FRONT END UI/UX PROJECT REPORT

----------------------------------------------------

**Project Title:** Holiday Planner

**Submitted By:**

* Aadithya Vimal (2462302)
* Vadapalli Lakshmi Kalyani Sahithi (2462370)
* Sacchit Kallara Viju (2462356)

**College Email ID:**

* aadithya.vimal@btech.christuniversity.in
* vadapalli.lakshmi@btech.christuniversity.in
* sacchit.kallara@btech.christuniversity.in

**Course:** UI/UX Design Fundamentals

**Instructor Name:** Ms. Nagaveena

**Institution:** Christ University

**Date of Submission:** 25/09/2025

**2. Abstract**

This project presents the design and development of an interactive, single-page Holiday Planner application built using **HTML, CSS, and JavaScript**, with the **Bootstrap 5 framework** as its foundation. The application provides a seamless user experience for planning vacations by allowing users to dynamically search for destinations, select accommodation and transport, and choose activities. Key features include a real-time budget tracker and an automatically generated itinerary that update instantly based on user selections. The user interface is designed with a modern, dark-themed aesthetic inspired by contemporary web design trends, ensuring full responsiveness across desktops, tablets, and mobile devices. This project demonstrates core front-end development skills, including DOM manipulation, event handling, and client-side state management, to deliver a functional and visually engaging planning tool.

**3. Objectives**

1. **Develop an Interactive Planning Interface:** Create a user-friendly platform where users can easily search for destinations and select holiday components like accommodation, transport, and activities.
2. **Implement Dynamic Content Filtering:** Allow users to search for destinations by name or filter them by interest (e.g., beach, mountains, city) for a personalized experience.
3. **Provide Real-Time Budget Tracking:** Automatically calculate and display a running total of expenses as the user makes different selections, offering immediate financial clarity.
4. **Generate an Automatic Itinerary:** Construct and display a simple, clear itinerary based on the activities chosen by the user for their selected destination.
5. **Design a Modern and Responsive UI:** Build a visually appealing, CRED-inspired dark-mode interface that provides a consistent and accessible experience on all screen sizes.
6. **Ensure Client-Side Functionality:** Utilize JavaScript to handle all application logic, including state management, DOM updates, and user interactions, without requiring a backend.
7. **Offer a Simple Reset Option:** Include a feature to clear all user selections at once, allowing for quick and easy replanning.

**4. Scope of the Project**

This project focuses on creating a **client-side front-end application** for holiday planning. The scope includes designing and building the user interface with **HTML5 and CSS3**, implementing interactivity with **JavaScript**, and ensuring responsiveness using the **Bootstrap 5 framework**. The application's features are limited to searching and selecting from a predefined list of destinations stored in a JavaScript array. It dynamically generates an itinerary and calculates a budget based on user inputs. The scope **does not include** backend integration, user authentication, database storage for saving plans, or real-time data fetching from external APIs for flights or hotels. All functionality is self-contained within the front-end code.

**5. Tools & Technologies Used**

* **HTML5:** For the core structure and semantic markup of the web page.
* **CSS3:** For custom styling, animations, gradients, and responsive design using media queries.
* **JavaScript (ES6):** For DOM manipulation, event handling, and all client-side application logic.
* **Bootstrap 5:** As the primary CSS framework for the grid system, components, and responsive layout.
* **jQuery:** Used as a dependency for certain Bootstrap components and for simplified DOM manipulation.
* **Visual Studio Code:** As the primary code editor for development.

**6. HTML Structure Overview**

* Used semantic tags: <header>, <section>, <body>, and <footer> elements implied through sectioning.
* Leveraged the **Bootstrap grid system** (container, row, col-md-6, etc.) for a responsive column-based layout.
* Structured the page into logical, reusable sections: **Hero, Destination Search, Accommodation & Transport, Activity Planner,** and **Itinerary & Budget**.
* Utilized interactive form elements like <input>, <select>, <button>, and radio buttons to capture user choices.
* Assigned unique IDs to elements for easy targeting with JavaScript for dynamic content updates.

**7. CSS Styling Strategy**

* Used a single external CSS file (styles.css) to organize all custom styles.
* Employed **CSS custom properties (variables)** in :root for a consistent dark-themed color palette and easy maintenance.
* Customized and extended Bootstrap 5 classes to achieve a unique, high-contrast aesthetic.
* Techniques used:
  + **Flexbox** for alignment within components.
  + **Linear Gradients** for backgrounds on the hero section and cards.
  + **Media Queries** for fine-tuning responsiveness on smaller devices.
  + **Transitions and Transforms** for subtle hover effects on interactive elements.

**8. Key Features**

* **Dynamic Destination Search:** A dual-function search that allows users to find destinations either by typing a name or by selecting an interest from a dropdown filter.
* **Interactive Planning Module:** Users can select accommodation, transport, and activities, with immediate visual feedback (e.g., selected activity cards are highlighted).
* **Real-Time Budget Tracker:** A budget component that instantly updates the cost breakdown and total amount as the user makes or changes selections.
* **Auto-Generated Itinerary:** A dedicated section that builds a simple day-by-day itinerary based on the activities the user has chosen.
* **Responsive Dark-Mode UI:** A modern, visually engaging interface that adapts seamlessly from large desktop monitors to small mobile screens.
* **One-Click Reset:** A "Reset Plan" button that clears all selections, the itinerary, and the budget, allowing the user to start over effortlessly.

**9. Challenges Faced & Solutions**

| Challenge | Solution |
| --- | --- |
| **Managing Application State** | Keeping track of the user's selected destination, activities, and costs was complex. We solved this by using simple **JavaScript variables** (selectedDestination, selectedActivities, budget object) to act as a central "state." All functions that updated the UI would read from this state, ensuring data consistency across the application. |
| **Dynamic DOM Manipulation** | Efficiently rendering and updating lists (destinations, activities, itinerary) without page reloads was a key challenge. This was addressed by writing dedicated rendering functions that use **JavaScript template literals** to build HTML strings from data arrays. Event listeners were re-attached to newly created elements to maintain interactivity. |
| **Ensuring UI Consistency** | Achieving a polished, modern dark-themed UI with consistent spacing, colors, and interactivity required careful styling. We utilized **CSS custom properties** for the color palette and created reusable custom classes (e.g., .bg-gradient-card) to apply consistent styles, which streamlined the process and made the CSS easier to manage. |
| **Responsive Layouts** | Making the multi-column layout for activities and itinerary/budget sections work on mobile was difficult. We used **Bootstrap's responsive grid system** (col-md-\*, col-sm-\*) as the foundation and added custom **CSS media queries** to override styles for smaller screens, such as stacking columns vertically and adjusting font sizes. |

Export to Sheets

**10. Outcome**

* Successfully developed a fully functional, interactive front-end application for planning holidays.
* The final product features a clean, modern, and visually appealing dark-mode UI that is fully responsive.
* All core objectives, including dynamic search, real-time budgeting, and itinerary generation, were achieved using only client-side technologies.
* Gained in-depth practical experience in JavaScript for DOM manipulation, client-side state management, and event handling within a structured application.

**11. Future Enhancements**

* **Backend Integration:** Connect the application to a backend server and database (e.g., using Node.js and MongoDB) to allow users to save and retrieve their holiday plans.
* **User Authentication:** Implement a login/signup system so users can have personal accounts and manage multiple saved itineraries.
* **Third-Party API Integration:** Fetch real-time data from external APIs for live flight prices, hotel availability, and weather forecasts for destinations.
* **Export and Share:** Add functionality to export the final itinerary and budget as a PDF file or share it with others via a unique link.
* **Interactive Map View:** Integrate a map API (like Google Maps or Leaflet) to show destination locations and visualize the planned trip.

**12. Sample Code**

This JavaScript function is central to the application's logic. It calculates the total budget by checking which accommodation and transport options are selected and by counting the number of chosen activities. It then updates the HTML to display the cost breakdown and the final total in real-time.

JavaScript

// Budget Tracker

function updateBudget() {

// Accommodation

const accRadio = document.querySelector('input[name="accommodation"]:checked');

if (accRadio) {

if (accRadio.value === 'hotel') budget.accommodation = 150;

else if (accRadio.value === 'resort') budget.accommodation = 250;

else if (accRadio.value === 'hostel') budget.accommodation = 50;

else budget.accommodation = 0;

} else {

budget.accommodation = 0;

}

// Transport

const transRadio = document.querySelector('input[name="transport"]:checked');

if (transRadio) {

if (transRadio.value === 'flight') budget.transport = 400;

else if (transRadio.value === 'train') budget.transport = 200;

else if (transRadio.value === 'car') budget.transport = 300;

else budget.transport = 0;

} else {

budget.transport = 0;

}

// Activities

budget.activities = selectedActivities.length \* 100;

// Render

budgetItems.innerHTML = `

<div class="budget-item"><span>Accommodation</span><span>$${budget.accommodation}</span></div>

<div class="budget-item"><span>Transport</span><span>$${budget.transport}</span></div>

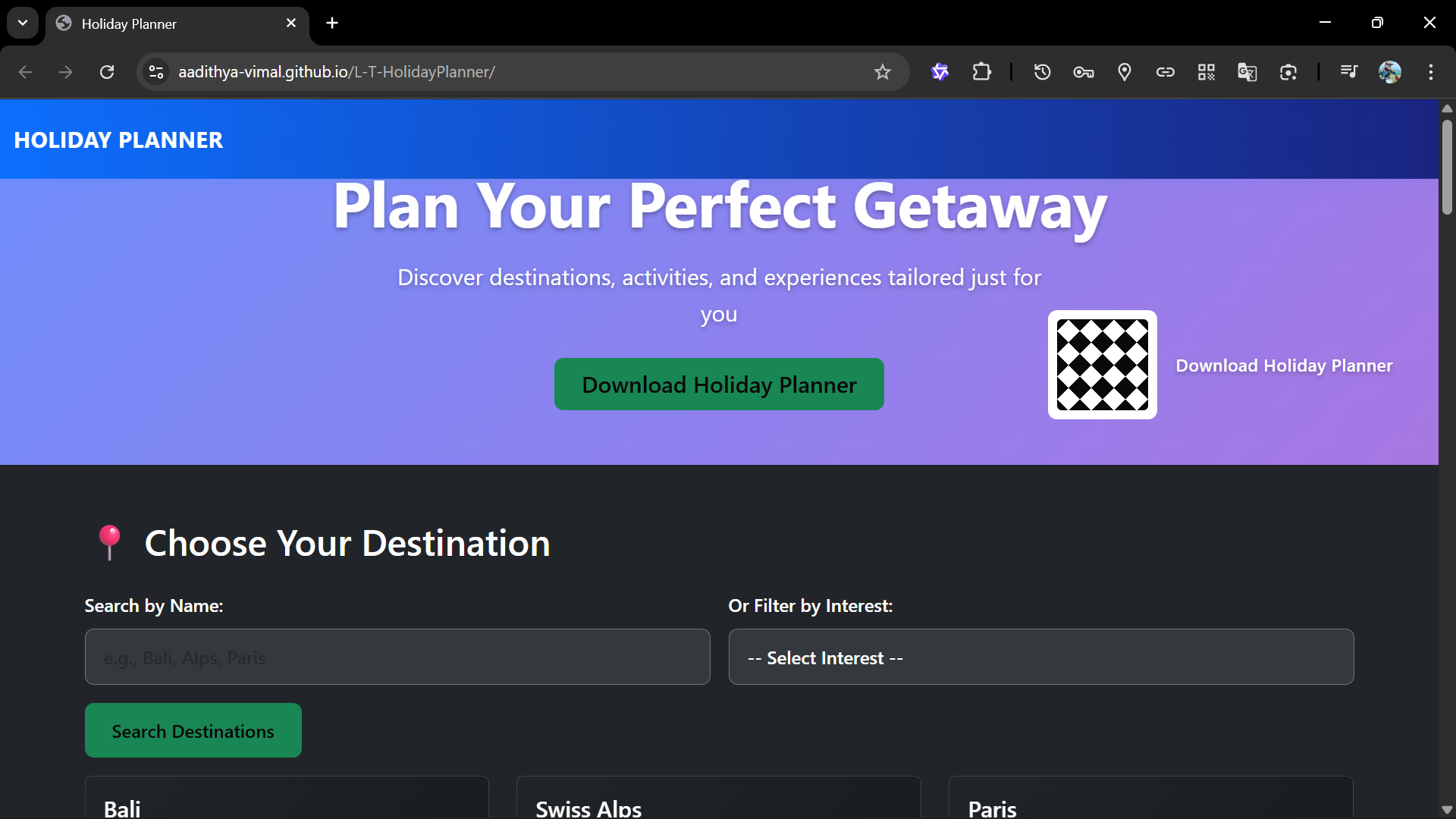
<div class="budget-item"><span>Activities</span><span>$${budget.activities}</span></div>

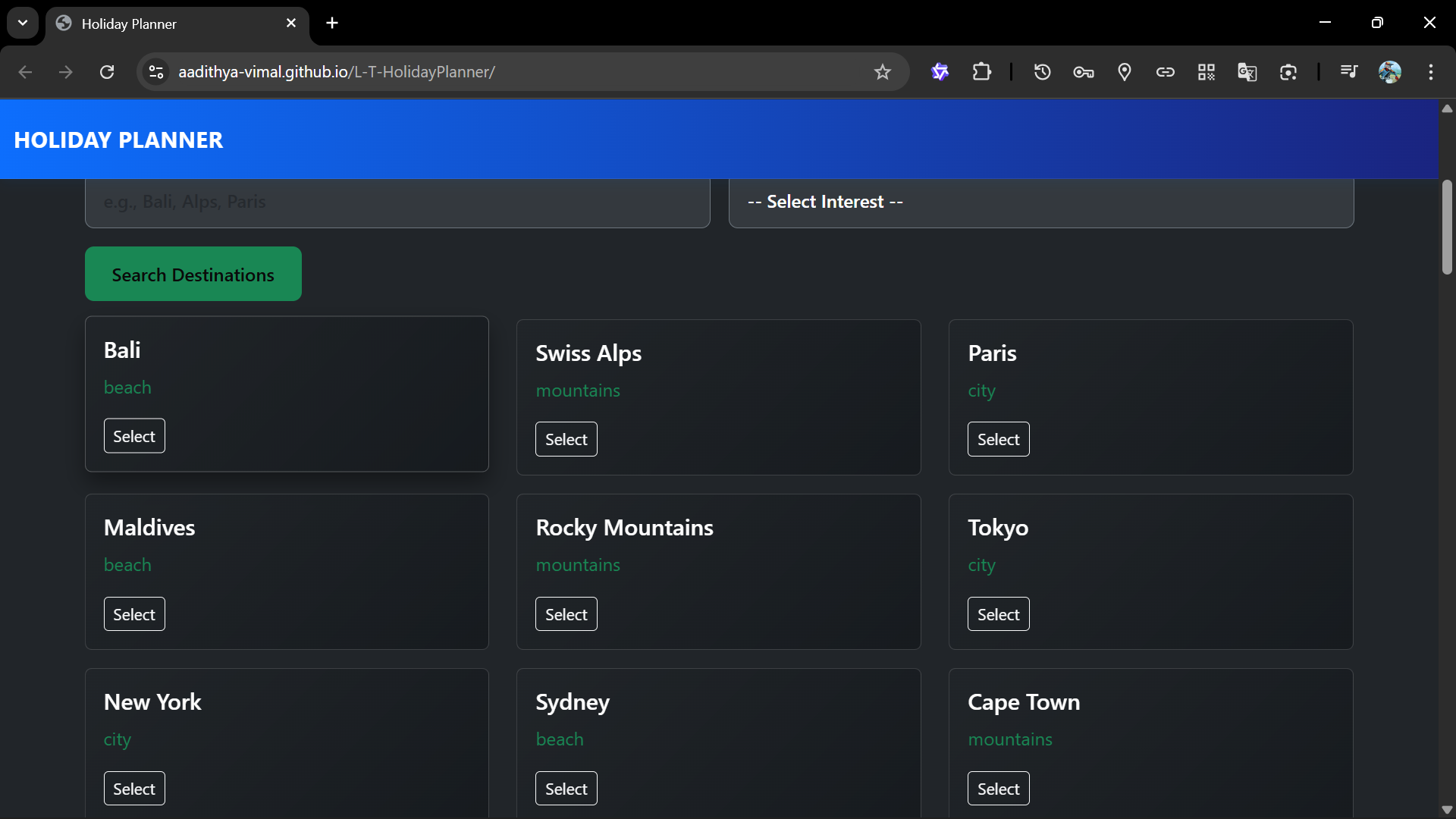
`;

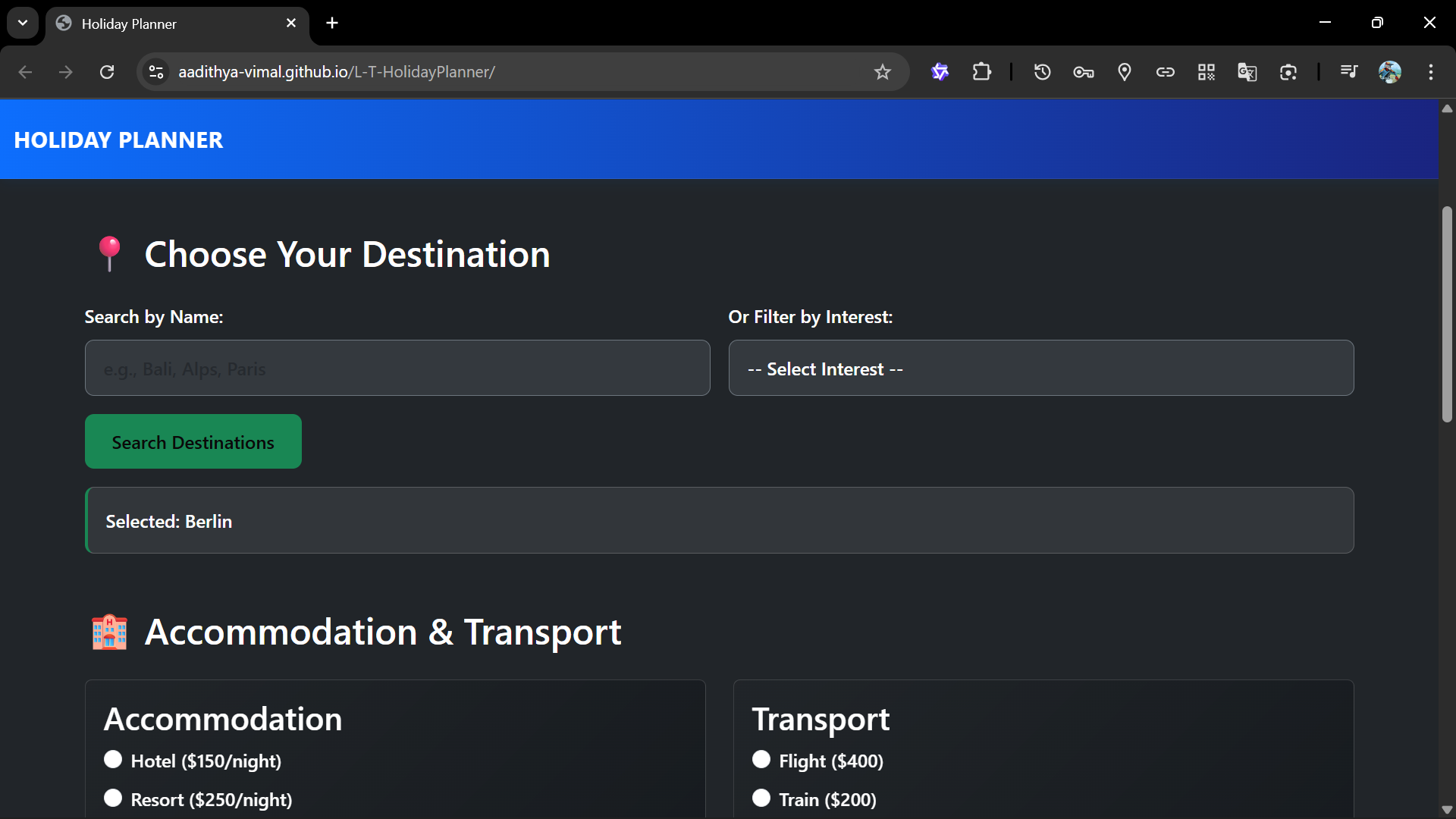
totalAmount.textContent = budget.accommodation + budget.transport + budget.activities;

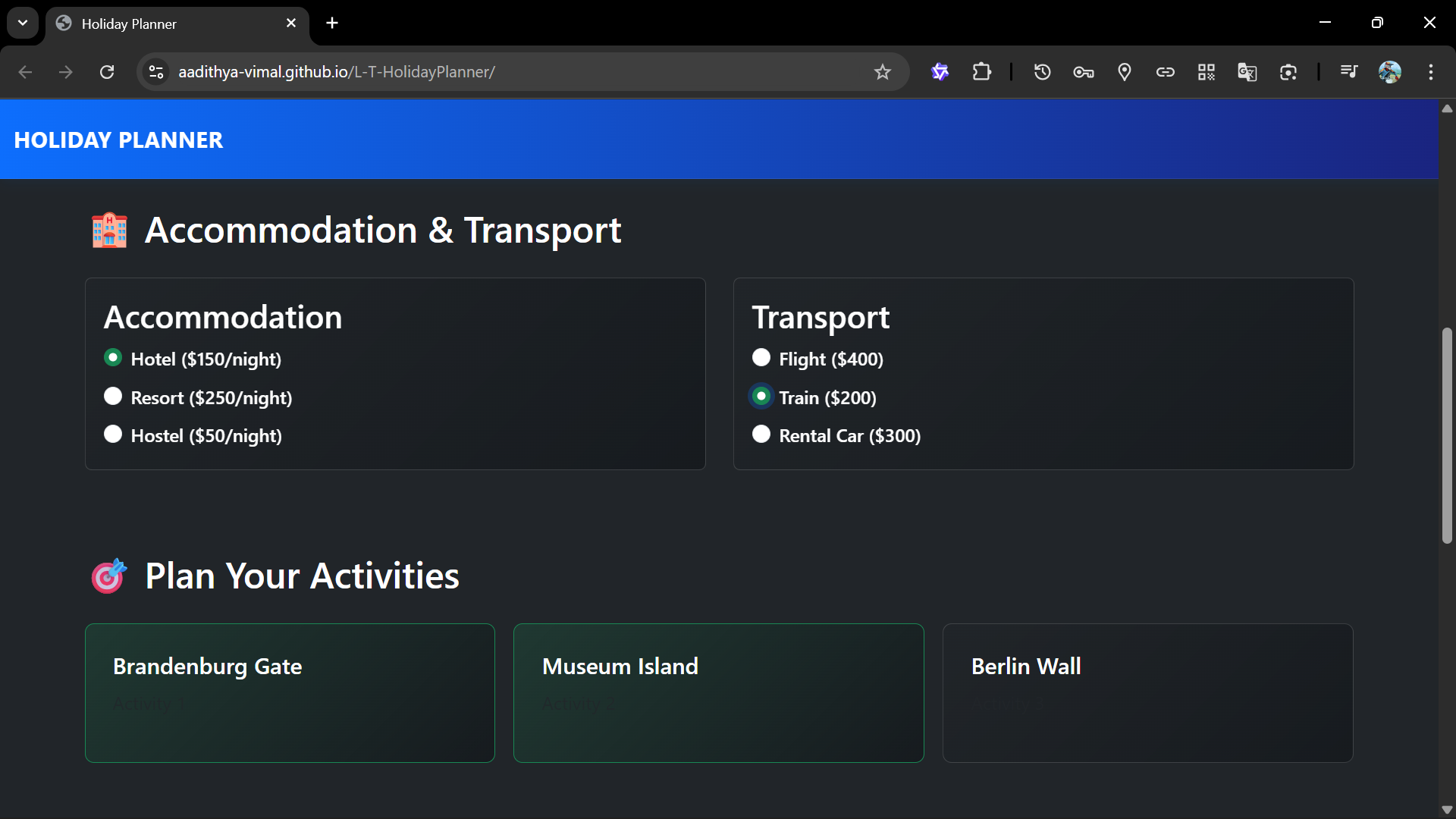
}

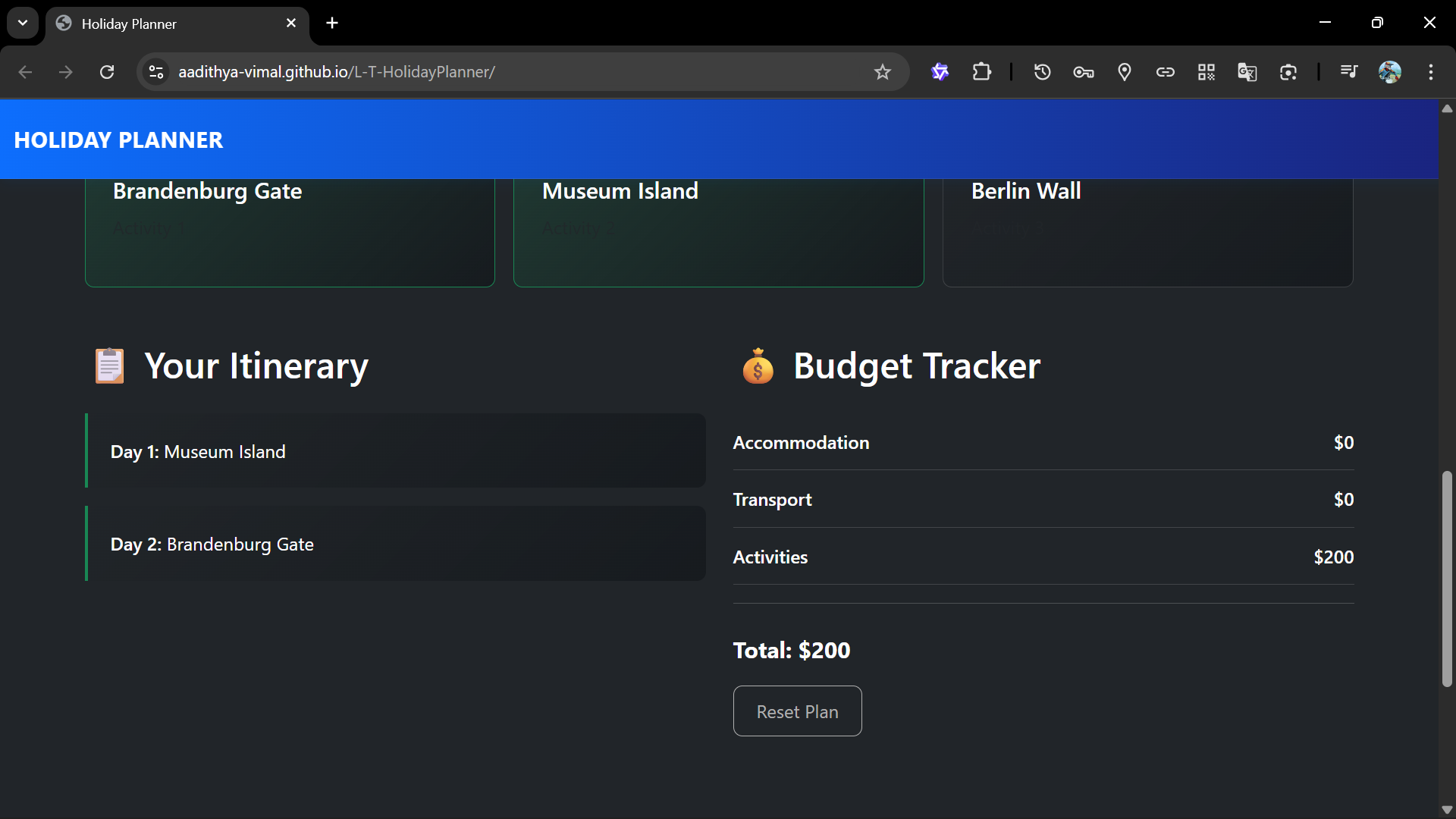
**13. Screenshots of Final Output**

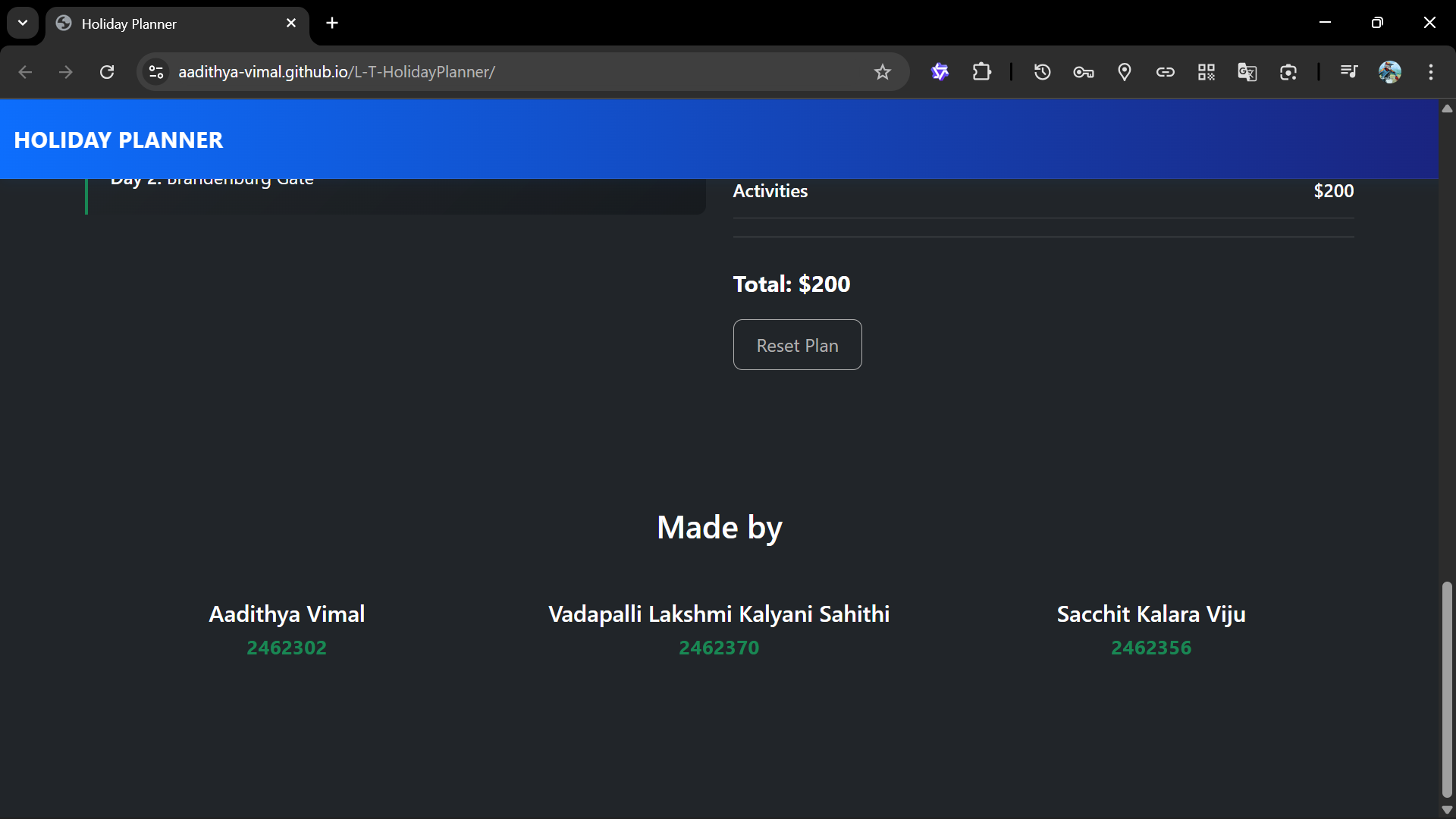












**14. Conclusion**

The Holiday Planner project successfully achieves its goal of creating an intuitive, dynamic, and visually engaging platform for trip planning. By effectively integrating **HTML, CSS, and JavaScript with the Bootstrap framework**, the application provides a seamless user experience with valuable features like real-time budget tracking and automatic itinerary generation. The project demonstrates a strong understanding of front-end development principles, including responsive design, DOM manipulation, and client-side logic. While it currently operates as a standalone front-end tool, it serves as a robust prototype and a solid foundation for future expansion into a full-stack, data-driven web application with enhanced features like user accounts and API integration.

**15. References**

* **MDN Web Docs:** For comprehensive documentation on HTML, CSS, and JavaScript.
  + <https://developer.mozilla.org/>
* **Bootstrap 5 Documentation:** For official documentation on framework classes, components, and utilities.
  + <https://getbootstrap.com/docs/5.3/>
* **jQuery API Documentation:** For reference on jQuery methods and syntax.
  + <https://api.jquery.com/>