**Document 1: Definition of Done:**

Definition of Done (DOD) checklist for my project “Implementation of front running alert in ATS system is given below.

**Produced Code for Presumed Functionalities:**

The code developed should fulfill the functionalities outlined in the user stories.

All intended features and scenarios should be implemented as described.

**Assumptions of User Story Met:**

All assumptions and conditions specified in the user stories should be validated and met.

User stories should be reviewed to ensure that all implicit and explicit assumptions are addressed.

**Project Builds Without Errors:**

The project should build successfully in the development environment.  
There should not be any compilation or build errors.

**Unit Tests Written and Passing:**

Unit tests should be created for all new features and functionalities.

All unit tests should pass successfully, ensuring the code behaves as expected.

**Project Deployed on the Test Environment Identical to Production Platform:**

The application which is deployed to a test environment should be the replication of the production setup.

Deployment scripts and procedures should be tested and validated.

**Tests on Devices/Browsers Listed in the Project Assumptions Passed:**

The application should be tested on all specified devices and browsers.

Compatibility and performance tests should be conducted to ensure cross-platform functionality.

**Feature OK-ed by UX Designer:**

The feature should be reviewed and approved by the UX designer.

Design and usability criteria should met, ensuring a positive user experience.

**QA Performed & Issues Resolved:**

Quality Assurance (QA) testing should be completed, including functional, regression, and performance tests.

All identified issues and bugs should be resolved before marking the user story as done.

**Feature Tested Against Acceptance Criteria:**

The feature should be tested thoroughly against the predefined acceptance criteria.

Acceptance tests should be documented and results should be reviewed with the Product Owner.

**Feature OK-ed by Product Owner:**

The Product Owner should review and approve the feature.

Any feedback or changes requested by the Product Owner should be addressed.

**Refactoring Completed:**

Code is refactored to improve readability, maintainability, and performance.

Refactoring should not introduce new bugs or regressions.

**Any Configuration or Build Changes Documented:**

All configuration changes and build processes should be documented.

Documentation should be updated to reflect the current state of the project.

**Documentation Updated:**

User manuals, technical documentation, and help guides should be updated to include new features and changes.

Documentation should be reviewed for accuracy and completeness.

**Peer Code Review Performed:**

Code changes should be peer-reviewed by other developers.

Feedback from code reviews should be incorporated, and any issues raised should be addressed.

**Document 2- Product Vision:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Scrum Project Name** | Implementation of Front Running Alert in ATS | | |
| **Venue** | Meeting Room at Client Location | | |
| **Date** | **Start Time:** | **End Time:** | **Duration:** |
| **Client** | CBA | | |
| **Stakeholder List** | Business Team | Compliance Team of Bank | |
| Trade Desk | |
| Bank Users | |
| Bank Technology Team | Abinitio Team | |
| Scrum Team | Product Owner | |
| Scrum Master | |
| Business Analyst | |
| 2 Kdb Developers | |
| 2 Testers | |
| **Scrum Team** | | | |
| **Scrum Master** | Julian Jacobs | | |
| **Product Owner** | Nikhil Bhatia | | |
| **Kdb Developer 1** | Jaganmayee Sahoo | | |
| **Kdb Developer 2** | Amit Kumar | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Vision** | **To establish a robust and efficient Front Running Alert mechanism within the Automated Trade Surveillance (ATS) system that ensures proactive detection of potential market abuse, safeguards market integrity, and enhances compliance with regulatory standards by leveraging advanced analytics, real-time data processing, and intuitive dashboards.** | | |
| **Market Segment** Financial institutions in institutional banking and capital markets, focusing on regulatory compliance and trade surveillance and Organizations in regulated markets requiring automated monitoring solutions to prevent market manipulation.   **Target Users** **Primary Users:** Compliance Officers, Trade Surveillance Teams, and Risk Management Professionals at CBA. **Secondary Users:** Regulatory Auditors and IT/Data Teams supporting the ATS system. | **Needs:** -Detects and prevents front running, a type of market manipulation, by analyzing order and trade data in real-time -Addresses gaps in existing trade surveillance systems by providing automated alerts for suspicious activities. -Reduces compliance risks by ensuring adherence to regulatory standards.   **Benefits Provided:** **Enhanced Market Integrity:** Builds trust by identifying and mitigating potential market abuse. **Improved Efficiency:** Automates the detection process, reducing manual efforts and investigation time. **Regulatory Compliance:** Ensures CBA meets local and international trade surveillance requirements. **User-Friendly Insights:** Offers intuitive dashboards for quick and effective alert review and decision-making. | **Product:** The Front Running Alert is a feature integrated into CBA's Automated Trade Surveillance (ATS) system. It is designed to detect and flag potential front running activities by analyzing order and trade data in real-time, ensuring proactive monitoring and compliance.    **What Makes It Desirable and Special?** **Proactive Detection:** Identifies suspicious trading patterns before they escalate into significant risks. **Data-Driven Insights:** Leverages high-performance Kdb and an intuitive Kx dashboard for actionable analytics. **Customization:** Offers configurable thresholds and filters to meet specific regulatory and business requirements. **Streamlined Investigations:** Provides a user-friendly interface with features like date selection for quick alert analysis.   **Is It Feasible to Develop the Product?** **Existing Infrastructure:** Leveraging CBA's ATS system and established data pipelines like MODS and Abinitio. **Technical Capability:** High-performance Kdb and Kx dashboard ensure scalability and real-time processing. **Vendor Expertise:** Luxoft's experience in developing surveillance systems and implementing regulatory solutions for financial institutions. | **How Is the Product Going to Benefit the Company?** -Ensures regulatory compliance and reduces penalties. -Builds trust by enhancing market integrity. -Automates monitoring, cutting costs and manual efforts. -Mitigates financial and reputational risks.    **What Are the Business Goals?** -Ensure regulatory compliance. -Reduce risks of market manipulation. -Improve monitoring efficiency and accuracy. -Strengthen client trust through robust surveillance.    **What Is the Business Model?** -Use existing ATS infrastructure for cost efficiency. -Avoid fines through compliance-driven ROI. -Provide a scalable system for future needs. -Regularly optimize based on market and regulatory changes. |

**Document 3: User stories:**