State Finder

State Finder Team: Alyiah Proctor, Emily Crabtree, Anne Nguyen, Darlyn Mendez

Purpose

- Tool that will help aid people with career and location decisions
- Aid people who are looking to move to another state,
- Allow college students to want to see the locations that their future career could lead them to
- Help people who are looking for a career change
- Show anyone that is curious where an occupation could allow them to live in the United States.

Overview and Goals

- Interactive web app
- Shows livable metropolitan areas based on users income and family size
- Users will have a range of occupations and household indicators to select from
- State infographics with other cost data found throughout development
- States vs Counties vs Metro Areas
- Areas are recommended based on livable wage data and Occupational employment and wage statistics

How It Will Be Achieved

- General Structure of Our Application
  - Flask - Python Web Framework
  - Possible Python Libraries - Pandas, Matplotlib, Flask Login
  - Frontend - HTML, CSS, JavaScript, Bootstrap
  - Backend - Microsoft Azure SQL Database
  - Deploy to Heroku
- Continuous Team Member Collaboration
  - Bi-weekly Scrum meetings
  - Communication on Discord server

How Will It Be Achieved - State Finder Overall Structure
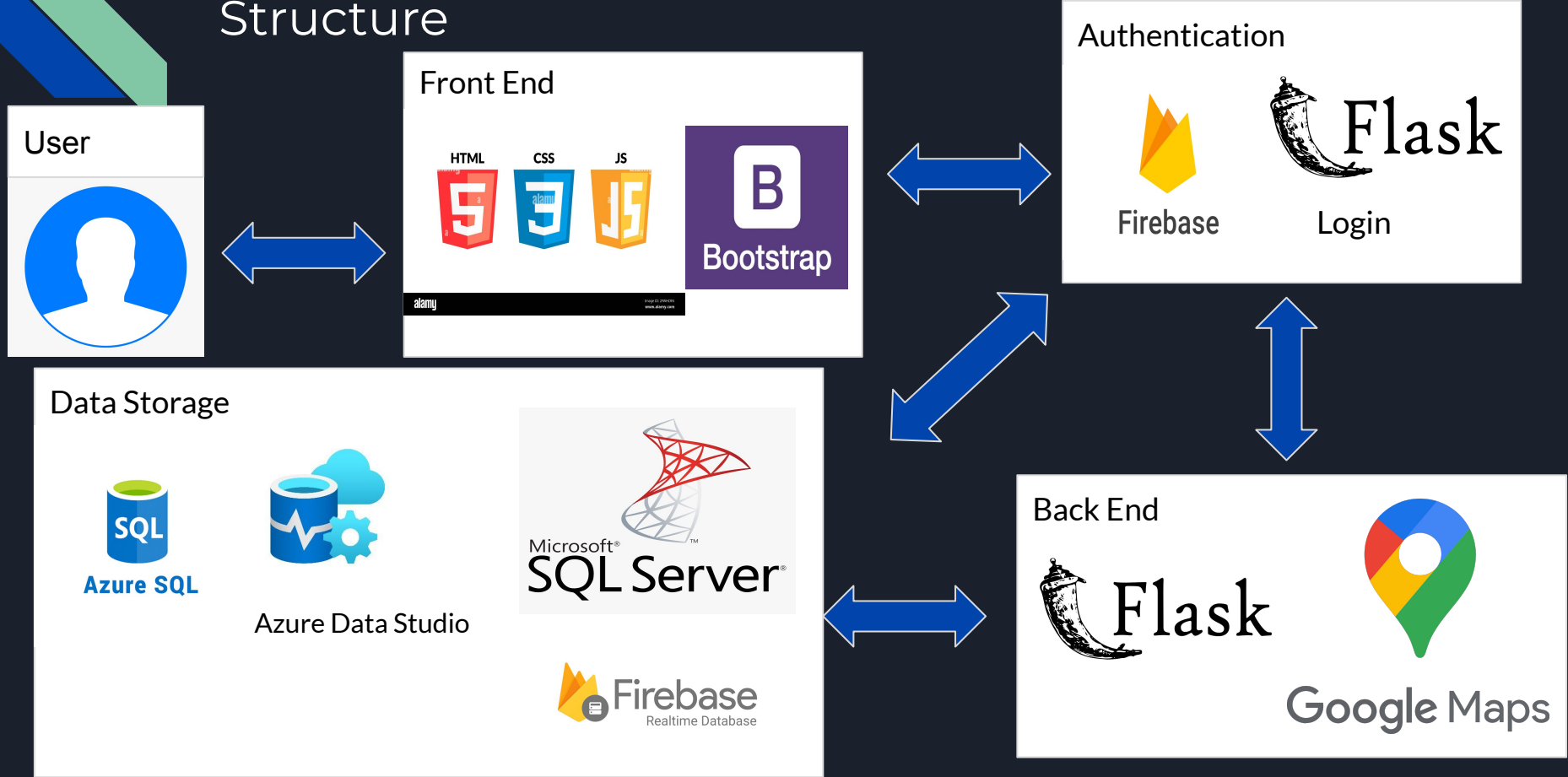
Project Specification - Frameworks

- Flask framework
  - create web server
  - Ensures everything on the client side functions properly
- Azure SQL Server Database
  - Make changes in Azure Data Studio
  - ER Diagram
  - Schema Diagram
  - All Team Members added as collaborators.
- Bootstrap - front-end development framework
  - Makes the UI visually appealing
  - Increases the web app's load time
  - Compatible with different browsers

Project Specification - Libraries

- Pandas for data manipulation
- Plotly for graphs
  - Create dashboard
- Matplotlib for graphs
  - Create interactive visualizations
- Python requests
  - Use Python requests with REST APIs
  - Extract data from API to use in our database

Project Specification - Development Environment

- Visual Studio Code
- Azure Data Studio
- Google Cloud Console Setup
- Firebase Console Setup
- Configuration files for connections and deployment

Functional Requirements and Usability

- "Find A State" page
  - Occupation, job level (entry, intermediate, senior)
  - Marital status, is spouse employed?
  - Number of kids
- Production of heat map
  - Based on user input
  - Color change indicates livability
  - State details on hover or below
- Usability
  - All user data on "Find A State" page is collected via dropdown menus
  - Reduces chance of input error

System Requirements

- Hardware
  - CPU
  - Monitor/display
  - Keyboard and mouse
- Software
  - Operating System
  - Graphics card
  - Web Browser
- Database
  - No requirement for user
  - Internet connection

Security

- Secure login portal
  - Unique username
  - Password
  - Unique ID will act as primary key, connect to login credentials
- Checks so cannot access user data
- Dropdown menus do not allow access to database