

## Integrative Programming and Technologies PIT Rubrics

Criteria	Excellent (5)	Good (4)	Satisfactory (3)	Needs Improvement (2)	Poor/Not Present (1)
<b>Git Contribution</b>	Regular commits with clear messages; use of branches, pull requests, and collaborative workflow	Frequent commits; some branching or collaboration evident	Infrequent commits; solo work; basic version control	Very few commits; poor version control practice	No Git usage
<b>Platform Integration</b>	Implemented a fully integrated backend server, web application, and mobile application, with seamless data synchronization. The mobile application replicates most of the features and functionality available in the web application.	Implemented a backend server, web application, and mobile application, but the mobile application has limited functionality or lacks some key features present in the web application.	Web and mobile applications are present but are not connected, resulting in isolated data and user experiences.	Backend server and web application are present, with an incomplete or partially connected mobile application, resulting in inconsistent data or functionality gaps.	Only the backend server and web application are present, with no mobile application or clear integration effort.
<b>Security Measures</b>	Authentication and Authorization is Implemented. Strong validation (frontend + backend), hashed passwords, email validation, rate limiting	Hashed passwords and validation; no email validation and authorization	Basic validation and hashed passwords	Weak validation or plaintext passwords	No security practices followed
<b>Web Socket/3<sup>rd</sup> Party API</b>	Implemented Socket.IO for real-time communication or	Implemented a similar real-time communication	Implemented a basic real-time communication	Attempted real-time communication or 3 <sup>rd</sup> party API integration, but	No real-time communication or 3 <sup>rd</sup> party API integration

Criteria	Excellent (5)	Good (4)	Satisfactory (3)	Needs Improvement (2)	Poor/Not Present (1)
	seamless integration with a 3rd party API, demonstrating comprehensive understanding and effective use of event handling and data synchronization.	technology (e.g., WebSockets, WebRTC, or server-sent events) or partially integrated a 3rd party API, with minor limitations in functionality or design.	feature or 3rd party API integration, but with noticeable performance issues, limited functionality, or inconsistent data handling.	the implementation lacks stability, is partially functional, or has significant issues that affect the user experience.	
<b>Docker Setup</b>	Docker is implemented for deployment. Docker image is present, <code>Dockerfile</code> is correctly configured, <code>docker-compose.yml</code> is present with proper service definitions, environment variables, and credentials. <code>.dockerignore</code> is included to exclude unnecessary files, and the entire system runs smoothly in Docker <b>or</b> the system is already deployed on a cloud platform (e.g., AWS, GCP, Azure) with	Docker is implemented with a correctly configured <code>Dockerfile</code> and <code>docker-compose.yml</code> , but some minor improvements are needed for performance or security, such as missing <code>.dockerignore</code> entries or incomplete environment variable configurations.	Docker is implemented, but the setup lacks critical components like a <code>docker-compose.yml</code> file or a properly configured <code>.dockerignore</code> , resulting in a partially functional or inefficient containerized environment.	Attempted to use Docker, but the configuration is incomplete or has significant issues, such as missing <code>Dockerfile</code> or improper service definitions, preventing the system from running reliably in Docker.	No Docker implementation, or a completely broken Docker setup that prevents the system from running.

Criteria	Excellent (5)	Good (4)	Satisfactory (3)	Needs Improvement (2)	Poor/Not Present (1)
	credentials securely managed.				

### Criteria Weight

Git Contribution 1x

Platform Integration 3x

Security Measures 2x

Web Socket/3rd Part API 2x

Docker Setup / Cloud Deployment 2x