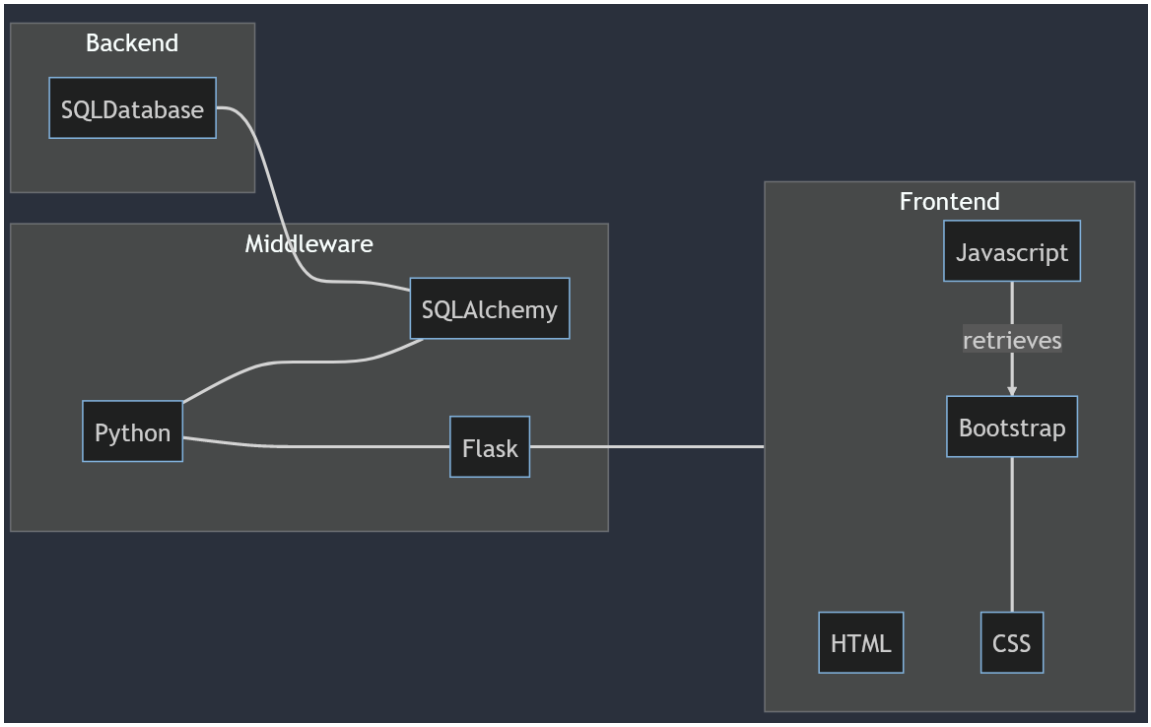
System Architecture

Our application is a web application which would mean that we would have to use web technologies, such as HTML, javascript, etc, to build it. We chose Python as the language which binds the front-end and back-end together as it is a simple but powerful language. Moreover, Flask is used as the web framework, SQLAlchemy is used to communicate with our database and Bootstrap is the CSS framework.

Backend = Entity, Middleware = Controller, Frontend = Boundary



Here is a description of how the system would work when a user opens the page and finds a carpark, to demonstrate how the system would work.

1. User opens the page:
    - The user accesses the website by entering the website URL in their web browser.
    - The user sends a request to the server.
    - The request is transferred to the Flask framework, and it retrieves the appropriate HTML, CSS and Javascript templates, and construct them based on the current context (e.g. If the user is in Ang Mo Kio, Flask will modify the current location text to “Ang Mo Kio”. If it is nighttime, Flask will modify the CSS so that the UI is dark-themed.)
    - The server then sends the modified HTML, CSS, and Javascript files required to display the website's landing page.
    - The user’s browser then retrieves the Bootstrap CSS Framework content - which further improves the styling of the landing page.
2. User finds a carpark:

- The landing page displays a search bar that allows the user to search for a car park by entering the location, date, and time they require.
- Once the user enters the required details and clicks on the search button, a request is sent to the server.
- The request is transferred and transformed accordingly from the server → Flask →Python → SQLAlchemy. The server retrieves the relevant data from the database using SQLAlchemy.
- The server sends back the search results to the client in the form of an HTML page, displaying the available car parks along with their prices, location, and other details.