Trip Planner

A website for the user to add and manage their upcoming trips.

# Objectives

The objective of the final project is to create a trip planner organisation system using C# Entity Framework. The website will be using ASP.NET MVC 5. The aim of the website is to make it simple for the user to manage and or create upcoming trips.

# Project Description

The aim of the website is to make it easier for the user to organise their upcoming trips in one place. The user can have multiple trips, with multiple flights and other categories. When a flight for example is created the Trip Id is saved.

The website allows the user to add multiple trips. Under a trip, the user can add flights, car rentals, other transportation, lodgings, activities and restaurant reservations to the trip. Each of these categories can have multiple items added to it. On the details page of the trip, the user can view their flights etc. according to date. The user can edit, and update each of the categories. If the user is wanting to delete an item, the website takes the user to the delete page and confirms if they want to delete the item or not.

The user can add a human under a flight or one of the other categories under the trip. To do this the user needs to go to the details of for example the flight. Then the user can click the link add human. By clicking this link, the Id of the flight is saved and is used when adding the human.

The website will look attractive and well labelled, so that the user can easily use it. The website has been made so that it is easy for them to add, edit, and delete a trip, flights etc.

Under each trip in the details page, the days that the user is away is displayed. Under each day, the flights etc. are displayed if the date is the same as the flight date for example. Under the flight, if the user has added any humans, they will be displayed here.

This project is using Entity Framework in C# using ASP.NET MVC 5. I am using SQLite for the database backend. The project will follow the CRUD system.

When I started the project I started with two database tables: Trip and Flight. Once I got these two tables working well, I added the Human Table. At first I called it Passenger, but after thinking about it, I decided to call it Human as then could be used with more than just the Flight table. After the three tables were working, I added the following tables: Lodging, CarRental, OtherTransportation, ActivityTask and Restaurant.

Each table has a primary key Id. The following had a foreign key of TripId: Flight, Lodging, CarRental, OtherTransportation, ActivityTask, Restaurant and Human. The Human has a foreign key for each of the tables apart from Trip. The foreign key helps to connect two tables together and means I can retrieve data from another table.

# Tables

|  |  |
| --- | --- |
| **Table Name** | **Foreign Key(s)** |
| Trip |  |
| Flight | TripId |
| Lodging | TripId |
| CarRental | TripId |
| OtherTransportation | TripId |
| ActivityTask | TripId |
| Restaurant | TripId |
| Human | TripId, FlightId, LodgingId, CarRentalId,OtherTransportationId,ActivityTaskId, RestaurantId |

# Flow Diagram

