**PROJECT PROPOSAL**

**Transport Management System (TMS) for ITL Logistics Group**

**PART 1:** **Quality Management**

Quality management is crucial in the software development process, especially when adopting the Scrum framework. Scrum emphasizes iterative and incremental development, with frequent releases and feedback loops. In this dynamic environment, effective quality management ensures that the software meets the desired standards and satisfies customer expectations. Recognizing this importance, our team choose to adapt ISO 25010 framework for quality management.

**Chosen characteristic**: Reliability.

**Characteristic and sub-characteristics details:**

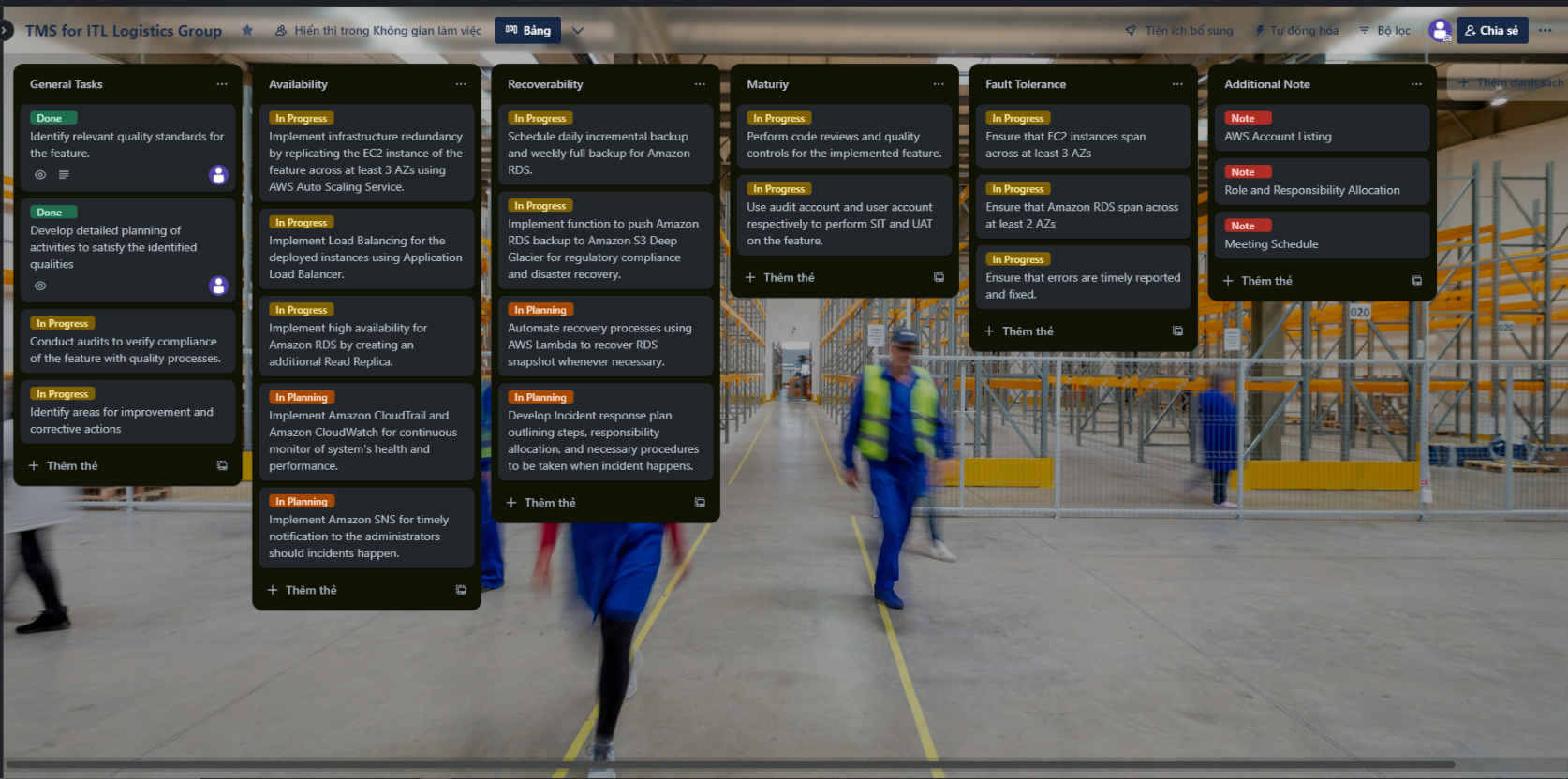
|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Characteristics** | **Sub-Characteristics** | **Definition of Done criteria** |
| 1 | Reliability | Availability | The feature should be accessible and responsive, downtime and unavailability must be significantly minimized to ensure 99.999% of availability. |
| 2 | Recoverability | At least 80% of the system’s main functionalities must be able to operates normally after recovery from failure. |
| 3 | Maturity | The module functions must undergo SIT and UAT to operate reliably and consistently, and various scheduling scenarios must be tested to handle unexpected issues or failures. |
| 4 | Fault Tolerance | The feature's database needs to be replicated across a minimum of two regions. Additionally, scaling on demand should be implemented to handle any potential failures in the application server. |

**Metrics and Threshold for sub-characteristics**

|  |  |  |  |
| --- | --- | --- | --- |
| **Sub-Characteristics** | **Metric** | **Threshold** | **Rationale** |
| Availability | Uptime | 99.999% | The feature should be operational and accessible for 99.999% of uptime period, experiencing minimal downtime. |
| Recoverability | Mean Time to Recover | < 1 hour | The average time take to restore the system to normal operation after a failure should be under 1 hour. |
| Maturity | Defect density | < 1.5 defects/KLOC | The number of defects identified per thousand lines of code (KLOC) should be under 1.5. |
| Fault Tolerance | Mean Time Between Failure | 500 hours | The average time between two consecutive failures should be under 500 hours |

**PART 2:** **Detailed planning of activities**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Sub-Characteristics** | **Definition of Done criteria** | **Status** |
| 1 | General Tasks | Identify relevant quality standards for the feature. | Done |
| 2 | Develop detailed planning of activities to satisfy the identified qualities | Done |
| 3 | Conduct audits to verify compliance of the feature with quality processes. | In Progress |
| 4 | Identify areas for improvement and corrective actions | In Progress |
| 5 | Availability | Implement infrastructure redundancy by replicating the EC2 instance of the feature across at least 3 AZs using AWS Auto Scaling Service. | In Progress |
| 6 | Implement Load Balancing for the deployed instances using Application Load Balancer. | In Progress |
| 7 | Implement high availability for Amazon RDS by creating an additional Read Replica. | In Progress |
| 8 | Implement Amazon CloudTrail and Amazon CloudWatch for continuous monitor of system’s health and performance. | In Planning |
| 9 | Implement Amazon SNS for timely notification to the administrators should incidents happen. | In Planning |
| 10 | Recoverability | Schedule daily incremental backup and weekly full backup for Amazon RDS. | In Progress |
| 11 | Implement function to push Amazon RDS backup to Amazon S3 Deep Glacier for regulatory compliance and disaster recovery. | In Progress |
| 12 | Automate recovery processes using AWS Lambda to recover RDS snapshot whenever necessary. | In Planning |
| 13 | Develop Incident response plan outlining steps, responsibility allocation, and necessary procedures to be taken when incident happens. | In Planning |
| 14 | Maturity | Perform code reviews and quality controls for the implemented feature. | In Progress |
| 15 | Use audit account and user account respectively to perform SIT and UAT on the feature. | In Progress |
| 16 | Fault Tolerance | Ensure that EC2 instances span across at least 3 AZs | In Progress |
| 17 | Ensure that Amazon RDS span across at least 2 AZs | In Progress |
| 18 | Ensure that errors are timely reported and fixed. | In Progress |

**Trello Dashboard**

*Figure 1: Trello Dashboard for Sprint 1*

**Rationale**

The project has made significant progress in achieving the desired qualities for the chosen feature. The identification of relevant quality standards and the development of a detailed plan have been completed. Audits to verify compliance and the identification of areas for improvement are currently in progress.

For availability, the project is implementing infrastructure redundancy and load balancing across multiple availability zones. High availability for Amazon RDS is being addressed by creating an additional Read Replica. Continuous monitoring of system health and performance is planned using Amazon CloudTrail and Amazon CloudWatch, along with timely notifications through Amazon SNS.

To ensure recoverability, daily incremental backups and weekly full backups for Amazon RDS are being scheduled. The project is also working on pushing RDS backups to Amazon S3 Deep Glacier for regulatory compliance and implementing automated recovery processes using AWS Lambda.

For maturity, code reviewing quality controls are underway, and system integration testing (SIT) and user acceptance testing (UAT) are being planned.

To achieve fault tolerance, efforts are being made to ensure that EC2 instances span across multiple availability zones, and Amazon RDS spans across at least two availability zones. Timely error reporting and fixing are also being addressed.

Overall, the project is taking comprehensive measures to meet the desired qualities of the feature, with several activities completed and others in progress.