

# Deliverable 4

27.04.2020

---

Houston Rockets

SENG2021

Luke Chen, Kenneth Mejico, Vicknesh Ravikumar, Mubarrat Hossain

## Problem Statements

1. There is no application that unifies relevant event information in addition to ticketing, such as weather, travel and venue details.
2. Current event tracking applications do not provide an easy way to apply multiple search filters to explore/search for events.
3. Current event tracking applications do not provide an efficient method to keep track of all events a user is interested in and display updates related to those events

## Our Aim

Our website, EventMaster, was built to provide a one-stop destination that unifies all relevant event information, in addition to ticketing, such as weather, travel, and venue details. Current event tracking systems do not provide an easy way to apply multiple search filters, thus making it difficult for users to find what they are looking for. Furthermore, there are no efficient methods that exist in current event tracking applications that allow users to keep track of all the events that the user is interested in, and display the relevant updates for those events.

## Features

1. Search event by
  - a. name
  - b. date
  - c. artist/team
  - d. genre/type
  - e. location
2. Display weather for event date and location on event page
3. Display 5 star rating system for event location on event page
4. Display travel directions on event date from user's location on event page
5. Link to artist/team social media pages
6. Link to purchase tickets on external site

7. User can register an account
8. User can login to their registered account
9. User can favourite events
10. User can reset password using their registered email

## User Stories

**(1A) Feature:** Search event by keyword/name

**As a:** teenager

**So that:** I can easily find an event spread by my year group

**I want to:** search for upcoming events by using keywords

**Scenario:**

**GIVEN** that I am on the homepage of “EventMaster”

**WHEN** I type “AFL” into the search field in the menu bar

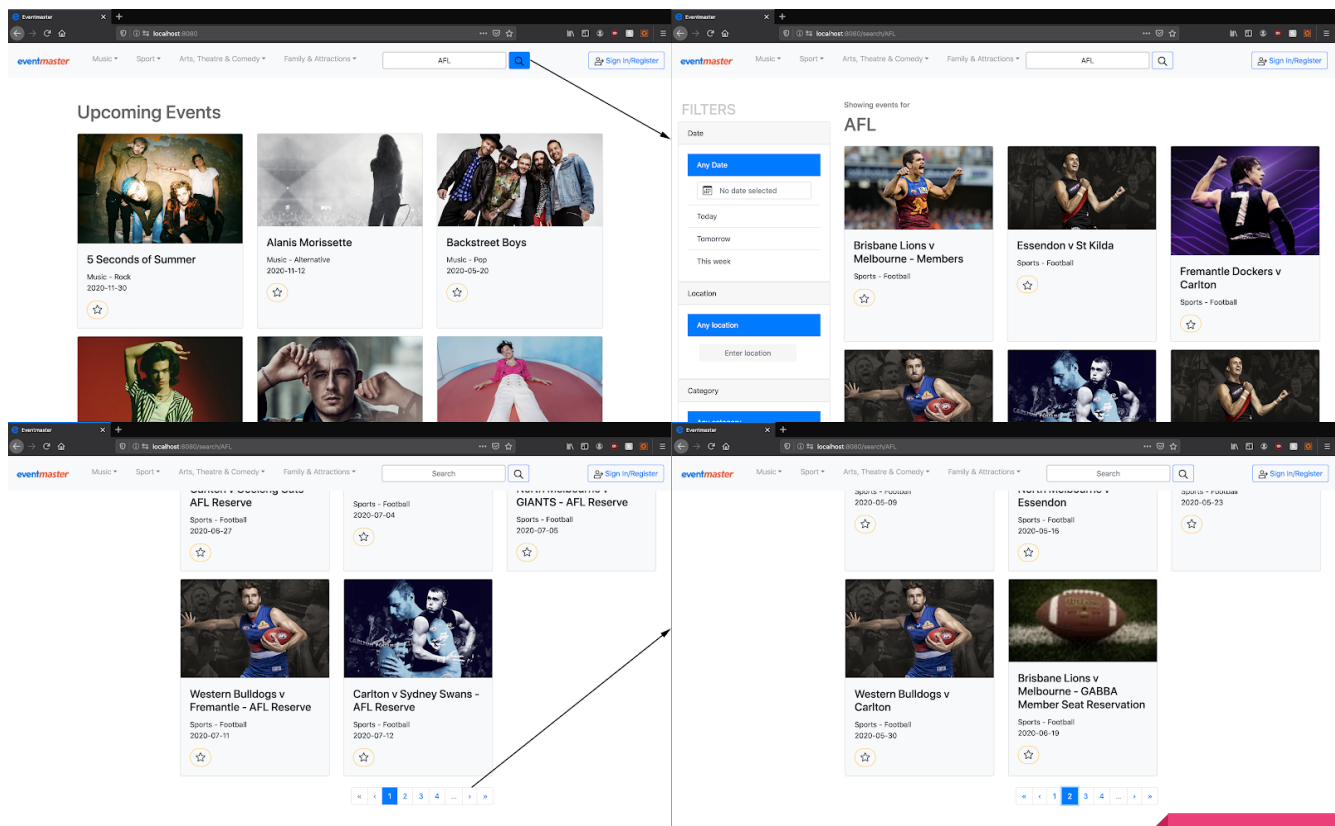
**AND** I press the “Search” button

**THEN** I should be on the “Results” page

**AND** I should see a list of all the events that include the keyword “AFL” in a paginated format with 20 results per page

**WHEN** I click on the right arrow symbol at the bottom of the page

**THEN** I should be on the second “Results” page



**(1B) Feature:** Search event by date

**As a:** working adult

**So that:** I can identify upcoming events that fit into my busy lifestyle

**I want to:** search for upcoming events by their date

**Scenario:**

**GIVEN** that I am on the search page for “AFL”

**THEN** I can see a list of all the events that include the keyword “AFL” that are happening on the current and following dates

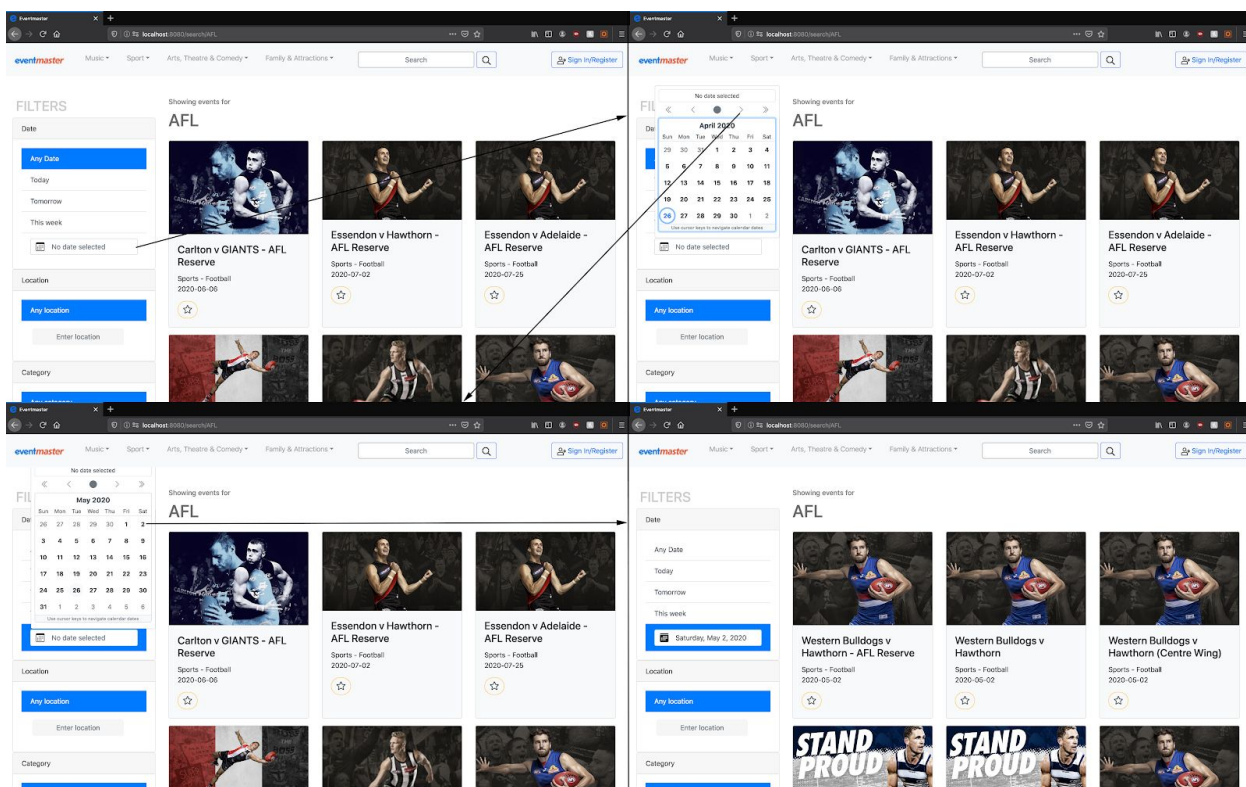
**AND** there is a calendar display on the left hand side of the page showing the current month and the current date highlighted

**WHEN** I click on the right-pointing button on the calendar

**THEN** the calendar will display the next month

**WHEN** I click on a date in the calendar

**THEN** the list of results will show all events that include the keyword “AFL” on the selected date



**(1C) Feature:** Search event by artist

**As an:** avid listener to rap

**So that:** I can find events hosted by my favourite artists

**I want to:** search for upcoming events by performing artists

**Scenario:**

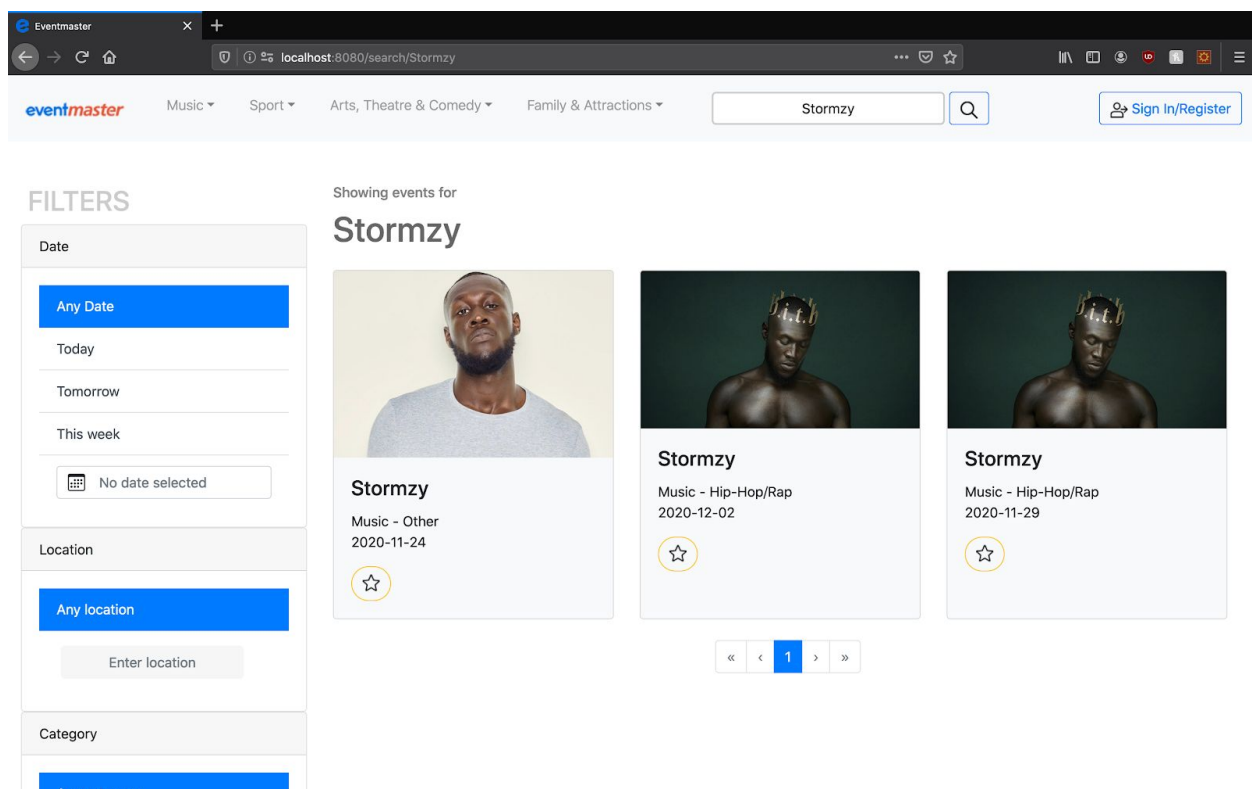
**GIVEN** that I am on the homepage of “EventMaster”

**WHEN** I type “Stormzy” into the search function at the top of the page in the menu bar

**AND** I press the “Search” button

**THEN** I should be on the “Results” page

**AND** I should see a list of all the upcoming events where “Stormzy” performs in, in a paginated format with 25 results per page, in chronological order



**(1D) Feature:** Search event by team

**As a:** Western Bulldogs fan

**So that:** I can identify upcoming games that my favourite team plays in

**I want to:** search for upcoming games by teams

**Scenario:**

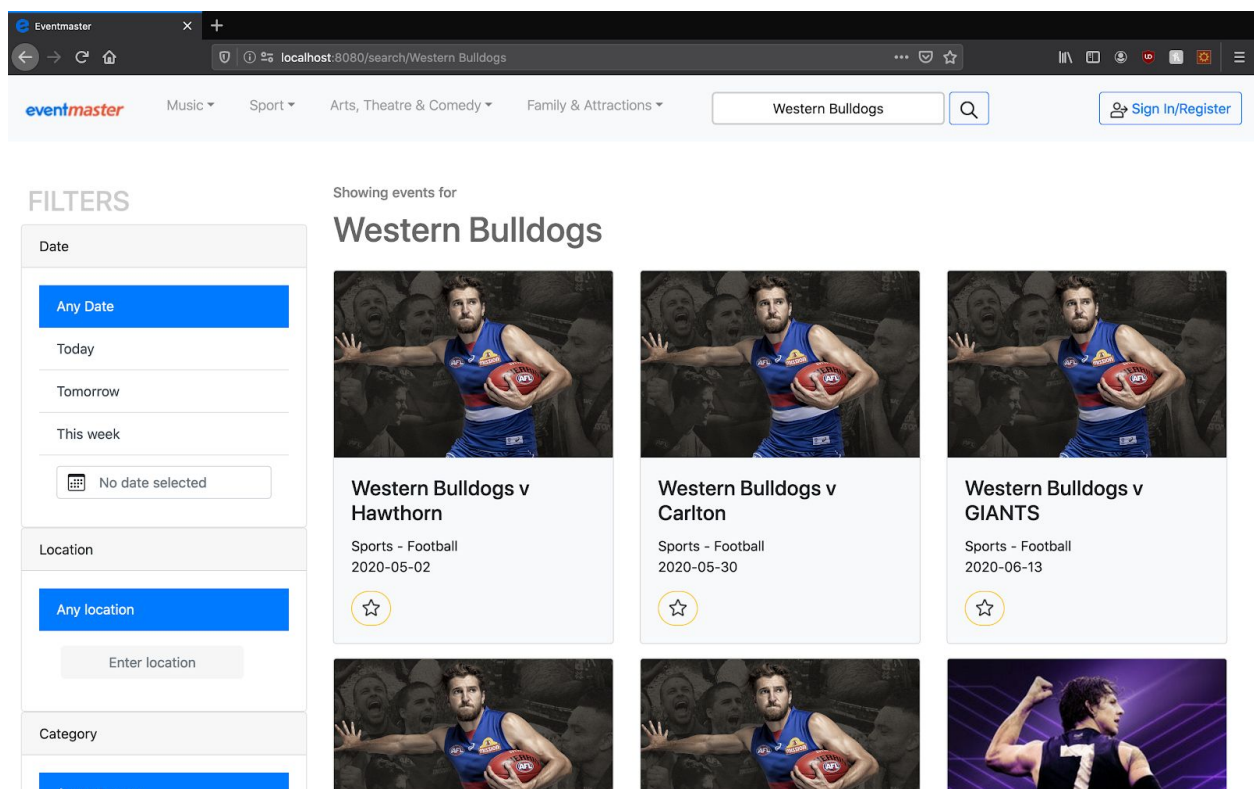
**GIVEN** that I am on the homepage of “EventMaster”

**WHEN** I type “Western Bulldogs” into the search function at the top of the page on the menu bar

**AND** I press the “Search” button

**THEN** I should be on the “Results” page

**AND** I should see a list of all the upcoming events where the team “Western Bulldogs” plays in, in a paginated format with 25 results per page, in chronological order





**(1E) Feature:** Search by genre/type

**As a:** sports analyst

**So that:** I can find sporting events that I can analyse

**I want to:** search for events by their genre/type

**Scenario:**

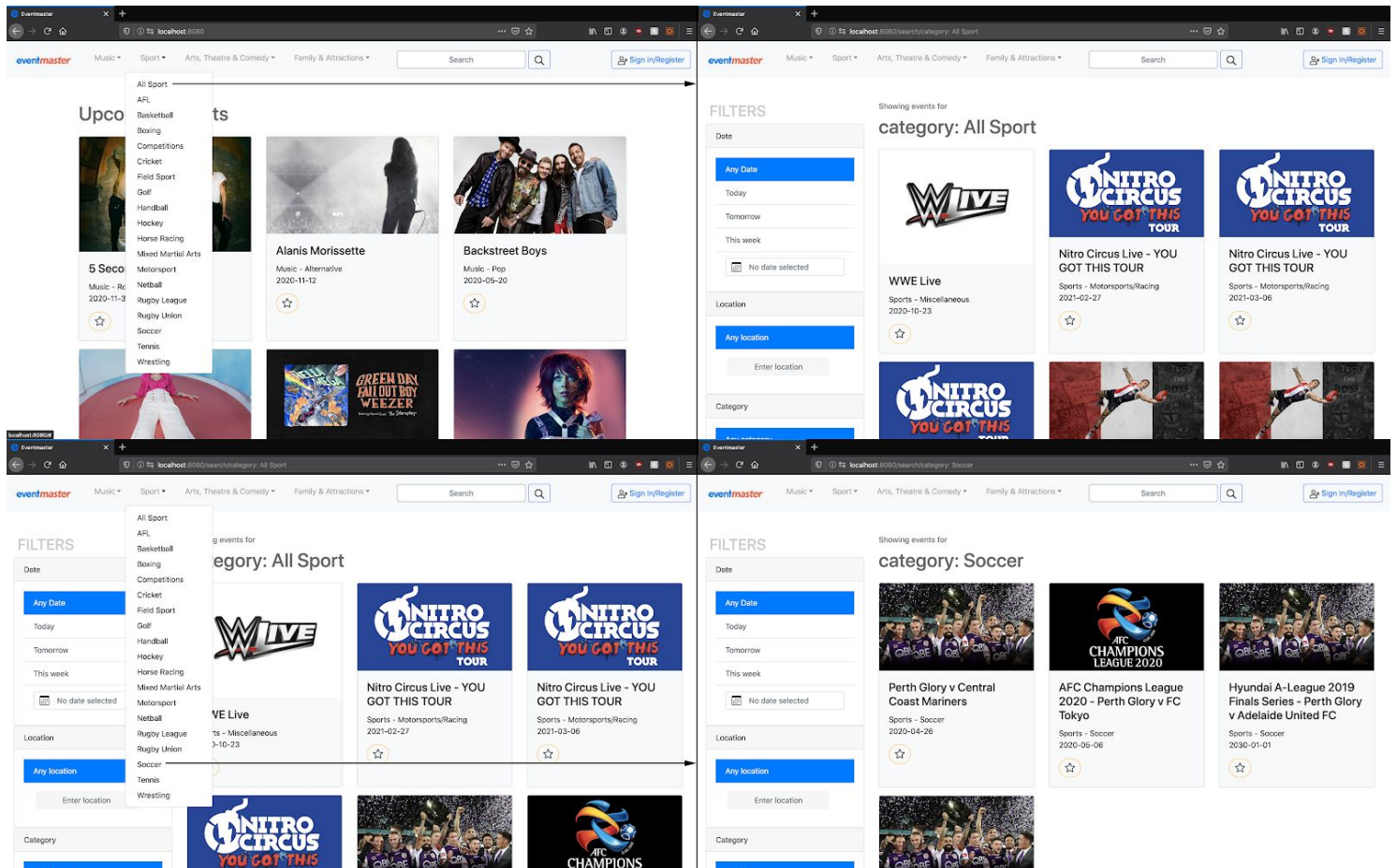
**GIVEN** that I am on the homepage of “EventMaster”

**WHEN** I click on the “All Sports” tab in the menu bar at the top of the page

**THEN** I will be on the “All Sports” page which shows upcoming sporting events in a paginated format showing 25 results per page, in chronological order

**WHEN** I click the “Soccer” tab in the menu bar at the top of the page

**THEN** I will be on the “Soccer” page which show only soccer related events in a paginated format showing 25 results per page





**(1F) Feature:** Search event by location/venue

**As a:** Rock and Pop fan living in Sydney

**So that:** I can find Rock and Pop events happening at a local venue

**I want to:** search for events by their location

**Scenario:**

**GIVEN** that I am on the homepage of “EventMaster”

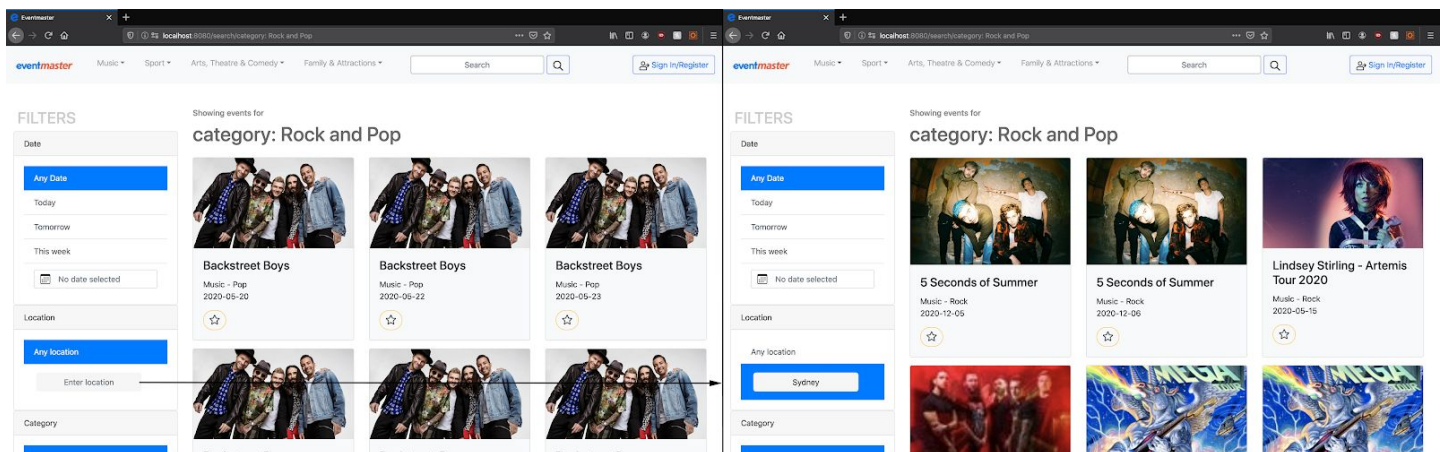
**WHEN** I click on the “Rock and Pop” tab in the menu bar at the top of the page

**THEN** I will be on the “Rock and Pop” page which shows upcoming Rock and Pop events in a paginated format showing 25 results per page in chronological order

**AND** there is a location filter on the left hand side of the page

**WHEN** I type “Sydney” into the location filter

**THEN** the list of results will show all “Rock and Pop” events that are in Sydney



(2) **Feature:** Display weather for event date and location on event page

**As an:** adult

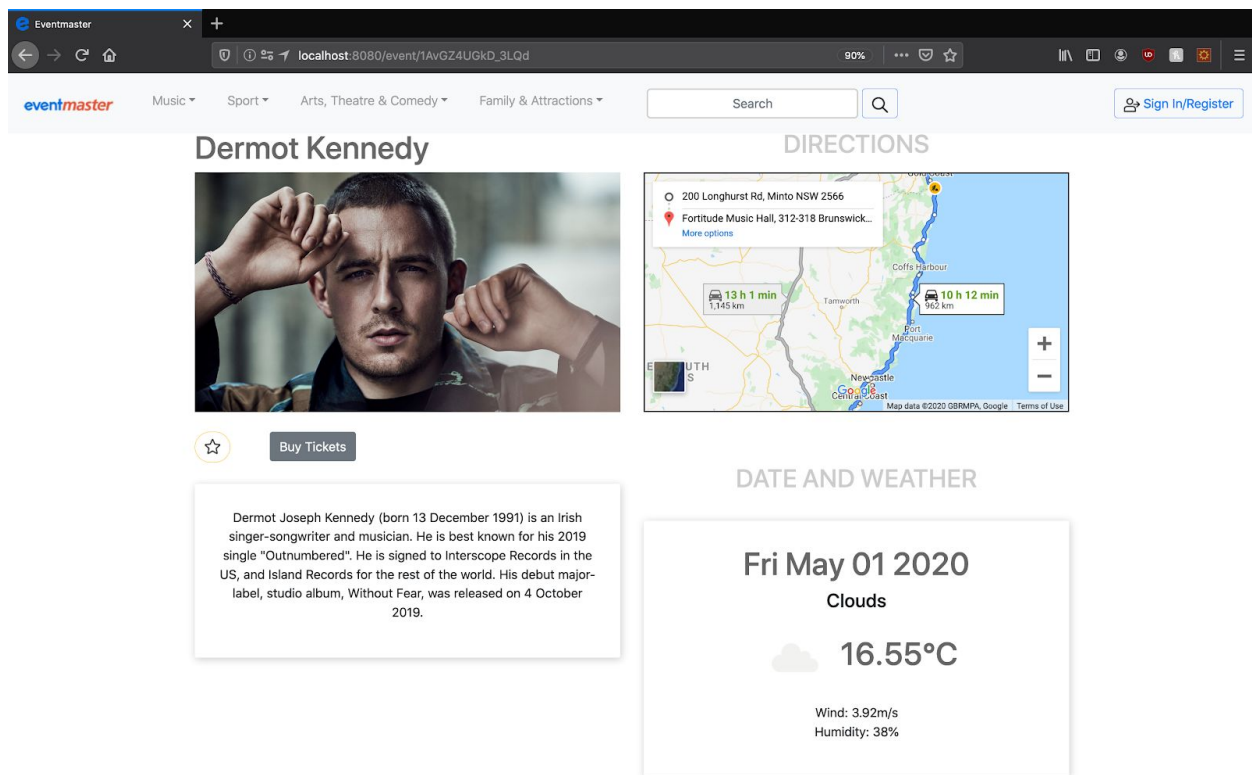
**So that:** I can decide on my outfit and what to bring to the event

**I want to:** know the weather during an event based on its location and date

**Scenario:**

**GIVEN** that I am on the event page for any particular event

**THEN** there is a weather forecast with temperature and conditions for the date and location of the event on the right hand side of the page



The screenshot shows the Eventmaster website interface. At the top, there's a navigation bar with categories like Music, Sport, Arts, Theatre & Comedy, and Family & Attractions. A search bar and a 'Sign In/Register' button are also present. The main content area is for an event titled 'Dermot Kennedy'. It features a large portrait of the artist, a 'Buy Tickets' button, and a bio: 'Dermot Joseph Kennedy (born 13 December 1991) is an Irish singer-songwriter and musician. He is best known for his 2019 single "Outnumbered". He is signed to Interscope Records in the US, and Island Records for the rest of the world. His debut major-label, studio album, Without Fear, was released on 4 October 2019.' To the right of the bio, there's a 'DIRECTIONS' section with a map showing the route from '200 Longhurst Rd, Minto NSW 2566' to 'Fortitude Music Hall, 312-318 Brunswick...'. The map indicates a travel time of 13 h 1 min (1.145 km) and 10 h 12 min (1.145 km). Below the map, there's a 'DATE AND WEATHER' section for 'Fri May 01 2020' showing 'Clouds', a temperature of '16.55°C', wind speed of '3.92m/s', and humidity of '38%'.

**(3) Feature:** Display 5 star rating system for event location on event page

**As a:** user

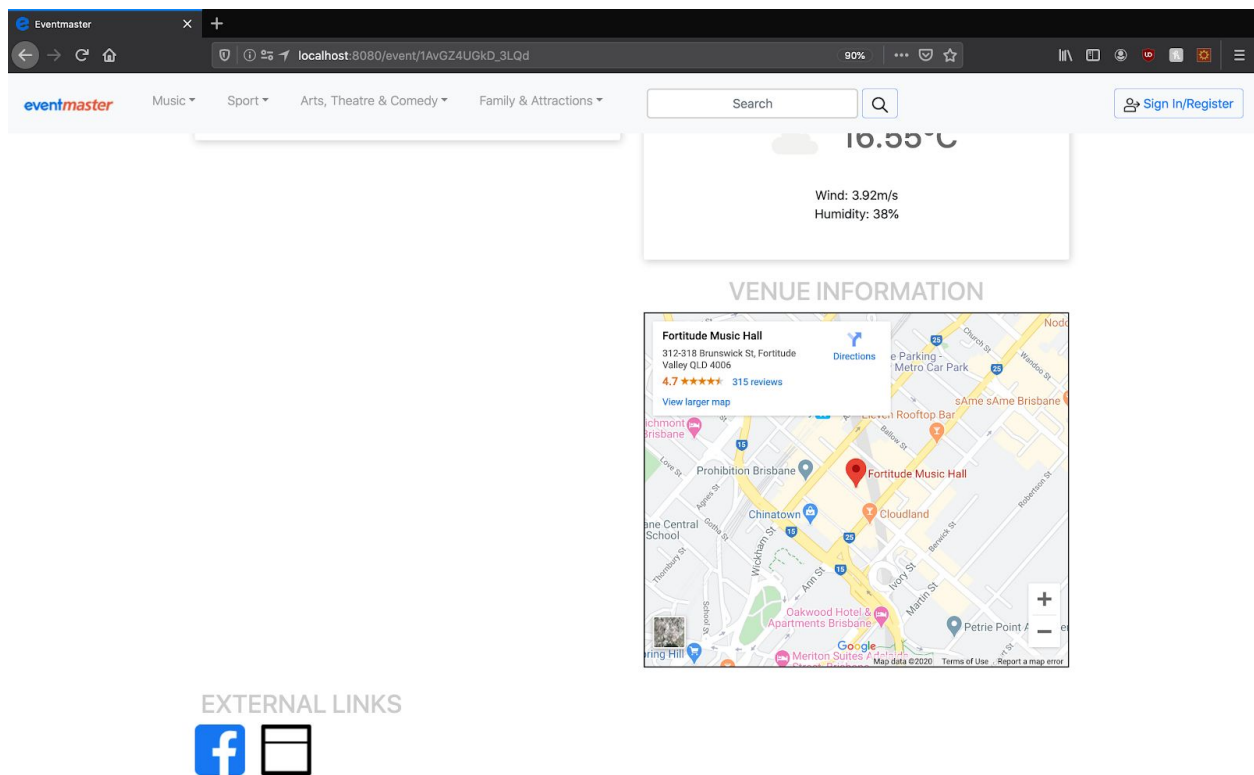
**So that:** I can decide between two different places the same event is being held at

**I want to:** know the overall ratings of the event locations

**Scenario:**

**GIVEN** that I am on the event page of “Dermot Kennedy 1/5/2020”

**THEN** there is a rating for the venue out of 5 at the bottom right of the page



**(4) Feature:** Display travel directions on event date from user's location on event page

**As a:** driving adult

**So that:** I can plan my route to the event beforehand

**I want to:** know directions to the event from my current location

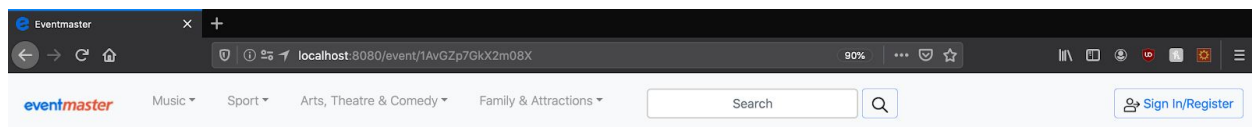
**Scenario:**

**GIVEN** that I am on the event page of "5 Seconds of Summer 5/12/2020"

**THEN** there is a map with a marker on the location of the event at the top right of the page

**AND** it will show me the fastest route to the venue from my current location by car

**AND** it will show an estimated time



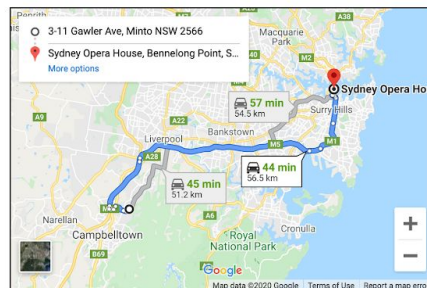
## 5 Seconds of Summer



Buy Tickets

5 Seconds of Summer, often shortened to 5SOS, are an Australian pop rock band from Sydney, New South Wales, formed in 2011. The group consists of lead vocalist and rhythm guitarist Luke Hemmings, lead guitarist Michael Clifford, bassist Calum Hood, and drummer Ashton Irwin. They were originally YouTube celebrities, posting videos of themselves covering songs from various artists during 2011 and early 2012. They rose to international fame while touring with English-Irish boy band One Direction on their Take Me Home Tour. They have since released four studio albums and headlined three world tours. In early 2014,

## DIRECTIONS



## DATE AND WEATHER

Sat Dec 05 2020

Weather data is currently unavailable for this event. Please check again closer to the event date.

## VENUE INFORMATION

(5) Feature: Link to artist/team social media pages

As a: user

So that: I can preview the artist/team's work

I want to: view the artist/team's various social media pages

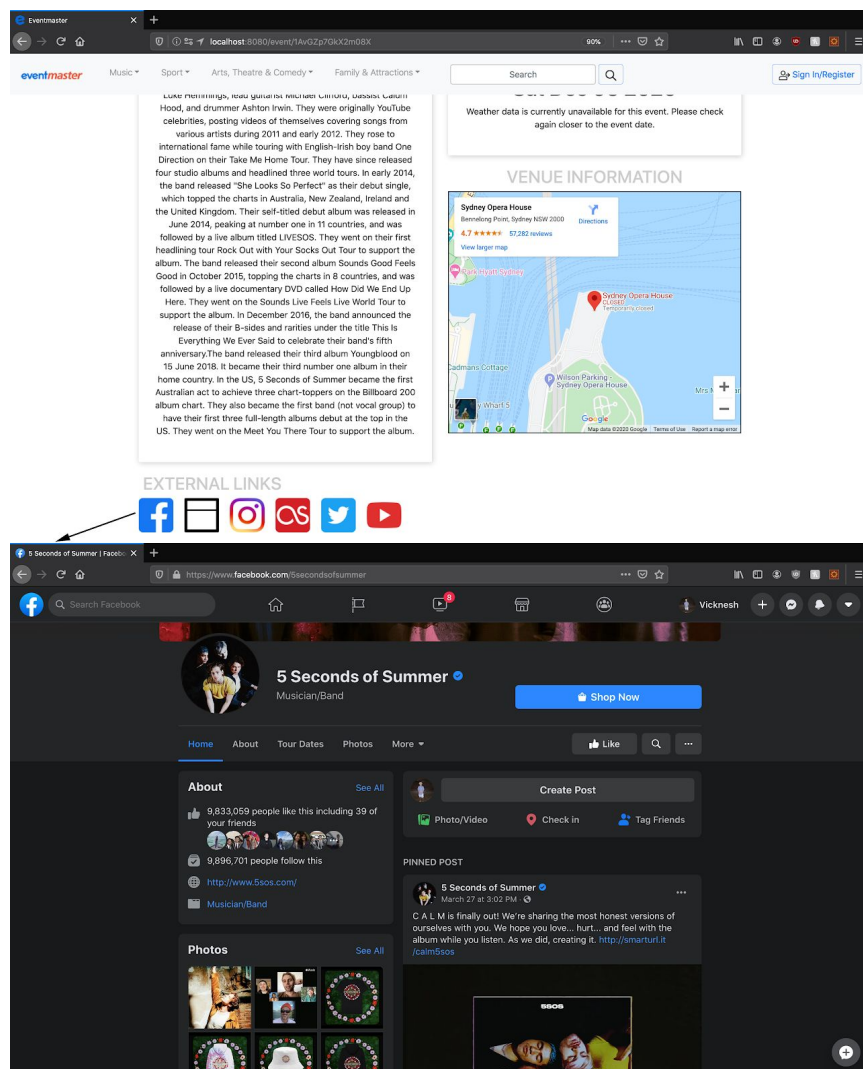
Scenario:

**GIVEN** that I am on the event page for "5 Seconds of Summer 5/12/2020"

**THEN** under the event description there are various social media logos

**WHEN** I click on the Facebook logo

**THEN** I will be taken to the Facebook profile of 5 Seconds of Summer



(6) **Feature:** Link to purchase tickets on external site

**As an:** adult

**So that:** I can attend the event through a receipt/ticket

**I want to:** be able to purchase a ticket

**Scenario:**

**GIVEN** that I am on the event page of “5 Seconds of Summer 5/12/2020”

**THEN** there is a “Buy Tickets” button above the event description

**WHEN** I click on the button

**THEN** it will direct me to a third-party site where I can purchase tickets from

The screenshot shows the Eventmaster website interface for the event "5 Seconds of Summer" on Saturday, December 5, 2020, at 6:00 pm. The page is titled "5 Seconds of Summer" and features a "Buy Tickets" button. Below the button is a description of the band, stating they are an Australian pop rock band from Sydney, New South Wales, formed in 2011. To the right of the description is a map showing the venue, Sydney Opera House, and the route from the user's location. Below the map is a "DATE AND WEATHER" section showing "Sat Dec 05 2020" and a message that weather data is currently unavailable. Below the date and weather section is a "VENUE INFORMATION" section. The "Choose Your Tickets" section on the right lists four ticket options: Standard Ticket (AU\$107.89 ea), Collector Ticket - Standard Ticket (AU\$118.09 ea), Collector Ticket - Soundcheck VIP Package (AU\$301.60 ea), and Soundcheck VIP Package (AU\$291.40 ea). Each option has a "Choose" button. At the bottom of the ticket selection section, there is a "Find Tickets" button.

**5 Seconds of Summer**

**Buy Tickets**

5 Seconds of Summer, often shortened to 5SOS, are an Australian pop rock band from Sydney, New South Wales, formed in 2011. The group consists of lead vocalist and rhythm guitarist Luke Hemmings, lead guitarist Michael Clifford, bassist Calum Hood, and drummer Ashton Irwin. They were originally YouTube celebrities, posting videos of themselves covering songs from various artists during 2011 and early 2012. They rose to international fame while touring with English-Irish boy band One Direction on their Take Me Home Tour. They have since released four studio albums and headlined three world tours. In early 2014,

**DATE AND WEATHER**

**Sat Dec 05 2020**

Weather data is currently unavailable for this event. Please check again closer to the event date.

**VENUE INFORMATION**

**Choose Your Tickets**

- Standard Ticket AU\$107.89 ea
- Collector Ticket - Standard Ticket AU\$118.09 ea
- Collector Ticket - Soundcheck VIP Package AU\$301.60 ea
- Soundcheck VIP Package AU\$291.40 ea

Choose Prices & Sections  
RFST AVAIL ARI F

**Find Tickets**

**These tickets are General Admission and do not have allocated seating.**



**(7) Feature:** User can register an account

**As a:** student

**So that:** I can have a personal “EventMaster” account

**I want to:** be able to sign up for an “EventMaster” account

**Scenario:**

**GIVEN** that I am on the homepage of “EventMaster”

**THEN** there is a “Sign Up” button on the top right hand side of the menu bar

**WHEN** I click on the button

**THEN** it will direct me to a sign up page and ask for my first name, last name, email, username, password and to confirm my password

**WHEN** I fill in my details and press “Register”

**THEN** I will be directed back to the Sign in page where I can sign in with my registered details

localhost:8080/register

eventmaster Music Sport Arts, Theatre & Comedy Family & Attractions Search Sign In/Register

### Register

luke

chen

email@email.com

user1

.....

.....

Register

Already have an account? [Sign in](#)

**(8) Feature:** User can login to their registered account

**As a:** student

**So that:** I can access my personal “EventMaster” account

**I want to:** be able to log in to my “EventMaster” account

**Scenario:**

**GIVEN** that I am on the homepage of “EventMaster”

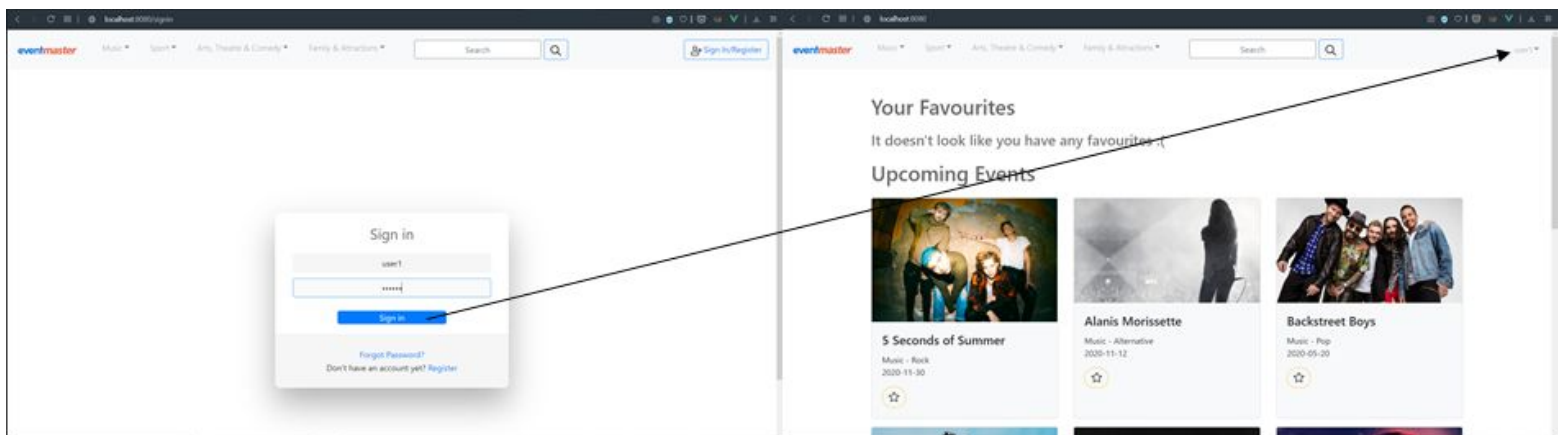
**THEN** there is a “Login” button on the top right hand side of the menu bar

**WHEN** I click on the button

**THEN** it will direct me to a login page and ask for my email or username and password

**WHEN** I fill in my details and press “Login”

**THEN** I will be directed back to the homepage logged in with the “Sign Up” button replaced with my username



**(9) Feature:** User can favourite events to track events when logged in

**As a:** teenager

**So that:** I can track an event for ticket sales and other updates

**I want to:** be able to favourite events and have them show up on my homepage

**Scenario:**

**GIVEN** that I am on the event page of "Stormzy"

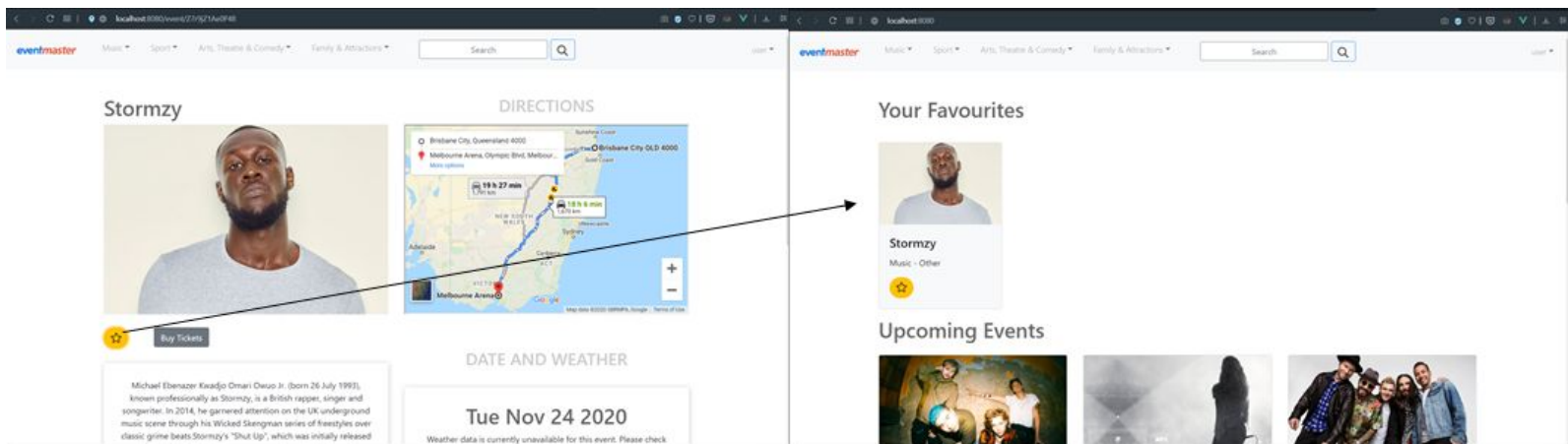
**THEN** there is a "Favourite" star shape button on the right of the event name

**WHEN** I click on the button

**THEN** the symbol will light up

**WHEN** I go to the Home page

**THEN** my favorited events list will have "Stormzy"



**(10) Feature:** User can reset password using their registered email

**As a:** forgetful user

**So that:** I can access my "EventMaster" account after forgetting my password

**I want to:** be able to reset my password

**Scenario:**

**GIVEN** that I am on the homepage of "EventMaster"

**WHEN** I click on the "Login" button on the top right hand side of the menu bar

**THEN** I will be directed to the login page and there will be a "Forgot Password" button under the "Login" button

**WHEN** I click on the "Forgot Password" button

**THEN** I will be on the "Forgot Password" page which has an "Email" entry field and a "Send Recovery Email" button

**WHEN** I fill in the "Email" entry field and click on the "Send Recovery Email" button

**THEN** I will be on the "Recovery Email Sent Page" which displays a message: "Recovery email has been sent."

**WHEN** I check my email and open the recovery email from "event.master.recovery@gmail.com"

**THEN** there is a temporary password

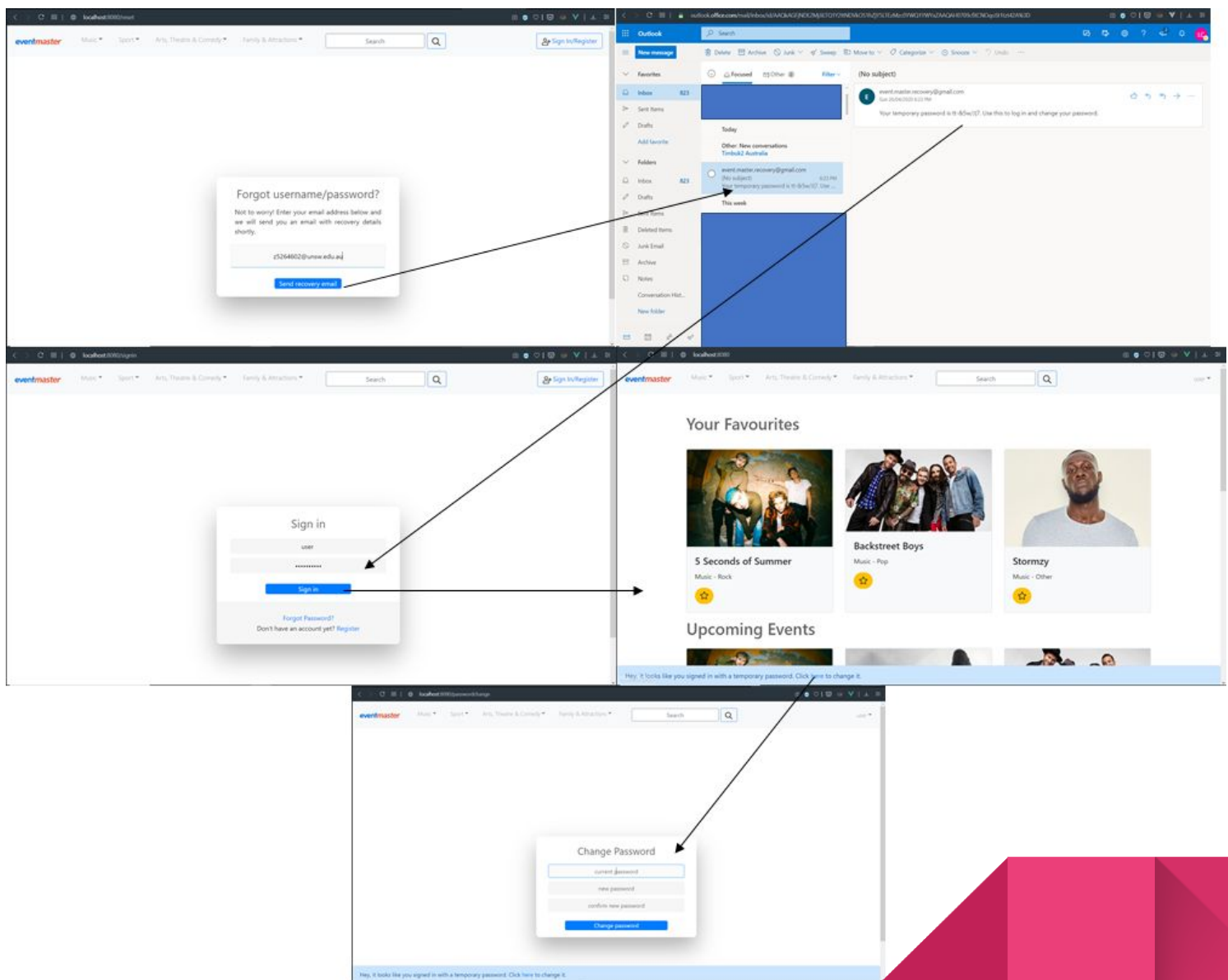
**WHEN** I sign in with the temporary password

**THEN** I am directed to the Home page with an alert to change my password

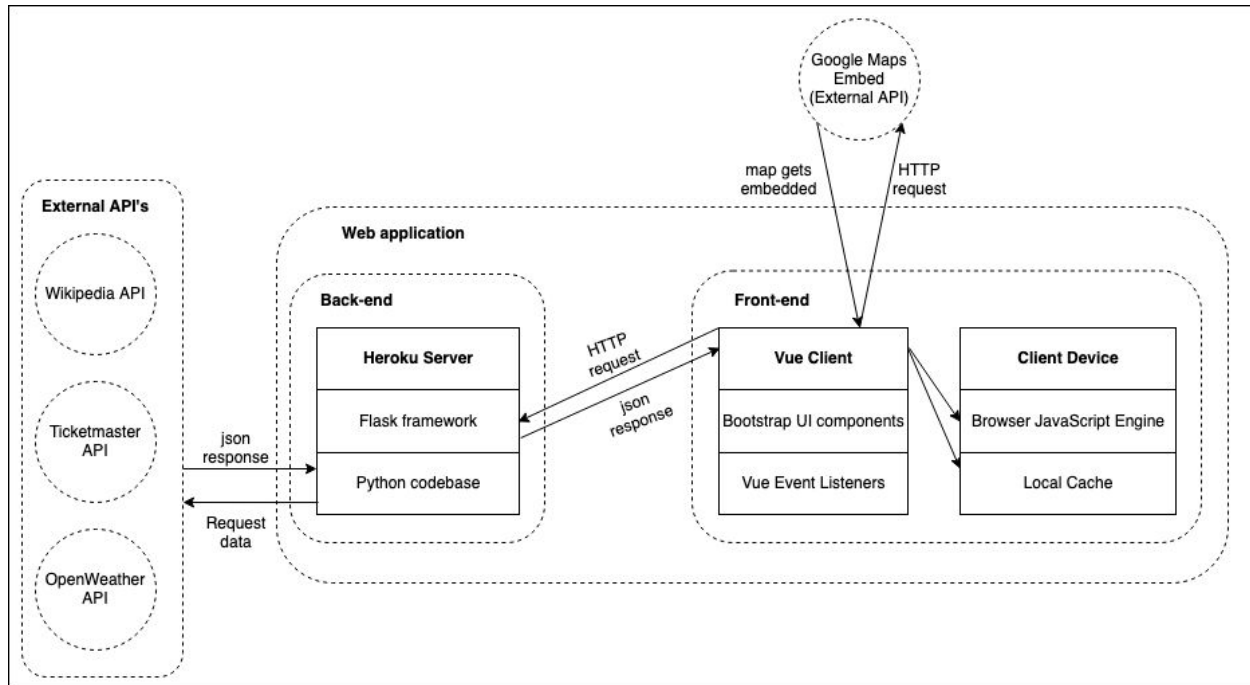
**AND** there are "Current Password", "New Password" and "Confirm new password" fields to fill in and a "Change Password" button under the fields

**WHEN** I fill these fields in with my new password and click on the "Change Password" button

**THEN** I will be taken to the home page with my password changed



## Software Architecture

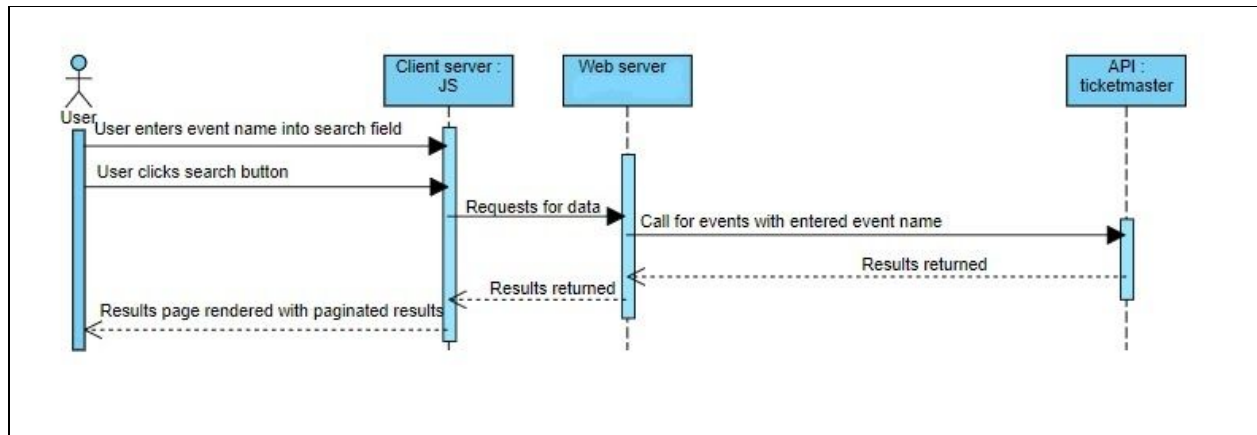


## External Data Sources

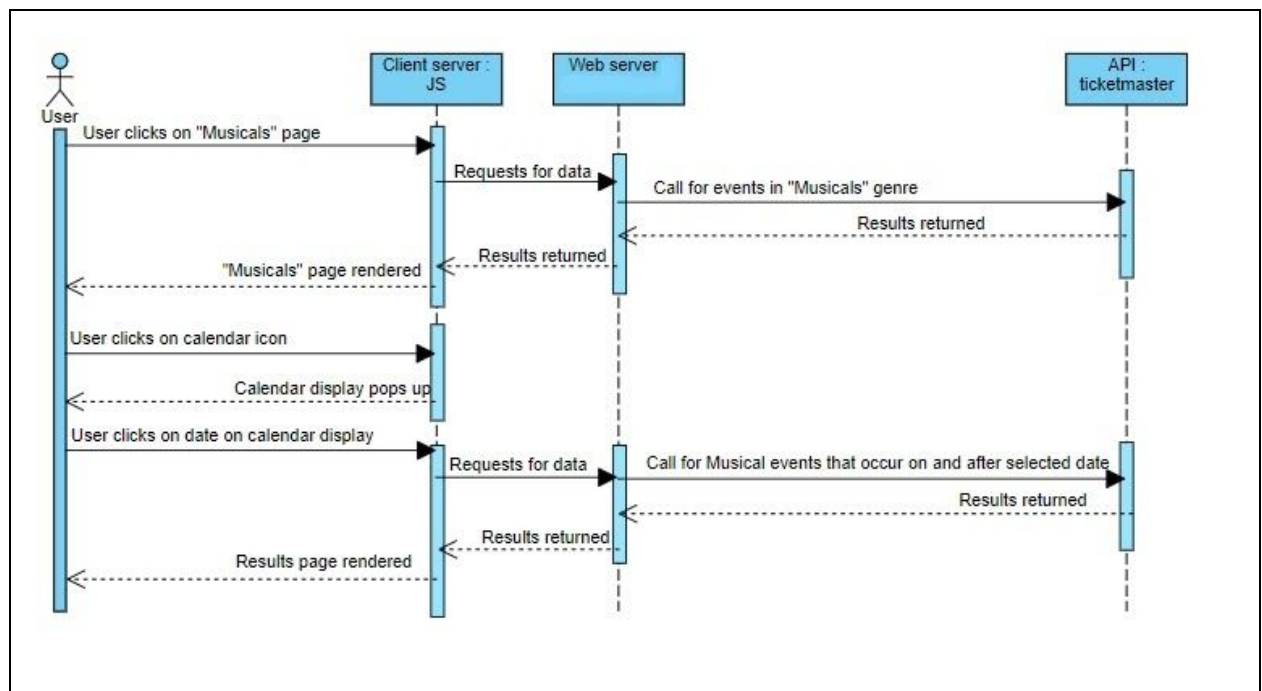
The application will be accessing various external data sources in order to meet the user's requirements. The system will be accessing the Ticketmaster API to get event information including name, date, location, artist etc. The event description will be retrieved from the first page of the first result when the event name is searched for using the Wikipedia API. The OpenWeather API will be used to get weather information up to a 5 day forecast, and the Google Maps API (Embed) will be accessed to obtain travel directions by car, foot, or public transport as well as venue location information such as user ratings. Additionally the Google Maps (Geocoding) API will be used to obtain the longitude and latitude coordinates to be passed to the ticketmaster API for the location filter.

## UML Sequence Diagrams

(1A) **Feature:** Search event by keyword/name

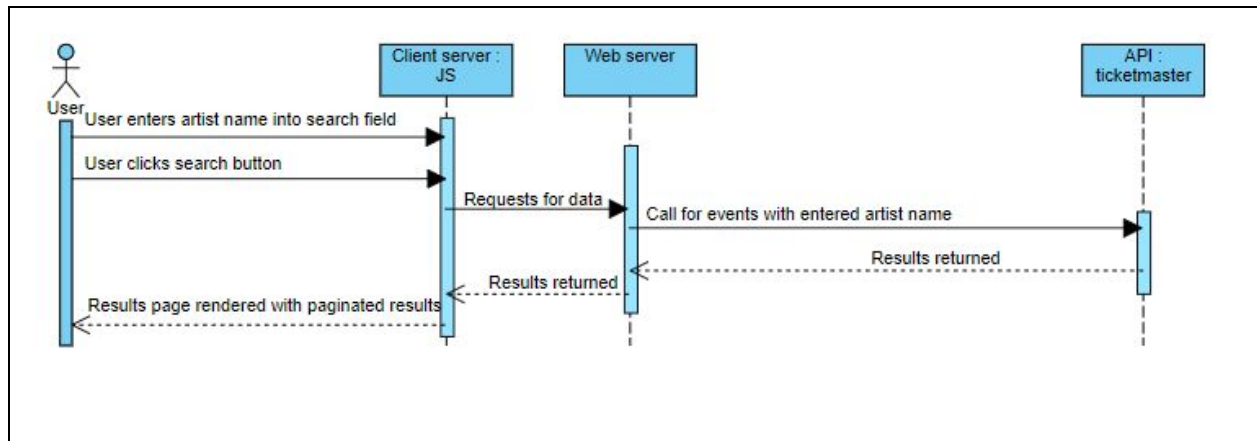


(1B) **Feature:** Search/filter event by date

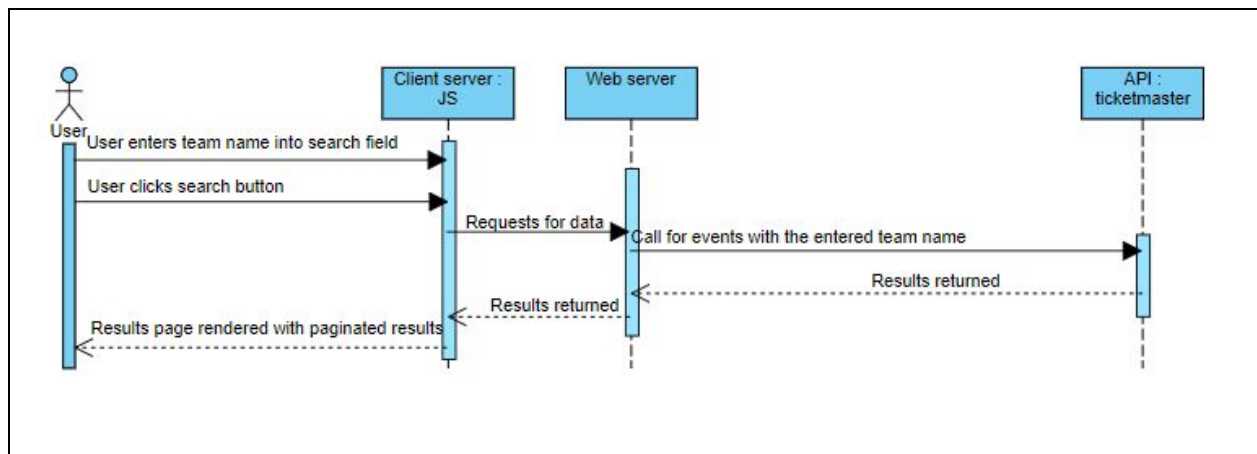




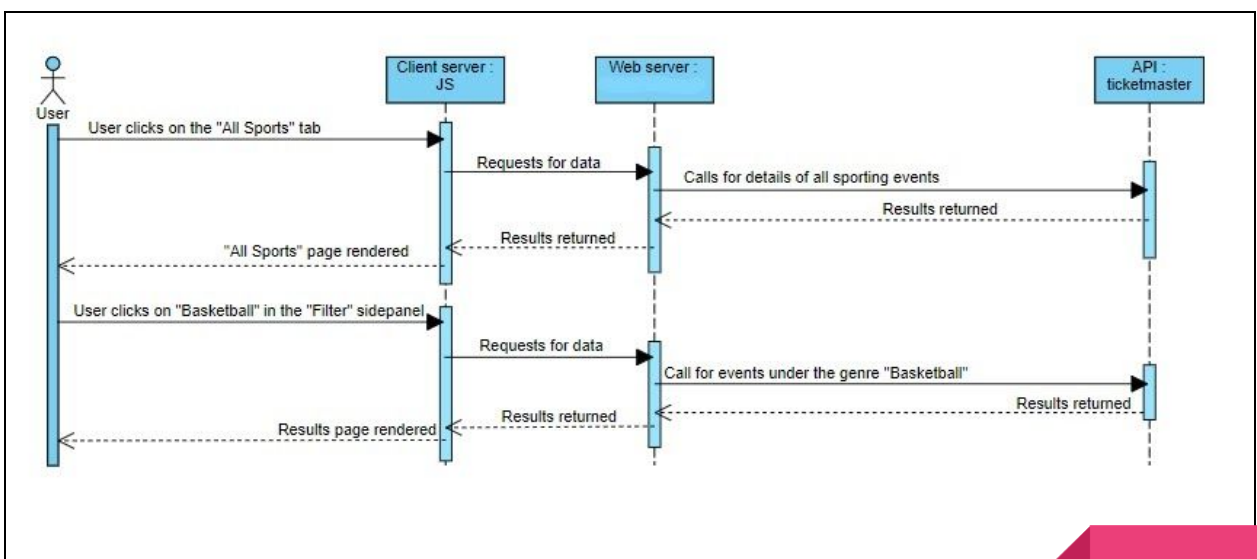
(1C) Feature: Search event by artist



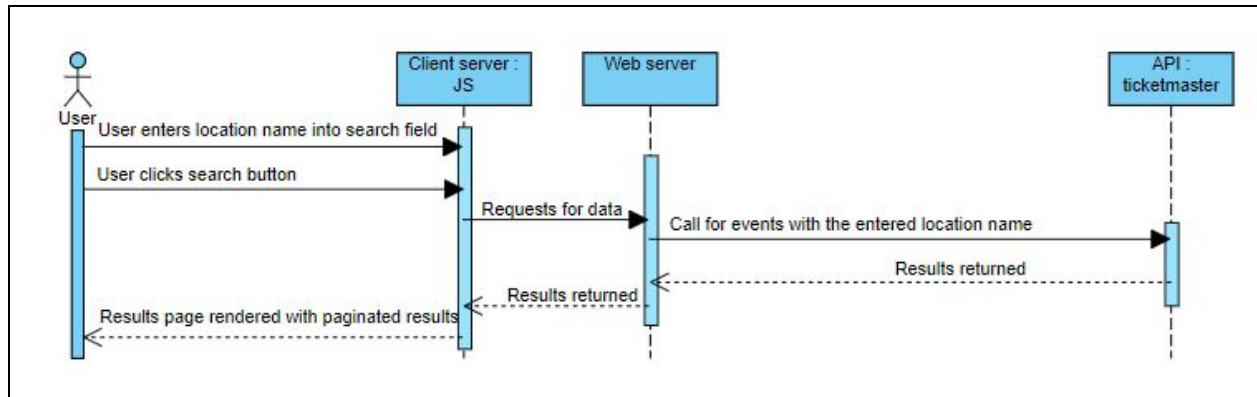
(1C) Feature: Search event by team



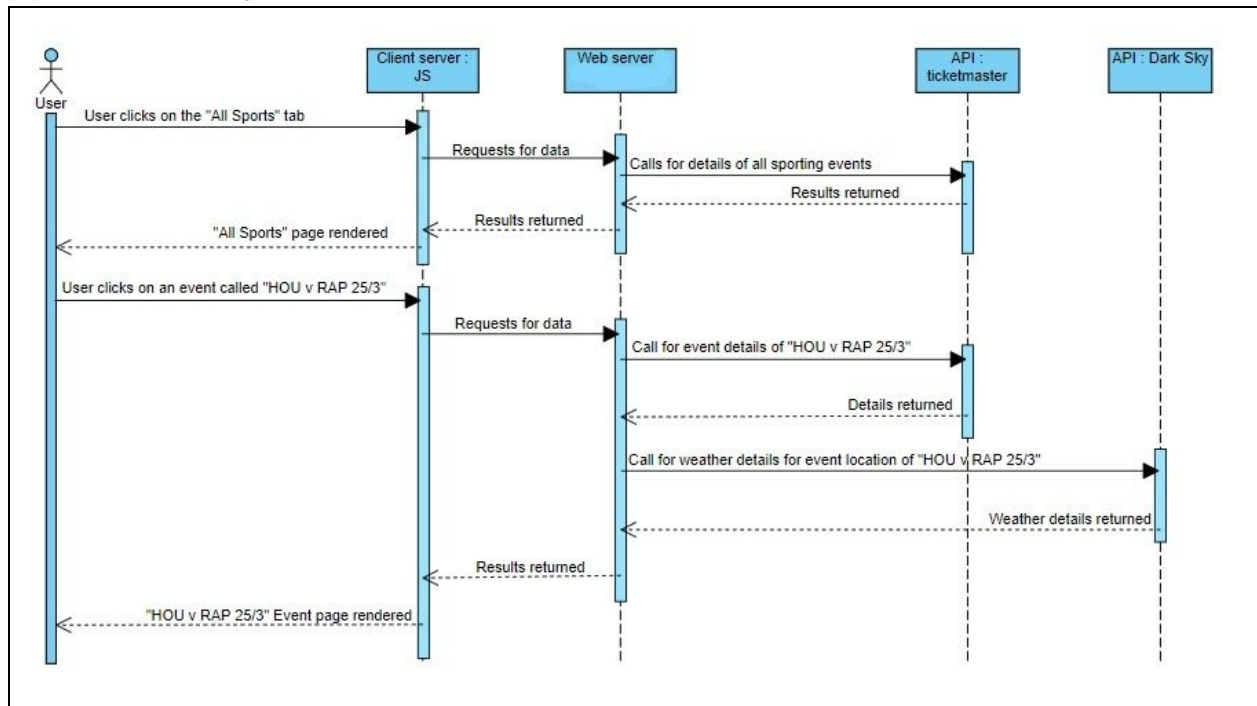
(1D) Feature: Search/filter events by genre/category



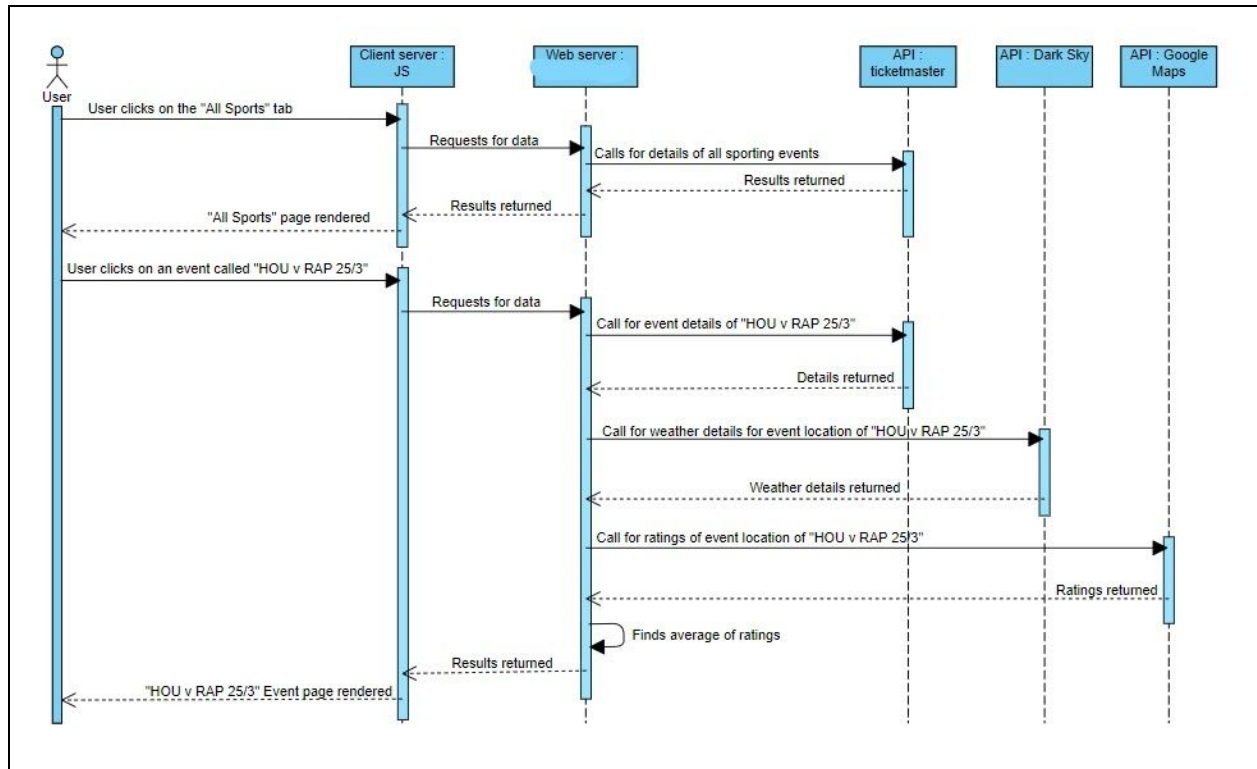
(1E) Feature: Search/filter events by location/venue



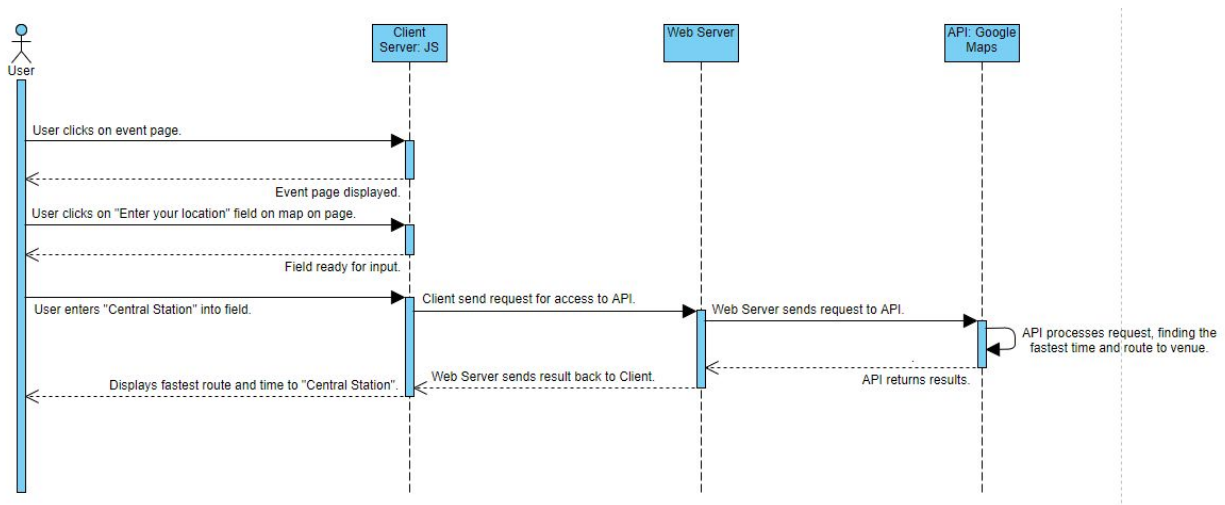
(2) Feature: Display weather for event date and location on event page



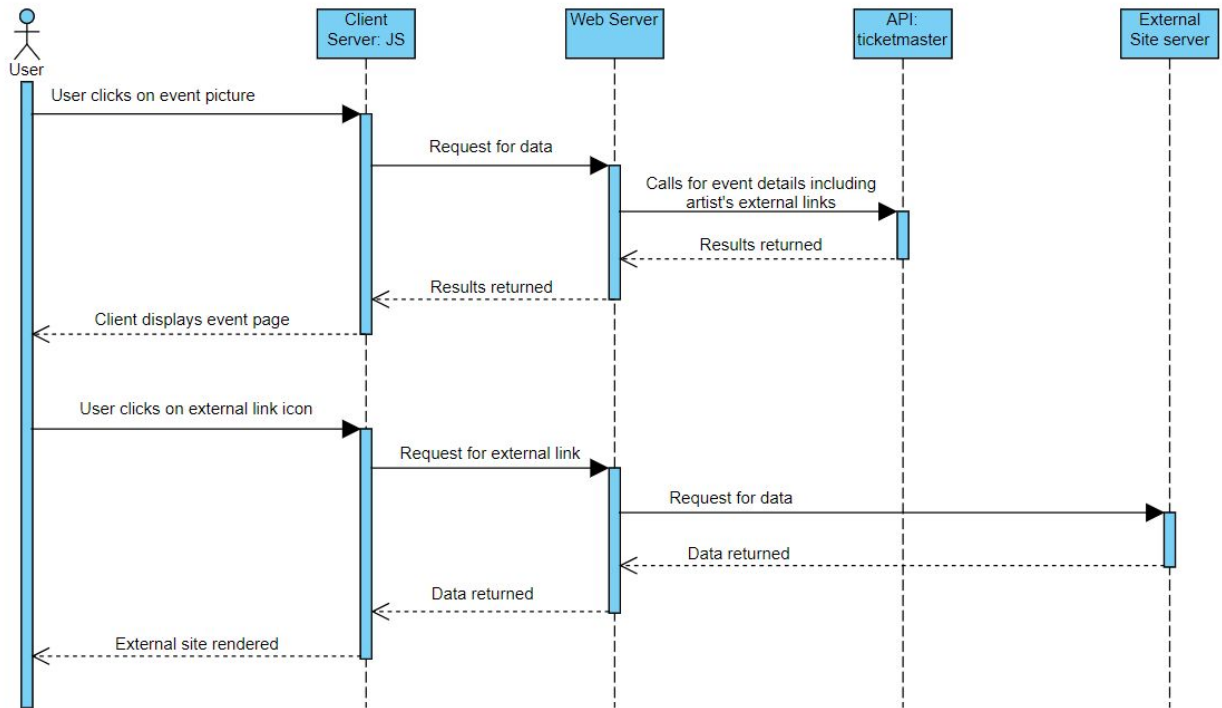
(3) Feature: Display 5 star rating system for event location on event page



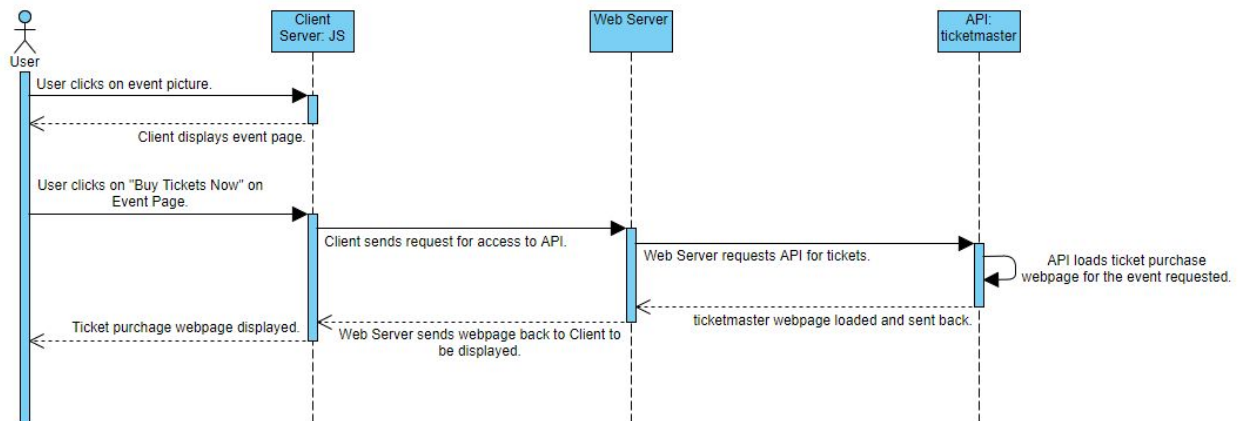
(4) Feature: Display travel directions on event date from user inputted location on event page



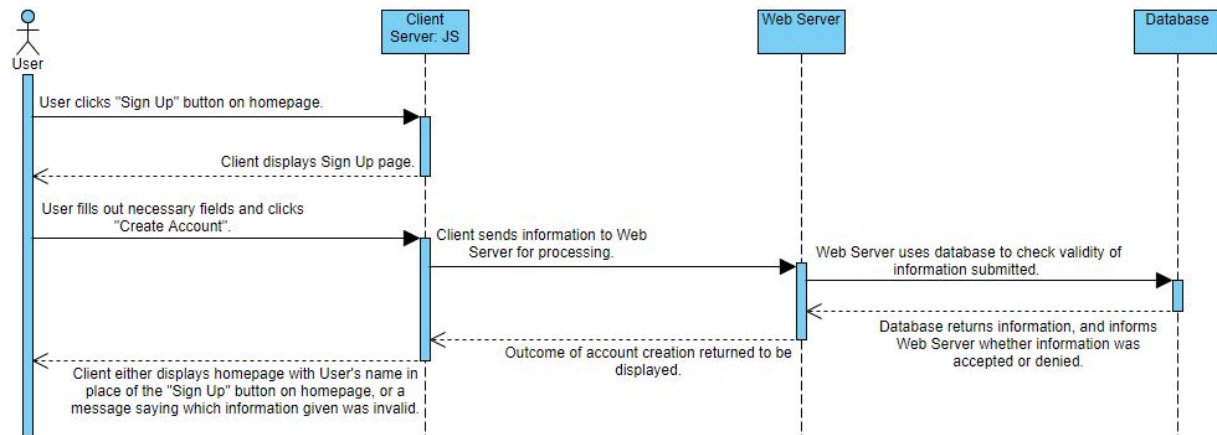
(5) Feature: Link to artist/team social media pages



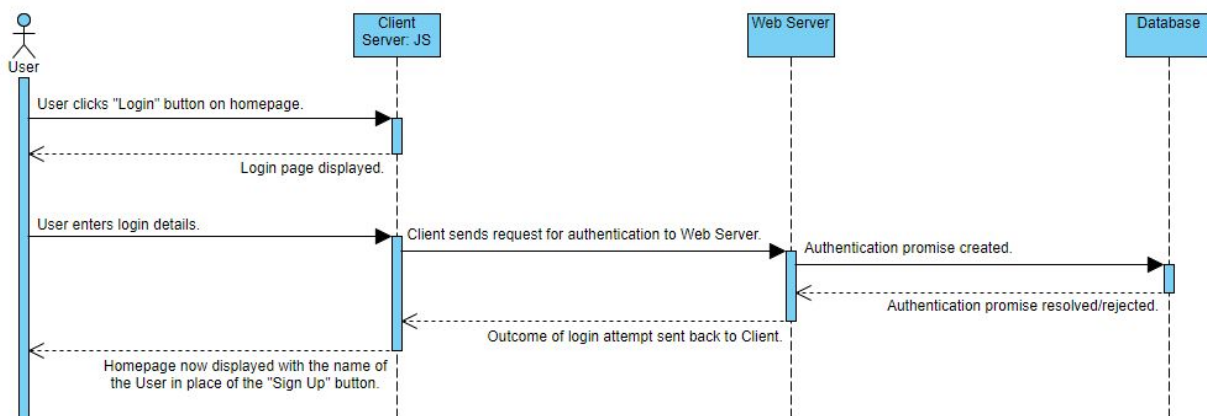
(6) Feature: Link to purchase tickets on external site



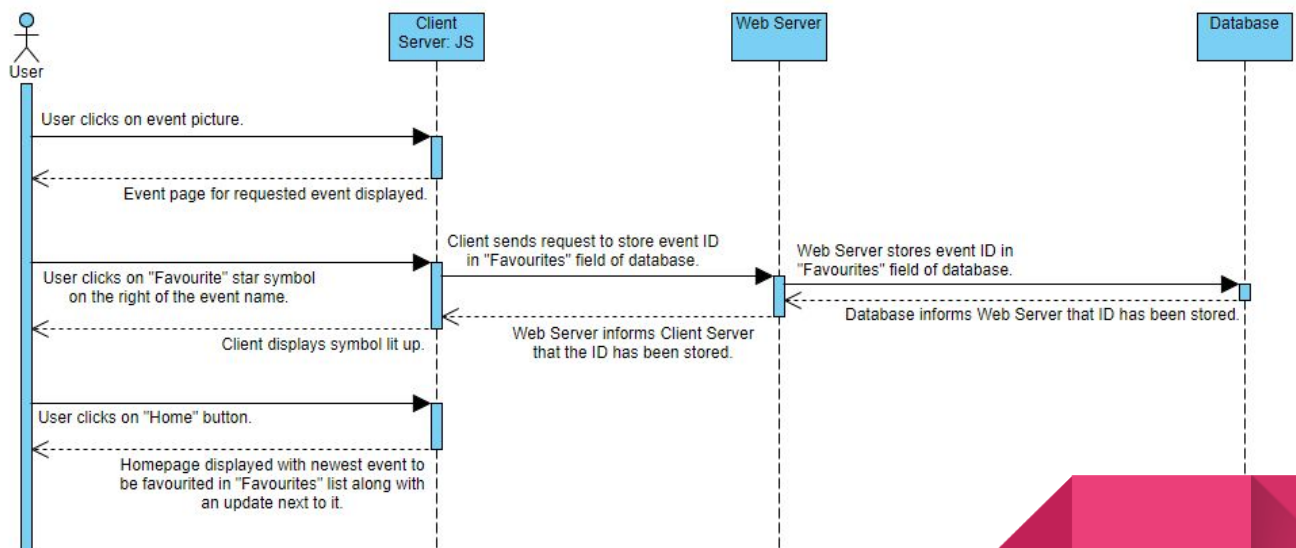
(7) Feature: User can register an account



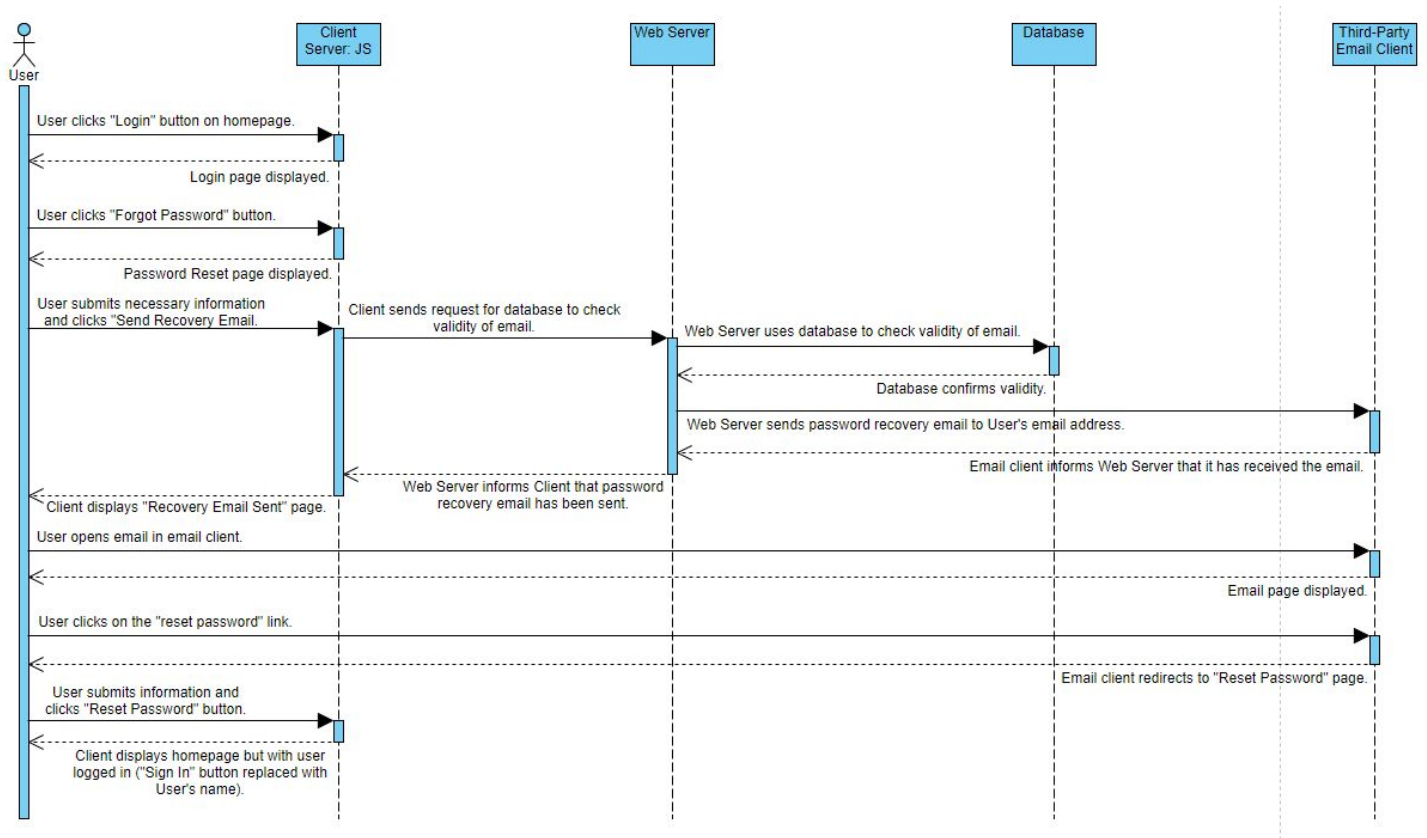
(8) Feature: User can login to their registered account



(9) Feature: User can favourite events to track prices/updates when logged in



(10) Feature: User can reset password using their registered email





## Language Choices

When deciding which language to use for each component of the software architecture, a variety of options were considered and evaluated to choose the languages which were most suitable for the needs of the application *eventmaster*.

### Back-end Development

Feature	Python	Java/C#	C/C++	Relevance
<b>Web frameworks</b>	Flask, Django - easier to use	Can be more complex and harder to use than something like Flask	Not much support, more suitable for mid-level/low-level applications	<b>Very High</b>
<b>Ease of use</b>	Very Good - good readability, automatic typing, automatic memory management, flexible structure	Good - automatic memory management but strongly typed language and object-oriented structure	Moderate - need to manually manage memory, strong typed	<b>High</b>
<b>Performance</b>	OK performance - slow due to interpreter and it being dynamically typed	Good performance - faster than python but slower than C/C++	Very efficient, very good performance	<b>Low - Moderate</b>

After evaluating all three languages, the team decided to use Python for the application's backend development due to its readability, modularity, portability and extensive library support making it ideal for developing a simple web application in a short time-frame. Python's modularity and extensive library support would reduce the amount of code required, shortening the development process whilst making it faster and easier to build and update prototypes.

Collaboration would also be made easier through the readability of the language, allowing team members to easily understand other members’ code.

Whilst Python does cause lower performance speeds due to the need of an interpreter to run the file line-by-line, the speed difference compared to Java/C# will be minimal due to the relatively small scale of the application and backend programming involved. Additionally, Python’s poor support for database access does not affect the current application as all event data will be accessed and stored on external sources with the exception of user data. Furthermore, Python’s lack of support for mobile development also does not factor into our decision due to *eventmaster* being developed as a web application.

### Front-end Development

The languages of JavaScript, HTML, and CSS were chosen for frontend development. This was a natural choice due to the nature of the program being a Web Application. There are alternatives to JavaScript, however, most of these alternatives are not widely used and hence do not have the support or features of JavaScript.

## Framework Choices

### Python Web Framework

Feature	Flask	Django	Relevance to Project
Mapping URLs to views	Yes with support for function and class-based views	Yes with support for function and class-based views	High
Ease of use	Good - easier to learn and set up	Moderate	High
Flexibility	Good - flexible app structure; large availability of extensions	Moderate- forces consistent app structure	Moderate
Performance	Good - smaller and has fewer layers than Django	Almost as good	Moderate

<b>Authentication and Authorisation with account management</b>	No, but can be done through extensions	Yes	Moderate
<b>Input handling, client and server side validation of forms</b>	No, but can be done through extensions	Yes with handling of various security concerns such as CSRF, XSS, and SQL injection	Moderate

After evaluating both Flask and Django web frameworks for Python, Flask was chosen as the preferred framework for the application due to its ease of use and flexibility. Due to the relatively small scale of the application, the many features that Django provides did not seem really relevant to the project. Additionally, the features that are relevant such as input handling and user account management can be implemented in Flask through the use of extensions. Hence, by using Flask, the team is able to easily set up a web server with only the required features for the application, improving both readability and performance.

### JavaScript Framework

Feature	Vanilla JavaScript	Vue	React	Relevance
<b>Ease of use</b>	Poor - have to manually write in features such as state management etc.	Good - lightweight, easy to use	Moderate - steep learning curve	High
<b>Maintainability</b>	Poor - have to manually keep UI in sync with state - a lot of code, low readability	Good with separate JavaScript, HTML, CSS files	Good - use of react components	High
<b>State Management</b>	No	Yes	Yes	High
<b>Routing</b>	No	Yes	Yes	High

<b>Virtual DOMs</b>	No	Yes	Yes	Moderate
<b>Server-side rendering</b>	No	Yes	Yes	Moderate
<b>Documentation</b>	Moderate	Good - considered better than React's	Good with large community	Moderate
<b>Scalability</b>	Poor - hard to scale up	Good - both up and down	Good up, not so good down	Low - Moderate

After evaluating the features of Vanilla Javascript, Vue framework, and React framework, the team decided on using the Vue framework for the front end development of the application. This is because Vue (and React) provide state management and routing among other features and components that significantly reduce the lines of code required, assisting the team in releasing a prototype in the given time frame. Additionally, frameworks make it easier to keep the UI in sync with the state of the application through a virtual DOM which improves efficiency especially in cases where data is constantly changing such as when the user applies filters to search results.

Vue was chosen over React as the preferred framework due to its ease of use in setting up as well as its more common structure of separating JavaScript, HTML, and CSS. React also has a steeper learning curve and lacks the ability to scale down making Vue the more suitable framework for a project on a relatively small scale.

In addition to the Vue framework, the CSS framework Bootstrap Vue will also be used as it offers a variety of design templates which will significantly reduce the amount of code required in styling the HTML elements. Additionally, these design templates are built with properties which simplify data processing and user interactions.

## Deployment Platform

Feature	AWS	Heroku	GitHub Pages	Relevance
<b>Python Flask Support</b>	Yes	Yes	No	Required
<b>Ease of use</b>	Poor - hard to use, requires manual configuration of server and app	Good - performs features automatically, poor configurability	Good - automatic deployment of Github repository	High
<b>Cost</b>	12 month free trial	Free for small projects	Free with Github account	High
<b>Performance</b>	Very Good	Slow first request	Only supports static content	Moderate
<b>GitHub Integration</b>	Yes	Yes	Yes	Moderate
<b>Deployment Speed</b>	Can be quite a hard process so may take some time	Fast to deploy, easy to make changes and updates after	Fast and easy with Github repository	Moderate

After evaluating the deployment platforms of Amazon Web Services (AWS), Heroku, and GitHub pages, Heroku was chosen as the preferred deployment platform due to its ease of use and rapid deployment, allowing for easy deployment of prototypes and updates to the application. Whilst AWS has high configurability and performance, these advantages are relatively irrelevant to a project of our scale. GitHub Pages was ruled out immediately due to its lack of support for Python, our chosen backend language.

## Summary of Choices


The chosen software architecture involves using the Python language and Flask framework for the backend whilst utilising the Vue JavaScript framework along with HTML and CSS to develop the frontend. The external sources of Ticketmaster, Dark Sky, and Google Maps APIs will be used to access the data required for the application.

The Python language was chosen due to its readability, modularity and extensive library support which all improve collaboration whilst streamlining the development process through reusable code. Most limitations of the language had minimal impact due to the small scale of the project as well as the sourcing of data externally, mitigating the need for database support. The Flask web framework was chosen for similar reasons being its ease of use, flexibility and lightweight nature outweighing the more complex and structured Django web framework which had many more built-in features that were deemed unnecessary for this application.

Vue was selected as the JavaScript framework for this application due to it being faster to set up and easier to use compared to React. By using a JavaScript framework as opposed to plain JavaScript, the amount of code required is significantly reduced through the many built in features of Vue such as state management and routing. This simplifies the development process whilst also making the application less fragile and more readable, maintainable and extensible.

Finally, the team elected to use Heroku as the deployment platform for the application due to its support for Python, as well as the rapid deployment and automatic configurations making it easy to set up and use. This allows the team to focus on the development of the application without needing to manually configure and maintain the server.

As with all the choices made in the software architecture for this project, the team prioritised ease of use and reusability whilst avoiding any unnecessary features to mitigate time spent on steep learning curves and the set up process. These decisions were made to streamline the development process, improve collaboration, and reduce code in order to release a fully functioning application in the given timeframe.

In the bottom right corner, there are several overlapping geometric shapes in shades of pink and magenta, including a large square and several triangles, creating a modern, abstract design element.



## Team Organization

The creation of 'Eventmaster' relied on the cooperation and contribution of all 'Houston Rockets' members. In each deliverable, we were faced with many obstacles, where success depended on our teamwork and organizational skills, conceptual and requirement awareness, as well as an ongoing notion of understanding, research and hard work.


### Responsibilities

As the project progressed, different responsibilities were molded and assigned to certain members. Each deliverable was different in their own right, requiring distinct skills and contributions from each member. Luke played the main role of leading the team, where everyone also applied themselves to areas they were needed or assigned to.

Deliverable 1 was split into four parts, though the first part required the collaboration and brainstorming between all members conceptually, in order to build the foundation for our project. Thus, it was organised that one person would complete part 2 (User Stories), two people for part 3 (Low-Fidelity), and one person for part 4 (High-Fidelity).

Deliverable 2 had two parts regarding software architecture and software design, where two people were assigned to each section. Similarly for deliverable 3, two people worked on the web application demo, where two people worked on the slides. This worked well in our group as everyone knew their responsibilities and people could concentrate on what they were working on. Deliverable 4 relied on everyone's contribution, whereby the foundation for the site and API handling was created first, before members could implement certain features they were assigned to.

All in all, the handling of responsibilities of our group were fair and adequate for the skills of group members, which ultimately allowed for a smooth and organised project.

In the bottom right corner, there are decorative geometric shapes. A large pink square is partially visible, with a smaller, darker pink triangle overlapping its bottom-left corner. Below these, a solid blue horizontal bar spans the width of the page.

## Project Fluidity

Despite the ongoing situation concerning COVID-19, the fluidity of our project was only slightly hindered. Given the nature of the project, in person meetings were easily substituted by online consultations and code sharing was done through GitHub. All components were completed on time with desired characteristics, as well as group members consistently communicating to each other about the project, formally and informally.

A main factor to how smooth the project went stems from active conversations and replies between group members, where constant feedback and support allowed for the project to continue moving at the same pace before the transition to online learning occurred.

## Issues

As mentioned before, the impact of COVID-19 had not hindered the project's fluidity, nor its progress, although it definitely affected many personal aspects of life between members. During the project, this transition from the usual face-to-face meetings to online had changed the traditional ways of communication, peer programming, and standups. Luckily, many methods were available in order to substitute these, such as consistent messaging between members for feedback and help through social media platforms, GitHub for code sharing, as well as Microsoft Teams being used for demo practice and feedback.

There were many other minor issues that we had encountered as a team, such as an underwhelming combined understanding of frontend softwares and stacks, as well as python modules and overall experience. This was overcome by putting a lot of work into research and consolidating understanding and ideas in order to formulate a web application that applied to most of what we envisioned.

## Retrospect

In retrospect, other than the finalised web application for deliverable 4, there would not be many changes that we would make in regards to the project. In terms of the finalised web application, we could have started developing the application sooner, looking for more feedback from our tutor much earlier on would have provided better insight on both design and the features of the web app. Not only this, but putting more effort into the web app would mean that many of the features could have been fine tuned, as well as the implementation of more visual interactions, such as tooltips for when a user would hover over a button.

## Reflection

### **Kenneth:**

In the working of this project, I have experienced a team with high organisational skills which allowed for everyone to work to their skills and balanced the pressure and workload out moderately. Throughout the deliverables, I learnt more and more of the processes that software developers must go through in order for a smooth and working finalised product, where every part required major importance as they were valuable foundations for further steps.

It was my first time programming around both frontend and backend, which taught me a lot about the interactions between all sorts of things, such as API's, hosting servers, web clients and more. Overall, it was a positive experience with a finalised web app that I am happy to be a part of.

### **Luke:**

Overall this project has been a positive experience for me as I was able to experience building a web app from the planning stage to the final application. Unlike other assignments I have done in the past, this project gave our team more control over what we were building which I believed increased the enthusiasm of the team as we were implementing features that we would personally want to use in a web app. Throughout this project, I have learnt the importance of the planning stages including detailed user stories and designs which provided a clear guideline on how each feature should work, streamlining the development process. I feel that the project could have been improved if each team member had more prior knowledge or were provided

with more resources regarding front-end development which would have allowed for easier planning of realistic features and the development stack as well as better task allocations among the team. Nevertheless, this project was a valuable experience for me in developing both technical and leadership skills.

**Mubarrat:**

SENG2021 was the first time I worked on a project that required both frontend and backend development. This alone made SENG2021 a valuable learning experience in my programming journey. In addition to this, by going through the necessary planning for the web application, drawing up the storyboard and writing the user stories, I understood the importance of a proper structured plan for each and every feature that will be going into the system. Furthermore, we were allowed to choose what we wanted to build (in the scope of the project specification of course), thus making it a much more enjoyable and freeing experience, rather than having something be imposed on us like most other assignments and projects I've done thus far.

The course could be improved, however, with more frontend resources provided so that everyone could have more knowledge on frontend development before the implementation of the features. All in all, this course was a valuable learning experience that will surely help me in future projects.

**Vicknesh:**

This project was overall a good learning experience as it taught us how to produce a webapp from scratch, including the pre-development steps that are often looked over. It was also the first time I was required to do front-end development, forcing me to learn Vue, Bootstrap, CSS etc. This was also the first time I worked with API's, which was another useful experience. SENG2021 was made even better by the fact that it was very open-ended in terms of what we could do for the project. By picking any project we wanted, we were more motivated to work on the project, as it was something we wanted to create.

However, I believe the course could be improved through the implementation of tutorials or labs that teach us about front-end development as the majority of what we used for the project

ended up being self-taught (which can often result in not knowing the best way to do things). Nevertheless, the course was a great learning experience in webapp development that will be quite useful to future projects as well as the workplace.