

Documentação do Sistema

ToDoList Diretoria

Sistema web para rastreamento de atividades diárias

Equipe: Amanda Carvalho de Andrade

Antônio Alves Neto

José Luiz Melo de Carvalho

Nadjane Ferreira Santos

SUMÁRIO

DEFINIÇÃO DO SISTEMA	2
1.1 PÚBLICO-ALVO	2
REQUISITOS DO SISTEMA	3
REQUISITOS FUNCIONAIS	3
REQUISITOS NÃO-FUNCIONAIS	3
DESCRIÇÃO DOS CASOS DE USO	3
GESTÃO DO PROJETO	12
5.1 ESTIMATIVA E PLANEJAMENTO DO MVP	13
5.2 GESTÃO DAS PESSOAS/TIME	14
5.3 METODOLOGIA DE DESENVOLVIMENTO	16
5.4 GESTÃO DE RISCOS	17
5.5 GESTÃO DA QUALIDADE DO PRODUTO	20
5.5 GESTÃO DA ENTREGA DO PRODUTO	20

1. DEFINIÇÃO DO SISTEMA

O projeto ToDoList Diretoria consiste numa aplicação web para rastreamento de atividades do dia a dia, uma *to do list* (lista de tarefas), com aplicação de sistema de premiações, por meio da gamificação do processo como forma de engajar o usuário com o sistema. A proposta inicial é desenvolver um sistema de fácil uso, com cadastro de tarefas, divididas por categorias, que também podem ser cadastradas pelo usuário. Cada tarefa pode ter um nível de prioridade. De acordo com a prioridade e categoria, o usuário ganha pontos e, ao concluir todo um ciclo de atividades, recebe um badge que pode ser compartilhado nas redes sociais.

O sistema tem dois papéis de usuário: grátis e pagante. Esse último tem acesso a todas as funcionalidades do sistema, tais como: cadastro de tarefas e categorias ilimitados, todos os badges e sem publicidades. Enquanto a versão grátis os usuários têm um limite diário de cadastro de 10 tarefas, três novas categorias e três badges.

A aplicação terá uma tela inicial de login, no qual pode ser feito a partir da integração com as principais redes sociais (Facebook, Instagram, LinkedIn), assim como link para cadastro de novo usuário (por redes sociais ou por preenchimento de formulário). A página do usuário conta com as seções Cadastrar e Buscar, além de um painel de gestão (*dashboard*) das atividades. Na opção Cadastrar, o usuário pode inserir nova tarefa ou nova categoria. Já em Buscar, pode-se pesquisar por tarefa, categoria e *badges* (selos das premiações).

Nos próximos tópicos serão explicitados os requisitos do sistema, as regras de negócios, os casos de uso e a documentação de Gestão do Projeto.

1.1 PÚBLICO-ALVO

Todas as pessoas que querem manter rastreamento das atividades cotidianas que devem ser realizadas em um determinado período.

2. REQUISITOS DO SISTEMA

2.1 REQUISITOS FUNCIONAIS

RF01 – Cadastrar Usuário

RF02 – Fazer login

RF03 – Gerenciar Tarefa

RF03 – Gerenciar Categoria

RF04 – Gerenciar Redes Sociais

RF05 – Gerenciar *Badges*

2.2 REQUISITOS NÃO-FUNCIONAIS

RNF01 – A interface com o usuário deve ser responsiva, se ajustando aos diversos tamanhos de dispositivo sem perder a legibilidade.

RNF02 – A área de usuário deve solicitar login e senha, essa criptografada, para evitar invasões e modificações por terceiros no Sistema.

RNF03 – O tempo de resposta de uma tela deve ser menor ou igual a 5 segundos em 95% dos casos.

RNF04 – O Banco de Dados utilizado será o MySQL que, apesar de gratuito, oferece grande confiabilidade.

RNF05 – As linguagens de programação será PHP e JavaScript e devem ser compatíveis com qualquer sistema.

3. DESCRIÇÃO DOS CASOS DE USO

UC01 – Cadastrar Usuário

Atores: Usuário

Descrição: O usuário tem a opção de fazer um cadastro no próprio sistema ou por suas redes sociais. O cadastro terá como campos: Nome, E-mail e Senha. Após cadastrado, o usuário tem acesso às funcionalidades do sistema de acordo com o seu plano (gratuito ou pago).

Descrição Detalhada:

Fluxo Principal	
Ator	Sistema
1. Usuário clica no botão “Cadastrar”.	2. Sistema abre a tela de Cadastro, com botões das principais redes sociais para cadastro por meio de seus perfis pessoais, e um formulário de cadastro.
3.1 Usuário preenche todos os campos obrigatórios e clica em “Cadastrar”. 3.2 Usuário clica no botão de alguma rede social.	4.1 Sistema salva os dados no Banco de Dados do Sistema, verificando a RN1. Sistema exibe M3 e, em seguida, retorna à tela principal para login. 4.2 Sistema abre a tela pop-up da integração da rede social selecionada e capta os dados necessários para cadastro (nome, email e token). Sistema verifica a RN2. Sistema exibe M3 e, em seguida, retorna à tela principal para login.
Fluxo de Exceção I	
Ator	Sistema
	2. Sistema exibe M1 e retorna ao passo 3 do Fluxo Principal.
Fluxo de Exceção II	
Ator	Sistema
	2. Sistema exibe M2 e retorna ao passo 3 do Fluxo Principal.

Regras de Negócio

RN1: Os campos obrigatórios devem ser preenchidos.

RN2: Não pode cadastrar o mesmo e-mail/perfil de rede social mais de uma vez.

Mensagens do Sistema

M1: “Este campo é obrigatório e deve ser preenchido”.

M2: “Este e-mail/perfil já está cadastrado. Por favor, insira um diferente.”.

M3: “Cadastro realizado com sucesso!”

UC02 – Fazer Login no Sistema

Atores: Usuário

Descrição: Para ter acesso às funcionalidades é preciso logar no sistema, informando os dados corretamente.

Descrição Detalhada:

Fluxo Principal	
Ator	Sistema
1. Usuário entra na aplicação.	2. Sistema abre a tela de Login.
3.1 Usuário preenche os campos obrigatórios de login e senha e clica em “Entrar”.	4. Sistema verifica os dados no Banco de Dados do Sistema, observando a RN1 e a RN2. Sistema exibe tela Principal da aplicação.
3.2 Usuário clica no botão da rede social cadastrada.	
Fluxo de Exceção I	
Ator	Sistema

	2. Sistema exibe M1 e retorna ao passo 3 do Fluxo Principal.
Fluxo de Exceção II	RN2 não foi atendida
Ator	Sistema
	2. Sistema exibe M2 e retorna ao passo 3 do Fluxo Principal.
Fluxo de Exceção III	RN3 não foi atendida
Ator	Sistema
	2. Sistema exibe M3 e retorna ao passo 3 do Fluxo Principal.

Regras de Negócio

RN1: Os campos obrigatórios devem ser preenchidos.

RN2: É preciso se cadastrar para ter acesso a sua lista de tarefas!

Mensagens do Sistema

M1: “Este campo é obrigatório e deve ser preenchido”.

M2: “Este e-mail/perfil não está cadastrado. Por favor, insira outro e-mail”.

M3: “E-mail e senha não conferem”.

**Caso o usuário tenha esquecido a senha, deve solicitar uma nova no botão “Esqueci minha senha”.*

UC03 – Gerenciar Tarefa

Atores: Usuário

Descrição: O usuário pode gerenciar suas tarefas: criar, pesquisar, editar e apagar. O formulário de tarefa tem os campos: título, status, descrição, categoria, prioridade e prazo final.

Descrição Detalhada:

Fluxo Principal	
Ator	Sistema
1. Usuário faz login no sistema.	2. Sistema abre a tela de dashboard, com as tarefas salvas separadas em cards de acordo com seus atributos.
Fluxo de Exceção I	
Ator	Sistema
1. Usuário seleciona a opção Tarefa, no menu Cadastrar.	2. Sistema exibe a tela de inserção de tarefa.
3. Usuário preenche todos os campos necessários. Depois clica em Cadastrar.	4. Sistema salva os dados no Banco de Dados do Sistema, exibe M2 e abre a página de busca com as tarefas cadastradas.
Fluxo de Exceção II	
Ator	Sistema
1. Usuário seleciona a opção Tarefas no menu Buscar.	2. Sistema exibe a página com a lista das tarefas salvas.
3. Na lista de tarefas, o usuário clica no ícone de Alterar (lápis).	4. Sistema abre formulário de cadastro da tarefa, com as informações salvas no banco de dados.
5. O usuário altera os campos desejados e depois clica em Cadastrar.	6. Sistema exibe M3, salva os dados no Banco de Dados do Sistema e apresenta a página de busca.
Fluxo de Exceção III	
Apagar Postagem	

Ator	Sistema
1. Usuário seleciona, na lista de tarefas, o ícone Apagar (lixeira).	2. Sistema exibe a M1.
3. Usuário pode clicar em “Sim” ou “Não”.	4.1 Se a resposta for “Sim”, o Sistema apaga a tarefa do Banco de Dados do Sistema e retorna à tela de dashboard. 4.2 Se a resposta foi “Não”, o Sistema retorna à tela de dashboard.

Mensagens do Sistema

M1: “Tem certeza que deseja apagar essa tarefa?”.

M2: “Tarefa salva com sucesso!”

M3: “Tarefa atualizada com sucesso!”

UC04 – Gerenciar Categoria

Atores: Usuário

Descrição: O usuário pode personalizar as categorias, cadastrando-as de acordo com as necessidades.

Descrição Detalhada:

Fluxo Principal	
Ator	Sistema
1. Usuário faz login no sistema.	2. Sistema abre a tela de dashboard, com as tarefas salvas separadas em cards de acordo com seus atributos.
Fluxo de Exceção I	
Ator	Sistema

1. Usuário seleciona a opção Categoria, no menu Cadastrar.	2. Sistema exibe a tela de inserção de categoria.
3. Usuário preenche todos os campos necessários. Depois clica em Cadastrar.	4. Sistema verifica a RN1: 4.1 Se não há conflito, salva os dados no Banco de Dados do Sistema e apresenta a categoria salva; 4.2 Se há conflito, exibe a M4 e permanece na tela de cadastro.
Fluxo de Exceção II	Alterar Categoria
Ator	Sistema
1. Usuário seleciona a opção Categorias no menu Buscar.	2. Sistema exibe a página com a lista das categorias salvas.
3. Na lista de categorias, o usuário clica no ícone de Alterar (lápis).	4. Sistema abre formulário de cadastro da categoria, com as informações salvas no banco de dados.
5. O usuário altera os campos desejados e depois clica em Cadastrar.	6. Sistema verifica RN1. 6.1 Se não há conflito, exibe M3, salva os dados no Banco de Dados do Sistema e apresenta a página de busca. 6.2 Se há conflito, exibe a M4 e permanece na tela de cadastro.
Fluxo de Exceção III	Apagar Categoria
Ator	Sistema
1. Usuário seleciona, na lista de categorias, o ícone Apagar (lixeira).	2. Sistema exibe a M1.
3. Usuário pode clicar em “Sim” ou “Não”.	4.1 Se a resposta for “Sim”, o Sistema apaga a categoria do Banco de Dados do Sistema e retorna à tela de dashboard.

	4.2 Se a resposta foi “Não”, o Sistema retorna à tela de dashboard.
--	---

Mensagens do Sistema

M1: “Tem certeza que deseja apagar essa categoria?”.

M2: “Categoria salva com sucesso!”

M3: “Categoria atualizada com sucesso!”

M4: “Já existe categoria com este nome!”

Regras de Negócio

RN1: Na seção categoria, o Sistema não aceitará categorias com nomes iguais.

UC05 – Gerenciar Redes Sociais

Atores: Usuário

Descrição: O usuário pode gerenciar as redes sociais as quais ele deseja integrar à aplicação.

Descrição Detalhada:

Fluxo Principal	
Ator	Sistema
1. Usuário faz login no sistema.	2. Sistema abre a tela de dashboard, com as tarefas salvas separadas em cards de acordo com seus atributos.
Fluxo de Exceção I	
Ator	Sistema

1. Usuário seleciona a opção Configurações ao clicar na seção de perfil, no canto superior direito.	2. Sistema exibe a tela configurações do perfil.
3. Usuário pode adicionar as redes sociais a partir do botão relativo a cada uma.	<p>4. Sistema verifica RN1.</p> <p>4.1 Se as informações estão corretas, salva os dados necessários no banco de dados. Exibe M2 e retorna a tela de configuração.</p> <p>4.2 Se as informações não estão corretas, sistema exibe M4 e retorna à tela de configuração.</p>
Fluxo de Exceção II	Apagar Rede Social
Ator	Sistema
1. Usuário clica no ícone de apagar (lixeira) ao lado da rede social desejada.	2. Sistema exibe a M1.
3. Usuário pode clicar em “Sim” ou “Não”.	<p>4.1 Se a resposta for “Sim”, o Sistema apaga o perfil do Banco de Dados do Sistema e retorna à tela de configuração.</p> <p>4.2 Se a resposta foi “Não”, o Sistema retorna à tela de configuração.</p>

Mensagens do Sistema

M1: “Tem certeza que deseja apagar esse perfil?”.

M2: “Perfil salvo com sucesso!”

Regras de Negócio

RN1: O Sistema não pode salvar o mesmo perfil de rede social mais de uma vez.

UC05 – Gerenciar *Badges*

Atores: Usuário

Descrição: O usuário pode gerenciar os *badges* ganhados no decorrer da realização das tarefas. Ele poderá acessar a lista de *badges* e compartilhar nas suas redes sociais.

Descrição Detalhada:

Fluxo Principal	
Ator	Sistema
1. Usuário faz login no sistema.	2. Sistema abre a tela de dashboard, com as tarefas salvas separadas em cards de acordo com seus atributos.
Fluxo de Exceção I	
Ator	Sistema
1. Usuário seleciona a opção Badges, no menu Buscar.	2. Sistema exibe a tela com a lista de badges do usuário.
Fluxo de Exceção II	
Ator	Sistema
1. Usuário seleciona, na lista de badges o ícone de compartilhar (seta para cima).	2. Sistema exibe a M1 com os botões das redes sociais disponíveis para compartilhamento.
3. Usuário clica no botão da rede social desejada.	4. Sistema abre a página de postagem da rede social escolhida, com um post default do badge.

Mensagens do Sistema

M1: “Oba! Selecione a rede social que você deseja compartilhar sua conquista!”

4. GESTÃO DO PROJETO

Este tópico apresentará a documentação da gestão do projeto, as estratégias que serão utilizadas para o desenvolvimento e entrega de um Produto Mínimo Viável (*Minimum Viable Product* - MVP). No primeiro ponto, abordaremos a estimativa e planejamento do MVP, em seguida serão apresentadas as práticas que serão adotadas para o gerenciamento das pessoas, seus papéis, responsabilidades e tarefas. Também será detalhada a metodologia de desenvolvimento, assim como projetos para gestão dos riscos, de qualidade e para a entrega do produto.

5.1 ESTIMATIVA E PLANEJAMENTO DO MVP

Neste tópico será apresentado um recorte das principais características da aplicação a serem trabalhadas e entregues como produto inicial aos usuários. Detalharemos o plano de desenvolvimento do Produto Mínimo Viável (MVP) e a estimativa de entrega.

A proposta do ToDoList Diretoria é oferecer uma opção atrativa para rastreamento das tarefas que devem ser realizadas em um determinado espaço de tempo, de forma a engajar e manter o usuário na aplicação. Para tanto, utilizaremos a gamificação na lista de tarefas, com sistema de pontuação por atividades realizadas e, ao final, premiação com *badges* para compartilhamento em redes sociais. Dessa forma, os usuários não esquecerão de fazer as suas atividades e, ainda, terão mais incentivo para realizar aquelas que, até então, pareciam impossíveis de serem concluídas.

Para o MVP, o sistema de gamificação estará funcionando em seu formato inicial: ao cadastrar uma tarefa, o usuário deverá indicar a categoria e prioridade. Esses dois campos determinarão o tipo de *badge* (por categoria) e o valor a ser pontuado (prioridade). A cada tarefa realizada, o usuário acumulará pontos e, ao atingir um determinado valor, será premiado com um *badge*.

O produto inicial estará com as funcionalidades propostas no tópico dos requisitos do sistema implementadas, a lembrar: realizar login no sistema, com a opção de realizá-lo com as credenciais de redes sociais; gerenciar tarefas, categorias, redes sociais e *badges*. Para o gerenciamento do projeto, a equipe usará os métodos ágeis Scrum e

Kanban, com algumas técnicas propostas pelo *Extreme Programming* (XP) para desenvolvimento da aplicação.

Na reunião de abertura, a equipe transformará os requisitos em histórias do usuário para, assim, avaliar a necessidade de dividi-las em tarefas menores, determinar o peso de cada uma, por meio da técnica *planning poker*, prevista no método XP, e o tempo para executá-las. A equipe determinará a prioridade de cada história para formação do backlog do produto. Também serão divididos os papéis que cada membro desempenhará no decorrer da sprint, conforme está descrito no tópico seguinte.

A expectativa é que a equipe termine a reunião com os artefatos para gerenciamento do projeto prontos: backlogs do produto e da sprint, time-box e os quadros de *burndown*, Scrum e Kanban.

A proposta inicial é que, a cada semana, a equipe entregue uma funcionalidade do sistema já testada. Para a implementação e testes da gamificação, a expectativa de duração é de duas semanas, sendo a única sprint quinzenal. O aumento do tempo da sprint se deve à complexidade do sistema de premiação, com suas características singulares e alta probabilidade de ocorrerem mudanças.

Dessa forma, a expectativa é que a aplicação esteja pronta para uso dos usuários beta em dois meses. No próximo tópico, serão detalhados os métodos de gestão de pessoas que serão adotados para o desenvolvimento da aplicação.

5.2 GESTÃO DAS PESSOAS/TIME

A gestão de pessoas é de suma importância e pode determinar o sucesso ou fracasso de um projeto. Por isso, se faz necessário ter um gerenciamento de recursos humanos para o melhor desempenho das atividades e tem papel de potencializar os resultados do negócio. Este tópico apresentará quais as boas práticas e atividades serão adotadas para gerenciamento do projeto e das pessoas (papéis, responsabilidades, tarefas). Como exposto anteriormente, a gestão do time será feita com base nos métodos ágeis Scrum e Kanban. Para gerir o time, será usada a ferramenta matriz de responsabilidade também conhecida como matriz RACI.

A matriz RACI tem como principal objetivo a atribuição de funções e responsabilidades dentro de um processo ou projeto facilitando o gerenciamento dos recursos humanos na organização. RACI é o acrônimo para Responsável, Aprovador, Consultado e Informado (*Responsible, Accountable, Consulted e Informed*). Assim, cada colaborador toma consciência de suas atribuições durante um projeto. Ou seja, é uma matriz de distribuição de responsabilidades.

A seguir, estão dispostos os papéis atribuídos pela matriz:

- Responsável (*Responsible*): é o papel responsável por completar as tarefas e as entregas. Por exemplo, caso o projeto for criar um novo blog e produzir conteúdo para ele, então os redatores, os designers e os programadores serão os responsáveis pela execução e pela entrega;
- Aprovador (*Accountable*): é quem tem a autoridade final sobre a aprovação do projeto. No caso acima, seria o papel do gerente de projeto;
- Consultado (*Consulted*): é o papel de quem é consultado, dentro ou fora da sua empresa, para que possa contribuir para a execução das tarefas. Alguém cuja participação agrega valor e/ou é essencial para a implementação final. Nesse caso, a comunicação é de duas vias (consulta \Leftrightarrow resposta);
- Informado (*Informed*): são clientes, stakeholders ou quaisquer pessoas que devem ser atualizadas sobre o andamento do projeto. Elas são notificadas de resultados ou ações tomadas, mas sem necessariamente estarem envolvidas no processo de tomada de decisão. A comunicação, nesse caso, ocorre num sentido.

A utilização desta ferramenta traz como benefício a clareza e objetividade das atividades e o papel que cada recurso humano representa dentro do projeto. Nas figuras abaixo, estão exemplos das tabelas de controle de atividades e da divisão por papéis.

MATRIZ RACI - ToDoList Diretoria			
Controle de Atividades			
Atividade	Data inicial	Data final	Status
Estudo de viabilidade	01/11/2020	03/11/2020	Concluído
Definição de escopo	03/11/2020	5/11/2012	Concluído
Levantamento de funcionalidades	09/11/2020	12/11/2012	Concluído
Planejamento do projeto	03/11/2020	12/11/2012	Em andamento
Funcionalidades do software	03/11/2020	12/11/2012	Em andamento
Estudo de viabilidade tecnologica	01/11/2020	03/11/2020	Concluído
Desenvolver o sistema	12/11/2012	12/12/2012	Em andamento
Documentar o sistema	12/11/2012	14/11/2012	Atrasado
Testar o sistema	30/11/2020	6/12/2020	Não iniciado
Validar versão final	7/12/2020	7/12/2020	Não iniciado
Implantar	15/12/2020	18/12/2020	Não iniciado
Treinar	19/12/2020	22/12/2020	Não iniciado

Figura 1 - Controle de atividades

MATRIZ RACI - ToDoList Diretoria				
Legenda:	R- Responsável - Execução e entrega da atividade		A- Aprovador - Autoridade final de tomada de decisão	C- Consultado - Agrega valor e deve ser incluído na atividade
	I- Informado - Clientes e outros que devem ser atualizados			
Atividade	Gerente de Projetos	Scrum Master	Desenvolvedor	
Estudo de viabilidade	R	A	I	I
Definição de escopo	R	A	I	I
Levantamento de funcionalidades	R	A	I	I
Planejamento do projeto	R	A	C	I
Funcionalidades do software	R	A	C	I
Estudo de viabilidade tecnologica	R	A	I	I
Desenvolver o sistema	C	A	R	
Documentar o sistema	I	C	R	
Testar o sistema	C	A	R	
Validar versão final	A	R	C	
Implantar	I	R	A	
Treinar	I	A	R	

Figura 2 - Atividade por papel

5.3 METODOLOGIA DE DESENVOLVIMENTO

Este tópico apresentará quais as boas práticas/atividades serão adotadas para o desenvolvimento do MVP da ToDoList Diretoria. A flexibilidade será o carro chefe para o processo de desenvolvimento, fazendo-se uso de uma metodologia única que se adapte às necessidades do usuário e disponibilize um produto/software inovador e inteligente.

A proposta do gerenciamento de desenvolvimento é de direcionar o time para evitar erros, reduzir os prazos, alinhar os processos conforme as metas vão sendo definidas e avaliar as ações tomadas no fim de cada etapa para decidir em quais cenários se adaptam melhor cada processo.

Para ficar atento a problemas, gargalos operacionais e pontos que necessitam de melhorias constantemente se fará como prática a análise de métricas por meio de indicadores de desempenho, desta forma também ficará evidente como o time está trabalhando, diversos parâmetros de performance serão criados para que se possa avaliar o número de erros no código fonte, frequência com que esses erros ocorrem e o desempenho da equipe quanto aos prazos.

Pensando na escolha de uma metodologia de desenvolvimento mais adequada, fez-se um mapeamento dos riscos, dos indicadores, dos objetivos, das funcionalidades básicas, dos padrões de segurança e dos padrões de controle. Pela característica da aplicação, optou-se por adotar a metodologia ágil *Extreme Programming* (XP), que é específica para desenvolvimento de software e recomendada para os sistemas cujos requisitos estão sujeitos a mudanças.

A equipe irá levar em consideração as práticas propostas pelo XP, como planejamento das releases, história do usuário, *planning poker*, desenvolvimento dirigido por testes (TDD), integração contínua, usando a ferramenta Jenkins, e métricas de gerenciamento do projeto. O objetivo é ter ciclos curtos de desenvolvimento, com a equipe engajada na entrega e teste das funcionalidades. Como o desenvolvimento das funcionalidades são independentes, a equipe vai implementá-las simultaneamente.

O método XP prevê a elaboração das histórias de usuários, escritas em cartões, que ficarão expostas nos quadros de gerenciamento. Para cada história, a equipe indicará o peso de cada uma (*story points*), por meio de *planning poker*. Ao final, será definida a quantidade de stories points que cada um consegue implementar por iteração, cujo prazo proposto inicialmente é de uma semana.

Para gerenciar as atividades e melhor distribuir as rotinas dentro do time, iremos utilizar uma ferramenta chamada Redmine, através dela é possível criar as tarefas e atribuir aos responsáveis no projeto, podendo-se acompanhar facilmente o desempenho e produtividade dos profissionais nas entregas de cada etapa, isso irá causar impacto positivo e possibilitar ações quando detectado problemas nos resultados.

Desta forma, pretendemos obter uma rotina de gerenciamento de desenvolvimento de software mais rápida e prática, será possível criar um software bem estruturado por meio de testes direcionados com uma estrutura correta para o uso ao qual será feito.

5.4 GESTÃO DE RISCOS

Neste tópico abarca a temática da gestão de riscos e como eles serão geridos, mitigados ou respondidos, assim como a sua classificação. Na tabela abaixo, estão descritos os principais riscos no desenvolvimento deste projeto.

1 - CAT	2 - TIPO	3 - DESCRIÇÃO	4 - PROB	5 - EXPECTATIVA	6 - IMPACTO	7 - ESTRATÉGIA / RESPOSTA
Intelec-tual	N	Imperícia no domínio de alguma tecnologia utilizada	MÉDIA	Possibilidade de um ou mais membros encontrarem dificuldade no uso de alguma tecnologia utilizada	ALTO - Afeta negativamente a velocidade do desenvolvimento.	Adotar tecnologias onde os membros tenham familiaridade. Caso surjam dificuldades, buscar solucioná-las através de cursos rápidos ou consulta com especialistas.
Pessoal	N	Infortúnios pessoais	MÉDIA	Possibilidade de um membro (ou familiar) adoecer. Bem como, qualquer infortúnio de natureza de pessoal (ex.: o telhado da casa desmoronou)	ALTO - Não é possível mensurar. Logo, estima-se o pior dos cenários.	Adotar alternativas ágeis para solucionar o problema, assim que este for de conhecimento da equipe.
Intelec-tual	N	Uso de tecnologia pouco familiar a equipe	BAIXA	Possibilidade do uso de tecnologia pouco familiar a equipe. Neste cenário, a necessidade do uso da tecnologia não havia sido prevista na fase de projeto.	ALTO - Afeta negativamente a velocidade do desenvolvimento.	Realização de cursos rápidos ou consulta com especialistas.
Projeto	-	Adição de requisitos ao sistema	MÉDIA	Possibilidade de adição de requisitos ao sistema em função de melhorias, segurança, escalabilidade etc.	ALTO - Não é possível mensurar. Logo, estima-se o pior dos cenários.	Adotar alternativas ágeis, a fim de diminuir o possível impacto negativo no projeto.

Projeto	N	BUGs em produção	ALTA	Possibilidade de surgimento de BUGs quando o sistema estiver em produção. Geralmente, esses BUGs são reportados pelos usuários.	ALTO - Atividade para correção de BUGs no sistema em produção tem prioridade em relação às demais, e, deve ser resolvida o quanto antes.	Identificar e corrigir o problema o quanto antes. Até, mesmo, fora do horário de expediente, sempre que houver possibilidade.
Projeto	N	Análise de riscos incompleta	ALTA	Possibilidade de algum risco não ter sido identificado	ALTO - Não é possível mensurar quais riscos ficaram fora da gestão de riscos. Logo, estima-se o pior dos cenários.	Ao identificar o risco, documentá-lo e definir um plano de contingência o quanto antes.
Projeto	N	Má distribuição de tempo	ALTA	Durante alguma fase de planejamento pode haver uma má distribuição do tempo para o desenvolvimento de alguma atividade	ALTO - Afeta negativamente a velocidade do desenvolvimento.	Realizar uma distribuição de tempo para as atividades de modo que a maior parte dos stakeholders envolvidos estejam de acordo.
Intelectual	P	Familiaridade com desenvolvimento web	ALTA	Todos os membros da equipe possuem alguma familiaridade com o desenvolvimento de aplicações web.	ALTO - Afeta positivamente a velocidade do desenvolvimento.	Seguir programando feliz :)
Pessoal	P	Trabalho em horários além do estimado	ALTA	Possibilidade dos membros trabalharem em horários além do estimado, quando necessário. Ex.: madrugada.	ALTO - Afeta positivamente a velocidade do desenvolvimento.	Entregar as atividades o quanto antes.

Legenda:

1. CATEGORIA: intelectual, pessoal e projeto
2. TIPO: negativo / positivo
3. DESCRIÇÃO: breve descrição do risco
4. PROBABILIDADE: baixa, média, alta
5. EXPECTATIVA: descrição da expectativa em relação ao risco
6. IMPACTO: baixo, médio, alto
7. ESTRATÉGIA / RESPOSTA AO RISCO

5.5 GESTÃO DA QUALIDADE DO PRODUTO

Para garantir uma melhor qualidade do produto/software que será produzido, será feito um controle rigoroso de testes de qualidade. Este tópico apresentará quais as boas práticas e atividades que serão adotadas para garantir a qualidade da aplicação.

Alguns pontos serão observados para garantir a qualidade da aplicação, dentre eles: a redução de defeitos, que pode ser controlada a partir de um conjunto de atividades/testes que detectem erros, dos mais simples aos graves, mitigando riscos e garantindo a entrega de um sistema confiável ao usuário final.

A prática de realizar diferentes tipos de teste irá proporcionar a confiabilidade e satisfação dos usuários, pontos de extrema importância para garantir consistência, estabilidade e o sucesso da aplicação.

A redução dos custos também fará parte das metas de qualidade. Espera-se alcançar isso ao realizar diversos testes de forma a detectar erros, evitando gastos de recursos com correções a longo prazo. Para garantir a produtividade e redução do tempo de entrega, serão trabalhados os ciclos de teste em paralelo com os de desenvolvimento.

Para essa aplicação, serão feitos testes automatizados utilizando a ferramenta *open source* Selenium que, por meio de scripts, realizará as verificações indispensáveis da aplicação. A vantagem dessa ferramenta é a possibilidade de testar o sistema em vários browsers, garantindo que a aplicação funcione em qualquer um da escolha do usuário.

5.5 GESTÃO DA ENTREGA DO PRODUTO

Neste tópico, serão apresentadas as boas práticas e atividades que a equipe adotará para entregar o sistema aos usuários. A aplicação será lançada em plataforma web,

compatível com os principais browsers do mercado, para alguns usuários beta que se cadastraram na *landing page*.

A ideia é usar essa amostra menor de usuários para analisar o comportamento de cada um deles, como estão usando as funcionalidades, identificar as dificuldades, o grau de engajamento e tempo passado na aplicação, e principalmente o funcionamento da gamificação. Como diferencial da ToDoList Diretoria, é de extrema importância que o sistema de premiações funcione adequadamente e que promova engajamento para o uso da aplicação.

Os usuários beta terão acesso a todas as funcionalidades do sistema, sem precisar pagar para isso. O fluxo de renda será feito a partir da segunda versão, aberta para todo o público. É nessa versão que será aplicada a restrição de usuário gratuito/usuário pagante. A expectativa é de que esta versão esteja com os possíveis bugs e problemas identificados e sanados.

Pensando na otimização do processo de entrega, se faz necessário adotar alguns padrões que vão tornar esse capítulo mais leve e dinâmico. Como citado anteriormente, o tempo de duração das sprints serão de quinze dias, desta forma, evitando aprovações longas e processos burocráticos.

Estabelecer um ciclo regular de entrega de software para dar maior previsibilidade e organização na gestão dos projetos da TI. Para tal, podemos destacar as seguintes ações:

- Maior confiança ao cliente de que tudo está dentro dos prazos acordados previamente;
- Possibilidade de discutir com maior intensidade sobre os testes não funcionais de que o software possa necessitar;
- Condições para criar uma rotina que facilite o trabalho das equipes de desenvolvimento, infra, qualidade, comercial, entre outros.

Portanto, após a primeira entrega do MVP, as atividades que serão realizadas são: analisar o comportamento do usuário, tempo gasto na aplicação, uso da ferramenta de gamificação e engajamento. Identificar bugs e problemas do software, corrigindo-os e realizando testes das funcionalidades. Verificar se a aplicação atendeu as necessidades do usuário e se ofereceu aquilo que eles realmente precisavam.

A gestão da equipe para a manutenção da aplicação será feita utilizando a mesma metodologia aplicada ao desenvolvimento do MVP, exposta nesse documento. A diferença é que as sprints serão quinzenais.