

# **Small World Travel Agency**

CS315 Project

Austin's group

Collaborators: Austin Davis

**Project purpose/outline:**























The purpose of this project is to simulate a fake company called Small World Travel Agency, that has developed a web application where users can visit multiple different HTML pages that represent popular tourist destination cities. Each page contains information on where the city is located, what its main attractions are, how to navigate around the cities, and what weather you can expect. There is also an interest form that uses JS to simulate a user submitting a form to the company that will allow them to select where they want to travel, as well as some of their own information.

**Files:**

- HTML files for each city (7 total cities)
- Index.html is the landing page
- Interest.html
- About.html
- Assets folder
  - Contains all images used for cities + logo
- Styling sheets
  - One primary styling sheet
  - One styling sheet for medium sized screens
  - One styling sheet for small sized screens

Screenshots followed by explanation:

✓ CS315PROJECT

- ✓ Assets
  -  Athens.jpg
  -  Dubai.jpg
  -  globe.png
  -  Nyc.jpg
  -  Paris.jpg
  -  Rio.jpg
  -  Rome.jpg
  -  SmallWorldLogo.png
  -  Tokyo.jpg
-  about.html
-  athens.html
-  dubai.html
-  index.html
-  interest.html
-  nyc.html
-  paris.html
-  rio.html
-  rome.html
-  styles.css
-  stylesMed.css
-  stylesSmall.css
-  tokyo.html

```
.nav {  
  display: flex;  
  width: 100%;  
  justify-content: center;  
  padding: 10px;  
  align-items: center;  
  background-color: #20569e;  
}  
  
.nav-item {  
  padding-left: 20px;  
  padding-right: 20px;  
  padding-top: 10px;  
  font-weight: bold;  
  border-radius: 10px;  
  font-family: 'Franklin Gothic Medium', 'A  
  font-size: 25px;  
  text-align: center;  
  color: white;  
}  
  
.nav-list {  
  display: flex;  
  justify-content: space-between;  
}
```

```
.globe {  
  width: 100px;  
  height: 100px;  
  background: url('./Assets/globe.png') no-repeat;  
  background-size: cover;  
  margin: 10px;  
  animation: rotate 4s linear infinite;  
  display: flex;  
  justify-content: center;  
  transition: width 2s, height 2s;  
}  
  
.globe:hover {  
  width: 200px;  
  height: 200px;  
}  
  
.globe-container {  
  width: 100%;  
  height: max-content;  
  display: flex;  
  justify-content: center;  
  align-items: center;  
}  
  
@keyframes rotate {  
  100% {  
    transform: rotate(360deg);  
  }  
}
```

```
script>
  //Select the travelForm and listen for when use submits it
  document.getElementById('travelForm').addEventListener('submit',
  event.preventDefault();

  //Select all the fields the user can fill out and grab the value
  var destination = document.getElementById('destination').value;
  var name = document.getElementById('name').value;
  var email = document.getElementById('email').value;
  var message = document.getElementById('message').value;

  //Build a JS object based on input field data
  var formData = {
    destination: destination,
    name: name,
    email: email,
    message: message
  };

  //Turn Object into JSON string using JSON.stringify()
  var jsonData = JSON.stringify(formData);
  console.log('JSON Data:', jsonData);
);
/script>
```

```
<form id="travelForm">
  <label for="destination">Select a Destination:</label>
  <select id="destination" name="destination">
    <option value="Athens">Athens</option>
    <option value="Dubai">Dubai</option>
    <option value="New York City">New York City</option>
    <option value="Paris">Paris</option>
    <option value="Rio de Janeiro">Rio de Janeiro</option>
    <option value="Rome">Rome</option>
    <option value="Tokyo">Tokyo</option>
  </select>

  <label for="name">Name:</label>
  <input type="text" id="name" name="name" required>

  <label for="email">Email:</label>
  <input type="email" id="email" name="email" required>

  <label for="message">Message:</label>
  <textarea id="message" name="message" rows="4"></textarea>

  <button type="submit">Submit</button>
</form>
```



```

<table class="team-table">
  <tr>
    <th>Name</th>
    <th>Role</th>
    <th>Experience</th>
  </tr>
  <tr>
    <td>John Doe</td>
    <td>Founder</td>
    <td>10 years</td>
  </tr>
  <tr>
    <td>Jane Smith</td>
    <td>Travel Consultant</td>
    <td>8 years</td>
  </tr>
  <tr>
    <td>Mike Johnson</td>
    <td>Customer Support</td>
    <td>6 years</td>
  </tr>
</table>

```

```

<link rel="stylesheet" href="./styles.css"/>
<link rel="stylesheet" media="screen and (max-width: 768px)" href="stylesMed.css">
<link rel="stylesheet" media="screen and (max-width: 480px)" href="stylesSmall.css">

```

As you can see, extensive CSS was used to obtain the desired result for the web page. Three files are used: one default file, one medium screen file, and one small screen file. In the stylesheet rel of each html page, we have made a media query to determine the size of the screen and thus select a CSS file based on that information.

Layout was used extensively with flexbox layout in multiple components (Such as the navbar) to make the HTML layout flexible and dynamic.

A form was used for the interest form, which also takes the information using JavaScript and converts it into JSON data. Comments are made within the <script> tag so you can see what operations are being done.

Tables/menus were used on the about us page when making a list of the employees/founders of the company.

Animation and transition were both used within the globe in the index.html file, which is the home page.

Github Link: [austin-davis1/Cs315Project: A WIP Project for CS315 \(Internet Programming\) at Truman State \(github.com\)](https://github.com/austin-davis1/Cs315Project)