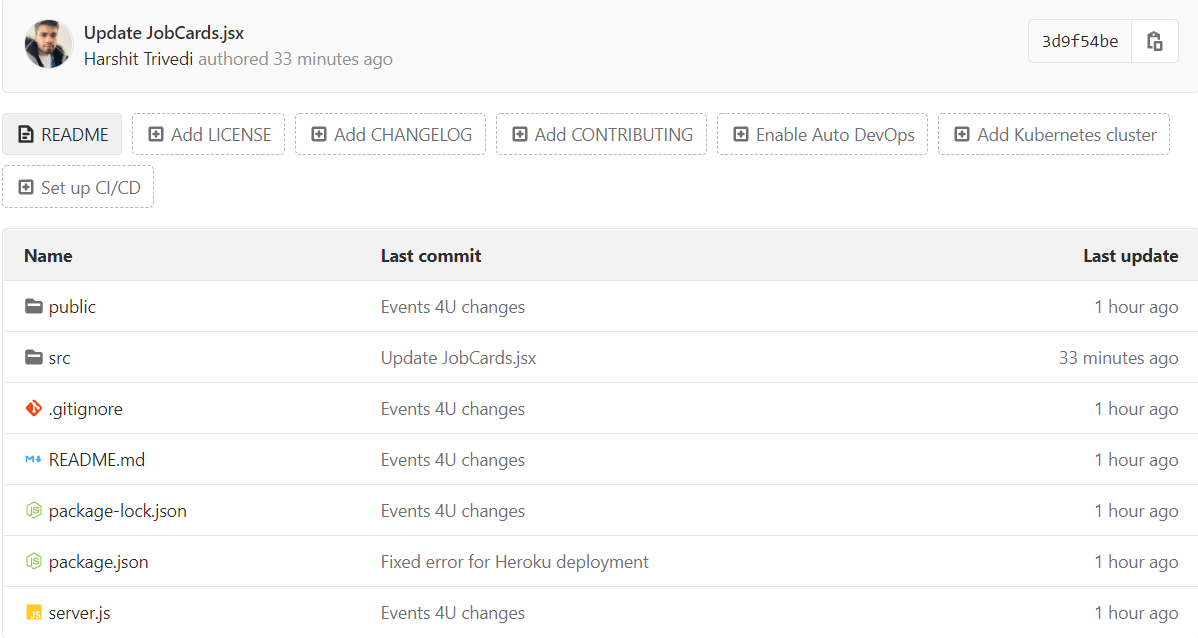
Three GitLab [8] repositories were used to collaborate on the project. The first, “group-25,” was used to store the code for all three companies in backend directories (Figure 4) as well as the frontend code for valiant venue. We have used ReactJS framework for all the three companies. A second repo, “Frontend-Group25” contained the code for the Lights N Stuff. The structure of the second repo is displayed in Figure 5. A third repo was used to store the frontend code for Events 4 U as shown in Figure 6.



*Figure 4. Directory structure of “group-25” master branch.*



*Figure 5. Directory structure of “Frontend\_Group25” master branch.*



*Figure 6. Directory structure of “cloudProject\_Events4U” master branch.*

# Resources-Related Information

## Cloud resources used:

EC2 instance for backend deployment of all the three companies i.e. Valiant Venues, Lights N Stuff and Events 4U.

## Frameworks used for development:

UI for Valiant Venues, Lights N Stuff and Events 4U developed using React JS.

Backend for Valiant Venues, Lights N Stuff and Events 4U developed using Node JS and Express.

# References

[1] “Transactions,” Sequelize. [Online]. Available: https://sequelize.org/master/manual/transactions.html. [Accessed: 08-Aug-2020].

[2] "React – A JavaScript library for building user interfaces", Reactjs.org, 2020. [Online]. Available: https://reactjs.org/. [Accessed: 07- Aug- 2020].

[3] "Axios", npm. [Online]. Available: https://www.npmjs.com/package/axios. [Accessed: 07- Aug- 2020]

[4] "Cloud Application Platform | Heroku", Heroku.com. [Online]. Available: https://www.heroku.com/. [Accessed: 07- Aug- 2020].

[5] “Amazon S3,” aws.amazon.com, 2020. [Online]. Available: https://aws.amazon.com/s3/. [Accessed: 07- Aug- 2020]

[6] "Free Chat, Video Calling, Collaboration | Microsoft Teams", Microsoft.com, 2020. [Online]. Available: https://www.microsoft.com/en-ca/microsoft-365/microsoft-teams/free. [Accessed: 07- Aug- 2020]

[7] "Trello", Trello.com, 2020. [Online]. Available: https://trello.com/. [Accessed: 07- Aug- 2020].

[8] "Projects · Explore", GitLab, 2020. [Online]. Available: https://gitlab.com/explore. [Accessed: 07- Aug- 2020].

[9] Group-25, “Assignment 7”, Dalhousie University [Online]. [Accessed: 08-Aug-2020].

[10] J. T. Mark Otto, “Buttons,” · Bootstrap. [Online]. Available: https://getbootstrap.com/docs/4.0/components/buttons/. [Accessed: 08-Aug-2020].