Web-based Event Management System

Design Report and Front-End Prototype

IAB207: Rapid Web Application Development Assessment Task 1

Katherine Pyne (n10458166)

Design Report

Local comedy scenes will be revitalized through LaughTrackr, a web-based event management system. This application will capitalize on this niche industry by introducing industry-specific features that are not currently available in mainstream event management systems. Users will be able to perform common tasks like registering accounts, creating and viewing events, purchasing tickets, posting about gigs, and reading comments. This app's comedy-specific features will include marketing tools that comedians can use to share gigs across various social media platforms. Users will also be able to easily determine the tone of different comedy gigs by using rating tags. Lastly, users will have access to comedian profiles and short clips from previous performances to gain a taste of a performance. The goal of LaughTracker is to offer a complete approach to event management by giving functionality to both audience members and industry stakeholders.

A variety of platform options are available for hosting an application. Applications can be either native to a device like iOS, Android, or desktop, or they can be web-based. Applications that were built for a particular platform or operating system and downloaded to the device are known as native applications (Jordan & Das, 2025). Web applications can be accessed through the internet and viewed on any platform or operating system (GeeksforGeeks, 2022).

Hosting a web application has many clear advantages. As mentioned, the clearest advantage is its ability to run on any operating system (OS). This flexibility allows for businesses to reach a vast range of customers (GeeksforGeeks, 2022). Users can access web apps immediately, unlike native apps, as there is no need to download software to the OS (GeeksforGeeks, 2022). In addition to being easily accessible, it doesn't need constant software updates like native apps to keep up with evolving OS updates (GeeksforGeeks, 2022). From a financial perspective, web applications are also incredibly appealing. Implementing applications without tailoring them to a specific OS can be significantly cheaper than using native apps (GeeksforGeeks, 2022). This makes it an appealing option for startup companies. Web apps have their own set of disadvantages, just like any solution. The most obvious disadvantage of web apps is their requirement for an internet connection (GeeksforGeeks, 2022). The application won't work if there's no internet connection. The second issue faced is scalability (GeeksforGeeks, 2022). Web apps should be flexible enough to fit a range of platforms, including small phone screens to ultra-wide monitors, while still being functional and pleasing to use. The lack of website scalability will lead to fewer opportunities for customers to adopt it.

Native mobile applications are an alternative to web-based applications. These applications can be downloaded from a mobile marketplace, such as the Google Play Store for Android devices or the App Store on iOS devices. Native applications can also be found in desktop applications. These apps are similar to mobile apps; different operating systems have their own approved applications that can be downloaded. On Macintosh devices, it's still called the App Store, while on Windows devices, it's called the Microsoft Store. In addition to this, desktop apps can also be downloaded directly from an organization's website, where they will have different versions for each operating system. There are several advantages to creating a native application. Offline

accessibility is its most advantageous feature over web-based apps (Mika & Vasylenko, 2025). This can have a significant impact on user adoption in communities where the internet is not widely reliable, such as rural or regional areas. An app that is specifically tailored for a single operating system can also enhance its performance capabilities (Singh, 2023). Alongside improved performance, users can also have an improved user interface (UI) design (Singh, 2023). By tailoring a particular operating system, developers can make use of platform-specific components that may not be accessible on all platforms (Singh, 2023). A final benefit for native apps is increased security. If the application is downloadable through an application store, then it has the full support of the OS developers to ensure that security procedures are in place to protect the user and the application (Singh, 2023). Like web-based applications, native applications also have their own set of disadvantages. Native applications require a much larger financial cost (Mika & Vasylenko, 2025). The development of platform-specific applications takes longer because each platform has its own requirements that need to be fulfilled. The lack of cross-platform compatibility means that different teams would have to develop the application for different environments (Mika & Vasylenko, 2025). If developers only choose to create an application for one platform, they are then excluding a vast segment of the market from their application. There is also a greater demand for maintenance and upgrades for native apps because they must remain updated with every newly released OS update in order to continue operating (Mika and Vasylenko, 2025).

When it comes to creating an Event Management System, each platform option has its own set of advantages and 6 disadvantages. The most appropriate option for LaughTrackr is a web-based application. For a new startup, the logical decision is to move forward with a more affordable, adaptable, and expansive option. Once the application has begun gaining popularity, developers may then consider the option of a hybrid application. This is a web-based application that is embedded into a native-shell application (Tariq, 2025). This blends the positive characteristics of native-based applications with the ease of web-based applications. However, it will necessitate a bigger financial investment since it will involve similar expenses and time commitment as a built-in application. As a result, it has been decided to proceed with LaughTrackr's initial implementation through a solely web-based application.

User stories:

1. **Title:** View upcoming shows
   Story: As a user, I want to see what upcoming shows are happening in my area so that I can plan the events I attend.
   Acceptance Criteria:
   1. Users should be able to select locations.
   2. Location selection should populate events within a pre-set distance radius.
2. **Title:** View the type of comedy
   Story: As a user, I want to know what types of comedy events are showing because not all comedy aligns with what I find funny.
   Acceptance Criteria:
   1. Each event must have a mandatory rating category.
   2. Category options should be “family friendly”, Mature audiences” and “18+”.
3. **Title:** See ticket availability
   Story: As a user, I want to see ticket availability because it will allow me to ascertain how popular an event is.
   Acceptance Criteria:
   1. Users will be able to see the level of availability for each event.
   2. Levels will include “available”, “limited availability” and “sold out”.
4. **Title:** Book tickets for a show
   Story: As a user, I want to book tickets to shows and know I will have a secured ticket before the event.
   Acceptance Criteria:
   1. Users should be able to see a historical record of their events, upcoming and past events.
   2. Ticket availability should be validated before purchase is able to be made.
   3. Tickets should default at 1 but allow users to buy multiple.
   4. Tickets will appear in their events record as an upcoming event.
5. **Title:** Review shows watched
   Story: As a user, I want to review the shows I have seen because it allows me to feel like my voice is being heard.
   Acceptance Criteria:
   1. Users should be prompted to login to an account in order to review performances.
   2. Users will get a prompt after event dates asking if they would like to leave a review.
   3. Reviews should have a 100 word character limit.
6. **Title:** See snippets of the performer's comedy
   Story: As a user, I want to see short clips of comedians' performances so that I can decide if it is a style of comedy I want to pay to see.
   Acceptance Criteria:
   1. Comedians will have profiles attached to their events.
   2. Users can click on the performer’s profile.
   3. Profiles will show a bio of the performer and short clips of their comedy.

    4. Clips are limited at 5 minutes maximum.
7. **Title:** Create a profile as a performer

   Story: As a user, I want to create my own performer profile because it will help me stand out to potential audience members.

   Acceptance Criteria:
    1. When a user is logged in, there will be a “create” tab. Here users will find the option to create a performer profile
    2. The performer profile will allow the user to write a bio and upload clips of their comedy
8. **Title:** Create an event

   Story: As a user, I want to create events so that my followers know when and where my next shows are.

   Acceptance Criteria:
    1. Under a “create” tab there will be an option to create an event
    2. Mandatory fields must include, location, date and time, performer, cost, venue capacity and rating (18+, family friendly, etc).
9. **Title:** Register an account

   Story: As a user, I want to register my account so that I know when my upcoming gigs are and I can review gigs after I have been to them.

   Acceptance Criteria:
    1. Users can sign up for an account using a valid email and password that conforms to the application security policies
    2. There will be a tab entitled “my events” if a user is not logged in, this tab will prompt them to log in
    3. Once logged in this tab will show their event history

Conceptual Model:

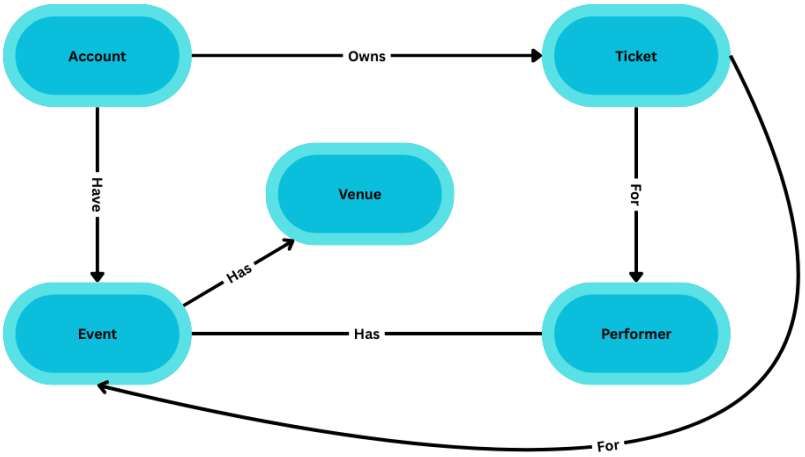
Concepts and Classes Identified from User Stories:

Account	**Event**	History	Create	Email
Password	Date	Time	**Performer**	Cost
Venue	Capacity	Rating	Profile	Clips
Payment	Availability	**Ticket**	Location	Review

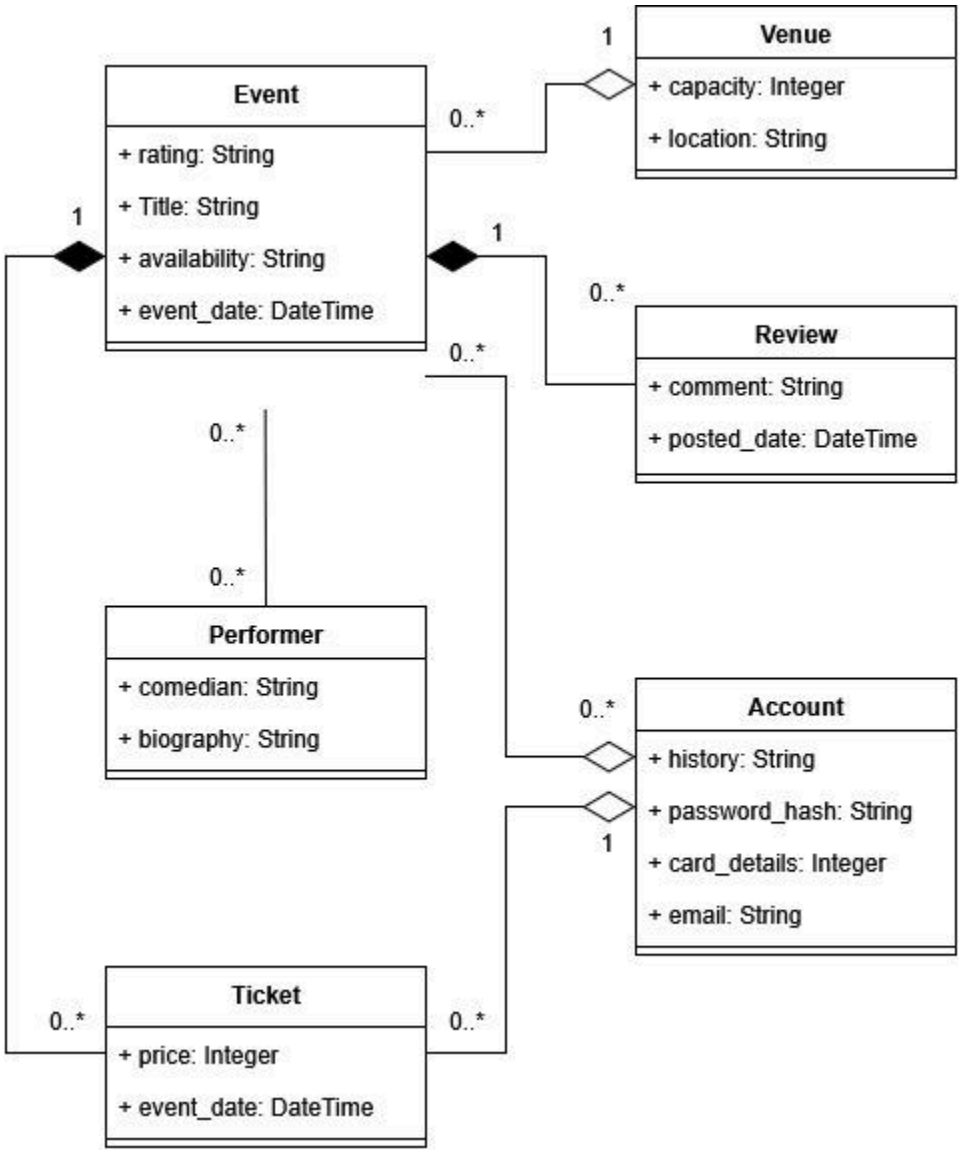
Refined List of Classes:

Account	**Event**	**Venue**	**Performer**	**Ticket**

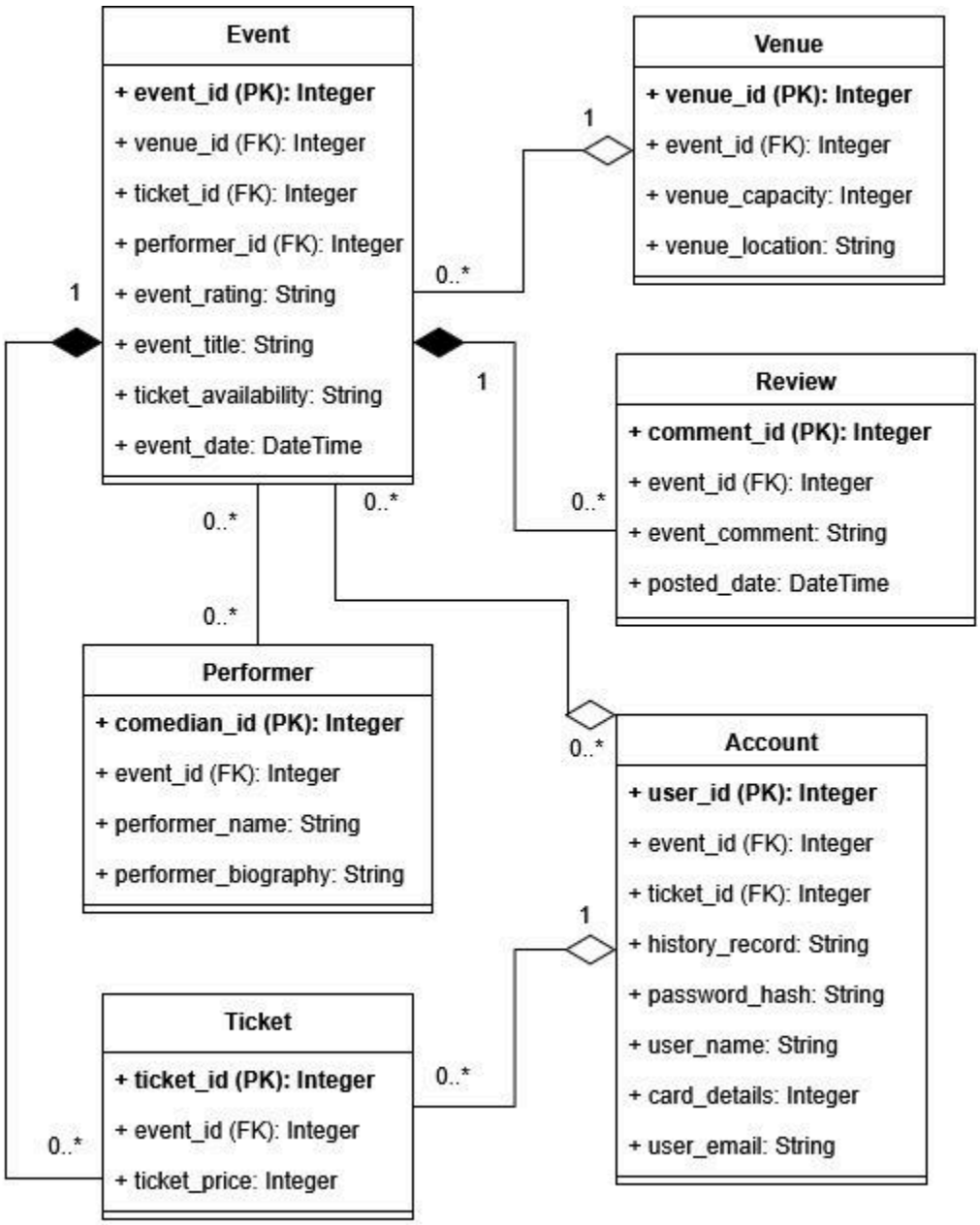
Responsibilities of Classes:



Final Conceptual Model:



Data Model:


Event
+ event_id (PK): Integer
+ venue_id (FK): Integer
+ ticket_id (FK): Integer
+ performer_id (FK): Integer
+ event_rating: String
+ event_title: String
+ ticket_availability: String
+ event_date: DateTime
Venue
+ venue_id (PK): Integer
+ event_id (FK): Integer
+ venue_capacity: Integer
+ venue_location: String
Review
+ comment_id (PK): Integer
+ event_id (FK): Integer
+ event_comment: String
+ posted_date: DateTime
Performer
+ comedian_id (PK): Integer
+ event_id (FK): Integer
+ performer_name: String
+ performer_biography: String
Account
+ user_id (PK): Integer
+ event_id (FK): Integer
+ ticket_id (FK): Integer
+ history_record: String
+ password_hash: String
+ user_name: String
+ card_details: Integer
+ user_email: String
Ticket
+ ticket_id (PK): Integer
+ event_id (FK): Integer
+ ticket_price: Integer
1
0..*
1
0..*
1
0..*
0..*
0..*
0..*
0..*
1
0..*
0..*

References For Part A:

GeeksforGeeks. (2022, December 20). *Advantages and Disadvantages of Web App Development*. GeeksforGeeks. https://www.geeksforgeeks.org/websites-apps/advantages-and-disadvantages-of-web-app-development/

Ginger Software. (2025). *English Grammar & Writing App*. Ginger Software. https://www.gingersoftware.com/grammarcheck

Jordan, B., & Das, G. (2025, July 21). *What Is a Native App? The Straightforward Guide to How It Works and Why It Matters*. AWS in Plain English. https://aws.plainenglish.io/what-is-a-native-app-the-straightforward-guide-to-how-it-works-and-why-it-matters-9e1140cfc778

Mika, A., & Vasylenko, J. (2025, July 7). *Web Application vs. Desktop Application: Difference, Pros & Cons*. Ramotion. https://www.ramotion.com/blog/web-application-vs-desktop-application/#section-desktop-application-advantages

Singh, H. (2023, July 11). *Native Mobile App Development: Pros, Cons, Challenges & Alternatives*. Medium. https://medium.com/debutinfotech/native-mobile-app-development-pros-cons-challenges-alternatives-5b42814f610b

Tariq, L. (2025, August 26). *What Is a Hybrid App? Examples and Benefits*. Adapty. https://adapty.io/blog/what-is-hybrid-app/

References For Part B:

For HTML Formatting:
Long, J. (2017, 04). *Homepage*. Prettier. https://prettier.io/

Bootstrap and CSS elements:
MIT. (2021). *Bootstrap*. Get started any way you want. https://getbootstrap.com/

Refsnes Data. (2025). *CSS Tutorial*. W3Schools. https://www.w3schools.com/css/default.asp

Placeholder Event Images and Details:
Google. (2025). *Image Search*. Google Images.
https://www.google.com/imghp?hl=en-GB&tab=ri&authuser=0&ogbl

McIvor, H. (2025). *Tour*. Randy Feltface. https://www.feltface.com/

Ruane, Z., Kelly, B., & Bonanno, M. (2025). *World Tour*. Aunty Donna.
https://tour.auntydonna.com/

Refsnes Data. (2025). *JavaScript DOM EventListener*. W3Schools.
https://www.w3schools.com/js/js_htmldom_eventlistener.asp

Appendix 1:

I confirm that I used AI in an authorized way (as described by the assignment brief).

Summary of AI use

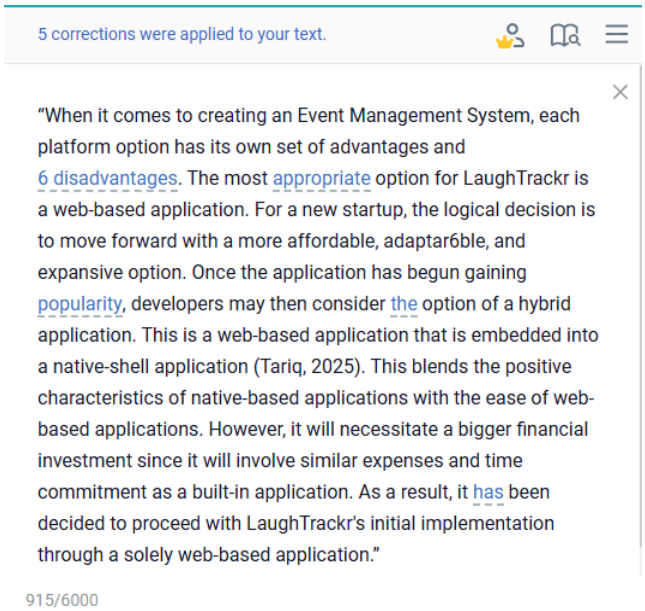
I used the VS Code plug-in called *Prettier* that ensures all output code conforms to a consistent style type. For example, when used it will reformat code with correct line lengths. I also used *Ginger Software*, an English Grammar and Writer application. I input my report paragraphs into this software to check for spelling and grammar errors.

Ginger:

Before Running:

"When it comes to creating an Event Management System, each platform option has its own set of advantages and 6disadvantages. The most apprsopriate option for LaughTrackr is a web-based application. For a new startup, the logical decision is to move forward with a more affordable, adaptar6ble, and expansive option. Once the application has begun gaining popularitdy,, developers may then consider tthe option of a hybrid application. This is a web-based application that is embedded into a native-shell application (Tariq, 2025). This blends the positive characteristics of native-based applications with the ease of web-based applications. However, it will necessitate a bigger financial investment since it will involve similar expenses and time commitment as a built-in application. As a result, it haes been decided to proceed with LaughTrackr's initial implementation through a solely web-based application."

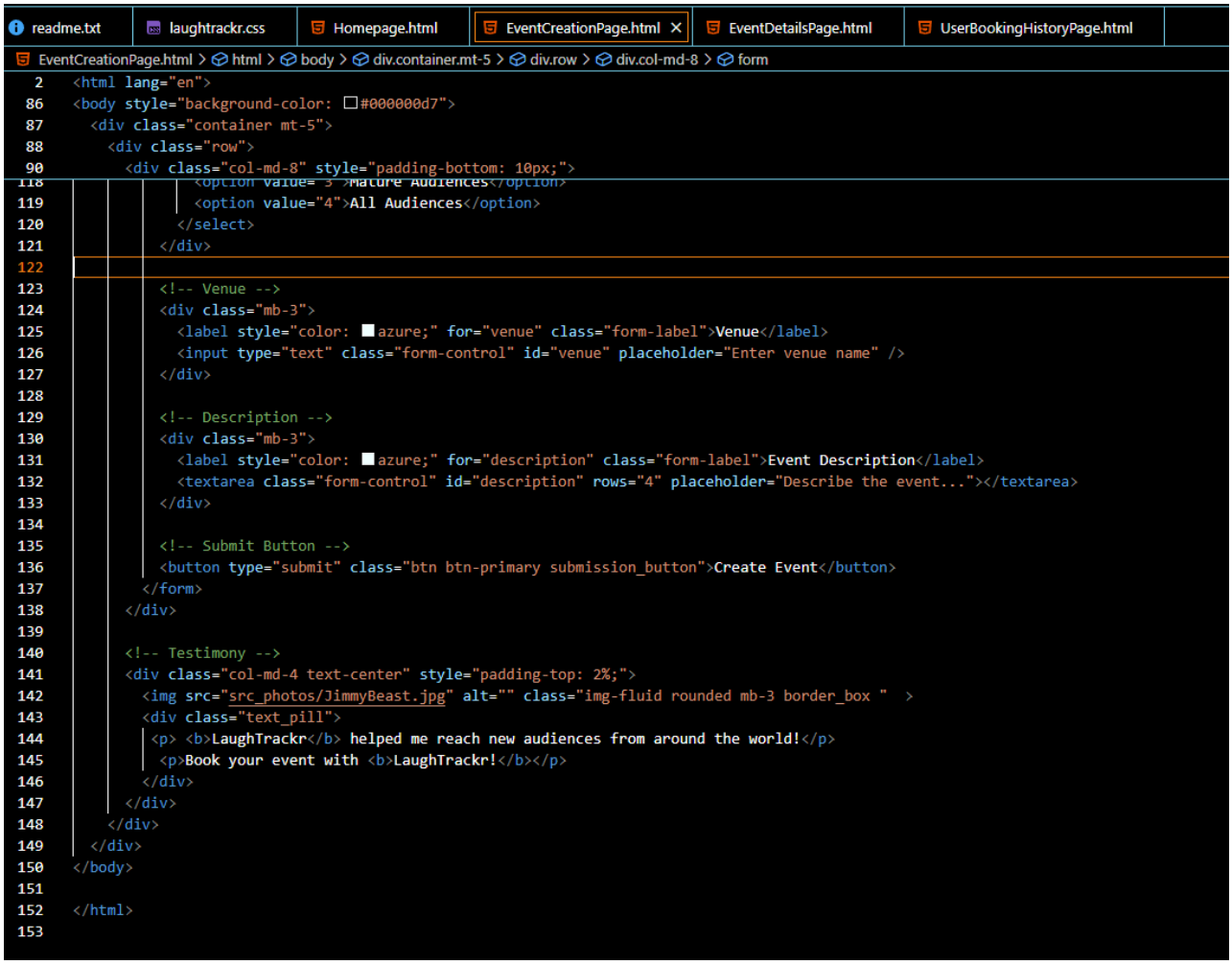
After Running:



Key examples of AI use

Prettier:

Before running:



After Running:

readme.txt | laughtrackr.css | Homepage.html | EventCreationPage.html | EventDetailsPage.html | UserBookingHistoryPage.html

EventCreationPage.html > ...

```
<html lang="en">
  <body style="background-color: #000000d7">
    <div class="container mt-5">
      <div class="row">
        <div class="col-md-8" style="padding-bottom: 10px">

          <!-- Description -->
          <div class="mb-3">
            <label style="color: azure" for="description" class="form-label"
              >Event Description</label
            >
            <textarea
              class="form-control"
              id="description"
              rows="4"
              placeholder="Describe the event..."
            ></textarea>
          </div>

          <!-- Submit Button -->
          <button type="submit" class="btn btn-primary submission_button">
            Create Event
          </button>
        </form>
      </div>

      <!-- Testimony -->
      <div class="col-md-4 text-center" style="padding-top: 2%">
        <img
          src="src_photos/JimmyBeast.jpg"
          alt=""
          class="img-fluid rounded mb-3 border_box"
        />
        <div class="text_pill">
          <p>
            <b>LaughTrackr</b> helped me reach new audiences from around the
            world!
          </p>
          <p>Book your event with <b>LaughTrackr!</b></p>
        </div>
      </div>
    </div>
  </body>
</html>
```