Application design:



**Filter Section**: A filter section allows users to select and customize the date range (e.g., "June-July"). This includes navigation arrows for easy date range adjustment and a manual selection option for precise customization.

**Main Dashboard**: The primary focus of the dashboard is a histogram titled "Case Active Days At Closure," providing a visual representation of data related to active case durations.

**Secondary Dashboards**: Below the main section, there are two smaller dashboards. One presents a pie chart titled "Closed on Time" and the other showcases a stacked bar chart titled "Reason for Closure," offering different insights into closed cases.

**Final Dashboard**: At the bottom, there's a dashboard displaying a boxplot titled "Active Days Distribution," potentially offering a different perspective on case durations.

**Navigation Icons**: The bottom of the page includes icons for easy navigation—home, profile, and about/more.

In terms of design choices and implications:

**User-Friendly Navigation**: The use of intuitive icons for settings, date range selection, and navigation enhances user experience and accessibility.

**Data Visualization Hierarchy**: The layout emphasizes the main histogram, followed by supporting visualizations, ensuring the primary focus is on the most critical data while offering supplementary insights.

**Dashboard Composition**: The combination of different chart types (histogram, pie chart, stacked bar chart, and boxplot) enables varied data representation, catering to different user preferences or analytical needs.

**Interactivity**: Consider adding interactive elements to the charts (e.g., tooltips, click functionality) to enhance user engagement and data exploration capabilities.

**Scalability**: Ensure the design accommodates scalability, allowing for additional visualizations or functionalities in the future without cluttering the dashboard.

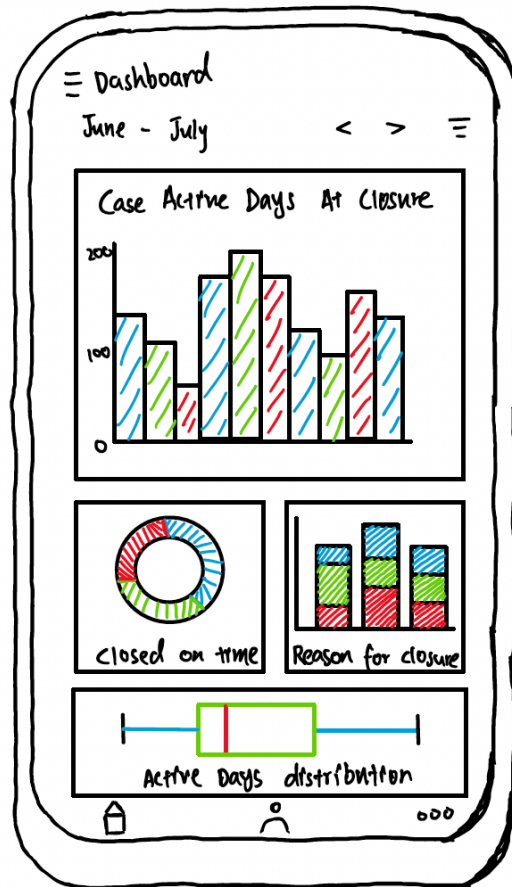**Title and Settings Icon**: The top section features a title "Dashboard" and three horizontal lines for settings, ensuring easy navigation and access to customization options.

Backend design:

**Data retrieval**

Responsibilities: Fetches data from the metropolitan Police Service dataset.

Components:
- Data retrieval functions: responsible for fetching data via API calls or database queries.
- Data processing helpers: Assists in initial data cleaning.

**Data processing**

Responsibilities: Handles data manipulation and pre-processing tasks.

Components:
- Data Transformation Functions: Cleans, filters, and transforms raw data into usable formats.
- Data Aggregation Tools: Groups and aggregates data for visualization purposes.

**Data visualisation**

Responsibilities: Focuses on generating visual representations of data.

Components:
- Chart Generation Functions: Creates different types of charts (e.g., bar charts, line plots) based on processed data.
- Interactive Elements: Adds functionalities for user interactions with visualizations (e.g., filtering, zooming).

**API layer**

Responsibilities: Exposes endpoints for frontend interaction.

Components:
- API Endpoints: Provides access to specific data subsets or visualizations for the frontend.
- Data Serialization: Converts data to appropriate formats (e.g., JSON) for frontend consumption.