Problem Statement

The company I work for, NextStop, is a large, publicly traded travel company that helps users find low and flexible flight, hotel, and destination rates. We not only specialize in providing access to low travel rates, but we also provide a best-in-class experience for our users by empowering them to build, schedule, and budget their entire trip all in one platform. Our objective is to be the industry leader in flexible travel and provide users with a best-in-class search, booking, and travel experience.

Our target audience is primarily millennials and Gen Zers who live an active and on the go lifestyle, commonly known as a digital nomad. During a booking experience, users go through each phase by picking a destination, flight, and hotel. Users may opt to skip choosing a hotel or flight. Over 75% of all user interaction comes through our iOS and Android mobile apps, so user experience and ease of use are one of our topmost priorities.

Having started my role at NextStop, my responsibility is to consolidate, structure, and organize the data that comes through the app. Currently, we have six primary entities, Accounts, Users, Trips, Flights, Destinations, and Hotels. Each piece of data is crucial to enhancing the overall user experience and internal decisions within the company as the information enables us to focus on getting the best rates possible on the most popular destinations. However, the issue is that although we are collecting pivotal data, each piece of data is stored in its own database. This database setup makes maintenance, system updates, and overall system architecture very complex and inefficient. My goal is to consolidate our data silos into one comprehensive database.

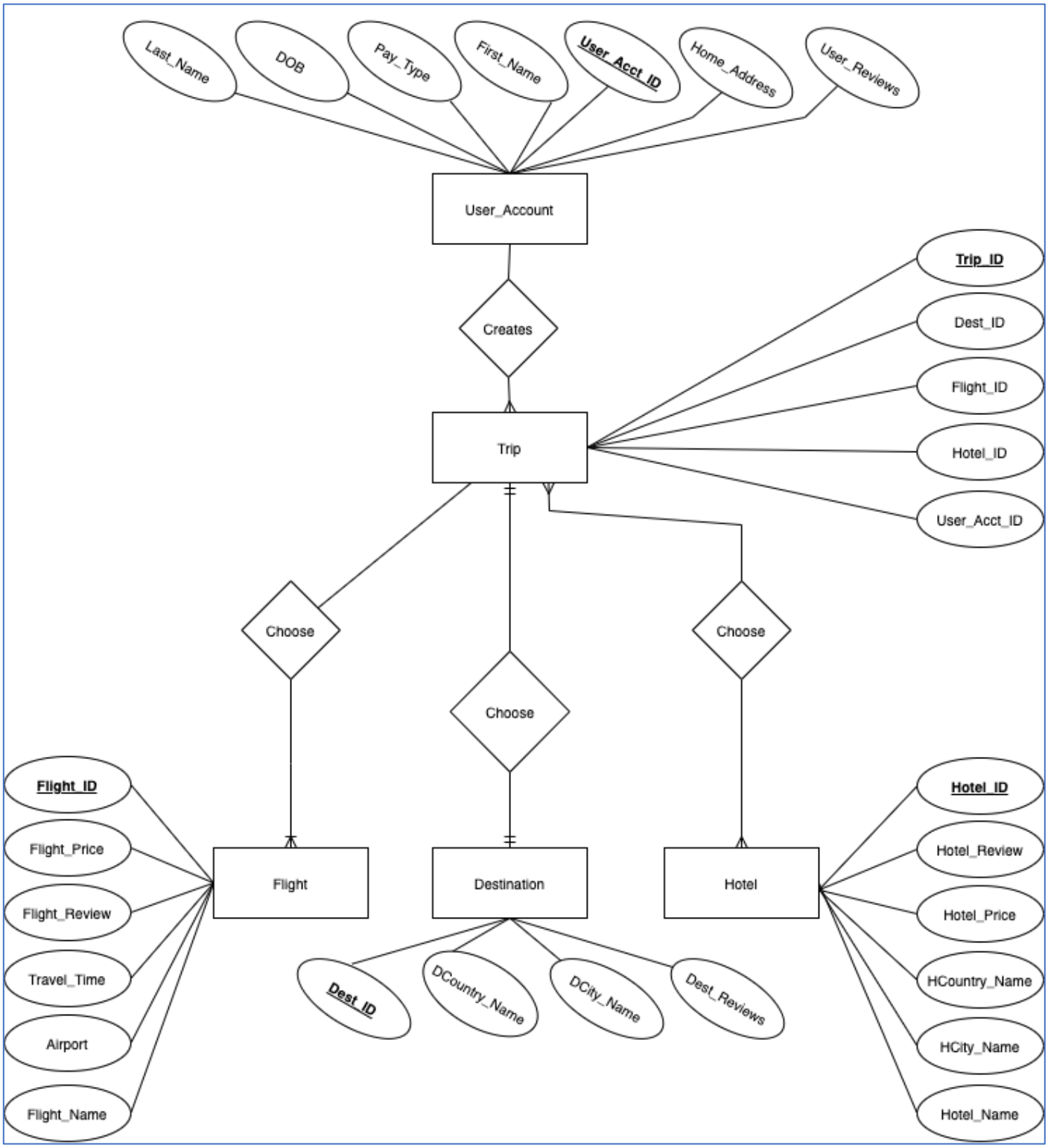
Conceptual Database Design

This database will be designed around six primary entities, which will be the foundation of our data. This data will enable us to make sound business decisions, provide customers with the results they want, and create a user-friendly experience, the destination, flight, hotel, account, user, and trip.

A user account is the end-user/customer who uses the platform to book specifics about the trip they wish to plan. Each user can browse the app, search for flights, hotels, etc., as much as they want. Users can have many trips, flights, hotels, and destinations, so there needs to be a way to track each instance back to the user. Each user's account is unique to them and only them. Accounts will contain trip reviews and personally identifiable information (PII) such as their date of birth and payment information.

A hotel, flight, and destination are all unique entities used to help the user schedule a trip. There are many similar attributes between the three entities, such as price, ID, reviews, location.

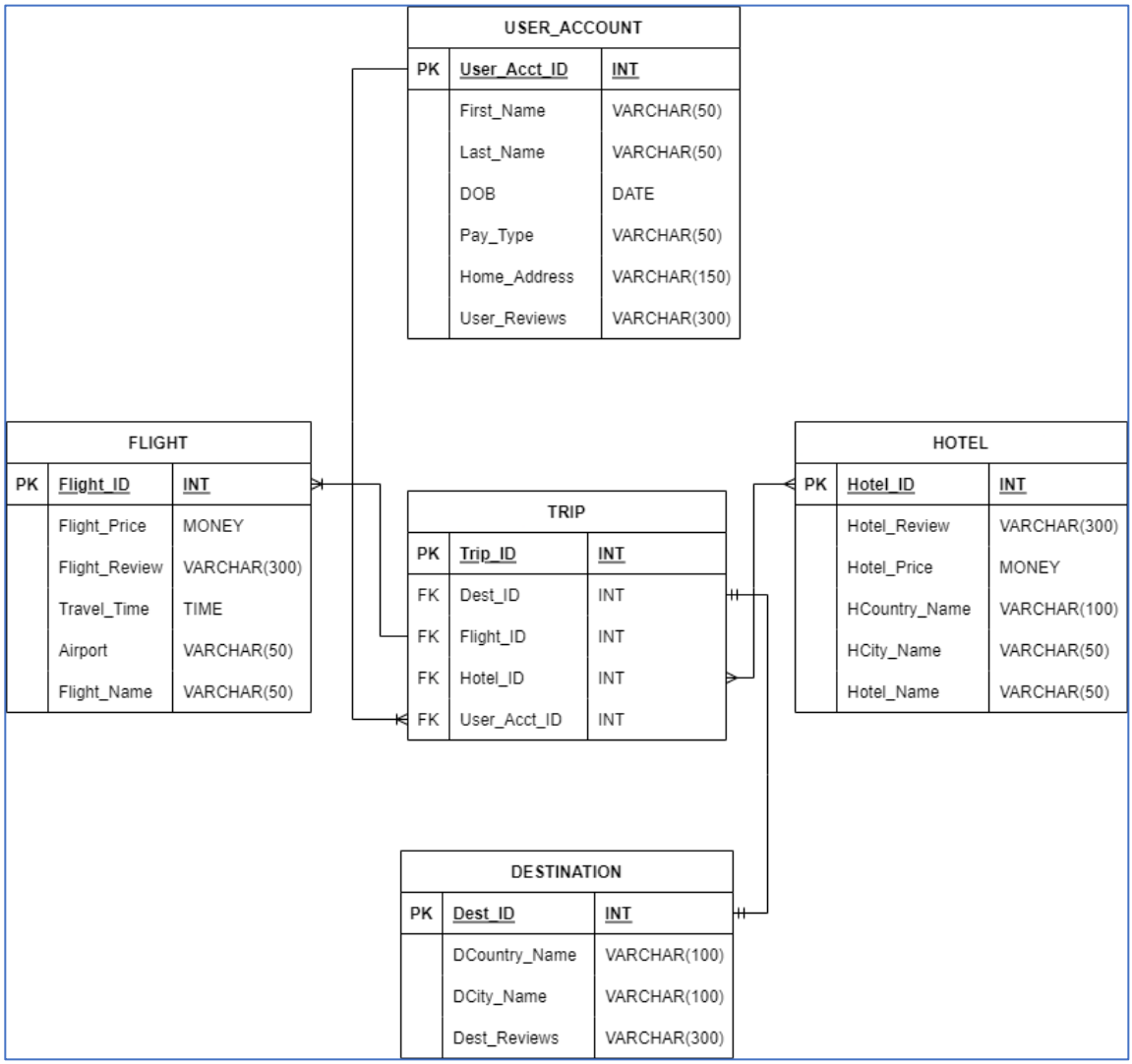
A trip is when a user successfully completes selecting a destination, hotel, and flight. Since a user can elect to skip picking a hotel or flight (i.e., they already have a place to stay and are driving, taking a train or bus), they can leave these options blank, and the database will read the values as NULL. The trip entity needs to be able to access data from other tables to reference the correct information about the trip accurately.


Last_Name
DOB
Pay_Type
First_Name
User_Acct_ID
Home_Address
User_Reviews
User_Account
Creates
Trip_ID
Dest_ID
Flight_ID
Hotel_ID
User_Acct_ID
Trip
Choose
Choose
Choose
Flight_ID
Flight_Price
Flight_Review
Travel_Time
Airport
Flight_Name
Flight
Destination
Hotel
Hotel_ID
Hotel_Review
Hotel_Price
HCountry_Name
HCity_Name
Hotel_Name
Dest_ID
DCountry_Name
DCity_Name
Dest_Reviews

Summary Table of Data Types

Table	Attribute	Type	Constraint
USER_ACCOUNT	User_Acct_ID	INT	PRIMARY KEY
USER_ACCOUNT	User_Reviews	VARCHAR(300)	
USER_ACCOUNT	Last_Name	VARCHAR(50)	NOT NULL
USER_ACCOUNT	DOB	DATE	NOT NULL
USER_ACCOUNT	Pay_Type	VARCHAR(50)	
USER_ACCOUNT	First_Name	VARCHAR(50)	NOT NULL
USER_ACCOUNT	Home_Address	VARCHAR(150)	NOT NULL
TRIP	Trip_ID	INT	PRIMARY KEY
TRIP	Dest_ID	INT	FOREIGN KEY
TRIP	Flight_ID	INT	FOREIGN KEY
TRIP	Hotel_ID	INT	FOREIGN KEY
TRIP	User_Acct_ID	INT	FOREIGN KEY
FLIGHT	Flight_ID	INT	PRIMARY KEY
FLIGHT	Flight_Price	MONEY	NOT NULL
FLIGHT	Flight_Review	VARCHAR(300)	
FLIGHT	Travel_Time	TIME	NOT NULL
FLIGHT	Airport	VARCHAR(50)	NOT NULL
FLIGHT	Flight_Name	VARCHAR(50)	NOT NULL
HOTEL	Hotel_ID	INT	PRIMARY KEY
HOTEL	Hotel_Review	VARCHAR(300)	
HOTEL	Hotel_Price	MONEY	NOT NULL
HOTEL	HCountry_Name	VARCHAR(100)	NOT NULL
HOTEL	HCity_Name	VARCHAR(50)	NOT NULL
HOTEL	Hotel_name	VARCHAR(50)	NOT NULL
DESITINATION	Dest_ID	INT	PRIMARY KEY
DESTINATION	DCountry_Name	VARCHAR(100)	NOT NULL
DESTINATION	DCity_Name	VARCHAR(50)	NOT NULL
DESTINATION	Dest_Reviews	VARCHAR(300)	

Logical Database Design



Functional Requirements

Consolidating all this data into five primary tables will help our engineering team and data analytics team immensely. By migrating our data from the old databases to the new one, we can have all our data under one roof. Our Analytics team will now be able to query one database, increasing our efficiency, better organizing our data, increasing visibility, and decreasing complexity. Concerning our engineering team, this will eliminate the need to maintain five database connections to our application. By removing the redundancy, we better secure our data, eliminate vulnerabilities, minimize our cost, reduce our infrastructure, and create a single point of failure, which will help troubleshoot issues.

In summary, by creating a single database to store our data, we make life easier for everyone in the company. Having five databases is extremely hard to maintain and cuts into our resources and is time-intensive to maintain. By migrating our five entities into one database, we can create relationships between our data so that we can see much more clearly the connections between flights and destinations, hotels, and users, and much more.

Application Program Design

Function 1: Create New User Account

1. **If Admin == True**
2. **CREATE user_account into USER_ACCOUNT table**
    a. **User_Acct_ID**
    b. **Last_Name,**
    c. **DOB,**
    d. **Pay_Type,**
    e. **First_Name,**
    f. **Home_Address,**
    g. **User_Reviews.**

Function 2: Create New Flight

1. **If Admin == True**
2. **CREATE flight into FLIGHT table**
    a. **Flight_ID,**
    b. **Flight_Price,**
    c. **Flight_Review,**
    d. **Travel_Time,**
    e. **Airport.**

Function 3: Create New Destination

1. **If Admin == True**
2. **CREATE destination into DESITINATION table**
    a. **Dest_ID,**
    b. **DCountry_Name,**
    c. **DCity_Name,**
    d. **Dest_Reviews.**

Function 4: Create New Hotel

1. **If Admin == True**
2. **CREATE hotel into HOTEL table**
    a. **Hotel_ID,**
    b. **Hotel_Review,**
    c. **Hotel_Price,**
    d. **HCountry_Name,**
    e. **HCity_Name.**

Function 5: Create New Trip

1. **If Admin == True**
2. **CREATE trip into TRIP table**
    a. **Trip_ID,**
    b. **Dest_ID,**
    c. **Flight_ID,**
    d. **Hotel_ID,**
    e. **User_Acct_ID.**

Installation Instructions

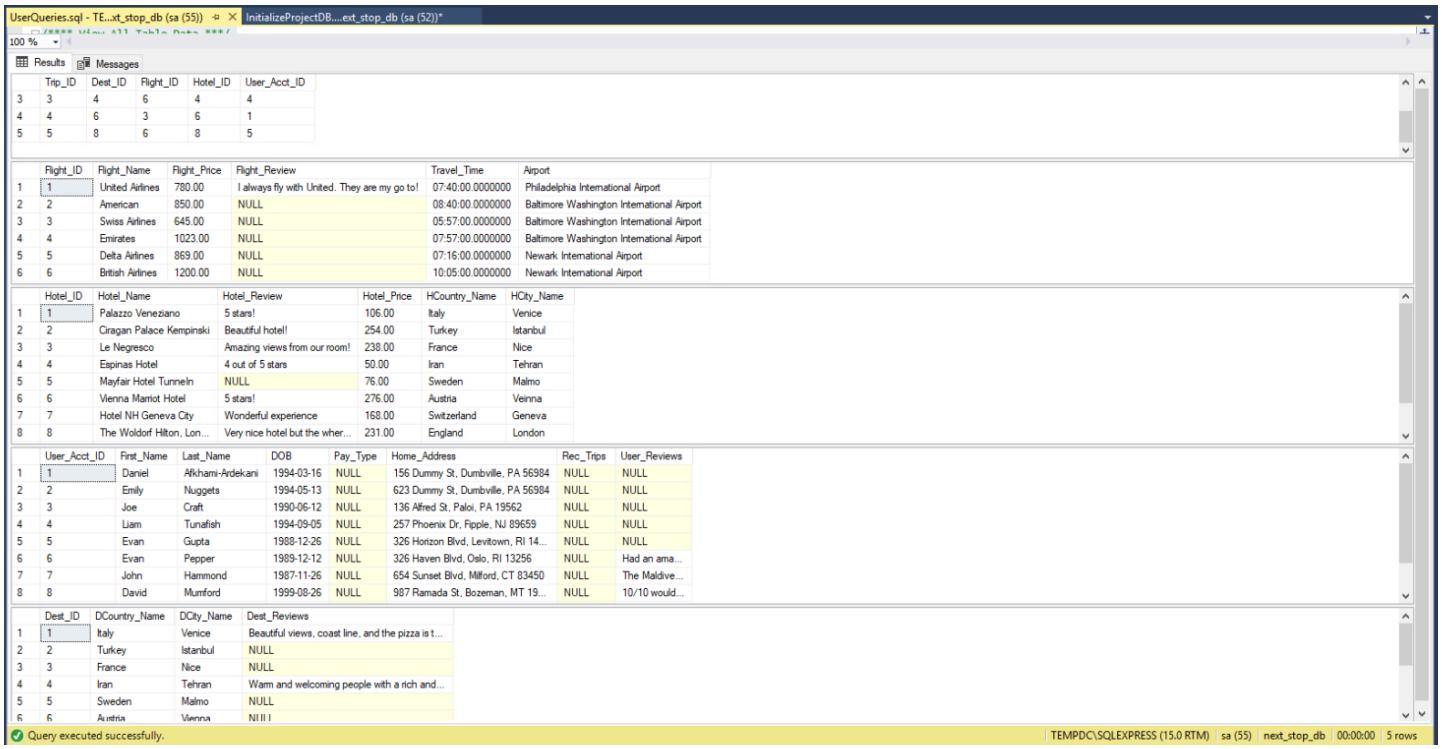
Software and Operating System Prerequisites

1. Windows machine running Windows 10 Home or Pro.
2. Microsoft SQL Server Express 2019
3. SQL Server Management Studio

Install Steps

1. Unzip the AfkhamiArdekani_Daniel_source.zip file and save it to file path of your choosing.
2. Open SQL Server Management Studio.
    a. Click on File > Open > File, and you should see three files: InitializeProjectDB, UserExamples, and UserQueries. Select and open all three.
3. First, you will execute InitializeProjectDB by clicking “Execute”, which is slightly upper left on the screen. This is going to create your tables and populate your tables with test data.
4. The remaining two files will be used throughout the user guide to simplify your experience.

Once you have executed the InitializeProjectDB script you will see the following (Slightly condensed to fit the screen).



User Guide

Inserting a New User Account

The first thing a new user does when they visit the app is create a user account. Before you create a new account let's first look at the existing User_Account table, so that you can see a few examples. Launch SQL Server Management Studio, open the three SQL files I provided, and then run the query below (located in UserQueries.sql).

```
SELECT TOP (100) [User_Acct_ID]
      ,[First_Name]
      ,[Last_Name]
      ,[DOB]
      ,[Pay_Type]
      ,[Home_Address]
      ,[User_Reviews]
  FROM [next_stop].[dbo].[User_Account]
```

If you run the command successfully you will be provided with a list of all the current users as shown below.

	User_Acct_ID	First_Name	Last_Name	DOB	Pay_Type	Home_Address	Rec_Trips	User_Reviews
1	1	Daniel	Afkhami-Ardekani	1994-03-16	NULL	156 Dummy St, Dumbville, PA 56984	NULL	NULL
2	2	Emily	Nuggets	1994-05-13	NULL	623 Dummy St, Dumbville, PA 56984	NULL	NULL
3	3	Joe	Craft	1990-06-12	NULL	136 Alfred St, Paloi, PA 19562	NULL	NULL
4	4	Liam	Tunafish	1994-09-05	NULL	257 Phoenix Dr, Fipple, NJ 89659	NULL	NULL
5	5	Evan	Gupta	1988-12-26	NULL	326 Horizon Blvd, Levitown, RI 14589	NULL	NULL
6	6	Evan	Pepper	1989-12-12	NULL	326 Haven Blvd, Oslo, RI 13256	NULL	Had an amazing time in Italy! The water was crystal clear and the fresh fish was to di...
7	7	John	Hammond	1987-11-26	NULL	654 Sunset Blvd, Milford, CT 83450	NULL	The Maldives were unbelievable. I hope I can get back there soon!!
8	8	David	Mumford	1999-08-26	NULL	987 Ramada St, Bozeman, MT 19456	NULL	10/10 would recommend visiting Banff National Park. Beautiful!!
9	9	Nick	DeCampino	1985-12-10	NULL	145 S. Rose St, Baltimore, MD 22421	NULL	Duck North Carolina never fails.. OBX is the spot for me and my family. Amazing time!
10	10	Frank	Trump	1965-12-10	NULL	52 North Shore Dr, Coralla, NC 23595	NULL	Ah, nothing in OBX is like it was back in my day. The 80s was the time to come! No ...

From the table shown above we can see that there are 10 active user accounts in the User_Account table. If wanted to add a new user, for example, we can run the following SQL code below to add the 11th user to the table (located in UserExamples.sql).

```
-- Creating New User Account --

USE next_stop_db
INSERT INTO User_Account(
    User_Acct_ID, First_Name, Last_Name, DOB, Home_Address, User_Reviews
)
VALUES(
    '11', 'Peggy', 'Jackson', '06/08/1978', '87 Fairfield Dr, Brickville, TX 12354',
    'Had an amazing time in Geneva with my husband. The view of the Swiss Alps was breath taking!'
);
```

We can validate that the SQL code worked by running following command, again and reviewing the results (located in UserQueries.sql).

```
SELECT TOP (100) [User_Acct_ID]
      ,[First_Name]
      ,[Last_Name]
      ,[DOB]
      ,[Pay_Type]
      ,[Home_Address]
      ,[User_Reviews]
  FROM [next_stop].[dbo].[User_Account]
```

Below we can see that we have successfully created our new user.

	User_Acct_ID	First_Name	Last_Name	DOB	Pay_Type	Home_Address	Rec_Trips	User_Reviews
1	1	Daniel	Afkhami-Ardekani	1994-03-16	NULL	156 Dummy St, Dumbville, PA 56984	NULL	NULL
2	2	Emily	Nuggets	1994-05-13	NULL	623 Dummy St, Dumbville, PA 56984	NULL	NULL
3	3	Joe	Craft	1990-06-12	NULL	136 Alfred St, Paloi, PA 19562	NULL	NULL
4	4	Liam	Tunafish	1994-09-05	NULL	257 Phoenix Dr, Fipple, NJ 89659	NULL	NULL
5	5	Evan	Gupta	1988-12-26	NULL	326 Horizon Blvd, Levitown, RI 14589	NULL	NULL
6	6	Evan	Pepper	1989-12-12	NULL	326 Haven Blvd, Oslo, RI 13256	NULL	Had an amazing time in Italy! The water was crystal c...
7	7	John	Hammond	1987-11-26	NULL	654 Sunset Blvd, Milford, CT 83450	NULL	The Maldives were unbelievable. I hope I can get ba...
8	8	David	Mumford	1999-08-26	NULL	987 Ramada St, Bozeman, MT 19456	NULL	10/10 would recommend visiting Banff National Park...
9	9	Nick	DeCampino	1985-12-10	NULL	145 S. Rose St, Baltimore, MD 22421	NULL	Duck North Carolina never fails.. OBX is the spot for ...
10	10	Frank	Trump	1965-12-10	NULL	52 North Shore Dr, Coralla, NC 23595	NULL	Ah, nothing in OBX is like it was back in my day. The...
11	11	Peggy	Jackson	1978-06-08	NULL	87 Fairfield Dr, Brickville, TX 12354	NULL	Had an amazing time in Geneva with my husband. T...

Create a Trip

One of the most exciting part about this app is the ability to create a trip. In the future a user will be able to search for flights and book directly through the web app app. In this next section we will mimic creating a trip. In NextStop a trip consists of a destination, flight, hotel, and user account.

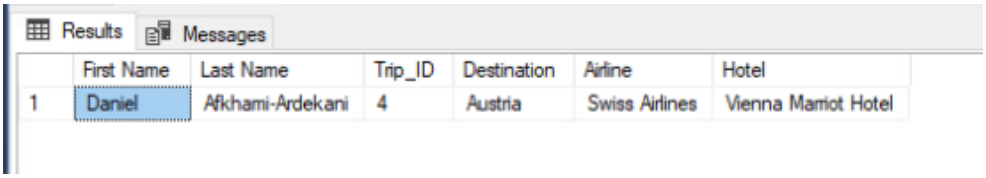
Run the following SQL code to insert a trip (located in UserExamples.sql).

```
-- Trip 4 --
INSERT INTO trip(Trip_ID, User_Acct_ID, Hotel_ID, Flight_ID, Dest_ID)
VALUES (04,(SELECT User_Acct_ID FROM User_Account WHERE Last_Name = 'Afkhami-Ardekani' )
      ,(SELECT Hotel_ID FROM hotel WHERE Hotel_Name = 'Vienna Marriot Hotel')
      ,(SELECT Flight_ID FROM flight WHERE Flight_Name = 'Swiss Airlines')
      ,(SELECT Dest_ID FROM destination WHERE DCountry_Name= 'Austria'));
```

The successful execution of this SQL code will result in a new trip entry for Daniel Afkhami-Ardekani. We can now prove this by running the SQL query shown below. For the query to run correctly we must make sure we select the right trip ID for the WHERE clause, which, in this case is 4 and can be found in the last row of the query (located in UserQueries.sql).

```sql
  -- View Trip Entries
USE next_stop_db
SELECT
    User_Account.First_Name AS 'First Name',
    User_Account. Last_Name AS 'Last Name',
    trip.Trip_ID,
    destination.DCountry_Name AS 'Destination',
    flight.Flight_Name AS 'Airline',
    hotel.Hotel_Name AS 'Hotel'
FROM trip
    INNER JOIN destination ON trip.Dest_ID = destination.Dest_ID
    INNER JOIN User_Account ON trip.User_Acct_ID = User_Account.User_Acct_ID
    LEFT OUTER JOIN flight on flight.Flight_ID = trip.Flight_ID
    LEFT OUTER JOIN hotel on hotel.Hotel_ID = trip.Hotel_ID
    WHERE trip.Trip_ID = 4
```

After you successfully run the SQL query you will get the following return.



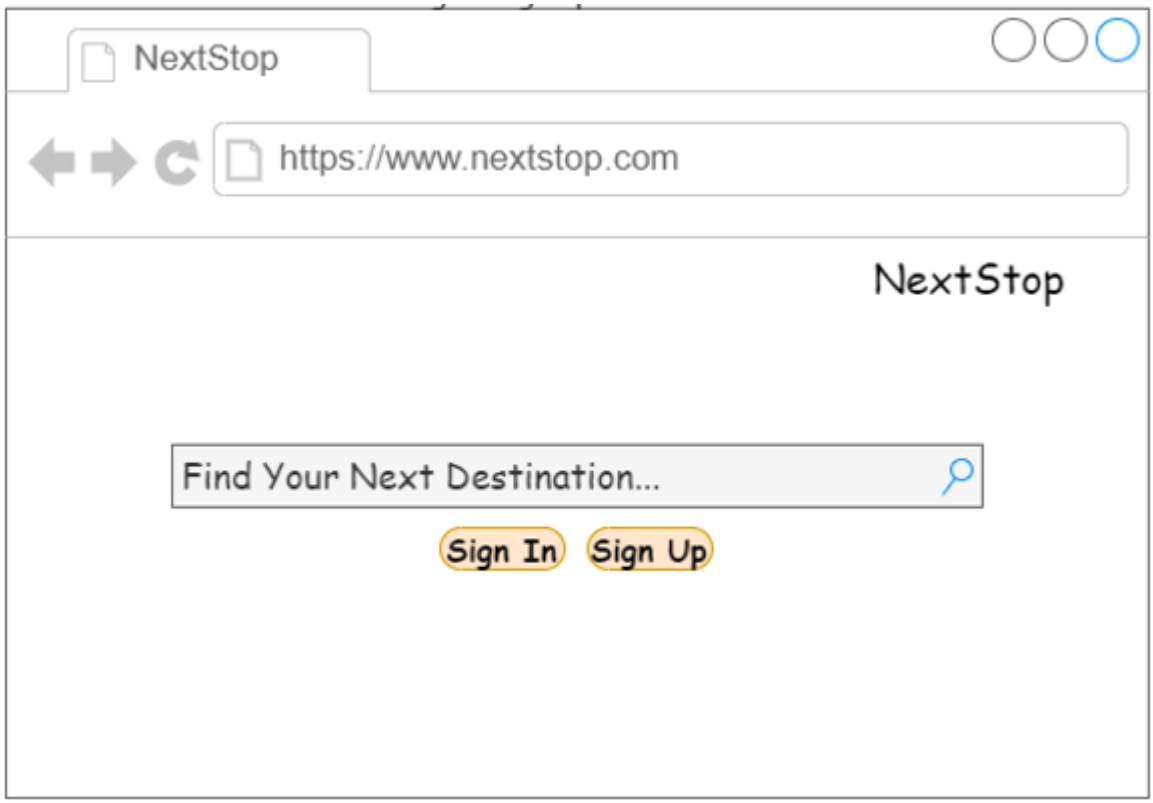
Results | Messages

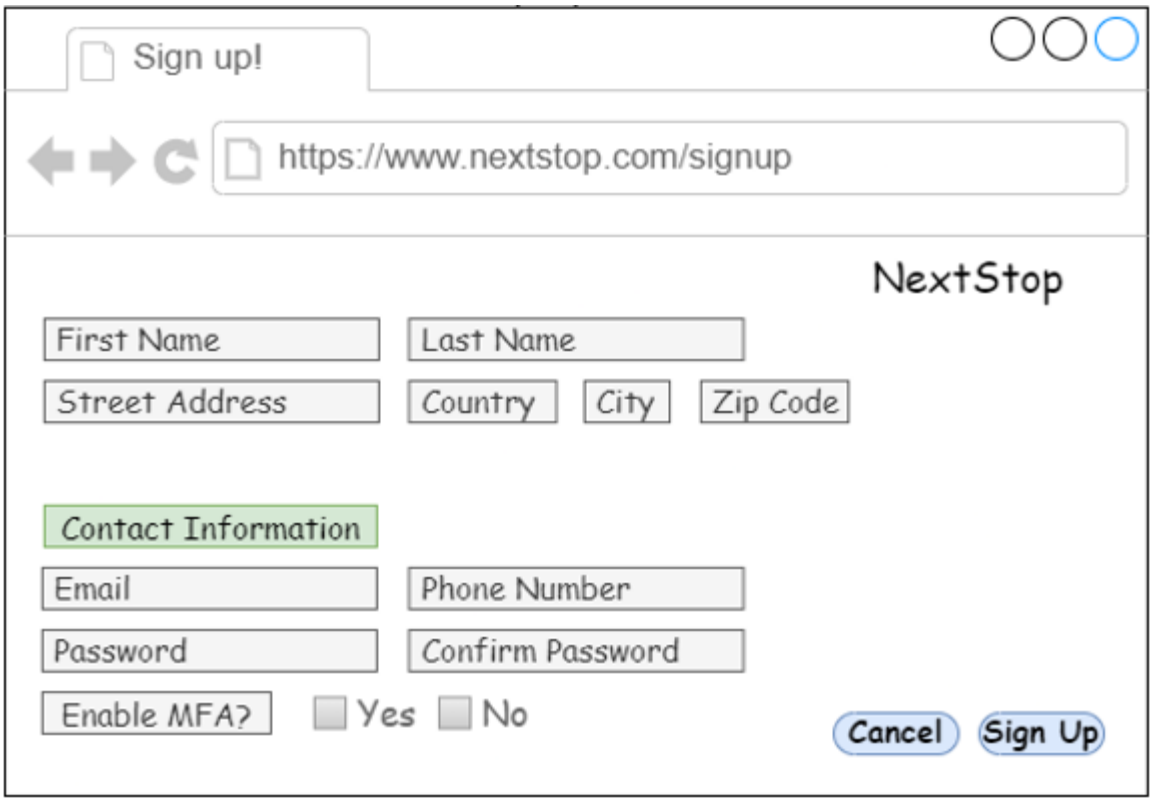
	First Name	Last Name	Trip_ID	Destination	Airline	Hotel
1	Daniel	Afkhami-Ardekani	4	Austria	Swiss Airlines	Vienna Marriot Hotel

Future Web Application Design

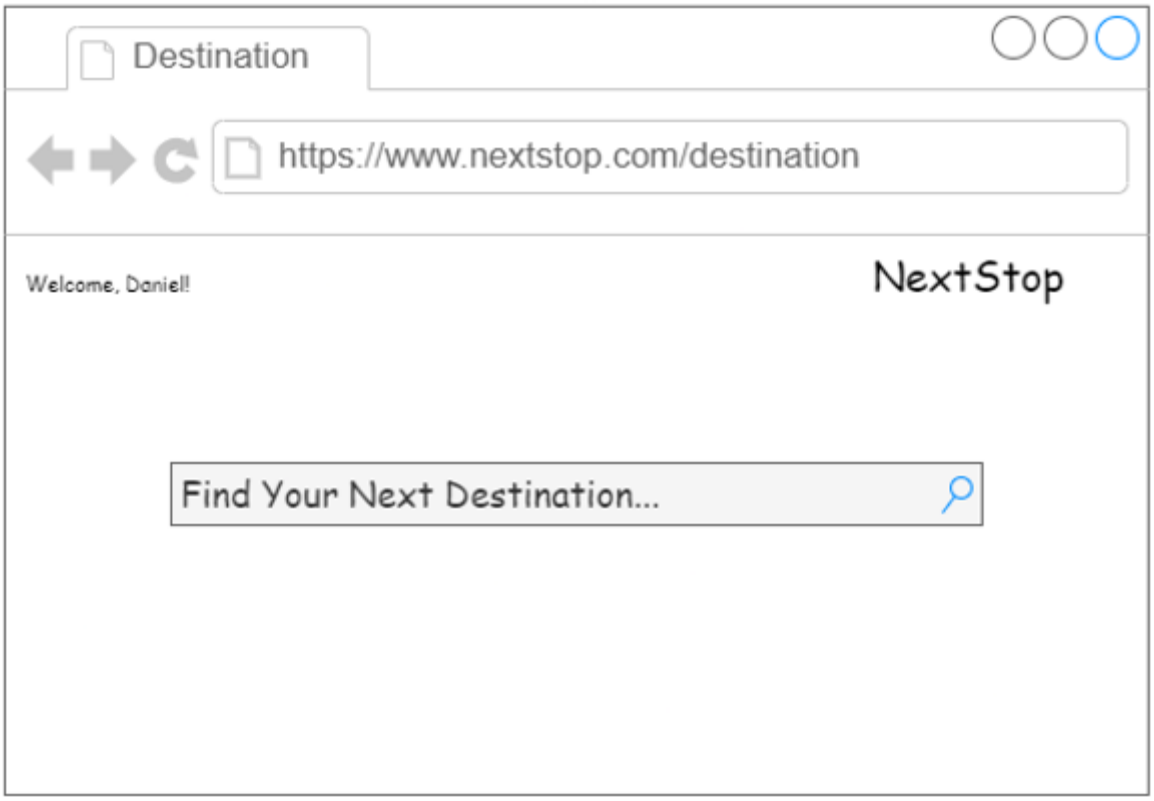
In the future iteration of this project, I will be creating a front-end web application using react js, python, and APIs from booking.com. Booking.com provides APIs for searching and booking hotels and flights. This will be necessary to provide users with the best rates and flexibility. Once the user has chosen their destination and booked their flight and hotel, that information will be stored in our in-house Microsoft SQL server. To accomplish this, I will use python libraries to map the data from the web application to the appropriate tables and columns in the database.

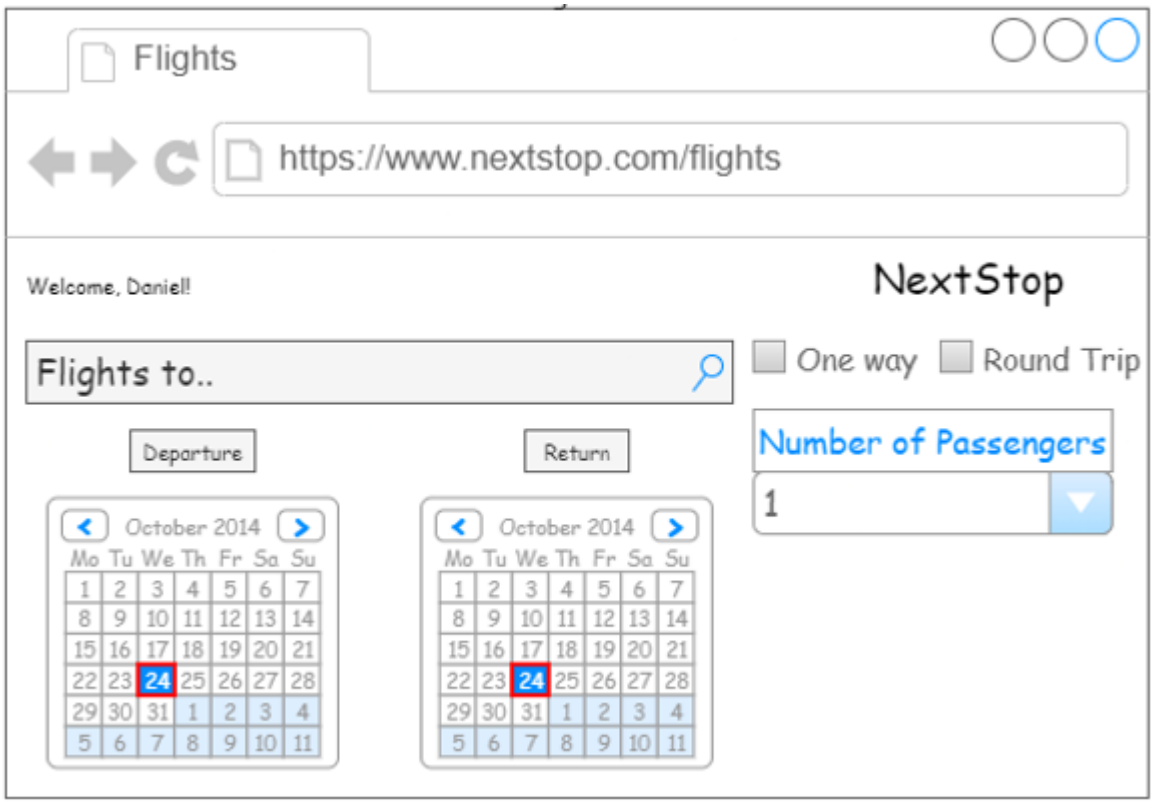
Web Application Wireframe

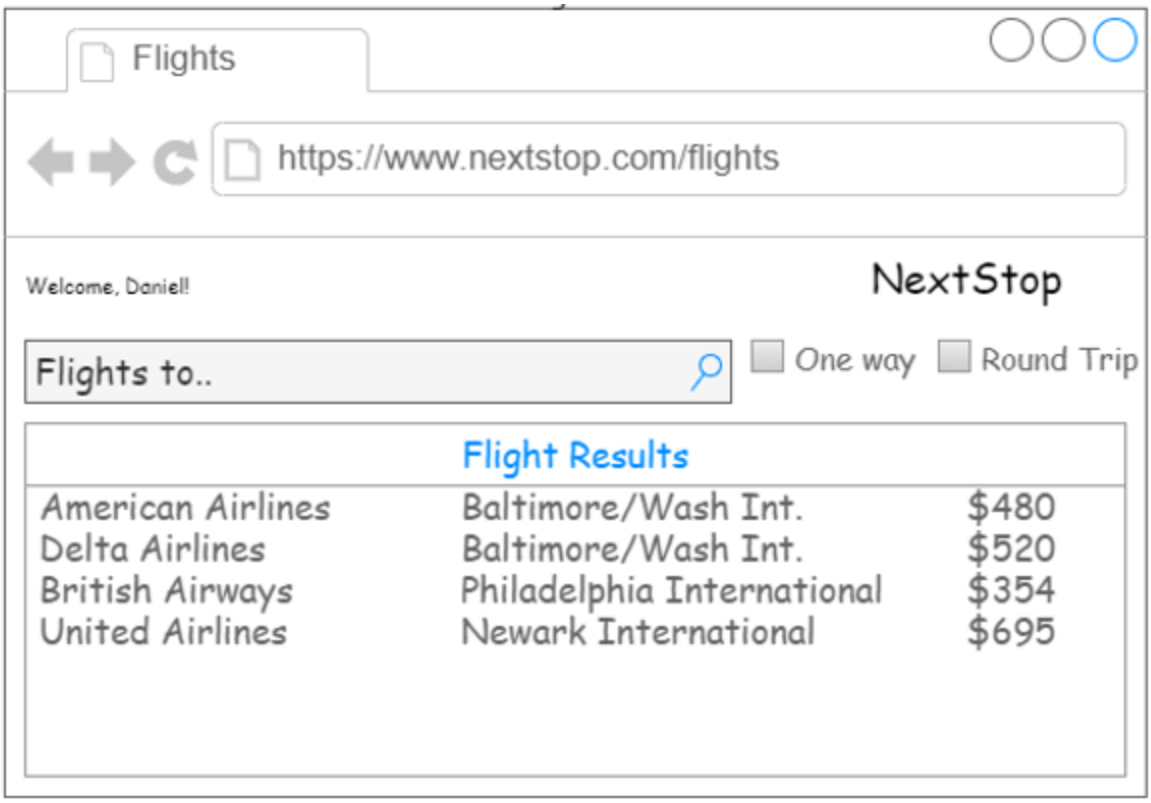


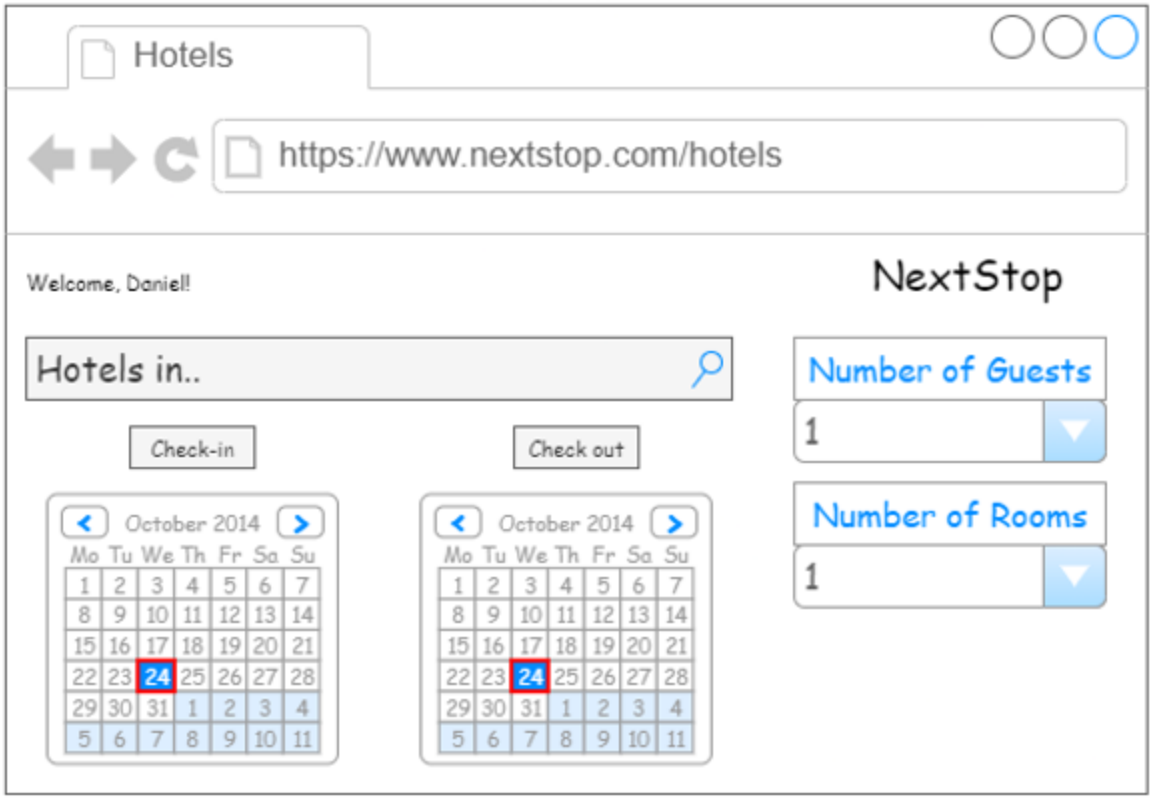

Sign up!
https://www.nextstop.com/signup
NextStop
First Name
Last Name
Street Address
Country
City
Zip Code
Contact Information
Email
Phone Number
Password
Confirm Password
Enable MFA?
Yes
No
Cancel
Sign Up

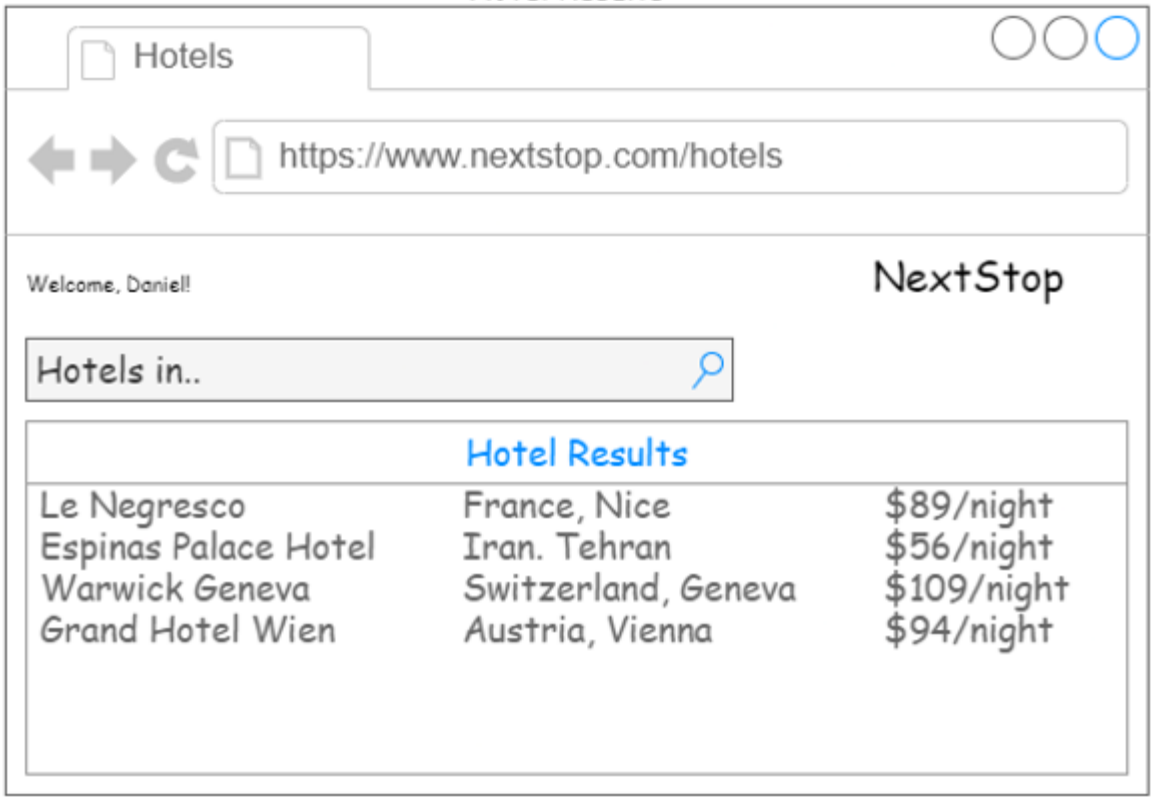

Sign in
https://www.nextstop.com/signin
NextStop
Email
Password
MFA Code
Sign In

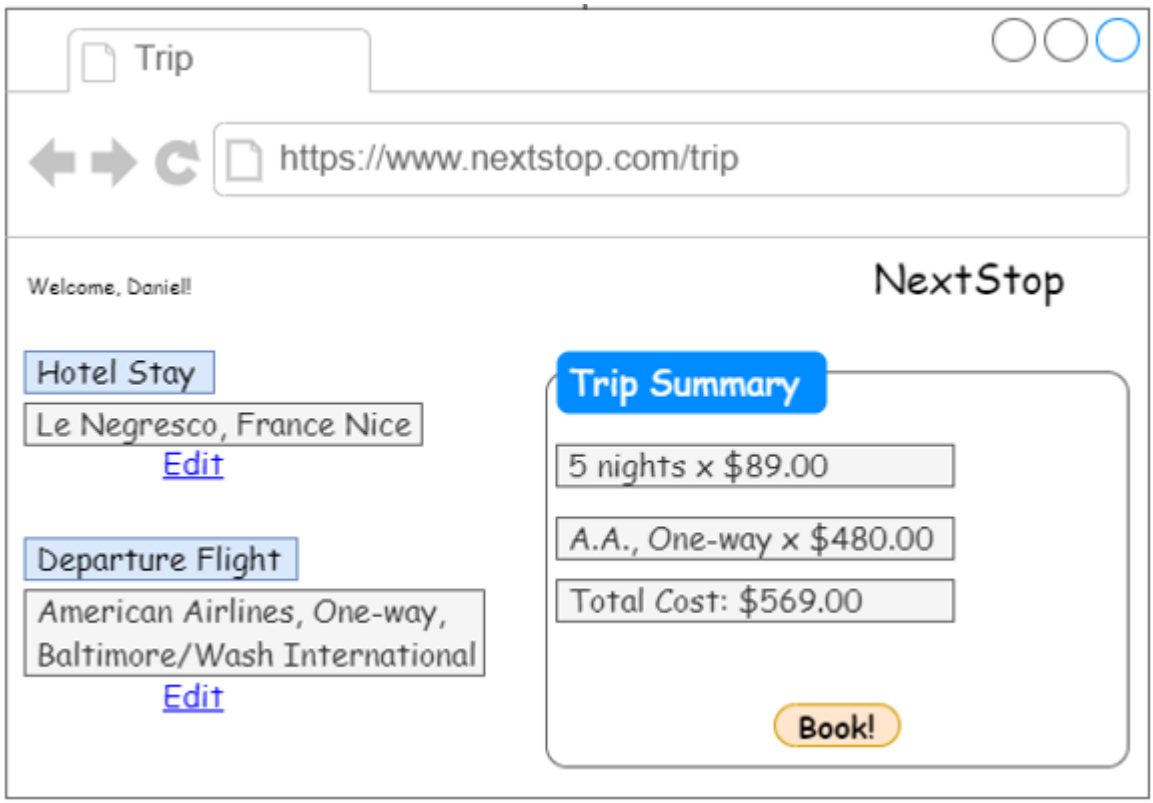

Destination
https://www.nextstop.com/destination
Welcome, Daniel!
NextStop
Find Your Next Destination...


Flights
https://www.nextstop.com/flights
Welcome, Daniel!
NextStop
Flights to..
One way
Round Trip
Departure
Return
Number of Passengers
1
October 2014
Mo Tu We Th Fr Sa Su
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 1 2 3 4
5 6 7 8 9 10 11
October 2014
Mo Tu We Th Fr Sa Su
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 1 2 3 4
5 6 7 8 9 10 11


Flights
https://www.nextstop.com/flights
Welcome, Daniel!
NextStop
Flights to..
One way
Round Trip
Flight Results
American Airlines Baltimore/Wash Int. $480
Delta Airlines Baltimore/Wash Int. $520
British Airways Philadelphia International $354
United Airlines Newark International $695


Hotels
https://www.nextstop.com/hotels
Welcome, Daniel!
NextStop
Hotels in..
Check-in
Check out
October 2014
Mo Tu We Th Fr Sa Su
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 1 2 3 4
5 6 7 8 9 10 11
Number of Guests
1
Number of Rooms
1


Hotels
https://www.nextstop.com/hotels
Welcome, Daniel!
NextStop
Hotels in..
Hotel Results
Le Negresco
France, Nice
$89/night
Espinas Palace Hotel
Iran. Tehran
$56/night
Warwick Geneva
Switzerland, Geneva
$109/night
Grand Hotel Wien
Austria, Vienna
$94/night


Trip
https://www.nextstop.com/trip
Welcome, Daniel!
NextStop
Hotel Stay
Le Negresco, France Nice
Edit
Departure Flight
American Airlines, One-way,
Baltimore/Wash International
Edit
Trip Summary
5 nights x $89.00
A.A., One-way x $480.00
Total Cost: $569.00
Book!