

---

# **Travel Agency Website - GoWander**

## **● System Design Document**

---

**Version**

**1.0**

**05/17/20**

**25**

# Table of Contents

<b>1. Introduction.....</b>	<b>iv</b>
1.1    Purpose of the SDD.....	iv
<b>2. General Overview and Design Guidelines/Approach.....</b>	<b>v</b>
2.1    General Overview.....	v
2.2    Assumptions/Constraints/Risks.....	v
2.2.1    Assumptions.....	v
2.2.2    Constraints.....	vi
2.2.3    Risks.....	vi
<b>3. Design Considerations.....</b>	<b>viii</b>
3.1    Goals and Guidelines.....	viii
3.2    Development Methods & Contingencies.....	viii
3.3    Architectural Strategies.....	ix
3.4    Performance Engineering.....	x
<b>4. System Architecture and Architecture Design.....</b>	<b>xi</b>
4.1    Logical View.....	xi
4.2    Software Detailed Design.....	xi
4.3    Hardware Architecture.....	xiii
4.3.1    Security Hardware Architecture.....	xiv
4.3.2    Performance Hardware Architecture.....	xiv
4.4    Software Architecture.....	xiv
4.4.1    Security Software Architecture.....	xv
4.4.2    Performance Software Architecture.....	xv
4.5    Information Architecture.....	xvi
4.5.1    Records Management.....	xvi
4.6    Internal Communications Architecture.....	xvii
4.7    Security Architecture.....	xvii
4.8    Performance.....	xvii
4.9    System Architecture Diagram.....	xviii
<b>5. System Design.....</b>	<b>xix</b>
5.1    Business Requirements.....	xix
5.2    Database Design.....	xxiv
5.2.1    Data Objects and Resultant Data Structures.....	xxiv
5.2.2    File and Database Structures.....	xxv
5.3    Data Conversion.....	xxviii
5.4    User Machine-Readable Interface.....	xxix
5.4.1    Inputs.....	xxix
5.4.2    Outputs.....	xxx
5.5    User Interface Design.....	xxxv

## 6. Project Testing Plan

Error! Bookmark not defined.

# **1. Introduction**

---

*Instructions: Provide identifying information for the existing and/or proposed automated system or situation for which the System Design Document (SDD) applies (e.g., the full names and acronyms for the development project, the existing system or situation, and the proposed system or situation, as applicable), and expected evolution of the document. Also describe any security or privacy considerations associated with use of this document. –*

This application is a website designed for users to plan out their vacation itinerary. On top of that, this is going to be a platform for both travelers (Registered Users) and 3<sup>rd</sup> Party Business Managers. GoWander wants to create a platform that helps bring both users together so that services on each end are fulfilled. Registered Users want to plan out their vacation easily and make sure that it falls within their budget. 3<sup>rd</sup> Party Business Managers want to advertise their hotels, flights, and events. While the itinerary is the focal point of this technical project, the business solution of fulfilling both the 3<sup>rd</sup> Party Business Manager and Registered Users needs is the overall goal of this project. The technical goal is to enable Registered Users and 3<sup>rd</sup> Party Business Managers to create an itinerary for their plans quickly and intuitively.

When creating a new itinerary, Registered Users will be able to view their airplane arrival and departure dates, hotels, and events before adding them to their itinerary. The itinerary will be viewed in a calendar and will also include the time of said events in the itinerary as well. The same will occur with 3<sup>rd</sup> Party Business Managers. 3<sup>rd</sup> Party Business Manager's roles include setting the time and date for events that Registered Users will be able to add to their itinerary. The event's time and date could also be altered by the 3<sup>rd</sup> Party Business Manager. The 3<sup>rd</sup> Party Business Manager will be the user who will experience higher security risks and will want more privacy since they are businesses. Registered Users will also have sensitive information that shouldn't be shared. This includes email, phone number, etc. These are the security and privacy concerns for this website. This is the overall premise of the GoWander travel agency website that is being created.

## **1.1 Purpose of the SDD –**

*Instructions: Provide the purpose of the SDD. This document should be tailored to fit a particular project's needs.*

The SDD documents tracks the necessary information required to effectively define the architecture and system design in order to give the development group guidance on the architecture of the system to be developed. Design documents are incrementally and iteratively produced during the system development life cycle, based on the particular circumstances of the information technology (IT) project and the system development methodology used for developing the system. Its intended audience is the project manager, project group, and development group. Some portions of this document, such as the user interface (UI), may be shared with the client/user, and other stakeholders whose input/approval into the UI is needed.

This document serves as an overview of the system development process. It also is a reference point for everyone involved in the design and development process including stakeholders. All the information and details of GoWander's system can be found in the System Design document. This includes the charts, figures, interfaces, system testing, and data models.

## **2. General Overview and Design Guidelines/Approach**

This section describes the principles and strategies to be used as guidelines when designing and implementing the system.

### **2.1 General Overview –**

*Instructions: Briefly introduce the system context and the basic design approach or organization. Provide a brief overview of the system and software architectures and the design goals. Include the high-level context diagram(s) for the system and subsystems previously provided in the High-Level Technical Design Concept/Alternatives and/or Requirements Document, updated as necessary to reflect any changes that have been made based on more current information or understanding. If the high-level context diagram has been updated, identify the changes that were made and why.*

GoWander has taken the venture to create an all-encompassing itinerary planner for vacations. This system will cover most ends of a typical vacation such as flights, hotels, and events. All of the relevant information is stored in our MySQL database. Our server is hosted locally through XAMPP. This approach was chosen because it is simple and allows us to get the groundwork of our project done in a limited timeframe. Once GoWander transitions into the commercialization stage the hardware architecture will be revamped through Microsoft Azure Cloud Computing Services. The overall design goal is to have a simplified user interface that is user-friendly. GoWander wants users to be able to navigate through the website easily and also make the task of planning an itinerary/event efficient. Users should be able to do all of this quickly.

### **2.2 Assumptions/Constraints/Risks**

#### **2.2.1 Assumptions –**

*Instructions: Describe any assumptions or dependencies regarding the system, software and its use. These may concern such issues as: related software or hardware, operating systems, end-user characteristics, and possible and/or probable changes in functionality.*

With the design of the travel agency website, GoWander has found that several key assumptions will be made while implementing:

First off, our design assumes that Registered Users and 3<sup>rd</sup> Party Business Managers will continuously provide reliable and accurate information when it comes to the details available for itinerary events. This is a key assumption because it is part of the foundation of the system's functionality and real-world application.

Secondly, continuing off the previous assumption our system will rely on the fact that 3<sup>rd</sup> Party Business Managers will stay active in updating information regarding the different events and their availability within a reasonable time. Real-time data and management is a crucial part of providing users with a useful and current list of options when planning a trip.

Additionally, it is assumed that the MySQL database management system will be able to effectively handle the storage and retrieval of user or event data for the itinerary whenever needed. This assumption showcases the importance of our database infrastructure for the system to function properly.

Lastly, our system assumes that the hosting environment, even though it is hosted locally, will be capable of effectively handling the website's operations and functionality with continuous data transfer/update. Once GoWander transitions into the commercialization stage the hardware architecture will be revamped through Microsoft Azure Cloud Computing Services. In addition to this, GoWander is assuming that users will mainly access the website through modern web browsers on desktops ensuring primal usability and stability.

### **2.2.2 Constraints –**

*Instructions: Describe any global limitations or constraints that have a significant impact on the design of the system's hardware, software and/or communications, and describe the associated impact. Such constraints may be imposed by any of the following (the list is not exhaustive):*

- *Hardware or software environment*
- *End-user environment*
- *Availability or volatility of resources*
- *Standards compliance*
- *Interoperability requirements*
- *Interface/protocol requirements*
- *Licensing requirements*
- *Data repository and distribution requirements*
- *Security requirements (or other such regulations)*
- *Memory or other capacity limitations*
- *Performance requirements*
- *Network communications*
- *Verification and validation requirements (testing)*
- *Other means of addressing quality goals*
- *Other requirements described in the Requirements Document*

The system must stick to industry standards and any best practices associated with web development, including CSS, HTML, JSON, JavaScript, and PHP standards to ensure compatibility/accessibility with any modern browsers. Not complying with these standards could result in compatibility issues that would negatively affect all users.

The system data cache hosted locally on the MySQL database management system must be able to efficiently handle data storage and retrieval operations within the constraints of the hosting environment. Constraints with data distribution could impact synchronization strategies that ensure data consistency and availability throughout the website.

The system must account for memory and other capacity limits from the hosting environment to help optimize resource usage and prevent performance issues. Different techniques like caching could be used to mitigate any capacity constraints and enhance system functionality.

### **2.2.3 Risks –**

*Instructions: Describe any risks associated with the system design and proposed mitigation strategies.*

There can be many risks associated with our system design. Examples include performance, scalability, maintainability, reliability, usability, and security.

As I previously mentioned in the Introduction in Section 1, sensitive information that wouldn't want to be shared is held in our database. For example, a Registered User's contact information and vacation itinerary plans are held in the database. This information should not be shared, especially the 3<sup>rd</sup> party Business Manager information.

GoWander will need to find a way to keep this information confidential. Mitigation strategies that can be implemented will include user authentication, assigning user security roles, data encryption, database activity monitoring, use of firewalls, and adding a proxy server.

### **3. Design Considerations**

---

*Instructions: Describe issues which need to be addressed or resolved before attempting to devise a complete design solution.*

#### **3.1 Goals and Guidelines –**

*Instructions: Describe any goals, guidelines, principles, or priorities which dominate or embody the design of the system and its software. Examples of such goals might be: an emphasis on speed versus memory use; or working, looking, or “feeling” like an existing product. Guidelines include coding guidelines and conventions. For each such goal or guideline, describe the reason for its desirability unless it is implicitly obvious. Describe any design policies and/or tactics that do not have sweeping architectural implications (meaning they would not significantly affect the overall organization of the system and its high-level structures), but which nonetheless affect the details of the interface and/or implementation of various aspects of the system (e.g., choice of which specific product to use).*

The overall goal of this travel agency website is to have Registered Users plan out their next vacation and for 3<sup>rd</sup> Party Business Managers to upload and manage events for Registered Users. Essentially, the core of this website is the itinerary. Therefore, the website must have guidelines that pertain to the itinerary.

With that being said, readers can see that everything revolves around the itinerary. Dates and times on the calendar need to be visible. If a user wants to see detailed information on said event, they can click on it, and it will redirect them to the Blown-Up Events Detail Window. The user's itinerary can also be seen on the Blown-Up Events Detail Page so that further planning and comparisons can be made.

Another guideline that will be implemented for our travel agency website is having purchases redirected. In the Blown-Up Events Detail Page there will be a hyperlink that redirects you to said event where a ticket could be purchased. However, there will be a price on the Blown-Up Events Detail Page. However, it is nonfunctional – the websites are not real.

The GoWander website needs to be simplistic and to the point so that it is user-friendly. This can be achieved by the travel agency website's layout, style, user interface components, text, and design patterns. Not only that, it needs to be viewable on a mobile device as well. This is achieved by inserting a script that automatically fits the width of the user's device.

These guidelines are set in place so that the goals I previously mentioned can be achieved. This website's simplicity allows users to utilize an itinerary because that is the focal point of our travel agency website.

#### **3.2 Development Methods & Contingencies –**

*Instructions: Briefly describe the method or approach used for the system and software design (e.g., structured, object-oriented, prototyping, J2EE, UML, XML, etc.). If one or more formal/ published methods were adopted or adapted, then include a reference to a more detailed description of these methods. If several methods were seriously considered, then each such method should be mentioned, along with a brief explanation of why all or part of it was used or not used. Describe any contingencies that might arise in the design of the system and software that may change the development direction.*

*Possibilities include lack of interface agreements with outside agencies or unstable architectures at the time the SDD is prepared. Address any possible workarounds or alternative plans.*

There were multiple development methods for this group project – UML use case diagrams, user stories, and ERD. GoWander initially wanted to make a travel agency application. We were going to use Flutterflow to help us build this application. However, we ran into issues. For example, when we would get together and try to edit the same document on Flutterflow the edits wouldn't fully register.

Essentially, only one person could make edits at a time. On top of that, there was a paywall issue with Flutterflow as well. We had a limited amount of screens we could generate with the free version. Not only that, we also couldn't download the code for our travel agency app as well.

The same situation happened with an API that we found. It was called Aviationstack. The pricing plans stated that over 100 requests could be made with the free version, but that wasn't the case. As soon as a free account was created, I wasn't able to do anything and was prompted to include credit card information. With that being the case, we will only be utilizing a MySQL database. Not only that, Linode didn't work for us as well. Instead, we will be hosting our server locally by using XAMPP.

With these abrupt changes, we decided to make a travel agency website instead. We used Figma to generate the layout of our website. Afterwards, the script of the website was created to closely replicate our Figma wireframe.

### **3.3 Architectural Strategies –**

*Instructions: Describe any design decisions and/or strategies that affect the overall organization of the system and its higher-level structures. These strategies should provide insight into the key abstractions and mechanisms used in the system architecture. Describe the reasoning employed for each decision and/or strategy (possibly referring to previously stated design goals and principles) and how any design goals or priorities were balanced or traded-off.*

*Describe compliance with CMS Enterprise Architecture (EA) and standards. Specifically identify any deviations that were made from the CMS EA and standards, and provide rationale to support the deviation(s). When describing a design decision, discuss any other significant alternatives that were considered, and the reasons for rejecting them (as well as the reasons for accepting the alternative finally chosen). Sometimes it may be most effective to employ the “pattern format” for describing a strategy.*

*Examples of design decisions might concern (but are not limited to) things like the following:*

- *Use of a particular type of product (programming language, database, library, commercial off-the-shelf (COTS) product, etc.)*
- *Reuse of existing software components to implement various parts/features of the system*
- *Future plans for extending or enhancing the software*
- *User interface paradigms (or system input and output models)*
- *Hardware and/or software interface paradigms*

- *Error detection and recovery*
- *Memory management policies*
- *External databases and/or data storage management and persistence*
- *Distributed data or control over a network*
- *Generalized approaches to control*
- *Concurrency and synchronization*
- *Communication mechanisms*
- *Management of other resources*

Essentially, what GoWander initially wanted for this project couldn't be achieved with the timespan that we have. Instead, we made two major decisions that have changed the forefront of this project. Those two decisions were making a travel agency website instead of an application and hosting our server locally instead of utilizing the cloud. Everything we initially wanted could have been achieved, but we are restricted to completing this in a semester.

This in return, has resulted in us making our own MySQL database along with a locally hosted server with XAMPP. There will be no external databases/APIs. Everything will be made from the ground up by using various types of programming languages; CSS, HTML, JSON, JavaScript, and PHP. The goal is to have a fully functional travel agency website that is rudimentary and if time allows it we will expand.

### **3.4 Performance Engineering –**

*Instructions:*

*Using the Performance Requirements defined in the Requirements Document, provide a detailed explanation that describes how the Performance Requirements were incorporated into the system's design. Please refer to Sections 2.0 of the CMS Performance Test Plan and Results Template for guidance on defining Performance Requirements.*

*Start preparing Production Load Model(s) in preparation for Performance testing. Please refer to Sections 2.1.1 of the CMS Performance Test Plan and Results Template for guidance on Load Model construction.*

Performance requirements are a contributing factor to system design. Performance requirements are the defined scalability or responsiveness expectations of specific workloads that process on a system.

Our performance engineering objectives will include the elimination of late system deployment because of performance issues, the identification of bottlenecks, avoiding additional unnecessary hardware, and the reduction of unnecessary software maintenance. This all needs to work properly and efficiently right away because a faulty inefficient website will result in the loss of business. As of 4/25/2024, we will try our best to have everything properly working through its prototype stage. Once our logistics and operations plan is completed and we transition into the commercial stage of the GoWander travel agency website the performance engineering will be completely revamped so that anything that was causing the website to not be efficient or stable will be rectified so that performance quota is met.

## 4. System Architecture and Architecture Design

---

*Instructions: Describe the system architecture, how the application interacts with other applications. Not necessarily how the application itself works but, how the appropriate data is correctly passed between applications. Provide an overview of how the functionality and responsibilities of the system were partitioned and then assigned to subsystems or components. Don't go into too much detail about the individual components themselves in this section. A subsequent section of the SDD will provide the detailed component descriptions. The main purpose here is to gain a general understanding of how and why the system was decomposed, and how the individual parts work together to provide the desired functionality.*

*At the top-most level, describe the major responsibilities that the software must undertake and the various roles that the system (or portion of the system) must play. Describe how the system was broken down into its components/subsystems (identifying each top-level component/subsystem and the roles/responsibilities assigned to it). Describe how the higher-level components collaborate with each other in order to achieve the required results. Provide some sort of rationale for choosing this particular decomposition of the system (perhaps discussing other proposed decompositions and why they were rejected).*

*Make use of design patterns whenever possible, either in describing parts of the architecture (in pattern format), or for referring to elements of the architecture that employ them. Provide rationale for choosing a particular algorithm or programming idiom (or design pattern) to implement portions of the system's functionality.*

This section outlines the system and hardware architecture design of the system.

### 4.1 Logical View –

*Instructions: Insert any related logical views or provide a reference to where they are stored.*

The logical view for our website will revolve around whether a user is a 3<sup>rd</sup> Party Business Manager or a Registered User. For example, Registered User's accounts specifically pertain to setting an itinerary – adding events, hotels, and planes to an interactive calendar/itinerary. For this to occur the Registered User needs to log in and on the Registered User Homepage, all the following fields will be recommended.

3<sup>rd</sup> Party Business Manager accounts specifically pertain to setting up the events said Registered Users can add to their itinerary. The same situation occurs with the 3<sup>rd</sup> Party Business Manager. They will also log in and once they are on the 3<sup>rd</sup> Party Business Manager Settings page they can click on the plus button so that they are navigated to the Blown Up 3<sup>rd</sup> Party Business Manager Events Detail page. This is where they can add event details and schedule events for Registered Users to add to their itinerary. All in all, the data between both users need to communicate with one another. This is crucial to our website since it is a platform for both users.

## 4.2 Software Detailed Design –

Provide a detailed description for each system software service that addresses the following software service attributes. Much of the information that appears in this section should be contained in the headers/prologues and comment sections of the source code for each component, subsystem, module, and subroutine. If so, this section may largely consist of references to or excerpts of annotated diagrams and source code. Any referenced diagrams or source code excerpts should be provided at any design reviews.

- *Service Identifier - The unique identifier and/or name of the software service*
- *Classification - The kind of service (e.g., application, data service, etc.)*
- *Definition - The specific purpose and semantic meaning of the service*
- *Requirements - The specific functional or nonfunctional requirements that the service satisfies*
- *Internal Data Structures - The internal data structures for the service*
- *Constraints - Any relevant, assumptions, limitations, or constraints for the service (this should include constraints on timing, storage, or service state, and might include rules for interacting with the service (encompassing pre-conditions, post- conditions, invariants, other constraints on input or output values and local or global values, data formats and data access, synchronization, exceptions, etc.))*
- *Composition - A description of the use and meaning of the subservices that are a part of the service*
- *Users/Interactions - A description of the service's collaborations with other services (what other services use this entity? what other services does this entity use (including any side-effects this service might have on other parts of the system)? this includes the method of interaction, as well as the interaction itself. Object-oriented designs should include a description of any known or anticipated sub-classes, super-classes, and meta-classes)*
- *Processing - A description of precisely how the service goes about performing the duties necessary to fulfill its responsibilities (this should encompass a description of any algorithms used; changes or state; relevant time or space complexity; concurrency; methods of creation, initialization, and cleanup; and handling of exceptional conditions)*
- *Interfaces/Exports - The set of services (resources, data types, constants, subroutines, and exceptions) that the service provides (the precise definition or declaration of each such element should be present, along with comments or annotations describing the meanings of values, parameters, etc.; for each service element described, include or provide a reference in its discussion to a description of its important software service attributes (Component Identifier, Classification, Language, SLOC Estimate, Definition, Responsibilities, Requirements, Internal Data Structures, Constraints, Composition, Uses/Interactions, Resources, Processing, and Interfaces/Exports))*
- *Reporting Design and Integration - If built in, provide details on data traffic and volumes*

Service Identifier – Quite a bit of software was utilized for this project. The main software was mySQL and XAMPP.

Classification & Definition – GoWander is creating a travel agency website where Registered Users can make an itinerary. Not only that, 3<sup>rd</sup> Party Business Managers set up these events that Registered Users can add to their itineraries.

Requirements – Registered Users need to be able to plan out their vacation by setting up their itinerary. 3<sup>rd</sup> Party Business Managers need to make a profit so they sell their services. This travel agency website offers the solution to both of these users.

Internal Data Structures – GoWander will be using a MySQL database to store data. MySQL is an open-source relational database management system. We will also be using XAMMP to locally host our server. XAMMP is an open-source cross-platform server solution.

Constraints – Inaccurate or missing data from the database will be a constraint once the travel agency website is up and running.

Users/Interaction – Registered Users make an account and plan out their vacation by creating an itinerary. 3<sup>rd</sup> Party Business Manager makes accounts as well and creates events that Registered Users can add to their itinerary.

Processing & Interfaces/Exports – The Registered Users and 3<sup>rd</sup> Party Business Managers will log into their account. The Registered User will then navigate to the Registered User homepage where they will input the start and end date of their vacation. Afterwards, the Registered User Homepage will be populated with flights, hotels, and events that the Registered User can add to their itinerary. The same situation applies to 3<sup>rd</sup> Party Business Managers. The 3<sup>rd</sup> Party Business Managers are navigated to the 3<sup>rd</sup> Party Business Managers Setting page. They just need to click on the plus sign and they are taken to the Blown Up 3<sup>rd</sup> Party Business Manager Events Detail Page. This is where 3<sup>rd</sup> Party Business Managers create new events for Registered Users to add to their itineraries.

Reporting Design and Integration – The travel agency website doesn't need a reporting design or integration. This does not apply to this project.

### **4.3 Hardware Architecture**

*Instructions: Describe the overall system hardware and organization, indicating whether the processing system is distributed or centralized. Identify the type, number, and location of all hardware components including the presentation, application, and data servers and any peripheral devices (e.g., load balancers, SSL accelerator, switches, firewalls, with a brief description of each item and diagrams showing the connectivity between the components along with required firewalls, ports, and network bands utilized (e.g., management band). Include resource estimates for processor capacity, memory, on-line storage, and auxiliary storage. –*

As of 4/25/2024, our hardware architecture is rudimentary. In this prototype stage, we want to make sure that the travel agency website is up and running. Once our logistics and operations plan is completed and we transition into the commercial stage of the GoWander travel agency website our hardware architecture will be completely revamped so that it can handle the demand of being commercialized. When this occurs GoWander will transition to Microsoft Azure Cloud Computing Services. This will impact us greatly because they have a pay-as-you-go

solution which is a perfect use case for GoWander. Instead of having to scale up or down we automatically pay for what is utilized.

### **4.3.1 Security Hardware Architecture –**

*Instructions: Describe the hardware components and configuration supporting the security and privacy of the system. Specify the architecture for (1) authentication to validate user identity before allowing access to the system, including the use of IACS/EUA or other type of Identity Vetting & Authentication system; (2) authorization of users to perform functional activity once logged into the system, (3) encryption protocol to support the business risks and the nature of information, and (4) logging and auditing design, if required. The design should be based on the designated system security level and provide adequate protection against threats and vulnerabilities.*

The security hardware architecture that our website will utilize is encryption. Passwords are encrypted and so is the database. Not only that but the password on the login page is hidden as well. Once GoWander becomes commercialized we plan on incorporating two-factor authentication and HTTPS encryption as well. GoWander does not want its users to have their information breached. In the commercial stage, we will be using Microsoft Azure Cloud Computing Services because of its industry-leading security.

### **4.3.2 Performance Hardware Architecture –**

*Instructions: Describe the hardware components and configuration supporting the performance and reliability of the system. Identify single points of failure and, if relevant, describe high availability design (e.g., clustering).*

As of 4/25/2024, our hardware architecture is rudimentary because it is in the prototype stage. Once GoWander transitions into the commercialization stage the hardware architecture will be revamped through Microsoft Azure Cloud Computing Services. Since our hardware architecture will be cloud-based and will be utilizing a pay-as-you-go solution the performance will be streamlined for GoWander. This is because of the automatic scaling of the cloud infrastructure.

## **4.4 Software Architecture**

*Instructions: Describe all software that is needed to support the system, the hardware component for which each software component is targeted, and specify the physical location of all software systems. List such things as logical components (e.g., JSP in presentation layer, JNDI in application layer, EJB and JDBC in data layer), database platforms, computer languages, compilers, utilities, operating systems, communications software, programming computer-aided software engineering tools, commercial off-the-shelf (COTS) software, open source frameworks, etc., with a brief description of the function of each item and any identifying information such as manufacturer, version number, number and types of licenses needed, etc., as appropriate. Identify all Computer Software Configuration Items (CSCIs), Computer Software Components (CSCCs) and Application Programming Interfaces (APIs) to include name, type, purpose and function for each; the interfaces, messaging, and protocols for those elements; and rationale for the software architectural design. Include software modules that are functions, subroutines, or classes. Use functional hierarchy diagrams, structured organization diagrams (e.g., structure charts), or object-oriented diagrams that show the*

*various segmentation levels down to the lowest level. All features on the diagrams should have reference numbers and names.*

*Include a narrative that expands on and enhances the understanding of the functional breakdown. If necessary, describe how a component was further divided into subcomponents, and the relationships and interactions between the subcomponents. Proceed into as many levels/subsections of discussion as needed in order to provide a high-level understanding of the entire system or subsystem, leaving the details for inclusion in a later section of the SDD. Include data flow diagrams that conform to appropriate standards (e.g., Yourdon-Demarco conventions) and provide the physical process and data flow related to the logical process and data flow decomposed to the primitive process level (describing how each input is processed/transformed into the resulting output). If there are parts of the system that already existed before this development effort began, then only describe the relationships and interactions between the old parts and the new parts. Pre-existing parts that are modified or enhanced need to be described only to the extent that is necessary to provide a sufficient understanding of the nature of the changes that are being made. –*

Everything was done on our laptops and the operating system that was mostly used is Windows OS, but some teammates are using Mac OS. To communicate with each other GroupMe, Discord, and our Kent State emails were utilized. GoWander also utilized Figma and Astah to help us in the beginning stages of the project and to help us create diagrams and wireframes. As we moved to the web development phase of our project we utilized XAMPP and MySQL for our server and database. The computer languages that were utilized are CSS, HTML, JSON, JavaScript, and PHP. The only other software that was used was Microsoft Word and Google Documents. Both were used so that this system design document could be worked on.

#### **4.4.1 Security Software Architecture –**

*Instructions: Describe the software components and configuration supporting the security and privacy of the system. Specify the architecture for (1) authentication to validate user identity before allowing access to the system, including the use of IACS/EUA or other type of Identity Vetting & Authentication system;(2) authorization of users to perform functional activity once logged into the system, (3) encryption protocol to support the business risks and the nature of information, and (4) logging and auditing design, if required. The design should be based on the designated system security level and provide adequate protection against threats and vulnerabilities.*

Since GoWander is a travel agency website there is no need for any security software. Instead, we will be focusing on security hardware architecture.

#### **4.4.2 Performance Software Architecture –**

*Instructions: Describe the software components and configuration supporting the performance and reliability of the system. Identify single points of failure and, if relevant, describe high availability design (e.g., clustering).*

The travel agency website doesn't need performance software architecture. This does not apply to this project.

## 4.5 Information Architecture

*Instructions: Describe the information that will be stored in the system (e.g. beneficiary information, claim data, etc.) Identify if any of the information is personally identifiable information (PII), individually identifiable information (IIF), or personal health information (PHI). –*

There are multiple tables in our MySQL database. Those tables are the business, events, flights, hotels, and users. The majority of the tables have the same data. For example, the events table has the event ID, event name, event length, event time, event description, event price, event street address, event state, event city, event ages, event dates, event highlights, business id, image paths, event end date, event time 2, event time 3, event time 4, event highlights 2, event highlights 3, event highlights 4. This makes identification easy.

### 4.5.1 Records Management

Federal regulations issued by the National Archives and Records Administration (NARA) are outlined in 36 Code of Federal Regulations (CFR) - Subchapter B - Records Management. Business owners must contact the Office of Strategic Operations and Regulatory Affairs (OSORA) to initiate the record management process.

#### 4.5.1.1 Data –

*Identify all data (as well as the format of the data — paper, manual input, electronic data) supplied to the system as well as who/what is supplying the data.*

This document along with the diagrams and wireframe are crucial pieces of data that explain how our system works. The other crucial pieces of data are the script of the travel agency website and PHP files for our database. These are crucial because they allow the website and database to communicate with each other and ensure that everything works properly.

#### 4.5.1.2 Manual/Electronic Inputs –

*Provide instructions on what happens to the manual/electronic inputs after they are entered into the master file/database and are verified.*

After Registered Users are logged into their account they can create their itinerary on the Registered User homepage where they will input the start and end date of their vacation. Afterwards, the Registered User Homepage will be populated with flights, hotels, and events that the Registered User can add to their itinerary. The same situation applies to 3<sup>rd</sup> Party Business Managers. The 3<sup>rd</sup> Party Business Managers are navigated to the 3<sup>rd</sup> Party Business Managers Setting page. They just need to click on the plus sign and they are taken to the Blown Up 3<sup>rd</sup> Party Business Manager Events Detail Page. This is where 3<sup>rd</sup> Party Business Managers create new events for Registered Users to add to their itineraries.

#### 4.5.1.3 Master Files –

*Provide a detailed description of the data maintained in the system/database.*

All of our script files are held in the XAMPP htdocs folder. Once our server is up and running locally the website and database communicate with one another.

## **4.6 Internal Communications Architecture –**

*Instructions: Provide a detailed description of the system's communications network, denoting the communications architecture(s) being implemented (e.g., Token Ring, Ethernet, etc.) and how system components are linked. Describe any Local or Wide Area Networks and buses. Include descriptions of necessary equipment (e.g., hubs, routers, cabling, transmitters, firewalls, ports, etc.). Provide a diagram depicting the communications flow between the system and subsystem components. Include resource estimates for communications network capacity (LAN and WAN) needed to install and execute each application on each platform.*

The only form of internal communication would be our locally hosted server. The server makes it possible for our database to communicate with our travel agency website.

## **4.7 Security Architecture –**

*Instructions: Insert any related security architecture documents, including integrity controls, or provide a reference to where they are stored.*

- *Authentication*
- *Authorization*
- *Logging and Auditing*
- *Encryption*
- *Network ports usage*
- *Intrusion Detection and Prevention (especially if hosted in a non-CMS data center)*

The travel agency website doesn't need security architecture documentation. This does not apply to this project.

## **4.8 Performance –**

*Instructions: Insert any performance documents or provide a reference to where they are stored.*

- *Capacity and volume requirements/estimates*
- *Performance expectations*
- *Availability requirements*
- *Performance design to meet capacity requirements*
- *Reliability design to meet availability requirements*
- *Backup, recovery, and archive design*

*Identify single points of failure and, if relevant, describe high availability design (e.g., clustering).*

The travel agency website doesn't need performance documentation. This does not apply to this project.

## 4.9 System Architecture Diagram –

*Instructions: Using the hardware, software, communications, and information designs described above, depict the overall, integrated structure of the system in terms of presentation, application, and data regions.*

*Diagram(s) should show the architecture of the “to-be-production-system,” i.e., the architectural design must map to the targeted production environment.*

*Diagram(s) must reflect the accuracy and clarity of the system architecture.*

*Diagram(s) must reflect the complete system’s context, i.e., more detailed software components, internal/external interfaces, and their underlying infrastructure, etc., not just pertain to the design under review.*

The travel agency website doesn't need a system architecture diagram. This does not apply to this project.

## 5. System Design

---

### 5.1 Business Requirements –

*Instructions: Insert any related project business requirements or provide a reference to where they are stored.*

There was extensive research into this project. The business requirements included location, roles, functions, user stories, requirements gathering, and the use of a survey. The prototyping location is Kailua-Kona, Hawaii. The roles, user stories, and functions are:

#### ❖ Registered User

##### • *Functions*

A user account that needs services to help them plan out their vacation and find events and airplane tickets. -

Use the website to schedule events for the itinerary. -

Use the website to add airplane arrival and departure dates. -

Add, remove, and cancel events for the itinerary. -

Log in and out of the website. -

Delete account. -

##### • *User Stories*

As a Registered User, I want to be able to add my airplane arrival and departure dates to the website's calendar. -

As a Registered User, I want to be able to add the dates I am staying at a specific hotel on the website's calendar. -

As a Registered User, I want to be able to make edits (add, delete, change time) to my vacation itinerary on the website's calendar. -

As a Registered User, I want to be able to delete my account if I don't want to use the website anymore. -

As a Registered user, I want to be able to filter through different kinds of activities. -

As a Registered User, I want to be able to save my itineraries so I can view them at a later date. -

#### ❖ Administrator (Commercialized Phase)

##### • *Functions*

User account that has administrative features for the website. - Have access to user and third-party business information. - Register and delete user and third-party business accounts. - Troubleshoot issues for user and third-party business accounts. -

##### • *User Stories*

As an Administrator User, I want to be able to register a new user's account into the system. -

As an Administrator User, I want to be able to delete a Registered User account if it is requested. -

❖ **System Clock (Commercialized Phase)**

● ***Functions***

Internal clock to help users plan their itinerary so that events don't overlap and remind them of upcoming events that they are signed up for. -

Notify users of all upcoming events. -

Notify users on stock for event and plane tickets. -

● ***User Stories***

As a System Clock, I want Registered Users and accounts to receive notifications on upcoming events in their vacation itinerary via email, phone number, or both. -

As a system clock, I want to be able to notify Registered Users if events held by 3<sup>rd</sup> party events are cancelled or tickets are sold out. -

As a system clock, I want to be able to notify Registered Users if airplane tickets they are interested in are almost sold out. -

❖ **3<sup>rd</sup> Party Business Managers**

● ***Functions***

Provide services for Registered Users so that they have a fun eventful vacation. -

● ***User Stories***

As a 3rd Party Business Manager, I want to be able to post my events/flights for users to see and implement into their itineraries. -

As a 3rd Party Business Manager, I want to be able to edit times, and dates, and be able to show users that events that I already posted are sold out. - **RA**

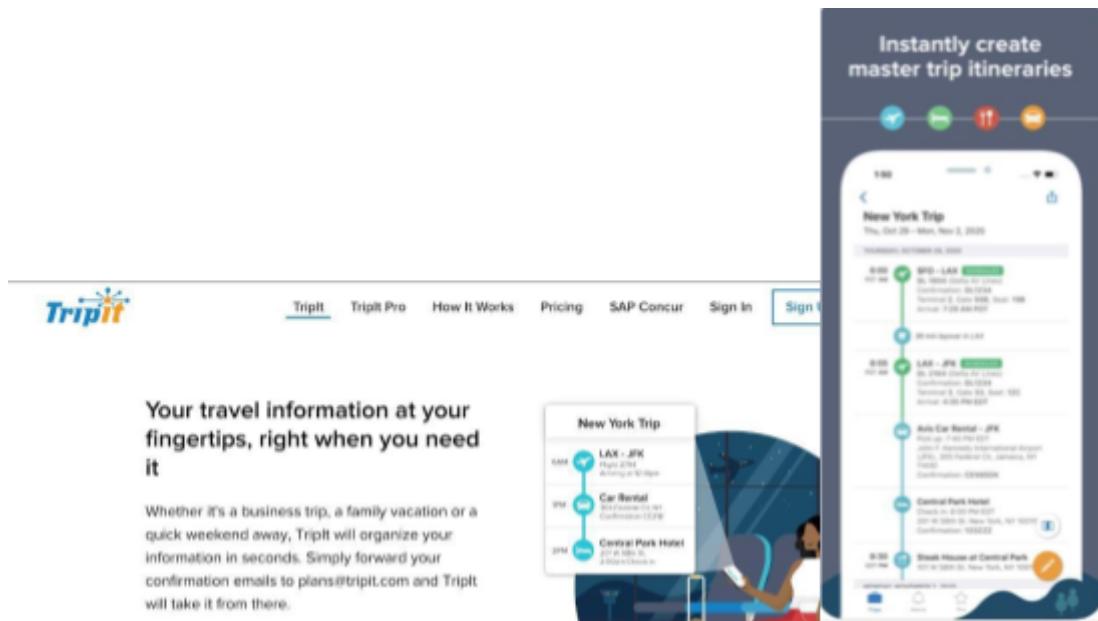
As a 3<sup>rd</sup> Party Business Manager, I want to be able to advertise specialized advertising on the website to attract Registered Users to my events (happy hour, discount, etc.). -

The requirements gathering for this project included the following:

❖ **Tripit –**

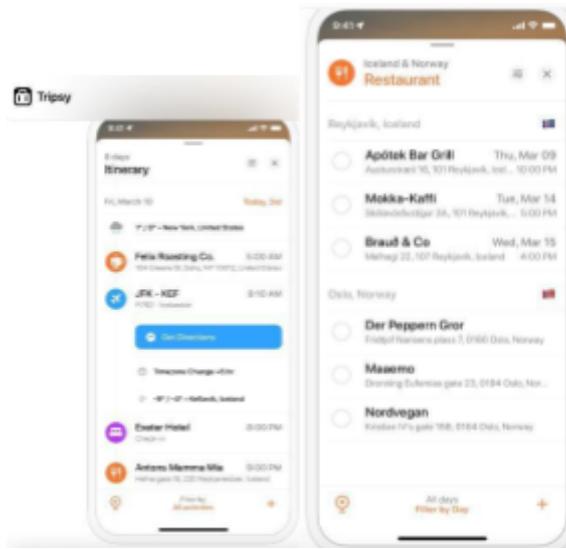
- Good daily list interface example.
- Does Not include a calendar or the ability to add local events.
- The length of the line between different activities based on the time between them is unique.
- The dashboard at the bottom focuses on Current trips compiled, alerts, and favorites. Each one has a tab.
- Activities stand out through colors that represent them.

- Includes a feature so it can be shared with others (top right).



#### ❖ Tripzy –

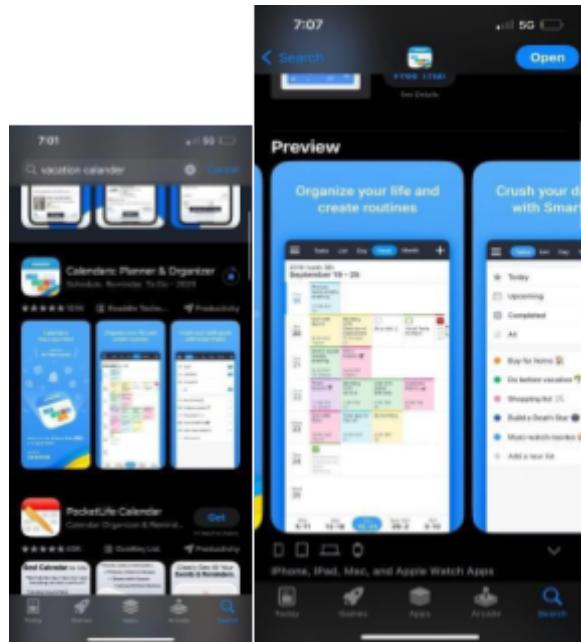
- Good examples of filtering activities in this case they are using restaurants.
- Also, a good user interface example of what a suggestions/ local event page could look like.



#### ❖ Planer & Organizer –

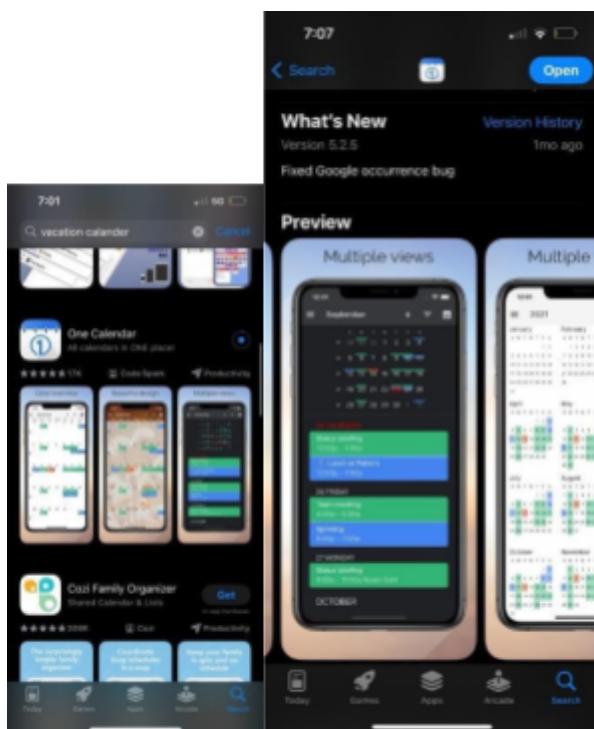
- Great calendar interface example that is color coordinated.
- A good example of the information we would want to be displayed (name, time, duration) is in the calendar blocks.

- Allows for events to be checked off (something we want to include here? Now we talked about it for the list).



### ❖ One Calendar –

- Best visual example of a calendar and a list being presented at the same time (very representative of what we discussed).
- The activity list is shown for whatever day is selected.



**❖ Survey –**

- <https://forms.gle/yCZtdTWjBDA7TDuH9>
- <https://docs.google.com/spreadsheets/d/1l2jNUbZ8nfSdK8faiKmW3Vp1f9qpS2a7huRt46RI1Gs/edit?resourcekey#gid=813589140>

**❖ FlutterFlow –**

- <https://flutterflow.io/>

**Pros**

- Builds responsive UI and logic visuals.
- Generates code script that can be edited.
- Allows Firebase integration.
- FireBase cloud functions with FlutterFlow.
- Also allows connection to supabase (another database).
- Allows connection with Google Maps – Geolocation.
- Allows 2 API endpoints.

**Cons**

- Does not have real-time collaboration unless the top premium subscription is purchased.
- The free version only offers 5 pages per user (would have to consider dividing up pages between free accounts, while trying to keep formatting the same).
- A Can only download code with the pay version.

**❖ Figma –****Pros**

- Allows collaboration.
- A Makes detailed version histories to track changes and revert to previous versions.
- A Good tool for interactive prototyping.
- Design libraries promote consistency allowing for a unified design language.

**Cons**

- Some users report issues with large files

**❖ Microsoft****Azure –****Pros**

- Potential to use as a database platform for information.
- Easily accessible for all members as it is free for students.

❖ **Quickbase** –

**Pros**

- Low code platform.
- Scalability.
- Capable of seeing multiple tables.
- AI features for easier building.

**Cons**

- Limited trial (30 days).
- Doesn't seem to put it all together more like a deconstructed series of sheets.

❖ **Flutter** –

- <https://flutter.dev/learn>
- Uses DART.
- Compatible with FireBase.

❖ **FireBase** –

- <https://firebase.google.com/>

❖ **Nandbox** –

- <https://nandbox.com/>

❖ **Travelpayouts Data API** –

- <https://support.travelpayouts.com/hc/en-us/categories/200358578-API-and-data>

## 5.2 Database Design

*Instructions: Describe the design of all database management system (DBMS) files and non-DBMS files associated with the system. Provide a comprehensive data dictionary showing data element name, type, length, source, validation rules, maintenance (create, read, update, delete (CRUD) capability), data stores, outputs, aliases, and description. The Data Design information can be included as an appendix or recorded in a separate Database Design Document (DDD), as appropriate, which would be referenced here. A template for a DDD is available from the CMS Integrated IT Investment & System Life Cycle Framework Web site located at <https://www.cms.gov/Research-Statistics-Data-and-Systems/CMS-Information-Technology/XLC/Downloads/DatabaseDesignDocument.docx>.*

### 5.2.1 Data Objects and Resultant Data Structures –

*Instructions: For each functional data object, specify the data structure(s) which will be used to store and process the data. Describe any data structures that are a major part of the system, including major data structures that are passed between components.*

*List all functions and function parameters. For functions, give function input and output names in the description. Refer as appropriate to the decomposition diagrams.*

The data objects for the travel agency website are denoted by the LDM/ERD. These data objects are crucial because they are utilized to create inputs and outputs which make our travel agency website function properly. For example, the data objects within the 3<sup>rd</sup> Party Business Manager are:

- BusinessID : int(11)
- BusinessName : varchar(100)
- BusinessPhone : varchar(100)
- BusinessEmail : varchar(100)
- BusinessPassword : varchar(100)
- BusinessDescription : varchar(100)
- BusinessStreetAddress : varchar(50)
- BusinessState : varchar(3)
- BusinessCity : varchar(50)
- BusinessWebsite : varchar(200)
- BusinessUsername : varchar(100)

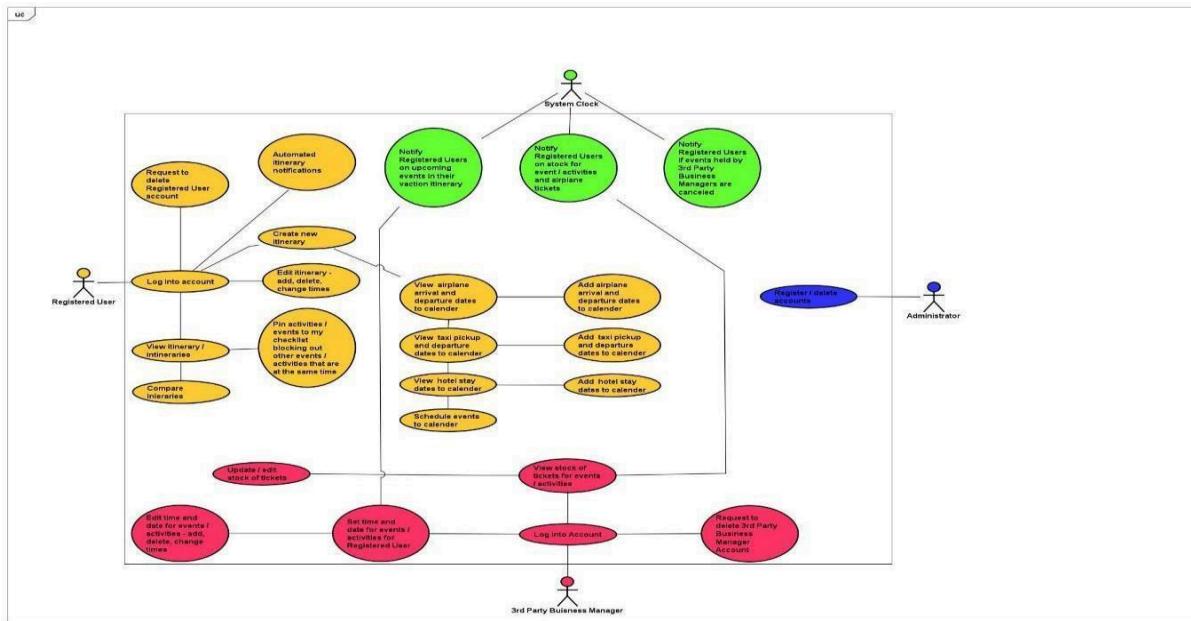
When these data objects are manipulated with the correct script GoWander can have inputs and outputs populate the travel agency website with 3<sup>rd</sup> Party Business Manager data. The data types of these attributes are mostly varchar and integers. For example, Business ID is an integer and Business Description is a varchar. All the data types and attributes can be seen in the LDM/ERD in Section 5.2.2.

### **5.2.2 File and Database Structures –**

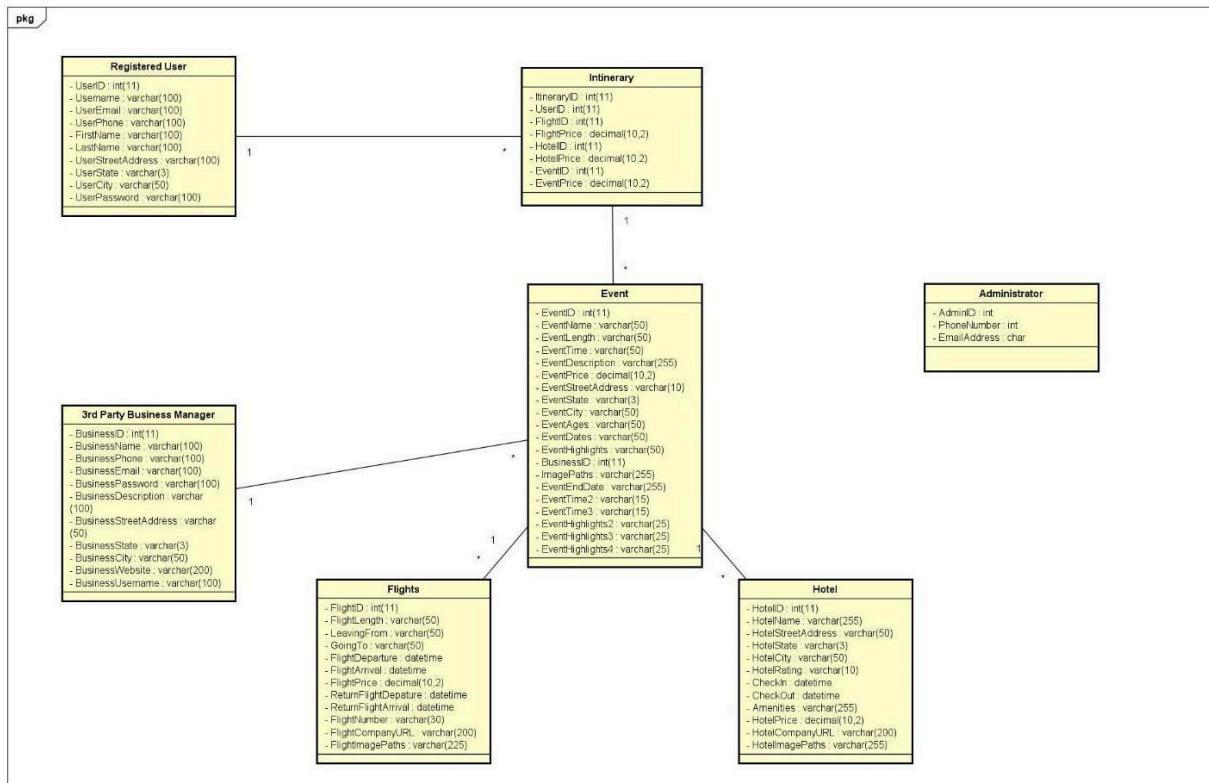
*Instructions: Using the Logical Data Model (LDM), create a physical data model that describes data storage and manipulation in the systems architectural setting. Describe file structures and their locations. Explain how data may be structured in the selected DBMS, if applicable. Refer to a separate DDD, as appropriate. For networks, detail the specific distribution of data. Note any changes to the LDM that occur because of software or hardware requirements.*

The Logical Data Model (LDM) is going to be the same as our ERD. This diagram contains multiple sections for the storage of data in our database. The sections that are in this diagram are – Registered User, Itinerary, 3<sup>rd</sup> Party Business Manager, Flights, Hotel, and Events. These sections can then be broken down into even more important pieces of data. For example, the Event section contains Event ID, Event Name, Event Length, Event Time, Event Description, Event Price, Event Street Address, Event State, Event City, Event Ages, Event Dates, Event Highlights, Business ID, Image Paths, Event End Date, Event Time 2, Event Time 3, Event Highlights 2, Event Highlights 3, and Event Highlights 4. The Event section also has relationships with the Hotel, Flights, and 3<sup>rd</sup> Party Business Manager sections. Essentially, everything is linked and correlates with the Itinerary section which is the focal point of our travel agency website.

## Class Diagram



## LDM / ERD



### 5.2.2.1 Database Management System Files –

*Instructions: Provide the detailed design of the DBMS files. Generally, this information should be documented in a separate DDD that should be referenced within this section. A template for a DDD is available from the CMS Integrated IT Investment & System Life Cycle Framework Web site located at <https://www.cms.gov/Research-Statistics-Data-and-Systems/CMS-Information-Technology/XLC/Downloads/DatabaseDesignDocument.docx>.*

The database management system files are in the XAMPP htdocs folder. Once you navigate into the htdocs folder there are multiple folders. Those folders are CalanderV1, FlightImages, HotellImages, Project-Itinerary-Plugin, and Uploads. The other files in the htdocs folder can be seen in the image down below.

Name	Date modified	Type	Size
CalanderV1	4/29/2024 6:10 PM	File folder	
FlightImages	4/29/2024 7:10 PM	File folder	
HotellImages	4/29/2024 7:10 PM	File folder	
Project-Itinerary-Plugin	4/29/2024 6:10 PM	File folder	
uploads	4/30/2024 2:29 PM	File folder	
.DS_Store	4/29/2024 5:49 PM	DS_STORE File	7 KB
BlownUpCompanyEvent.html	4/29/2024 5:49 PM	Chrome HTML Do...	7 KB
BlownUpCompanyEventCard.html	4/29/2024 5:49 PM	Chrome HTML Do...	4 KB
BlownUpRegisteredUserEventCard.html	4/29/2024 5:49 PM	Chrome HTML Do...	7 KB
business_manage_event.php	4/29/2024 5:50 PM	PHP Source File	15 KB
business_register_events.php	4/29/2024 5:49 PM	PHP Source File	17 KB
# create_events_style.css	4/29/2024 5:49 PM	CSS Source File	3 KB
# delete_event.php	4/29/2024 5:50 PM	PHP Source File	2 KB
# errors.php	4/29/2024 5:49 PM	PHP Source File	1 KB
# events.php	4/29/2024 5:49 PM	PHP Source File	0 KB
# Example.css	4/29/2024 5:49 PM	CSS Source File	1 KB
# fetch_event_details.php	4/29/2024 5:49 PM	PHP Source File	3 KB
# fetch_events.php	4/29/2024 5:49 PM	PHP Source File	4 KB
# flights.php	4/29/2024 5:49 PM	PHP Source File	3 KB
# goWanderAPI.json	4/29/2024 5:49 PM	JSON Source File	50 KB
# hotels.php	4/29/2024 5:49 PM	PHP Source File	4 KB
# login.php	4/29/2024 5:49 PM	PHP Source File	4 KB
# login_business.php	4/29/2024 5:49 PM	PHP Source File	4 KB
# loginstyle.css	4/29/2024 10:02 PM	CSS Source File	2 KB
# logout.php	4/29/2024 5:49 PM	PHP Source File	1 KB
# logout_business.php	4/29/2024 5:49 PM	PHP Source File	1 KB
# MainItineraryPage.php	4/30/2024 6:07 PM	PHP Source File	62 KB

# MainPage.css	4/29/2024 10:23 PM	CSS Source File	11 KB
process_business_profile.php	4/29/2024 5:49 PM	PHP Source File	3 KB
register.php	4/29/2024 5:49 PM	PHP Source File	4 KB
register_business.php	4/29/2024 10:06 PM	PHP Source File	8 KB
save_changes.php	4/29/2024 5:49 PM	PHP Source File	3 KB
save_changes_business.php	4/29/2024 5:49 PM	PHP Source File	3 KB
server.php	4/29/2024 5:49 PM	PHP Source File	9 KB
settings_business.php	4/30/2024 12:41 PM	PHP Source File	10 KB
SettingsUser.php	4/29/2024 5:49 PM	PHP Source File	9 KB
update_profile.php	4/30/2024 12:41 PM	PHP Source File	3 KB
users.php	4/29/2024 5:49 PM	PHP Source File	2 KB

### 5.2.2.2 Non-Database Management System Files –

*Instructions: Provide the detailed description of all non-DBMS files and include a narrative description of the usage of each file that identifies if the file is used for input, output, or both, and if the file is a temporary file. Also provide an indication of which modules read and write the file and include file structures (refer to the data dictionary). As appropriate, the file structure information should include the following:*

- Record structures, record keys or indexes, and data elements referenced within the records
- Record length (fixed or maximum variable length) and blocking factors
- Access method (e.g., index sequential, virtual sequential, random access, etc.)
- Estimate of the file size or volume of data within the file, including overhead resulting from file access methods
- Definition of the update frequency of the file (If the file is part of an online transaction-based system, provide the estimated number of transactions per unit of time, and the statistical mean, mode, and distribution of those transactions.)
- Backup and recovery specifications

There is a handful of system files that are not intrinsically correlated to the database. That would include this document, the screenshots that can be found in this document, the system design backups, and the Astah documents.

Other than that, some files are being utilized for GoWander and those are the images that are being used in the travel agency website. They are the background images, flight images, and event images. They all have their designated folder. Those folders are the FlightImages folder, the HotellImages folder, and the Uploads folder. The uploads folder is the folder that has all of the background images and event images while the FlightImages folder has all the images for flights. The HotellImages folder has all the hotel images for the website.

### 5.3 Data Conversion –

*Instructions: Insert any documents describing any necessary data conversions or provide a reference to where they are stored.*

The travel agency website doesn't utilize any data conversions. This does not apply to this project because we did not need to pull any data from online sources.

## **5.4 User Machine-Readable Interface –**

*Instructions: Provide a description of each user class or role associated with the system. A user class is distinguished by the ways in which users interact with the proposed system or situation. Factors that distinguish a user class include common responsibilities, skill levels, work activities, and modes of interaction with the system. In this context, a user is anyone who interacts with the proposed system, including operational users, data entry personnel, system operators, operational support personnel, system maintainers, and trainers. For each user class, provide estimates of the total number of users anticipated, a maximum number of concurrent users, and the number of external users.*

Registered User - This user class will be the people using the system primarily. They will be using the website to schedule events for their itinerary. This system will help them plan out their vacation by finding flights, events, and hotels in the area where they will be traveling. They will be able to add all of this to their vacation itinerary. GoWander anticipates that we will have about 50 users when the system is up and running, The maximum number of users that can use the system is 250, and 0 external users.

3<sup>rd</sup> Party Business Manager – Will provide services to Registered Users that involve advertisements of flights and events for Registered Users to add to their itinerary. GoWander anticipates that we will have 3 users using the system, The maximum number of concurrent users will be 10, and the number of external users will be 0.

Administrator - Will be able to assist Registered Users if they request tickets from a different website. They will be able to delete Registered Users' accounts if requested along with registering new user accounts into the system. GoWander will have 1 administrator using the system.

### **5.4.1 Inputs –**

*Instructions: Provide a description of the input media used by the user/operator for providing information to the system. Show a mapping to the high-level data flows (e.g., data entry screens, optical character readers, bar scanners, etc.). If appropriate, the input record types, file structures, and database structures provided in the section for Data Design, may be referenced. Include data element definitions, or refer to the data dictionary. Provide the layout of all input data screens or graphical user interfaces (GUIs) (e.g., windows). Define all data elements associated with each screen or GUI, or reference the data dictionary. Provide edit criteria for the data elements, including specific values, range of values, mandatory/optional, alphanumeric values, and length. Also address data entry controls to prevent edit bypassing. Discuss the miscellaneous messages associated with user/operator inputs, including (but not limited to) the following:*

- Copies of form(s) if the input data are keyed or scanned for data entry from printed forms
- Description of any access restrictions or security considerations

- *Each transaction name, code, and definition, if the system is a transaction-based processing system*
- *Incorporation of the Privacy Act statement into the screen flow, if the system is covered by the Privacy Act*
- *Description of accessibility provisions to comply with Section 508 of the Rehabilitation Act*

The inputs for the travel agency website would be from Registered Users and 3<sup>rd</sup> Party Business Managers. These users are the ones who initiate the website change because they are the ones making requests for the data to be manipulated and then populate the screen in front of them. This can be seen when either the Registered User or 3<sup>rd</sup> Party Business Manager logs into the website. Another example would be A Registered User looking for hotels or flights on August 20, 2024.

#### **5.4.2 Outputs –**

*Instructions: Describe the system output design relative to the user/operator. Show a mapping to the high-level data flows. System outputs include reports, data display screens and GUIs, query results, etc. The output files described in the section for Data Design may be referenced. The following should be provided, if appropriate:*

- *Identification of codes and names for reports and data display screens*
- *Description of report and screen contents (provide a graphical representation of each layout and define all data elements associated with the layout or reference the data dictionary)*
- *Description of the purpose of the output, including identification of the primary users*
- *Report distribution requirements, if any (include frequency for periodic reports)*
- *Description of any access restrictions or security considerations*

The outputs from the database are anything that can be seen in the LDM/ERD. Essentially, anything that needs to be populated within the travel agency website will be an output. This occurs for Registered Users and 3<sup>rd</sup> Party Business Managers. The Registered User has outputs in the:

- Registered User Settings page
- Registered User Home page,
- Blown Up Registered User Events Detail page.

The outputs in the Registered User Settings page are:

- First and last name
- Email
- Address
- Contact number
- City
- State
- Password.

The outputs in the Registered User Home Page are:

- Flights
- Hotels

- Events scorecard

The outputs in the Blown Up Registered User Events Detail page are:

- Event name
- Description
- Ages
- Start time
- Language
- Highlights
- Images.

The outputs for the 3<sup>rd</sup> Party Business Manager are:

- 3<sup>rd</sup> Party Business Manager Settings
- Blown Up 3<sup>rd</sup> Party Business Manager Events Detail Page

The outputs in the 3<sup>rd</sup> Party Business Manager Settings are:

- Company name
- Events within the events section

The outputs in the Blown Up 3<sup>rd</sup> Party Business Manager Events Detail Page are:

- Event Image
- Event name
- Business Name
- Description
- Times
- Dates
- Location
- Event Length
- Address
- Ages
- Highlights
- Phone
- Website
- Price

As of 4/12/24, the backend is still being finalized. Once the backend is completed the script that triggers the output to be brought up to the front end will be included in this section. Down below the script for our flights can be seen. It locates the database and grabs the information located in the flight table.



The screenshot shows a code editor with two tabs: 'flight.php' and 'MainitineraryPage.php'. The 'flight.php' tab is active, displaying PHP code for managing flights. The code includes database connection logic, a query to fetch flights from the database or a JSON file, and a script to insert new flight records. It also includes a section to fetch flights based on leavingFrom and goingTo parameters. The 'MainitineraryPage.php' tab is visible in the background.

```
flight.php
1 <?php
2 // Connect to MySQL
3 $servername = "localhost";
4 $username = "root"; // Your MySQL username
5 $password = ""; // Your MySQL password
6 $dbname = "gowander"; // Your database name
7
8 // Create connection
9 $conn = new mysqli($servername, $username, $password, $dbname);
10
11 // Check connection
12 if ($conn->connect_error) {
13     die("Connection failed: " . $conn->connect_error);
14 }
15
16 // Query to fetch flights
17 $sql = "SELECT * FROM flights";
18 $result = $conn->query($sql);
19
20 // If no data in the database, fetch from JSON file
21 if ($result->num_rows == 0) {
22     // Fetch flight data from JSON file
23     $json_data = file_get_contents('gowanderAPI.json');
24     $data = json_decode($json_data, true);
25
26     $flights = $data['flights'];
27
28     // Insert flights into the database
29     foreach ($flights as $flight) {
30         $flightID = $flight['FlightID'];
31         $Flightlength = $flight['Flightlength'];
32         $Leavingfrom = $flight['LeavingFrom'];
33         $Goingto = $flight['GoingTo'];
34         $Flightdeparture = $flight['FlightDeparture'];
35         $Flightarrival = $flight['FlightArrival'];
36         $Flightprice = $flight['FlightPrice'];
37         $Returnflightdeparture = $flight['ReturnFlightDeparture'];
38         $Returnflightarrival = $flight['ReturnFlightArrival'];
39         $Flightnumber = $flight['FlightNumber'];
40         $Flightcompanyurl = $flight['FlightCompanyURL'];
41         $Flightimagepaths = $flight['FlightImagePaths'];
42
43         // Insert data into flights table
44         $sql_insert = "INSERT INTO flights (FlightID, Flightlength, LeavingFrom, GoingTo, FlightDeparture, FlightArrival, FlightPrice, ReturnFlightDeparture, ReturnFlightArrival, FlightNumber, FlightCompanyURL, FlightImagePaths) VALUES ('{$flightID}', '{$Flightlength}', '{$Leavingfrom}', '{$Goingto}', '{$Flightdeparture}', '{$Flightarrival}', '{$Flightprice}', '{$Returnflightdeparture}', '{$Returnflightarrival}', '{$Flightnumber}', '{$Flightcompanyurl}', '{$Flightimagepaths}')";
45         if ($conn->query($sql_insert) === TRUE) {
46             echo "New record created successfully";
47         } else {
48             echo "Error: " . $sql_insert . "<br>" . $conn->error;
49         }
50     }
51 }
52 }
53
54 // Fetch flights from the database based on leaving from and going to parameters
55 $sql_fetch = "SELECT * FROM flights";
56 if($_GET['leavingFrom']) {
57     $leavingFrom = $_GET['leavingFrom'];
58     $goingTo = $_GET['goingTo'];
59     $sql_fetch .= " WHERE LeavingFrom = '$leavingFrom' AND GoingTo = '$goingTo'";
60 }
61
62 $result_fetch = $conn->query($sql_fetch);
63
64 // Convert result to JSON and output
65 $flights = [];
66 if ($result_fetch->num_rows > 0) {
67     while($row = $result_fetch->fetch_assoc()) {
68         $flights[] = $row;
69     }
70 }
71 echo json_encode($flights);
72
```

Down below the script for our main itinerary page can be seen. It pulls the information from the flight table.

```
792   <script>
793     <document.addEventListener("DOMContentLoaded", function() {
794       const flightList = document.getElementById('flight-list');
795       const leavingFromInput = document.getElementById('leavingFrom');
796       const goingToInput = document.getElementById('goingTo');
797
798       // Function to format date specifically for flight information
799       function formatDate(dateString) {
800         const date = new Date(dateString);
801         const options = { year: 'numeric', month: 'long', day: 'numeric', hour: 'numeric', minute: 'numeric', hour12: true };
802         return date.toLocaleDateString('en-US', options);
803     }
804
805     function fetchFlights() {
806       fetch('flights.php')
807         .then(response => response.json())
808         .then(data => {
809           |   populateFlights(data);
810         })
811         .catch(error => {
812           |   console.error('Error fetching flight data:', error);
813         });
814     }
815
816     function fetchFilteredFlights() {
817       const leavingFrom = leavingFromInput.value.trim();
818       const goingTo = goingToInput.value.trim();
819
820       fetch(`flights.php?leavingFrom=${leavingFrom}&goingTo=${goingTo}`)
821         .then(response => response.json())
822         .then(data => {
823           |   populateFlights(data);
824         })
825         .catch(error => {
826           |   console.error('Error fetching filtered flight data:', error);
827         });
828     }
829
830     function populateFlights(data) {
```

```
830     function populateFlights(data) {
831         flightList.innerHTML = ''; // Clear previous flight cards
832         data.forEach(flight => {
833             const card = document.createElement('div');
834             card.classList.add('card');
835             card.innerHTML =
836                 `![Flight Image](${flight.FlightImagePaths})
837                 <p><strong>Leaving From:</strong> ${flight.LeavingFrom}</p>
838                 <p><strong>Going To:</strong> ${flight.GoingTo}</p>
839                 <p><strong>Departure:</strong> ${formatDate(flight.FlightDeparture)}</p>
840                 <p><strong>Arrival:</strong> ${formatDate(flight.FlightArrival)}</p>
841                 <p><strong>Return Departure:</strong> ${formatDate(flight.ReturnFlightDeparture)}</p>
842                 <p><strong>Return Arrival:</strong> ${formatDate(flight.ReturnFlightArrival)}</p>
843                 <p><strong>Flight Number:</strong> ${flight.FlightNumber}</p>
844                 <p><strong>Flight Website:</strong> ${flight.FlightCompanyURL}</p>
845                 <p class="price"><strong>Price:</strong> ${flight.FlightPrice}</p>
846             </div>
847             <button class="card-button">Add</button>
848         `;
849         flightList.appendChild(card);
850     });
851 }
852
853 // Fetch and populate all flights when the page loads
854 fetchFlights();
855
856 document.getElementById('searchButton').addEventListener('click', function() {
857     fetchFilteredFlights();
858 });
859
860 // Attach event listeners to the flight cards for the Add button
861 flightList.addEventListener('click', function(event) {
862     if (event.target.classList.contains('card-button')) {
863         const card = event.target.closest('.card');
864         // Extract and update information
865         updateFlightInfo(card);
866     }
867 });
868
869 function updateFlightInfo(card) {
870     const flightInfo = extractFlightInfo(card);
871     displayFlightInfo(flightInfo);
872 }
873
874 function extractFlightInfo(card) {
875     return {
```

```
875  v  return {  
876    leavingFrom: card.querySelector('p:nth-child(2)').textContent.replace('Leaving From:', '').trim(),  
877    goingTo: card.querySelector('p:nth-child(3)').textContent.replace('Going To:', '').trim(),  
878    departure: card.querySelector('p:nth-child(4)').textContent.replace('Departure:', '').trim(),  
879    arrival: card.querySelector('p:nth-child(5)').textContent.replace('Arrival:', '').trim(),  
880    returnDeparture: card.querySelector('p:nth-child(6)').textContent.replace('Return Departure:', '').trim(),  
881    returnArrival: card.querySelector('p:nth-child(7)').textContent.replace('Return Arrival:', '').trim(),  
882    flightNumber: card.querySelector('p:nth-child(8)').textContent.replace('Flight Number:', '').trim(),  
883    flightCompanyURL: card.querySelector('p:nth-child(9)').textContent.replace('Flight Website:', '').trim(),  
884    flightPrice: card.querySelector('.price').textContent.replace('Price:', '').trim()  
885  };  
886 };  
887  
888  v  function displayFlightInfo(flightInfo) {  
889    document.getElementById('flight-leaving-from').textContent = flightInfo.leavingFrom;  
890    document.getElementById('flight-going-to').textContent = flightInfo.goingTo;  
891    document.getElementById('flight-departure').textContent = flightInfo.departure;  
892    document.getElementById('flight-arrival').textContent = flightInfo.arrival;  
893    document.getElementById('flight-return-departure').textContent = flightInfo.returnDeparture;  
894    document.getElementById('flight-return-arrival').textContent = flightInfo.returnArrival;  
895    document.getElementById('flight-number').textContent = flightInfo.flightNumber;  
896    // Optionally, update the link back to the card if you have an ID for each card  
897    const backlink = document.querySelector('#bottom-flight-box a');  
898    backLink.href = flightInfo.flightCompanyURL;  
899    backLink.textContent = 'Flight Website';  
900  }  
901});  
902 v </script>
```

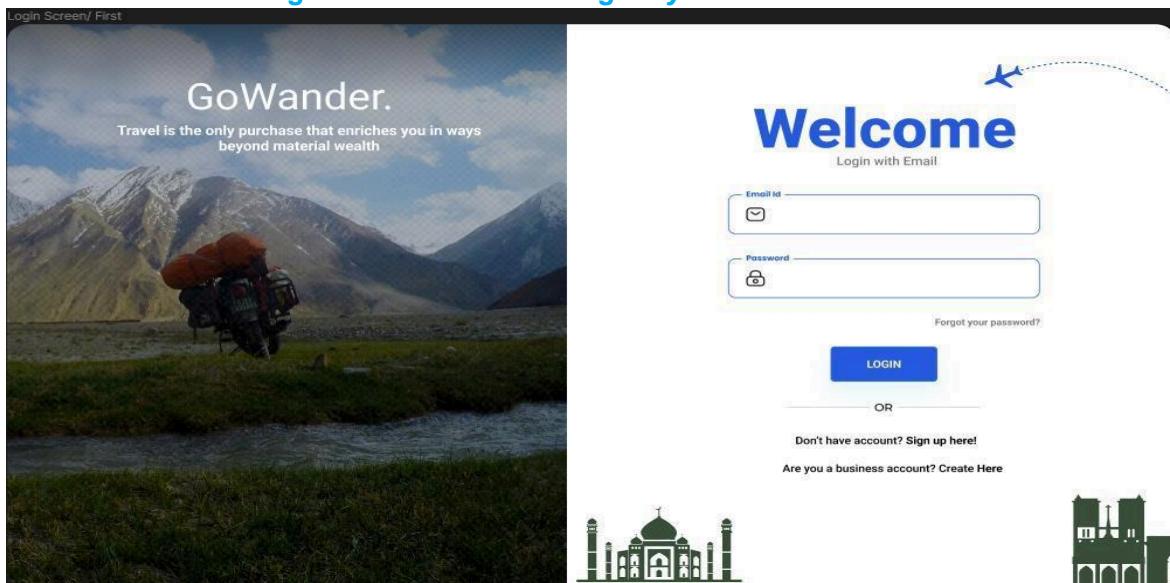
## 5.5 User Interface Design –

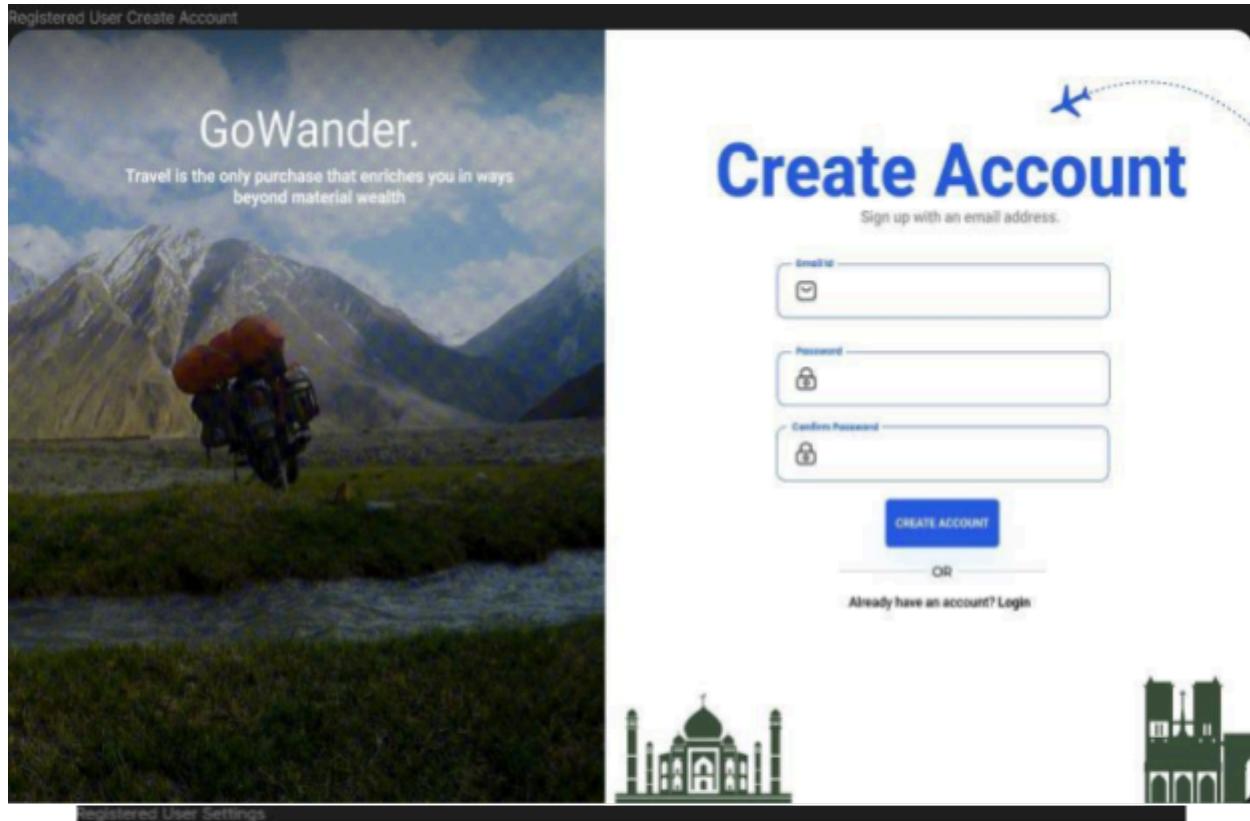
*Instructions: Insert any user interface design documents or provide a reference to where they are stored.*

The user interface design can be found in Section 3.4. These interface designs were created in Figma. Once the travel agency website is completed the updated user interface designs will be updated in this section (04/12/24).

Down below you can find screenshots taken from Figma that GoWander will be trying to replicate for our travel agency website.

*Registered User Travel Agency Website Interface*





Registered User Settings

The screenshot shows the 'Edit profile' page under 'settings'. On the left, there's a sidebar with navigation links: 'Edit profile' (selected, indicated by a pencil icon), 'Log Out', and 'Delete profile'. At the top right is a circular profile picture of a man. The main form contains fields for 'First Name' (Mehrab), 'Last Name' (Bozorgi), 'Email' (Mehrabbozorgi.business@gmail.com), 'Address' (33062 Zboncak isle), 'Contact Number' (58077.79), 'City' (Kent), 'State' (Ohio), and 'Password' (sbdfbnd65sfdb). Each field has a checked checkbox to its right. At the bottom are 'Cancel' and 'Save' buttons, with a gear icon in the bottom-left corner of the sidebar.

Registered User Home Page

GoWander.

Home

Search

Scroll to start planning now.

Start date → End date

January 2024 February 2024

Flights:

**spirit** airlines

Cleveland (CLE) - Maui (OGG)

Highway 101

→ 6:00 - 12:00pm | 14h 0m (1 Stop)  
28,300 in 2 bags

→ Carry on included

\$568 including taxes & fees

**spirit** airlines

Cleveland (CLE) - Maui (OGG)

Highway 101

→ 6:00 - 12:00pm | 14h 0m (1 Stop)  
28,300 in 2 bags

→ Carry on included

\$568 including taxes & fees

**spirit** airlines

Cleveland (CLE) - Maui (OGG)

Highway 101

→ 6:00 - 12:00pm | 14h 0m (1 Stop)  
28,300 in 2 bags

→ Carry on included

\$568 including taxes & fees

**spirit** airlines

Cleveland (CLE) - Maui (OGG)

Highway 101

→ 6:00 - 12:00pm | 14h 0m (1 Stop)  
28,300 in 2 bags

→ Carry on included

\$568 including taxes & fees

Hotels:

The Westin Princeville Ocean Resort Villas

3.0 stars

• Pool  
• Hot Tubs  
• Spectacular villas on the North Shore. Perfect for a resort stay, all rooms are fully equipped with a kitchen, washer, and dry clothes.

\$1,500 including taxes & fees

The Ritz-Carlton Maui, Kapalua Fire Lane Ocean View

3.0 stars

• Pool access and ocean views  
• Fire Pit and Cabana fire pits  
• Enjoy the ultimate fire pit experience at the Ritz-Carlton Maui, Kapalua. The only resort that is a haven with their private fire pits, including a fire pit cabana.

\$6,000 including taxes & fees

Hana-Maui Resort King Studio with Partial Ocean View

3.0 stars

• Private Cabana  
• Use of Tennis Courts  
• Walk up to the gates of the most iconic of Maui resorts in one long walk with a paved path complete the landscape.

\$5,750 including taxes & fees

Events:

Hawaii Snorkeling

1 hour tour

Depart from Maui shores on a boat, enjoying scenic views of the coastline and island peaks. Arrive at a snorkeling site with clear waters, where you'll see several fish and other marine coastal creatures. Encounter a variety of sea creatures, including tropical fish, sea turtles, and playful dolphins. Soak up the feeling of relaxation and wonder, soaking in the warm Hawaiian sun and cherishing unforgettable memories of the snorkeling adventure.

\$200 per person

Kō - Best Maui Dining

2 hour tour

Kō, which translates to "staple" or "Hawaiian food," inspired by the signature plantation era, a time when people worked the land and could only afford Hawaiian Islands to work in the sugar cane fields. This plantation evolved into bright, spacious and cooking techniques from Hawaii, while utilizing the ingredients available to them in the islands. This culinary evolution, eventually developed into what is now considered modern Hawaiian cuisine.

\$50 - 100 per person

Hawaii Hiking

3 hour tour

Depart from Maui's shores on a trail, views of the coastline and volcanic peaks. The trail leads through lush tropical forests, up and down hills, and into crystal-clear streams where a variety of sea creatures, including fish, sea turtles, and playful dolphins, can be seen. The hike features a mix of natural and man-made structures, including a suspension bridge and a waterfall, providing a unique perspective on the natural beauty of the Hawaiian Islands.

\$200 per person

**Itinerary:**Start date → End date  
YearHawaii Hiking  
Monday PM, 8th  
3 Hour \$200 per person**Flights:**

<input type="text"/> 9	<input type="button" value="Departing"/>	<input type="text"/> 9	<input type="button" value="Arriving"/>
Departing		Arriving	
<input type="text"/> 9	<input type="button" value="Departing"/>	<input type="text"/> 9	<input type="button" value="Arriving"/>
Departing		Arriving	

**Hotels:**

<input type="text"/>	<input type="button" value="Check-in"/>	<input type="text"/>	<input type="button" value="Check-out"/>
<input type="text"/>	<input type="button" value="Check-in"/>	<input type="text"/>	<input type="button" value="Check-out"/>

**Events:**Hawaii Hiking  
Friday PM, 12th  
3 Hour \$200 per personKA Private Dining Event:  
Thursday PM, 11th  
2 Hour \$150 - \$200 per personThe Big Cottonwood, Kapolei  
Friday PM, 10th - Saturday PM, 11th  
\$1000

	Mon	Tue	Wed	Thu	Fri	Sat	Sun
8:00am							
10:00am							
11:00am							
12:00pm							
1:00pm							
2:00pm							
3:00pm							
4:00pm							
5:00pm							

Hawaii  
Hiking**GoWander.**[Contact Us](#)[Site Map](#)[Terms of Use](#)[Privacy and Cookies Statement](#)

© 2011 GoWander Inc.



Blown Up Registered User Events Detail Page



## Hawaiian Hiking \$200<sup>Per person</sup>

3 Hour Event

Depart from Maui's shores on a boat, enjoying scenic views of the coastline and volcanic peaks. Arrive at a secluded bay abundant with colorful marine life, don snorkel gear, and dive into crystal-clear waters. Encounter a variety of sea creatures, including tropical fish, sea turtles, and playful dolphins. Surface feeling refreshed and exhilarated, basking in the warm Hawaiian sun and cherishing unforgettable memories of the snorkeling adventure.

**Ages:** 18-99

**Start Time:** 8am,10am,12pm,2pm

**Language:** English

### Highlights

\*View amazing sights of Hawaii Big Island

\*Encounter wildlife of Hawaii

\*Enrichment opportunity to learn about Hawaii

\*Designated look out points great for photography

[Add](#)

### 3<sup>rd</sup> Party Business Manager Travel Agency Website Interface

Login Screen/ First



## Welcome

Login with Email

Email Id

Password

[Forgot your password?](#)

**LOGIN**

OR

[Don't have account? Sign up here!](#)

[Are you a business account? Create Here](#)



3rd Party Business Manager Create Account



## Create Account

[Sign up to start listing events now!](#)

Business Name

Contact Name

Business Email

Business Phone

Password

Confirm Password

Brief Business Description

Business Address

Website URL (Optional)

**SUBMIT**



3rd Party Business Manager Settings

Manage Events

Hawaiian Event Company

 Hawaiian Hiking

Depart from Maui's shores on a boat, enjoying scenic views of the coastline and volcanic peaks. Arrive at a secluded bay abundant with colorful marine life, don snorkel gear, and dive into crystal-clear waters. Encounter a variety of sea creatures, including tropical fish, sea turtles, and playful dolphins. Surface feeling refreshed and exhilarated, basking in the warm Hawaiian sun and cherishing unforgettable memories of the snorkeling adventure.

\$200 Per person [View](#)



Up 3rd Party Business Manager Events Detail Page (Edit)





Event Name: Hawaiian Hiking

Price: \$200 Per person

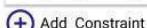
Average Time: 3 Hour Event

Description:

Depart from Maui's shores on a boat, enjoying scenic views of the coastline and volcanic peaks. Arrive at a secluded bay abundant with colorful marine life, don snorkel gear, and dive into crystal-clear waters. Encounter a variety of sea creatures, including tropical fish, sea turtles, and playful dolphins. Surface feeling refreshed and exhilarated, basking in the warm Hawaiian sun and cherishing unforgettable memories of the snorkeling adventure.

Ages: 18-99  
Start Time: 8am,10am,12pm,2pm  
Language: English

\*View amazing sights of Hawaii Big Island  
\*Encounter wildlife of Hawaii  
\*Enrichment opportunity to learn about Hawaii  
\*Designated look out points great for photography

 Add Constraint

 Add Highlight



## Blown Up 3rd Party Business Manager Events Detail Page



## Hawaiian Hiking \$200Per person

3 Hour Event

Depart from Maui's shores on a boat, enjoying scenic views of the coastline and volcanic peaks. Arrive at a secluded bay abundant with colorful marine life, don snorkel gear, and dive into crystal-clear waters. Encounter a variety of sea creatures, including tropical fish, sea turtles, and playful dolphins. Surface feeling refreshed and exhilarated, basking in the warm Hawaiian sun and cherishing unforgettable memories of the snorkeling adventure.

**Ages:** 18-99

**Start Time:** 8am,10am,12pm,2pm

**Language:** English

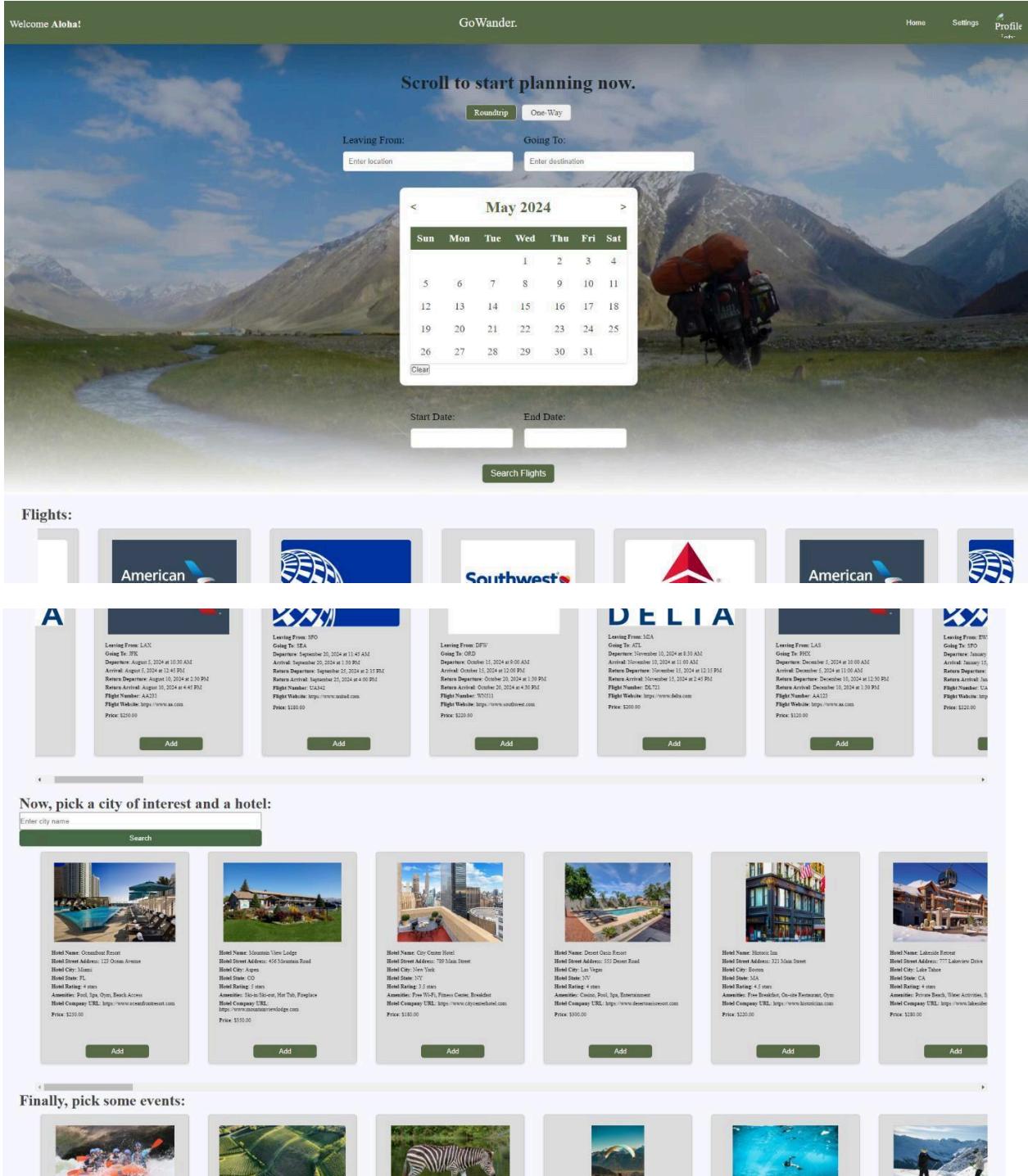
### Highlights

- \*View amazing sights of Hawaii Big Island
- \*Encounter wildlife of Hawaii
- \*Enrichment opportunity to learn about Hawaii
- \*Designated look out points great for photography

[Delete](#)[Edit](#)

The last images were our wireframe. The images down below will be from the GoWander Travel Agency website. It will include up-to-date user interfaces.

## Main Itinerary Page



Welcome Aloha!

GoWander.

Scroll to start planning now.

Leaving From:  Going To:

One-Way Roundtrip

May 2024

Sun	Mon	Tue	Wed	Thu	Fri	Sat
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

Start Date:  End Date:

Search Flights

**Flights:**

- American
- KLM
- Southwest
- Delta
- American
- KLM

**A**

Leaving From: LAX  
Going To: JFK  
Departure: August 1, 2024 at 03:55 AM  
Arrival: August 1, 2024 at 10:45 PM  
Return Departure: August 10, 2024 at 2:30 PM  
Return Arrival: August 10, 2024 at 4:45 PM  
Flight Number: AA237  
Flight Website: [www.aa.com](http://www.aa.com)  
Price: \$200.00

Add

Leaving From: SFO  
Going To: IAD  
Departure: September 20, 2024 at 11:45 AM  
Arrival: September 20, 2024 at 1:30 PM  
Return Departure: September 25, 2024 at 2:30 PM  
Return Arrival: September 25, 2024 at 4:00 PM  
Flight Number: UA342  
Flight Website: [www.united.com](http://www.united.com)  
Price: \$180.00

Add

Leaving From: DFW  
Going To: OBD  
Departure: October 10, 2024 at 09:00 AM  
Arrival: October 10, 2024 at 11:00 AM  
Return Departure: October 15, 2024 at 12:00 PM  
Return Arrival: October 15, 2024 at 1:30 PM  
Flight Number: UA342  
Flight Website: [www.united.com](http://www.united.com)  
Price: \$220.00

Add

Leaving From: MIA  
Going To: ATL  
Departure: November 10, 2024 at 09:30 AM  
Arrival: November 10, 2024 at 11:30 AM  
Return Departure: November 15, 2024 at 12:30 PM  
Return Arrival: November 15, 2024 at 2:30 PM  
Flight Number: DL721  
Flight Website: [www.delta.com](http://www.delta.com)  
Price: \$200.00

Add

Leaving From: EWR  
Going To: PHL  
Departure: December 5, 2024 at 10:00 AM  
Arrival: December 5, 2024 at 11:00 AM  
Return Departure: December 10, 2024 at 12:30 PM  
Return Arrival: December 10, 2024 at 1:30 PM  
Flight Number: AA233  
Flight Website: [www.aa.com](http://www.aa.com)  
Price: \$200.00

Add

**Now, pick a city of interest and a hotel:**

Enter city name  Search

 Hotel Name: Oceanfront Resort Hotel Street Address: 123 Ocean Avenue Hotel City: Miami Hotel State: FL Hotel Rating: 4 stars Additional Info: Spa, Gym, Beach Access Hotel Company URL: <a href="http://www.oceanfrontresort.com">www.oceanfrontresort.com</a> Price: \$250.00	 Hotel Name: Mountain View Lodge Hotel Street Address: 456 Mountain Road Hotel City: Aspen Hotel State: CO Hotel Rating: 3 stars Additional Info: Skiing, Hot Tub, Fireplace Hotel Company URL: <a href="http://www.mountainviewlodge.com">www.mountainviewlodge.com</a> Price: \$350.00	 Hotel Name: City Center Hotel Hotel Street Address: 789 Main Street Hotel City: New York Hotel State: NY Hotel Rating: 5 stars Additional Info: Gym, Pool, Fitness Center, Breakfast Hotel Company URL: <a href="http://www.citycenterhotel.com">www.citycenterhotel.com</a> Price: \$500.00	 Hotel Name: Desert Oasis Resort Hotel Street Address: 123 Main Street Hotel City: Las Vegas Hotel State: NV Hotel Rating: 4 stars Additional Info: Spa, Pool, Gym, Entertainment Hotel Company URL: <a href="http://www.desertoasisresort.com">www.desertoasisresort.com</a> Price: \$350.00	 Hotel Name: Historic Inn Hotel Street Address: 321 Main Street Hotel City: Boston Hotel State: MA Hotel Rating: 4 stars Additional Info: Historic, Quaint, Permanent, Quiet Hotel Company URL: <a href="http://www.historicinn.com">www.historicinn.com</a> Price: \$220.00	 Hotel Name: Lakeside Resort Hotel Street Address: 777 Lakeside Drive Hotel City: Lake Tahoe Hotel State: CA Hotel Rating: 4 stars Additional Info: Skiing, Rock Climbing, Water Activities, N Hotel Company URL: <a href="http://www.lakesideresort.com">www.lakesideresort.com</a> Price: \$350.00
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Add Add Add Add Add Add

**Finally, pick some events:**

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

**Whitewater Rafting Expedition**  
Business: Hiking  
Description: Experience the adrenaline rush of navigating rapids with our expert guides.  
Times: 09:00 AM  
Dates: 2024-07-15  
Location: Colorado Springs, CO

**Cycling Tour of Wine Country**  
Business: Adventure Tours  
Description: Pedal through scenic vineyards and enjoy wine tastings along the route.  
Times: 10:00 AM  
Dates: 2024-08-20  
Location: San Diego, CA

**Wildlife Photography Safari**  
Business: Outdoors Gear Emporium  
Description: Capture stunning images of wildlife in their natural habitat with guidance from professional photographers.  
Times: 07:00 AM  
Dates: 2024-09-10  
Location: Orlando, FL

**Paragliding Adventure**  
Business: Nature Necessities  
Description: Soar through the skies and enjoy breathtaking views of Aspen's landscape.  
Times: 11:00 AM  
Dates: 2024-07-25  
Location: Aspen, CO

**Scuba Diving Expedition**  
Business: Wilderness Wonders  
Description: Explore the vibrant underwater world of Maui's coral reefs.  
Times: 08:00 AM  
Dates: 2024-08-05  
Location: Maui, HI

**Skiing Lessons**  
Business: Trailblazers  
Description: Learn the basics or improve skills on Vail's pristine slopes.  
Times: 10:00 AM  
Dates: 2024-12-12  
Location: Vail, CO

**GoWander Personal Itinerary**

Start Date: 04/28/2024

End Date: 05/04/2024

**Flights:**

Leaving From: City A  
Going To: City B  
Departure: MM/DD/YYYY  
Arrival: MM/DD/YYYY  
Return Departure: MM/DD/YYYY  
Return Arrival: MM/DD/YYYY  
Flight Number: XXXX

**Hotels:**

Hotel Name: Hotel Name  
Location: City, State  
Amenities:

< > today
**May 5 – 11, 2024**
month week day list
**Draggable Events**[Clear All Events](#)

	Sun 5/5	Mon 5/6	Tue 5/7	Wed 5/8	Thu 5/9	Fri 5/10	Sat 5/11
all-day							
12am							
1am							
2am							

Flight Number: XXXX

< > today
**May 5 – 11, 2024**
month week day list
**Draggable Events**[Clear All Events](#)

	Sun 5/5	Mon 5/6	Tue 5/7	Wed 5/8	Thu 5/9	Fri 5/10	Sat 5/11
all-day							
12am							
1am							
2am							
3am							
4am							
5am							
6am							
7am							
8am							
9am							
10am							
11am							
12pm							

## Edit Profile Page

Home      Manage Events      Edit Profile      Log Out      Delete Profile

Edit Profile



<b>Business Name:</b>	Aloha Travel
<b>Business Email:</b>	AlohaTravel@gmail.com
<b>Business Street Address:</b>	1739 Ala Moana Blvd, Honolulu, HI 96815
<b>Business State :</b>	HI
<b>Business City:</b>	Honolulu
<b>Business Phone Number:</b>	808-575-2262
<b>Password:</b>	*****
<b>Brief Business Description:</b>	Providing beautiful experiences in Honolulu
<b>Website URL (Optional):</b>	AlohaTravel.com

**Save Changes**

## Manage Events Page

localhost/test/business\_manage\_event.php?username=Aloha123

**Business Events**

Event Type	Description	Date	Time	Price	Action
Surfing Lesson	Learn to catch the waves with our experienced instructors.	2024-04-28	9:00 AM	\$ 80.00	For person <a href="#">View</a> <a href="#">Delete</a>
Snorkeling Adventure	Discover the beauty of Honolulu's reefs and marine life.	2024-05-01	12:00 PM	\$ 120.00	For person <a href="#">View</a> <a href="#">Delete</a>
Honolulu Luau Festival	Experience the rich culture of Hawaii at our annual luau festival.	2024-05-03	All Day	\$ 150.00	For person <a href="#">View</a> <a href="#">Delete</a>
Scuba Diving Expedition	Explore the vibrant underwater world of Honolulu.	2024-04-30	9:00 AM	\$ 150.00	For person <a href="#">View</a> <a href="#">Delete</a>
Whale Watching Tour	Spot humpback whales in their natural habitat.	2024-04-28	2:00 PM	\$ 50.00	For person <a href="#">View</a> <a href="#">Delete</a>
Bike Tour	Explore the scenic beauty of Honolulu on a guided bike tour.	2024-05-01	9:30 AM	\$ 75.00	For person <a href="#">View</a> <a href="#">Delete</a>

Create new event: [+](#)

## Register Events Page

← Manage Events

### Register Event

Event Images

Event Name

Event Price

Event Length (in hours)

Event Description

Event Ages

Event Times

All Day

Event Date / Start Date

(Optional) Event End Date

Event Street Address

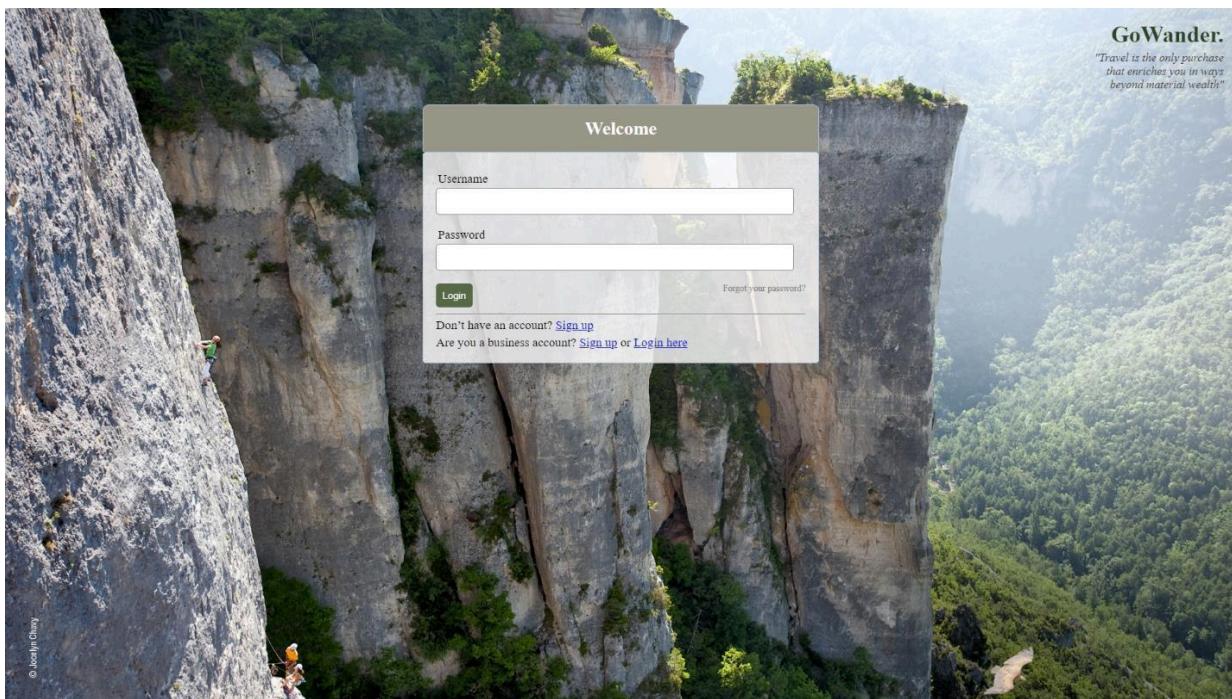
Event City

Event State

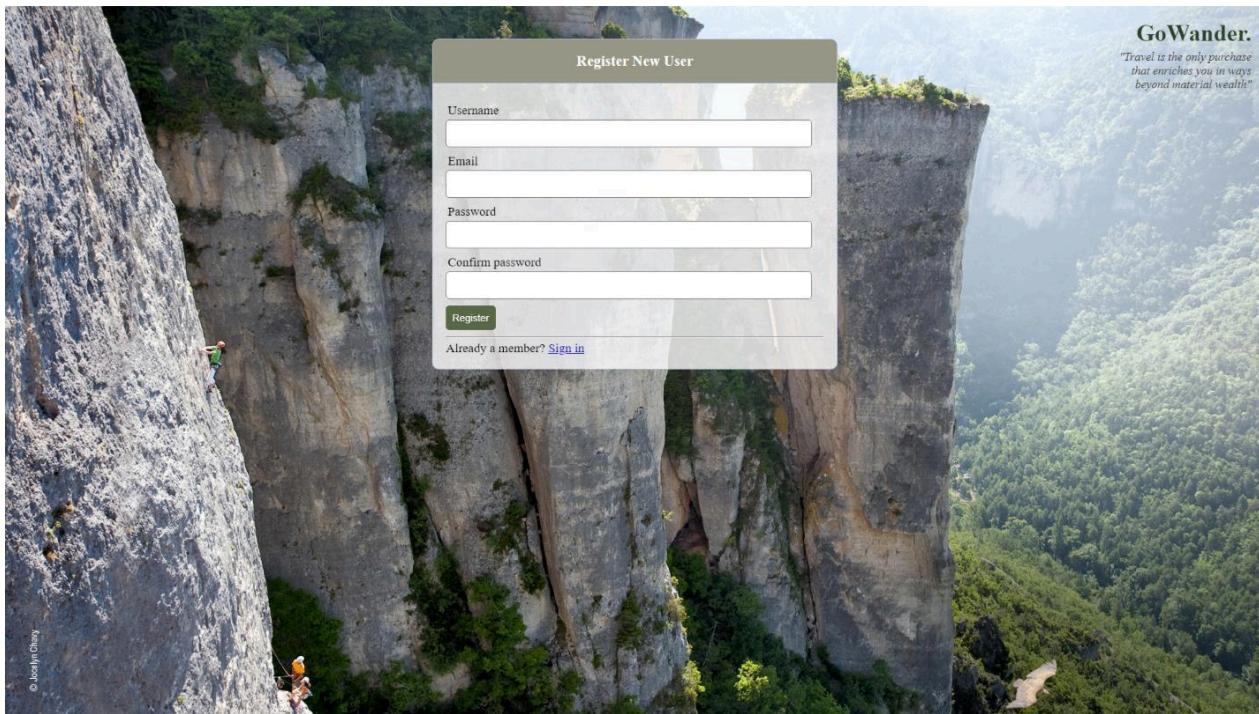
Event Highlights

Register Event

## User Login Page

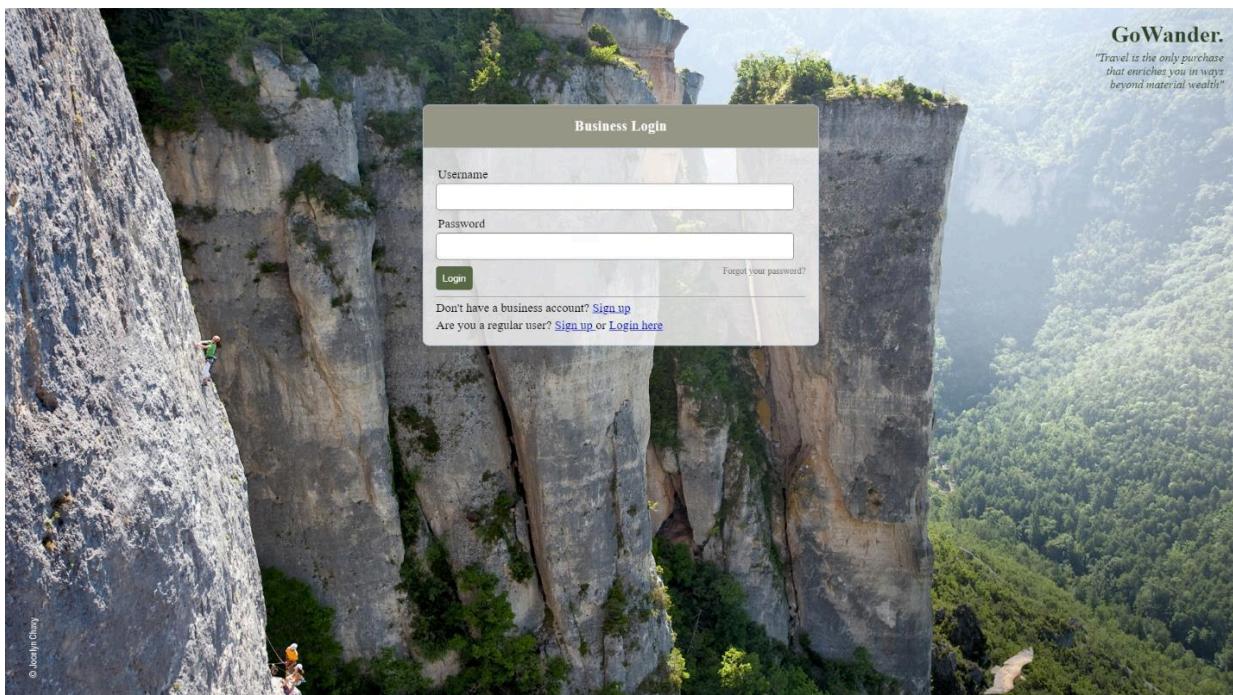


## Register New User Page



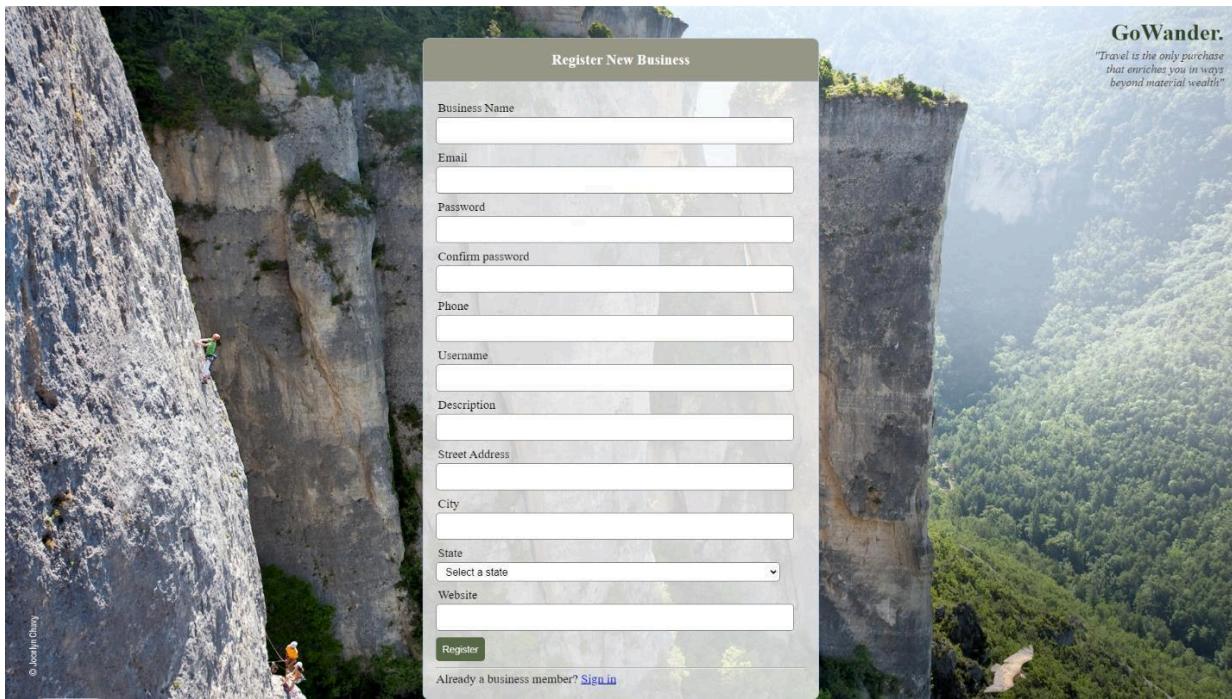
A photograph of a limestone cliff face with two climbers. One climber is high up on the left, and another is lower down on the right. A registration form is overlaid on the upper right portion of the image. The form has a dark header "Register New User". It contains four input fields: "Username", "Email", "Password", and "Confirm password". Below these is a green "Register" button. At the bottom of the form, it says "Already a member? [Sign in](#)". In the top right corner of the image, there is a logo for "GoWander" with the tagline "Travel is the only purchase that enriches you in ways beyond material wealth".

## Business Login Page



A photograph of a limestone cliff face with two climbers. One climber is high up on the left, and another is lower down on the right. A login form is overlaid on the upper right portion of the image. The form has a dark header "Business Login". It contains two input fields: "Username" and "Password". Below these is a green "Login" button. To the right of the "Login" button is a link "Forgot your password?". At the bottom of the form, it says "Don't have a business account? [Sign up](#)" and "Are you a regular user? [Sign up](#) or [Login here](#)". In the top right corner of the image, there is a logo for "GoWander" with the tagline "Travel is the only purchase that enriches you in ways beyond material wealth".

## Register New Business Page



The background image shows a person climbing a large, vertical rock face. To the right, there is a vertical banner with the text "GoWander." and a quote: "Travel is the only purchase that enriches you in ways beyond material wealth".

**Register New Business**

Business Name

Email

Password

Confirm password

Phone

Username

Description

Street Address

City

State

Website

**Register**

Already a business member? [Sign in](#)

## 6. Project Testing Plan

---

### 6.1 User Create Account

1. The new user directs the web browser to the GoWander welcome page.
2. The system displays the welcome page.
3. The new user clicks on “Don’t have an account?” Sign-up button.
4. The system displays a sign-up page.
5. The new user enters in their email, username, and password into the account creation form.
6. The new user clicks the “Register” button.
7. The system processes the information to verify all required information has been entered.
  - a. If credentials are invalid, the system displays an error message and highlights the errors.
  - b. If credentials are valid, the system creates the account and updates the user field in the GoWander database with the information.
8. The Registered User is displayed on the main itinerary page with a welcome message in the upper left corner.

### 6.2 User Login

1. The new user directs the web browser to the GoWander welcome page.
2. System displays the welcome page.
3. The new user clicks on “Don’t have an account?” sign-up button.
4. The system displays the sign-up page.
5. The new user enters their email, username, and password into the account creation form.
6. The new user clicks the “Register” button.
7. The system processes information to verify all required information has been entered.
  - a. If credentials are invalid, the system displays an error message and highlights the errors.
  - b. If credentials are valid, the system creates the account and updates the user field in the Go Wander database with the information.
8. The Registered User is displayed on the main itinerary page with a welcome message in the upper left corner.

### 6.3 Business Create Account

1. 3<sup>rd</sup> Party Business Manager directs the web browser to the GoWander welcome page.
2. The system displays the welcome page.

3. 3<sup>rd</sup> Party Business Manager clicks on “Are you a business account?” Sign up button.
4. The system displays the register new business page.
5. 3<sup>rd</sup> Party Business Manager enters the business name, email, password, phone number, username, description of the business, street address, city, state, and website.
6. 3<sup>rd</sup> Party Business Manager clicks the “Register” button.
7. The system processes information to verify all required information has been entered.
  - a. If credentials are invalid, the system displays an error message and highlights the errors.
  - b. If credentials are valid, the system creates the account and updates the businesses field in the GoWander database with the information.
8. 3<sup>rd</sup> Party Business Manager is displayed on the main itinerary page with a welcome message in the upper left corner.

#### **6.4 *Business Login***

1. The system assumes the 3rd Party Business Manager has completed account setup.
2. 3<sup>rd</sup> Party Business Manager directs the web browser to the GoWander welcome page.
3. The system displays the welcome page.
4. 3<sup>rd</sup> Party Business Manager clicks on “Are you a business account?” Login here button.
5. The system displays the business login page.
6. 3<sup>rd</sup> Party Business Manager enters their username and password in the login form.
7. 3<sup>rd</sup> Party Business Manager clicks the “Login” button.
8. The system processes information to verify all required information has been entered.
  - a. If credentials are invalid, the system displays an error message and highlights the errors.
  - b. If credentials are valid, the system takes the 3rd Party Business Manager to the main itinerary page with a welcome message in the upper left corner.

#### **6.5 *Business Settings – Edit Profile***

1. It is assumed that the 3<sup>rd</sup> Party Business Manager has completed the login.
2. The 3<sup>rd</sup> Party Business Manager will click the settings button in the top right corner.
3. The system will display the main settings page.
4. The 3<sup>rd</sup> Party Business Manager needs to click the edit profile button on the left-hand side.
5. The edit profile page will be displayed by the system.

## **6.6 Business Manage Events -**

1. The system assumes that the 3<sup>rd</sup> Party Business Manager has completed the business login.
2. 3<sup>rd</sup> Party Business Manager navigates to the edit profile page. This is done by clicking the settings button at the top right of the screen.
3. The 3<sup>rd</sup> Party Business Manager clicks on the manage events button on the left side of the screen.
4. The 3<sup>rd</sup> Party Business Manager views the events that they are hosting.
  - a. The 3<sup>rd</sup> Party Business Manager can either view or delete the events.

## **6.7 Business Register Event -**

1. The system assumes that the 3<sup>rd</sup> Party Business Manager has completed the business login.
2. 3<sup>rd</sup> Party Business Manager navigates to the edit profile page. This is done by clicking the settings button at the top right of the screen.
3. The 3<sup>rd</sup> Party Business Manager clicks on the manage events button on the left side of the screen.
4. The 3<sup>rd</sup> Party Business Manager views the events that they are hosting.
  - a. The 3<sup>rd</sup> Party Business Manager can either view or delete the events.
5. The 3<sup>rd</sup> Party Business Manager scrolls to the bottom of the screen and clicks the button underneath the create a new event text.
6. This then brings the 3<sup>rd</sup> Party Business Manager to the Register Event Page.
7. The 3<sup>rd</sup> Party Business Manager then needs to add all the necessary event details to the Register Event Page. The event details that need to be filled out are the event image, event name, event price, event length (in hours), event description, event ages, event times (there is an all-day event button), event date (start date), the optional event end date, event street address, event city, event state, and event highlights.
8. Once all the necessary information on the event is input into the Register Event Page the 3<sup>rd</sup> Party Business Manager needs to click on the Register Event button on the bottom of the page.
9. The event will then show up on the Manage Events page.

## **6.8 User Settings / Save Changes**

1. The system assumes that the Registered User is logged in and has clicked on the settings button on the main page to appear on the settings page.
2. The Registered User can manipulate the text boxes to change their information.
3. The Registered User can press the save changes button, which will allow the user to update their information.

- a. The system should update the information in the database so that by refreshing the page, the new data is displayed.
4. The Registered User can press the home button that will take them to the Registered User building an itinerary page.
5. The Registered User can press the log-out page, which will log them out of the system, taking them back to the user log-in page.
6. The Registered User can press the Delete Profile button, which will delete the Registered User's account and take the user to the log-in page.
  - a. The system will delete their row in the database containing all of their information.

## **6.9 *User Building an Itinerary - RA***

1. The system assumes that the Registered User has completed the Registered User login.
  - a. Credentials can be verified by viewing a welcome message followed by the username in the upper left-hand corner.
2. The Registered User should be able to see the entire page from the beginning with all cards populating in their respective fields.
3. The Registered User chooses between roundtrip and one way, this is done by clicking the round trip or one-way button.
4. The Registered User inputs a leaving from airport abbreviation and a going to airport abbreviation in their respective boxes.
5. The Registered User can click on two days on the calendar, upon two being clicked every day in between including the two dates highlighted in blue.
  - a. Simultaneously the start date and end date boxes should self populate with the start and end date that the user selected within the calendar.
6. The Registered User can click the "Search Flights" button.
7. The system should sort flights to only display flight cards within the dates that the Registered User selected.
8. The user can press the "Add" button on one of the flight cards, which will then populate the itinerary at the bottom.
9. The Registered User can input a city within the "Enter City Name" text box.
10. The search button for the "Enter City Name" Textbox will sort the hotel and event cards to display only events and hotels within the selected city.
11. The Registered User can select a hotel using the add button within the card.
  - a. The system will display the hotel selected at the button within the itinerary section.
12. The Registered User can press view on the event cards.
  - a. Upon clicking the view button the card blows up showing more details.
13. Within the blown-up card the Registered User can select event times.
14. The Registered User can press "Add" on the blown-up card to add an event to the itinerary.
15. The Registered User can drag events to the respective event to the respective day within the itinerary calendar.

16. The Registered User can save and delete itineraries.