**PROVA – CONTINUADA 2 – ENGENHARIA DE SOFTWARE**

1. Escolho o Processo de Cascata com Realimentação. Pois, por se tratar de um sistema crítico (Área da Saúde e também é um projeto com a Prefeitura- Governamental) . Devido ser uma equipe metódica e as responsabilidades bem definidas constituem requisitos bem definidos e mensurados, e seus muitos documentos será útil quando o projeto for implementado, a duração de 1 ano não seria um problema para processo como o Cascata com realimentação, pois são mais longos. E a possibilidade de realimentação ajudará a rever algo da etapa anterior. Porém após todos requisitos levantados, eles serão congelados.
2. As atividades seriam:
3. Especificação: Realizando reuniões para que possa levantar os requisitos de maneira eficaz e transparente. Pois, após essa etapa serão congelados (requisitos). Como a duração do projeto é de 1 ano, essa etapa terá uma duração de 1 mês, tendo reuniões semanais para esculpir os requisitos.
4. Prototipação: Por ser um sistema que terá muitas integrações com API, ferramentas e aparelhos médicos, é importante ter um protótipo para garantir o alinhamento de todos na equipe sobre o novo projeto, tornando uma nova documentação para o processo. Duração de 2 Semanas \*uma reunião por semana.
5. Desenvolvimento: Como é um projeto de cascata, e não trabalha com pequenas entregas, terá um período longo para fazer todo o desenvolvimento do projeto. Duração de 7 meses.
6. Testes: Para que possa colocar em produção o projeto da melhor maneira possível e que não dê bug, terá a etapa para realizar debugs e testes. Duração de 1 mês e meio.
7. Cronograma: Para que possamos controlar todas as atividades facilitando a percepção e gerenciamento de cada etapa de maneira eficaz. E por ser um processo de cascata com realimentação, terá 2 meses para que se durante os testes perceber que precisa de melhorias, voltará para a etapa de desenvolvimento sem atrapalhar o prazo de entrega.
8. Ferramentas:

Existem muitas ferramentas importantes para esse projeto, desde controle de versionamento a reste de rotas. Porém, vou citar duas que sem elas (básicas), as demais não surtirão efeito. São elas:

1. MsProject, como já mencionado na questão acima, ele será muito útil para que tenha um controle efetivo e eficaz de maneira a poder fazer alguma correção a tempo sem prejudicar a entrega do projeto. Atendendo assim o prazo proposto. Estamos falando de um projeto na área da saúde e governamental, estamos lidando com vidas.
2. Plataforma/IDE Netbeans, por se tratar de respiradores com Sistemas/API’s que integram com a linguagem de programação JAVA e C++. O Netbenas é uma IDE ideal, pois é possível desenvolver com essas duas linguagens. E sem uma IDE correta e preparada para dar suporte e infraestrutura para o desenvolvedor codificar, não será possível atender ao prazo e desenvolver com o máximo de potencial que o projeto necessita.
3. A arquitetura do Software será separada em partes, dentre elas:
4. Sistema DataBase:

1- Banco de Dados (Azure – SQL Server): para que armazene todos os dados do Sistema, equipamentos, relatórios e demais informações capturadas.

1. Sistema de Gestão:
2. Web Application: Container- SpringMVC, que terá aplicação de cadastros e relatórios (Ex: ORM e Report Controller);
3. Mobile App: Devido precisar um módulo de comunicação rodando em Android Kotlin;
4. ClientSide Web: Com dashbords, relatórios e possíveis cadastros (React, css e JS.
5. Sistema Coletor:
6. Conexão com o BD através de JDBC, controlando essas transferências de informações;
7. Log TXT: para que envie possíveis erros da aplicação em extensão TXT;
8. Componentes Java HTTP. Webhook, como o Slack para que envie alertas e outras mensagens caso necessário, que chamará a API de mensageria;
9. Captura de dados: como o Java OSHI, pois o sistema deverá avisar um NOC e os enfermeiros caso seja necessário.
10. SLACK Application:
11. Container com API Slack: Enviando assim seus alertas quando necessário.

OBS: O único item que não terá controle nosso e usaremos de uma ferramenta externa é o item “d) Slack Application”, caso fosse desenhada a Arquitetura de Software esse item/container seria tracejado, revelando que o controle da API é externo.

QUESTÃO EXTRA:

5- Já que se trata de um Sistema de UTIs de Hospitais de Campanha que monitora respiradores, um adicional seria verificar a a quantidade de acordo com a capacidade de cada Hospital para que não ocorra lotação/super lotação prevenindo que os socorristas (SAMU) levem os pacientes para Hospital com sua capacidade já limitante. Esse sistema seria integrado com os funcionários de dentro do hospital e o SAMU, para que os socorristas levem os pacientes para hospitais com disponibilidade. Outro caso é se chegar algum paciente, já seja informado de maneira fácil, prática e rápida para hospital com disponibilidade. E também melhorar a comunicação entre as Redes de UTIs, tendo uma comunicação mais efetiva. Essa solução plus pode até ser informada em algum totem na entrada do hospital avisando que tal aréa, nesse caso os respiradores está atingindo seu limite máximo.