Functional Requirements

1. File Upload and Processing

- The system shall allow users to upload data files in **CSV, JSON, and Excel formats**.
- The system shall parse and validate uploaded files to ensure they are correctly formatted.
- The system shall provide an error message if an unsupported file type is uploaded.

2. Data Visualization

- The system shall generate visualizations in **bar charts, line graphs, scatter plots, and pie charts**.
- The system shall allow users to select different chart types based on their dataset.
- The system shall support interactive features such as **zooming, filtering, and hover tooltips**.

3. Data Transformation and Customization

- The system shall allow users to **sort, filter, and group** data before visualization.
- The system shall provide options to customize charts, including **color selection, axis labels, and legends**.
- The system shall enable users to download or export the generated visualizations as **PNG, SVG, or PDF**.

4. User Interface (UI) and Accessibility

- The system shall provide a **responsive and user-friendly** web interface.
- The system shall ensure compatibility with major browsers such as **Chrome, Firefox, Edge, and Safari**.
- The system shall provide **tooltips and guidelines** to help users navigate and use the tool effectively.

5. Performance and Security

- The system shall process datasets up to **10,000 rows** without significant lag.
- The system shall ensure that uploaded files are **not stored permanently** for security reasons.
- The system shall prevent malicious file uploads by validating file content.

6. Deployment and Hosting

- The system shall be accessible as a **web-based application** hosted on **GitHub Pages, Vercel, or Netlify**.
- The system shall not require user authentication or account creation.

Non-Functional Requirements for Data Visualiser

1. Performance Requirements

- The system shall load and render **visualizations within 2 seconds** for datasets of up to **10,000 rows**.
- The system shall support real-time interaction (e.g., zooming, filtering) without noticeable lag.

2. Usability Requirements

- The user interface shall be **intuitive and easy to navigate**, requiring minimal learning effort.
- The system shall provide **tooltips, labels, and error messages** to guide users.
- The system shall support **keyboard and mouse interactions** for accessibility.

3. Compatibility and Portability

- The system shall be **browser-independent** and work on **Chrome, Firefox, Edge, and Safari**.
- The system shall be responsive and work on **both desktop and mobile devices**.
- The system shall not require additional software installation.

4. Security Requirements

- The system shall not store uploaded files permanently, ensuring **user data privacy**.
- The system shall prevent the execution of **malicious scripts** within uploaded files.
- The system shall use HTTPS for secure communication.

5. Scalability and Maintainability

- The system shall be designed to allow easy **future expansion**, such as adding more visualization types.
- The system shall follow **modular coding practices** to support maintenance and updates.

6. Availability and Reliability

- The system shall have **99% uptime** when deployed on hosting services like GitHub Pages, Vercel, or Netlify.
- The system shall handle multiple users simultaneously without significant performance degradation.