

Universidade Federal Rural de Pernambuco

Cloud Assistant

Plano de Desenvolvimento de Software

Desenvolvedor:

Marcelo Manchester

1 - Introdução

No plano de projeto, apresentamos a descrição do Cloud Assistent. Este sistema visa automatizar e cooperar ao máximo com o processo de deploy e manutenção de aplicações na nuvem, levando em conta funcionalidades indispensáveis tais como: Implantação da aplicação no Google Kubernetes Engine, update de versão, autoescalamento dos recursos computacionais destinados a aplicação e configurações zero-downtime para que as aplicações estejam sempre disponíveis aos usuários.

O Cloud Assitent irá utilizar a interface do telegram para interagir com seus usuários.

Finalidade

O Plano do Projeto tem como objetivo explicitar a maneira de como será dada a condução do desenvolvimento do software. Nesse documento, serão expostos fatores levados em conta no planejamento, como recursos de hardware e software, assim também como vamos modular o desenvolvimento das aplicações que juntas comporão o sistema como um todo e uma estimativa do nosso cronograma, podendo este ser alterado. Enfim, o Plano de Desenvolvimento de Software é utilizado para controlar o projeto.

Escopo

Este *Plano de Desenvolvimento de Software* descreve o plano geral a ser usado pelo projeto *Cloud Assitent*, relativo à disciplina de Fundamentos de Engenharia de Software, ministrada pelo professor Jones Albuquerque, incluindo a implantação do produto. A equipe usará o conteúdo deste documento ao máximo para elaborar um projeto fiel às descrições que serão listadas no decorrer, pois mantendo essa fidelidade, as chances de sucesso do desenvolvimento aumentarão.

1. - Analise geral do Projeto

Objetivos e finalidade do projeto

O sistema é voltado para desenvolvedores de software, tendo o objetivo de facilitar o processo de deploy dos seus projetos, além de prover toda a manutenção necessária para que o sistema esteja sempre disponível. Os usuários do sistema poderão fazer todo o gerenciamento de suas aplicações na nuvem do Google, isto inclui: implantação (deploy), atualização de versão (essencial para entregas incrementais), verificar estado de suas aplicações e configurar regras de autoescalamento dos containers das aplicações.

Todo esse gerenciamento será feito utilizando um BOT do Telegram, portanto os usuários terão que utilizar a interface do telegram para gerenciar suas aplicações.

O sistema tornará mais eficaz a realização da entrega do software para os desenvolvedores, que poderão ter suas aplicações rodando de forma eficaz, escalar e segura.

Algumas restrições do Projeto

:

- As atividades do projeto serão desenvolvidas por um desenvolvedor.
- A linguagem de programação utilizada será JavaScript (Node js).
- O projeto utilizará os recursos do Google Kubernetes Engine.
- A gerenciamento das aplicações será feita utilizando o orquestrador de containers Kubernetes.

Evolução do plano de Desenvolvimento

Antes do começo de cada fase do projeto O Plano de Desenvolvimento de Software será analisado e poderá sofrer alguma alteração para poder completar alguma especificação esquecida ou uma alteração necessária que antes não foi observada.

2. - Organização do Projeto

Estrutura Organizacional

O software será desenvolvido, gerenciado e entregue por Marcelo Manchester.

3. - Processo de Gerenciamento

Estimativas do Projeto quanto ao tempo e custo

A estimativa é que o projeto comece a ser desenvolvido no começo do mês de setembro e termine no começo do mês de novembro, com uma pequena margem de erro sobre essa data.

Nas versões posteriores do documento traremos mais detalhes sobre datas de entrega.

Plano de Projeto

Inicialmente ficou decidido que o desenvolvedor irá desenvolver o projeto em Sprints semanais, ao final de cada sprint será feita uma análise sobre o andamento do desenvolvimento do software e a seleção das tarefas a serem realizadas na sprint posterior. As tarefas serão gerenciadas utilizando a plataforma Trello.

Recursos Utilizados

Os recursos de Hardware utilizados serão o laptop pessoal do desenvolvedor e os recursos do Google Cloud Platform.

No caso dos recursos de software utilizados projetos serão:

- *O interpretador de código JavaScript Node.js.*
- *Docker para o deploy da aplicação.*
- *O orquestrador de Containers (Docker) Kubernetes.*
- *As APIs do Telegram para construir o BOT.*
- *O processador de linguagem natural do Google Dialogflow.*

Gerenciamento de Requisitos

Os requisitos desse sistema são capturados no Documento de Requisitos. Após passarem por todo processo os requisitos são inseridos no documento e a partir dele, poderemos gerenciar que precisa ser feito e como o documento está sujeito a modificações se necessário de acordo com a metodologia que mais se adequa ao andamento do projeto.

Controle de Cronograma e Orçamento

O controle do cronograma, como já foi definido, será monitorado pelo gerente do projeto. A partir do cronograma, o gerente do projeto poderá ter noção de como anda o rendimento e analisar a necessidade de uma possível alocação de recursos necessários, no decorrer do projeto visando o seu gerenciamento.

O cronograma deverá manter uma programação mostrando a data esperada de cada marco, se ocorrer problemas em alguns marcos todo cronograma deverá ser revisto, e reformulado visando solucionar esse problema. Os itens de linha na programação incluem pacotes de trabalho que devem serem implementados dentro do prazo estipulado, embora possa haver exceções.

Relatórios e Métricas

Relatórios serão expostos ao fim de cada tarefa, visando um bom gerenciamento do prazo, neste documento, afim das estimativas estarem o mais próximo possível do cronograma real do projeto, evitando atrasos o que não é recomendável.

Análise de Riscos

Serão realizadas análises de riscos no início de cada sprint, com o intuito de avaliar os possíveis riscos e com isso tentar eliminar algum eventual atraso de entrega no transcorrer do desenvolvimento do projeto do grupo.

Custos

N/A