CS5213 Group B Ticket 9

SPRINT 1 EXECUTION

Sprint Execution Overview

Capstone Management System Introduction:

- Web-based platform to manage capstone projects.
- Streamlines project management for students, faculty, and administrators.
- Facilitates efficient project tracking, submission, and assessment.

Current Sprint Objectives:

- Build a solid foundation through infrastructure setup.
- Prepare a test environment aligned with development needs.
- Implement initial system functionalities in a Django-based environment.
- Configure production and development settings for optimal operation.
- Begin integration of essential services, including the CI/CD pipeline.
- Ensure the system is scalable, secure, and user-friendly for future expansions.

Team Coordination and Roles

Scrum Masters:

- Lead sprint planning and facilitate in tracking progress.
- Address project impediments and maintain development momentum.

Developers:

- Evaluate and select appropriate technologies.
- Align technical setup with the project's architectural and functional requirements.

Quality Assurance (QA) and Testing:

- Set up automated testing environments for continuous software integrity checks.

Project Managers (PMs):

- Oversee project strategy, aligning activities with goals and timelines.
- Facilitate team communication, resource management, and progress tracking.

Milestone Progress and Issues

Milestone M1 (Project Setup) by Group A

Issues by scrum masters:

- Development Environment Configuration
- Django project initialization
- Production Environment Configuration - Security Implementation
- Production Environment Configuration - Web/Database Selection
- Production Environment Configuration - CI/CD Pipeline Plan

Comments by project managers:

- Divide work into specific, individual issues or tasks, focusing on distinct types of work.
- Replace "deliverables" with a "definition of done" to clearly state what successful completion looks like for each task.
- Ensure each team member has individual issues assigned to them, avoiding the situation where the entire team works on a single issue.

Milestone Progress and Issues

Milestone M2 (Production Environment Setup) by Group F

Issues by scrum masters:

- Backup and Recovery Strategy
- Security Measures
- Database Configuration
- Application Deployment
- Server Setup and configuration
- Cloud Service Provider Selection

Comments by project managers:

- Appreciation for the insights provided on the hosting issue.
- Request to explore hosting options with OU and cloud services concurrently for a comprehensive evaluation.

Continuation

- Emphasis on the value of examining both hosting possibilities simultaneously.
- The well-defined main issue should be broken down into smaller, manageable tasks.
- Each smaller task should be assigned to individual team members for focused responsibility.
- Positive recognition of the current issue's "definition of done."
- Suggestion to integrate security measures implementation into Issue #23 to balance the workload.
- Necessity of a clear "definition of done" for each issue to specify the criteria for successful completion.

Milestone Progress and Issues

Milestone M3 (Test Environment Setup) by Group E

Issues by scrum masters:

- Automated testing and continuous integration
- Resource provisioning and auto-scaling
- Infrastructure automation
- Application deployment
- Configuring the Database
- Test Suite Integration
- Selection of IaC tool and configuration management tool

Comments by project managers:

- Automate quality checks and performance metrics in the 'Definition of Done'.
- Integrate these checks into the CI process to validate commits against benchmarks and security standards.

Milestone Progress and Issues

Milestone M4 (ORM Objects Definition Report) by Group D

Issues by scrum masters:

- Database Migrations Completed
- ORM Classes Defined
- Database Schema Creation
- Migration Scripts
- Relationship Mapping
- Python Class Creation
- Data Model Design

Comments by project managers:

- Confirm data model's entity relationships enhance database scalability and adhere to security standards.

Continuation

- Update progress timeline in the issue for tracking against project plan milestones.
- Verify the updated database schema meets development requirements and ORM object definitions from milestone M4.
- Check that Python classes for the data model are committed to version control, well-documented, consistent with design specifications, and dependencies are noted.

Milestone Progress and Issues

Milestone M5 (Automated CI/CD Deployment Report) by Group C

Issues by scrum masters:

- Implement manual triggers for production deployment
- Configure deployment workflow for test environment
- Set up CI workflow for pull requests
- Document CI/CD process and deployment strategies

Comments by project managers:

- Documentation should be concise, mainly at the README level, without extensive writeups.
- Ensure tasks and workflows are well-commented and include references to external resources.
- Incorporate a "definition of done" to clarify the success criteria for each task.

Continuation

- Consider splitting large tasks into smaller, manageable issues for team distribution.
- Positive acknowledgment of the issue's quality, but still, a "definition of done" is needed to specify the completion outcome.

Issue Tracking and Management

In the current sprint of our Capstone Management System, the Issue Tracking and Management process is central to maintaining organization, ensuring accountability, and facilitating communication within our team. Our approach consists of the following key practices:

- **Issue Creation and Categorization:** Each task, whether it be a feature, bug, or improvement, is recorded as an issue in our GitHub repository. These issues are then categorized by type, priority, and associated milestone to ensure clarity and focus.

- **Assignment and Ownership:** Upon creation, issues are assigned to team members based on their expertise and current workload. This fosters a sense of ownership and responsibility towards task completion.

Issue Tracking and Management

- **Workflow Integration:** Our issue tracking workflow follows a structured path—beginning with issues in the 'Backlog', moving to 'In Progress' as they are actively worked on, advancing to 'Ready' when they await peer review, and finally transitioning to 'In Review' for quality checks before closure.

- **Progress Updates:** Team members are encouraged to regularly update the status of issues, comment with progress reports, or raise any blockers that might hinder the task completion. This enables real-time tracking and swift resolution of impediments.

- **Collaborative Review:** Prior to closing an issue, designated code reviewers ensure that all contributions adhere to our project's quality standards. This step includes scrutinizing code changes, documentation, and adherence to the project's style guide.

Issue Tracking and Management

- **Documentation:** Each issue is documented thoroughly to ensure that the rationale, decisions made, and lessons learned are captured for future reference.
- **Continuous Learning:** We actively use retrospectives to reflect on the issues faced and resolved, allowing us to identify improvement opportunities in our tracking and management processes.
- **Integration with Communication Tools:** Our use of Discord and Zoom for communication complements our issue tracking efforts, providing platforms for synchronous discussions and decision-making related to issue resolution.
- **Monitoring and Reporting:** Our Scrum Masters and Project Managers monitor the issue dashboard to provide regular updates on progress and challenges during sprint reviews, ensuring that the project remains on track with its objectives.

Challenges and Solutions

Groups A and F scrum masters raised concerns regarding hosting the application in the cloud (cost) and there was unclear communication with OU CS IT about using a server.

A communication breakdown occurred and the result of the meeting with OU CS IT was not clearly communicated to the project management team.

Communication and Collaboration Tools

Throughout the sprint, communication between the stakeholders, scrum masters, and project managers were done through the following tools

- Discord
- Zoom
- GitHub Issues

Throughout the week the PM team worked with the various scrum masters to answer any questions and concerns that were brought up, as well as working through GitHub issues that were created by the various scrum masters

Continuous Learning and Improvement

Our team (group B, project managers) are learning about project management in a practical firsthand way.

Not being directly responsible for developing code for the application, but rather facilitating and coordinating between development teams requires different skills than typically expected in class projects.

We will try to use the insights for project management gained in the first sprint to improve in the remaining sprints.

Next Steps and Future Planning

Currently we are still working through issues with acquiring server resources for the application, as communication with the CS department continues we hope to gain insight on the resources available to us to host our application

- This will affect which database we use and how we will deploy the test and production environments, opening the work available for M2 and M3

By evaluating the engagement of the various groups we can adjust the assignments of the various milestones which will best aid in the completion of the project

Conclusion and Feedback

The first sprint was rocky and the project is going in fits and starts.

The groups actively contributing to the project are starting to make headway.

With continued communication and the collaboration of the remaining teams who have heretofore been silent, perhaps better progress will be made.

It is unclear whether we will achieve the milestones created for the 4 1-week sprints.