

Assignment2

QUESTION2:

1. About why it's important that software products are developed and delivered quickly.
 - For the software products we focus on the software rather than its documentation, develop software in a series of increments, and aim to reduce process bureaucracy as much as possible
 - Incremental development: Implement and test feature. Breaking down the development process into smaller, manageable increments allows for more focused work on specific features. Then Integrate the feature into the system. This approach aligns well with Agile methodologies, enabling continuous delivery and continuous improvement.
 - Reduced Time-to-Market: Speedy development and delivery reduce the time it takes for a product to reach the market. Some companies use the "Waterfall" method to reduce the time.
2. Why it sometimes sensible to deliver an unfinished product and then issue new versions of that product after deliver.
 - Quicker Involve the customer: Involve customers closely with the software development team. Early product releases can engage users and create a user base that provides valuable insights for future development. We can also get feedback from the customer, and according the market to improve our products.
 - Embrace change: Expect the features of the product and the details of these features to change as the development team and the product manager learn more about the product. Adapt the software to cope with changes as they are made.
 - Agile methodologies: Agile methodologies advocate for iterative development and continuous delivery. Agile makes the products Able to move quickly and easily.

QUESTION3:

- Reduction of Technical Debt: Minimizes technical debt by addressing defects and maintenance tasks during each sprint planning session. Sprints offer opportunities to fix issues alongside new feature development.
- Adaptability to Change: Encourages teams to quickly embrace and adapt to change. Time-boxed iterations eliminate the need for lengthy approval processes, allowing teams to efficiently handle changes and maintenance tasks.
- Alignment and Transparency in Mobile App Development: Agile fosters collaboration among team members. Before each sprint, the entire team reviews and agrees on user stories, working closely throughout the sprint to ensure alignment. Daily meetings maintain transparency and ensure everyone stays on the same page.
- Risk Minimization: Agile minimizes uncertainty by providing daily feedback and immediate action. Sprints allow teams to adjust quickly, ensuring they are on track and delivering value at every release, reducing the risk associated with traditional waterfall projects.

- **Higher Quality Product:** Focusing on incremental development improves product quality. Assigning a subset of features to each sprint allows developers more time for proper testing. Continuous integration ensures daily testing and prompt issue resolution.
- **Predictable Delivery Dates:** Result in working products at each release, providing predictability in delivery dates. Product owners can expect new features at the end of every sprint, unlike the uncertainty in lengthy waterfall cycles.
- **Enhanced Stakeholder Engagement:** Agile promotes active engagement of product owners throughout the development process, in contrast to the limited involvement in waterfall projects. This increased ownership encourages sustained stakeholder engagement.
- **User-Focused Testing:** This user-focused approach ensures a clear understanding of customer needs.
- **Greater Customer Satisfaction:** Agile sprints leads to higher engagement and satisfaction. Regular working product releases at the end of each sprint contribute to customer contentment.
- **Improved Project Control:** Daily meetings, meet face to face, planning sessions, and retrospectives enhance project control, allowing quick issue resolution and continuous improvement.

QUESTION4:

1. Comments on these ideas:

- **Able to move quickly and easily:** These small releases allow users to experience and provide feedback on the product early in the development process, enabling the team to make necessary adjustments. The small releases is also easy to control and test .
- **Reduced Risk:** break the product into small releases, if we meet some problems or bugs in one section, other sections will not be influenced.
- But these will still have some disadvantages, may not be acceptable to the users.

2. May not be acceptable to the users:

- **Security Concerns:** Users are worried about their security and personal information. Frequent releases may raise concerns about security vulnerabilities, especially if updates are not thoroughly tested.
- **Incomplete function:** Users may unhappy if essential features are incomplete or unavailable in the initial releases. Incomplete functionality might hinder their ability to fully evaluate the software and could lead to a negative user experience.

QUESTION5:

1. It is important that each sprint normally produces a potentially shippable product increment.

- **Short duration activities:** Sprints are short duration activities (1-4 weeks) and take place between a defined start and end date. Sprints are timeboxed, which means that development stops at the end of a sprint whether or not the work has been completed.
- **Continuous Progress and Customer Feedback:** Ensures consistent progress in product development and allows for quick incorporation of customer feedback.

- Stakeholder Involvement: Stakeholders are consulted during Product Backlog Refinement, ensuring their feedback and ideas are considered for effective product backlog management.
- Review and Adaptation: Regular reviews are conducted to assess completed deliverables and those yet to be finished, making necessary changes based on user feedback. This is crucial before moving into the delivery phase and planning for the next sprint.
- Risk Reduction: By ensuring that only tested and working features are released at any given time, the approach helps minimize risks associated with product development.

2. When might the team relax this rule and produce something that is not ready to ship.

- When there is an urgent user need: Because we prioritize user needs, if the product is of utmost importance to users and there is an urgent demand, we may relax this rule to meet user requirements promptly.
- When significant portions are already completed: If the core features are largely complete and based on testing feedback, some functionalities are deemed unnecessary, we can relax this rule.

QUESTION6:

1. What problems might arise if you try to use Scrum for student team projects in which members work together to develop a program.

- Timeboxed sprints :Scrum adhere to fixed sprint lengths Fixed-time (2-4 week) periods. It is hard for student to have a fixed time.
- Self-organizing teams: Student only has limited experience in practices. So it is hard for them to Self-organizing teams make their own decisions

2. What parts of Scrum could be used in this situation.

- Management Part: The ScrumMaster is a Scrum expert whose job is to guide the team in the effective use of the Scrum method.
- Sprint Planning: Work items to be completed in that sprint are selected and, if necessary, refined to create a sprint backlog. This should not last more than a day at the beginning of the sprint.
- Backlog Management: This is a to-do list of items to be implemented that is reviewed and updated before each sprint.