

Cygni - AGIS

VISÃO DO PRODUTO E DO PROJETO

Versão 0.1

Tabela - Integrantes do Grupo:

Matrícula	Nome	Função (responsabilidade)
180114964	Adriano Fonseca Carvalho Costa	BackEnd, Banco de Dados, Infra.
221007798	Ana Julia Mendes Santos	FrontEnd e Infra.
221022462	Arthur da Silveira Sousa	FrontEnd.
221022490	Cauã Araujo dos Santos	FrontEnd e Banco de Dados.
222014859	Ian Costa Guimarães	BackEnd, Infra e Banco de Dados.
200025571	Necivaldo Amaral Pinto Junior	BackEnd.
231036980	Pedro Luiz Fonseca da Silva	BackEnd.
221022767	Weverton Rodrigues da Costa Silva	FrontEnd e BackEnd.

Histórico de Revisões

Data	Versão	Descrição	Autor
03/05/2024	0.1	Criação da primeira versão da documentação com informações sobre nosso produto e estratégias de desenvolvimento(Documento de visão).	Equipe Cygni

Sumário

1	<i>VISÃO GERAL DO PRODUTO</i>	4
1.1	Problema	4
1.2	Declaração de Posição do Produto	4
1.3	Objetivos do Produto	5
1.4	Tecnologias a Serem Utilizadas	5
2	<i>VISÃO GERAL DO PROJETO</i>	6
2.1	Ciclo de vida do projeto de desenvolvimento de software	6
2.2	Organização do Projeto	7
2.3	Planejamento das Fases e/ou Iterações do Projeto	7
2.4	Matriz de Comunicação	8
2.5	Gerenciamento de Riscos	8
2.6	Critérios de Replanejamento	9
3	<i>PROCESSO DE DESENVOLVIMENTO DE SOFTWARE</i>	9
4	<i>DECLARAÇÃO DE ESCOPO DO PROJETO</i>	10
4.1	Backlog do produto	10
4.2	Perfis	11
4.3	Cenários	11
4.4	Tabela de Backlog do produto	12
5	<i>REFERÊNCIAS BIBLIOGRÁFICAS</i>	13

VISÃO DO PRODUTO E PROJETO

1 VISÃO GERAL DO PRODUTO

1.1 Problema

A utilização de múltiplas plataformas para desempenhar tarefas semelhantes resulta em uma série de desafios. Em uma sessão de administração de recursos humanos, por exemplo, é comum a necessidade de recorrer a mais de um sistema para a realização do trabalho, gerando uma repetição das mesmas atividades. Esse trabalho não integrado, onde as informações muitas vezes estão dispersas em sistemas isolados, não apenas consome tempo e recursos, mas também aumenta as chances de erros e inconsistência nos dados. Como resultado, a produtividade pode ser afetada e a tomada de decisões baseada em dados pode se tornar menos precisa, limitando a capacidade de uma organização de alcançar seus objetivos de forma eficaz e eficiente.

1.2 Declaração de Posição do Produto

1. *Qual é o produto que o grupo se propõe a desenvolver?*

O AGIS é uma aplicação desktop de gestão de recursos humanos, que visa fornecer todas as funcionalidades desejadas pelo cliente, como a geração de relatórios, criação de escalas de trabalho e comunicação interna das fichas dos funcionários, além da geração de campanhas - calendário que mostra quem está apto a trabalhar no dia e mês.

2. *O que torna este produto diferente dos seus concorrentes (se existirem concorrentes)?*

Este produto se destaca pela geração automática de campanhas e escalas com base nas fichas dos funcionários. Sua capacidade de operação offline é uma vantagem significativa para a segurança, garantindo a continuidade do trabalho em diferentes ambientes e condições de conectividade.

3. *Quem são os usuários-alvo e clientes do produto? Quais suas características e porque o produto é importante para eles.*

Os usuários-alvo são gestores de pessoas, especificamente escalantes (caso do nosso cliente), responsáveis pela criação e administração de escalas de trabalho dentro das empresas. Para eles, o produto é de extrema importância, pois oferece uma solução que simplifica e otimiza suas tarefas diárias.

4. *Por que os clientes deveriam utilizar / comprar este produto?*

O produto oferece uma solução confiável e eficiente para a criação de escalas e manuseio de fichas de funcionários. Com o servidor operando internamente na empresa, garantimos segurança e controle sobre os dados. Além disso, proporcionamos funcionalidades como a geração automática de relatórios, campanhas e escalas de trabalho, simplificando processos e tornando o trabalho mais eficiente e otimizado.

Diagrama de Ishikawa

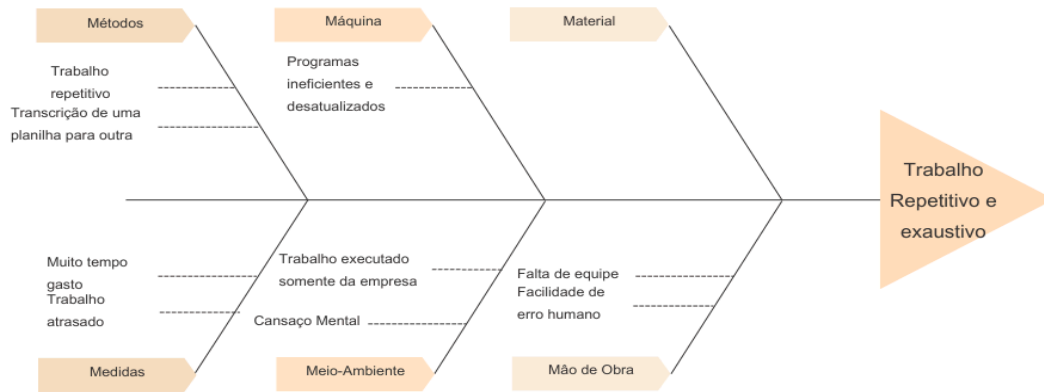


Imagem 01- Diagrama de Ishikawa

Para:	Alisson, gestores de pessoas, em especial escalantes
Necessidade:	gerenciamento dos funcionários, geração de relatórios de pessoal
O AGIS:	é uma aplicação desktop.
Que:	visa a economia de tempo, eficiência e confiabilidade para a administração de RH
Ao contrário:	da utilização de múltiplos sistemas e plataformas não integradas
Nosso produto:	proporciona a centralização de dados e processos, além de fornecer a geração automática de relatórios, campanhas e escalas, funcionando em um servidor interno.

Tabela 01- Tabela de organização do Projeto

1.3 Objetivos do Produto

- Gerir pessoas com eficiência, e principalmente, com confiabilidade.
- Integrar funções que funcionam separadamente. Integrar vários softwares de gerência em somente uma plataforma.
- Objetivo secundário: Geração de escalas.

1.4 Tecnologias a Serem Utilizadas

- **Linguagens de programação e Frameworks:** Utilizaremos o HTML, CSS e JavaScript como marcadores e linguagens de programação, em conjunto com o framework Bootstrap,

para FrontEnd. Para o Backend, usaremos JavaScript, em conjunto com o NestJS e Express.

- **Ambiente de desenvolvimento:** Como a maioria dos integrantes utilizam o Visual Studio Code, o utilizaremos para a edição de códigos, aproveitando de suas tecnologias de Live Sharing etc. Durante o desenvolvimento do projeto, o controle das versões será feito por Commits no GitHub. Para facilitar a configuração do ambiente de desenvolvimento, será usado o Docker para rodar a aplicação em ambiente local.
- **Arquitetura de Software:** Utilizaremos a Arquitetura em Camadas, na qual a separação de preocupações é a principal propriedade da arquitetura, cada camada de software tem uma função específica. Isso facilita a atualização de camadas individuais e também permite que as equipes separem as cargas de trabalho.
- **Arquitetura do Banco de Dados:** Optamos pelo MongoDB, um banco de dados No-SQL que oferece praticidade e flexibilidade no controle de coleções de dados e, devido a maturidade do projeto, também oferece de forma nativa ferramentas para sua gestão, dispensando o uso de ORMs e ferramentas de visualização de dados à parte.
- **Ferramentas:** A fim de organizar as atas de reuniões, funções e tarefas utilizamos o Miro (com o uso dos quadros Kanban). Para as reuniões, usamos o Discord. Electron para desenvolvimento da aplicação.

2 VISÃO GERAL DO PROJETO

2.1 Ciclo de vida do projeto de desenvolvimento de software

- **Metodologia:** Para maximizar a flexibilidade, facilitar a colaboração e otimizar o tempo disponível, optamos por adotar uma abordagem filosófica baseada em metodologia ágil. Também permite contato constante com o cliente, podendo receber feedback constante ao longo do desenvolvimento do software
- **Processo:** Este processo combina elementos do SCRUM e do XP. Para as práticas e princípios do SCRUM, referenciamos o ‘Guia do SCRUM’.
- **Ferramentas:** Utilizaremos o GitHub para gerenciar submissões e alterações de código, o Discord para reuniões, que oferece controle sobre interações e é reconhecido por sua segurança, e o Figma para a criação de protótipos, designs e colaboração na interface do usuário. Para manter a comunicação síncrona e assíncrona, utilizaremos o Discord, Teams e WhatsApp.
- **Métodos:** como anteriormente citado, aproveitamos dos métodos provenientes do SCRUM/XP como por exemplo: Pair Programming, Code Review Sprint Daily, Sprint Planning, Sprint Review, Sprint Retrospective e Refinamento de Backlog de Produto. Agendaremos encontros semanais para manter a equipe em sintonia e fomentar uma comunicação eficiente durante nossas atividades.

2.2 Organização do Projeto

Papel	Atribuições	Responsável	Participantes
Desenvolvedor	Escrever os códigos da aplicação. Escrever os códigos de teste. Documentar o código da aplicação	Ana Júlia	Todos do grupo
Dono do Produto	Atualizar o escopo do produto, organizar o escopo das sprints, validar as entregas	Adriano	Todos do grupo
Analista de Qualidade	Definir padrões para a qualidade do produto. Garantir a qualidade do produto, garantir o cumprimento do conceito de pronto, realizar inspeções de código	-	Todos do grupo
Cliente	Verificar se o que foi definido no escopo está sendo entregue	Alisson	Ian

Tabela 02- Tabela de divisão de tarefas do Projeto

2.3 Planejamento das Fases e/ou Iterações do Projeto

Sprint	Produto (Entrega)	Data Início	Data Fim	Entregáveis	Responsáveis	Conclusão
<i>Sprint 0</i>	Definição do produto	24/04/2024	01/05/2024	Definição do escopo do projeto	Todos	100%
<i>Sprint 1</i>	Definição de tecnologias e começo do protótipo de telas no Figma	01/05/2024	08/05/2024	Linguagens e frameworks identificados.	Todos	50%
<i>Sprint 2</i>	Terminar o protótipo, começo da criação do banco de dados e Início da criação das telas	08/05/2024	15/05/2024		Todos	0%
<i>Sprint 3</i>	Término da criação do banco e cadastro do administrador	15/05/2024	22/05/2024		Todos	0%
<i>Sprint 4</i>	CRUD do administrador	22/05/2024	29/05/2024		Todos	0%
<i>Sprint 5</i>	Geração de relatórios	29/05/2024	05/06/2024		Todos	0%
<i>Sprint 6</i>		05/06/2024	12/06/2024		Todos	0%
<i>Sprint 7</i>		12/06/2024	19/06/2024		Todos	0%

Tabela 03- Tabela de sprints do Projeto

2.4 Matriz de Comunicação

Descrição	Área/Envolvidos	Periodicidade	Produtos Gerados
<ul style="list-style-type: none"> Acompanhamento das Atividades em Andamento 	<ul style="list-style-type: none"> Equipe do Projeto 	<ul style="list-style-type: none"> Semanal 	<ul style="list-style-type: none"> Ata de reunião Relatório de situação do projeto
<ul style="list-style-type: none"> Acompanhamento dos Riscos, Compromissos, Ações Pendentes, Indicadores 		<ul style="list-style-type: none"> Quinzenal 	
<ul style="list-style-type: none"> Comunicar situação do projeto 	<ul style="list-style-type: none"> Equipe Prof/Monitor 	<ul style="list-style-type: none"> Semanal 	<ul style="list-style-type: none"> Ata de reunião, e Relatório de situação do projeto

Tabela 04- Tabela de matriz de comunicação.

2.5 Gerenciamento de Riscos

Risco	Grau de Exposição	Mitigação	Plano de Contingência
Trancamento da disciplina por parte de algum membro da equipe	Médio	Manter a equipe motivada, unida e informada.	Reajustar as funções para os membros restantes.
Atraso nas releases	Baixa	Manter o planejamento organizado de forma que não cause mais atraso e transtorno.	Reorganizar as tarefas para que o cronograma do projeto volte à normalidade.
Integração Complexa	Média	Realizar testes rigorosos de integração durante o desenvolvimento.	Manter as partes interessadas informadas sobre quaisquer impactos no cronograma.
Falta de conhecimento técnico do projeto de algum membro da equipe	Alta	Manter os membros sempre informados das tecnologias e métodos utilizados no projeto. Estimular a ajuda dos membros que possuem mais conhecimento na área.	Troca de funções, colocando o membro que possui mais experiência na função. Recomendações com monitor e equilibrar os pares de programadores nas issues.
Mudança de Backlog para entregar releases	Médio	Revisão de backlog constante.	Reajuste das sprints de todo o projeto.

Tabela 05- Tabela de gerenciamento de erros do Projeto

2.6 Critérios de Replanejamento

Mudanças significativas no escopo: caso haja mudanças significativas no escopo que possam afetar os objetivos, entregas ou recursos, o plano de contingência será efetivo, deverá ser replanejado para ajustar o projeto de acordo com as novas especificações.

Atrasos: Em caso de atrasos que possam impactar o cronograma do projeto, faremos revisões para reavaliar e ajustar as datas de entrega, garantindo sua viabilidade.

Mudança de funções entre os membros: caso haja mudanças de função dos membros, enfatizaremos a documentação detalhada, o treinamento cruzado, a comunicação transparente e o replanejamento ágil.

Novos requisitos para o projeto: é necessário uma avaliação dos requisitos adicionais, mantendo uma comunicação clara com a equipe para entender o impacto nos requisitos que já existem e focar na priorização dos novos requisitos com base no valor de agregação e urgência. Também será preciso um ajuste do plano de projeto conforme necessário para acomodar as mudanças.

Desentendimento entre os membros da equipe: torna-se extremamente necessário revisar certas partes do projeto, já que existe a possibilidade de separar membros da equipe que estavam colaborando ou refazer algumas seções do produto. Em casos de desentendimento o conflito será mediado em uma tentativa de resolução por meio da CNV (Comunicação Não Violenta).

3 PROCESSO DE DESENVOLVIMENTO DE SOFTWARE

A equipe optou por adotar abordagens ágeis, como SCRUM e XP, visando uma maior agilidade e responsividade a mudanças durante o processo de desenvolvimento do sistema.

O SCRUM é um framework de gestão de projetos que tem como foco a entrega incremental do produto em interações chamadas *sprints*. Promove a colaboração, adaptação contínua e a auto-organização da equipe. Já o XP é uma metodologia focada na qualidade do software e satisfação do cliente, apoiada em valores como respeito, coragem, feedback, comunicação e simplicidade, bem como visto em práticas como programação em pares, integração contínua, além da comunicação constante entre os membros da equipe e da entrega de releases curtas, a fim de satisfazer o cliente.

Dentro das abordagens escolhidas, utilizaremos métodos como sprints semanais, revisão do código em pares, reuniões mais extensas (checkpoints) e serão elaborados documentos/diagramas para ajudar na organização da equipe.

- Cerimônias do SCRUM:

Atividade	Método	Ferramenta	Entrega
Checkpoint	No lugar das reuniões diárias, optamos por realizar 2 reuniões online por semana de aproximadamente 30 minutos	Discord	Registro de progresso, identificação de problemas e planejamento futuro.
Planning	Reunião online	Discord	Planejamento e quais

			tarefas serão realizadas durante a sprint.
Refinamento	Reunião online	Discord	Estórias de usuário refinadas e com pontuação definida pela equipe usando a escala fibonacci com o objetivo de facilitar a criação de sprints futuras e replanejamento, em casos necessários, de sprints já planejadas.
Retrospectiva	Reunião de análise do que foi feito (final da sprint), abrindo espaço para melhorias.	Discord e Miro	Quadro da retrospectiva.
Daily Scrum	não serão realizadas , tendo em vista conflito de horários dos membros da equipe para a Daily Scrum		

Tabela 06- Tabela de cerimônias do SCRUM que serão implementados no decorrer do projeto.

- Práticas do XP utilizadas:

Prática XP	Motivo
Pair programming	Visa a qualidade do código, o aprendizado colaborativo e reduz erros, pois envolve dois programadores trabalhando juntos e simultaneamente.
Releases pequenos	Tem como objetivo entregar valor ao cliente em incrementos menores, proporcionando uma abordagem mais interativa e adaptativa, além de permitir uma resposta ágil às mudanças e requisitos do cliente.
Integração contínua	Permite verificar se novas funcionalidades não criaram novos defeitos ao projeto.
Padrões de codificação	Facilita a compreensão e organização de código.
Refatoramento	Permite manter o código limpo, eficiente e de fácil manutenção, sem alterar a sua funcionalidade.
Testes funcionais	Importante processo a fim de garantir a qualidade e adesão aos requisitos.
Jogo do planejamento	Não será utilizado , pois julgamos ser uma prática que demanda um tempo considerável, tendo em vista o tempo limitado que temos com o cliente.
Cliente no local	Não será utilizado devido a indisponibilidade do cliente para que essa prática seja realizada.

Tabela 07- Tabela de práticas do XP que serão usufruídas no decorrer do projeto.

4 DECLARAÇÃO DE ESCOPO DO PROJETO

4.1 Backlog do produto

O backlog do produto é uma lista dinâmica de todos os requisitos, funcionalidades, correções e melhorias que precisam ser feitas em um produto. O backlog é constantemente revisado e atualizado com o decorrer do tempo, novos itens serão adicionados, prioridades serão ajustadas e detalhes sendo aprimorados à medida que o entendimento do produto evolui. O backlog do produto está detalhado na tabela 10 no item 4.4.

4.2 Perfis

Tabela : Perfis de acesso

#	Nome do perfil	Características do perfil	Permissões de acesso
<1>	Administrador	O administrador tem acesso a todas as áreas da aplicação, pode acessar e editar fichas de pessoas, gerar relatórios, gerir os afastamentos, cadastrar outro administrador e gerir escalas.	Acessar e editar fichas de pessoas, que contém nome, endereço, telefone, curso, tipo de trabalho. Gerar relatórios de afastamentos, abono, contingente e férias. Administrar escalas de trabalho.
<2>	Usuário (Leitor)	Apenas ler informações inseridas pelos administradores.	Leitura.

Tabela 08 - Tabela de perfis de acesso do projeto.

4.3 Cenários

Tabela : Cenários funcionais

Sistema: xxx – Cenários funcionais		
Numeração do cenário	Nome do cenário	Sprints

Tabela 09 - Tabela de cenários

4.4 Tabela : Backlog do produto

Sistema: xxxx – Backlog do produto						
Numeração (Cenário / requisito)	Sprint	Nome do requisito	Tipo de requisito (Funcional / não funcional)	Priorização do requisito Must, Should, Could	Descrição sucinta do requisito	User stories (U.S.) associadas Identifique as U.S. associadas ao requisito
01	sprint 02		Funcional	Must	O administrador deve ser capaz, primeiramente, de implementar as informações relativas aos funcionários que serão persistentes no programa.	Como um administrador quero cadastrar funcionários no banco de dados.
02	sprint 04		Funcional	Should	A fim de implementar uma das principais funções da aplicação , que é de gerar diversos tipos de relatório, o administrador deve ser capaz de interagir com o banco de dados da forma que melhor lhe convir.	Como administrador, quero editar, visualizar e excluir funcionários do banco de dados.
03	sprint 05		Funcional	Could	O relatório de escalas faz uma busca nos funcionários e verifica quais estão de férias, quais estão de abono ou afastados e quais estão aptos a trabalhar, se o funcionário não têm nenhuma restrição e está em horário de trabalho, ele pode ser escalado.	Como um administrador quero gerar/acessar relatórios de escala

04	sprint 05		Funcional	Could	A ficha de pessoas dá uma visualização geral de todo o quadro de pessoal, mostrando informações como nome, telefone, cursos, escalas e outras informações gerais.	Como um administrador quero gerar/acessar fichas de pessoas.
05	sprint 3		Funcional	Must	Para começar a cadastrar funcionários e gerar relatórios o administrador deverá estar devidamente logado, a fim de obter os acessos.	O administrador deve ser capaz de se cadastrar no sistema e atualizar suas informações.
06	sprint 3		Não funcional	Must	Os administradores têm exclusividade em qualquer tipo de operação de escrita "W" de dados. Outros possíveis usuários são apenas leitores.	O ambiente deve ser seguro e todas as atribuições de edição de dados são exclusivas do administrador
07	sprint 4		Funcional	Should	Os usuários não podem editar nenhum tipo de informação no sistema.	Como usuário devo ser capaz de acessar as informações e relatórios que eu tiver acesso.
08	sprint 4		Funcional	Could	O administrador tem total acesso ao sistema, podendo inclusive, atribuir acessos de leitura aos usuários.	Como administrador devo poder atribuir diferentes formas de acesso aos arquivos e relatórios gerados

Tabela 10- Tabela detalhada das sprints do projeto.

5 REFERÊNCIAS BIBLIOGRÁFICAS

1. RABELLO, Guilherme. **Diagrama de Ishikawa: o que é e como fazer [Com template]** 04/03/2024. Disponível em:
<<https://www.siteware.com.br/blog/metodologias/diagrama-de-ishikawa/>>
Acesso em: 07 maio 2024
2. Grupo Cygni. *Miro Mind Maps*. Disponível em: [Miro](#)
3. AUDY, Jorge. *Scrum 360: Um guia completo e prático de agilidade*. São Paulo: Casa do Código, 2015.