
Plano de Projeto MediControl

Versão 1.0

Preparado por:

Antonio Neto

Beatriz Falcão

Danilo Melo

Hermano Costa

Thamires Lopes

Julho/2019

Histórico de Revisões

Nome	Data	Motivos da Mudança	Versão

Sumário

1. VISÃO GERAL DO PROJETO	4
1.1. ESCOPO DO PROJETO	4
1.2. PRINCIPAIS FUNÇÕES DO SOFTWARE.....	5
1.2.1 Gerenciador de alarmes	5
1.2.2 Gerenciador de alarmes de terceiros	5
1.3. ENTREGÁVEIS	5
2. GERENCIAMENTO DE COMUNICAÇÃO.....	7
3. ESTRUTURA DE GESTÃO.....	8
3.1. CICLO DE VIDA DO PROJETO	8
3.2. ORGANIZAÇÃO DO PROJETO.....	8
3.2.1 Funções e Responsabilidade.....	8
3.2.2 Pessoal.....	8
3.3. GERENCIAMENTO DE RISCO	9
4. PLANEJAMENTO E CONTROLE.....	11
4.1. IDENTIFICAÇÃO DE RECURSOS.....	11
4.1.1 Pessoal.....	11
4.1.2 Tempo	11
4.1.3 Custo.....	12
5. GERENCIAMENTO DE ESCOPO	13
5.1. REGISTRO DE TAREFAS.....	13
5.2. GERENCIAMENTO DE MUDANÇAS	13
5.3. AGRUPAMENTO DE ENTREGAS	13

1. Visão Geral do Projeto

O projeto MediControl tem como objetivo contribuir com a manutenção dos horários dos medicamentos dos “acompanhados” (pessoas que dependem do uso diário de remédios) e o monitoramento dos “cuidadores” (familiares, responsáveis ou dos propriamente ditos cuidadores). Esse aplicativo visa ser mais do que somente um simples organizador de horários, pretende também auxiliar no monitoramento por parte dos “cuidadores”, facilitando seu acompanhamento diário através de registros e alertas.

Esse projeto nasceu da realidade dos seus próprios desenvolvedores, que enfrentam as dificuldades diárias relacionadas ao monitoramento do cuidado de terceiros. Assim, sabendo das reais necessidades inerentes ao assunto, pretende-se entregar um aplicativo completo e que faça diferença na vida de seus usuários.

A metodologia Scrum será seguida pela equipe de desenvolvimento para entregar este aplicativo. A previsão inicial é de que este projeto será completado, por uma única equipe de desenvolvimento, em um período de 15 semanas. Assim como outras equipes Scrum, essa equipe consistirá de um Scrum Master, um Product Owner e um grupo de desenvolvedores. Esse modelo de processo ajudará a mitigar riscos potenciais, fornecendo várias iterações de desenvolvimento e feedback do cliente.

1.1. Escopo do Projeto

O propósito do MediControl é principalmente facilitar o monitoramento dos cuidadores, já que muitas pessoas exigem cuidados diários em relação ao uso de medicamentos e muitas vezes a vida corrida das mesmas e de seus responsáveis torna o acompanhamento difícil, com o monitoramento do MediControl toda essa questão seria facilitada e até melhorada. Além disso temos também como um outro propósito o lembrete diário para os acompanhados, visto que muitas vezes com uma grande quantidade de remédios diários alguns podem ser negligenciados, e o sistema de lembrete e o cronograma dos medicamentos fornecidos pelo MediControl pretendem facilitar a manutenção do uso desses medicamentos.

Embora, na fase inicial de implementação, exista apenas um único tipo de usuário no sistema MediControl, o uso que cada usuário pode fazer o sistema pode ser dividido em 3 tipos:

- a) Usuário comum: O usuário comum utilizará o sistema apenas para gerir seus próprios remédios, sem intervir ou sem sofrer intervenções de terceiros.
 - a.1) Usuário comum com amigo: O usuário comum com amigo permite que terceiros (o amigo) visualize seus alarmes e receba notificações caso não administre o remédio no momento agendado.
- b) Usuário acompanhado: O usuário acompanhado, além de dispor de todos os recursos disponíveis ao usuário comum, poderá delegar a terceiros o cadastramento de alarmes em sua agenda. Ademais, este terceiro poderá monitorar a administração dos medicamentos do usuário acompanhado, sendo inclusive notificados dos eventos gerados.

- c) Usuário cuidador: O usuário cuidador, além de dispor de todos os recursos disponíveis ao usuário comum, poderá gerir a agenda de medicamentos de terceiros. Além disso, o usuário cuidador receberá notificações sobre a administração, ou não, dos medicamentos agendados para o usuário acompanhado.

1.2. Principais Funções do Software

O MediControl pode ser dividido em duas partes:

- Gerenciador de alarmes;
- Gerenciador de alarmes de terceiros.

1.2.1 Gerenciador de alarmes

Através do gerenciamento de alarmes, um usuário poderá administrar os alarmes por ele registrado para controlar a ingestão de medicamentos.

1.2.2 Gerenciador de alarmes de terceiros

Esta funcionalidade permite que um usuário conceda para outro a possibilidade de acompanhar, ou até mesmo gerir, seus alarmes.

1.3. Entregáveis

A lista detalhada de recursos será criada após a coleta de requisitos.

A figura abaixo descreve as liberações de alto nível para o MediControl:

Sprint 1 (5ª semana)	Sprint 2 (7ª semana)	Sprint 3 (9ª semana)
Login Operações de CRUD <ul style="list-style-type: none">• Usuário• Alarme	Gerenciamento de Alarmes de Terceiros	Reconhecimento de imagem para desativação de alarme

Para cada lançamento, as seguintes entregas estarão disponíveis:

Entregáveis	Descrição
Sprint Functionality	MediControl demonstrando todas as funcionalidades até a sprint atual.
Product Architecture	Uma arquitetura que suporta a funcionalidade completa, incluindo o sprint atual e funcionalidades adicionais a serem incorporadas em sprints subsequentes.
Source Code	Código-fonte necessário para construir um produto executável.

Entregáveis	Descrição
Product Backlog	Uma lista priorizada das funcionalidades remanescente a serem implementadas nas versões subsequentes, indicando o trabalho restante que deve ser concluído e o status atual do projeto.

2. Gerenciamento de Comunicação

A comunicação da equipe responsável pelo projeto se dará através de reuniões semanais presenciais nas terças-feiras em dois horários (Sprint Review Meeting pela manhã e Sprint Planning Meeting pela tarde), como também reuniões diárias através do aplicativo Discord (Daily Scrum).

Aplicativos de mensagem, como Whatsapp, email e ligações telefônicas também poderão ser utilizados para trocas de informações entre os participantes do projeto fora das reuniões já estabelecidas.

3. Estrutura de Gestão

3.1. Ciclo de Vida do Projeto

As funcionalidades são priorizadas no início do projeto e as liberações, previstas para ocorrerem em intervalos de tempo fixo, são entregues quando incorporadas ao código e funcionais. A cada lançamento consecutivo, as atividades de desenvolvimento anteriores podem ser revisadas conforme necessário para abordar mudanças de prioridade e/ou requisitos.

3.2. Organização do Projeto

Este projeto pode ser concluído com uma equipe de cinco pessoas em um período de 15 semanas. A equipe consistirá de engenheiros de software capazes de executar várias tarefas, desde a coleta de requisitos, criação, gerenciamento de banco de dados, desenvolvimento, teste e gerenciamento. Todos os membros irão alternar entre diferentes funções para o produto final entregue.

3.2.1 Funções e Responsabilidade

Função	Responsabilidade
Scrum Master	O objetivo principal é mitigar os problemas do projeto que podem impedir o progresso à medida que surgem. Além disso, devido a recursos limitados, o Scrum Master também será responsável pela implementação do MediControl.
Product Owner	Controla o orçamento e os recursos alocados para o projeto.
Project Team	Implementa as metas e ideias do projeto

3.2.2 Pessoal

O projeto será concluído com uma equipe de cinco membros em um período de 15 semanas.

A equipe incluirá os seguintes membros:

- 1 Scrum Master (Desenvolvedor)
- 4 Desenvolvedores

3.3. Gerenciamento de Risco

Risco:	Inexperiência da equipe com Projeto		
Probabilidade:	50%	Impacto:	Crítico
Monitoramento:	Monitorar a contribuição de cada membro da equipe com o projeto e o cumprimento do cronograma estabelecido, como também monitorar erros recorrentes.		
Mitigação:	Pedir auxílio aos professores e outros desenvolvedores mais experientes; Pesquisar projetos parecidos, conversar com desenvolvedores envolvidos na mesma área e pedir sempre que necessário a ajuda dos professores.		

Risco:	Inexperiência da equipe com o ambiente de desenvolvimento Android Studio		
Probabilidade:	70%	Impacto:	Crítico
Monitoramento:	Monitorar a contribuição de cada membro da equipe com o projeto e o cumprimento do cronograma estabelecido, como também monitorar erros recorrentes.		
Mitigação:	Separar tempo para estudos do ambiente antes da implementação do projeto.		

Risco:	Inexperiência da equipe com a plataforma de controle de versão (GitHub)		
Probabilidade:	90%	Impacto:	Catastrófico
Monitoramento:	Monitorar a contribuição de cada membro da equipe no GitHub.		
Mitigação:	Aprender os principais comandos da plataforma utilizada (commit, push e pull); Sempre comentar os commits e dar pull antes de dar push; Trocar informações sobre a plataforma de controle de versão sempre que surgir alguma dúvida.		

Risco:	Comunicação da Equipe		
Probabilidade:	50%	Impacto:	Catastrófico
Monitoramento:	Acompanhar o uso dos meios de comunicação definidos pela equipe.		
Mitigação:	Definir um meio de comunicação eficaz para a equipe; Deixar claro os canais de comunicação da equipe; Dar o direito a fala a todos os membros; Ressaltar o uso da plataforma de controle de versão escolhida – GitHub; Lembrar que o diálogo precisa ocorrer sempre de forma educada e sensata; Comentar o código sempre que necessário.		

Risco:	Comunicação da Equipe		
Probabilidade:	30%	Impacto:	Crítico
Monitoramento:	Monitorar se os Sprints estão sendo seguidos conforme o combinado utilizando a ferramenta Trello.		
Mitigação:	Sempre seguir o que foi combinado em cada Sprint.		

Risco:	Código mal escrito		
Probabilidade:	98%	Impacto:	Marginal
Monitoramento:	Utilização de ferramentas de análise estática de código e posterior correção dos erros encontrados.		
Mitigação:	Revisar o código, seguir as boas práticas de programação e utilizar ferramentas de análise estática de código.		

Risco:	Inexperiência da equipe com desenvolvimento Android		
Probabilidade:	50%	Impacto:	Crítico
Monitoramento:	Monitorar a contribuição de cada membro da equipe com o projeto e o cumprimento do cronograma estabelecido, como também monitorar erros recorrentes.		
Mitigação:	Separar tempo para estudos sobre desenvolvimento Android.		

Risco:	Inexperiência da equipe com a linguagem de programação Java		
Probabilidade:	50%	Impacto:	Crítico
Monitoramento:	Monitorar a contribuição de cada membro da equipe com o projeto e o cumprimento do cronograma estabelecido, como também monitorar erros recorrentes.		
Mitigação:	Separar tempo para estudos da linguagem antes da implementação do projeto.		

4. Planejamento e Controle

4.1. Identificação de recursos

4.1.1 Pessoal

A equipe do projeto inicialmente era composta por 6 (seis) integrantes, sendo eles 4 (quatro) desenvolvedores Android full stack, 1 (um) designer de software e um Scrum Master. Ainda no início do projeto, tivemos a saída de um dos desenvolvedores, Hermano Costa, o que deixou o time com um total de 5 (cinco) integrantes, sendo 3 (três) deles desenvolvedores.

Na parte final do projeto, onde foi necessária a integração do projeto com a ferramenta ML Kit, foi necessária a recontratação do desenvolvedor Hermano Costa em razão da sua experiência com a ferramenta em questão.

Integrante	Função
Antônio Neto	Desenvolvedor Android
Beatriz Falcão	Designer de software
Danilo Melo	Desenvolvedor Android
Hermano Costa	Desenvolvedor Android
Jones Albuquerque	Product Owner
Thamires Lopes	Scrum Master (Desenvolvedor)

4.1.2 Tempo

O projeto utilizará o método **SCRUM** e será conduzido conforme descrito abaixo:

- **Sprints:** O projeto será dividido em ciclos que são chamados de Sprints. Cada Sprint terá duração de uma semana (7 dias). E cada Sprint terá um conjunto de tarefas a serem realizadas nesse tempo.

- **Sprint Planning Meeting:** São as reuniões que acontecem no início de cada Sprint e que contam com a participação de todo o Scrum Team, como também do Product Owner. Essas reuniões têm como objetivo a definição do Sprint Backlog - que é a lista de tarefas que o time desenvolvedor se compromete a fazer no tempo do Sprint. Essas reuniões terão duração entre 1 (uma) e 2 (duas) horas.

- **Daily Scrum:** É a reunião diária da equipe de desenvolvimento do projeto para o acompanhamento do mesmo. Devido ao choque de horários e a dificuldade do encontro presencial, o Daily Scrum será feito exclusivamente via internet através do aplicativo Discord. Terá entre 15 (quinze) minutos e cada membro da equipe deverá responder três perguntas: O que você fez ontem? O que fará hoje? Há algum impedimento no seu caminho?

- **Sprint Review Meeting:** É a reunião que acontece no final de cada Sprint. Tem como objetivo mostrar o que foi alcançado pela equipe e é quando se faz um balanço dos pontos positivos e negativos do Sprint em questão. Tem duração entre 1 (uma) e 2 (duas) horas e conta com a presença do Product Owner e Scrum Team.

Data de início: 26 de março de 2019.

Data de fim: 17 de julho de 2019.

A data final de entrega é 18 de julho de 2019.

4.1.3 Custo

Os custos foram gerados através de pesquisa de mercado sobre o salário médio de cada função exercida por cada integrante do projeto. A equipe do projeto é formada por 1 (um) Scrum Master (desenvolvedor), 4 (quatro) desenvolvedores Android full stack e 1 (um) designer. Abaixo se encontra as informações com os salários de cada cargo e uma tabela com o custo total do projeto:

Função	Custo (R\$)
Scrum Master (desenvolvedor)	8.000,00
Desenvolvedor Android full stack	4.892,00
Designer de software	6.648,00

Considerando que um computador tem custo de R\$ 3.000,00 e o projeto terá duração de 4 (quatro) meses:

Item	Valor Bruto (R\$)	Qtd.	Tempo no projeto (mês)	Total (R\$)
Pessoal				
Scrum Master (desenvolvedor)	8.000,00	1	4	32.000,00
Desenvolvedor	4.892,00	4	4	78.272,00
Designer	6.648,00	1	4	26.592,00
Equipamentos				
Computador	3.000,00	5	-	15.0000
			TOTAL	151.864,00

5. Gerenciamento de Escopo

Este gerenciamento visa controlar o que está incluso no projeto, obtendo assim uma facilidade no desenvolver do sistema ao longo das etapas.

5.1. Registro de tarefas

O registro das atividades será feito através da plataforma Trello, que se encontra no link <https://trello.com/b/j5lvVyE6/sprints>, na qual serão alocadas em seus respectivos Sprints. Todos os interessados no projeto terão acesso às informações contidas na plataforma.

5.2. Gerenciamento de mudanças

As mudanças (adição ou exclusão de funcionalidades) no escopo do projeto sempre serão negociadas com o cliente por intermédio do Product Owner. Todas as mudanças precisam ser aprovadas pela equipe de desenvolvimento (Scrum Team).

5.3. Agrupamento de Entregas

As entregas serão feitas de acordo com o que foi acordado em cada Sprint Planning Meeting. Visando sempre a entrega do que foi combinado entre Product Owner, Scrum Team e cliente.