



Unioeste - Universidade Estadual do Oeste do Paraná
CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS
Colegiado de Ciência da Computação
Curso de Bacharelado em Ciência da Computação

Equipe: Sistema de Gestão para a Guarda Mirim

Fabricio Cordeiro Marcos

Fernando Schumaker Fiedler

Luiz Eduardo Garzon de Oliveira

Docente: Victor Francisco Araya Santander

Cascavel 16 de abril de 2023

Questões técnicas, humanas e organizacionais na escolha de metodologias de desenvolvimento de software.

A maioria dos projetos incluem elementos de processos dirigidos a planos e ágeis. Decidir no equilíbrio depende de:

1. É importante ter uma especificação e projeto bem detalhados antes de passar para a implementação? Caso seja, provavelmente você precisa usar uma abordagem dirigida a planos.

O projeto pode ser considerado genérico por se tratar de um sistema de gestão de turmas e um dos desenvolvedores fazerem parte do stakeholders então existe uma visão previa do que o sistema deve objetivar e prover. Mas para algo realmente direcionado ao concreto e palpável uma especificação de projeto antes de implementar é bem-vinda.

2. Uma estratégia de entrega incremental onde você entrega o software para os clientes e recebe feedback rápido deles é possível? Caso seja, considere usar métodos ágeis.

Sim, a entrega pode facilmente ser incremental pois existe o constante contato com o cliente sendo assim o feedback é rápido permitindo alterações e desenvolvimento constante no projeto.

3. Qual o tamanho do sistema a ser desenvolvido? Os métodos ágeis são mais efetivos quando o sistema pode ser desenvolvido com uma equipe pequena que pode se comunicar informalmente. O que pode não ser possível para sistemas grandes que requerem grandes equipes de desenvolvimento, nesses casos, deve ser usada uma abordagem dirigida a planos.

O sistema é pequeno/médio os seus objetivos são claros e diretos e ainda é restrito a uma organização pequena e para um grupo seleto de usuários o que não torna o escopo complexo.

4. Que tipo de sistema está sendo desenvolvido? Abordagens dirigidas a planos podem ser necessárias para sistemas que requerem muita análise antes da implementação (ex. sistema que opere em tempo real com requisitos de temporização complexos).

De modo geral é um sistema de gestão de turmas e alunos do projeto de extensão entre a guarda mirim e a Unioeste sendo restrito aos monitores e coordenadores do projeto. O sistema deve prover as principais funcionalidades relacionadas a este tipo software no quesito gestão de turmas.

5. Qual é o tempo de vida esperado para o sistema? Sistemas com longo tempo de vida podem precisar de mais documentação de projeto para comunicar as intenções originais dos desenvolvedores do sistema para a equipe de suporte.

Estima-se que o sistema esteja presente permanentemente com os integrantes do projeto pois atualmente nenhuma solução existe ao problema que enfrentam e visto que um dos integrantes do projeto faz parte desse projeto de extensão além do que outros integrantes também são da área de tecnologia a manutenção do sistema não deve carecer de mão de obra e dificuldades no uso, manutenção e evolução o sistema não devem gerar cenários caóticos no decorrer do tempo.

6. Quais tecnologias estão disponíveis para manter o desenvolvimento do sistema? Métodos ágeis dependem de boas ferramentas para acompanhar um sistema em evolução.

Como o time é pequeno e inexperiente foi firmado o uso das principais tecnologias para o desenvolvimento web como HTML, CSS E JS além da possível utilização do framework Bootstrap muito conhecido e utilizado ainda além de ser de fácil compreensão e uso. Além disso o Backend contara com mysql integrado a linguagem Java. Sendo assim a base de desenvolvimento se utiliza de ferramentas solidas no desenvolvimento de software.

7. Como está organizada a equipe de desenvolvimento? Se a equipe de desenvolvimento está distribuída ou se parte do desenvolvimento está sendo terceirizado você pode precisar desenvolver documentos de projeto para que haja comunicação entre as equipes de desenvolvimento.

A equipe é pequena e trabalha em conjunto, a comunicação é ampla sendo assim as etapas dos processos de desenvolvimento do projeto são cientes por parte de cada integrante da equipe.

8. Existem questões culturais ou organizacionais que podem afetar o desenvolvimento do sistema? As organizações tradicionais de engenharia têm uma cultura de desenvolvimento dirigido a planos, o que é padrão em engenharia.

A organização em questão não possui nenhum padrão ou exigência em modelos de projeto e desenvolvimento que devem ser seguidos, a necessidade é advinda do cliente e as metodologias utilizadas são critérios do time desenvolvedor que obviamente possui comunicação constante com o cliente que replica o feedback.

9. O quão bons são os projetistas e os programadores da equipe de desenvolvimento? É dito que os métodos ágeis requerem um nível de habilidade mais alto do que as abordagens dirigidas a planos, nas quais os programadores simplesmente traduzem um projeto detalhado em código.

Apesar de inexperientes os programadores possuem noções acerca do desenvolvimento a ser feito além alta capacidade de aprendizagem para resolver o problema em questão. Devido ao curto período disponível se torna mais vantajoso e produtivo aos desenvolvedores que implementem de forma constante e livre o sistema pois assim aprimoram suas habilidades e conhecimentos de forma rápida e eficiente.

10. O sistema está sujeito a regulamentação externa? Se o sistema precisa ser aprovado por um regulador externo (ex. O FAA aprova softwares críticos para a operação de um avião) então provavelmente exigem de você a produção de documentação detalhada como parte da documentação de segurança do sistema.

O sistema é restrito aos integrantes do projeto de extensão por tanto não necessita de uma regulamentação externa.

Fonte: Engenharia de Software. Ian Sommerville. 9a. edição. Addison Wesley, 2011.

O link a seguir direciona ao arquivo.xls contendo o cronograma para a alternativa escolhida no estudo de viabilidade do projeto.

[Cronograma.xlsx](#)

Metodologia escolhida de acordo com os critérios avaliados acima:

Metodologia Ágil - SCRUM

Processos ágeis são mais flexíveis pois o desenvolvimento é incremental tornando a modificação do processo de fácil implementação nos requisitos do cliente.

É um modelo incremental e iterativo baseado na entrega contínua de software que responde rapidamente as alterações dos requisitos do projeto.

Como nossos requisitos já estão bem definidos sobre como o software deve funcionar, o que deve fazer, o que deve executar e ser capaz de prover o processo ágil será útil no quesito tempo de desenvolvimento visto que o prazo para a implementação é curto.

Como o software é restrito a um grupo seleto sua evolução tende a inclinar-se a uma capacidade rápida de atualizações conforme novas necessidades, modificações e requisitos surgirem.

Visando uma entrega mais contínua e rápida para o cliente e considerando o constante contato com ele os métodos ágeis são úteis para o curto período que existem em todo o processo de desenvolvimento de software.

No nosso caso não existe inclinação a problemas com overhead (recursos que são necessários para executar tarefas que não estão diretamente relacionadas à criação do produto de software) que os requisitos são bem direcionados ao objetivo sendo assim não existe perda de tempo com extensas documentações sujeitas a mudanças.

Por existe o constante contato com o cliente e ainda considerando que o feedback é rápido além da colaboração e participação no projeto estarem presentes a metodologia ágil se encaixa perfeitamente no desenvolvimento do software.

Por se tratar de um sistema restrito e que não precisa de funcionalidades complexas a visão de um processo simples e direto são prioridades no desenvolvimento além do que as possíveis mudanças nas implementações poderão ser facilmente realocadas nas decisões de projeto.

O escopo do sistema é pequeno, a equipe do projeto é restrita, o envolvimento com o cliente é claro, os objetivos e requisitos em relação ao projeto são bem definidos considerando o número pequeno de stakeholders e ainda, o tempo para implementação é curto. Todas essas características favorecem o uso de métodos ágeis onde a busca pela entrega de um software funcional constantemente aprimorado é o foco do desenvolvimento. A grande vantagem está no feedback constante.

Como o cliente não possui nenhum sistema e alternativa sendo desenvolvida já é uma grande melhoria para suas necessidades a entrega de código funcional é mais relevante que a documentação extensa, restrita e complexa.

O Scrum favorece a essa entrega ágil e contínua de código funcional definido em Sprints que são ciclos curtos e iterativos. Como o software é entregue em partes menores o risco no desenvolvimento advindo de mudanças nas funcionalidades e requisitos, por exemplo, é minimizado.

Como a equipe é inexperiente e está em constante progresso em suas habilidades e conhecimentos no desenvolvimento de software o SCRUM é mais flexível pois foca mais na entrega funcional frequente de software e não na alta qualidade em si como na XP.

Scrum se encaixa bem na realidade da equipe pois é restrita e organizada, possui contato frequente com o cliente e seu feedback e visa a entrega rápida de um sistema funcional devido ao curto tempo buscando a maior gestão do projeto do que um código em alta qualidade constante.

Planejamento SCRUM:

A seguir é definido o planejamento das atividades que serão seguidas dentro de cada sprint presente no cronograma desenvolvido:

Product Owner (Luiz): Manter o backlog do produto e contato com stakeholders. Definição da priorização dos requisitos necessitados pelos stakeholders. Membro presente em todas as reuniões Scrum (Sprint Planning, Sprint Review, Histórias de usuário etc.) e tomadas de decisões.

Product BackLog: Lista priorizada dos requisitos mantida e atualizada pelo Product Owner (Luiz) de forma a atender as necessidades dos stakeholders.

Sprint Planning: Definição dos requisitos de maior prioridade a serem implementados pela equipe. A equipe se reúne para definir conforme as necessidades dos stakeholders qual a ordem de requisitos a serem implementados de forma a contemplar as expectativas sobre o software a ser desenvolvido. Essa etapa pode ser executada nas linhas “análise” e “projeto” no cronograma definido para cada interação.

Sprint: Implementação dos requisitos, fase de desenvolvimento sobre a história selecionada. Os passos selecionar, desenvolver, revisar e avaliar entram em ação. Fase definida na linha “implementação” do cronograma e “testes”.

Daily Scrum: breve contato entre a equipe para atualizações sobre a sprint em execução. Serve para sanar dúvidas e solucionar possíveis problemas que podem surgir ou surgiram no decorrer do processo. Este contato garante maior sincronia e harmonia entre a equipe e garante maior foco e objetividade sobre o requisito desenvolvido. Este passo pode ser contemplado tanto da “implementação” quanto nos “testes” no cronograma.

Sprint Review: Deploy para o cliente e reunião entre equipe e stakeholders para avaliar o requisito desenvolvido na sprint. Esta fase esta definida nas linhas “Deployment e Avaliação do Cliente”. Este processo visa receber o feedback dos stakeholder e pode retornar uma atualização do backlog do produto.

Sprint Retrospective ou Closure: Ocorre após o feedback dos stakeholders, ocorre entre os desenvolvedores, análises sobre a sprint são feitas, oportunidades e mudanças sobre a sprint são avaliadas, mudanças podem ocorrer conforme o feedback dos stakeholders, encerramento da sprint atual para planejamento da próxima sprint com base no backlog.