Jantzen Springer
CS-250: SDLC
Professor McWain
2/22/2024

Week 8 Journal

Throughout this class, one standout aspect of Agile is its emphasis on the pivotal role of users and stakeholders in product development. The genesis of project ideas and the key to crafting successful products lies in customer collaboration. Achieving this requires leveraging interpersonal skills and incorporating effective listening, aspects I aim to carry forward post this class.

As previously mentioned, customers hold paramount importance. Agile and the Scrum cycle adeptly underline this, providing a manageable approach. The user's needs are best addressed through professional interpersonal skills. Understanding the complexities of individuals is crucial to creating detailed user stories, forming the basis for product development by the team.

Once user requirements are identified, maintaining consistency in documentation becomes essential. This optimizes the process, clearly highlighting project requirements. In a Scrum environment, this is achieved through user stories. Each specification in the product backlog is associated with a user story, providing detailed insights into user desires. These user-centric descriptions commence with an action, benefit, and value statement, such as, "As a user, I want to do this thing, so that I can have this benefit." Supporting materials like test cases or user feedback build upon these user stories, ensuring development aligns with desired user quality. Additionally, effective communication is pivotal to Agile success. In case of questions or uncertainties, immediate feedback from the user is sought by the Scrum team, fostering adaptability throughout the process.

My current approach to programming problems still lacks foundational organization. I initially created a flowchart to clearly illustrate ideas and problem-solving steps. My problem resides in transitioning from the flowchart to the development cycle. Pivoting from the flowchart, I approach programming as a narrative, solving small problems incrementally. This incremental approach is necessitated by the inherent complexity of programming. Addressing one problem at a time allows for effective management of the myriad questions that arise along the way.

Approaches from Agile that I find intriguing include problem management and classification based on nature rather than a fixed timeline. While I do solve programming problems one at a time, I acknowledge that I currently classify functionalities at a scale beyond my experience level. Breaking down functionalities into more manageable "user stories" in the context of programming promises increased fluidity and simplicity in my processes.

Success in programming extends beyond technical proficiency; effective teamwork is crucial. Regardless of individual skills, success hinges on communication, a strong work ethic, and openness to diverse perspectives. Articulating ideas, addressing issues, and contributing to team success are essential. Collaboration thrives in an environment where professional standards transcend personalities, differences, and project priorities reign supreme. In such an environment, individuals appreciate each other for the skills they bring to the team. Professional respect fosters a culture where teammates seek help, collaborate on problems, and share ideas and solutions. Ultimately, success is rooted in being a good person and fulfilling assigned responsibilities.