**6. Solution Requirements**

This section outlines the key **technical and functional requirements** necessary to implement the *College Food Choices Visualization System*. The goal is to ensure a robust, scalable, and user-friendly analytics platform that effectively communicates dietary patterns and trends among students.

**6.1 Functional Requirements**

| **ID** | **Requirement** |
| --- | --- |
| FR-01 | The system must load and preprocess dietary CSV data for visualization |
| FR-02 | The Tableau dashboard must support interactive filtering (e.g., gender, GPA) |
| FR-03 | Users should be able to view visual summaries of calorie intake, habits, etc. |
| FR-04 | Dashboards must be embedded in a Flask web interface for external access |
| FR-05 | The system must support storytelling features (multi-scene dashboard navigation) |
| FR-06 | Users should not need to log in or authenticate to access dashboards |

**6.2 Non-Functional Requirements**

| **ID** | **Requirement** |
| --- | --- |
| NFR-01 | The system should be responsive and render across desktop and mobile browsers |
| NFR-02 | The dashboard must load within 3–5 seconds under normal data load conditions |
| NFR-03 | The embedded dashboard must maintain secure access without exposing raw data |
| NFR-04 | Visuals should maintain consistency in color, layout, and label clarity |
| NFR-05 | The system must support scalability for future datasets (e.g., semester updates) |

**6.3 Platform & Tool Requirements**

| **Component** | **Requirement** |
| --- | --- |
| **Data Format** | CSV (.csv) |
| **Visualization Tool** | Tableau Desktop & Tableau Public/Server |
| **Web Framework** | Python Flask |
| **Hosting** | Render / Vercel (for Flask deployment, optional) |
| **Hardware** | Minimum 8 GB RAM, internet access, modern browser |
| **Optional Tools** | Tableau Prep (for advanced data cleaning) |

**6.4 Key Assumptions**

* Users will access dashboards passively (no input required).
* The dataset is static and doesn’t require real-time data syncing.
* Flask is used only for UI presentation and not backend processing.
* Visualizations are publicly shareable and do not contain sensitive PII.