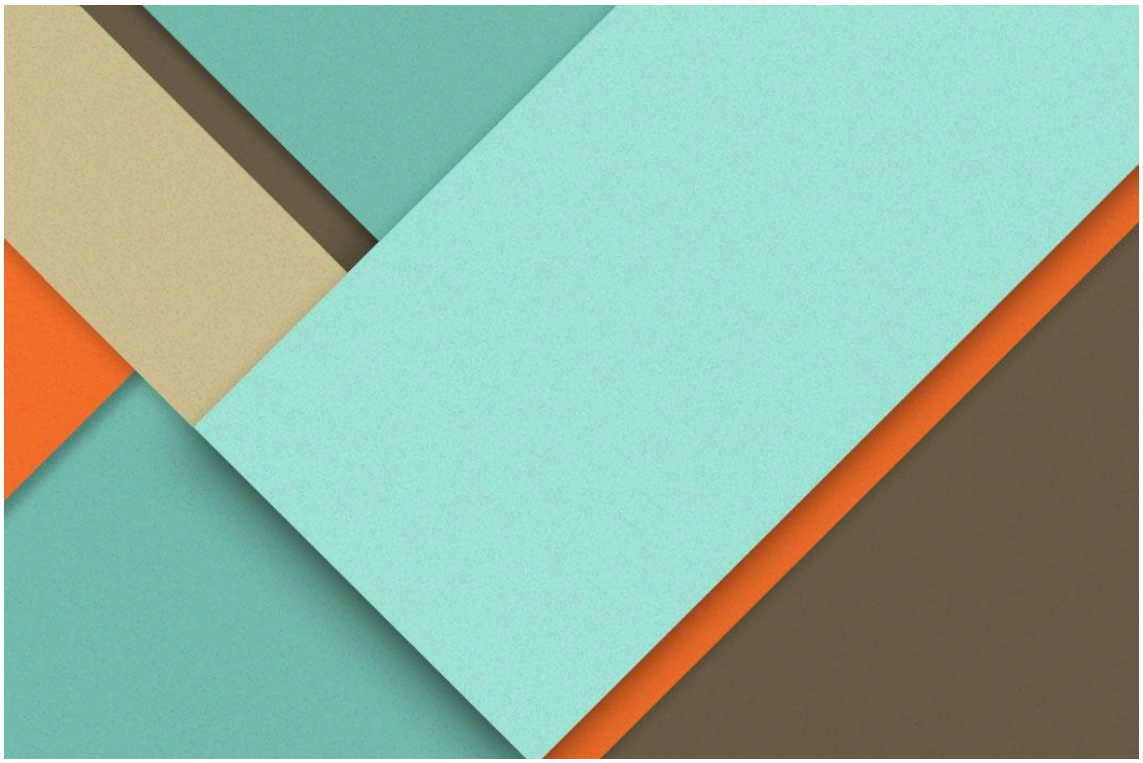


Project 7 - Event Management System Project Report

COMP3900 - Computer Science Project

—

Due Date: Friday 4th of August, 2023 @ 9pm

Team 3900W16Aeventlite

Ethan Lam, ethan.lam@student.unsw.edu.au, z5205328, SCRUM Master

Eddy Gu, z5309049@ad.unsw.edu.au, z5309049, Frontend

Minseok Kim, z5364523@ad.unsw.edu.au z5364523, Backend, Database, Frontend Lead

Mihika Goyal, z5361077@ad.unsw.edu.au, z5361077, Backend

1.0 Table of contents

2.0 Overview

Event organisation and attendance are a social norm in today's world, where people are spoilt for choices. In 2019, 43.7 million people attended more than 484,382 events across Australia (Business Events Council of Australia, 2021). Nowadays, attending events is a way to exercise one's interests and socialise when a person has free time. There are numerous companies who want to host a variety of events and use comprehensive platforms to do so. These platforms aid event organisation by providing functions such as advertisement, ticket sales, event management and so on. They also aim to improve customer experience by collating a range of events in one concise platform where customers can browse through events conveniently. With events being in demand, the need for event platforms has grown significantly. There are numerous ticket selling companies in Australia, including Eventbrite, Ticketmaster and Ticketek (audience republic, 2023) with Ticketmaster being ranked number 1 as the most popular ticketing platform (similarweb, 2023).

With the rapidly growing trend of events and social gatherings, Team EventLite desired to create a comprehensive event management platform which placed emphasis on user-friendliness and personalisation whilst also integrating the developing technology of AI. As such, the team has invented The Event Management System, a social web application designed to organise and promote several types of events to a wide audience, similar to Eventbrite or Ticketek although with additional features. This project was designed using NodeJS for the business Layer, ReactJS for the Client Layer, and PostGresQL for the database layer. The functionalities integrated into this project were chosen because they were common features implemented across all competitive platforms and due to popular demand. They allow users to create profiles and events, join, search, share, edit, recommend and review events, interact with a chatbot for faster event searching, and customise profile pages for a unique, user-friendly experience.

3.0 Description of functionalities

3.1 Authentication System

3.1.1 Description

The Authentication System is a collaboration of features that allows a user to create an account by specifying their username, password, email address, date of birth, first name, last name, gender, credit card details, biography, and whether they are a student. With an account, the user will then be able to create and edit their events as well as join and review other events. The account will also track what events the user has hosted and joined, however, it will not track events that the user got a refund for or have been cancelled by the event host.

3.1.2 Project objectives/User stories

- YDCG-21: As a general user, I want to create my own account so that I can customise my own profile page.
- YDCG-41: As a general user, I want to save my payment details into my account so that I don't have to repeatedly add my payment details to sign up for an event.
- YDCG-42: As a user, I want my profile to track the type of events that I have gone to so that I can observe the events that I like the most.
- YDCG-43: As a user, I want to create a biography of myself in my profile to add personality to my account and so that people can know me a bit better.

3.2 Event Creation System

3.2.1 Description

The Event Creation System consists of features that allow a user to create an event. With this system, an event creator/host is able to specify the event's name, description, date, time, location, tags, price, max capacity and the age restriction of the event. After an event is created, the creator will then have options to edit their event or cancel it. With editing the event, the creator can change any details of the event except the price, and with the cancelling of an event, when the event is cancelled, all the customers who joined will automatically receive a refund.

3.2.2 Project objectives/User stories

- YDCG-5: As an event host, I want to be able to categorise my event to target specific audiences who are interested in my events.

- YDCG-11: As an event manager, I want to be able to track attendance to ensure the number of participants does not somehow go over the intended amount.
- YDCG-16: As an event participant, I want to scan a qr code to easily access an event.
- YDCG-32: As an event organiser, I want to be able to set tickets at different prices.

3.3 Recommendation System

3.3.1 Description

When a user opens the main page of the website, they are greeted with a list of events. This list of events is controlled by the Recommendation System which displays events based on the user's activity, specifically what events the user previously attended. Furthermore, the user can control what events are recommended based on price, capacity or types of events the user has not joined before. It also takes into consideration the tags of the events that the user has joined before and the age restriction of the events. However, in the instance where a user has not joined any events and therefore does not have any tags associated with them, they get recommended all the events that exist in the database.

3.3.2 Project objectives/User stories

- YDCG-3: As an event browser, I want to know what the most highly anticipated upcoming events are because they usually are the best.
- YDCG-10: As an event organiser, I want my events to be recommended to a range of different audiences to maximise exposure.
- YDCG-13: As a student, I want to have free/discounted events recommended to me as I have no income.
- YDCG-14: As a parent, I want my child to see child friendly events.

3.4 Seating Arrangement System

3.4.1 Description

The Seating Arrangement System is a function that allows a customer to choose which seats they want to purchase for the event. In the project's current version, the prices of the seats are split evenly amongst the number of seats. For

example, if an event has a max capacity of 50 and there are 2 separate prices, $10 and $20, then the system will automatically assign 25 seats to be $10 and the other 25 seats $20.

3.4.2 Project objectives/User stories

- YDCG-33: As an event creator, I want to be able to set a seating arrangement for my event.
- YDCG-34: As an event customer, I want to be able to choose a seat for the event.
- YDCG-35: As an event organiser, I want to be able to set ticket prices for each individual seat.
- YDCG-36: As an event participant, I want to be able to see the price of individual seats before I pay.

3.5 Star Rating Review System

3.5.1 Description

The Star Rating Review System is a collection of features that allows an attendee of an event to create reviews for an event and reply to other reviews as well. In addition to leaving a message, the user can also leave a star rating score between 1 and 5. This system will also calculate the average score from all the other reviews and display it as the overall score for the event. Additionally, the host of the event can also reply to the reviews.

3.5.2 Project objectives/User stories

- YDCG-19: As an event participant, I want to be able to review events to share my opinion.
- YDCG-37: As an event reviewer, I would like to post a score or point system for the event to leave my overall opinion.
- YDCG-38: As an event host, I want to be able to leave replies to reviews.
- YDCG-39: As an event organiser, I want to be able to see an overall score of my event to see how well people enjoyed it.

3.6 Subscription System

3.6.1 Description

The Subscription System is a function that has two components to it. The first allows a user to simply like an event without having to buy tickets for it. This system is focused on users that are interested in events but are not fully committed to buying tickets just yet. The second component of the subscription system is where updates are shared with users who have joined an event. This consists of the user receiving an email once they join an event, when an event is edited or if an event is cancelled.

3.6.2 Project objectives/User stories

- YDCG-6: As an event participant, I want to receive updates and news about the events that I am interested in/participating to remain informed of changes.
- YDCG-8: As a browser, I want to subscribe to an event before deciding on whether or not I want to participate in it just in case I change my mind about the event.
- YDCG-17: As an event host, I want to be able to send an authentication code in the form of a qr code to verify the correct customers.
- YDCG-40: As an event host, if I cancel my event, I want to be able to communicate that to participants.

3.7 Search and Filter System

3.7.1 Description

The Search and Filter System is a function that allows a user to search for events. When the user inputs a search phrase, the system collects events which have a similar phrase in either the event's name, location or description. This system also has a filter feature that allows the user to filter their searches based on certain tags of the events. Additionally, there is an option for the user to sort the results of their search in either ascending or descending order of the event dates.

3.7.2 Project objectives/User stories

- YDCG-4: As an event browser, I want to be able to view events in different orders
- YDCG-9: As a thrill seeker, I want to explore new events that I have not previously been interested in to diversify my search.

- YDCG-12: As someone who cannot travel far, I want to see events in my area, as it will be more convenient to attend.
- YDCG-20: As an event searcher, I want to filter out types of events that I am not really interested in.

3.8 Refund system

3.8.1 Description

The Refund System handles the returning of credit to the customer from the organiser of the event. If an event organiser decides to completely cancel their event, the event is removed from the website and all the attendees will receive an email notifying them of the cancellation. If an attendee decides that they want to refund their ticket purchases, then the system will calculate the amount that they are refunded and refund the credit back to the credit card they provided in their account.

3.8.2 Project objectives/User stories

- YDCG-44: As an event participant, I want to be able to refund my purchases in case I cannot attend an event.
- YDCG-45: As an event host, to automatically send refunds to all customers when my event is cancelled.
- YDCG-46: As an event creator, when someone refunds their ticket for my event, I want a free spot to appear for any other interested customer to join.
- YDCG-47: As an event organiser, I want customers to only be able to refund their tickets when there is more than 7 days before the event date to remove any last minute refunds.

3.9 Chatbot system

3.9.1 Description

The chatbot is a popup that acts as a tool to help users streamline their experience using the application. It creates an interface to chat with the user and allows users to be able to more efficiently complete tasks such as searching for events.

3.9.2 Project objectives/User stories

- YDCG-15: As an event browser, I want to communicate with a chat bot to reduce downtime looking for events that I am interested in.
- YDCG-48: As an event browser, I want a chatbot to recommend me events based on the interests that I give it.
- YDCG-53: As an event seeker, I want a chatbot to act and respond like a normal person to make the interaction authentic rather than robotic.
- YDCG: 54: As an event seeker, I want the chatbot to look up events that suit me so that I do not have to click and search through events manually myself.

3.10 Avatar Customisation system

3.10.1 Description

The Avatar Customisation System is a tool which allows a user to upload and edit a profile image. The image editing tool allows the user to add a dynamic border around their picture and create sketches within the image using tools similar to microsoft paint.

3.10.2 Project objectives/User stories

- YDCG-49: As an artist, I want to be able to add my creativity to my profile picture.
- YDCG-50: As a user, I want to add creative, dynamic profile picture borders to my account to make it more appealing.
- YDCG-51: As a general user, I want to customise my profile picture using pre-formatted/default pictures already provided to me by the website.
- YDCG-52: As a general user, I want to use drawing tools like a paint brush and shapes to personalise my profile picture to add my own creativity.

4.0 Description of third-party functionalities

4.1 Sequelize

4.1.1 Description

Sequelize is an Object-Relational Mapping library for Node.js which works on databases such as PostgreSQL (Sequelize.org, 2023). It simplifies database interactions by allowing us to manipulate database models as JavaScript classes without SQL commands. The classes represent tables and sequelize allows us to

create, delete and update data in a database using a preferred language. Sequelize also offers functions to create seed files to initiate dummy queries into the database and migration files to change parameters on certain tables.

4.1.2 Use

Sequelize was used to create the database tables. This includes the User, Event and Tags table which were all represented as classes in javascript and the instances represented rows. This was used to add data to the tables in the cases such as a new user being registered, edit data when an event was edited or delete data when an event was deleted.

4.2 Bcrypt

4.2.1 Description

Bcrypt is a function that enables password-hashing which allows passwords to be stored in a secure manner (npm, 2023). When a user creates a password, it is stored as a hash which makes it harder to guess the original password, and reduces the possibility of reverse-engineering. A random salt and specified work factor is also generated along with the hash value and is then retrieved when a user attempts to log in.

4.2.2 Uses

Bcrypt was used to hash the password that the user created when registering or editing their information. The hash was saved in the database and was retrieved when the user attempted to sign in. If the hash stored in the database matched the hash and salt combined with the entered password, the user would be able to log in successfully.

4.3 Validator

4.3.1 Description

Validator is a library used to validate data through the use of functions that make validation easier across numerous types of data (npm, 2023). This simplifies the process of checking if the users have entered valid values which meet the criteria of the application. It can also be used for the sanitization of data where unwanted characters can be removed.

4.3.2 Uses

Validator was used to check the input that the user provided for fields such as email and credit card number. This was because both these fields have a specific format and would be invalid if not followed. It simplified the process of performing error checks for these user inputs.

4.4 JWT Token

4.4.1 Description

JSON Web Token (JWT) is used for authentication and authorization between two parties in a secure manner (npm, 2023). It has three parts, the head, payload and signature which together form a secure JSON object. This is generated whenever a user logs in and is used to verify a user and extract their information. This is advantageous as the server does not have to keep a track of the user sessions.

4.4.2 Use

JWT was used to generate tokens for a user whenever they registered or logged in. This allowed the user to be verified and helped determine the rights that the user had, such as the ability to edit or delete an event.

4.5 Sendgrid

4.5.1 Description

Sendgrid offers an API for emails which can be integrated into websites and applications to send and handle emails, including automated messages and notifications (docs.sendgrid.com, 2023). It allows a large volume of emails to be delivered and also tracks email analytics such as bounced emails, open rates and delivery rates.

4.5.2 Use

Sendgrid was used in order to achieve the subscription system functionality. The email API was used to automatically send users an email when they joined an event, when a joined event was edited or when a joined event was cancelled. The API was fairly easy to use and understand and also provided email analytics such as number of emails delivered, opened and bounced.

4.6 Material-UI

4.6.1 Description

Material-UI is an open-source user interface framework for React.js. It provides a variety of design patterns and principles which allows visually appealing user interfaces to be created (mui.com, 2023). This includes components such as colours, buttons, app bars and other visual components.

4.6.2 Use

The styling component from Material-UI was used to build a user-friendly interface. This included the text field, button, card, divider, dropdown menu and header. The styling feature allowed us to build the frontend styling without any separate CSS file.

5.0 Implementation challenges

5.1 Lack of expertise on Frontend

One of the major challenges faced was the lack of expertise that all team members had on the frontend. Due to minimal experience, the team members on the frontend had to learn the implementation from the very beginning. This took away time from the actual implementation and caused the frontend falling behind the backend until the final sprint.

5.2 Lack of communication between backend and frontend

The frontend falling behind the backend in sprints 1 and 2 was also due to the lack of communication between the team members working on the backend and team members working on the frontend. This was due to team members not joining the scheduled meetings, not keeping up to date with the messages on the group chat or not using the most updated versions of the code from github. This mainly caused confusion on the frontend side as it was working behind the backend and led to team members not being aware of certain changes or updates in the code. This resulted in some code having to be rewritten or edited which put the frontend behind the decided timeline.

5.3 Hacking

One day before the final demo the amazon database that was being used throughout the term was hacked. All the data had been erased and we were unable to find a way to bring back the tables. This resulted in us having to spend the majority of our time researching how to make a new database instead of being able to work on the final demo or stress testing on the frontend. We were later able to create a new database which was hosted locally, however, numerous errors were encountered during this process. We then also had to spend additional time on repopulating the database with enough users and events to test all the different functionalities. This change in the database was also hard to adjust to as the data was no longer shared amongst all team members and everyone had to work locally. This made testing more challenging at a very crucial time.

5.4 Creating Jest test suites

Creating test suites using Jest and the Supertest package was very difficult to implement for every function on the backend. One reason was because for most of the project's development cycle, the team did not have a dedicated database to test on. That meant that even though the tests may succeed for one person, there was no guarantee that it would succeed for another person later on because the database structure and the dummy queries that were used for testing kept changing. Another reason too was the limited time constraint. From extensive research, supertest was the most recommended NodeJS package for testing backend HTTP requests, however, there was a monumental learning curve. Thus, with the restricted time frame that the team had, it seemed more efficient to use the Requestly API client app to test the functionality of the backend server.

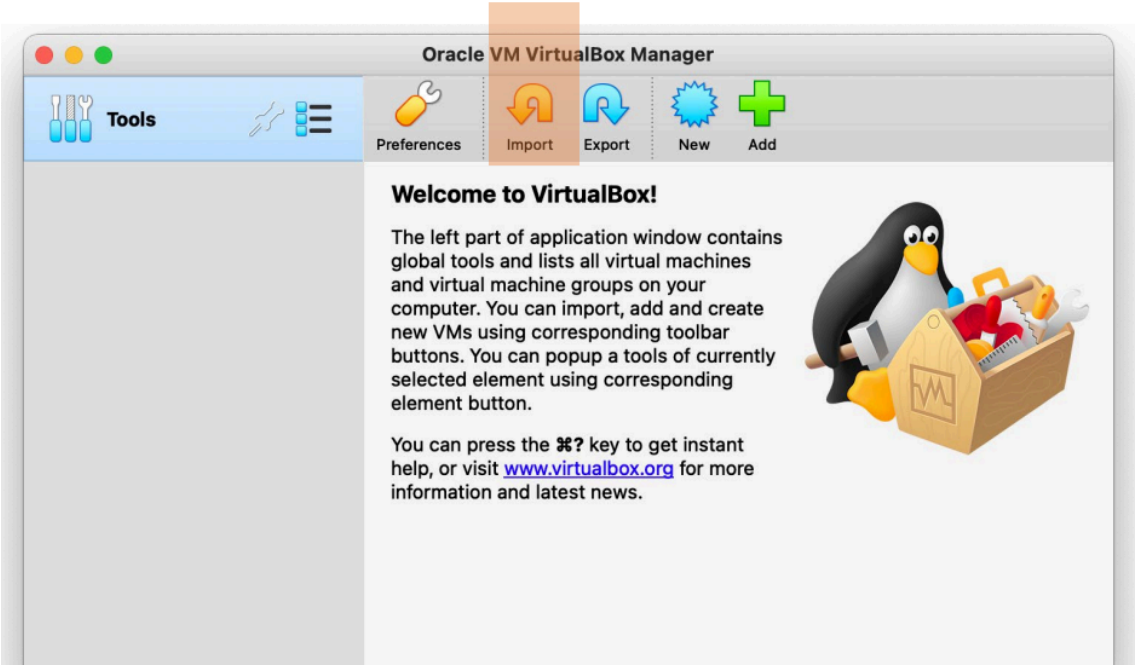
6.0 Installation/User document manual

For this project, the software will operate in the Lubuntu virtual machine on a virtual box.

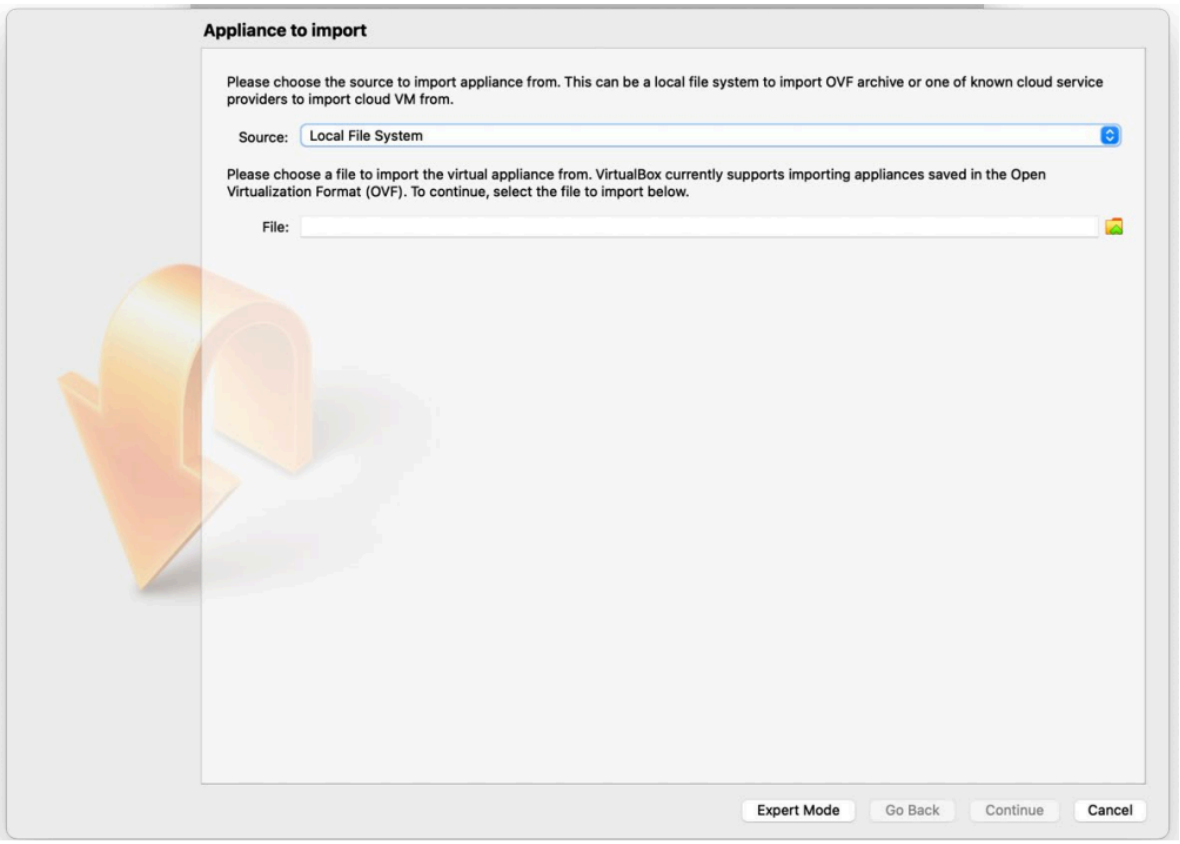
6.1 Installation manual

6.1.1 Installing Lubuntu Virtual machine on virtual box

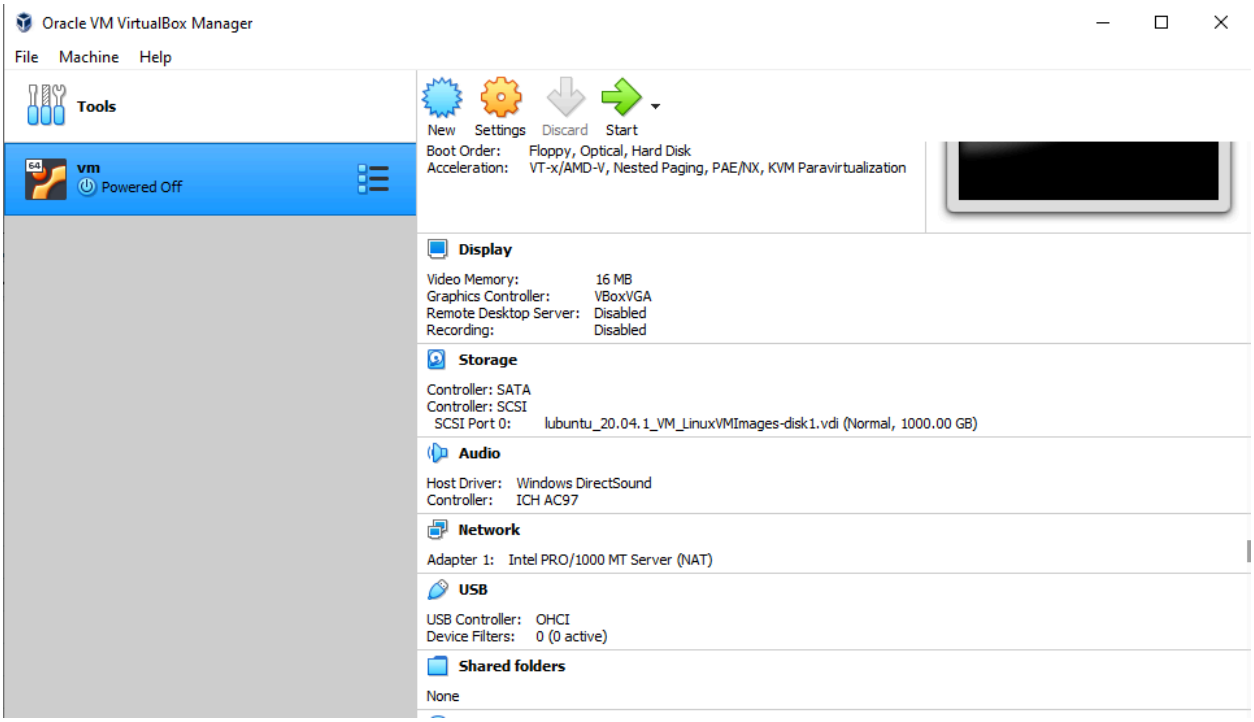
1. Download the Lubuntu virtual machine image using the link below
   - https://sourceforge.net/projects/linuxvmimages/files/VMware/L/lubuntu_20.04.1_VM.zip/download
   - Make sure to install the 6.1.22 version or later.
2. Download the VirtualBox using the link below
   - https://www.virtualbox.org/wiki/Download_Old_Builds_6_1
   - Make sure to install the 6.1.46 version or later.
3. Run the executable file for Lubuntu. After completing the installation steps, the window should appear.



4. Extract the Virtual box zip file that you downloaded from step 2
5. Click on import in the Oracle VM virtual box manager displayed in image 1 and the Appliance to import window should appear.



6. Make sure that the “Local File System” option is selected for the source field. In the file field, click on the folder icon and select the VM image file with the .ovf extension that you downloaded in step 2. After that click continue.
7. After that, the appliance settings window will appear. Leave the settings as is and click on import. After that, you should see the “vm” virtual machine option on the left of the Oracle manager.



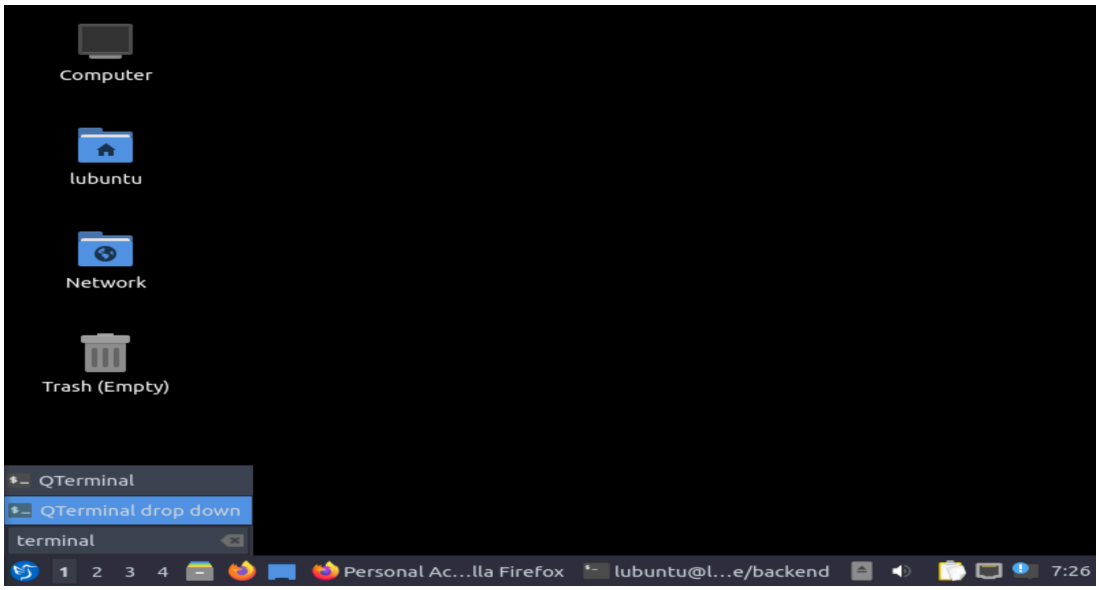
8. Select the “vm” virtual machine and click the green arrow “start” button. Note: if an error appears to activate AMD_V, follow the instructions below to enable SMV mode in the BIOS. Instructions in the link below for ASUS BIOS,
    - https://www.asus.com/support/FAQ/1043992
9. Use the login credentials below to log into the LUBUNTU user,
    - Password: lubuntu

Note: When you first start up your vm, the resolution may be a bit off. If you wish to fix this issue, after starting up your vm for the first time, shutdown the vm and restart your computer. Afterwards, open up the vm again and a window will pop up saying that the vm’s display will be scaled to match your native resolution.

6.1.2 Building the EventLite project

Cloning the github repository

1. In the vm, open a terminal by opening the home tab and searching for “terminal”



2. Then, run the git clone command below using the credentials
   - "git clone https://github.com/unsw-cse-comp3900-9900-23T2/3900W16Aeventlite.git"
   - Username: Ethanlammie
   - Password: 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

```
lubuntu@lubuntu2004:~/Documents$ git clone https://github.com/unsw-cse-comp39
00-9900-23T2/3900W16Aeventlite.git
Cloning into '3900W16Aeventlite'...
Username for 'https://github.com': Ethanlammie
Password for 'https://Ethanlammie@github.com':
remote: Enumerating objects: 8709, done.
remote: Counting objects: 100% (1663/1663), done.
remote: Compressing objects: 100% (545/545), done.
remote: Total 8709 (delta 1162), reused 1543 (delta 1110), pack-reused 7046
Receiving objects: 100% (8709/8709), 6.96 MiB | 6.29 MiB/s, done.
Resolving deltas: 100% (2948/2948), done.
lubuntu@lubuntu2004:~/Documents$ ls
3900W16Aeventlite
```

   - Note: copy and pasting between your local machine and the vm will most likely not be enabled. Therefore, to copy and paste the command line and the password, you will need to have this report open in the vm to copy and paste from the code. Remember, to copy and paste from the terminal, use ctrl+shift+c and ctrl+shift+v.

Building the database

1. Run “sudo apt update” and “sudo apt install postgresql postgresql-contrib”.
    - Password: lubuntu
2. Ensure that the service is started using the command line “sudo systemctl start postgresql.service”
    - Password: lubuntu
3. Change the password for the default postgres user to “postgres”.
    a. Enter the command line “sudo -u postgres psql” which will take you to the postgresql client for the default postgres user
        - Password: lubuntu
    b. Enter “\password” and change the password to “postgres”
    c. Enter “exit” to exit out of the postgresql client.

Building the backend and frontend, and preparing the database

1. Go into the repository in your terminal i.e. “cd 3900W16Aeventlite”
2. Run the command “sudo apt update” and “sudo apt install nodejs npm” to install npm.
    a. Password: lubuntu
3. Update nodeJS to version 20.5.0
    a. First install curl using the command line “sudo snap install curl”
        i. Password: lubuntu
    b. Run the command line “sudo apt-get install build-essential libssl-dev curl git-core”
    c. Run the command line “curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.35.3/install.sh | bash” and after the installation is done, restart your terminal

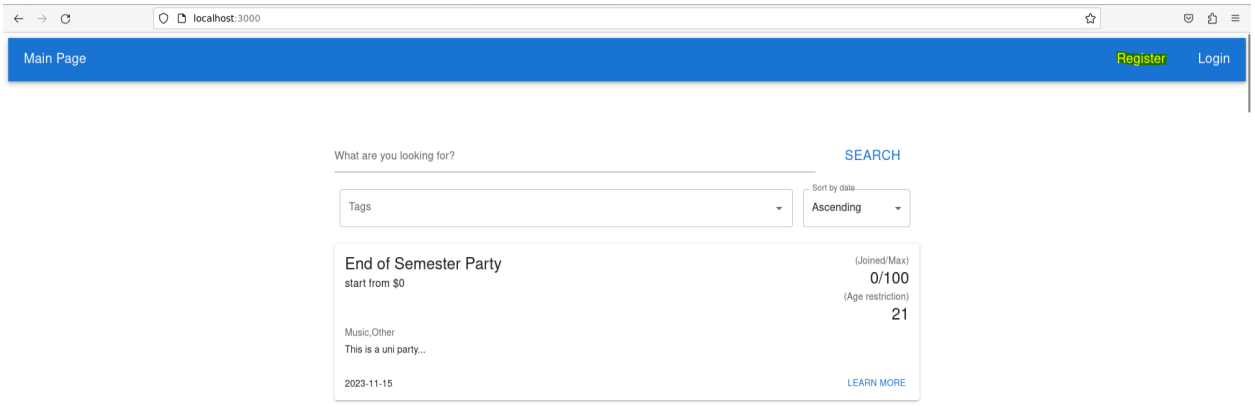
d. Navigate back to the git repository, i.e. “cd 3900W16Aeventlite” and run “nvm install 20.5.0” to install the 20.5.0 version of nodejs.

4. Navigate into the backend folder, i.e. “cd backend” and run “npm install” to install the node packages.
5. Run the command line “npm run dev” to run the backend and initialise the database tables. After that is done, press ctrl+c to stop the backend server.
6. Remaining in the backend folder, go into the database folder “cd database” and run the command “npx sequelize db:seed:all” to initialise users and events.
7. Navigate back into the backend folder, “cd ..” and run “npm run dev” to run the backend server.
8. After that, start a new terminal and navigate to the frontend folder and run “npm install”
9. Run “npm start” to initiate the frontend. Make sure that the backend server is running before initiating the frontend server.

6.2 User manual

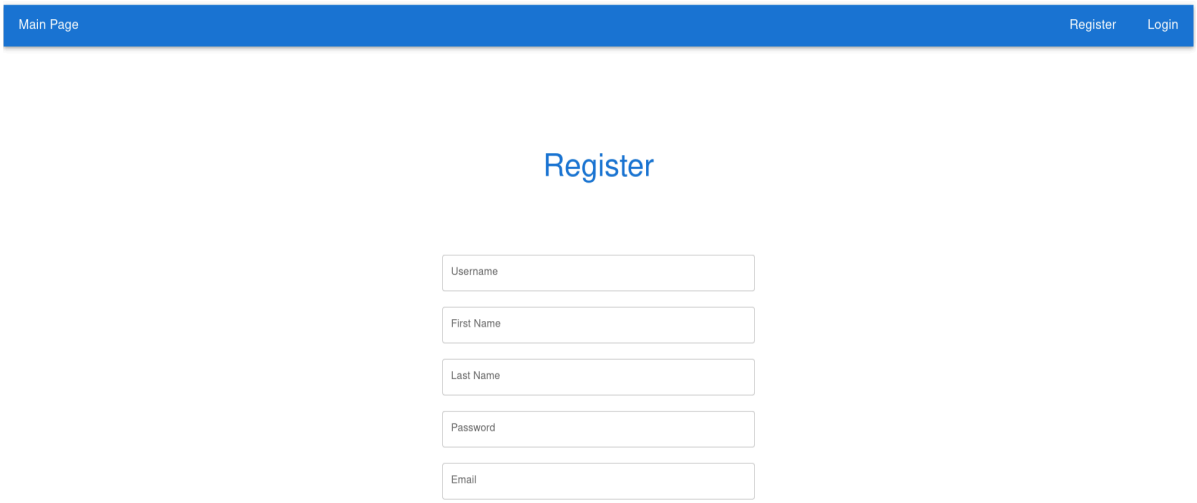
6.2.1 Creating an account

1. After running the backend and frontend, firefox should automatically open with the main page appearing. To create an account, click on the “Register” button on the top right corner of the website.



2. Fill in all the fields in the register page, making sure:
   - The username is 5 characters long or more
   - Password is 6 characters long or more
   - Email is in a valid format
   - Email is not already taken by another user

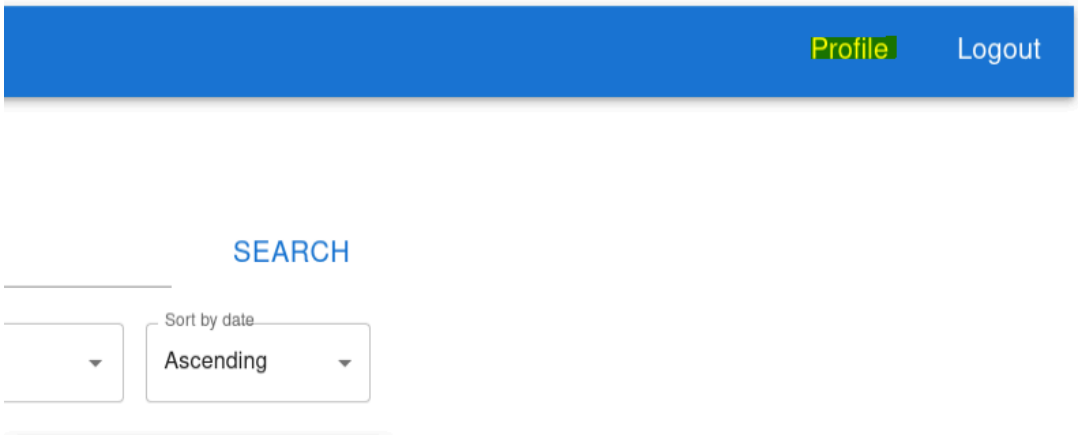
- DOB must be before the current date
- Expiry date for credit card is after the current date
- Credit card number is valid



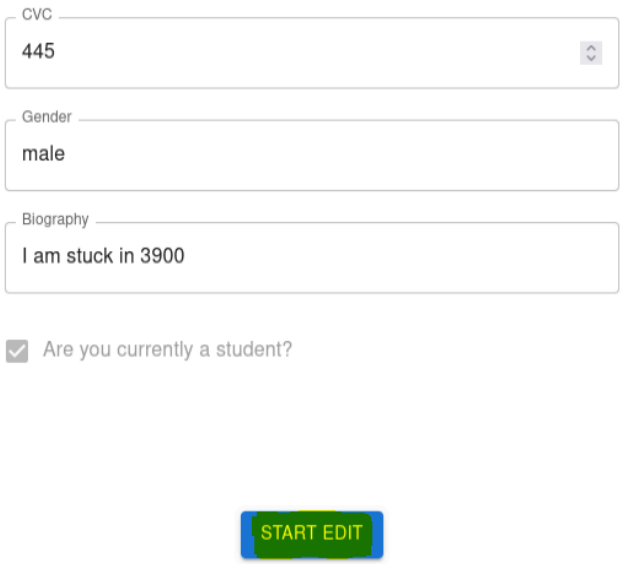
3. Click “Submit” to create the account.

6.2.2 View and edit your profile page

1. After creating an account, you will be redirected to the main page where you will then see the button “Profile” on the top right corner of the screen. Click on it to view your profile



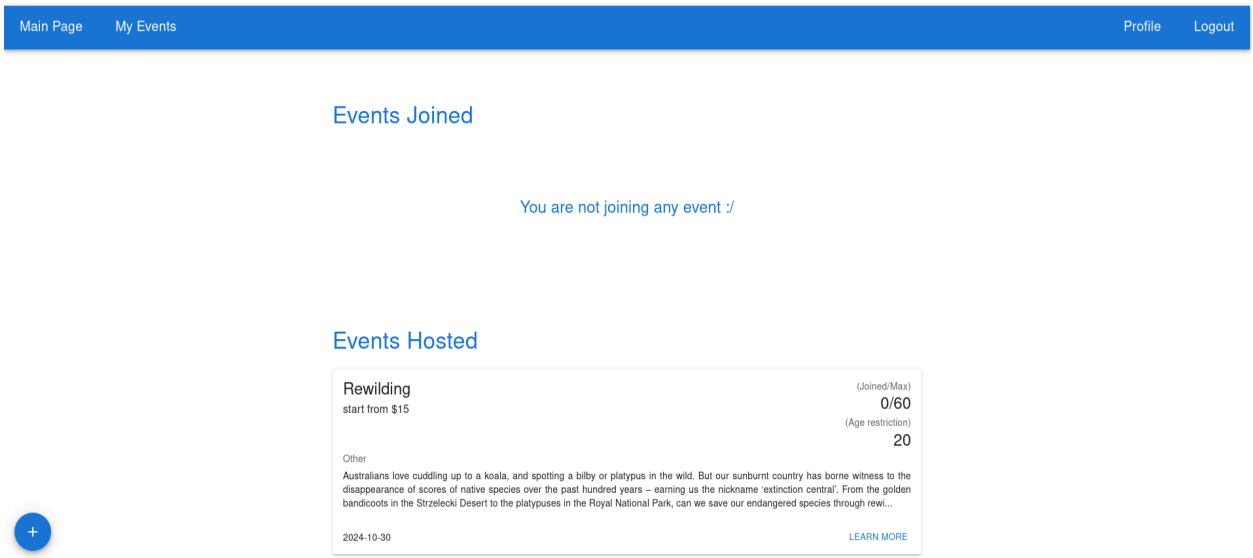
2. Here, you can view your profile details and choose to make edits by scrolling down to the bottom of the page and clicking on the “Start edit” button.
   - Note; you do not have to fill all the fields on the edit page. You can choose to edit whichever field you want.



3. Once you are done editing your profile page, scroll down and click on “Submit”. If you choose not to edit anything, click on “Back”. Either option will redirect you to the profile page where you can see the changes to your account details.

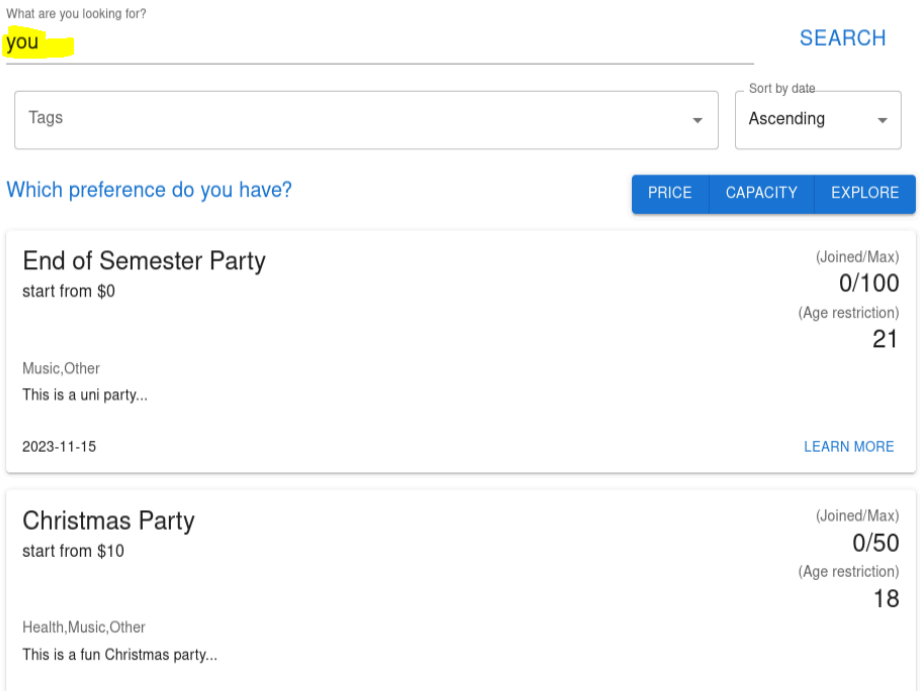
6.2.3 Viewing events you have joined or hosted

1. On the top left corner of the screen, there is the “My Events” button, that when clicked, will take you to a page listing the events that you have either joined or hosted. This page will start to populate once you start hosting and joining events
    - Note: you can only see this button if you have logged in.

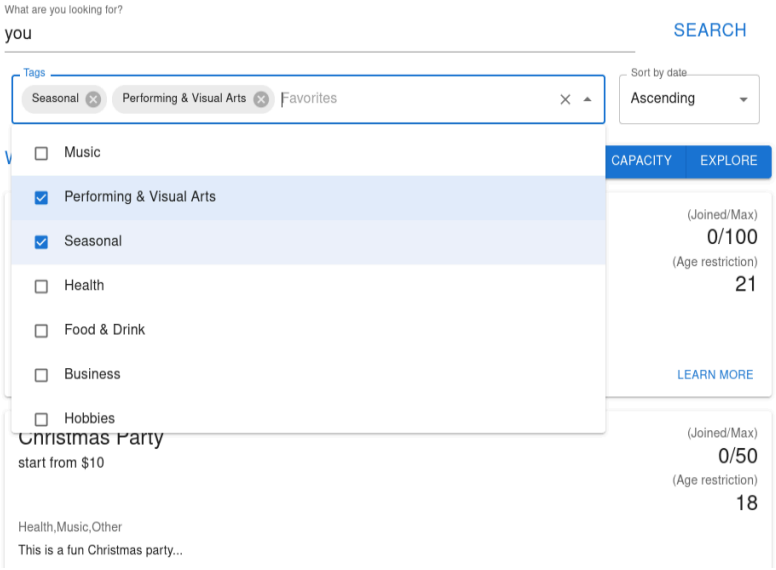


6.2.4 Searching for events

1. On the main page, you will see a search bar on top of a list of events. By default, if you are not searching anything, the website will recommend you events based on previous events that you have attended. If you have not attended any, then you will be recommended random events. To search for an event, type a phrase into the search bar and hit the search button.



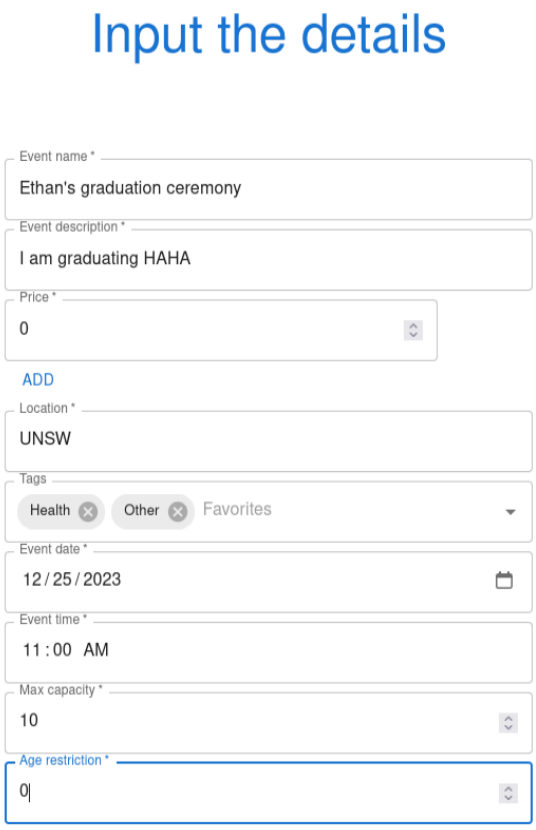
2. You can also filter events based on certain tags of the events using the “Tags” drop list underneath the search bar. Hence, you can sort the events by either ascending or descending order of the event date by clicking on the “Sort by date” drop list to the right of the “Tags” field. By default, the events are sorted by ascending order of the events’ date. After selecting your tags and sort, click “SEARCH”



6.2.5 Creating an event

1. To create an event, click on the plus icon on the bottom left corner of the page.
    - Note: you have to be logged in to create an event.

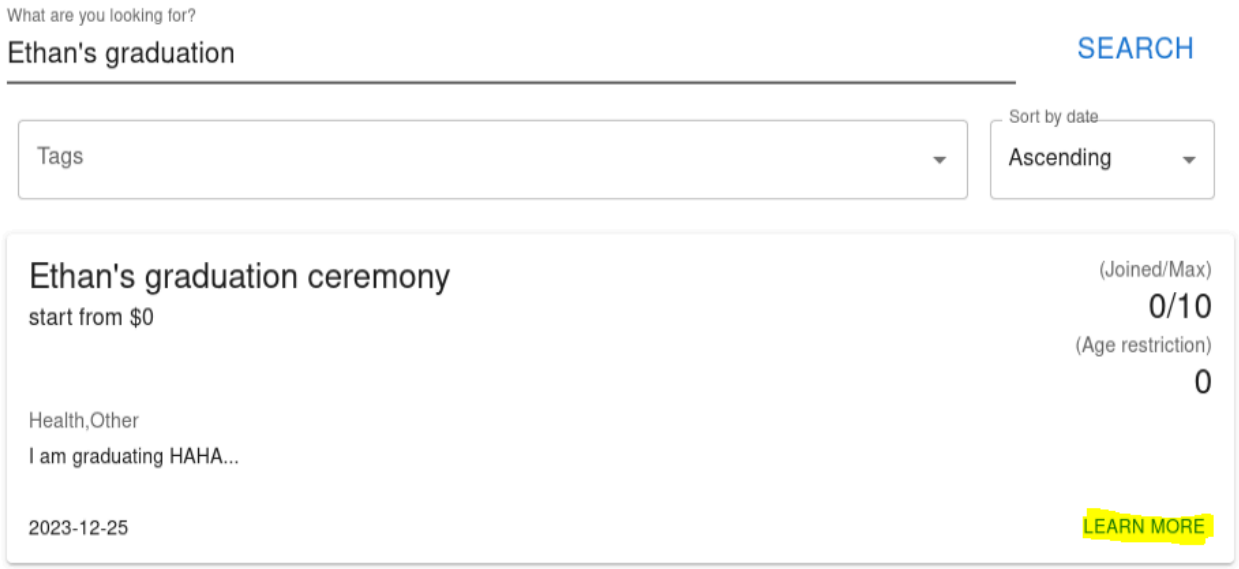
2. Then, you can add the details of your event, including the event's name, description, price, location, tags, date, time, max capacity and a certain age restriction.
    - Note: if your event is free, type 0 into the field. If you have multiple prices for your event for each seat, then click on the "ADD" button to add more prices. The prices will be evenly distributed amongst the max capacity of your event i.e. if you have two prices and a max capacity of 50, then 25 seats will be one price and the other 25 seats will be the other price.
    - You cannot change the price of your event. So double check your prices.
    - If your event does not have an age restriction, type 0.



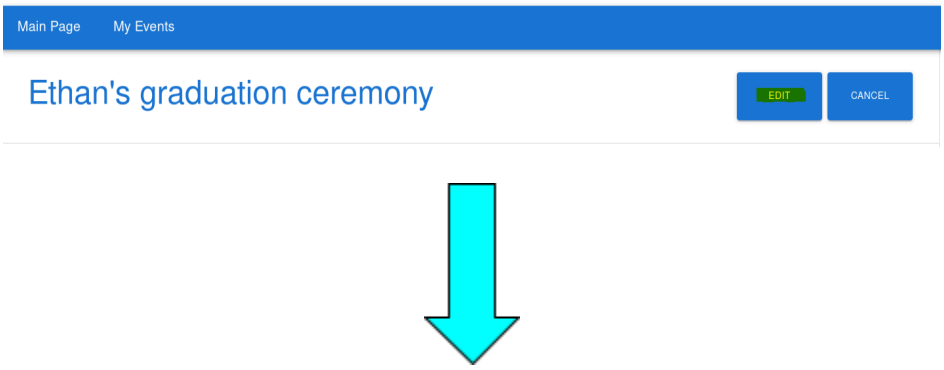
3. Once you are done, click "Create" and you will be redirected to the main page.
    - Note: if you click on the "My Events" page, you will see the event you created under the "Events Hosted" section.

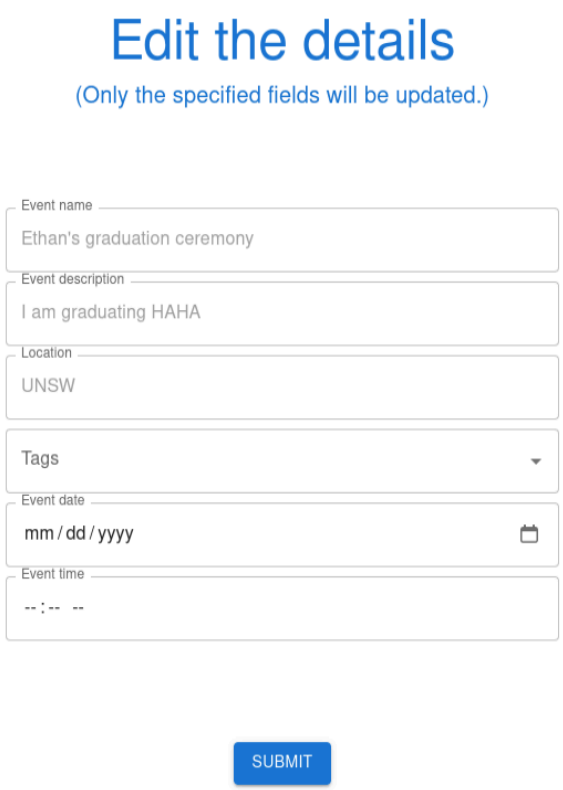
6.2.6 Editing and cancelling an event

1. Find your event in either the main page, by searching for it, or going to the “My Events” page and clicking on “LEARN MORE”.



2. Click on the “EDIT” button and edit the fields you want to edit.
   - Note: you do not have to edit all the fields on the page.

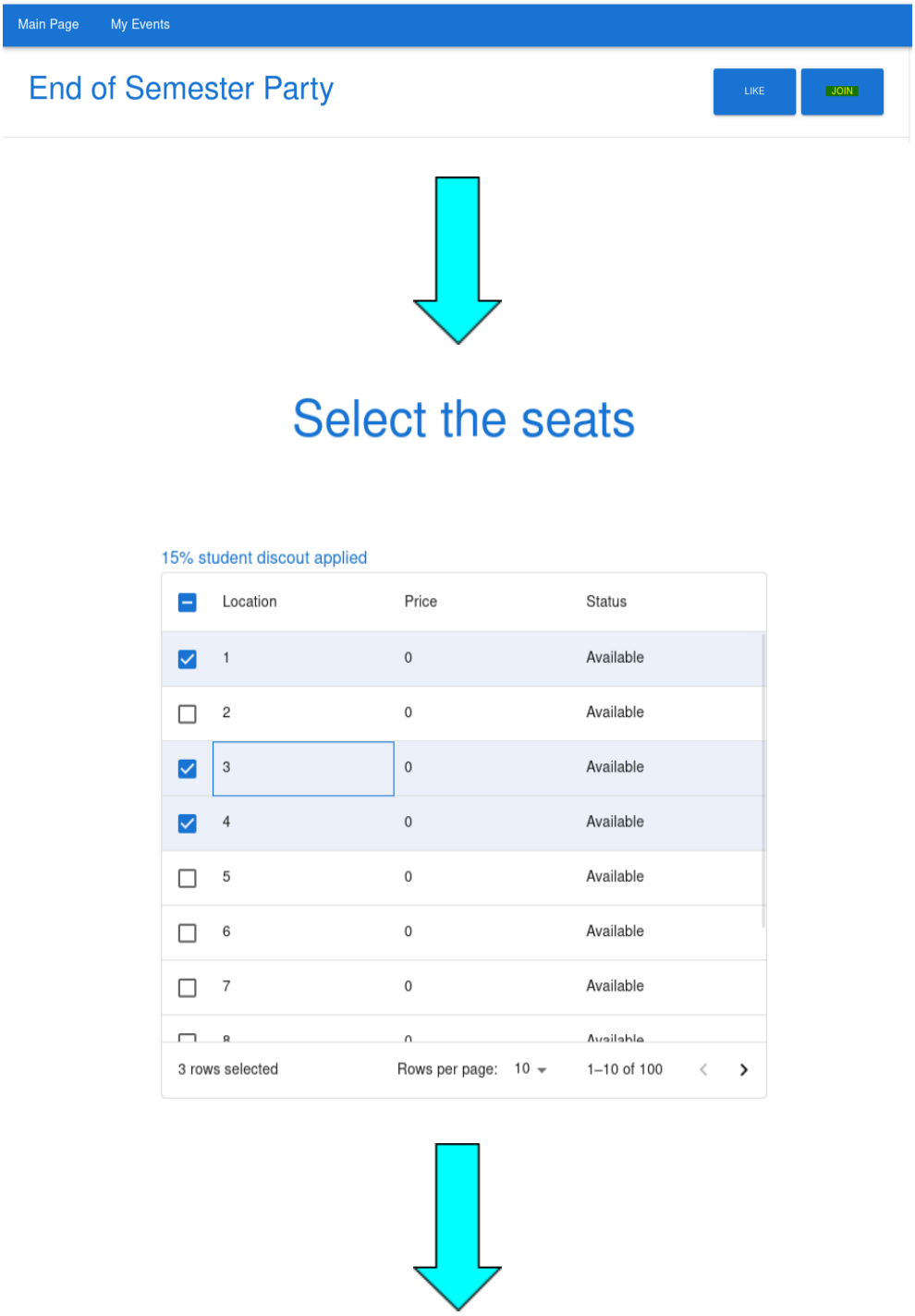


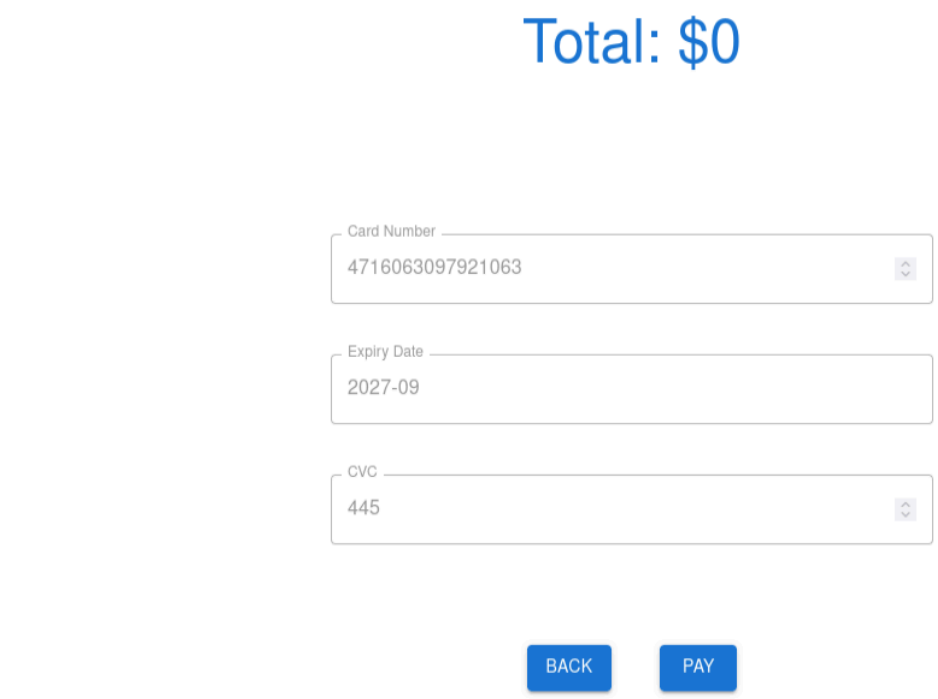


3. Once you are done, click “SUBMIT” and you will be redirected to the event details page where you can see your changes.
    - Note: when you make edits to your event, your customers who have joined the event will receive emails detailing what has changed
4. If you decide to cancel an event, you can click on the “CANCEL” button next to “EDIT”. This will completely remove the event and refund all the money that you received from your customers. You will also no longer be able to see the event in the “My Events” page. Cancelling an event will also send an email to your attendees saying that the event has been cancelled.

6.2.7 Joining an event

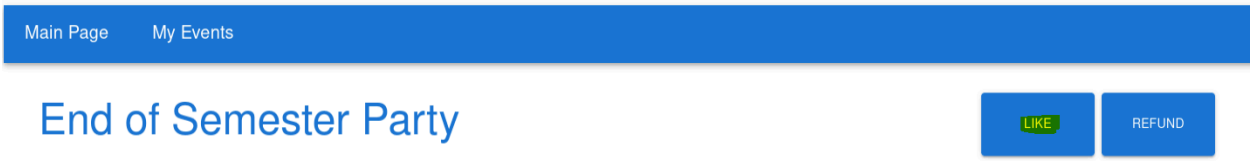
1. Click on the “LEARN MORE” button for an event you are interested in.
2. To join an event, you can click on “JOIN”, select the seats you want to purchase, and finalise the payment by clicking on “PAY”.
    a. Note: in the event’s detail page, on the right side, in the address section, you can click on the address which will open a new window with google maps to show you the event’s location.


Main Page
My Events
End of Semester Party
LIKE
JOIN
Select the seats
15% student discout applied
Location
Price
Status
1
0
Available
2
0
Available
3
0
Available
4
0
Available
5
0
Available
6
0
Available
7
0
Available
3 rows selected
Rows per page:
10
1–10 of 100



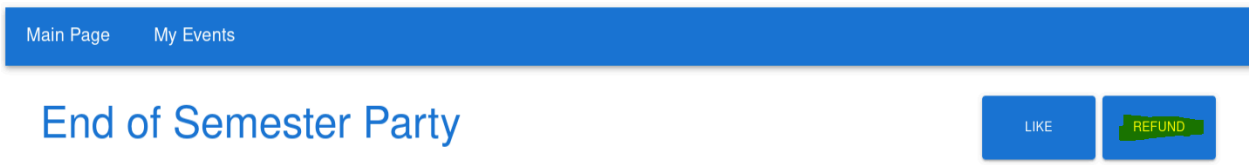
6.2.8 Liking an event

1. If you do not want to join an event just yet but you are interested, you can click on "LIKE" and you will receive emails about any updates for the event.



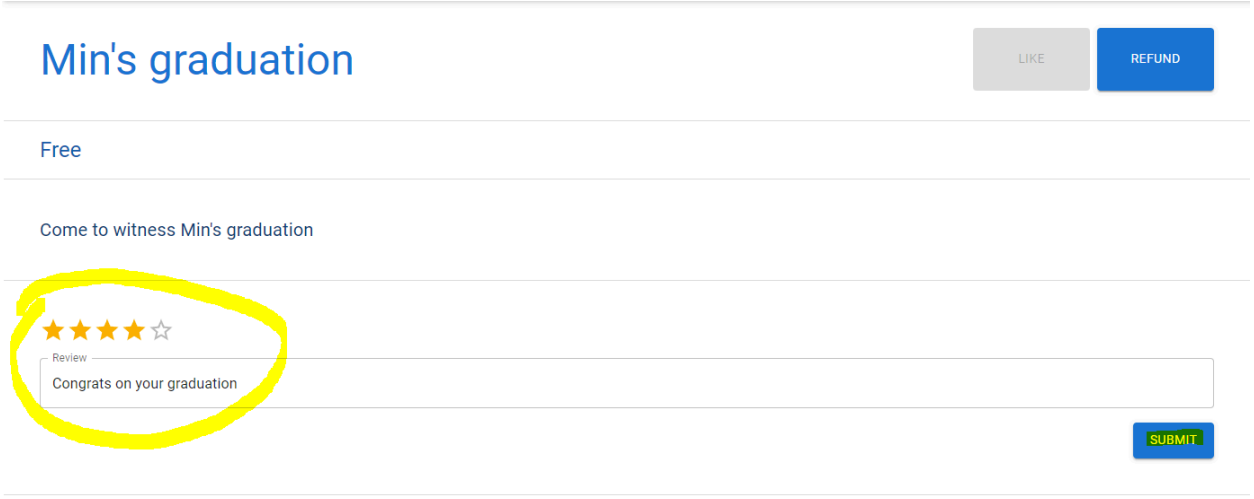
6.2.9 Cancelling your purchase to an event.

1. If you decide not to attend an event you purchased, then go to the event's detail page and click on the refund button. Then, you will be automatically refunded the amount that you used to buy your tickets.
    - Note: you can only ask for a refund when there is more than 7 days left until the event's date.



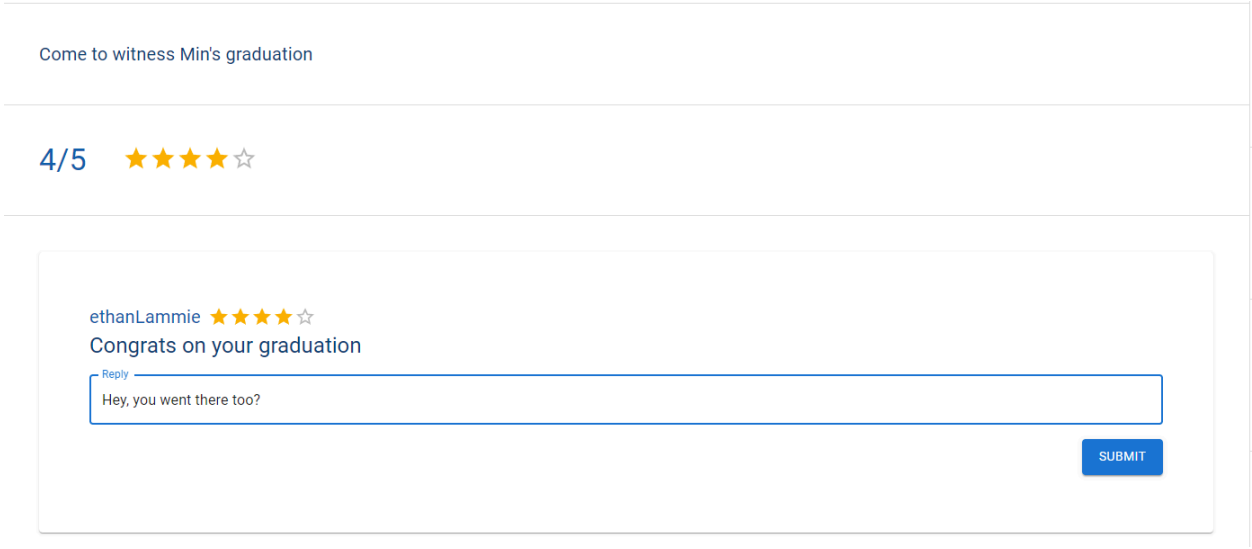
6.2.10 Leaving a review

1. You can write a review for an event you have joined by clicking on the review field, typing your review, leaving a rating, and clicking on “SUBMIT”



6.2.11 Replying to a review

1. You can reply to a review by clicking on the reply field, typing your reply and clicking on “SUBMIT”.



7.0 References

1. audience republic, 2023. *List of Event Ticketing Platforms in Australia.* [Online] Available at: https://www.audiencerepublic.com/ticketing-directory/list-of-ticketing-platforms-in-australia [Accessed 15 June 2023].
2. Business Events Council of Australia , 2021. *COVID-19 Recovery and Rebound Framework* , s.l.: Business Events Council of Australia .
3. similarweb, 2023. *Top Websites Ranking.* [Online] Available at: https://www.similarweb.com/top-websites/e-commerce-and-shopping/tickets/#:~:text=ticketmaster.com%20is%20ranked%20number,Tickets%20category%20in%20May%202023 [Accessed 15 June 2023].
4. *Sequelize V6* (no date) *Sequelize*. Available at: https://sequelize.org/docs/v6/ (Accessed: 04 August 2023).
5. npm. (n.d.). bcrypt. [online] Available at: https://www.npmjs.com/package/bcrypt (Accessed: 04 August 2023).
6. npm. (n.d.). validator. [online] Available at: https://www.npmjs.com/package/validator (Accessed: 04 August 2023).
7. npm. (2023). jsonwebtoken. [online] Available at: https://www.npmjs.com/package//jsonwebtoken [Accessed 4 Aug. 2023]
8. docs.sendgrid.com. (n.d.). Getting started with the SendGrid API. [online] Available at: https://docs.sendgrid.com/for-developers/sending-email/api-getting-started. [Accessed 4 Aug. 2023]
9. mui.com. (n.d.). React Components - Material UI. [online] Available at: https://mui.com/material-ui/. [Accessed 4 Aug. 2023]