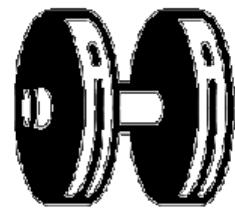


Strong Core



Strong Core

**Cliente: Polícia Militar do Paraná
(PMPR)**
**Responsáveis no cliente: Elisangela F.
Rezende**

**Projeto: Strong Core
Versão: 1.0**

Curitiba
2021

Strong Core

Responsáveis no cliente: Elisangela F. Rezende.

Responsáveis pelo projeto e desenvolvimento: Renan Nunes Avila (CEO da Soft Software)

Curitiba
2021

Histórico de Modificações

Data	Versão	Descrição	Autor
29/02/2021	1.0	Introdução / Objetivos e Levantamento de Requisitos	Renan N. Avila
10/03/2021	1.0	Estudo de Viabilidade	Renan N. Avila
19/03/2021	1.0	Atualizações nos Levantamentos de Requisitos	Renan N. Avila
22/03/2021	1.0	Finalização Estudo de Viabilidade	Renan N. Avila
25/03/2021	1.0	Resultados: Diagramas de caso / Modelagem	Renan N. Avila
02/04/2021	1.0	Resultados: Conteúdo dos Resultados	Renan N. Avila
06/04/2021	1.0	Resultados: Diagrama de Classes / Dicionário de Informações / Diagrama de Objetos	Renan N. Avila
15/04/2021	1.0	Resultado: Diagramas de Sequências	Renan N. Avila
16/04/2021	1.0	Finalizações Resultados	Renan N. Avila
19/04/2021	1.0	Conclusões / Referências Bibliográficas	Renan N. Avila

SUMÁRIO

1 INTRODUÇÃO.....	6
1.1 Objetivo Geral	6
1.2 Objetivos Específicos.....	6
1.3 Conteúdo do Plano do Projeto	6
2 LEVANTAMENTO DE REQUISITOS	8
2.1 Questões Organizacionais.....	8
2.2 Questões Econômicas	9
2.3 Questões Técnicas	10
2.4 Questões Operacionais.....	11
2.5 Requisitos Funcionais, Não Funcionais, Restrições de Projeto e Requisitos de Experiência do Usuário	13
3 ESTUDO DE VIABILIDADE	17
3.1 Viabilidade Organizacional	17
3.2 Viabilidade Econômica.....	17
3.3 Viabilidade Técnica	18
3.4 Viabilidade Operacional	19
3.5 Recursos a serem utilizados	19
4 RESULTADOS	21
4.1 Conteúdo dos Resultados	21
4.2 Modelagem	25
5 CONCLUSÕES.....	50
6 REFERÊNCIAS BIBLIOGRÁFICAS.....	51

LISTA DE FIGURAS

Figura 1. Autenticação do Strong Core.....	24
Figura 2. Cadastro Aplicativo Strong Core.....	25
Figura 3. Menu do Aplicativo Strong Core.....	26
Figura 4. Core Training do Aplicativo Strong Core.....	27
Figura 5. Cofigurações Personalizadas do Aplicativo.....	27
Figura 6. Caso de uso Login aplicativo Strong Core.....	28
Figura 7. Caso de uso Cadastro aplicativo Strong Core.....	32
Figura 8. Caso de uso Configurações aplicativo Strong Core.....	35
Figura 9. Caso de uso Planilha Treino aplicativo Strong Core.....	39
Figura 10. Caso de uso Abrir ticket aplicativo Strong Core.....	42
Figura 11. Diagrama de Classes Strong Core.....	44
Figura 12. Diagrama de Sequência Login Strong Core.....	49
Figura 13. Diagrama de Sequência Register Strong Core.....	50
Figura 14. Diagrama de Sequência Configurations Strong Core.....	50
Figura 15. Diagrama de Sequência Training Worksheet Strong Core.....	51
Figura 16. Diagrama de Sequência Open Ticket Strong Core.....	52

LISTA DE TABELAS E QUADROS

Tabela 1. Planning Poker dos Requisitos do aplicativo Strong Core.	17
Gráfico 1. Reduções de custos mensais médicos com a utilização do aplicativo Strong Core.....	20

LISTA DE ABREVIATURAS E SIGLAS

PMPR: Polícia Militar do Paraná.

App: aplicativo.

RH: Recurso humano.

Wi-fi: wireless fidelity.

1 INTRODUÇÃO

Em parceria com a **PMPR**, a empresa Soft Software visa desenvolver o **app** Strong Core que pretende auxiliar no desempenho físico dos profissionais da **PMPR**. Por meio deste documento a empresa Soft Software vem esclarecer e apresentar uma proposta de escopo para desenvolvimento do projeto, apresentando objetivos, coletando requisitos para o sistema, demonstrando custos e resultados da eficácia do plano do projeto.

1.1 Objetivo Geral

Com o **app** Strong Core a empresa Soft Software visa desenvolver um aplicativo de treinamento físico para os policiais militares do Paraná, com o intuito de reduzir futuros custos médicos, proporcionar melhor condicionamento físico ao policial e melhorar a disposição dos mesmos a partir da execução de exercícios físicos.

A empresa Soft Software se compromete a fazer um aplicativo versátil, acessível / “amigável” ao seu público-alvo e que proporcionara excelentes resultados a médio / longo prazo.

1.2 Objetivos Específicos

- Aumentar a disposição do profissional da **PMPR**;
- Reduzir custos com ciclos de revisão médicas mensais (possivelmente passando para ciclos trimestrais);
- Desenvolver um aplicativo no qual seja intuitivo ao usuário e autodidático;
- Compor uma planilha de treinos específica para cada usuário, capacitando reforçar os pontos fracos de cada profissional (dores / lesões).

1.3 Conteúdo do Plano do Projeto

A seguir a Soft Software apresenta a divisão do conteúdo a ser abordado neste documento:

Capítulo 2: Levantamento de Requisitos.

- 2.1: Questões Organizacionais.
- 2.2: Questões Econômicas.
- 2.3: Questões Técnicas.
- 2.4: Questões operacionais.
- 2.5: Requisitos Funcionais, Não Funcionais, Restrições de Projeto e Requisitos de Experiência do Usuário.
- 2.6: Estimativa de Esforço de Software com Planning Poker.

Capítulo 3: Estudo de Viabilidade.

- 3.1: Viabilidade Organizacional.
- 3.2: Viabilidade Econômica.
- 3.3: Viabilidade Técnica.
- 3.4: Viabilidade Operacional.
- 3.5: Recursos a serem utilizados.

Capítulo 4: Resultados.

- 4.1: Conteúdo dos Resultados.
- 4.2: Modelagem.

Capítulo 5: Conclusões.

Capítulo 6: Referências Bibliográficas.

2 LEVANTAMENTO DE REQUISITOS

Neste capítulo a empresa Soft Software apresenta o levantamento de requisitos do projeto.

2.1 Questões Organizacionais

- Quais recursos humanos serão necessários para o desenvolvimento do **app Strong Core** ?

- 1 Desenvolvedor Android / Java;
- 1 Gerente de Projetos;
- 1 Engenheiro de Software;
- 1 Representante da **PMPR**;

- Quanto tempo será requerido pela empresa Soft Software para desenvolvimento e conclusão do **app** ?

A empresa Soft Software desenvolverá o **app Strong Core** no prazo de 4 meses, validado e testado.

- Qual será as etapas do desenvolvimento do Strong Core ?

- 1º) Acordo e levantamento de requisitos entre as partes interessadas;
- 2º) Planejamento e aquisição do **RH** necessário;
- 3º) Desenvolvimento do projeto;
- 4º) Testes, validação de software;
- 5º) Primeira entrega (onde o cliente **PMPR** poderá ajustar certos pontos do projeto);
- 6º) Ajustes e implementações finais;
- 7º) Testes, validação de software;
- 8º) Entrega do produto final.

- Quantos encontros a empresa Soft Software fará com as partes interessadas para apresentar o andamento do projeto ?

A empresa Soft Software se compromete em organizar 2 encontros mensais (quinzenais) durante o prazo de desenvolvimento do aplicativo

Strong Core, com o intuito de apresentar o andamento do desenvolvimento do produto e receber feedbacks constantes da parte do cliente.

- Quais resultados se deseja obter com o aplicativo para a organização da **PMPR** ?

Além de redução de custos médicos, o aplicativo irá concentrar todas informações e dados de cada policial militar, como exemplo: Nome, posto, graduação, estatura, peso, dores musculares e evolução nos treinos. Possibilitando uma desconcentração de armazenamento físico para dados remotos.

- Qual estrutura a **PMPR** precisara implementar para receber o aplicativo Strong Core ?

O aplicativo Strong Core será para dispositivos mobile, no caso a única estrutura necessária para receber e utilizar do aplicativo é uma rede **Wi-fi**, que será necessitada para autenticação dos profissionais da Policia Militar.

- Quais benefícios o aplicativo trará para o profissional da **PMPR** ?

Maior resistência física, bem estar para consigo mesmo como individuo e maior disposição no serviço.

2.2 Questões Econômicas

- Atualmente a empresa Soft Software oferece o melhor orçamento do mercado ?

Não só o melhor preço final do produto como melhor qualidade de software. Sendo desenvolvido diretamente junto ao cliente para atingir todas expectativas e necessidades dos usuários que se direciona.

- Qual custo do RH envolvido no projeto ?

- 1 Desenvolvedor Android / Java = R\$4.600,00 / mensal;
- 1 Gerente de Projetos = R\$9.000,00 / mensal;
- 1 Engenheiro de Software = R\$6.000,00 / mensal.

- Qual custo de recursos tecnológicos ?

Será designado e solicitado uma verba de R\$10.000,00 para gastos com licenças e tecnologias a serem adotadas no desenvolvimento do produto final.

- Qual custo total do projeto ?

O preço final do projeto Strong Core será de R\$88.400,00 incluindo os gastos do **RH** aos tecnológicos.

- Caso houver, quais retornos financeiros o **app** Strong Core pode proporcionar a **PMPR** ?

Os retornos previstos serão reduções significativas no gasto em ciclos de revisão médica de cada policial militar, assim possibilitando manter mais dinheiro em caixa para investimentos em outras áreas.

- Gastos e orçamentos imprevistos no escopo do projeto que possam surgir serão cobertos por parte da empresa contratada Soft Software ?

Caso haja gastos imprevistos será agendado reuniões para debate e apresentação do problema e novos orçamentos. A empresa Soft Software assumira 50% do valor desses novos gastos.

- Quais seriam os gastos pra manutenção do aplicativo ?

A empresa Soft Software visa criar um **app** que possua uma vida útil bem ampla, por isso não será requerido gastos significativos com manutenção do aplicativo e a Soft Core proporcionara suporte para futuras falhas ou atualizações que sejam respectivas a mesma versão do produto.

2.3 Questões Técnicas

- Qual será a melhor linguagem para desenvolvimento do aplicativo ?

O aplicativo será desenvolvido em Java orientado a objetos.

- Quais plataformas o **app** Strong Core irá suportar ?

As plataformas que irão ser suportadas serão Android e IOS.

- Qual banco de dados a ser utilizado ?

O banco de dados a ser utilizado será SQL Server, que funciona por dinâmica relacional. De linguagem T-SQL, trabalha com dados criptografados que garantem mais segurança aos usuários.

- Quais versões de Android serão capazes de rodar o aplicativo ?

Qualquer dispositivo com versão Android 4.4: KitKat. igual ou superior a essa será apto a ter o aplicativo Strong Core.

- Quais Versões IOS serão capazes de suportar o aplicativo ?

Versões IOS 11 ou superior serão aptas a instalar e usufruir do aplicativo em seus dispositivos.

- Qual peso/armazenamento o **app** Strong Core necessitará ?

- 160 MB de espaço de armazenamento livre em disco.

- Quanto de RAM o aplicativo irá exigir para rodar de forma linear, sem travar

- 800 MB de RAM para rodar linearmente sem falhas e sem travamentos.

2.4 Questões Operacionais

- Como será feita a autenticação do profissional no sistema do aplicativo ?

Através do e-mail e senha, após cadastro completo do policial militar no sistema.

- O aplicativo será intuitivo de forma didática e fácil utilização do usuário ?

Sim, irá conter instruções e auxílios de como acompanhar seu desempenho físico e evolução nos treinos, de forma acessível e clara aos usuários do **app**.

- Quais licenças serão necessárias para a produção do **app** ?

Nenhuma licença será necessária. Tudo será produzido do zero com todos direitos da **PMPR** sobre o que for produzido pela empresa Soft Software relativo ao projeto Strong Core.

- Os futuros usuários desse aplicativo apoiam o desenvolvimento do mesmo ?

Sim apoiam o desenvolvimento devido a manter seus treinos e evoluções em um local prático e de rápido acesso, quanto um significativo aumento da motivação para iniciar uma rotina de exercícios físicos linear.

- Quais são especificamente os futuros usuários do aplicativo ?

Policiais Militares da organização da **PMPR**.

- Foram feitas pesquisas com os usuários que irão utilizar o aplicativo para saber seus pontos de vista e solicitações para o **app** ?

Sim, foram feitos brainstorms com profissionais da **PMPR** que foram registrados todos dados importantes e ideias para implementar no produto final.

- Quais requisitos dos usuários para o **app** ?

Um aplicativo onde contenha: Uma forma de estabelecer planilhas de treinos de forma individual progressiva com base nos pontos fracos dos usuários, um aplicativo leve que não ocupe muito espaço de armazenamento no dispositivo e que seja intuitivo e acolhedor ao usuário.

2.5 Requisitos Funcionais, Não Funcionais, Restrições de Projeto e Requisitos de Experiência do Usuário

Requisitos Funcionais:

#RF01 → O usuário deve ser capaz de fazer entrada no aplicativo a partir de e-mail e senha;

#RF02 → A primeira etapa de cadastro do usuário deve requisitar dados como: nome, sobrenome, sexo, posto, graduação, estado, cidade, estatura, peso, e-mail e senha.

#RF03 → A segunda etapa do cadastro deve solicitar um histórico de dores ou preparo físico.

#RF04 → O aplicativo Strong Core deve ser apto a montar uma tabela de treino tendo como objetivo fortalecer os pontos fracos de cada indivíduo.

#RF05 → O aplicativo deve mostrar a evolução do usuário nos treinos.

#RF06 → O aplicativo deve ser capaz de reproduzir música.

#RF07 → Os treinos devem ser divididos em níveis, de forma progressiva. Limitados por semana sem que o usuário possa pular ou burlar os treinos.

#RF08 → O aplicativo deve conter um avatar, no qual o usuário poderá escolher o avatar que deseja para instruir de como se faz os exercícios.

#RF09 → O aplicativo deve ter uma seção para entrar em contato com o suporte técnico, seja para feedbacks / reclamações ou perguntas.

#RF10 → O aplicativo deve conter uma agenda inclusa, responsável por notificar os usuários de seus treinos em certos horários.

Requisitos Não Funcionais:

Restrições de Projeto:

#RNF01 → Para garantir as segurança dos dados do usuário deve ser o cadastro será feito em SQL-Server como banco de dados, que utiliza de criptografia nos dados.

#RNF02 → Para criação do avatar deverá ser composto de uma forma leve tais como (.OBJ, .FBX, .3DS, .PLY, .STL) para não comprometer a leveza do aplicativo ou gerar travamentos.

Requisitos de Experiência do Usuário:

#RNF03 → O aplicativo deve conter explicações claras e objetivas para o usuário não ter complicações na utilização do mesmo.

#RNF04 → O aplicativo deve oferecer uma customização personalizada para o avatar como: modelo feminino, masculino e alteração de uniforme.

#RNF05 → O avatar deve mostrar como se faz a execução de forma correta do exercício antes, para que o usuário possa repetir de forma correta sem perigo de causar futuras lesões.

2.6 Estimativa de Esforço de Software com Planning Poker

Tabela 1. Planning Poker dos Requisitos do aplicativo Strong Core.

#	Item do Backlog(User Story)	Story Points
#RF01	O usuário deve ser capaz de fazer entrada no aplicativo a partir de e-mail e senha.	0
#RF02	A primeira etapa de cadastro do usuário deve requisitar dados como: nome, sobrenome, sexo, posto, graduação, estado, cidade, estatura, peso, e-mail e senha.	0
#RF03	A segunda etapa do cadastro deve solicitar um histórico de dores ou preparo físico.	9

#RF04	O aplicativo Strong Core deve ser apto a montar uma tabela de treino tendo como objetivo fortalecer os pontos fracos de cada indivíduo.	20
#RF05	O aplicativo deve mostrar a evolução do usuário nos treinos.	8
#RF06	O aplicativo deve ser capaz de reproduzir música.	0
#RF07	Os treinos devem ser divididos em níveis, de forma progressiva. Limitados por semana sem que o usuário possa pular ou burlar os treinos.	100
#RF08	O aplicativo deve conter um avatar, no qual o usuário poderá escolher o avatar que deseja para instruir de como se faz os exercícios.	5
#RF09	O aplicativo deve ter uma seção para entrar em contato com o suporte técnico, seja para feedbacks / reclamações ou perguntas.	1/2
#RF10	O aplicativo deve conter uma agenda inclusa, responsável por notificar os usuários de seus treinos em certos horários.	3
#RNF01	Para garantir as segurança dos dados do usuário deve ser o cadastro será feito em SQL-Server como banco de dados, que utiliza de criptografia nos dados.	9
#RNF02	Para criação do avatar deverá ser composto de uma forma leve tais como (.OBJ, .FBX, .3DS, .PLY, .STL) para não comprometer a leveza do aplicativo ou gerar travamentos.	π
#RNF03	O aplicativo deve conter explicações claras e objetivas para o usuário não ter complicações na utilização do	1/2

	mesmo.	
#RNF04	O aplicativo deve oferecer uma customização personalizada para o avatar como: modelo feminino, masculino e alteração de uniforme.	20
#RNF05	O avatar deve mostrar como se faz a execução de forma correta do exercício antes, para que o usuário possa repetir de forma correta sem perigo de causar futuras lesões.	∞

3 ESTUDO DE VIABILIDADE

Nesta seção a empresa Soft Software visa demonstrar a eficácia do aplicativo e especificações mais complexas de benefícios que o app Strong Core poderá proporcionar em médio/longo prazo.

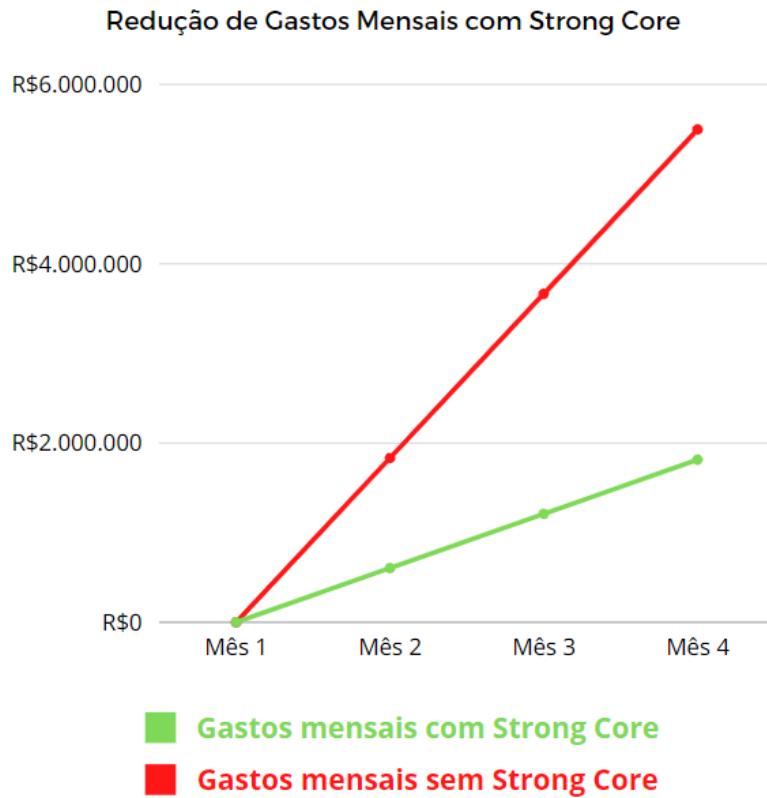
3.1 Viabilidade Organizacional

A aquisição do aplicativo proporcionará uma redução de 50% de arquivos médicos armazenados fisicamente para mídia digital, de fácil e rápido acesso sobre a saúde e parte física do policial militar. Além disso com os treinos progressivos que serão disponibilizados na plataforma do aplicativo o policial militar com o tempo desenvolverá mais resistência física, consequentemente aumentando a disposição para prestar seu nobre serviço.

3.2 Viabilidade Econômica

Atualmente há 18.329 policiais militares registrados no estado do Paraná, com isso vamos usar o valor simbólico de R\$100,00 por mês com ciclos mensais de exames para cada policial militar, o que dariam gastos de R\$1.832.900,00 mensais nos modelos atuais. Com a implementação do aplicativo Strong Core e os resultados a médio/longo prazo os ciclos mensais poderiam ser passados para ciclos de revisão médicas trimestrais, representando uma economia de 67% mensal de cada policial militar, conforme apresenta o gráfico abaixo.

Grafico 1. Reduções de custos mensais médicos com a utilização do aplicativo Strong Core.



3.3 Viabilidade Técnica

O desenvolvimento do software Strong Core foi modelado e prototipado com as melhores tecnologias de mercado, com intuito de entregar um produto versátil e eficaz.

Atualmente o sistema do aplicativo Strong Core foi projetado para suportar até 50.000 registros de usuários na plataforma. Com um serviço de manutenção e atualizações durante 2 anos de ciclo de vida do aplicativo incluídos no orçamento do projeto, capacitando uma longa vida útil do aplicativo sem ficar obsoleto.

Os sistemas operacionais suportados pelo aplicativo são Android e IOS, cujas versões iniciais requeridas para download são Android 4.4: KitKat e IOS 11 respectivamente. Requerendo apenas 160 MB de espaço para armazenamento e 800 MB de RAM, demonstrando-se um aplicativo que será capaz de ser instalado e usado na maioria dos modelos de celulares.

Para reforçar a segurança dos usuários foi optado por usar o banco de dados SQL Server, que utiliza linguagem T-SQL, onde trabalha com dados criptografados que garantem mais segurança aos usuários.

Para desenvolvimento do sistema foi escolhido o software Unity Pro que disponibiliza das ultimas ferramentas para desenvolvimento Mobile do mercado, onde o CEO da Soft Software Renan Nunes Avila se responsabilizou pelo desenvolvimento e programação em Java do aplicativo. Também responsável pela gerência e andamento do projeto o CEO optou e conjunto com sua equipe o software 3D Studio Max para partes gráficas do aplicativo.

O sistema foi prototipado para ser amigável e didático ao público alvo que são os Policiais Militares do Paraná. O aplicativo não requer nenhum conhecimento especial do usuário, tornando acessível a todos e buscando tornar a vida dos policiais mais saudável e disposta.

3.4 Viabilidade Operacional

A aceitação do aplicativo foi concebida por 90% da organização da **PMPR**, com total apoio administrativo. Todos requisitos foram considerados e implementados para atender ao máximo de necessidades possíveis dos policiais militares, que irão ser os principais usuários do aplicativo.

3.5 Recursos a serem utilizados

A seção a seguir apresenta detalhado os profissionais que serão utilizados, horas previstas, custos e gastos com softwares/equipamentos.

- 1 Desenvolvedor Android / Java = R\$4.600,00 / mensal.

(80 Horas previstas de serviço durante 4 meses)

Renan Nunes Avila (CEO da Soft Software)

- 1 Gerente de Projetos = R\$9.000,00 / mensal.

(130 Horas previstas de serviço durante 4 meses)

Renan Nunes Avila (CEO da Soft Software)

- 1 Engenheiro de Software = R\$6.000,00 / mensal.
(100 Horas previstas de serviço durante 4 meses)
Providenciado pela Soft Software

- Unity Pro 1 Year License = R\$5.760,00 / anual
- 3D Studio Max 4 Months = R\$797,00 / mensal

Custo total Recursos Humanos:

R\$78.400,00

Custo total Licenças de Softwares:

R\$8.948,00

Custo Total do Projeto:

R\$87.348,00

4 RESULTADOS

4.1 Conteúdo dos Resultados

Nesta seção a empresa Soft Software vai demonstrar modelos de protótipos do andamento do projeto e como prevemos para o futuro do aplicativo.

Primeiramente temos os resultados da tela inicial do aplicativo, conforme requisitos para autenticação ser feita através do e-mail e senha do policial militar.

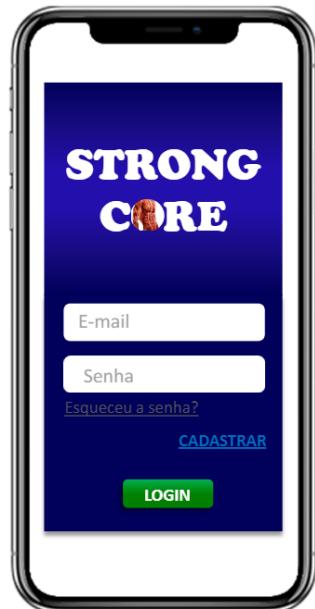


Figura 1. Autenticação do Strong Core.

As páginas de cadastros foram projetadas para a coleta do máximo de informação possíveis sobre históricos de atividade física, quanto dores musculares.

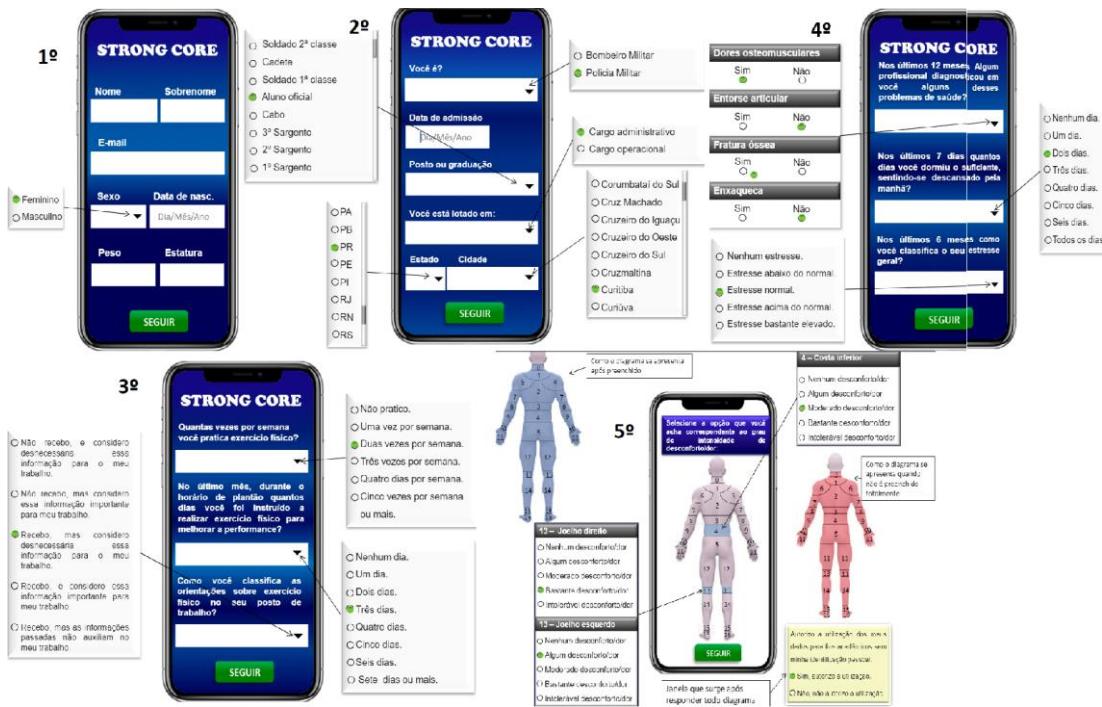


Figura 2. Cadastro Aplicativo Strong Core.

Em seguida temos o menu do usuário, onde ele desfrutará de uma série de opções para melhorar e personalizar o **app** de forma mais prática para seus gostos.



Figura 3. Menu do Aplicativo Strong Core.

A seguir, conforme estipulado nos requisitos a equipe da Soft Software desenvolveu um sistema de montagens de treinos individual progressivo para cada indivíduo baseado nas especificações dos cadastros de cada usuário. Os treinos foram divididos em etapas como solicitado e apresentam um avatar com instrutor com intuito de mostrar a forma correta de execução do exercício, como havia sido solicitado, também adicionamos um relatório dos treinos.

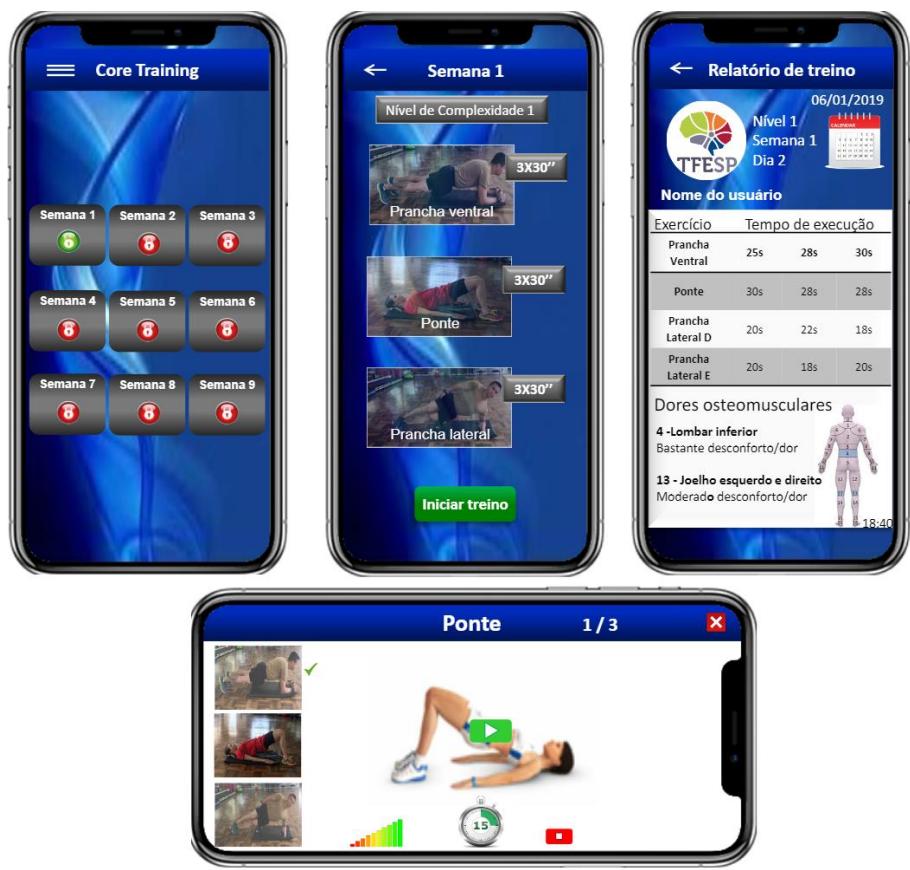


Figura 4. Core Training do Aplicativo Strong Core.

Por fim adicionamos os recursos solicitados de calendário (agenda para notificação), implementação de música durante os treinos, customização personalizada do avatar do instrutor e um meio de contato ao suporte para envios de sugestões ou dúvidas.



Figura 5. Configurações Personalizadas do Aplicativo.

4.2 Modelagem

Nesta seção a Soft Software apresenta a modelagem do projeto Strong Core.

Strong Core – Login Use Case:

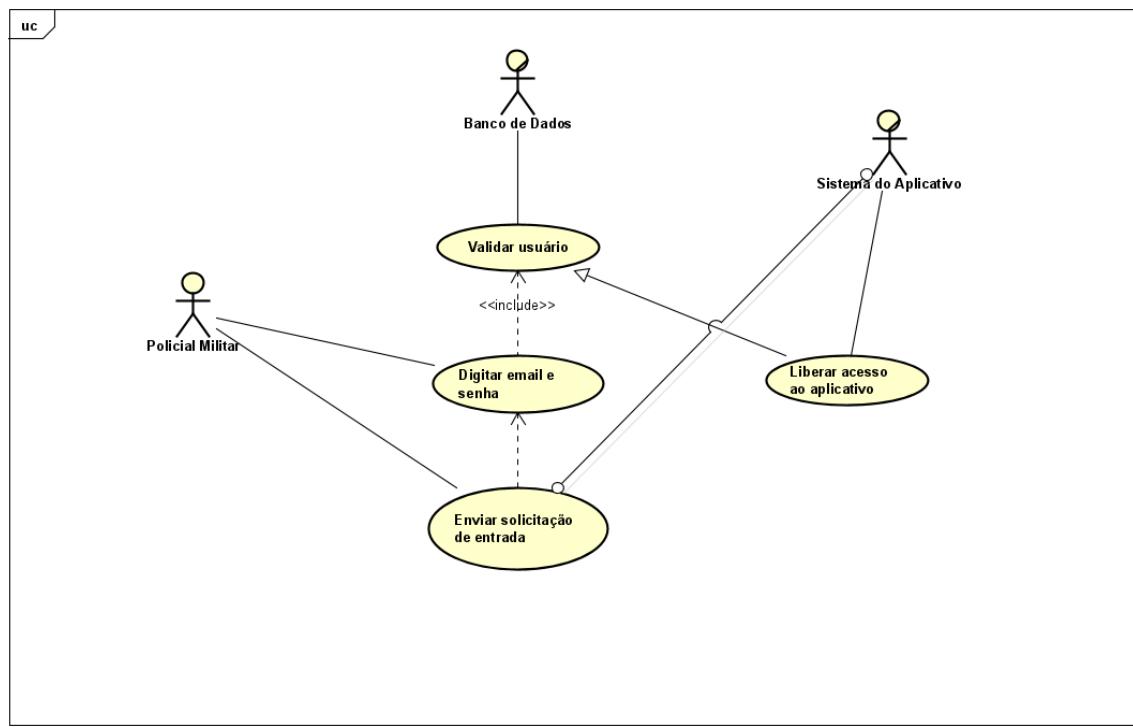


Figura 6. Caso de uso Login aplicativo Strong Core.

- **Requisitos Funcionais Atendidos:**

#RF01 → O usuário deve ser capaz de fazer entrada no aplicativo a partir de e-mail e senha;

- **Especificações do Caso de Uso: Digitar email e senha**

Nome	UC01 – Digitar email e senha
Atores	Autor Principal: Policial Militar.
Descrição	Caso de uso executado quando o usuário insere seus dados de login.
Pré-condições	- Usuário deve estar registrado.

Pós-condições	Campos de email e senha preenchidos.
Fluxo Básico	
Ações dos atores	Ações do Sistema
	1- Abre a tela inicial de login solicitando o preenchimento de email e senha, ou oferece opção de cadastro.
2 – O Policial Militar preenche os dados.	
Regras de Negócio	
Não há neste caso de uso.	
Fluxo Alternativo 1 – Deixar campos em branco	
Ações dos atores	Ações do Sistema
	1- Sistema solicita preenchimento do email e senha, ou cadastro de novo usuário.
Fluxo Exceção	
Não há neste caso de uso.	

- Especificações do Caso de Uso: Enviar solicitação de entrada

Nome	UC02 – Enviar solicitação de entrada
Atores	Autor Principal: Policial Militar. Autor Suporte: Sistema do Aplicativo.
Descrição	Caso de uso executado quando o usuário envia os dados preenchidos ao sistema.
Pré-condições	- Usuário deve estar registrado. - Usuário deve ter completado UC01.
Pós-condições	Dados enviados para análise e validação.

Fluxo Básico	
Ações dos atores	Ações do Sistema
1- Clicar para enviar os dados de email e senha ao sistema.	

	2- Captura os dados e leva para validação.
Regras de Negócio	
[RN001] – Usuário ter completado UC01.	
Fluxo Exceção	
Ações dos atores	Ações do Sistema
	2.a – Caso [RN001] não atendido, retornar como usuário inválido.

- Especificações do Caso de Uso: Validar usuário

Nome	UC03 – Validar usuário
Atores	Autor Principal: Banco de Dados.
Descrição	Verifica e retorna se o usuário está cadastrado na plataforma.
Pré-condições	<ul style="list-style-type: none"> - Usuário deve estar registrado. - Usuário deve ter completado UC01. - Usuário deve ter feito UC02.
Pós-condições	Validação dos dados.

Fluxo Básico	
Ações dos atores	Ações do Sistema
1- Banco de Dados verifica se o usuário está registrado.	
	2- Retorna se usuário é válido ou não.
Regras de Negócio	
[RN001] – Usuário ter completado UC01.	
[RN002] – Usuário ter feito UC02.	
Fluxo Alternativo 1 – Usuário não ser válido	
Ações dos atores	Ações do Sistema
	1- Sistema retorna usuário inválido.
	2- Sistema volta para UC01.
Fluxo Alternativo 2 – Ser válido	
Ações dos atores	Ações do Sistema

	1- Retornar usuário válido.
Fluxo Exceção	
Ações dos atores	Ações do Sistema
	2.a – Caso [RN001] não atendido, retornar como usuário inválido.

- **Especificações do Caso de Uso: Liberar acesso ao aplicativo**

Nome	UC04 – Liberar acesso ao aplicativo
Atores	Ator Principal: Sistema do Aplicativo.
Descrição	Recebe a informação se o usuário é válido e libera acesso.
Pré-condições	- Usuário deve ser válido no UC03.
Pós-condições	Mudar tela para entrar no aplicativo (menu).

Fluxo Básico	
Ações dos atores	Ações do Sistema
	1- Liberar acesso ao aplicativo.
Regras de Negócio	
[RN003] – Usuário ser válido no UC03.	
Fluxo Exceção	
Ações dos atores	Ações do Sistema
	1- Caso [RN003] não atendido, retornar para UC01.

Strong Core - Register Use Case:

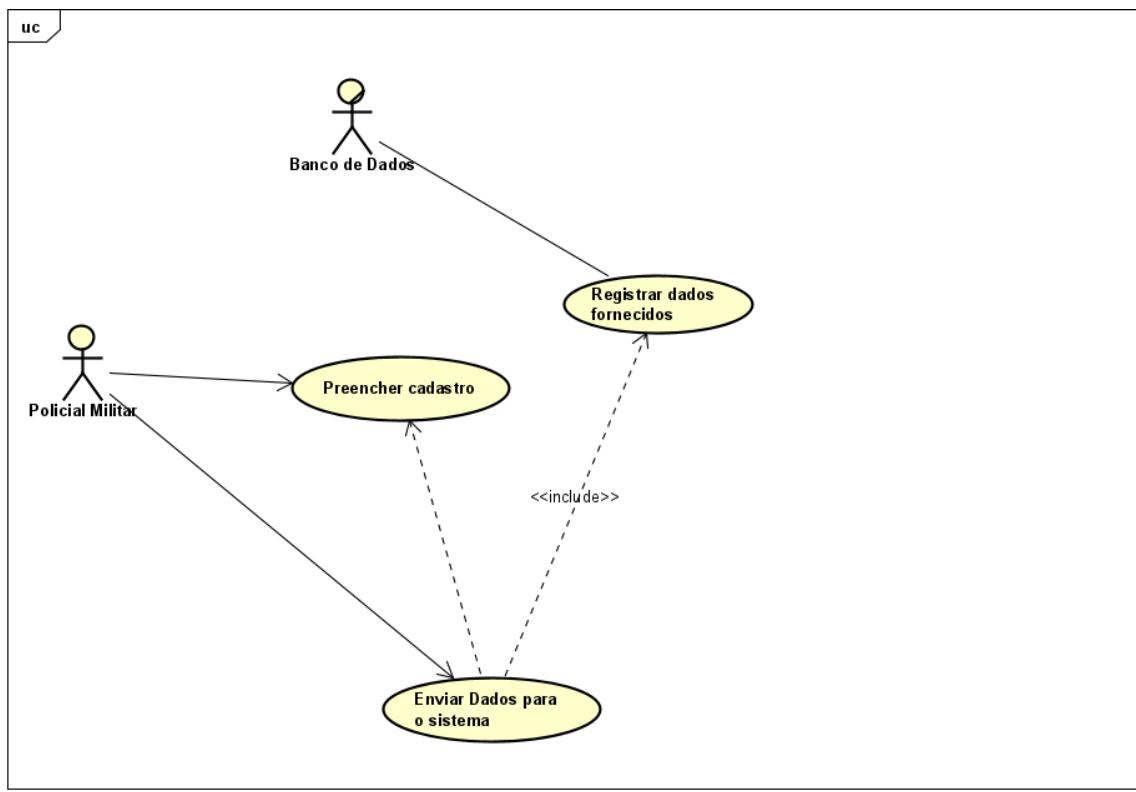


Figura 7. Caso de uso Cadastro aplicativo Strong Core.

- **Requisitos Funcionais Atendidos:**

#RF02 → A primeira etapa de cadastro do usuário deve requisitar dados como: nome, sobrenome, sexo, posto, graduação, estado, cidade, estatura, peso, e-mail e senha.

#RF03 → A segunda etapa do cadastro deve solicitar um histórico de dores ou preparo físico.

- **Especificações do Caso de Uso: Preencher cadastro**

Nome	UC05 – Preencher cadastro
Atores	Autor Principal: Policial Militar.

Descrição	Caso de uso executado ao usuário inserir informações de cadastro: Nome, Sobrenome, sexo, posto, graduação, estado, cidade, estatura, peso, email, senha, histórico físico.
Pré-condições	- Clicar na opção de cadastrar usuário.
Pós-condições	Dados preenchidos.

Fluxo Básico	
Ações dos atores	Ações do Sistema
	1- Solicita preenchimento dos dados de cadastro.
2- Policial Militar preenche as informações requeridas.	
Regras de Negócio	
[RN004] – Todos campos são obrigatórios.	
Fluxo Alternativo 1 – Cancelar cadastro	
Ações dos atores	Ações do Sistema
1- Policial pode cancelar o cadastro a qualquer hora.	
	2- Desconsiderar informações digitadas.
	3- Voltar para tela de início de Login do UC01.
Fluxo Exceção	
Ações dos atores	Ações do Sistema
	1.a – Caso [RN004] não esteja sendo atendida, retorna 'Todos campos devem ser preenchidos!', após isso sistema volta ao Fluxo Básico 2.

- **Especificações do Caso de Uso: Enviar Dados para o sistema**

Nome	UC06 – Enviar Dados para o sistema
Atores	Autor Principal: Policial Militar.
	Caso de uso executado ao usuário

Descrição	confirmar cadastro e enviar seus dados ao sistema.
Pré-condições	- Ter completo UC05.
Pós-condições	Enviar dados ao sistema.

Fluxo Básico	
Ações dos atores	Ações do Sistema
1- Clica para enviar seus dados.	
	2- Captura os dados para levar para registro e validação.
Regras de Negócio	
[RN005] – User Case UC05 completo.	
Fluxo Exceção	
Ações dos atores	Ações do Sistema
	1.a – Caso [RN005] não esteja sendo atendida, retorna ‘Todos campos devem ser preenchidos!’, após isso sistema volta ao Fluxo Básico 2.

- **Especificações do Caso de Uso: Registrar dados fornecidos**

Nome	UC07 – Registrar dados fornecidos.
Atores	Autor Principal: Banco de Dados.
Descrição	Caso de uso executado ao registrar dados enviados pelo usuário.
Pré-condições	- Ter completo UC05. - Ter feito UC06.
Pós-condições	Dados armazenados no banco de dados, novo usuário cadastrado.

Fluxo Básico	
Ações dos atores	Ações do Sistema
	1- Envia os dados ao Banco de dados para validação.
2- Banco de dados recebe os dados e cria novo usuário.	

	3- Sistema retorna como usuário cadastrado com sucesso.
	4- Sistema volta para tela de Login do UC01.
Regras de Negócio	
[RN005] – User Case UC05 completo.	
[RN006] – User Case UC06 completo.	
[RN007] – Campo Email não pode ser cadastrado repetido no Banco de Dados.	
Fluxo Alternativo 1 – Email já cadastrado	
Ações dos atores	Ações do Sistema
	1- Retorna a mensagem 'Email já cadastrado, por favor insira novo email!' e retorna para UC05.
Fluxo Exceção	
Não há neste caso de uso.	

Strong Core – Configurations Use Case:

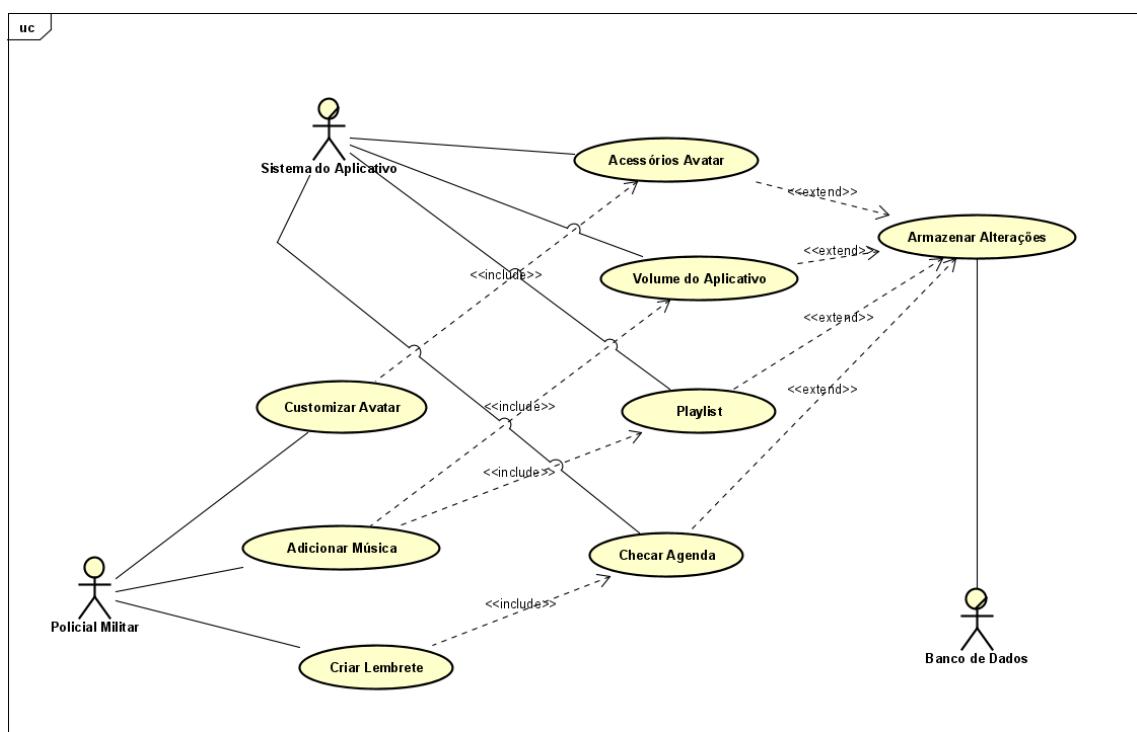


Figura 8. Caso de uso Configurações aplicativo Strong Core.

- **Requisitos Funcionais Atendidos:**

#RF06 → O aplicativo deve ser capaz de reproduzir música.

#RF08 → O aplicativo deve conter um avatar, no qual o usuário poderá escolher o avatar que deseja para instruir de como se faz os exercícios.

#RF10 → O aplicativo deve conter uma agenda inclusa, responsável por notificar os usuários de seus treinos em certos horários.

- **Especificações do Caso de Uso: Customizar Avatar**

Nome	UC08 – Customizar Avatar
Atores	Autor Principal: Policial Militar. Aktor Suporte: Banco de Dados.
Descrição	Caso de uso executado quando usuário selecionar para personalizar seu avatar (instrutor) no aplicativo Strong Core.
Pré-condições	- Usuário estar autenticado.
Pós-condições	Salvar alterações de customização de avatar (instrutor).

Fluxo Básico	
Ações dos atores	Ações do Sistema
	1- Sistema disponibiliza uma serie de customizações para o avatar: sexo, cor da pele, roupa.
2- Policial seleciona customizações preferidas.	
3- Policial Militar confirma as customizações.	
4- Banco de Dados armazena as alterações feitas no avatar.	
	5- Sistema mostra o novo avatar.
Regras de Negócio	
Não há neste caso de uso.	

Fluxo Exceção
Não há neste caso de uso.

- **Especificações do Caso de Uso: Adicionar Música**

Nome	UC09 – Adicionar Música
Atores	Autor Principal: Policial Militar. Atores Suporte: Banco de Dados.
Descrição	Usuário adiciona música ou playlist para escutar durante treino.
Pré-condições	- Usuário estar autenticado.
Pós-condições	Música selecionada.

Fluxo Básico	
Ações dos atores	Ações do Sistema
	1- Sistema disponibiliza uma série de playlists e músicas para usuário.
2- Policial Militar seleciona a música ou playlist desejada.	
3- Banco de Dados armazena as seleções.	
	4- Sistema reproduz música no aplicativo.
Regras de Negócio	
Não há neste caso de uso.	
Fluxo Alternativo 1 – Usuário não seleciona música	
Ações dos atores	Ações do Sistema
1- Policial Militar não seleciona música.	
2- Banco de Dados armazena.	
	3- Aplicativo não reproduz audio.
Fluxo Exceção	
Não há neste caso de uso.	

- **Especificações do Caso de Uso: Criar Lembrete**

Nome	UC10 – Criar Lembrete
Atores	Autor Principal: Policial Militar. Aktor Suporte: Banco de Dados.
Descrição	Caso de uso executado quando usuário criar um lembrete na agenda disponibilizada pelo aplicativo.
Pré-condições	- Usuário estar autenticado.
Pós-condições	Lembrete criado e notificará na data/hora programado.

Fluxo Básico	
Ações dos atores	Ações do Sistema
	1- Sistema disponibiliza um layout de agenda no aplicativo: data e hora.
2- Policial Militar insere lembrete em dia e hora que deseja ser notificado.	
3- Banco de Dados armazena a mensagem, data e hora do lembrete.	
	4- Sistema notificará o lembrete criado na data e hora escolhida pelo usuário.
Regras de Negócio	
Não há neste caso de uso.	
Fluxo Alternativo 1 – Cancelar um lembrete	
Ações dos atores	Ações do Sistema
1- Policial Militar pode cancelar a qualquer momento o lembrete.	
2- Banco de Dados não armazena caso processo não completado.	
	3- Aplicativo retorna 'lembrete não criado'.
Fluxo Exceção	

Não há neste caso de uso.

Strong Core – Training Worksheet Use Case:

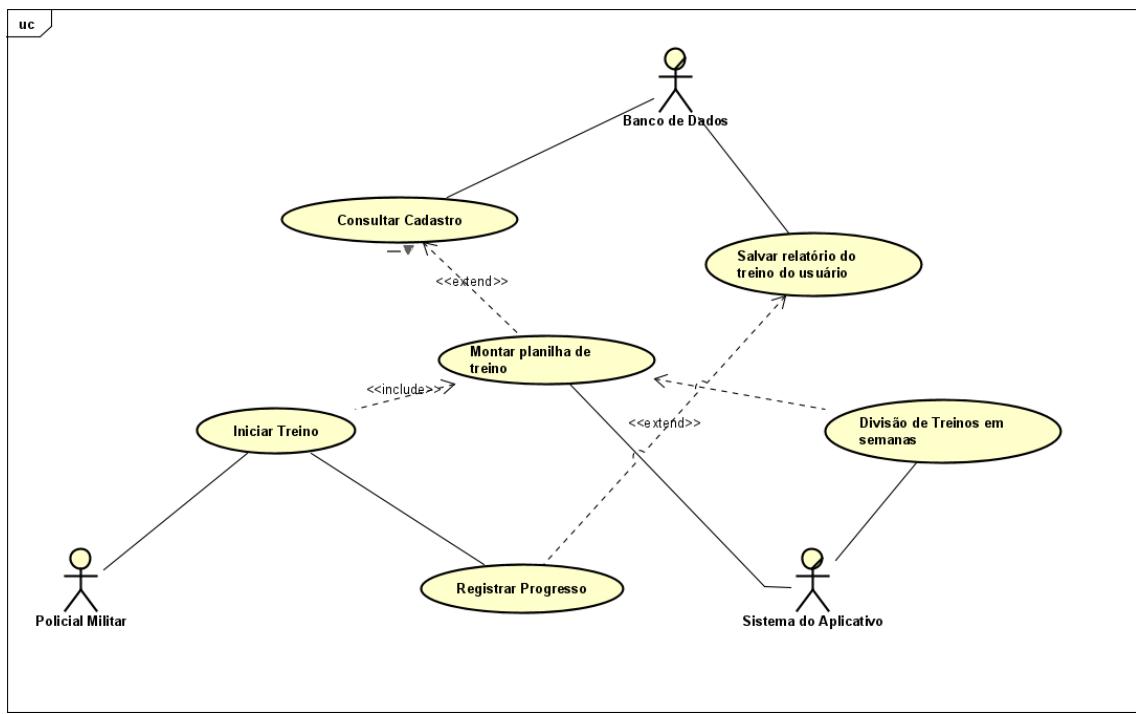


Figura 9. Caso de uso Planilha Treino aplicativo Strong Core.

- **Requisitos Funcionais Atendidos:**

#RF04 → O aplicativo Strong Core deve ser apto a montar uma tabela de treino tendo como objetivo fortalecer os pontos fracos de cada indivíduo.

#RF05 → O aplicativo deve mostrar a evolução do usuário nos treinos.

#RF07 → Os treinos devem ser divididos em níveis, de forma progressiva. Limitados por semana sem que o usuário possa pular ou burlar os treinos.

- **Especificações do Caso de Uso: Iniciar Treino**

Nome	UC11 – Iniciar Treino
Atores	Autor Principal: Policial Militar.
Descrição	Caso executado quando usuário começa a prática de exercícios através do aplicativo.
Pré-condições	- Usuário estar autenticado.
Pós-condições	Treino iniciado.

Fluxo Básico	
Ações dos atores	Ações do Sistema
1- Policial Militar clica em iniciar treino.	
	2- Sistema requisita UC12.
	3- Sistema libera os treinos.
Regras de Negócio	
Não há neste caso de uso.	
Fluxo Alternativo 1 – Fechar aplicativo no meio do treino	
Ações dos atores	Ações do Sistema
	1- Sistema requisita UC13.
	2- Sistema salva progresso do treino para continuar mais tarde.
Fluxo Exceção	
Não há neste caso de uso.	

- Especificações do Caso de Uso: Montar planilha de treino

Nome	UC12 – Montar planilha de treino
Atores	Autor Principal: Banco de Dados.
Descrição	Caso executado quando usuário ativar UC11.
Pré-condições	- Usuário estar autenticado. - Usuário iniciar treino UC11.
Pós-condições	Tabela de Treino criada.

Fluxo Básico	
Ações dos atores	Ações do Sistema

1- Banco de Dados confere cadastro físico do usuário.	
	2- Sistema captura os históricos físicos do usuário e gera um ciclo de treinos personalizado ao usuário.
	3- Sistema divide os treinos em níveis e semanas.
	4- Sistema retorna a Planilha de treinos para o usuário no Fluxo Básico 2 do UC11.
Regras de Negócio	
[RN008] – Usuário ter clicado em iniciar treino pela primeira vez.	
Fluxo Exceção	
Ações dos atores	Ações do Sistema
	Caso [RN008] não se aplique ao usuário, o sistema irá carregar UC13, para seguir o treino do usuário.

- **Especificações do Caso de Uso: Salvar relatório de treino do usuário**

Nome	UC13 – Salvar relatório de treino do usuário.
Atores	Autor Principal: Banco de Dados.
Descrição	Caso executado quando usuário terminar ou parar o treino.
Pré-condições	- Usuário estar autenticado. - Usuário ter iniciado o treino.
Pós-condições	Relatório de Treino armazenado.

Fluxo Básico	
Ações dos atores	Ações do Sistema
	1- Sistema envia dados do progresso dos exercícios completos pelo usuário.
2- Banco de Dados armazena dados fornecidos pelo sistema.	

	3- Sistema está apto a retomar/continuar de onde parou.
Regras de Negócio	
	Não há neste caso de uso.
Fluxo Exceção	
	Não há neste caso de uso.

Strong Core – Open Ticket Use Case:

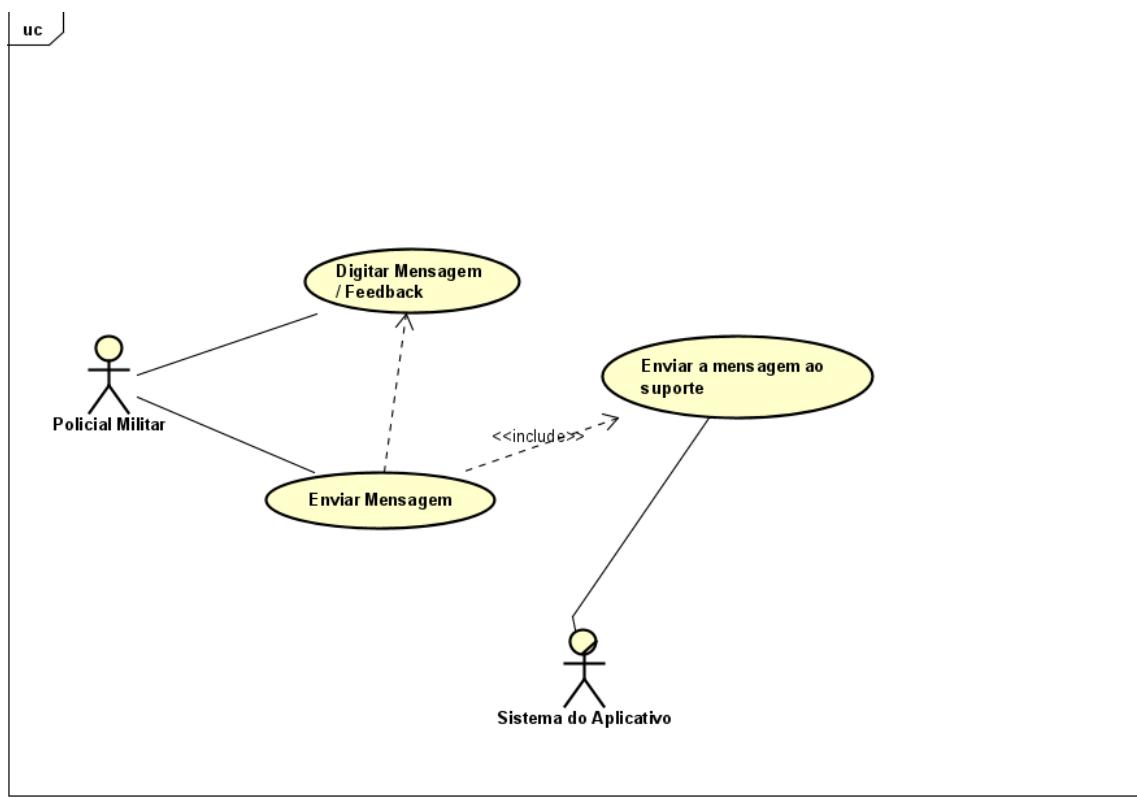


Figura 10. Caso de uso Abrir ticket aplicativo Strong Core.

- **Requisitos Funcionais Atendidos:**

#RF09 → O aplicativo deve ter uma seção para entrar em contato com o suporte técnico, seja para feedbacks / reclamações ou perguntas.

- **Especificações do Caso de Uso: Digitar Mensagem / Feedback**

Nome	UC014 – Digitar Mensagem / Feedback
Atores	Autor Principal: Policial Militar.
Descrição	Caso de uso executado quando o usuário insere mensagem para enviar.
Pré-condições	- Usuário deve estar autenticado.
Pós-condições	Mensagem preenchida.

Fluxo Básico	
Ações dos atores	Ações do Sistema
	1- Sistema mostra tela para usuário digitar mensagem.
2 – O Policial Militar digita a mensagem.	
Regras de Negócio	
[RN009] – Campo da mensagem não pode ser vazio.	
Fluxo Alternativo 1 – Cancelar mensagem	
Ações dos atores	Ações do Sistema
	1- Sistema volta ao menu principal e ignora qualquer coisa digitada pelo usuário.
Fluxo Exceção	
Ações dos atores	Ações do Sistema
	1- Caso [RN009] esteja vazio solicitar usuário a inserir mensagem.

- Especificações do Caso de Uso: Enviar Mensagem

Nome	UC015 – Enviar Mensagem
Atores	Autor Principal: Policial Militar.
Descrição	Caso de uso executado quando o usuário clica para enviar mensagem
Pré-condições	- Usuário deve estar autenticado. - Usuário deve ter completo UC14.
Pós-condições	Mensagem enviada.

Fluxo Básico	
Ações dos atores	Ações do Sistema

1- Policial Militar clica para enviar mensagem ao suporte.	
	2- Sistema captura dados da mensagem e envia ao suporte.
Regras de Negócio	
[RN010] – UC14 deve ser válido.	
Fluxo Exceção	
Ações dos atores	Ações do Sistema
	2- Caso [RN010] não seja válido solicitar usuário a inserir mensagem.

Strong Core – Class Diagram:

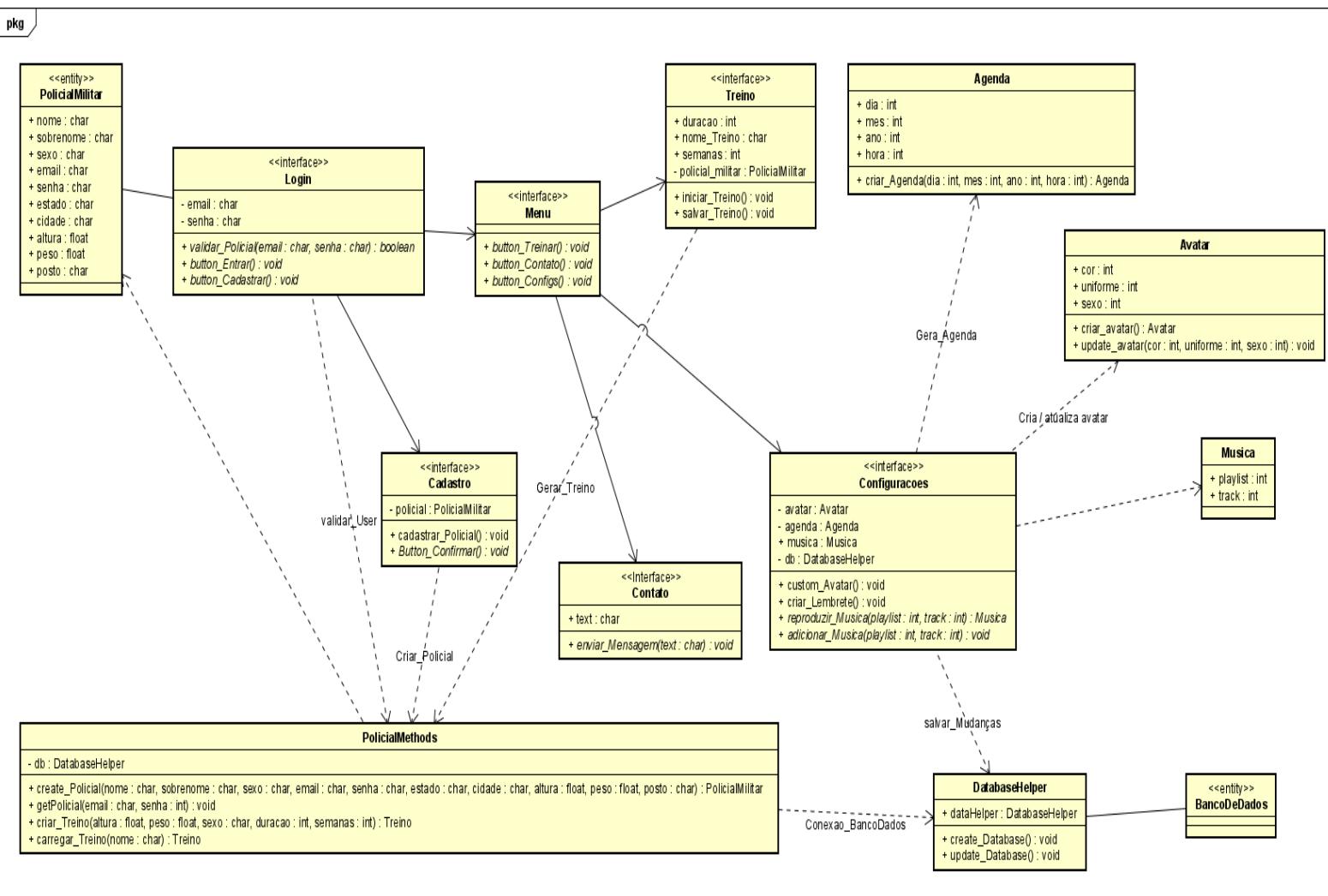


Figura 11. Diagrama de Classes Strong Core.

Strong Core – Dicionário de Informações:

- **Classe PolicialMilitar:**

PolicialMilitar: Usuário que irá interagir com o aplicativo.					
Atributo	Descrição	Tamanho	Tipo	Formato	Domínio
nome	Primeiro nome.	15	Alfanumérico	1{X}15	Contínuo
sobrenome	Refere-se ao segundo nome.	30	Alfanumérico	1{X}30	Contínuo
sexo	Sexo do Usuário.	1	Alfabético	A	Discreto F- Feminino M- Masculino
email	Endereço de e-mail.	40	Alfanumérico	1{X}40	Contínuo
senha	Senha para login no aplicativo.	15	Alfanumérico	1{X}15	Contínuo
estado	Refere-se ao estado.	2	Alfabético	AA	Discreto PR - Paraná
cidade	Cidade do usuário.	20	Alfanumérico	1{X}20	Contínuo
altura	Altura do usuário.	4	Numérico	99,99	Contínuo
peso	Peso do usuário.	4	Numérico	99,99	Contínuo
posto	Cargo ocupado pelo usuário na PMPR.	20	Alfanumérico	1{X}20	Contínuo

- **Classe PolicialMethods:**

PolicialMethods: Responsável por métodos e por conexão com banco de dados.					
Atributo	Descrição	Tamanho	Tipo	Formato	Domínio
db	Conexão ao				

	banco de dados.		DatabaseHelper		
--	-----------------	--	----------------	--	--

- **Classe Login:**

Login: Classe da interface do login e responsável por requisitar validação.					
Atributo	Descrição	Tamanho	Tipo	Formato	Domínio
email	Email digitado pelo usuário.	40	Alfanumérico	1{X}40	Contínuo
senha	Senha digitada pelo usuário.	15	Alfanumérico	1{X}15	Contínuo

- **Classe Cadastro:**

Cadastro: Classe da interface cadastro, responsável por retornar um objeto policial novo ao sistema.					
Atributo	Descrição	Tamanho	Tipo	Formato	Domínio
policial	Objeto do tipo PolicialMilitar.		PolicialMilitar		

- **Classe Menu:**

Menu: Classe da interface menu, pela navegação dentro do aplicativo, apenas contendo métodos.					
Atributo	Descrição	Tamanho	Tipo	Formato	Domínio

- **Classe Contato:**

Contato: Classe da interface contato, responsável por enviar a mensagem do usuário ao suporte.					
Atributo	Descrição	Tamanho	Tipo	Formato	Domínio
text	Texto da mensagem inserida pelo usuário.	1000	Alfanumérico	20{X}1000	Contínuo

- **Classe Treino:**

Treino: Classe da interface treino, responsável pela requisição de gerar ou carregar um trino já existente.					
Atributo	Descrição	Tamanho	Tipo	Formato	Domínio
duracao	Minutos de duração do treino.	2	Numérico	99	Contínuo
nome_Treino	Região que o treino é especificada, exemplo: Lombar.	30	Alfanumérico	1{X}30	Contínuo
semanas	Divisão em semanas dos treinos.	2	Numérico	99	Contínuo
policial_militar	Atributo do tipo PolicialMilitar com intuito de usar alguns dados do usuário para geração do treino.		PolicialMilitar		

- **Classe Configuracoes:**

Treino: Classe da interface treino, responsável pela requisição de gerar ou carregar um trino já existente.					
Atributo	Descrição	Tamanho	Tipo	Formato	Domínio
avatar	Customiza objeto avatar.		Avatar		
agenda	Cria objeto Agenda.		Agenda		

musica	Adiciona objeto Musica ou reproduz.		Musica		
db	Solicita salvar alterações no Banco de Dados.		DatabaseHelper		

- **Classe DatabaseHelper:**

DatabaseHelper: Responsável pela conexão com o Banco de Dados diretamente.

Atributo	Descrição	Tamanho	Tipo	Formato	Domínio
db	Conexão ao banco de dados.		DatabaseHelper		

- **Classe Agenda:**

Agenda: Classe responsável por criar e retornar um objeto Agenda ao usuário.

Atributo	Descrição	Tamanho	Tipo	Formato	Domínio
dia	Refere-se ao dia.	2	Numérico	99	Contínuo
mes	Refere-se ao mês.	2	Numérico	99	Contínuo
ano	Refere-se ao ano.	4	Numérico	99	Contínuo
hora	Hora do lembrete.	4	Numérico	99:99	Contínuo

- **Classe Avatar:**

Agenda: Classe responsável por criar/retornar e atualizar um objeto Avatar

Atributo	Descrição	Tamanho	Tipo	Formato	Domínio
cor	Tonalidade de pele do personagem.	1	Numérico	9	Discreto 1- Claro 2- Escuro 3- Amarelo
uniforme	Roupa (Skin) do personagem.	1	Numérico	9	Discreto 1- Azul 2- Preto

					3- Vermelho
sexo	Sexo do personagem.	1	Numérico	9	Discreto 1- Masculino 2- Feminino

- **Classe Musica:**

Musica: Classe responsável por armazenar os atributos de uma música.					
Atributo	Descrição	Tamanho	Tipo	Formato	Domínio
Playlist	Conjunto de sequências de músicas a serem reproduzidas.	2	Numérico	99	Contínuo
track	Número da música no sistema.	2	Numérico	99	Contínuo

Diagrama de Sequência Login:

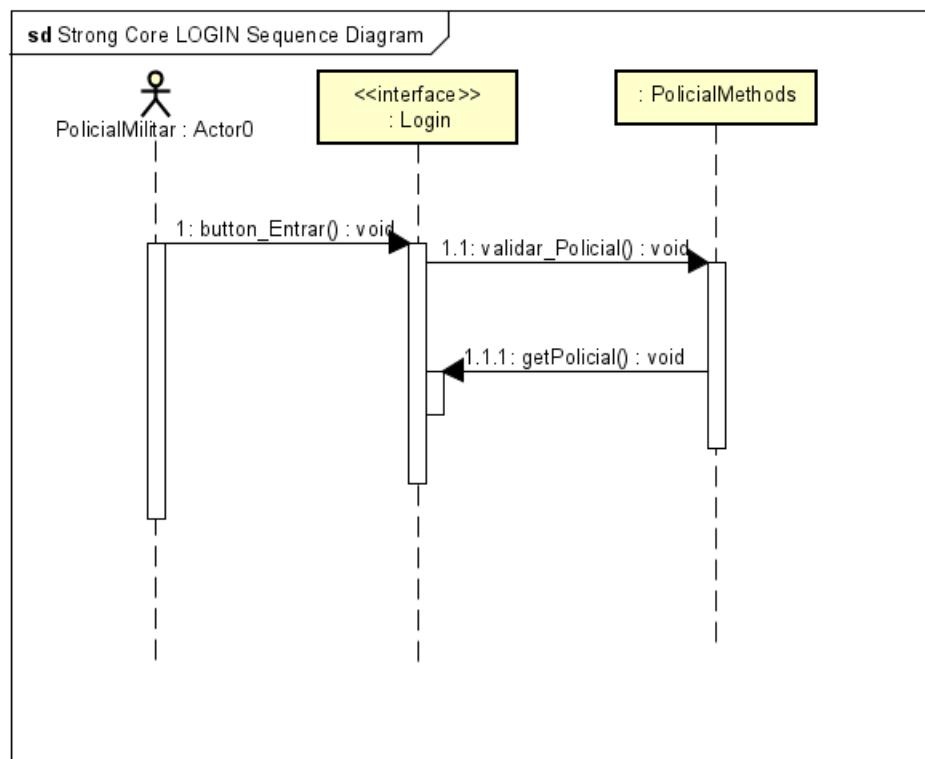


Figura 12. Diagrama de Sequência Login Strong Core.

Diagrama de Sequência Register:

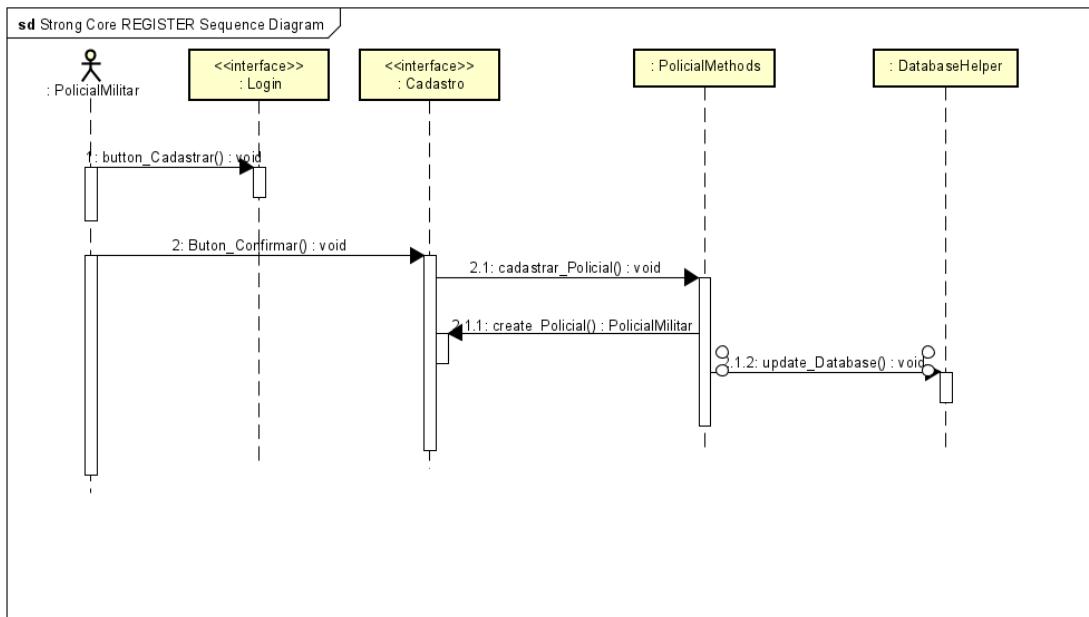


Figura 13. Diagrama de Sequência Register Strong Core.

Diagrama de Sequência Configurations:

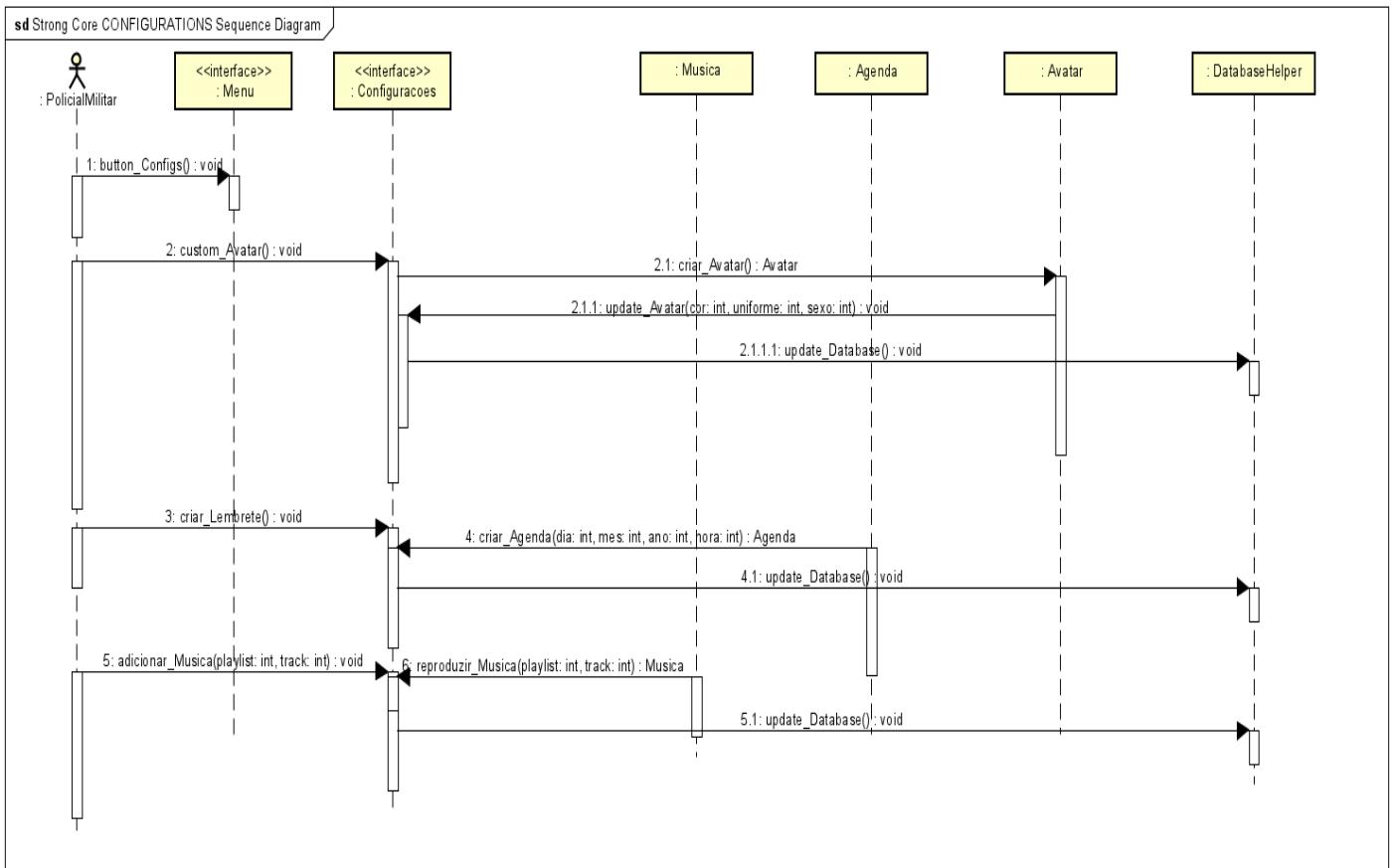


Figura 14. Diagrama de Sequência Configurations Strong Core.

Diagrama de Sequência Training Worksheet:

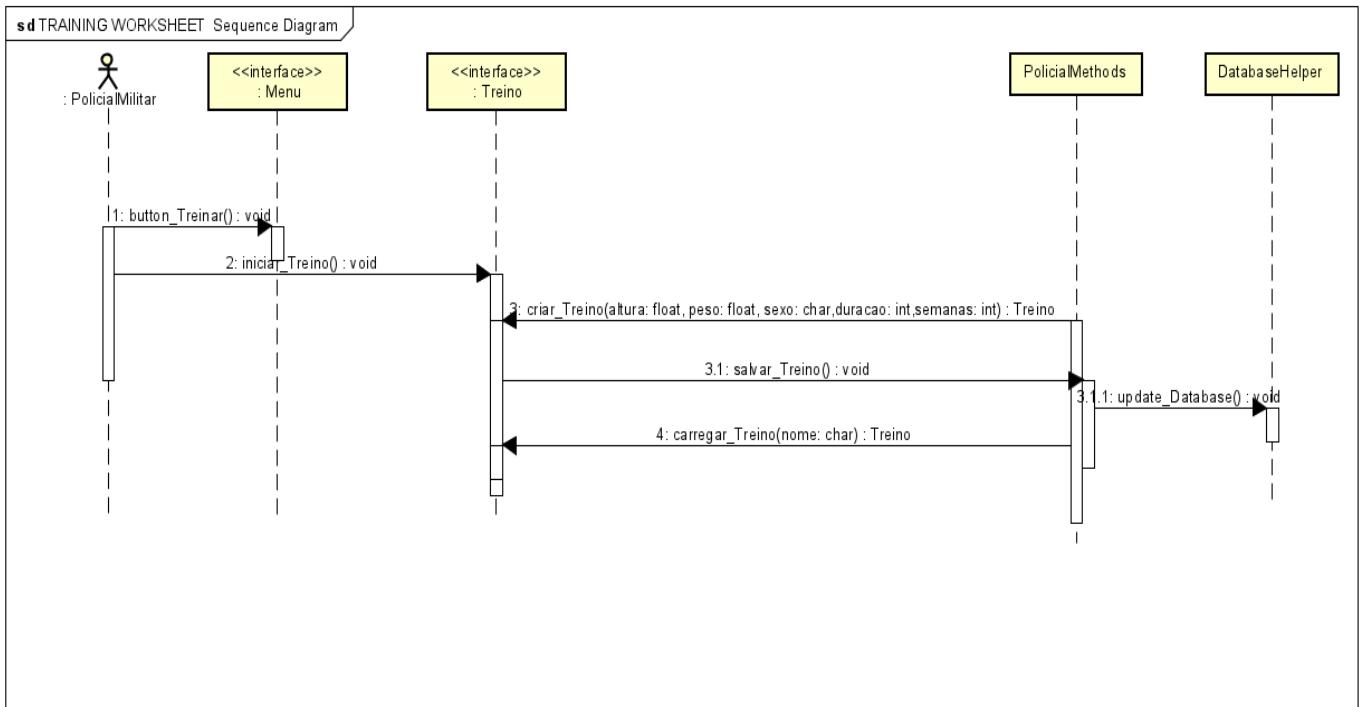


Figura 15. Diagrama de Sequência Training Worksheet Strong Core.

Diagrama de Sequência Training Worksheet:

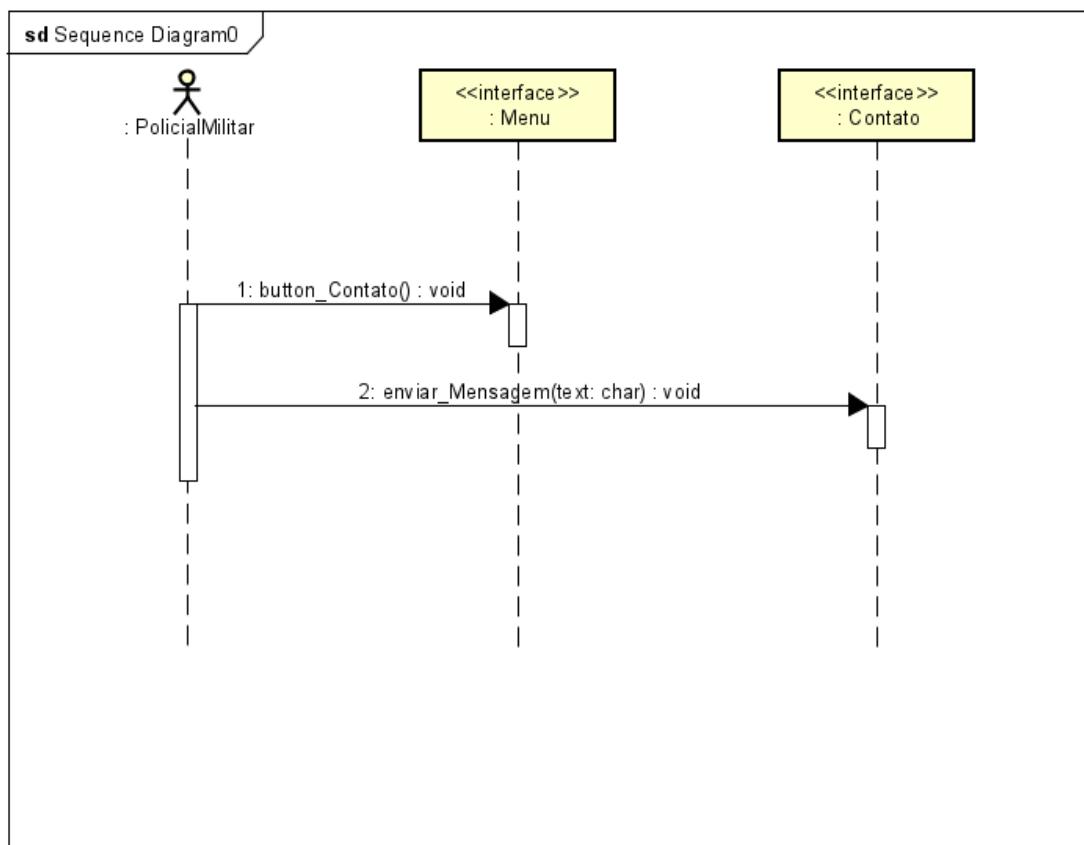


Figura 16. Diagrama de Sequência Open Ticket Strong Core.

5 CONCLUSÕES

O projeto Strong Core teve seus altos e baixos, ao começar o projeto estava bem visível quais benefícios econômicos e organizacionais poderiasse proporcionar para a organização da Polícia Militar do Paraná a médio/longo prazo com a aquisição do aplicativo. Já em relação as questões operacionais e técnicas foram várias vezes debatidas e analisadas para chegar em uma definição de quais estariam de acordo para melhor eficácia de suportar as necessidades da **PMPR**.

A empresa Soft Software fez a coleta de requisitos para o sistema do aplicativo, para com isso começar a modelagem do projeto Strong Core como apresentado neste documento. No processo de modelagem foi usado conceitos de engenharia reversa e apoiado por referencias bibliográficas de Engenharia Software e Gerência de Projetos.

Por fim, a empresa Soft Software agradece a oportunidade de desenvolver esse amplo projeto que atende a missão da organização **PMPR**.

6 REFERÊNCIAS BIBLIOGRÁFICAS

SOMMERVILLE, Ian. **Engenharia de software** 9 ed. São Paulo, SP: Pearson Prentice Hall. ISBN: 9788579361081, 2011.

PRESSMAN, R. S **Engenharia de Software - Uma abordagem Profissional** - 8 ed. Porto Alegre, RS: Bookman/Amgh Editora. ISBN: 9788580555332, 2016.

Frakes, W.; W. Kang, K. **Software Reuse Research: Status and Future**. IEEE Transactions on Softare Engineering, 2005.

Bezerra, Eduardo. **Princípio de Análise e Projeto de Sistemas com UML**. Elsevier Editora Campus, 2006.

PMI. **Um guia do conhecimento em gerenciamento de projetos**. Guia PMBOK® 6a. ed. EUA: Project Management Institute, 2017.