

Group #8: Travel Assist

Zavier, Jasper, and Joon

Description:

- Problem: Group trips are notoriously hard to plan and coordinate. Keeping track of all the tasks that need to be completed can be incredibly challenging to do across instant messaging.
- Solution: The travel coordinator aids in planning both solo and group trips by providing a platform to keep all your information in one place: dates, locations, budgets, and itineraries can all be viewed and adjusted here.

Key Features:

- Task List: a to-do-list-like feature to assist individuals in planning what tasks need to be done before, during, or after the trip. Optionally adding a modifiable packing list.
- Messaging between group members.
- Budgeting features: Group members can include fees for travel, food, fun, etc.
- Itinerary: Group members can add events/activities they are interested in or intend on participating in, including when, where, what it is, and optional costs.

[Link to tagged repo](#)

[Link to issues/milestones](#)

Meet ✨*The Team*✨

Jasper

- Role: homepage manager
- 3 Assigned Issues:
 - New trip implementation
 - InMemoryTripModel
 - Repository work

Zavier

- Role: messaging manager
- 3 assigned issues:
 - Trips Data
 - Messaging UI
 - Messaging Data

Joon

- Trip-page manager
- 3 Assigned Issues:
 - Trip-page UI
 - Sidebar Connection
 - Todo UI

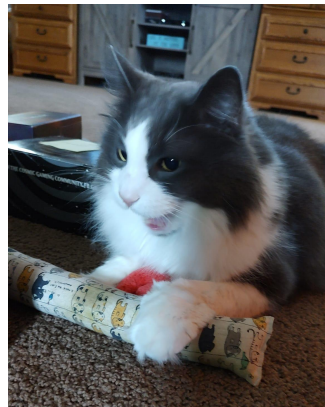
Zavier Morales

Email: zmorales@umass.edu

About me: I'm a Junior in CS, and I'm interested in biotechnology, machine learning, and LOTR! I have some experience in TS/JS, React, Vue, Html, CSS, and Google APIs, but I learn fast and I'm willing to work with any stack we choose.

Favorite quote: "Some infinities are bigger than other infinities." — John Green, *The Fault in Our Stars*

Reflection: Coordination for trips has always been a headache for friends and I, especially agreeing on times, tickets, pricing, and even finding a time to meet up to talk about the trip. This app seems like a really good idea to centralize all of these problems, and also it includes a range of tools to implement, which sounds fun.



Jasper McCormack (He/Him)

caleighmccor@umass.edu

I'm a transfer student from Holyoke Community College and in my Junior year. I've taken CS 220, have some HTML experience, and have worked with GUI applications in Java already. I have an interest in visual design, digital or physical artwork, and website development.

I believe that a travel planner is a fun and interesting application to build since I personally have trouble organizing my personal plans and budgeting for trips, and coordinating with other people to complete tasks can be incredibly difficult with generic messaging and calendar apps. A consolidated application would ease a lot of stress.



I don't know anything with any certainty, but the sight of the stars makes me dream"
- Vincent Van Gogh

Assigned Work Summary - Joon

In this milestone, I focused on developing Trip Page to Frontend Component structure, updated backend to restore the data. Fix the main problem of my code didn't saved on data.

Pull request:

<https://github.com/BakingPancakes/Travel-Assistant/pull/63>

<https://github.com/BakingPancakes/Travel-Assistant/pull/67>

Issues:

<https://github.com/BakingPancakes/Travel-Assistant/issues/6>

Feature Demonstration - Joon

My Trips				+ New Trip
Trip	Who's Going?	When?	Actions	
123	123 + 123	May 8 - May 14, 2025	Edit	✕
123	123 + 123	May 8 - May 14, 2025	Edit	✕
123	123 + 123	May 8 - May 14, 2025	Edit	✕
nono	joon + oon	May 28 - Jun 5, 2025	Edit	✕
nono	joon + oon	May 28 - Jun 5, 2025	Edit	✕

I added new trip form to write and we can list on the table.

After saved, we can change our data for specific detail

Accommodation

Traveler	Type	Price	Actions
No accommodations added yet.			

Add Accommodation
Traveler:
123
Type:
Hotel (\$200)
[Add Accommodation](#)

Budget Summary
Total Budget: \$0

To-Do List

Before Trip
No tasks added yet.

[Add](#)

During Trip
No tasks added yet.

[Add](#)

After Trip
No tasks added yet.

[Add](#)

[Save Trip Details](#)

Code Explanation - Joon

```
selectTrip(tripId) {
  this._selectedTripId = tripId;

  // get data
  const savedTripsJson = localStorage.getItem('savedTrips');
  if (!savedTripsJson) return;

  const savedTrips = JSON.parse(savedTripsJson);
  if (!savedTrips[tripId]) return;

  this._tripData = savedTrips[tripId];

  // update travel section
  this._container.querySelector('#selected-trip-name').textContent =

  // show section
  this._container.style.display = 'block';

  // publish event
  this._hub.publish(Events.TRIP_SELECTED, attachEventListeners() {
```

```
  // todo add button
  const addTodoButtons = this._container.querySelectorAll('.add-todo-btn');
  addTodoButtons.forEach(btn => {
    btn.addEventListener('click', () => {
      const section = btn.getAttribute('data-section');
      const inputId = `${section}-todo-input`;
      this.addTodoItem(section, inputId);
    });
  });

  // key event for add todo field
  ['before', 'during', 'after'].forEach(section => {
    const input = this._container.querySelector(`#${section}-todo-input`);
    input.addEventListener('keypress', (e) => {
      if (e.key === 'Enter') {
        this.addTodoItem(section, `${section}-todo-input`);
      }
    });
  });

  // subscribe event
  this._hub.subscribe(Events.TRIP_SELECTED, (data) => {
    this.loadTodoItems(data.tripId);
  });
});
```

When you click "New Trip":

- An event called `CREATE_NEW_TRIP` is triggered
- TripInputComponent shows a form
- You enter trip details and save

When you add accommodations or to-dos:

- The information is saved in memory
- When you click "Save Trip Details", all changes are stored in `localStorage`

Challenges and Insights- Joon

I am still having trouble that when I add trip it is keep shooting infinite loop but couldn't find out the code is still working but it doesn't work well.

Future Improvements & Next Steps - Joon

I still worked on backend my server would open but it doesn't work on css stuff. I felt little busy last few weeks but still you can check my pull request.

Timeline - Joon

may 1st:

- Meet teammates and discuss our plan.

may 6th:

- Updated budgets, todo list, accommodations
- Added local storage

May 7th:

- Tried to imply backend.

Assigned Work Summary - Xavier

- Frontend integration with repository service
 - Connect UI with
- Test frontend interactivity with mock data in fake service
 - Loading user data
 - Loading chat data associated with user, then displaying chat data in contacts list and storing locally
- Pretty up Messages UI
 - Modified css to create consistent formatting between pages
- Pull requests:
 - <https://github.com/BakingPancakes/Travel-Assistant/pull/64>
 - <https://github.com/BakingPancakes/Travel-Assistant/pull/66>
- Closed issues
 - <https://github.com/BakingPancakes/Travel-Assistant/issues/52>

Timeline - Xavier

- May 1st
 - Meet with teammates, organize our next big tasks to accomplish within the next week for demo day.
- May 6th
 - Implement code for user and chat retrieval between interface and backend.
- May 7th
 - Debug user and chat retrieval and displaying chat information

Feature demonstration - Xavier

<https://zoom.us/clips/share/btIK34nLR-KbMRYGvIW5FQ>

The above shows the mock chat data functionality, which fetches the user data and associated chat data, stores chat data locally, and displays chat data.

Code Explanation- Xavier

Challenges/Insights - Xavier

- The biggest challenge was confidently integrating frontend with backend, given that we were not sure how to begin debugging issues within our backend.

Future Improvements/Next Steps - Xavier

- Improved message interactivity including sending messages between users, saving messages to server, and allowing trips to be edited within messages
 - The issue with dedicated user profiles is the need to create/retrieve a given user profile from the server, then create an eventhandler in [main.js](#) to provide that user's credentials to all page components so that those pages can retrieve the user's respective data.
 - Sending messages between users would also only work within the same browser.
- Integrating trip data between pages/views
-

Assigned Work Summary - Jasper McCormack

- Dynamic Updates of Homepage Tables
- Basic SQLite Implementation
 - Quickly wrote implementation of a SQLiteTripModel to use for backend storage.
- Pull Requests:
 - Have committed and merged content, but pull requests were not created for some reason.
- Closed Issues:
 - <https://github.com/BakingPancakes/Travel-Assistant/issues/40>
 - <https://github.com/BakingPancakes/Travel-Assistant/issues/37>
 - <https://github.com/BakingPancakes/Travel-Assistant/issues/36>
 - <https://github.com/BakingPancakes/Travel-Assistant/issues/31>
 -

Assigned Work - Jasper McCormack

- Outstanding issues:
 - There are a few outstanding issues given we didn't complete since we ran out of time to implement everything, including profiles and user verification.

Timeline - Jasper

- April 29:
 - Began completing multi-view, but was waiting for other teammates to finish creating components of their pages, did some troubleshooting.
- May 7:
 - Homepage tables now update alongside trip updates
 - Installed SQLiteTripModel
 - Multiview now seamlessly transitions between pages

Feature Demonstration - Jasper

Feature: Homepage Table Display

As trips are added to the application, the tables on the homepage update to give the user a brief overview of what trips they are participating in and what tasks they need to complete.

The basic functionality of the front end is implemented successfully, aside from profiles and users. The back-end is written, but has not been fully tested in regards to whether or not it functions correctly.

Front-End - Jasper McCormack

Trip			Who's Going?		When?	
Beach			Zavier, Joon		May 13 - May 14, 2025	

Back-End - Jasper McCormack

```
class _SQLiteTripModel {
  constructor () {}

  async init(fresh = false) {
    await sequelize.authenticate();
    await sequelize.sync({ force: true });
    // An exception will be thrown if either of these operations fail.

    if (fresh) {
      await this.delete();

      await this.create({
        // Create mock data here
      });
    }
  }

  async create(trip) {
    return await Trip.create(trip);
  }

  async read(id = null) {
    if (id) {
      return await Trip.findByPk(id);
    }

    return await Trip.findAll();
  }

  async update(trip) {
    const tripu = await Trip.findByPk(trip.id);
    if (!tripu) {
      return null;
    }

    await tripu.update(trip);
    return tripu;
  }
}
```

```
// Define the trip model
const Trip = sequelize.define("Trip", {
  id: {
    type: DataTypes.UUID,
    defaultValue: DataTypes.UUIDV4,
    primaryKey: true,
  },
  name: {
    type: DataTypes.STRING,
    allowNull: false,
  },
  destination: {
    type: DataTypes.STRING,
    allowNull: true,
  },
  traveler: {
    type: DataTypes.STRING,
    allowNull: true,
  },
  companions: {
    type: DataTypes.STRING,
    allowNull: true,
  },
  from: {
    type: DataTypes.STRING,
    allowNull: false,
  },
  to: {
    type: DataTypes.STRING,
    allowNull: false,
  },
  budget: {
    type: DataTypes.INTEGER,
    allowNull: true,
  },
  accommodations: {
    type: DataTypes.ARRAY(DataTypes.JSON),
    allowNull: true,
  },
  todoItems: {
    type: DataTypes.ARRAY(DataTypes.JSON),
    allowNull: true,
  },
});
```

Challenges/Insights - Jasper

- All in all, the backend was extremely challenging to understand and implement. In addition, since we all had different ideas for the project, it did lead to some challenges within connecting our different work.
- Given that this was also my full-stack application, I gained a lot of crucial insight in how to better plan, code, and execute future projects. I greatly enjoyed working on the project, and now I want to begin working on a personal project over the summer.
- Overall, the project was a valuable learning experience, and I believe that it helped me better understand how group work will go in the future.