****

GROUP PROJECT II

GROUP G32

**SCS 3214 / IS 3113: Group Project II - 2022**

**Project Proposal**

**Proposed Project Title: Entero**

**Project Group Details**

1. Group number: G32
2. Group members:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Name*** | ***Reg. Number*** | ***Index Number*** | ***Email address*** | ***Mobile Phone*** |
| S. Kavishan | 2019/CS/068 | 19000685 | [kavishansukumar@gmail.com](mailto:kavishansukumar@gmail.com) | 775844312 |
| S. A. R. P Athauda | 2019/IS/007 | 19020074 | [shivakumar.priskila@gmail.com](mailto:shivakumar.priskila@gmail.com) | 772150504 |
| B.Y. M. Fernando | 2019/CS/044 | 19000448 | [fernandoymadhushika@gmail.com](file:///C:\Users\hp\Desktop\fernandoymadhushika@gmail.com) | 771687443 |
| K.K. S. Punsara | 2019/CS/124 | 19001241 | [kksenalpunsara@gmail.com](mailto:kksenalpunsara@gmail.com) | 710554474 |
| M.S.M. Shakir | 2019/IS/080 | 19020805 | [shakirsaheel123@gmail.com](mailto:shakirsaheel123@gmail.com) | 771979029 |

**Details of Project Supervisor, Co-supervisor, Advisors and Clients**

**Proposed Project Supervisor (Academic Staff of UCSC):**

Name of the supervisor: Ms. Mathangi Krishnathasan

Signature of the supervisor:

Date:

**Proposed Project Co-Supervisor (Assigned by Course Coordinator):**

Name of the co-supervisor: Ms. Yasmi Himaya

Signature of the co-supervisor:

Date:

**Project Advisors:** (External industry advisors, if any)

(Please provide, Name, Organization, email address and institute)

1. …………………………………………………………………………………………….
2. …………………………………………………………………………………………….
3. …………………………………………………………………………………………….

**The client of the Project**

Name of the client: Ms. Mathangi Krishnathasan

Address of the client: University of Colombo School of Computing, UCSC Building Complex,35, Reid Avenue, Colombo, Sri Lanka

Contact number of the contact person: +94 77 331 1798

e-mail address of the contact person: [tmk@ucsc.cmb.ac.lk](mailto:tmk@ucsc.cmb.ac.lk)

***Table of Contents***

[1. Project title: 1](#_Toc106804797)

[2. The goal: 1](#_Toc106804798)

[3. Objectives: 1](#_Toc106804799)

[4. Tentative problem definition 1](#_Toc106804800)

[5. A brief introduction to the project 2](#_Toc106804801)

[6. The scope of the project 3](#_Toc106804802)

[6.1. In-Scope 3](#_Toc106804803)

[6.1.1. Users (possible actors) of the system: 3](#_Toc106804804)

[6.1.2. Main functionalities of the system: 3](#_Toc106804805)

[6.2. Out of scope 4](#_Toc106804806)

[7. Quality attribute identification 4](#_Toc106804807)

[8. Tentative technologies 6](#_Toc106804808)

[9. Feasibility study 10](#_Toc106804809)

[9.1. Technical feasibility 10](#_Toc106804810)

[9.2. Operational feasibility 10](#_Toc106804811)

[9.3. Economic feasibility 11](#_Toc106804812)

[9.4. Schedule feasibility 11](#_Toc106804813)

[9.5. Legal and Ethical feasibility 12](#_Toc106804814)

[10. Main deliverables of the system 12](#_Toc106804815)

[11. The project plan 13](#_Toc106804816)

[12. References 14](#_Toc106804817)

[13. Declaration 15](#_Toc106804818)

**Project Details:**

# Project title:

Entero – All in one (Event Management System)

# The goal:

The main goal of the project ‘Entero’ is to provide a web based platform for event management which facilitate and satisfy the users towards various services. System ‘Entero’ assist the service providers to reach and promote their business as well as customer to fulfill their needed services through this single system.

# Objectives:

* To provide user friendly system to the users.
* To facilitate the customer to receive the required service from the service providers.
* To facilitate the customer to schedule appointments by checking available time slots.
* To facilitate the customer to make online bookings for event services.
* To provide chat facility to maintain the communication with service providers.
* Design a system to provide and view the reviews and feedbacks regarding the services.

# Tentative problem definition

* Finding event services and proper arrangement for an event had been more complicated.
* Customers are not aware about the service providers or contractors to contact or book for the event services.
* Customers are not ensured about the services whether they provide the services according to our requirements or needs and the service standard assurance will be shallow.
* Mainly there were no single system which provide event services together in Sri Lanka.

# A brief introduction to the project

This “Entero” system is an event management system with three user roles of customer, service providers and admin. The main purpose of this system is to provide several event services such as refreshments, decorations, hall bookings, photography by combining different service providers and make a common platform for both service providers and customers for their ease of engagement for business purpose and feasible event service contracts. Here the service providers can register to the system and make their organizational portfolios by using the given templates in the system which contains information about the service providing company, appointment fixing booking schedule, ratings, reviews, related images, etc. which are used to expose about their organization and provided services. Service providers ought to pay required percentage of service charge (Subscription fee) for the system which will be calculated according to the number of booking had been done through the system.

As well as customers can register to the system and they are able to view or search for events and services and they can view the organizational portfolios and choose the required and their preferred services, as the next step customers can build a conversation with the organization about customer needs and requirements through chat and appointments. So if the requirements are accepted by both service providers and customers then as the next level of the procedure, customers will be facilitated to book their needed services as per their schedule. And after the event concluded both customers and service providers have to update the status of the bookings to the system whether the booked event has been completed or cancelled. In addition, customers are able to give ratings and feedbacks which can viewed by the users of the system.

Admin is to manage the overall system and the users. He is the ones who can have a track on the events and services. And admin will be handling the service charges (subscription payments) which are paid by the service providers. He has a responsibility to update the business policies to secure the system and users. As a business purpose admin can view and generate needed reports to make important decisions about the future and current progresses.

# The scope of the project

## In-Scope

### Users (possible actors) of the system:

* Admin
* Customers
* Service Providers

### Main functionalities of the system:

**Admin**

* View past and pending events
* View and remove (if needed) customers
* View and blacklist (if needed) service providers
* View feedback and ratings
* Able to generate reports
* Update business policies
* Manage service charges
* View and answer the questions asked in Contact Us page

**Customers**

* Register
* Manage customer account details
* Customers can search events
* Search for services
* Make bookings for services
* Make appointments with the service providers
* Chat with the service providers
* Update the status of the event service
* Provide ratings and feedbacks
* Use contact Us page for questioning

**Service providers**

* Register to the system
* Manage account
* Create and manage organizational portfolios
* Manage and schedule appointments
* View and confirm bookings
* Chat with the customers
* Update the service status
* View ratings and Feedback
* Pay service charges
* Use contact Us page for questioning

## Out of scope

* Mobile application
* Trilingual interface
* The system does not cater for monetary transactions for bookings since we are getting a service charge from the service providers.

# Quality attribute identification

Quality Attributes are important inputs into the Architecture of a system and provide the criteria that define what the stakeholder's expectations are about how well the system will operate. These discovered attributes will cover the major areas of the standard software quality model.

***Performance***

When considering the performance, the system that we are developing is a web based system. So when requesting a page and getting back the response form from the server is much depends on the network connectivity of the user. The platform that we are developing is progressive web application so the pages get loaded faster and in a smooth way on any sort of device. Furthermore, to improve the performance the site caching and server caching are enabled, where the response time can be reduced for both client side and server side. To avoid efficiency getting degraded will be made while establish new connections for every single file they want to transfer, Keep Alive on the web server is enabled. Performance of the back-end database is achieved by using indexes and passing variables in queries.

***User-friendliness***

User interfaces will be less complex and easy to use by the users and the colour combinations will be tested using [Coolors.co](file:///C:\Users\hp\Pictures\Coolors.co) to have a good user experience.

***Usability***

The user interface will be user friendly where the user doesn’t need a high adequate knowledge on handling the website. We thought of increasing the usability of our website by having a good responsive interface. We are here to build interfaces in a way that they are simple and easily accessible. The usability components like Learnability, Efficiency, Memorability, Errors and Satisfaction are also achieved by performing the following tasks,

* **Learnability**

When a user enters the system, he could easily find out the places where he could fulfil their requirements. This is achieved by making the interface user-friendly as well as making it less complex. Also, we can add google maps to see the location of the service providers easily. The interfaces are design in a standard way where the position of the buttons that are common to a website are placed as it is.

* **Maintainability**

Once a user enters the page, he could identify all the functions easily, this is achieved by making the interfaces simple and efficient handling. So that once a user logged into the system, they don’t need to remember the places of the functions available. And the system components are trouble free to handle.

* **Errors**

User errors are notified clearly and in an understandable way. There will be a Contact us section which will help a user in all ways. The questions posted will be answered and user complaints on the website will be escalated immediately by the admin.

***Modifiability***

Achieving modifiability strongly relies on how the coding part is done and it should be achieved in a way that when another programmer is assigned to maintain the system and able to understand the system which will be effortless and convenient to modify the system. To achieve modifiability in the development phase, comments are used inside the program to make it more understandable and the names of the variables and other functions are defining meaningfully. Development is basically done using a component-based architecture where each component behaves independently and can be modified independently. We are using RESTful API and the API is test using the Postman software and the front-end is developed individually. This architecture will give the website a high-level modifiability.

***Security***

Since the website contains a user’s sensitive data, we are about to implement following techniques to improve our website security. The database access is only given to a certain number of authorized users (Admin of the Entero system). Username and passwords are not stored in a pure form in the database these details exist in an encrypted format. There will be a payment module for subscription payment that a high security payment gateway is used and the credentials and card details are saved according to the customer’s preference and those saved details are saved in a secure way. For demonstration purpose though a local host is used and deployment is done is Heroku. Maintaining a referential integrity in the database to prevent improper modification.

***Testability***

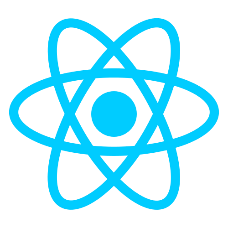
As we are following iterative waterfall method during the development phase each and every unit is tested independently (Unit testing). Front-end is tested separately and for back-end we are use RESTful API and the API is tested using Postman. Component wise testing is also done to make sure the communication between the components are well established. After all components are developed system testing will be done to make sure that the whole system works as expected. Regression testing is done to make sure that the relevant error messages are displayed correctly. And finally, acceptance testing is done during the deployment period.

***Availability***

Website will have its maximum availability during the run time where a user can use the website 24/7. Users are pre notified if there is a maintenance hold in the website. Where the system downfalls are properly notified with the time taken for recovery.

# Tentative technologies

***Front-End: React JS with Tailwind CSS***



React JS is quicker in DOM manipulation by using a virtual DOM and also it supports reusable component so that it makes the development much easier and faster. React JS provides already tested, full-featured and responsive component and mainly by using React JS we can develop dynamically changing components. Because of its virtual DOM implementation and rendering optimizations, React JS outperforms Angular. It’s also simple to switch between React versions; unlike Angular, you don’t have to install updates one by one. The learning curve is short and development is faster than other front-end technologies and also when compared with Angular JS, React JS has a strong community support as this is a learning and developing project community support is much more required.

Node package manager provides the tailwind package where we can easily install it to the application and also Tailwind CSS has a good documentation compared to other front-end CSS technologies. Tailwind CSS has a good compatibility with the React Application as we install using the Node Package Manager. Tailwind CSS uses a set of utility classes to create a neat UI with more flexibility and uniqueness. So, we have chosen to use Tailwind CSS in our system.

***Database: PostgreSQL and MongoDB***

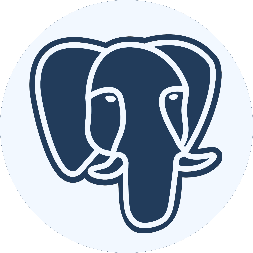
We have decided to use PostgreSQL and MongoDB in our system. Where we use PostgreSQL as our main database while MongoDB as a database for the chat component.

* ***MongoDB***



We have decided to use MongoDB for our chat component as we know that MongoDB is write efficient so when considering a chat application, it should be real time when delivering message and there will be a large load of message getting loaded so we have decided to use socketio with node to have a fast deliver of messages and at the same time it gets stored in the database. Also, for a chat component writing should be fast and as mentioned before MongoDB is write efficient and MongoDB supports rapidly changing, multi-structured data so using MongoDB will be more suitable.

* ***PostgreSQL***

******

As we are building an event management system, we can identify relationship so we can use a relational database. PostgreSQL is highly reliable, stable, scalable and secure system. The main reason to select PostgreSQL as our database is because in future, we are trying to grow this application to an enterprise level so, as PostgreSQL offers more customization and more advanced features like in-database functions and user-defined data types this will provide more adaptability and flexibility in future developments. Among MySQL and PostgreSQL, PostgreSQL is more efficient as PostgreSQL is an object-relational database, while MySQL is purely relational. PostgreSQL uses Multi-version Concurrency Control that works best for high INSERT and SELECT workloads. And mainly as both PostgreSQL and MySQL have similar queries it would be advantageous using PostgreSQL to have a technological exposure as we have a prior knowledge in MySQL.

***Middleware Architecture (Web Application): Express JS***

******

Express JS is a free and open source web application framework for Node JS. Express is fully based on java script it becomes easier for secure and faster development. Express derives various features from Node JS, and one of them is the non-blocking servers that can handle user requests better. In the project we have chosen to use RESTful APIs using the Express JS which is more compatible and fast to code as it hides most of the complexity in backend. As a result, it is easier for us to create easily scalable web apps. And also, as we are using PostgreSQL and well as MongoDB it supports both the technologies very well. Mainly Express JS is easy to configured and customize.

***Back-End: Node JS***

******

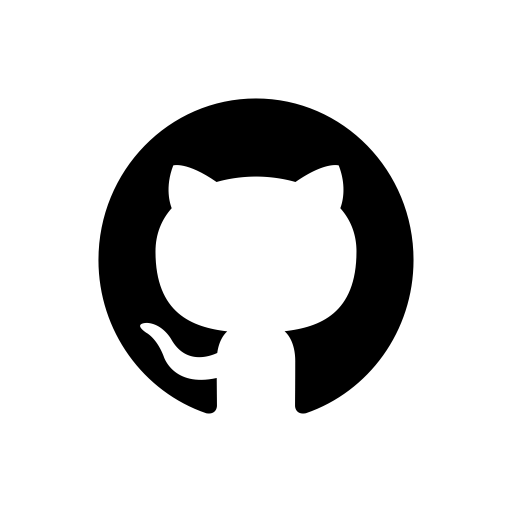
Considering the scalability, NodeJS can be used to manage the request and responses easily. Considering the technology, Node.js is an open-source server environment. Node.js uses JavaScript on the server side allows coders to write server-side code in JavaScript, along with JavaScript code on the front-end with absolute ease is data type conversions like using JSON format but when considering Spring boot, the compatibility is less compared to Node JS. Node JS can generate dynamic page content.

When considering the spring boot, the compilation time takes a considerable time where each time the code changes it need to be recompiled and that will be a bottle neck but in Node JS the code is ready to run so it is the more advantageous.

Node JS can create, open, read, write, delete, and close files on the server and it can add,

delete, modify data in the database. Therefore, Node JS is very convenient with the database model that we are going to use.

***Version Controlling – GitHub***

******

***Project management tool – Trello***

******

***Collaboration – Zoom***



***Integrated development environment (IDE) – Visual Studio Code***

******

# Feasibility study

We have conducted the following feasibility studies about our system.

* Technical feasibility
* Operational feasibility
* Economic feasibility
* Schedule feasibility
* Legal and Ethical feasibility

## Technical feasibility

The system we expected to design is a web based software. We are planning to develop it with for the front end React JS with tailwind while for the back end Express JS as a middleware. Databases for the system are PostgreSQL and MongoDB. Also we will be using NodeJS as our web server environment to test the system. These are free accessible technologies. As for hosting this system Heroku will be used and also to share source code we will be using GitHub.

Online collaboration tools such as Google drive, Google docs, Trello, Zoom,

WhatsApp are also used to collaborate with team members during times when the members cannot physically meet. The above-mentioned technologies are readily available for development, and our team is willing to enhance our project through learning skills on certain technologies that we are not experienced in.

With the use of these technologies, our main aim is to build an effective and efficient website to provide users a platform to deal with events and services. From this website we are determined to make users comfortable and secure to communicate and book. The proposed solution is feasible, the technologies are free, and as a team,

we are eager to improve our skills and understanding in this scope in order to match

the system's criteria. As a result, our website is theoretically and technically feasible.

## Operational feasibility

Entero is a website-based application which is easy to learn and it will require a very short time to learn the operations and functionalities. The graphical user interface would be developed in a very user-friendly manner and could be easily operated. Customers are able to find the services that they want from a one place. When this system is achieved with complete automation, all the user related information will be managed well and it would provide maximum ease in usage to all users.

The chatting system would assist the users to create a community between organization. Not a lot of training would be needed for people with minimal computer knowledge. The admin would get continuous reports which would help in taking decisions to plan future tasks of the company. Customers are able to find organizational services and manage their event services easily. The services provided by this system would be flexible and expandable.

## Economic feasibility

***Development cost***

* Since this is a university based project there are no development cost.
* There will be no consultation payment even though we have a supervisor and co-supervisor.
* No software cost will be incurred as every tool used to develop a system is

open-source software tools.

***Operational cost***

* This web application is hosted on Heroku free of charge.

Therefore, it can be stated that the system will bear next to zero costs during

development phases, and hence it is economically feasible.

## Schedule feasibility

The development of the website should be completed by the end of the academic

first semester. Since it is allowed to use frameworks, the development process will be fast and efficient. We are using the iterative waterfall method for development process where we finish the documentation and go for the development phase. We will divide workload and complete the project. This 15 weeks’ timeline is believed to be sufficient to complete the project.

|  |  |
| --- | --- |
| *Work time allocation* | |
| No of weeks | 15 |
| Working hours per day on the project(weekdays) | 4 hrs |
| Working hours per day on the project(weekend) | 5 hrs |
| Time allocated on the project per week by a single member | 9 hrs |
| Total time allocated on the project by a single member | 135 hrs |
| Total time allocated by the group (5 members) for the project | 675 hrs |

## Legal and Ethical feasibility

This system is designed only for Event management and service booking. This

system does not violate rules and regulations under the law of Sri Lanka.

* Sensitive data of users that collect by the system will not disclose to unauthorized viewers.
* User information such as organization portfolio, user information, contact

numbers, etc. will not be sold to any third party.

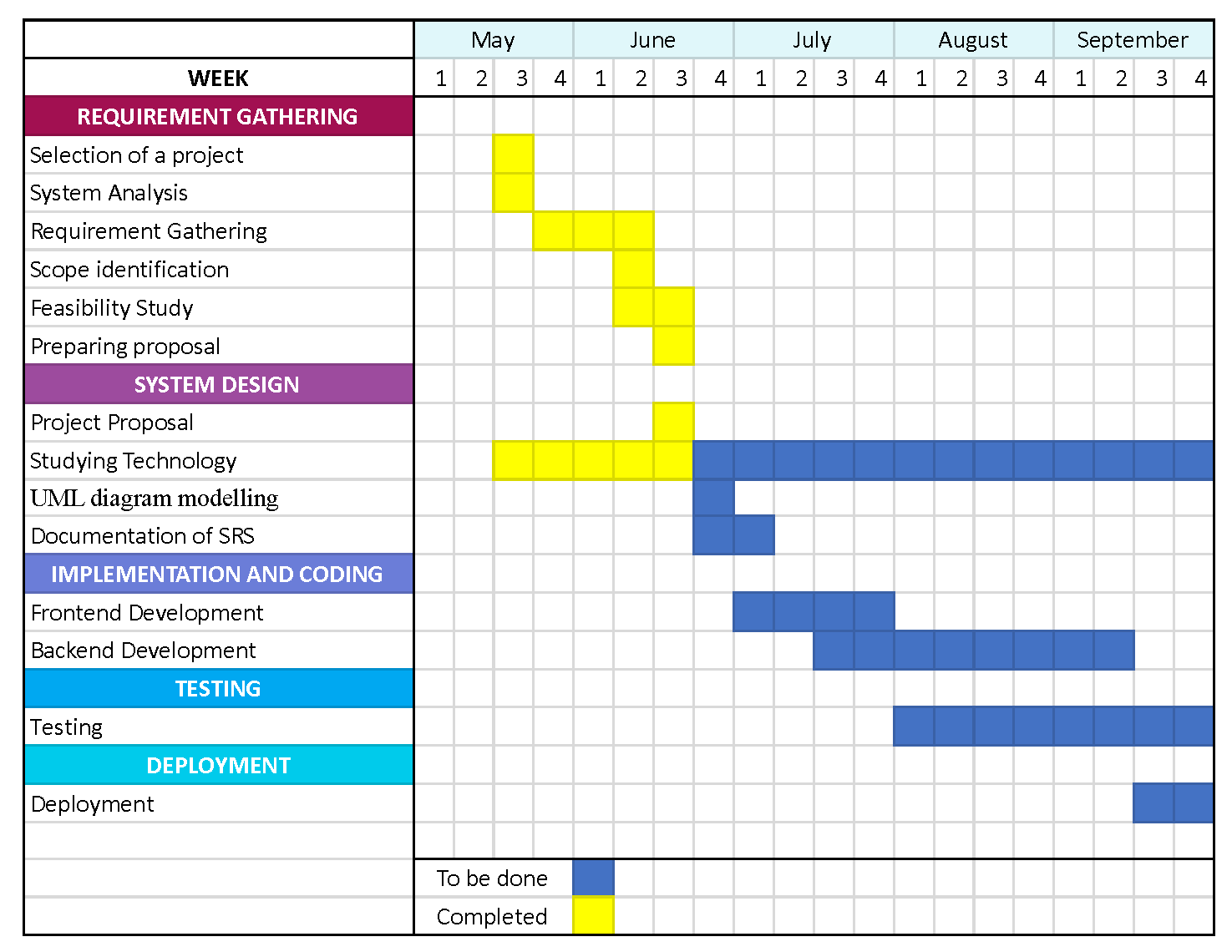
The above facts state this system is legally feasible

# Main deliverables of the system

* Complete working software and source code of the web based application.
* Complete Software Requirement Specification
* User manual
* Administrators manual together with deployment instructions
* Software License: Proprietary software

# The project plan

We plan to start the project implementation on the first week of July and also we are testing developed modules partially to modules developing. We plan to finish the complete system in September 2022. The development methodology used will be the iterative waterfall methodology. Requirements might be changed later. Moreover, you can see our Gantt chart below and get an idea of how we plan to finish it.



# References

[1]"Getting Started – React", [Reactjs.org](mailto:Reactjs.org), 2022. [Online]. Available[:.https://reactjs.org/docs/getting-started.html](mailto:https://reactjs.org/docs/getting-started.html) [Accessed: 18- Jun- 2022]

[2]"Install Tailwind CSS with Create React App - Tailwind CSS", [Tailwindcss.com](mailto:Tailwindcss.com), 2022. [Online]. Available[:.https://tailwindcss.com/docs/guides/create-react-app](mailto:https://tailwindcss.com/docs/guides/create-react-app) [Accessed: 18- Jun- 2022]

[3]"Installing Express", [Expressjs.com](mailto:Expressjs.com), 2022. [Online]. Available[:.https://expressjs.com/en/starter/installing.html](mailto:https://expressjs.com/en/starter/installing.html) [Accessed: 18- Jun- 2022]

[4]"PostgreSQL: Documentation", [Postgresql.org](mailto:Postgresql.org), 2022. [Online]. Available[:.https://www.postgresql.org/docs/](mailto:https://www.postgresql.org/docs/) [Accessed: 18- Jun- 2022]

[5]"Wedding Planning Software - Online Wedding Planner for Professionals", [Planningpod.com](mailto:Planningpod.com), 2022. [Online]. Available[:.https://www.planningpod.com/wedding-event-planning-software.cfm](https://www.planningpod.com/wedding-event-planning-software.cfm) [Accessed: 18- Jun- 2022]

[6]"Top 100 Birthday Party Planners, Best Event Organisers | Sulekha", [Sulekha.com](file:///C:\Users\hp\Desktop\Sulekha.com), 2022. [Online]. Available[:.https://www.sulekha.com/birthday-party-organisers/](https://www.sulekha.com/birthday-party-organisers/) [Accessed: 18- Jun- 2022]

# Declaration

We as members of the project titled ‘Entero – Event Management System’, certify that we will carry out this project according to guidelines provided by the coordinators and supervisors of the course as well as we will not incorporate, without acknowledgement, any material previously submitted for a degree or diploma in any university. To the best of our knowledge and belief, the project work will not contain any material previously published or written by another person or ourselves except where due reference is made in the text of appropriate places.

|  |  |
| --- | --- |
| ***Name*** | ***Signature*** |
| S. Kavishan |  |
| S. A. R. P Athauda |  |
| B.Y. M. Fernando |  |
| K.K. S. Punsara |  |
| M.S.M. Shakir |  |

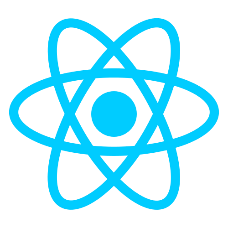
**SCS 3214 / IS 3113: Group Project II - 2022**

**Group No: G32**

**Project Name: Entero (Event Management System)**

***Technology Justification***

***Front-End: React JS with Tailwind CSS***



React JS is quicker in DOM manipulation by using a virtual DOM and also it supports reusable component so that it makes the development much easier and faster. React JS provides already tested, full-featured and responsive component and mainly by using React JS we can develop dynamically changing components. Because of its virtual DOM implementation and rendering optimizations, React JS outperforms Angular. It’s also simple to switch between React versions; unlike Angular, you don’t have to install updates one by one. The learning curve is short and development is faster than other front-end technologies and also when compared with Angular JS, React JS has a strong community support as this is a learning and developing project community support is much more required.

Node package manager provides the tailwind package where we can easily install it to the application and also Tailwind CSS has a good documentation compared to other front-end CSS technologies. Tailwind CSS has a good compatibility with the React Application as we install using the Node Package Manager. Tailwind CSS uses a set of utility classes to create a neat UI with more flexibility and uniqueness. So, we have chosen to use Tailwind CSS in our system.

***Database: PostgreSQL and MongoDB***

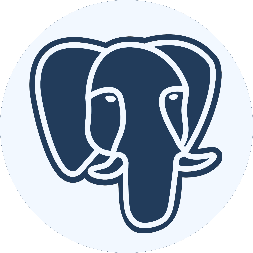
We have decided to use PostgreSQL and MongoDB in our system. Where we use PostgreSQL as our main database while MongoDB as a database for the chat component.

* ***MongoDB***



We have decided to use MongoDB for our chat component as we know that MongoDB is write efficient so when considering a chat application, it should be real time when delivering message and there will be a large load of message getting loaded so we have decided to use socketio with node to have a fast deliver of messages and at the same time it gets stored in the database. Also, for a chat component writing should be fast and as mentioned before MongoDB is write efficient and MongoDB supports rapidly changing, multi-structured data so using MongoDB will be more suitable.

* ***PostgreSQL***

******

As we are building an event management system, we can identify relationship so we can use a relational database. PostgreSQL is highly reliable, stable, scalable and secure system. The main reason to select PostgreSQL as our database is because in future, we are trying to grow this application to an enterprise level so, as PostgreSQL offers more customization and more advanced features like in-database functions and user-defined data types this will provide more adaptability and flexibility in future developments. Among MySQL and PostgreSQL, PostgreSQL is more efficient as PostgreSQL is an object-relational database, while MySQL is purely relational. PostgreSQL uses Multi-Version Concurrency Control that works best for high INSERT and SELECT workloads. And mainly as both PostgreSQL and MySQL have similar queries it would be advantageous using PostgreSQL to have a technological exposure as we have a prior knowledge in MySQL.

***Middleware Architecture (Web Application): Express JS***

******

Express JS is a free and open source web application framework for Node JS. Express is fully based on java script it becomes easier for secure and faster development. Express derives various features from Node JS, and one of them is the non-blocking servers that can handle user requests better. In the project we have chosen to use RESTful APIs using the Express JS which is more compatible and fast to code as it hides most of the complexity in backend. As a result, it is easier for us to create easily scalable web apps. And also, as we are using PostgreSQL and well as MongoDB it supports both the technologies very well. Mainly Express JS is easy to configured and customize.

***Back-End: Node JS***

******

Considering the scalability, NodeJS can be used to manage the request and responses easily. Considering the technology, Node.js is an open-source server environment. Node.js uses JavaScript on the server side allows coders to write server-side code in JavaScript, along with JavaScript code on the front-end with absolute ease is data type conversions like using JSON format but when considering Spring boot, the compatibility is less compared to Node JS. Node JS can generate dynamic page content.

When considering the spring boot, the compilation time takes a considerable time where each time the code changes it need to be recompiled and that will be a bottle neck but in Node JS the code is ready to run so it is the more advantageous.

Node JS can create, open, read, write, delete, and close files on the server and it can add,

delete, modify data in the database. Therefore, Node JS is very convenient with the database model that we are going to use.