**OmniStock Inventory Tracker**

**CSC154 Group 7**

**1. Functional Requirements**

* **User Authentication (MUST)**: Users must be able to register, log in, and manage access credentials.
* **Inventory Management (MUST)**: CRUD operations must be available for inventory items.
* **Stock Alerts (SHOULD)**: The system should notify users when stock levels fall below a predefined threshold.
* **Data Import/Export (COULD)**: Users could import/export inventory data in CSV format.
* **Role-Based Access Control (WON’T in initial release)**: Different levels of permissions for administrators and standard users.

**2. Non-Functional Requirements**

* **Performance (MUST)**: The system must handle up to 1000 concurrent users with minimal latency.
* **Security (MUST)**: Implements encrypted password storage and follows OWASP security guidelines.
* **Scalability (SHOULD)**: The system should scale to support increased data loads without performance degradation.
* **Usability (COULD)**: UI could be optimized for accessibility and responsiveness across multiple devices.
* **Availability (WON’T in initial release)**: The system will aim for 99.9% uptime but won't guarantee it initially.

**3. Implementation Requirements**

* **Technology Stack:**
* Frontend: React.js
* Backend: Node.js with Express
* Database: PostgreSQL
* Authentication: OAuth 2.0 / JWT
* **Version Control:** GitHub for source control and CI/CD pipelines.
* **Project Management:** Jira for task tracking and sprint planning.
* **Testing Strategy:** Unit testing using Jest, integration testing using Cypress.

**4. Technical Constraints**

* **Database Storage:** PostgreSQL database must be hosted on AWS RDS.
* **Deployment Platform:** System will be deployed using Docker containers.
* **Response Time:** API responses must not exceed 200ms for standard queries.
* **Cross-Platform Compatibility:** Must function seamlessly on Windows, macOS, and Linux.
* **Third-Party Integrations:** API compatibility with external inventory management tools.

**5. Assumptions & Dependencies**

* Users must have internet access to use the application.
* Integration with third-party tools requires stable APIs.
* Development will follow Agile principles and Scrum methodology.