Written Submission: Internship Assessment – Touchstone Institute

Authored by Promise Ononokpono

1. SCRUM Roles & Events

Over the course of my journey in Software I have come to understand the impact of implementing SCRUM in software teams. By following this framework teams are able to work more efficiently together to deliver more value and adapt quickly to changes within the project.

Typically, SCRUM teams consist of three core roles:

1. ***Scrum Master***, who facilitates the process and removes obstacles
2. ***Product Owner***, who represents the voice of the users and prioritizes the backlog
3. ***Developers***, who deliver the product increment during each sprint.

As an intern, I would take on the Developer role, eager to contribute meaningfully while learning from the senior developers and mentors, focusing on writing code, testing features, and collaborating with the team to meet sprint goals. I've already experienced this kind of collaboration in Agile environments during my internships at UNB and Global Vision, where I regularly attended SCRUM ceremonies, contributed updates, and participated in team planning. I also have experience assuming the responsibilities of a scrum master and a product owner informally in previous personal and academic projects. I am always ready for a challenge and seeking growth opportunities!

SCRUM Ceremonies & My Contributions:

1. Daily Stand-Ups: I would provide concise updates on my progress, raise any blockers, and stay aligned with team priorities.

2. Sprint Planning: I would help break down tasks into manageable pieces and contribute estimates where possible, asking clarifying questions.

3. Sprint Reviews: I would demo completed features and receive feedback constructively.

4. Sprint Retrospectives: I would share insights on what helped or hindered my productivity, suggest improvements to the process, and celebrate team wins.

5. Backlog Refinement: I would help review user stories with the Product Owner and ask questions to ensure feasibility and testability.

By being engaged in these ceremonies and being proactive in communication, I would support the team not only by contributing to the codebase but by helping keep things on track, consistent, and user-centered.

2. SDLC Planning: Redesigning the Automatic Flagging System

To redesign the automatic flagging system, I would use a structured approach following the SDLC. This would help ensure we build something reliable, maintainable, and genuinely useful to end users which is my priority as the developer. With my previous internship and technical experiences, I understand that each step is crucial to the success of the overall project.

Requirements: I would begin with understanding the current system's strengths and weaknesses by gathering feedback and reviewing the current flagging rules. I would also gather feedback using stakeholder interviews, collaborating to clarify functional expectations such as input structure, flag formats etc. and non-functional requirements such as acceptable response time, UI responsiveness and scalability. I would document the requirements and use cases, and create acceptance criteria for each flagging rule.

Design: The next step would be to draft the data structure specifications as either JASON or in a table format as this would be the most critical part. Next, plan the backend service architecture (using modular rule evaluators) and create frontend mockups and diagrams to map out how the new system would work, ensuring gaps from the previous system are addressed. I would also create wireframes to visualize the user flow and logic of both the frontend(user input) and backend (API payload). This step is a collaborative process with the development team and designers.

Implementation:
Once the design flow and mapping is complete, the coding can begin. In this step, the goal is to write clean, modular code using version control. Coding is an incremental process, I start with a proof of concept and build out features progressively. For the backend implementation would involve using Express and TypeScript, and write reusable functions to implement each rule and return as structure list of flags. On the Frontend, I would use AngularJS and PrimeNG to build a responsive form,

Testing: Next, I would write unit and integration tests, simulate real-world scenarios, and conduct user testing. For backend unit tests will be done using Jest or Mocha, and frontend would use Karma and Jasmine to test AngularJS components. Overall, end-to-end testing is to be done to confirm the correct flow from input to visual output.

Deployment: Then I would use a staging environment with Docker/cloud deployment and monitor post-release performance by setting up API monitoring and error logging.

Maintenance: Although the system is deployed, it is important to then review performance regularly by scheduling periodic testing and updates for dependencies, incorporate user feedback, and keep documentation up to date. The documentation would clearly include the flagging logic and API specifications.

By approaching the flagging system redesign with an SDLC view, I can contribute meaningfully at every step—whether it's gathering insights, helping with implementation, or participating in user testing.

I’m excited about the opportunity to intern with Touchstone Institute, where I can apply these skills to real-world problems while learning in a collaborative and purpose-driven environment. Thank you for your time and consideration.