

Cuisine Companion

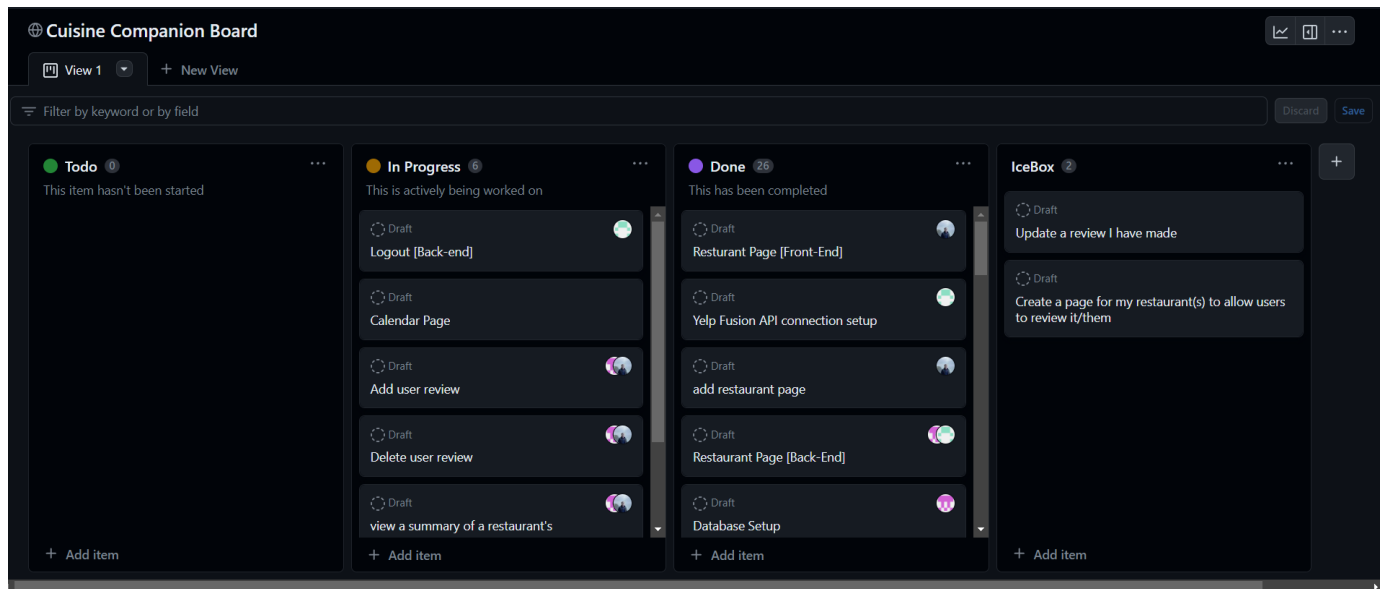
Chloe Cantalupo, Ethan Ansborg, Joel Luciano, Taylore Todd, Hayden Moll, and Rio Parsons

Project Description

We built a website called Cuisine Companion that CU students can use to explore nearby restaurants. The site has a ‘Discover’ page where users can view the most popular and highest ratings, as well as the events and deals going on that day. There is also a ‘Deals’ page that shows the deals going on at local restaurants in the next two days, as well as an ‘Events’ page that shows the same for events. Finally, there is a ‘Calendar’ page that shows all of the deals and events. On the deals, events, and calendar pages the user will be able to filter the results they see. If the user clicks on the restaurant on any of the pages, they will be taken to the restaurant’s page. Here, they can see the restaurant’s address, phone number, and reviews for the restaurant. The user can also leave a review, but they first have to log in or register. If they try to leave a review while not logged in, they will be rerouted to the login page. Users can also delete their old reviews.

Project Tracker

The project board can be seen [here](#). Below is a screenshot of the board.



Video

<https://www.youtube.com/watch?v=dhwPyaspUDg>

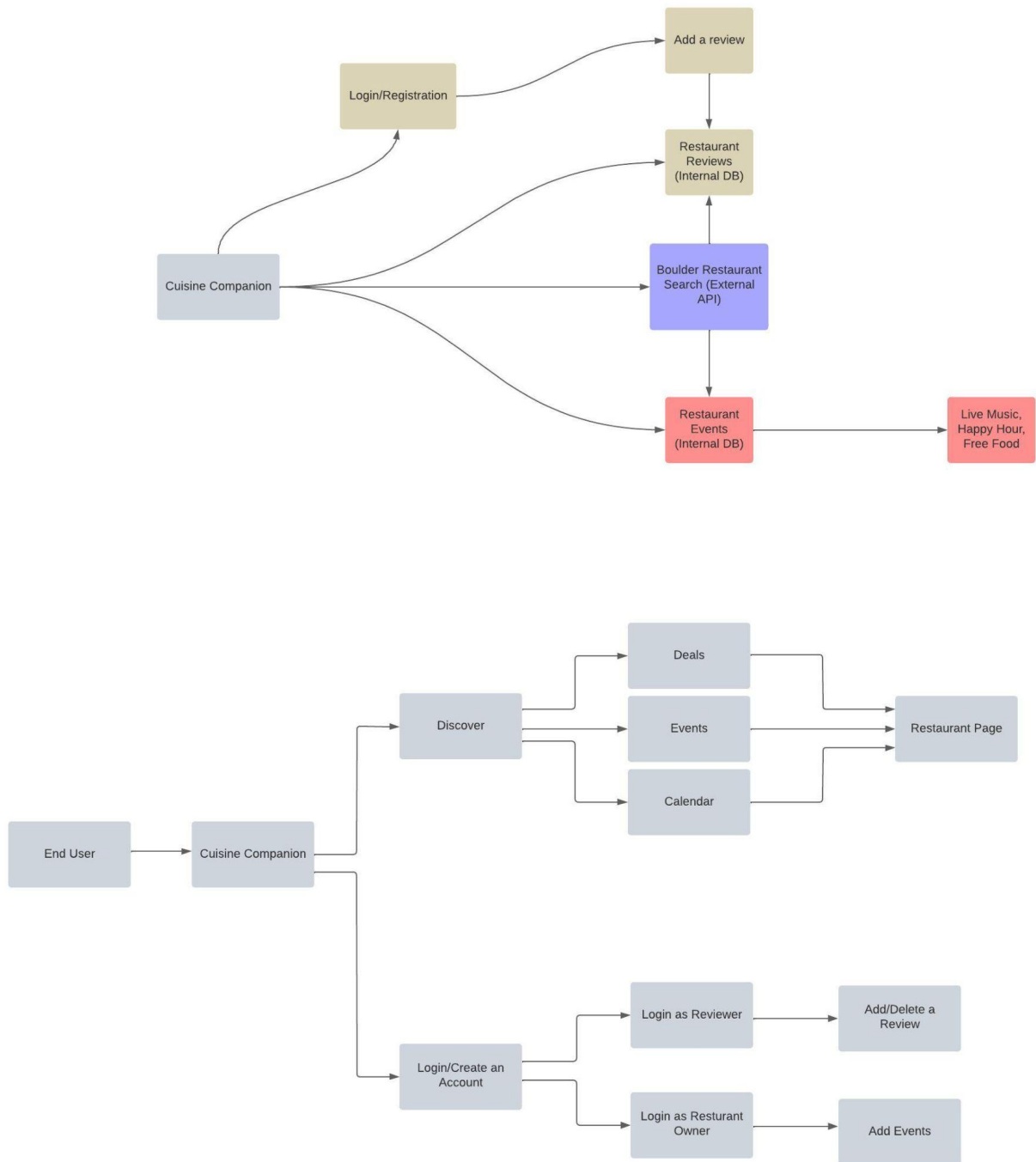
VCS

The project repository can be found [here](#).

Contributions

- **Chloe Cantalupo**
 - I was responsible for developing the front end of the discover page, log in page, and navigation bar. I also created the calendar page including the back and front end functionality by gathering relevant data from the database to populate the page.
- **Ethan Ansborg**
 - I worked on the design of the database, and wrote the code allowing users to post and delete reviews. I worked with Express, EJS, and SQL.
- **Joel Luciano**
 - I worked mostly on the discover page, implementing all of the back-end endpoints that allowed the page to be populated. I also helped a little bit with the front-end of the page, adding some of the EJS code. I also implemented the login/logout and registration endpoints for the website, and generated some of the sample data in the database.
- **Taylor Todd**
 - I worked on the deals page and brainstorming the content we want to display on the discover page. I also gathered the data for the deals database and handled the dynamic date display on the events and deals pages. Additionally, I created the slides for the project presentation.
- **Hayden Moll**
 - I worked on the restaurant page front end design allowing users to view Restaurant data such as the Yelp rating, address, and phone number from the Yelp API and the restaurant name and our rating from our own database. Additionally, users are able to see reviews from our database and post a review to our database.
- **Rio Parsons**
 - I generated the data for events and restaurants and added them to the database. I also formatted the insert statement for deals and built the events page (front and back end). Finally, I also added the filter functionality to deals, events, and calendar.

Use Case Diagram



Test Results

- User Behavior 1: Leaving a review
 - Test Cases
 - Case 1: Leaving a review while logged in
 - Case 2: The user will not be able to leave a review if they are not logged in.
 - Case 3: The user will not be able to leave a review if they do not include all of the required information
 - Case 4: The user will not be able to leave a review if they have already reviewed that restaurant.
 - Results
 - Case 1: Pass, the user is able to leave a review as long as they are logged in
 - Case 2: Pass, if the user tries to leave a review while not logged in, they receive an error message.
 - Case 3: Fail, we realized that we did not implement a check for this and are unable to at this point.
 - Case 4: Pass, the user receives an error message
 - Observations
 - Case 1: The user was able to easily navigate through leaving a review.
 - Case 2: The user was able to log in and then leave a review. However, the tester did note that it was annoying to have to manually select the login page instead of being redirected to it. If we were going to continue working on the page, it would be good to make it automatically redirect to the login page.
 - Case 3: The tester noted that they were able to leave a blank review, then they deleted it.
 - Case 4: The tester received the message and deleted the review to make a new one.
- User Behavior 2: Once a user selects a restaurant from the main page, they should be able to see the following: overall rating, reviews, and ability to leave a review
 - Test Cases
 - The user will select a restaurant with a reviews and rating, and the information will display correctly
 - The user will select a restaurant with a rating, but no reviews, and the information will display correctly
 - The user will select a restaurant with no rating and no reviews, and the information will display correctly
 - Results
 - Pass, the user selected 'Bohemian Biergarten', which has reviews and a rating, and the information displayed correctly
 - Cannot test, there are no restaurants with a rating but no reviews.
 - Cannot test, there are no restaurants with no ratings and no reviews.

- Observations
 - The tester was impressed with the amount of information and appearance of the restaurant page.
- User Behavior 3: Polling the top 3 rated restaurants of the day, week, and month.
 - Test Cases
 - The user will request the top restaurants of the day
 - The user will request the top restaurants of the week
 - The user will request the top restaurants of the month
 - Results
 - These tests could not be executed as we replaced a search function with the discover page. Instead, the Discover page displays the most rated restaurants and the highest rated restaurants of all time.
 - Observations
 - The tester was impressed with the amount of information on the Discover page. They were not disappointed that they could not choose different timeframes.

Deployment

Website Link: <http://recitation-012-team-02.eastus.cloudapp.azure.com:3000/discover>