

ARP - Aprendendo a Resolver Problema

Anápolis, 26 de junho de 2023

Nome da disciplina: Requisitos e métricas de Software

Aluno: Matheus Marques Portela

RA: 2310823

Estratégias e Métricas

Existem inúmeras formas que se estimar um software a ser desenvolvido, e podemos escolher algumas estratégias para medi-lo. No caso estamos desenvolvendo um sistema de Cadastro e Controle de Pacientes da Rede Pública de Saúde para o governo, logo abaixo listaremos duas estratégias que julgamos como essenciais para o desenvolvimento de um software integro e perfeito:

- **Gráficos de burndown e burnup:**

- Os gráficos Burndown e Burnup são ferramentas importantes para mostrar a evolução do trabalho ao longo do tempo e quanto trabalho está faltando ser concluído. Ele será de extrema importância pois esses gráficos auxiliaram a equipe de desenvolvimento e os stakeholders a identificar se o trabalho está progredindo conforme o planejado, permitindo que sejam tomadas ações corretivas caso haja desvios significativos. Além disso, eles facilitaram a comunicação e a transparência sobre o andamento do projeto assim não ficaremos perdidos em questão do tempo. Esses dois gráficos podem ser feitos em conjunto para fornecer uma visão completa do progresso do projeto.
- **Justificativa:** Os gráficos de burndown e burnup fornecem uma representação clara e visual do progresso do projeto. Eles permitem que a equipe de desenvolvimento identifique rapidamente se o trabalho está seguindo conforme o planejado e se ajustes são necessários. Além disso, os gráficos de burndown e burnup fornecem insights para a melhoria contínua do processo de desenvolvimento. Portanto, essas ferramentas são essenciais para um gerenciamento eficaz do projeto, garantindo transparência, comunicação, controle e sucesso na entrega do software.

- **Utilização de métricas de satisfação do usuário:**

- Uma outra métrica que optamos em utilizar é a satisfação do usuário, pois ao medir a satisfação do usuário é uma estratégia valiosa para avaliar o sucesso do desenvolvimento do software, graças ao feedback obtido podemos analisar como nosso sistema está se comportando, se ele entrega os dados de forma rápida e integra, se está fácil e intuitivo, tudo isso faz com que melhores cada vez mais o software. E isso pode ser feito por meio de

pesquisas de satisfação, entrevistas com os usuários ou análise de dados de feedback e avaliações, que nesse caso nossos Stakeholders fariam essa ponte em busca destas informações.

- **Justificativa:** A satisfação do usuário é um indicador importante do valor entregue pelo software. Ao coletar informações sobre a satisfação dos usuários, é possível identificar pontos fortes e áreas de melhoria do sistema. Essa estratégia ajuda a validar a eficácia do software em atender às necessidades dos usuários que utilizaram desse sistema e pode direcionar ajustes ou melhorias futuras.

- **WIP (WORK IN PROGRESS):**

- E por fim uma outra estratégia importante é a utilização WIP. No WIP incluímos a quantidade de trabalho em progresso, que vai desde de o refinamento que seria todo o BackLog do nosso sistema até a aprovação dos mesmos. Com ele sabemos quanto trabalho está e cada etapa, quanto tempo ele está parado ali e com isso podemos identificar se durante o desenvolvimento está ocorrendo algum gargalo em alguma etapa e assim sancionar o problema mais rápido possível.
- **Justificativa:** Utilizar o WIP no desenvolvimento de software traz várias vantagens. Controlar e limitar a quantidade de trabalho em progresso ajuda a equipe a gerenciar o fluxo de trabalho, evitar sobrecarga e maximizar a eficiência. Além disso, o WIP permite manter o foco nas tarefas mais importantes e priorizadas, identificar gargalos e otimizar o processo de desenvolvimento. Com menos trabalho em andamento, a equipe pode responder mais rapidamente, reduzir retrabalho e desperdício, e melhorar a qualidade do produto final. Essa abordagem resulta em um ciclo de feedback mais curto e em uma entrega mais consistente, alinhada às expectativas dos usuários. Em resumo, o WIP promove um fluxo de trabalho mais controlado, focado e eficiente, resultando em melhores resultados no desenvolvimento de software.

Portanto, para desenvolver o sistema é essencial adotar estratégias que auxiliem no gerenciamento eficaz do projeto, garantindo transparência, comunicação, controle e sucesso na entrega do software. Ao combinar essas estratégias, nossa equipe de desenvolvimento será capaz de ter uma visão clara do progresso do projeto, identificar possíveis desvios e áreas de melhoria, garantir a satisfação dos usuários e entregar um software de qualidade, atendendo às necessidades do requerente de forma eficiente e eficaz.