

EAP – Estrutura Analítica do Projeto

A Estrutura Analítica do Projeto (EAP) é uma ferramentaria de gerenciamento de projetos que divide o trabalho em partes menores e mais gerenciáveis. A EAP é uma representação gráfica da decomposição do trabalho a ser executado em um projeto que permite a visualização de todas as entregas que o projeto deve realizar. A EAP é uma ferramenta fundamental para o planejamento e controle de projetos, pois permite a visualização de todas as entregas que o projeto deve realizar.

Este documento ficará estruturado da seguinte forma

- Análise de Riscos
- Escopo do projeto - Requisitos
- Cronograma - Gráfico GANTT
- CPM - Caminho Crítico
- Custos - Orçamento do Projeto

Análise de Riscos – SWOT

Análise de riscos utilizando a Matriz SWOT (Strengths, Weaknesses, Opportunities, Threats), uma sigla em inglês que significa Forças, Fraquezas, Oportunidades e Ameaças. FOFA em português.

Forças	Fraquezas
<ul style="list-style-type: none">- Facilidade e simplicidade na utilização- Utiliza princípios de UI – Responsividade e UX – Simplicidade- Registros centralizados em um Banco de dados Relacional- Segurança- Flexibilidade de implantação com recursos de código atualizados- Facilidade de manutenção devido a princípios MVC	<ul style="list-style-type: none">- Concorrência de diversas fontes- Poucas funcionalidades- Necessário implantação em servidor, nuvem para plena funcionalidade- Necessidade de manutenção constante- Necessidade de Backup
Oportunidades	Ameaças
<ul style="list-style-type: none">- Diversas empresas necessitam de sistemas semelhantes- Tanto pequenas, médias ou grandes empresas- Disponibilidade gratuita através de Monetização- Ganhos com fidelização de clientes	<ul style="list-style-type: none">- Concorrência- Investimento em divulgação pode não retornar resultados.- Erros funcionais podem retornar informações imprecisas- Divulgação de dados sensíveis- Burocracia

A análise conclui que o projeto é viável pois as forças e oportunidades são maiores do que as ameaças e fraquezas.

Escopo do projeto – Requisitos

O propósito deste documento é definir o escopo do projeto a partir das regras de negócio estabelecidas, que foram elencadas de diversas formas e fontes.

Regras de negócio

As regras de negócio do sistema ServiFacil foram obtidas com base em pesquisas de mercado realizadas através da internet, observação direta da atuação dos colaboradores da manutenção da escola, entrevista com professores e funcionários.

Estão elencadas a seguir de forma ordenada a partir da sigla RN (Regra de Negócio) seguida da ordem do requisito, conforme exemplo a seguir: [RN001]

[RN001] Para a utilização do sistema o funcionário, também chamado de colaborador, pois pode ser terceiro a empresa, precisa fazer um cadastro, tendo como identificador sua matrícula e um PIN (Senha numérica simples) além dos campos nome completo, cargo e setor.

[RN002] Ao realizar seu cadastro se o colaborador preencher o cargo como "Manutenção" ele será considerado um executor, caso contrário será considerado um colaborador.

[RN003] O executor pode visualizar as ordens de serviço, adicionar comentários e marcar como concluída, além de visualizar o histórico de ordens de serviço concluídas e um dashboard com gráficos de desempenho.

[RN004] O colaborador pode cadastrar, editar, excluir e visualizar as suas próprias ordens de serviço.

[RN005] Enquanto a ordem de serviço estiver em aberto, o colaborador pode editar, excluir e adicionar comentários. somente o executor pode marcar como concluída.

[RN006] Caso a ordem de serviço seja marcada como concluída, o colaborador não poderá mais editar, excluir ou adicionar comentários, somente visualizar no seu histórico, com cor diferente das ordens em aberto.

[RN007] Tanto o colaborador quanto o executor podem abrir novas ordens de serviço, que serão consideradas em aberto.

[RN008] O sistema na versão Web ao gerar novas ordens de serviço, deve posicionar a latitude e longitude do local da escola, para que o executor possa visualizar no mapa.

[RN009] O sistema na versão Mobile deve utilizar a geolocalização do dispositivo para posicionar a latitude e longitude do local da ocorrência, para que o executor possa visualizar no mapa.

[RN010] O sistema utilizará um SGBD free, como o MySQL, MariaDB e será dividido em API, Front-End e Mobile.

[RN011] O sistema Web deve ser responsivo, funcionando em dispositivos móveis e desktop, e apresentar dados de Produção e um Dashboard

[RN012] A API Back-End deve ser desenvolvida em NodeJS, utilizando o ORM Prisma para o banco de dados e autenticação JWT.

[RN013] O Front-End Web deve ser desenvolvido com HTML, CSS e JavaScript Vanilla ou React, consumindo a API Back-End.

[RN014] O sistema Mobile deve ser desenvolvido em Flutter, consumindo a API Back-End, em duas versões, uma para o colaborador e outra para o executor.

[RN015] A versão mobile do colaborador deve permitir cadastro, login e apenas abrir novas ordens de serviço e visualizar o histórico.

[RN016] A versão mobile do executor deve permitir cadastro, login, visualizar as ordens de serviço abertas e sua localização no mapa, adicionar comentários e marcar como concluída, além de visualizar o histórico de ordens de serviço concluídas.

[RN017] O sistema deve ser desenvolvido em português, com documentação clara e objetiva também em português seguindo as normas da ABNT.

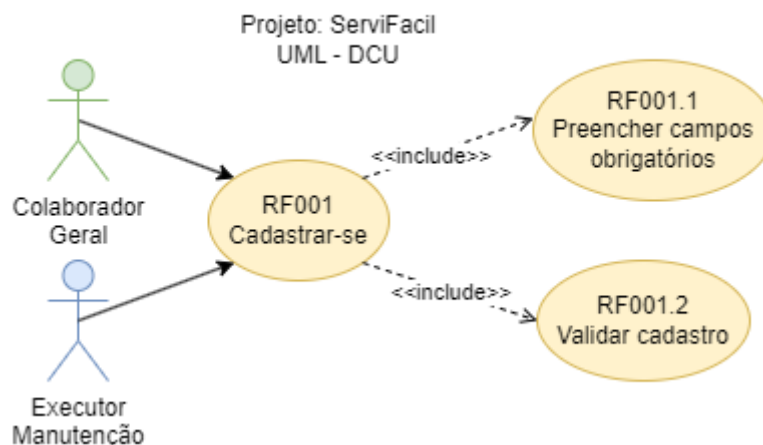
Requisitos funcionais

Os requisitos funcionais do sistema ServiFacil foram obtidos com base nas regras de negócio e estão elencados a seguir de forma ordenada a partir da sigla RF (Requisito Funcional) seguida da ordem do requisito, conforme exemplo a seguir: [RF001]

De acordo com a metodologia Ágil XP cada requisito é obtido através de uma breve história, que será ilustrada neste documento através da UML DCU (Diagrama de Casos de Uso).

Também será atribuído um nível de prioridade para o desenvolvimento entre os três: essencial, importante e desejável.

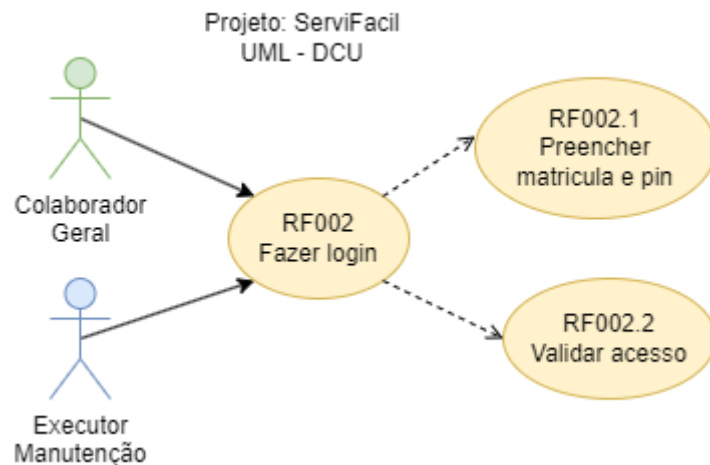
[RF001] **Cadastrar-se** - O sistema deve permitir o cadastro de colaboradores e executores.



Prioridade: [x] Essencial [] Importante [] Desejável

Referência: [RN001] [RN002]

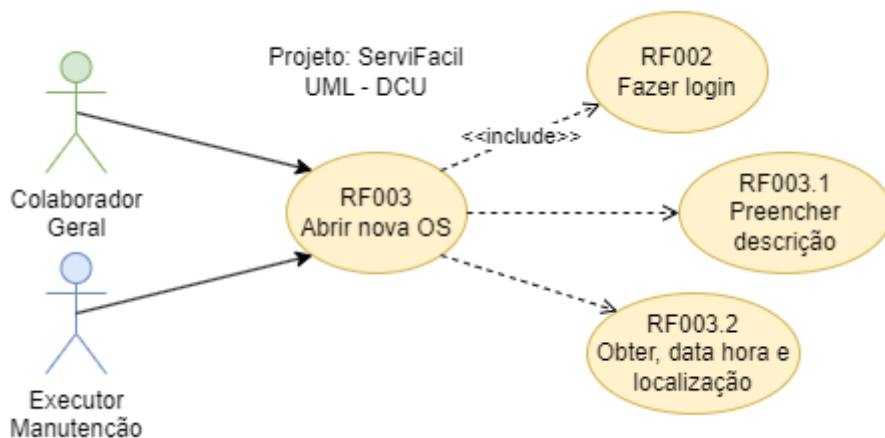
[RF002] **Login** - O sistema deve permitir o login de colaboradores e executores, aplicando autenticação JWT.



Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

Referência: [RN002] [RN003]

[RF003] **Abrir nova** - O sistema deve permitir o cadastro de ordens de serviço obtendo data e hora do sistema.



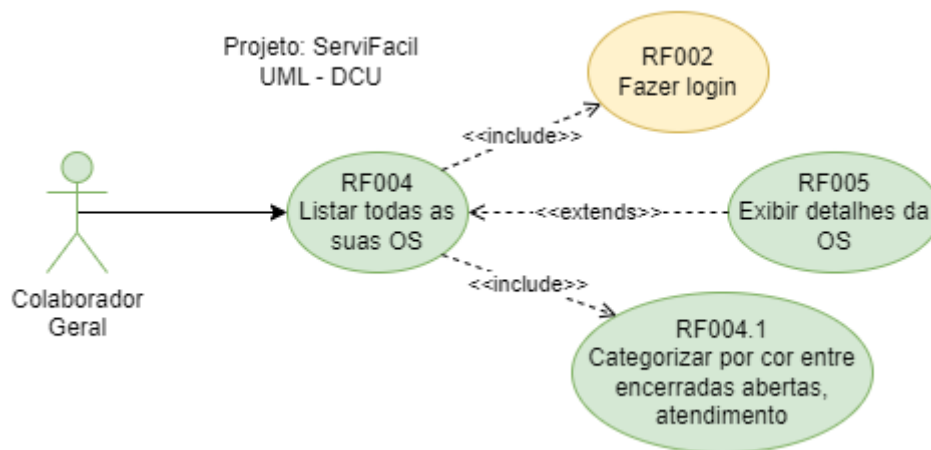
Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

Referência: [RN004] [RN005] [RN007] [RN008]

Obs: Obter posição geográfica na versão mobile, na versão Web preencher com a localização da sede da empresa, em nosso caso a escola.

Prioridade: ☐ Essencial ☒ Importante ☐ Desejável

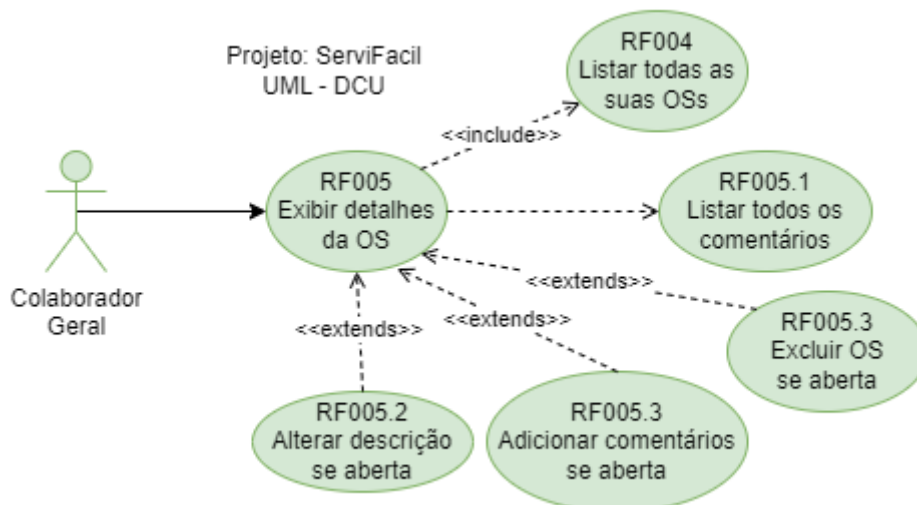
[RF004] **Listar todas as suas OSs** - O sistema deve permitir a visualização das ordens de serviço classificadas com cores diferentes para novas, em atendimento e concluídas.



Prioridade: [x] Essencial [] Importante [] Desejável

Referência: [RN005] [RN006]

[RF005] **Exibir detalhes da OS** - O sistema deve permitir a visualização dos detalhes da ordem de serviço, listando todos os comentários, se estiver aberta permitir acrescentar comentários, excluir ou alterar a descrição

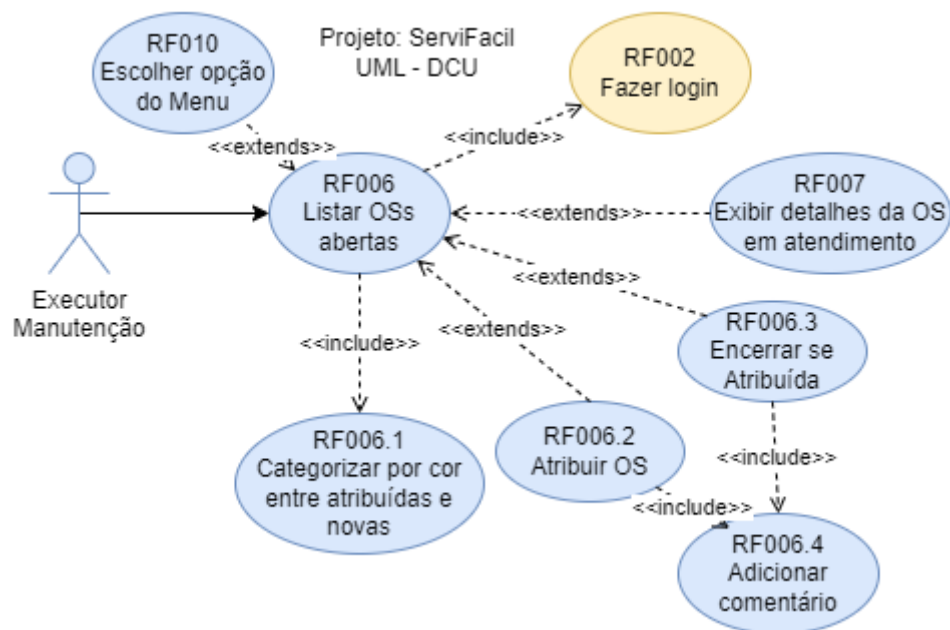


Prioridade: [] Essencial [x] Importante [] Desejável

Referência: [RN005] [RN006]

[RF006] **Listar OSs abertas** – A tela principal do Executor deve apresentar todas as OS que estão abertas aguardando atendimento, classificadas por cor entre atribuídas ou não, permitir que atribua qualquer OS para realizar o atendimento, mediante adição de um comentário de atribuição e permitir encerrar OSs quando atribuídas ao próprio colaborador mediante adição de um comentário de encerramento.

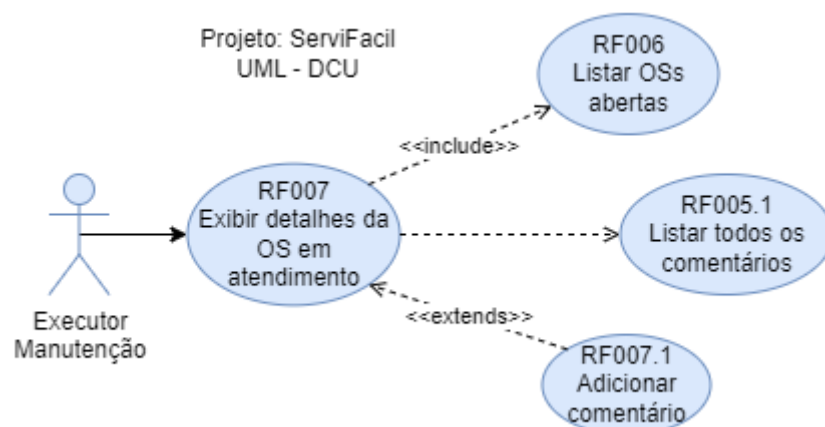
Também permitir var os detalhes de qualquer OS listada.



Prioridade: [x] Essencial [] Importante [] Desejável

Referência: [RN003]

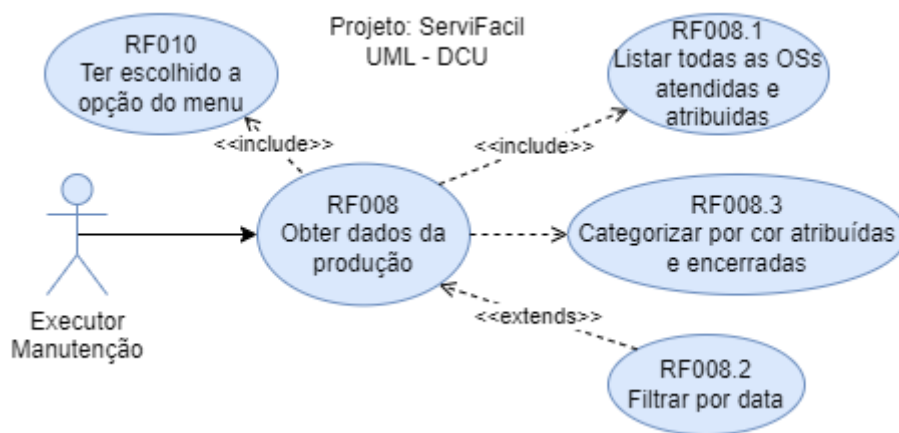
[RF007] **Exibir detalhes do atendimento** - O sistema deve permitir apenas listar os comentários e adicionar novos.



Prioridade: [] Essencial [x] Importante [] Desejável

Referência: [RN003]

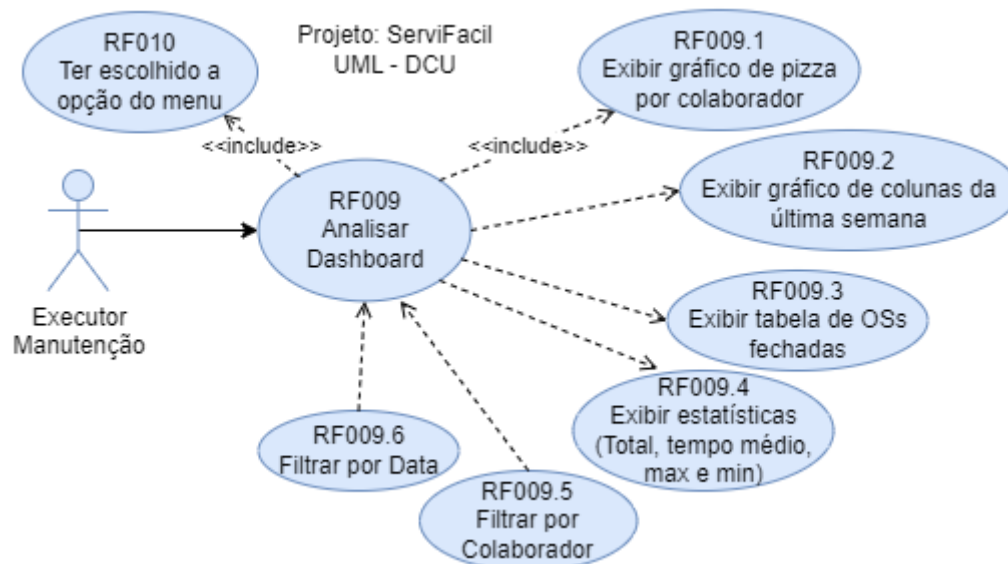
[RF008] **Obter dados da produção** – Deve listar todas as OSs encerradas pela equipe permitindo filtrar por data.



Prioridade: [] Essencial [] Importante [x] Desejável

Referência: [RN011]

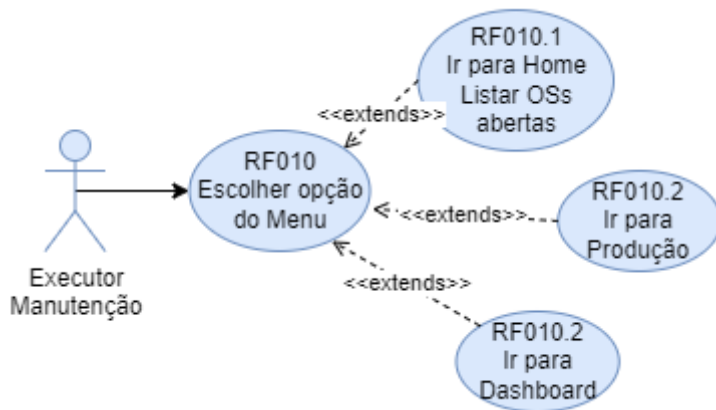
[RF009] **Analisar dashboard** – Deve apresentar gráficos comparando as produções dos colaboradores, dados estatísticos e permitir filtrar por data e colaborador.



Prioridade: [] Essencial [] Importante [x] Desejável

Referência: [RN011]

[RF010] **Escolher opção do menu** – Quando logado como executor deve ter acesso a um menu, de preferência responsivo tipo sandwich com as opções de Produção e Dashboard além de voltar a sua Home listando as OSs abertas.



Prioridade: [] Essencial [] Importante [x] Desejável

Referência: [RN011]

Requisitos não funcionais

Os requisitos não funcionais do sistema ServiFacil foram obtidos com base em pesquisas de mercado realizadas através da internet, observação direta das tecnologias de implantação em nuvem atuais. Para a classificação a sigla NF (Não Funcional) foi utilizada seguida da ordem do requisito, conforme exemplo a seguir: [NF001]

[NF001] Será observada a UX (User Experience) e a UI (User Interface) para garantir uma boa experiência do usuário.

Prioridade: [x] Essencial [] Importante [] Desejável

Referência: [RN013]

[NF002] A questão de segurança será observada com a utilização de autenticação JWT e a utilização de HTTPS.

Prioridade: [x] Essencial [] Importante [] Desejável

Referência: [RN012]

[NF003] O desempenho do sistema será observado com a utilização de um banco de dados relacional e a utilização de um ORM para otimizar as consultas.

Prioridade: [] Essencial [x] Importante [] Desejável

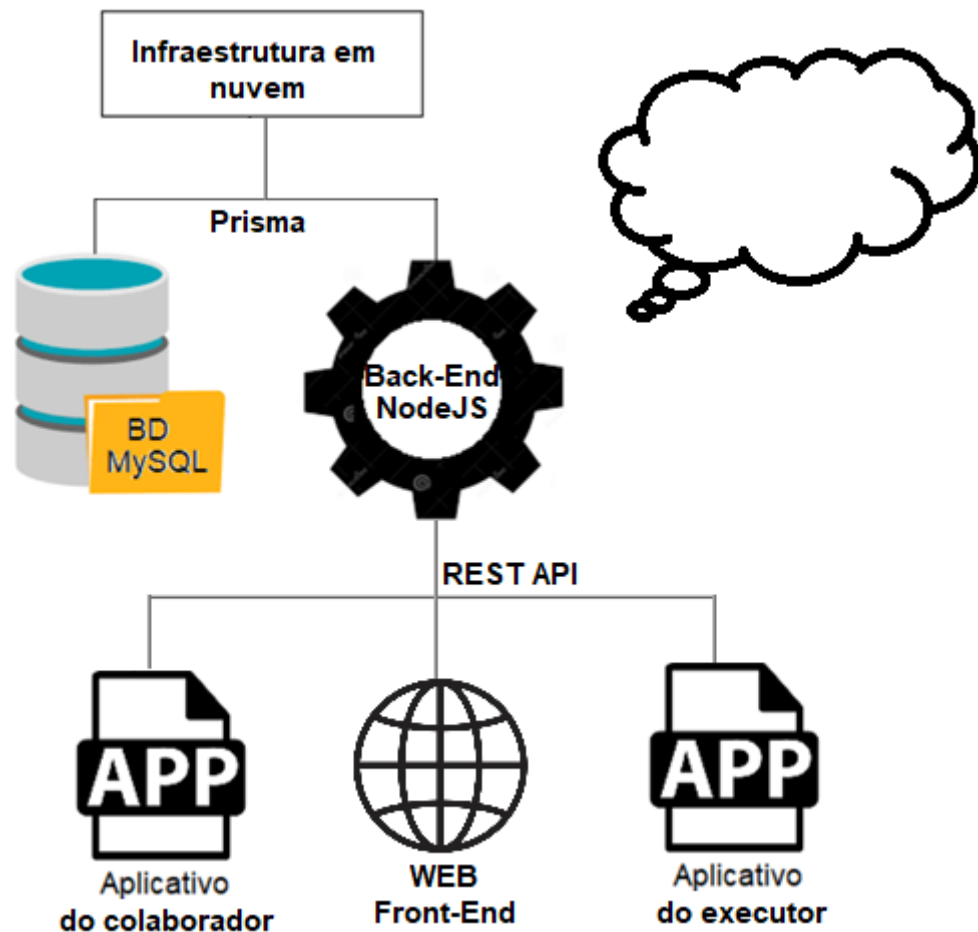
Referência: [RN012]

[NF004] Mobile desenvolvido com o framework Flutter, consumindo a API Back-End.

Prioridade: [] Essencial [] Importante [x] Desejável

Referência: [RN014]

[NF005] O sistema deve ser implantado em uma infraestrutura em nuvem. A ilustração a seguir representa como deve ser a implantação do sistema.

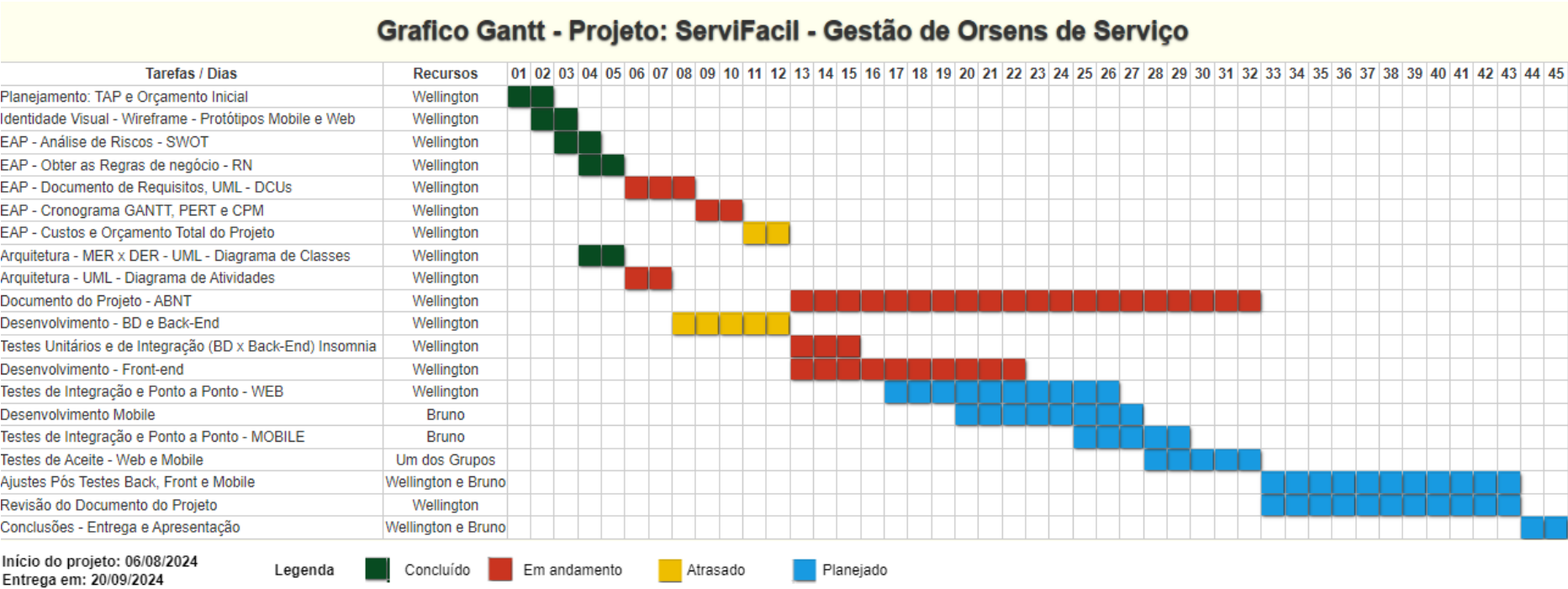


Prioridade: [] Essencial [x] Importante [] Desejável

Referência: [RN001]

Cronograma – Gráfico Gantt

As etapas deste projeto serão organizadas no backlog do produto conforme a metodologia Ágil SRUM e divididas em 4 Sprints de duas semanas, com duração total de 45 dias conforme o gráfico de Gantt na imagem a seguir:



COM – Caminho crítico

Para a elaboração do cronograma inicial acima foram aplicadas as ferramentas PERT e CPM.

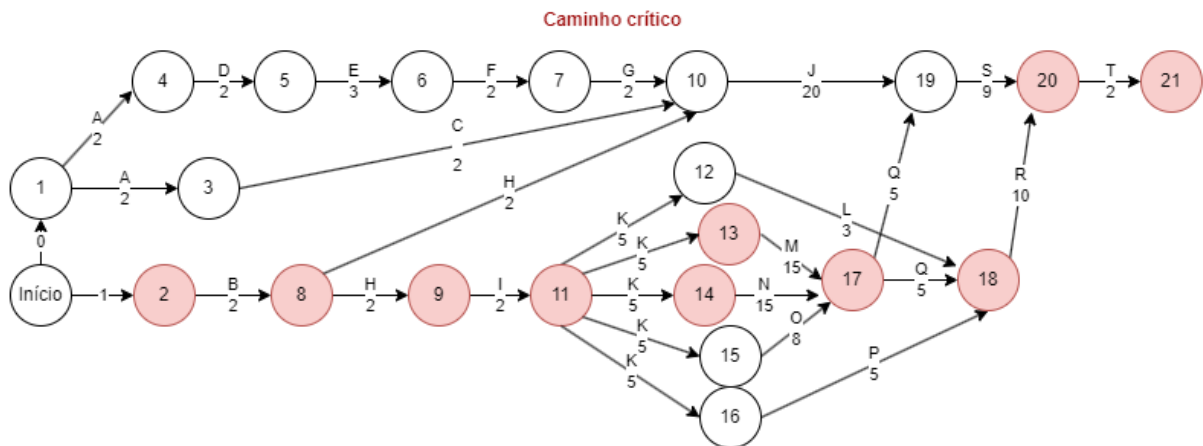
O PERT (Program Evaluation and Review Technique) e o CPM (Critical Path Method) são técnicas de gerenciamento de projetos que ajudam a planejar, agendar e coordenar tarefas dentro de um projeto. O PERT é usado principalmente em projetos de pesquisa e desenvolvimento, enquanto o CPM é usado para projetos de construção e manufatura, em nosso caso utilizamos o CPM pois o tempo para cada tarefa já foi estimado pela equipe e acordado entre os stakeholders no TAP (Termo de Abertura do Projeto).

A revisão do tempo das atividades está ilustrada na tabela a seguir:

Projeto: ServiFacil, Rede PERT - CPM			
Atividade	Dependência	Nós	Duração
A - Planejamento: TAP e Orçamento Inicial		1-3 1-4	2
B - Identidade Visual - Wireframe - Protótipos Mobile e Web		2-8	2
C - EAP - Análise de Riscos - SWOT	A	3-10	2
D - EAP - Obter as Regras de negócio - RN	A	4-5	2
E - EAP - Documento de Requisitos, UML - DCUs	D	5-6	3
F - EAP - Cronograma GANTT, PERT e CPM	E	6-7	2
G - EAP - Custos e Orçamento Total do Projeto	F	7-10	2
H - Arquitetura - MER x DER - UML - Diagrama de Classes	B	8-10 8-11	2
I - Arquitetura - UML - Diagrama de Atividades	H	9-11	2
J - Documento do Projeto - ABNT	C-G-H	10-19	20
K - Desenvolvimento - BD e Back-End	I	11-12 11-13 11-14 11-15 11-16	5
L - Testes Unitários e de Integração (BD x Back-End) Insomnia	K	12-18	3
M - Desenvolvimento - Front-end	K	13-17	15
N - Testes de Integração e Ponto a Ponto - WEB	K	14-17	15
O - Desenvolvimento Mobile	K	15-17	8
P - Testes de Integração e Ponto a Ponto - MOBILE	K	16-18	5
Q - Testes de Aceite - Web e Mobile	M-O	17-19	5
R - Ajustes Pós Testes Back, Front e Mobile	L-N-P-Q	18-20	11
S - Revisão do documento do projeto	J-Q	19-20	11
T - Conclusões - Entrega e Apresentação	R-S	20-21	2

Caminho crítico

O caminho crítico traçado na rede apresentada na imagem abaixo, calculado somando as atividades de maior duração em sequência, totalizando 43 dias.



O cálculo para traçar este caminho está destacado em vermelho na imagem a seguir:

Caminhos

$$A2 + D2 + E3 + F2 + G2 + J20 + S9 + T2 = 42$$

$$A2 + C2 + J20 + S9 + T2 = 35$$

$$B2 + H2 + J20 + S9 + T2 = 35$$

$$B2 + H2 + I2 + k5 + L3 + R10 + T2 = 26$$

$$B2 + H2 + I2 + k5 + M15 + Q5 + S9 + T2 = 42$$

$$B2 + H2 + I2 + k5 + N15 + Q5 + S9 + T2 = 42$$

$$B2 + H2 + I2 + k5 + O8 + Q5 + S9 + T2 = 35$$

$$B2 + H2 + I2 + k5 + P5 + R10 + T2 = 28$$

$$B2 + H2 + I2 + k5 + M15 + Q5 + R10 + T2 = 43$$

$$B2 + H2 + I2 + k5 + N15 + Q5 + R10 + T2 = 43$$

$$B2 + H2 + I2 + k5 + O8 + Q5 + R10 + T2 = 36$$

Orçamento

Estimativa de Custos / Orçamento inicial

Recurso	Hora	Valor Hora	Total
Design - Identidade visual, protótipo	20	30,00	600,00
Back-End + Banco de dados	40	30,00	1200,00
Front-End	80	30,00	1200,00
Mobile	100	30,00	3000,00
Integração Full-Stack	40	30,00	1200,00
Documentação do Projeto	20	30,00	600,00
Testes	20	30,00	600,00
Total			R\$ 8400,00

Orçamento oficial por funcionalidade

Requisito	Valor Tela	Funcionalidades	Total
RF001 – Cadastro	100	100	200
RF002 – Login	100	100	200
RF003 – Nova OS	100	150	250
RF004 – Listar suas	200	150	350
RF005 – Detalhes	200	250	450
RF006 – Listar Abertas	200	150	350
RF007 – Detalhes Atendimento	200	150	350
RF008 – Produção	500	100	600
RF009 – Dashboard	1000	700	1700
RF010 – Menu	100	100	200
Mobile Colaborador	1000	750	1750
Mobile Executor	1000	1000	2000
Total			R\$ 8400,00