
**Alexssander Siqueira Lima
Andre Luiz de Almeida Camargo
Guilherme Gustavo Gevaerd
Jonathan Cavali Scheffer
Marcelo Rudolf Junior**

**Sistema de Análise de Veículos Elétricos
Documento de Arquitetura de Software**

Versão 1.0

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Histórico da Revisão

Data	Versão	Descrição	Autor
29/mai/19	1.0	Primeiro documento de arquitetura	Alexssander Siqueira Lima Andre Luiz de Almeida Camargo Guilherme Gustavo Gevard Jonathan Cavali Scheffer Marcelo Rudolf Junior

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Índice Analítico

1. INTRODUÇÃO	5
1.1 Finalidade	5
1.2 Definições, Acrônimos e Abreviações	5
1.3 Visão Geral	5
2. DECLARAÇÃO DO PROBLEMA	6
3. VISÃO GERAL DOS REQUISITOS	6
REQ001 - Isolamento galvânico do processador	6
REQ002 - Tensão de operação	7
REQ003 - Temperatura de operação	7
REQ004 - Resistência à trepidação	7
REQ005 - Microprocessador	7
REQ006 - Resistor de 120 Ohms	7
REQ007 - Buffer circular de memória externa	7
REQ008 - Linguagem de programação do microprocessador	8
REQ009 - Linguagem de programação do software	8
REQ010 - Controlar timestamps de dados	8
REQ011 - Alarme de falha de comunicação	8
REQ012 - Conector OBD 2	9
REQ013 - Camadas de atuação do dispositivo (Modelo OSI)	9
REQ014 - BaudRates de operação da rede CAN	9
REQ015 - Interfaces de comunicação	9
REQ016 - Memória de armazenamento não volátil	10
REQ017 - Led indicador de estado	10
REQ018 - Rotação do motor	10
REQ019 - Corrente do motor	10
REQ020 - Tensão da bateria	10
REQ021 - Estado do inversor	11
REQ022 - Torque do motor	11
REQ023 - Corrente da bateria	11
REQ024 - Estimativa de duração da bateria	11
REQ025 - Estado de carga da bateria	12
REQ026 - Sistema de gerenciamento de baterias	12
REQ027 - Estado do relé	12
REQ028 - Verificar limites de temperatura	12
REQ029 - Verificar histórico de velocidades	12
REQ030 - Geração e exibição de gráficos	13
REQ031 - Salvar Template	13
REQ032 - Exportação de Template em formato CSV	13
REQ033 - Relatório de desempenho gerais	13
REQ034 - Curva de performance esperada	14
REQ035 - Leitura dos códigos de falha fornecidos pela central eletrônica	14
REQ036 - Descrição dos códigos de falha	14
REQ037 - Pesquisa por códigos de falha	14
REQ038 - Controle de fluxo em pesquisa de código de falha	15
REQ039 - Verificar estado de memória	15
REQ040 - Mostrar espaço restante da memória	15
REQ041 - Tentativas de comunicação	15
REQ042 - Integridade da mensagem	16

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

REQ043 - Dicionário de objetos	16
4. UML	18
4.1 Descrição de Caso de Uso	18
4.2 Diagrama Caso de Uso	42
4.3 Diagrama de Classes	43
4.4 Diagrama de Sequência	44
4.5 Diagrama de Atividades	57
5. ARQUITETURA DE COMPONENTES	73
5.1 Padrão de arquitetura	73
6. PROTÓTIPO	74
7. ITEM DE CRIATIVIDADE	81
8. RESPONSABILIDADES	85

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Documento de Arquitetura de Software

1. INTRODUÇÃO

1.1 Finalidade

O Documento de Arquitetura de Software tem como principal papel a definição de modelos comportamentais e ambientais do projeto a ser desenvolvido. A partir dessa modelagem será possível visualizar como o software funcionará, como o usuário conseguirá interagir com ele, como o será o comportamento dinâmico do sistema e também o conceito de como será implementado. O documento é dividido nas seguintes partes: Declaração do problema, Visão geral dos requisitos, UML, Arquitetura de componentes, Protótipo, Item de Criatividade e Responsabilidades.

1.2 Definições, Acrônimos e Abreviações

BMS – *Battery Management System* (Sistema de Gerenciamento de Bateria)

ECU – *Engine Control Unit* (Unidade de Controle do Motor)

CAN – *Controller Area Network*

OBDII – *On Board Diagnostics II*

OTA – *Over The Air*

SADVE UI – Interface do Usuário do sistema SADVE.

UML – *Unified Modeling Language* (Linguagem de Modelagem Unificada)

USB – *Universal Serial Bus* (Barramento Serial Universal)

1.3 Visão Geral

O documento é dividido nas seguintes partes:

- Declaração do problema, onde é descrito o problema levantado, explorando o cenário onde o sistema atuará.
- Visão geral dos requisitos, onde estão descritos os requisitos funcionais e não-funcionais do sistema, levantados no Estudo de Viabilidade, revisada.
- UML, onde estão os Casos de Uso, extraídos a partir dos requisitos funcionais do item anterior, e os diagramas que descrevem e modelam o sistema a ser desenvolvido. O sistema está modelado por meio de diagramas comportamentais (Diagramas de Casos de Uso, de Sequência e de Atividades), e também por meio de diagramas estruturais (Diagrama de Classe).
- Arquitetura de componentes, onde será apresentado a arquitetura do sistema por meio de um padrão de arquitetura baseado em um sistema em camadas, demonstrando a comunicação entre os componentes de entrada e o sistema.

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

- Protótipo, onde será apresentado um Manual do Usuário, com o passo-a-passo para a utilização do sistema, com o auxílio de imagens protótipo da interface do sistema.

2. DECLARAÇÃO DO PROBLEMA

Embora o primeiro carro com motor elétrico seja de 1835, o veículo elétrico só está se consolidando no mercado atualmente, visto que as grandes empresas recentemente começaram projetos para explorar a área. Devido a tal, é um mercado inexplorado, onde somente grandes empresas e startups do ramo possuem algum conhecimento.

Para a análise do mercado, as funcionalidades dos potenciais concorrentes foram separadas em Hardware, Firmware e Software, utilizando como comparação os produtos destinados para carros a combustão, abstraindo que os problemas serão os mesmos caso as empresas decidam abranger carros elétricos.

Para fins organizacionais, criou-se um diagnóstico das necessidades e problemas a partir das categorias mencionadas acima (Hardware, Firmware e Software) sendo que, para cada uma delas, exploramos os pontos fortes e fracos do mercado para focarmos no diferencial do nosso produto em relação ao ambiente atual.

Em relação ao Software, o mercado carece de interfaces de usuário de fácil compreensão na explicitação dos defeitos e na análise de dados referentes à performance do carro tanto momentânea quanto na no registro de informações e na geração de relatórios temporais. Em relação ao Firmware, uma das maiores dificuldades é a padronização do protocolo, sendo que existem várias alterações de fabricante para fabricante em relação aos modos de operação do sistema OBDII.

Quanto a hardware, existe uma grande diversidade para fabricantes de Analisadores de Redes CAN para veículos na maneira com que os dados são transmitidos, podendo ser tanto OTA (Over The Air) quando cabeados. Alguns fabricantes também utilizam protocolo IP fornecendo interfaces Ethernet e Wi-fi para seus produtos, fazendo com que o cliente possa escolher quais módulos deseja comprar.

Baseado nas dificuldades citadas anteriormente, e no fato de que os veículos elétricos, por estarem se consolidando no mercado recentemente, carecem de ferramentas para análises e manutenção, foi visto uma oportunidade de desenvolver um produto para análise de veículos elétricos.

3. VISÃO GERAL DOS REQUISITOS

REQ001 – Isolamento galvânico do processador			
PRIORIDADE:	Média	ESTABILIDADE	Média
SOLICITANTE:	Equipe de hardware	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Não funcional	IMPACTO NA ARQUITETURA:	Alto
Descrição:	<i>Por conta do sistema de Força do veículo elétrico Powertrain poder operar em alta tensão, existe a possibilidade do transceptor CAN sofrer interferências externas, gerando erros de leitura e diminuição da qualidade do sinal. Para contornar esta situação, é recomendável que o transceptor CAN possua um Circuito Integrado ISO1048 capaz de diminuir a influência destas interferências, isolando o transceptor e evitando possíveis falhas.</i>		

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

REQ002– Tensão de operação			
PRIORIDADE:	Alta	ESTABILIDADE	Média
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Não Funcional	IMPACTO NA ARQUITETURA:	Alto
Descrição:	<i>O dispositivo deverá atuar somente nas tensões fornecidas pelo sistema sendo elas entre +12V e +24V via conector OBD2.</i>		

REQ003– Temperatura de Operação			
PRIORIDADE:	Média	ESTABILIDADE	Baixa
SOLICITANTE:	Equipe de Hardware	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Não funcional	IMPACTO NA ARQUITETURA:	Baixo
Descrição:	<i>O dispositivo deverá conseguir operar em uma temperatura de -20°C até +120°C, considerando que o veículo poderá estar sob condições de inverno rigoroso ou estacionado sob sol forte.</i>		

REQ004– Resistência à trepidação			
PRIORIDADE:	Baixa	ESTABILIDADE	Baixa
SOLICITANTE:	Equipe de Hardware	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Não funcional	IMPACTO NA ARQUITETURA:	Baixo
Descrição:	<i>O dispositivo deverá conseguir suportar a vibração em função das diferentes condições da dinâmica veicular como, por exemplo, as frequências geradas ao dirigir, do sistema de suspensão, ao frear o carro e etc.</i>		

REQ005 – Microprocessador			
PRIORIDADE:	Alta	ESTABILIDADE	Alta
SOLICITANTE:	Equipe de Firmware	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Não funcional	IMPACTO NA ARQUITETURA:	Alto
Descrição:	<i>O microprocessador utilizado deve possuir uma arquitetura ARM Cortex que cubra todas as funcionalidades de periféricos descritas nos requisitos.</i>		

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

REQ006 – Resistores de 120 Ohms			
PRIORIDADE:	Alta	ESTABILIDADE	Alta
SOLICITANTE:	Equipe de hardware	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Não funcional	IMPACTO NA ARQUITETURA:	Baixo
Descrição:	<i>Dois resistores de 120 Ohms nas extremidades do barramento para garantir que o transiente passivo da onda lógica CAN esteja correto. É de uso obrigatório os dois resistores com o risco de o barramento cair se houver um número superior ou inferior do que o especificado aqui.</i>		

REQ007 – Buffer circular da memória externa			
PRIORIDADE:	Alta	ESTABILIDADE	Média
SOLICITANTE:	Equipe de Firmware	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Não Funcional	IMPACTO NA ARQUITETURA:	Alto
Descrição:	<i>A memória externa deverá operar com um buffer circular, reciclando seus dados quando o limite de armazenamento for alcançado. Toda vez que o espaço é insuficiente, o firmware deverá sobreescrever sua fila apagando os dados do início do buffer.</i>		

REQ008 – Linguagem de programação do microprocessador			
PRIORIDADE:	Baixa	ESTABILIDADE	Alta
SOLICITANTE:	Equipe de Firmware	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Não Funcional	IMPACTO NA ARQUITETURA:	Baixo
Descrição:	<i>O firmware deverá ser programado na linguagem estruturada C.</i>		

REQ009 – Linguagem de programação do software			
PRIORIDADE:	Baixa	ESTABILIDADE	Alta
SOLICITANTE:	Equipe de Software	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Não Funcional	IMPACTO NA ARQUITETURA:	Baixo
Descrição:	<i>O software deverá ser programado na linguagem de programação orientada a objetos Python.</i>		

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

REQ010 – Controlar <i>timestamps</i> de dados			
PRIORIDADE:	Baixa	ESTABILIDADE	Alta
SOLICITANTE:	Equipe de Software	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Funcional	IMPACTO NA ARQUITETURA:	Baixo
Descrição:	<i>A partir do sinal gerado do cristal de tempo real, o microprocessador deverá salvar os dados lidos pelo barramento CAN e armazenar na memória externa com uma timestamp.</i>		

REQ011 – Alarme de falha de comunicação			
PRIORIDADE:	Alta	ESTABILIDADE	Média
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Funcional	IMPACTO NA ARQUITETURA:	Alto
Descrição:	<i>Indica ao usuário via interface gráfica com um ícone e uma descrição do erro que indicam explicitamente que houve um erro durante a comunicação.</i>		

REQ012 – Conector OBD2			
PRIORIDADE:	Alta	ESTABILIDADE	Alta
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Não Funcional	IMPACTO NA ARQUITETURA:	Alto
Descrição:	<i>O padrão de conectores OBD2 é utilizado no mercado de Analisadores de Rede CAN por conta da sua utilização pelas montadoras de veículos. Este conector possui pinos específicos para alimentação e transferência de dados e pode trabalhar em diferentes modos de operação, podendo variar tanto a tensão quanto as taxas de transmissão, como também pode ser utilizado com outros protocolos de comunicação. Será utilizado para alimentar e conduzir os dados para o SADVE.</i>		

REQ013 – Camadas de atuação do dispositivo (Modelo OSI)			
PRIORIDADE:	Alta	ESTABILIDADE	Alta
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Não Funcional	IMPACTO NA ARQUITETURA:	Alto
Descrição:	<i>O dispositivo deverá atuar somente nas camadas físicas e de enlace seguindo o padrão CAN conforme a ISO 11898.</i>		

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

REQ014 – BaudRates de operação da rede CAN			
PRIORIDADE:	Alta	ESTABILIDADE	Média
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Não funcional	IMPACTO NA ARQUITETURA:	Alto
Descrição:	<i>Os BaudRates de operação do dispositivo devem estar entre 10,4kbps até 500,0 kbps desde que pertençam aos padrões utilizados pelas ISO 15765, ISO 14230 e SAE J1939.</i>		

REQ015 – Interfaces de comunicação			
PRIORIDADE:	Alta	ESTABILIDADE	Alta
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Não Funcional	IMPACTO NA ARQUITETURA:	Alto
Descrição:	<i>O SADVE irá transmitir seus dados lidos da rede CAN através de uma interface USB.</i>		

REQ016 – Memória de armazenamento não volátil			
PRIORIDADE:	Alta	ESTABILIDADE	Alta
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Não Funcional	IMPACTO NA ARQUITETURA:	Alto
Descrição:	<i>O SADVE deverá ser capaz de trabalhar com uma memória flash externa para armazenar dados por pelo menos 10 horas gravadas de maneira cíclica.</i>		

REQ017 – LED indicador de estado			
PRIORIDADE:	Alta	ESTABILIDADE	Alta
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Funcional	IMPACTO NA ARQUITETURA:	Médio
Descrição:	<i>O dispositivo deverá conter dois LEDS indicadores de estado. Um LED verde, responsável por representar o estado de Ligado ao usuário e um LED azul para indicar a transmissão de pacotes, piscando após a transmissão de cada pacote, desde que possa ter boa visibilidade para o usuário(até 35Hz).</i>		

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

REQ018 – Rotação do motor			
PRIORIDADE:	Alta	ESTABILIDADE	Alta
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Funcional	IMPACTO NA ARQUITETURA:	Médio
Descrição:	<i>O usuário poderá selecionar na interface gráfica a opção de visualizar a evolução temporal da rotação do motor, mostrando picos e vales relacionando como todo o sistema de transmissão respondeu na velocidade.</i>		

REQ019 – Corrente do Motor			
PRIORIDADE:	Alta	ESTABILIDADE	Alta
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Funcional	IMPACTO NA ARQUITETURA:	Médio
Descrição:	<i>O usuário poderá selecionar na interface gráfica a opção de visualizar o histórico da corrente no motor, destacando quando a corrente do motor extrapola os limites de operação, relacionando se o motor estava travado, parado ou transmitindo energia mecânica. O usuário poderá também visualizar quando houver picos de corrente.</i>		

REQ020 – Tensão da bateria			
PRIORIDADE:	Alta	ESTABILIDADE	Alta
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Funcional	IMPACTO NA ARQUITETURA:	Médio
Descrição:	<i>O usuário terá a disponibilidade de verificar o histórico da tensão bateria na interface gráfica, buscando comportamentos irregulares ao longo da utilização. Será relacionado com a carga da bateria para identificar a vida útil da bateria.</i>		

REQ021 – Estado do inversor			
PRIORIDADE:	Alta	ESTABILIDADE	Alta
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Funcional	IMPACTO NA ARQUITETURA:	Médio
Descrição:	<i>O histórico de estados do inversor poderá ser acessado pelo usuário, mostrando quanto tempo o inversor esteve funcionando, travado ou desligado com suas respectivas timestamps. Será relacionado com a tensão e a corrente do motor para verificação de avarias.</i>		

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

REQ022 – Torque do motor			
PRIORIDADE:	Alta	ESTABILIDADE	Alta
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Funcional	IMPACTO NA ARQUITETURA:	Médio
Descrição:	<i>O usuário deve ter acesso ao histórico do torque do motor, correlacionar a tensão da bateria, a velocidade e a rotação do motor, gerando o gráfico de performance de todo o sistema de tração do motor.</i>		

REQ023 – Corrente da bateria			
PRIORIDADE:	Alta	ESTABILIDADE	Alta
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Funcional	IMPACTO NA ARQUITETURA:	Médio
Descrição:	<i>A bateria deverá possuir histórico de corrente acessível ao usuário, demonstrando o comportamento ao longo do tempo de utilização.</i>		

REQ024 – Estimativa de duração da bateria			
PRIORIDADE:	Alta	ESTABILIDADE	Alta
SOLICITANTE:	Persona	REQ. ORIGEM:	REQ023
TIPO DO REQUISITO:	Funcional	IMPACTO NA ARQUITETURA:	Médio
Descrição:	<i>A partir dos dados de corrente da bateria por um determinado período de tempo, o sistema deverá estimar a carga da bateria e o tempo restante de duração.</i>		

REQ025 - Estado de carga da bateria			
PRIORIDADE:	Alta	ESTABILIDADE	Baixa
SOLICITANTE:	Persona	REQ. ORIGEM:	REQ023 e REQ024
TIPO DO REQUISITO:	Funcional	IMPACTO NA ARQUITETURA:	Alto
Descrição:	<i>Será possível acessar o histórico do estado de carga da bateria, com o desempenho médio em uso e tempo até carregamento completo.</i>		

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

REQ026 – Sistema de gerenciamento de baterias			
PRIORIDADE:	Média	ESTABILIDADE	Média
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Funcional	IMPACTO NA ARQUITETURA:	Médio
Descrição:	<i>O usuário terá acesso ao sistema de gerenciamento das baterias e verificar se o sistema está operando corretamente. A opção deverá indicar também se teve caso de correntes, tensões ou temperaturas fora do ótimo para funcionamento do BMS (sistema de gerenciamento de baterias), ou se o aterramento está inadequado.</i>		

REQ027 – Estado do Relé			
PRIORIDADE:	Média	ESTABILIDADE	Alta
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Funcional	IMPACTO NA ARQUITETURA:	Médio
Descrição:	<i>O usuário poderá acessar o histórico do estado dos relés do sistema de gerenciamento das baterias e dos inversores. Destacando quando não atuaram, relacionando as características da bateria e do motor, identificando possíveis avarias.</i>		

REQ028 – Verificar limites de temperatura			
PRIORIDADE:	Alta	ESTABILIDADE	Baixa
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Funcional	IMPACTO NA ARQUITETURA:	Médio
Descrição:	<i>O usuário terá acesso às temperaturas máximas e mínimas de todo o sistema (freio, motor, bateria, inversor) durante todo o tempo de uso. Haverá a opção de procurar pontos críticos que excederam a temperatura ótima de uso.</i>		

REQ029 – Verificar histórico de velocidades			
PRIORIDADE:	Alta	ESTABILIDADE	Média
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Funcional	IMPACTO NA ARQUITETURA:	Alto
Descrição:	<i>O usuário terá acesso à evolução da velocidade no histórico, correlacionando ao torque e a rotação do motor, emitindo a performance do carro sinalizando se há perda de potência e avarias no sistema de tração.</i>		

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

REQ030 – Gerar e exibir de gráficos			
PRIORIDADE:	Alta	ESTABILIDADE	Alta
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Funcional	IMPACTO NA ARQUITETURA:	Baixo
Descrição:	<i>O usuário poderá criar um template. Este template poderá possuir gráficos de maneira customizável pelo usuário. Estes gráficos são obtidos a partir de um botão na interface gráfica com o nome “Criar Gráfico” em que o usuário seleciona os dados fornecidos pelo sistema para compor os eixos cartesianos de interesse.</i>		

REQ031 – Salvar Template			
PRIORIDADE:	Média	ESTABILIDADE	Alta
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Funcional	IMPACTO NA ARQUITETURA:	Médio
Descrição:	<i>O usuário poderá salvar um template previamente configurado conforme o REQ030, a partir de um botão chamado “Salvar Template” na interface gráfica. Este template salvo poderá ser reutilizado pelo usuário para facilitar a visualização das curvas desejadas.</i>		

REQ032 – Exportação de Template em formato PDF			
PRIORIDADE:	Baixa	ESTABILIDADE	Alta
SOLICITANTE:	Equipe de Software	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Funcional	IMPACTO NA ARQUITETURA:	Médio
Descrição:	<i>Todos os Templates salvos poderão ser exportados em formato Comma Separated Values(.csv) para que possuam boa interoperabilidade.</i>		

REQ033 – Relatório de desempenho gerais			
PRIORIDADE:	Alta	ESTABILIDADE	Alta
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Funcional	IMPACTO NA ARQUITETURA:	Médio
Descrição:	<i>Mostrar um template padrão com características gerais da evolução temporal do veículo, da Eficiência, das Temperaturas (baterias, motores, pneus, sistema de arrefecimento, sistema de freios), Tensão da bateria, Corrente Consumida, Velocidade do veículo, Torque do motor. Este template poderá ser utilizado como base para criação de outros templates pelo usuário.</i>		

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

REQ034 – Curva de performance esperada			
PRIORIDADE:	Média	ESTABILIDADE	Baixa
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Funcional	IMPACTO NA ARQUITETURA:	Médio
Descrição:	<i>O sistema deverá ser capaz de mostrar ao usuário a curva de performance esperada de cada modelo de veículo e relacionar com os dados lidos, podendo ser adicionados aos gráficos.</i>		

REQ035 – Leitura de códigos de falha fornecidos pela central eletrônica			
PRIORIDADE:	Alta	ESTABILIDADE	Média
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Funcional	IMPACTO NA ARQUITETURA:	Médio
Descrição:	<i>O sistema deverá ler e salvar todos os códigos de falhas fornecidos pela central eletrônica em sua memória contendo timestamp e o código de identificação da falha. O usuário deverá ter acesso a estes gerados pela central eletrônica através da interface gráfica em forma de tabela.</i>		

REQ036 – Descrição de códigos de falha			
PRIORIDADE:	Alta	ESTABILIDADE	Alta
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Funcional	IMPACTO NA ARQUITETURA:	Médio
Descrição:	<i>Ao selecionar um código de falha a partir da tabela gerada na interface usuário, o software deverá disponibilizar ao usuário uma breve descrição da falha. Estes códigos de falha são catalogados nas documentações dos diferentes fabricantes de veículos e serão mapeados no sistema para que o usuário não precise acessar toda a documentação do fabricante.</i>		

REQ037 – Pesquisa de códigos de falhas			
PRIORIDADE:	Média	ESTABILIDADE	Alta
SOLICITANTE:	Equipe de Software	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Funcional	IMPACTO NA ARQUITETURA:	Baixo
Descrição:	<i>O software deverá ter um campo de pesquisa que permitirá ao usuário buscar falhas a partir de seus códigos previamente gravados em um banco de dados. Estes códigos de falha são catalogados nas documentações dos diferentes fabricantes de veículos e serão mapeados no sistema.</i>		

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

REQ038 – Controle de fluxo em pesquisa de códigos de falha			
PRIORIDADE:	Média	ESTABILIDADE	Alta
SOLICITANTE:	Equipe de Software	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Funcional	IMPACTO NA ARQUITETURA:	Baixo
Descrição:	<i>Este campo deverá conter o controle de fluxo necessário para não interromper a execução das Queries do banco de dados e na ocorrência do valor inserido ser inválido, retornar uma mensagem ao usuário detalhando o erro.</i>		

REQ039 – Verificar estado da memória			
PRIORIDADE:	Média	ESTABILIDADE	Alta
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Funcional	IMPACTO NA ARQUITETURA:	Alto
Descrição:	<i>Todas as vezes em que o limite da memória de armazenamento externo é alcançado, a interface envia uma caixa de mensagem ao usuário com dois botões (Sim e Não) perguntando se ele deseja sobreescriver os dados da memória com buffer circular. Se o usuário optar por não sobreescriver o armazenamento de dados da memória deverá ser bloqueado.</i>		

REQ040 – Mostrar espaço restante da memória			
PRIORIDADE:	Média	ESTABILIDADE	Alta
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Funcional	IMPACTO NA ARQUITETURA:	Alto
Descrição:	<i>Informa ao usuário o espaço livre e ocupado da memória externa e representa graficamente uma barra nas cores: verde (tamanho de espaço disponível para gravação) e vermelho (tamanho do espaço ocupado).</i>		

REQ041 – Tentativas de comunicação			
PRIORIDADE:	Alta	ESTABILIDADE	Alta
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Não Funcional	IMPACTO NA ARQUITETURA:	Alto
Descrição:	<i>O SADVE executará uma consulta ao dispositivo a ser analisado, se a comunicação falhar mais 5 tentativas serão realizadas em intervalos de 20ms, se todas as 5 tentativas falharem o usuário é informado do problema de comunicação com esse dispositivo específico.</i>		

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

REQ042 – Integridade da mensagem			
PRIORIDADE:	Alta	ESTABILIDADE	Alta
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Não Funcional	IMPACTO NA ARQUITETURA:	Alto
Descrição:	<i>O SADVE verificará a integridade da mensagem recebida a partir do dispositivo consultado, se constatado que a mensagem está corrompida ou falha na verificação do CRC uma nova requisição é enviada pelo SADVE ao dispositivo consultado sendo, então, realizadas 5 tentativas em intervalos de 20 ms, se todas as tentativas resultarem em falha na integridade da mensagem o usuário é informado do problema de mal funcionamento do dispositivo.</i>		

REQ043 – Dicionário de objetos			
PRIORIDADE:	Alta	ESTABILIDADE	Alta
SOLICITANTE:	Persona	REQ. ORIGEM:	-
TIPO DO REQUISITO:	Não Funcional	IMPACTO NA ARQUITETURA:	Alto
Descrição:	<i>Contém os identificadores e os seus respectivos elementos de dados para cada variável dos dispositivos presentes no barramento de comunicação.</i>		

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

4. UML

4.1 Descrição de Caso de Uso

Nome de referência	<i>UC01 – Gerar gráficos</i>
Requisitos	<i>REQ030</i>
Breve descritivo	<i>Descreve como o usuário seleciona os dados e visualiza os dados desejados.</i>
Atores	<i>Usuário</i>
Pré-condições	<i>Todas as informações solicitadas pelo usuário precisam ser válidas (sem erros e formatação e valores adequados).</i>
Cenário principal	<ol style="list-style-type: none"> 1. Usuário seleciona opção “Gráficos” da interface gráfica 2. Seleciona os dados temporais selecionando os intervalos de datas marcando nos calendários 3. Seleciona as caixas com as informações que deseja no gráfico 4. Salvar gráfico 5. O gráfico é gerado e mostrado em outra janela.
Cenário alternativo	<i>A janela possui as opções de adicionar template, alterar gráfico, alterar dados, renomear eixos, descartar gráfico e mostrar curva de performance esperada</i> <i>Breve descrição o que cada item faz;</i>
Requisitos especiais	<i>Falha na leitura da memória ou na consistência do dado requisitado pelo usuário (Ex:Datas inválidas).</i>

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Nome de referência	<i>UC02 – Salvar Template</i>
Requisitos	<i>REQ031</i>
Breve descritivo	<i>Descreve como o usuário cria um template a partir da geração de gráficos</i>
Atores	<i>Usuário</i>
Pré-condições	<i>Os gráficos gerados pelo usuário precisam ser válidos conforme descrito no UC01</i>
Cenário principal	<ol style="list-style-type: none"> 1. Seleciona o botão Adicionar gráfico ao template 2. Uma nova janela abre com o template selecionado (pode ser um previamente selecionado ou novo) 3. O usuário arrasta o gráfico conforme desejar pela janela do template 4. O usuário clica em salvar template 5. O template é salvo e uma mensagem de confirmação é mostrada ao usuário
Cenário alternativo	<p>O usuário pode excluir um gráfico do seu template atual</p> <p>O usuário pode clicar em exportar template para pdf</p> <p>O usuário pode acessar o Relatório de desempenhos gerais (Template Padrão)</p>
Requisitos especiais	<p>Sobrecarga de gráficos inseridos pelo usuário podem tornar o programa lento</p> <p>Gráficos com muitos dados podem tornar a execução lenta</p>

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Nome de referência	<i>UC03 – Visualizar relatório de desempenhos gerais</i>
Requisitos	<i>REQ033</i>
Breve descritivo	<i>Descreve como o usuário configura e utiliza o relatório de desempenhos gerais</i>
Atores	<i>Usuário</i>
Pré-condições	<i>Os gráficos gerados pelo usuário precisam ser válidos</i>
Cenário principal	<ol style="list-style-type: none"> 1. Usuário seleciona uma aba na tela principal chamada “Templates” 2. Seleciona na lista de templates salvos a opção Relatório de desempenhos gerais 3. A tela abre mostrando os gráficos de x,y,z conforme mostra a figura x.
Cenário alternativo	<i>Ao salvar um gráfico o usuário poderá salvá-lo no relatório de desempenhos gerais. Desta maneira, o Template(“Relatório de desempenhos gerais”) é aberto e o usuário pode editá-lo conforme sua preferência. Após a edição, o usuário poderá somente salvá-lo com nome diferente após alterado, criando um Template baseado no relatório de desempenhos gerais.</i>
Requisitos especiais	<i>Sobrecarga de gráficos inseridos pelo usuário podem tornar o programa lento Gráficos com muitos dados podem tornar a execução lenta</i>

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Nome de referência	<i>UC04 – Mostrar alarme de falha de comunicação</i>
Requisitos	<i>REQ011 – Alarme de falha de comunicação</i>
Breve descritivo	<i>Indica ao usuário que houve uma comunicação malsucedida entre o SW e o dispositivo de leitura da ECU.</i>
Atores	-
Pré-condições	<i>Dispositivo de leitura está desenergizado ou fora de alcance.</i>
Cenário principal	<p>O SW envia um comando ao dispositivo de leitura requisitando alguma informação.</p> <ul style="list-style-type: none"> ▪ O SW não recebe resposta ao comando enviado. Reenvia o comando. ▪ Após 3 tentativas de comunicar com o dispositivo, o SW mostra na sua interface uma janela de aviso, com o texto “Erro de comunicação com o dispositivo de leitura. Verifique se o dispositivo está conectado corretamente e tente novamente”, junto com um ícone de aviso.
Cenário alternativo	-
Requisitos especiais	-

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Nome de referência	<i>UC05 – Ler códigos de falha da ECU</i>
Requisitos	<i>REQ035 – Leitura de códigos de falha fornecidos pela central eletrônica</i>
Breve descriptivo	<i>O SW requisita à ECU, por meio do dispositivo de leitura, todos os logs de códigos de falha que ela tem salvado em sua memória, e os salva em um banco de dados.</i>
Atores	<i>Usuário.</i>
Pré-condições	<i>Dispositivo de leitura deve estar conectado ao barramento da ECU.</i>
Cenário principal	<ul style="list-style-type: none"> ▪ O usuário clica no botão “Ler todos códigos de falha”; ▪ O Sistema envia um comando à ECU, requisitando todos os logs de código de falhas que ela possui em sua memória; ▪ A ECU retornará os logs que possui; ▪ O sistema armazenará os logs em seu banco de dados. ▪ O sistema exibirá na interface a mensagem “n logs foram encontrados”, onde “n” é o número de logs.
Cenário alternativo	<i>Caso não tenha nenhuma log na central, o sistema não irá fazer nada.</i>
Requisitos especiais	-

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Nome de referência	<i>UC06– Listar logs de códigos de falha</i>
Requisitos	<i>REQ036 – Descrição de códigos de falha</i>
Breve descritivo	<i>Ao clicar no botão “Listar códigos de falha”, o usuário terá disponível na interface gráfica uma tabela com os logs dos códigos de falha lidos previamente. A tabela possuirá os códigos das falhas na primeira coluna, o timestamp da falha na segunda coluna e uma breve descrição da falha na terceira coluna.</i>
Atores	<i>Usuário.</i>
Pré-condições	<i>-</i>
Cenário principal	<ul style="list-style-type: none"> ▪ <i>O usuário clica no botão “Listar códigos de falha”;</i> ▪ <i>O sistema busca em seu banco de dados por logs de falhas e as exibirá na interface no formato de uma tabela, contendo o código da falha, o timestamp e uma descrição padrão da falha;</i>
Cenário alternativo	<i>Caso não tenha nenhum log salvo, o sistema exibirá a mensagem “Nenhum log encontrado.”</i>
Requisitos especiais	<i>-</i>

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Nome de referência	<i>UC07 – Pesquisar códigos de falhas</i>
Requisitos	<i>REQ037 – Pesquisa de códigos de falhas</i>
Breve descritivo	<i>O usuário pode pesquisar os códigos de falhas conhecidos pelo sistema por meio de um campo de texto, onde é possível digitar o código de falha desejado. O sistema exibirá na interface o título, a descrição da falha e demais informações disponíveis (ex: fabricantes que utilizam esse código de falha).</i>
Atores	<i>Usuário.</i>
Pré-condições	<i>-</i>
Cenário principal	<ul style="list-style-type: none"> ▪ <i>O usuário digita um código de falha no campo de texto “Pesquisar códigos de falha”;</i> ▪ <i>O sistema busca em seu banco de dados códigos de falha que tenham relação com o código digitado;</i> ▪ <i>Caso encontre algum código que corresponda ao digitado, exibe na tela o código, uma descrição da falha e demais informações que estejam disponíveis.</i>
Cenário alternativo	
Requisitos especiais	<i>-</i>

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Nome de referência	<i>UC08 – Estimar o tempo de duração da bateria</i>
Requisitos	<i>REQ024, REQ25, REQ026</i>
Breve descriptivo	<i>Informar ao usuário o nível de carga da bateria e estimar o tempo restante.</i>
Atores	<i>Sistema de gerenciamento da bateria (BMS) e usuário.</i>
Pré-condições	<i>Acesso ao barramento principal de comunicação do veículo.</i>
Cenário principal	<ol style="list-style-type: none"> 1. O usuário seleciona o botão de dados instantâneos 2. O requerimento é enviado ao SADVE 3. O SADVE consultará o seu dicionário de objetos 4. O identificador necessário a ser usado é enviado ao barramento 5. O sistema de gerenciamento da bateria recebe o requerimento e retorna com a resposta 6. O usuário recebe a resposta do requerimento, em formato de hora, através do software.
Cenário alternativo	<p><i>Caso não haja uma resposta do BMS após 5 tentativas, retornar uma mensagem de erro ao usuário informando que não foi possível realizar a consulta por falha de comunicação ao BMS.</i></p> <p><i>No momento do recebimento, caso constatado que a informação está corrompida, mais 5 tentativas são feitas para garantir a integridade da informação, se as 5 tentativas retornarem informações corrompidas, uma mensagem de erro é enviada ao usuário indicando mal funcionamento do dispositivo.</i></p>
Requisitos especiais	<i>REQ41 e REQ42</i>

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Nome de referência	<i>UC09 – Verificar estado de carga da bateria</i>
Requisitos	<i>REQ025, REQ026</i>
Breve descritivo	<i>Mostra como o histórico de uso da carga da bateria e informar, também, o tempo médio para finalização da carga.</i>
Atores	<i>Sistema de gerenciamento da bateria (BMS) e usuário</i>
Pré-condições	<i>Acesso ao barramento principal de comunicação do veículo.</i>
Cenário principal	<ol style="list-style-type: none"> 1. O usuário seleciona o botão de dados instantâneos 2. O requerimento é enviado ao SADVE 3. O SADVE consultará o seu dicionário de objetos 4. O identificador necessário a ser usado é enviado ao barramento 5. O sistema de gerenciamento da bateria recebe o requerimento e retorna com a resposta 6. Após o recebimento da resposta, o SADVE averiguará o seu histórico para o uso da carga da bateria 7. A resposta, contendo o histórico e o dado recebido pelo sistema de gerenciamento de bateria, é retornado ao usuário 8. O estado de carga da bateria atual, como também a sua utilização em momentos anteriores, pode ser mostrada ao usuário como um dado pontual ou em formato de gráfico cartesiano com Ah em função da quilometragem ou Ah em função do tempo, para o caso de carga da bateria.
Cenário alternativo	<p><i>Caso não haja uma resposta do BMS após 5 tentativas, retornar uma mensagem de erro ao usuário informando que não foi possível realizar a consulta por falha de comunicação ao BMS.</i></p> <p><i>No momento do recebimento, caso constatado que a informação está corrompida, mais 5 tentativas são feitas para garantir a integridade da informação, se as 5 tentativas retornarem informações corrompidas, uma mensagem de erro é enviada ao usuário indicando mal funcionamento do dispositivo.</i></p>
Requisitos especiais	<i>REQ41 e REQ42</i>

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Nome de referência	<i>UC10 – Acessar sistema de gerenciamento da bateria</i>
Requisitos	<i>REQ026</i>
Breve descriptivo	<i>Mostra como o usuário acessa de somente leitura ao histórico e ao sistema de gerenciamento da bateria</i>
Atores	<i>Sistema de gerenciamento da bateria (BMS), usuário.</i>
Pré-condições	<i>Acesso ao barramento principal de comunicação do veículo.</i>
Cenário principal	<ol style="list-style-type: none"> 1. O usuário seleciona o botão dados instantâneos 2. O requerimento é enviado ao SADVE 3. O SADVE consultará o seu dicionário de objetos 4. O identificador necessário a ser usado é enviado ao barramento 5. O sistema de gerenciamento da bateria recebe o requerimento e retorna com a resposta 6. Após o recebimento da resposta, o SADVE averiguará o seu histórico para de diagnóstico do sistema de gerenciamento da bateria 7. A resposta, contendo o histórico e o dado recebido pelo sistema de gerenciamento de bateria, é retornado ao usuário 8. O diagnóstico do sistema de gerenciamento da bateria verificará a integridade do mesmo, como também a sua utilização em momentos anteriores, pode ser mostrada ao usuário como um dado pontual ou em formato de gráfico cartesiano com corrente, tensão e isolamento do veículo, em função do tempo.
Cenário alternativo	<p><i>Caso não haja uma resposta do BMS após 5 tentativas, retornar uma mensagem de erro ao usuário informando que não foi possível realizar a consulta por falha de comunicação ao BMS.</i></p> <p><i>No momento do recebimento, caso constatado que a informação está corrompida, mais 5 tentativas são feitas para garantir a integridade da informação, se as 5 tentativas retornarem informações corrompidas, uma mensagem de erro é enviada ao usuário indicando mal funcionamento do dispositivo.</i></p>
Requisitos especiais	<i>REQ41 e REQ42</i>

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Nome de referência	<i>UC11 – Verificar estado do relé</i>
Requisitos	<i>REQ026, REQ027, REQ021</i>
Breve descritivo	<i>Mostra como o usuário acessa de somente leitura ao histórico dos relés do sistema.</i>
Atores	<i>Sistema de gerenciamento da bateria (BMS), Inversor(es) e usuário.</i>
Pré-condições	<i>Acesso ao barramento principal de comunicação do veículo.</i>
Cenário principal	<ol style="list-style-type: none"> 1. O usuário seleciona o botão de dados instantâneos 2. O requerimento é enviado ao SADVE 3. O SADVE consultará o seu dicionário de objetos 4. Os identificadores necessários a serem usados é enviado ao barramento 5. O sistema de gerenciamento da bateria e o(s) inversor(es) recebem os requerimentos e retorna com a resposta 6. Após o recebimento das respostas, o SADVE averiguará o seu histórico do estado do relé para o sistema de gerenciamento da bateria e do(s) inversor(es) 7. As respostas, contendo os históricos e os dados recebidos pelo sistema de gerenciamento de bateria e o(s) inversor(es), são retornados ao usuário 8. O estado do relé pode ser mostrado ao usuário como um dado pontual ou em formato de gráfico em função do tempo.
Cenário alternativo	<p><i>Caso não haja uma resposta do BMS após 5 tentativas, retornar uma mensagem de erro ao usuário informando que não foi possível realizar a consulta por falha de comunicação ao BMS.</i></p> <p><i>No momento do recebimento, caso constatado que a informação está corrompida, mais 5 tentativas são feitas para garantir a integridade da informação, se as 5 tentativas retornarem informações corrompidas, uma mensagem de erro é enviada ao usuário indicando mal funcionamento do dispositivo.</i></p>
Requisitos especiais	<i>REQ41 e REQ42</i>

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Nome de referência	<i>UC12 – Verificar limites de temperatura</i>
Requisitos	<i>REQ021, REQ026, REQ028</i>
Breve descritivo	<i>Mostra como o usuário acessa de somente leitura ao histórico de temperaturas de todos os sistemas críticos do veículo.</i>
Atores	<i>Sistema de gerenciamento da bateria (BMS), Inversor(es), sensores e usuário.</i>
Pré-condições	<i>Acesso ao barramento principal de comunicação do veículo.</i>
Cenário principal	<ol style="list-style-type: none"> 1. O usuário seleciona o botão de dados instantâneos 2. O requerimento é enviado ao SADVE 3. O SADVE consultará o seu dicionário de objetos 4. Os identificadores necessários a serem usados é enviado ao barramento 5. O sistema de gerenciamento da bateria, o(s) inversor(es) e os sensores recebem os requerimentos e retornam com as respostas 6. Após o recebimento das respostas, o SADVE averiguará o seu histórico de temperaturas para o sistema de gerenciamento da bateria, do(s) inversor(es) e dos sensores 7. As respostas, contendo os históricos e os dados recebidos pelo sistema de gerenciamento de bateria, do(s) inversor(es) e dos sensores, são retornados ao usuário 8. Os limites de temperatura podem ser mostrados ao usuário como um dado pontual ou em formato de gráfico em função do tempo.
Cenário alternativo	<p><i>Caso não haja uma resposta do BMS após 5 tentativas, retornar uma mensagem de erro ao usuário informando que não foi possível realizar a consulta por falha de comunicação ao BMS.</i></p> <p><i>No momento do recebimento, caso constatado que a informação está corrompida, mais 5 tentativas são feitas para garantir a integridade da informação, se as 5 tentativas retornarem informações corrompidas, uma mensagem de erro é enviada ao usuário indicando mal funcionamento do dispositivo.</i></p>
Requisitos especiais	<i>REQ41 e REQ42</i>

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Nome de referência	<i>UC13 – Consultar histórico de velocidades</i>
Requisitos	<i>REQ029</i>
Breve desritivo	<i>O usuário poderá ter acesso de somente leitura ao histórico de velocidades do veículo.</i>
Atores	<i>Sistema de gerenciamento da bateria (BMS), Inversor(res) e usuário.</i>
Pré-condições	<i>Acesso ao barramento principal de comunicação do veículo.</i>
Cenário principal	<ol style="list-style-type: none"> 1. O usuário seleciona o botão de histórico 2. O requerimento é enviado ao SADVE 3. O SADVE consultará o seu registro de mesmo nome 4. A resposta, contendo o histórico de velocidades, é retornado ao usuário 5. As velocidades podem ser mostradas ao usuário como um dado pontual ou em formato de gráfico em função do tempo.
Cenário alternativo	<p><i>Caso não haja uma resposta do BMS após 5 tentativas, retornar uma mensagem de erro ao usuário informando que não foi possível realizar a consulta por falha de comunicação ao BMS.</i></p> <p><i>No momento do recebimento, caso constatado que a informação está corrompida, mais 5 tentativas são feitas para garantir a integridade da informação, se as 5 tentativas retornarem informações corrompidas, uma mensagem de erro é enviada ao usuário indicando mal funcionamento do dispositivo.</i></p>
Requisitos especiais	<i>REQ41 e REQ42</i>

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Nome de Referência	<i>UC14 – Controlar timestamps de dados</i>
Requisitos	<i>REQ010</i>
Breve descritivo	<i>Este caso de uso possui os passos necessários para a realização do armazenamento e controle de timestamps de dados.</i>
Pré-condições	<i>É necessário que o módulo esteja conectado e funcionando e que exista uma memória externa com conexão funcional.</i>
Atores Envolvidos	<i>Módulo de controle de dados (Monitor CAN) Memória externa</i>
Cenário Principal	<ol style="list-style-type: none"> 1. Módulo envia mensagens para a memória externa para salvar o timestamp e o dado (Velocidade, rotação, etc); 2. Memória responde um OK.
Cenário Alternativo	<ol style="list-style-type: none"> 1. Memória não responde OK. 2. Módulo reenvia a mensagem para salvar o dado.
Requisitos Especiais	<i>Salvamento extremo de dados pode ocasionar na falta de espaço para salvar mais informações.</i>

Nome de Referência	<i>UC15 – Acionar LED indicador de estado</i>
Requisitos	<i>REQ017</i>
Breve descritivo	<i>Este caso de uso possui os passos necessários para o controle dos LEDs indicadores de estado.</i>
Pré-condições	<i>É necessário que o módulo esteja conectado e funcionando e que exista algum dado salvo.</i>
Atores Envolvidos	<i>Módulo de controle de dados (Monitor CAN) Usuário</i>
Cenário Principal	<ol style="list-style-type: none"> 1. Usuário realiza a conexão com o módulo; 2. Módulo acende LED verde e responde OK; 3. Usuário busca algum pacote de dados salvo (E.g.: seleciona uma data para o eixo X e outro dado para o eixo Y e clica no botão gerar gráfico); 4. Módulo pisca o LED azul e envia os pacotes de dados.
Cenário Alternativo	<ol style="list-style-type: none"> 1. Módulo não acende o LED e não responde; 2. Usuário tenta a reconexão e inicia o uso a partir do início do cenário principal. 3. Módulo não responde com nenhum pacote; 4. Usuário não verifica o LED aceso e pede novamente o dado.
Requisitos Especiais	

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Nome de Referência	<i>UC16 – Visualizar rotação do motor</i>
Requisitos	<i>REQ018</i>
Breve descriptivo	<i>Este caso de uso possui os passos necessários para a seleção da interface gráfica relacionada a velocidade de rotação do motor.</i>
Pré-condições	<i>É necessário que o módulo esteja conectado e funcionando, que o usuário esteja conectado ao módulo e que exista algum dado relacionado a rotação do motor salvo.</i>
Atores Envolvidos	<i>Módulo de controle de dados (Monitor CAN) Memória Usuário</i>
Cenário Principal	<ol style="list-style-type: none"> 1. Usuário clica no botão gráficos; 2. Usuário seleciona a data em que ele deseja buscar a informação; 3. Usuário seleciona 'Rotação do motor' para o eixo Y 4. Usuário clica no botão 'Gerar gráfico'; 5. Módulo busca o dado no sistema de armazenamento; 6. Sistema de armazenamento retorna o dado; 7. Módulo monta a tabela retorna o dado para o usuário.
Cenário Alternativo	<ol style="list-style-type: none"> 1. Sistema de armazenamento não retorna nenhum dado; 2. Módulo realiza a busca novamente; 3. Sistema de armazenamento ainda não retorna nenhum dado; 4. Módulo responde ao usuário que não existem dados relacionados a rotação do motor.
Requisitos Especiais	<i>Muitas requisições podem sobrecarregar o sistema.</i>

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Nome de Referência	<i>UC17 – Verificar estado da memória</i>
Requisitos	<i>REQ039</i>
Breve descritivo	<i>Este caso de uso possui os passos necessários para o controle de armazenamento de dados quando a memória já está cheia.</i>
Pré-condições	<i>É necessário que o módulo esteja conectado e funcionando, que o usuário esteja conectado ao módulo e que a memória esteja sem espaço para novos dados.</i>
Atores Envolvidos	<i>Módulo de controle de dados (Monitor CAN) Usuário</i>
Cenário Principal	<ol style="list-style-type: none"> 1. Módulo envia informação e apresenta na tela um pop-up para o usuário dizendo que memória está cheia; 2. Usuário clica em 'sim', dizendo que deseja sobrescrever os dados existentes; 3. Módulo responde OK;
Cenário Alternativo	<ol style="list-style-type: none"> 1. Usuário clica em 'não', dizendo que não deseja sobrescrever os dados existentes; 2. Módulo realiza o bloqueio de salvamento de dados e responde OK.
Requisitos Especiais	

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Nome de Referência	<i>UC18 – Mostrar espaço restante da memória</i>
Requisitos	<i>REQ040</i>
Breve descritivo	<i>Este caso de uso possui os passos necessários para o controle de armazenamento de dados.</i>
Pré-condições	<i>É necessário que o módulo esteja conectado e funcionando, que o usuário esteja conectado ao módulo e que exista uma memória.</i>
Atores Envolvidos	<i>Módulo de controle de dados (Monitor CAN) Memória Usuário</i>
Cenário Principal	<ol style="list-style-type: none"> 1. Usuário abre a tela inicial do aplicativo; 2. Módulo busca a informação sobre o espaço da memória; 3. Memória responde o espaço livre e utilizado; 4. Módulo gera o gráfico e coloca na tela.
Cenário Alternativo	<ol style="list-style-type: none"> 1. Memória não responde as informações referentes ao espaço livre/utilizado; 2. Módulo realiza a busca novamente; 3. Memória ainda não retorna nenhum dado; 4. Módulo responde ao usuário que a memória está com problemas.
Requisitos Especiais	

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Nome de referência	<i>UC19 – Verificar corrente do Motor</i>
Requisitos	<i>REQ019</i>
Breve descritivo	<i>Como o usuário interage com o sistema afim de retirar os dados pertinentes a visualização da corrente do motor</i>
Atores	<i>Usuário</i>
Pré-condições	<i>O sistema deve ser bem aterrado, sem fuga de corrente, os dados não podem estar corrompidos e serem válidos</i>
Cenário principal	<ol style="list-style-type: none"> 1. Seleciona um template que possui os dados referentes a corrente de motor. 2. Insere o período relativo aos gráficos desejados. 3. Visualiza o gráfico da corrente do motor referentes ao tempo selecionado. 4. Seleciona para ver os timestamps de quando a corrente fugiu do esperado. 5. Opta por salvar, exportar ou descartar os dados lidos.
Cenário alternativo	<p>(1) Quando não há template criado para os dados requisitados, cria template e adiciona a opção referente a corrente do motor.</p> <p>(2) O usuário seleciona ver a corrente instantânea do motor</p> <p>(5) O usuário não realiza nenhuma operação com os dados</p>
Requisitos especiais	<i>Os dados não podem estar corrompidos e serem válidos, o sistema deve estar comunicando devidamente sem erros.</i>

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Nome de referência	<i>UC20 – Verificar tensão das Baterias</i>
Requisitos	<i>REQ020</i>
Breve descritivo	<i>Histórico da tensão das baterias e como elas se comportaram no decorrer do uso do veículo</i>
Atores	<i>Usuário</i>
Pré-condições	<i>O sistema deve ser bem aterrado, sem fuga de corrente, os dados não podem estar corrompidos e serem válidos</i>
Cenário principal	<ol style="list-style-type: none"> 1. <i>Seleciona um template que possui os dados referentes à tensão das baterias.</i> 2. <i>Insere o período relativo ao fornecimento de tensão das baterias ao sistema.</i> 3. <i>Visualiza o gráfico da tensão das baterias conforme o espaço de tempo selecionado</i> 4. <i>Seleciona para ver os timestamps de dados fora do esperado</i> 5. <i>Seleciona se deseja salvar, exportar ou apagar os dados lidos</i>
Cenário alternativo	<p>(1) <i>O usuário seleciona criar template e adiciona a opção referente a tensão das baterias</i></p> <p>(2) <i>O usuário seleciona ver a carga das baterias</i></p> <p>(5) <i>O usuário não realiza nenhuma operação com os dados</i></p>
Requisitos especiais	<i>Os dados não podem estar corrompidos e serem válidos, o sistema deve estar comunicando devidamente sem erros.</i>

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Nome de referência	<i>UC21 – Verificar estado do Inversor</i>
Requisitos	<i>REQ021</i>
Breve descritivo	<i>Demonstra como os inversores atuaram ao decorrer do tempo de funcionamento</i>
Atores	<i>Usuário</i>
Pré-condições	<i>O sistema deve ser bem aterrado, sem fuga de corrente</i>
Cenário principal	<ol style="list-style-type: none"> 1. <i>Seleciona um template que possui os dados referentes aos inversores</i> 2. <i>Insere o período de interesse para a análise</i> 3. <i>Escolhe se deseja visualizar as timestamps de erros que o sistema detectou</i> 4. <i>Escolhe a opção de checar o estado atual dos inversores</i> 5. <i>Seleciona se deseja salvar, apagar ou excluir os dados lidos.</i>
Cenário alternativo	<p>(1) <i>O usuário seleciona criar novo template e adiciona a opção referente ao estado dos inversores.</i></p> <p>(2 – 3) <i>O usuário decide pular para a quarta etapa, ignorando o histórico de estado dos inversores.</i></p> <p>(4) <i>Em caso avarias nos inversores, o usuário selecionará a opção de emissão de relatório de erros com base nos documentos gerados pelo fabricante</i></p> <p>(5) <i>Usuário não realiza nenhuma operação com os dados</i></p>
Requisitos especiais	<i>Os inversores deverão estar adequadamente instalados no sistema, não apresentando falhas que comprometam outros componentes do sistema. Os dados não podem estar corrompidos e serem válidos, o sistema deve estar comunicando devidamente sem erros.</i>

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

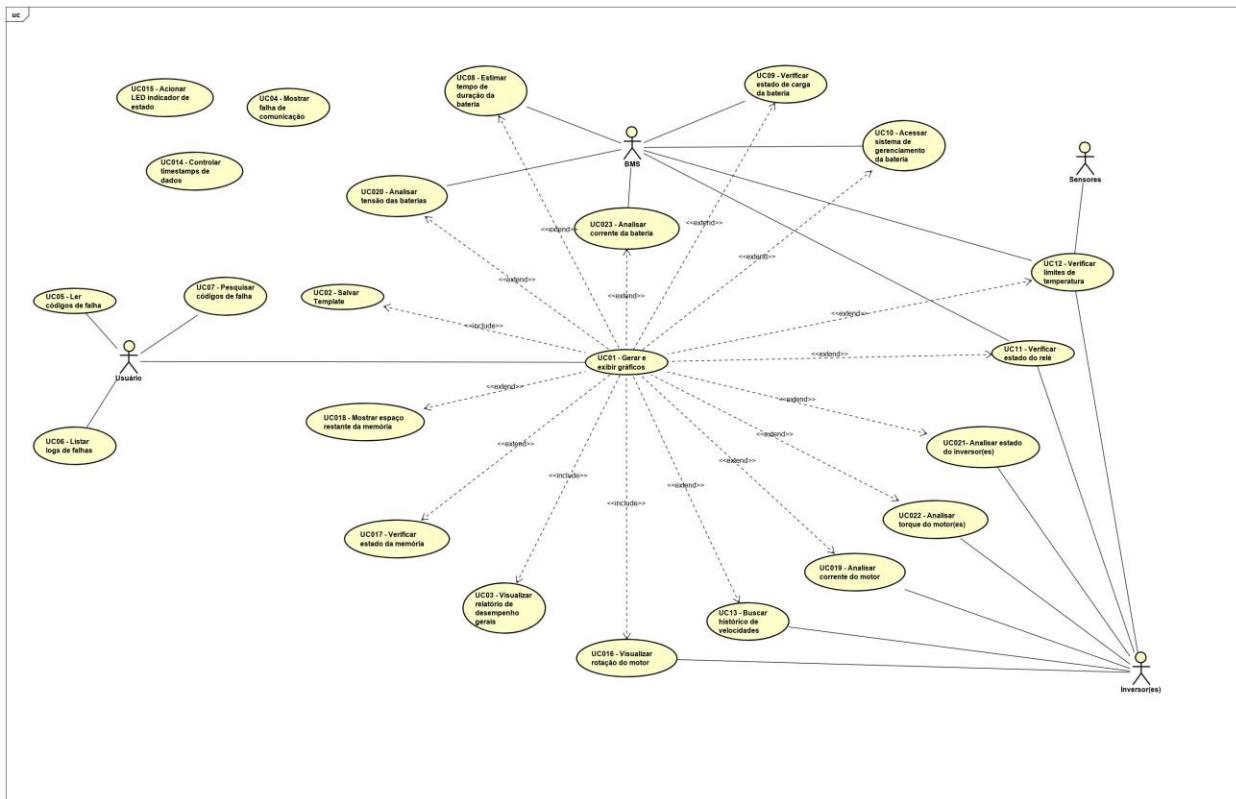
Nome de referência	<i>UC22 – Verificar torque do motor</i>
Requisitos	<i>REQ022</i>
Breve descritivo	<i>Histórico de dados do torque do motor durante o período de funcionamento do veículo.</i>
Atores	<i>Usuário</i>
Pré-condições	<i>O sistema deve ser bem aterrado, sem fuga de corrente, os dados não podem estar corrompidos e serem válidos, o sistema deve estar comunicando devidamente sem erros.</i>
Cenário principal	<ol style="list-style-type: none"> 1. Seleciona um template que possui os dados referentes ao histórico do torque do motor. 2. Seleciona o período de funcionamento do veículo que achar mais pertinente para a análise 3. Seleciona para ver os timestamps de performance fora do esperado. 4. Seleciona se deseja salvar, exportar ou apagar os dados lidos
Cenário alternativo	<p>(1) O usuário seleciona criar template e adiciona a opção referente ao histórico do motor.</p> <p>(4) O usuário não realiza nenhuma operação com os dados</p>
Requisitos especiais	<i>Os dados não podem estar corrompidos e serem válidos, o sistema deve estar comunicando devidamente sem erros.</i>

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Nome de referência	<i>UC23 – Verificar corrente da Bateria</i>
Requisitos	<i>REQ023</i>
Breve descritivo	<i>Histórico das correntes das baterias e como elas se comportaram no decorrer do uso do veículo</i>
Atores	<i>Usuário</i>
Pré-condições	<i>O sistema deve ser bem aterrado, sem fuga de corrente.</i>
Cenário principal	<ol style="list-style-type: none"> 1. O usuário seleciona um template que possui os dados referentes ao histórico da corrente da bateria. 2. O usuário seleciona o período de funcionamento do veículo que achar mais pertinente para a análise 3. O usuário seleciona para ver os timestamps de performance fora do esperado. 4. Usuário seleciona se deseja salvar, exportar ou apagar os dados lidos
Cenário alternativo	<p>(1) O usuário seleciona criar template e adiciona a opção referente ao histórico do motor.</p> <p>(4) O usuário não realiza nenhuma operação com os dados</p>
Requisitos especiais	<i>os dados não podem estar corrompidos e serem válidos, o sistema deve estar comunicando devidamente sem erros.</i>

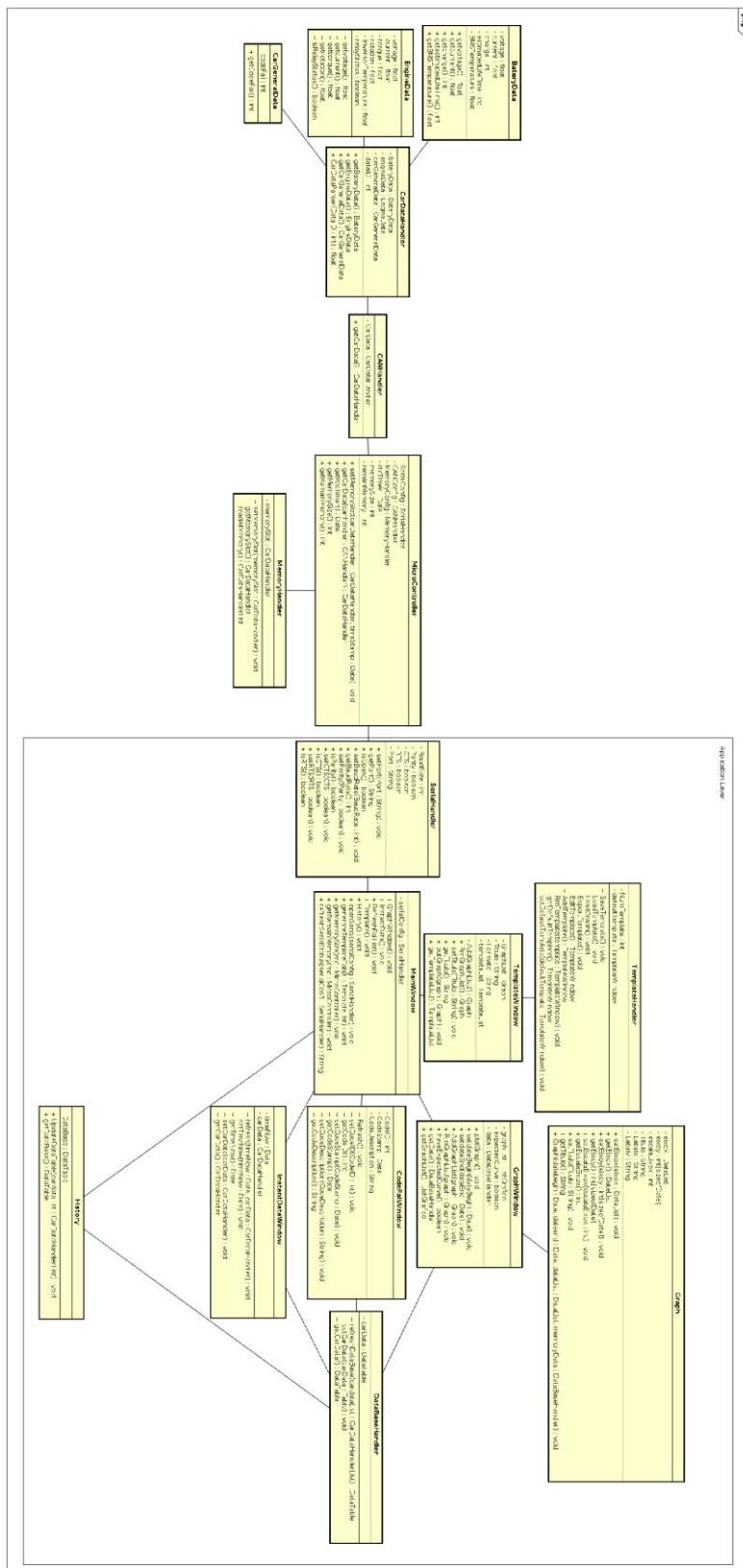
Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

4.2 Diagrama Caso de Uso



Sistema de Análise de Veículos Elétricos | Versão: 1.0
Documento de Arquitetura de Software | Data: 29/05/19

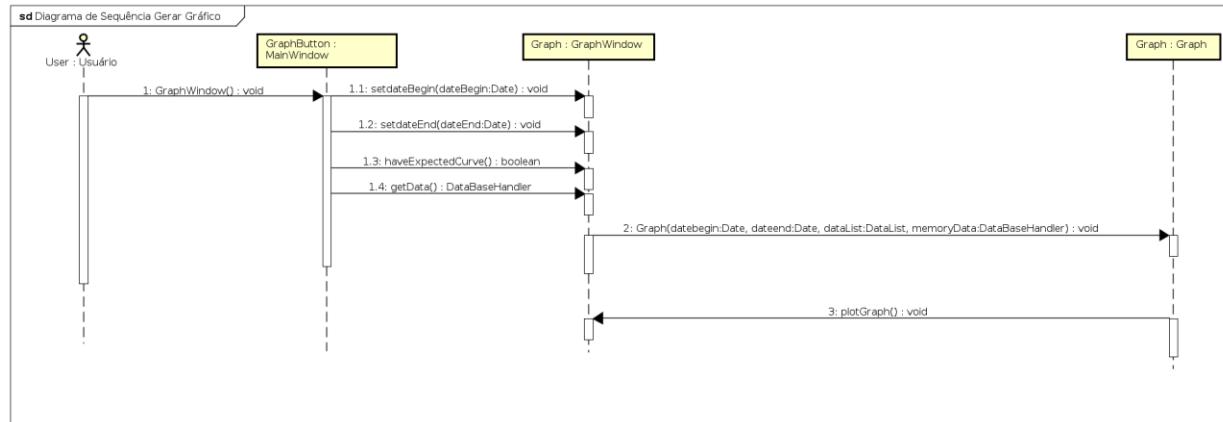
4.3 Diagrama de Classes



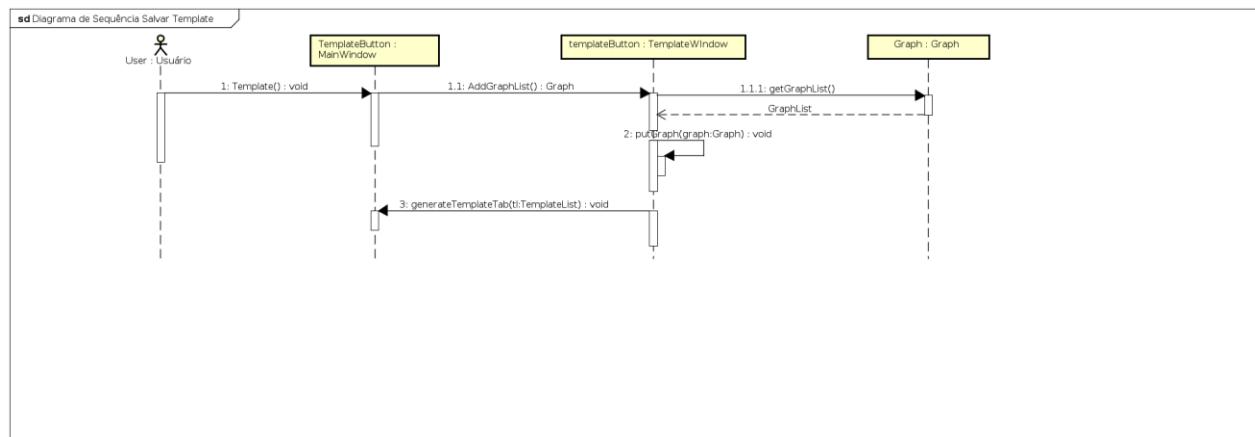
Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

4.4 Diagramas de Sequência

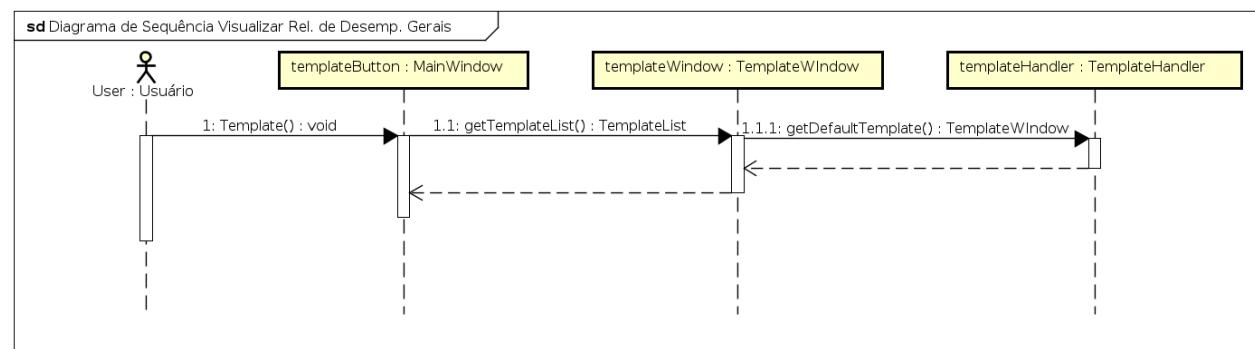
Use Case 1:



Use Case 2:

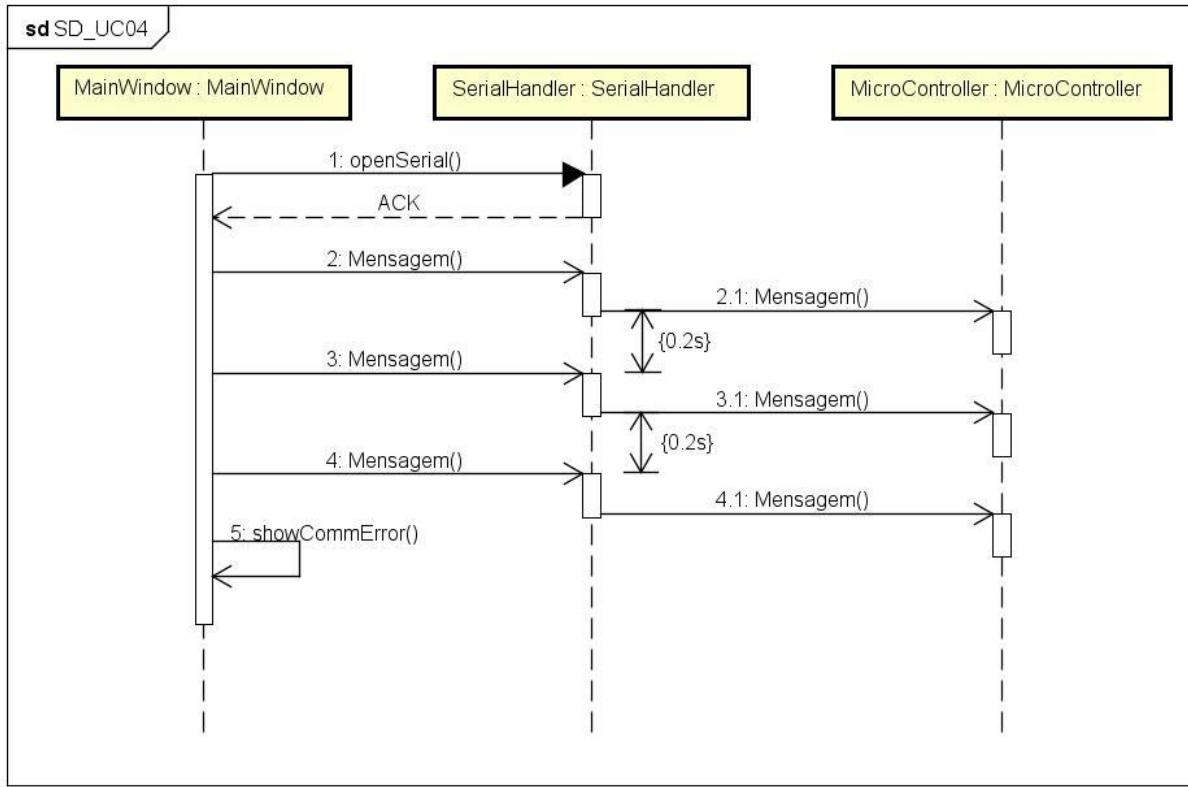


Use Case 3:

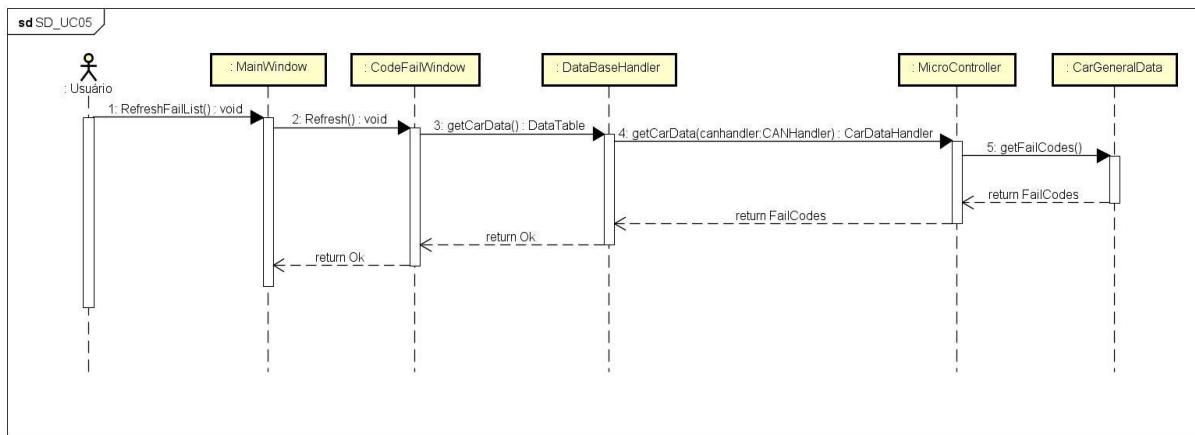


Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Use Case 4:

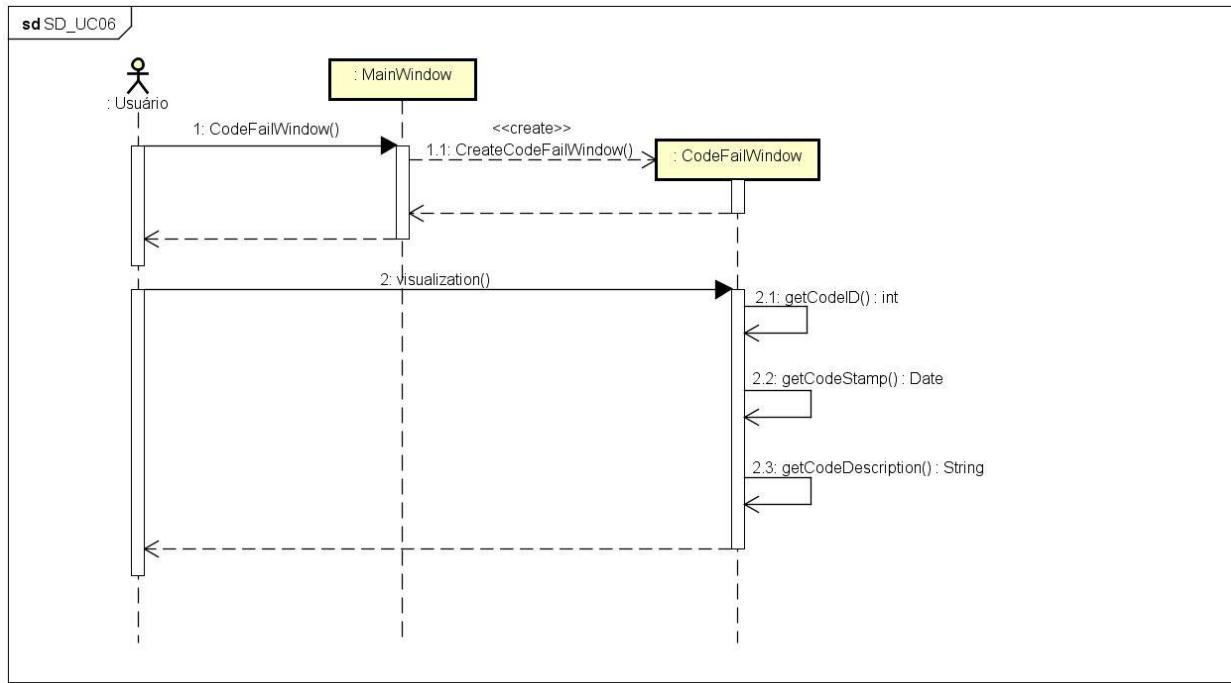


Use Case 5:

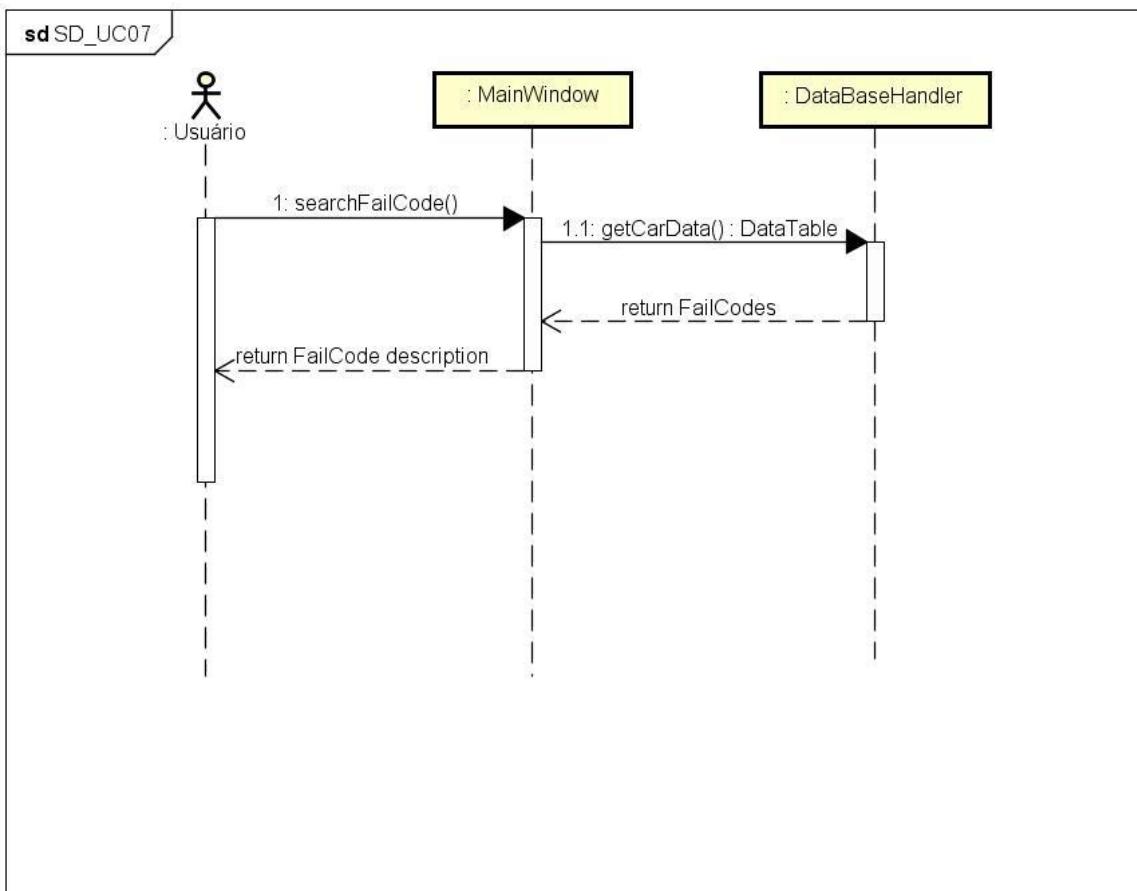


Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

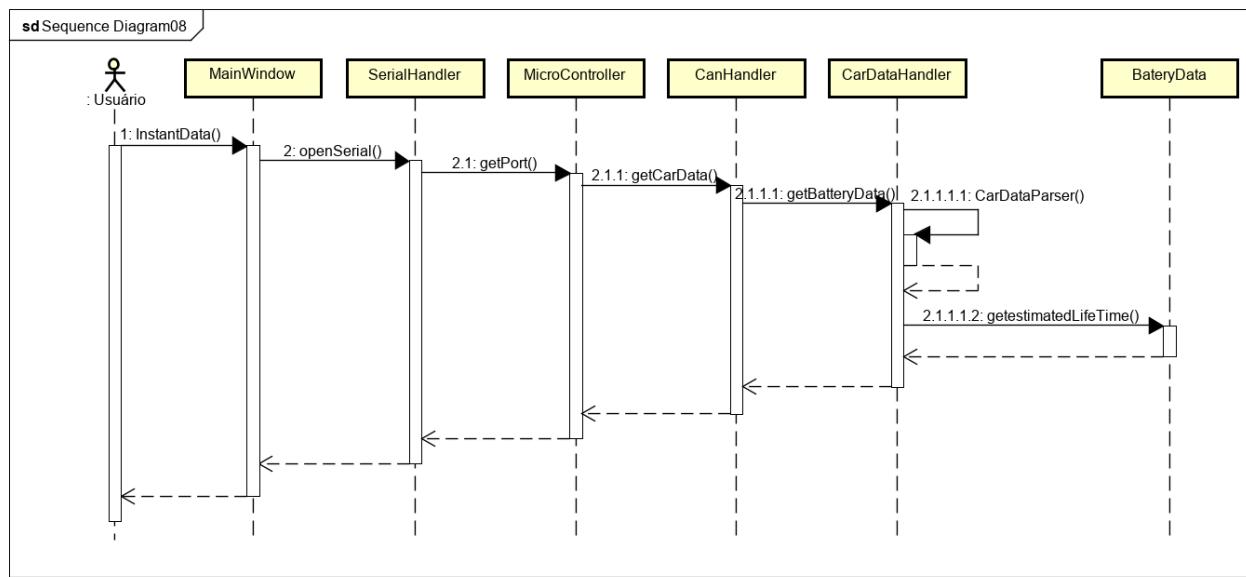
Use Case 6:



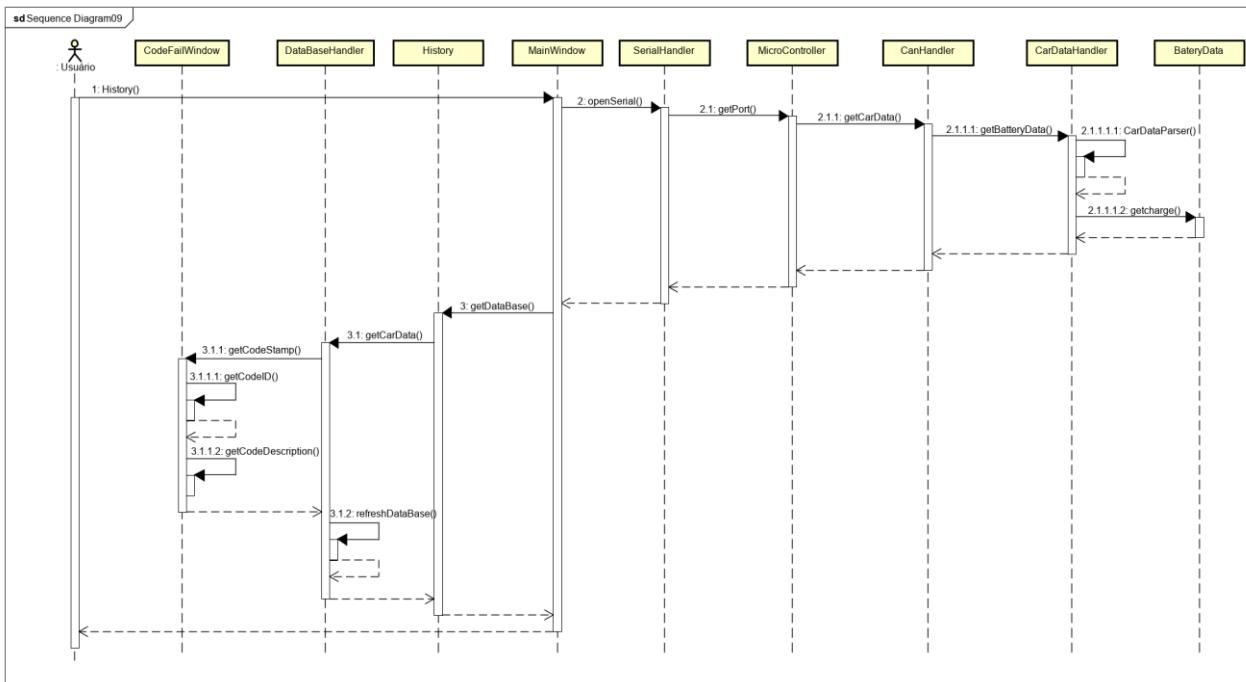
Use Case 7:



Use Case 8:

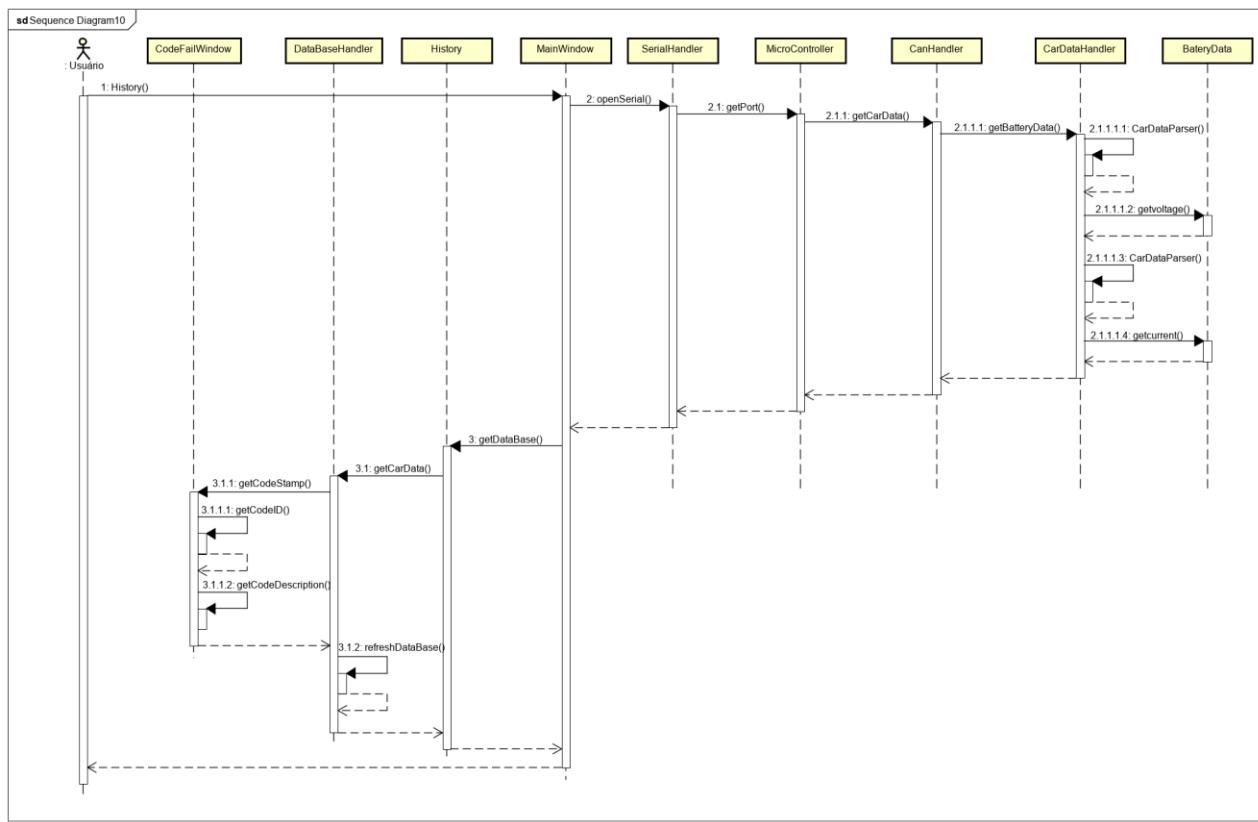


Use Case 9:



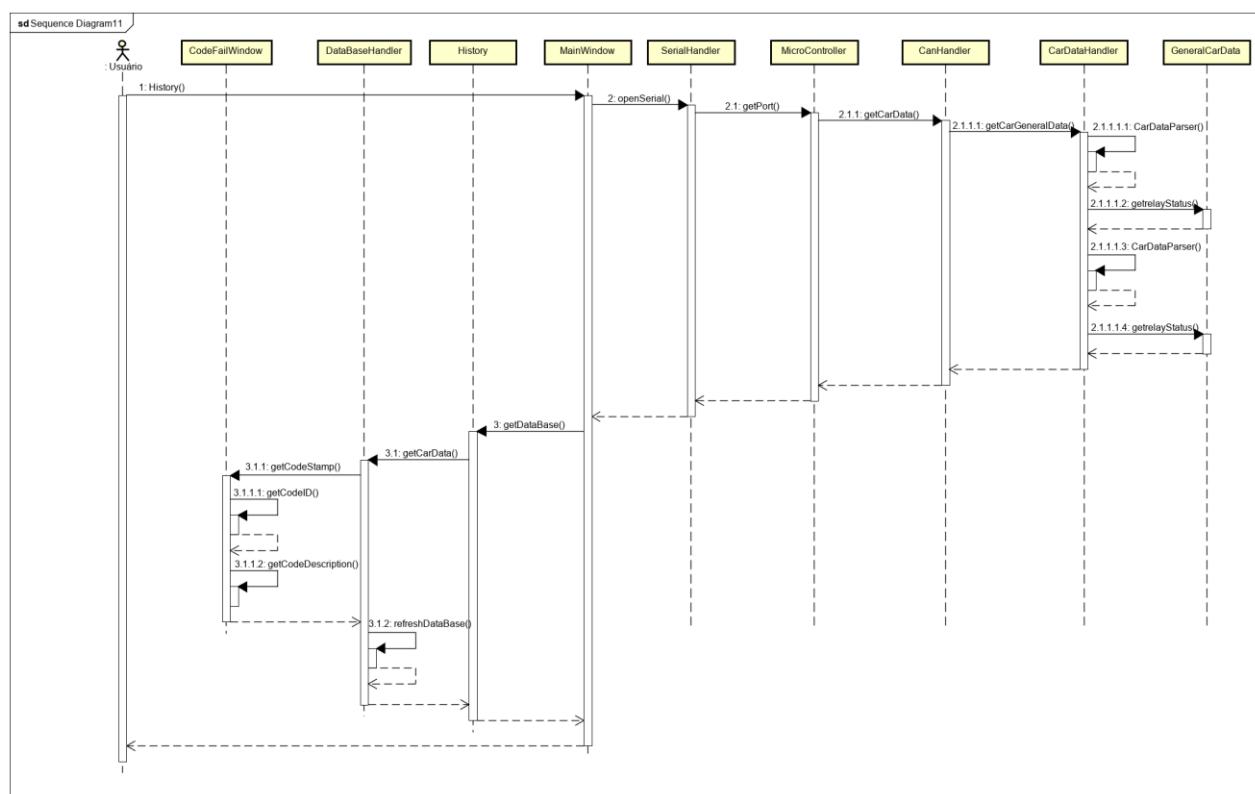
Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Use Case 10:

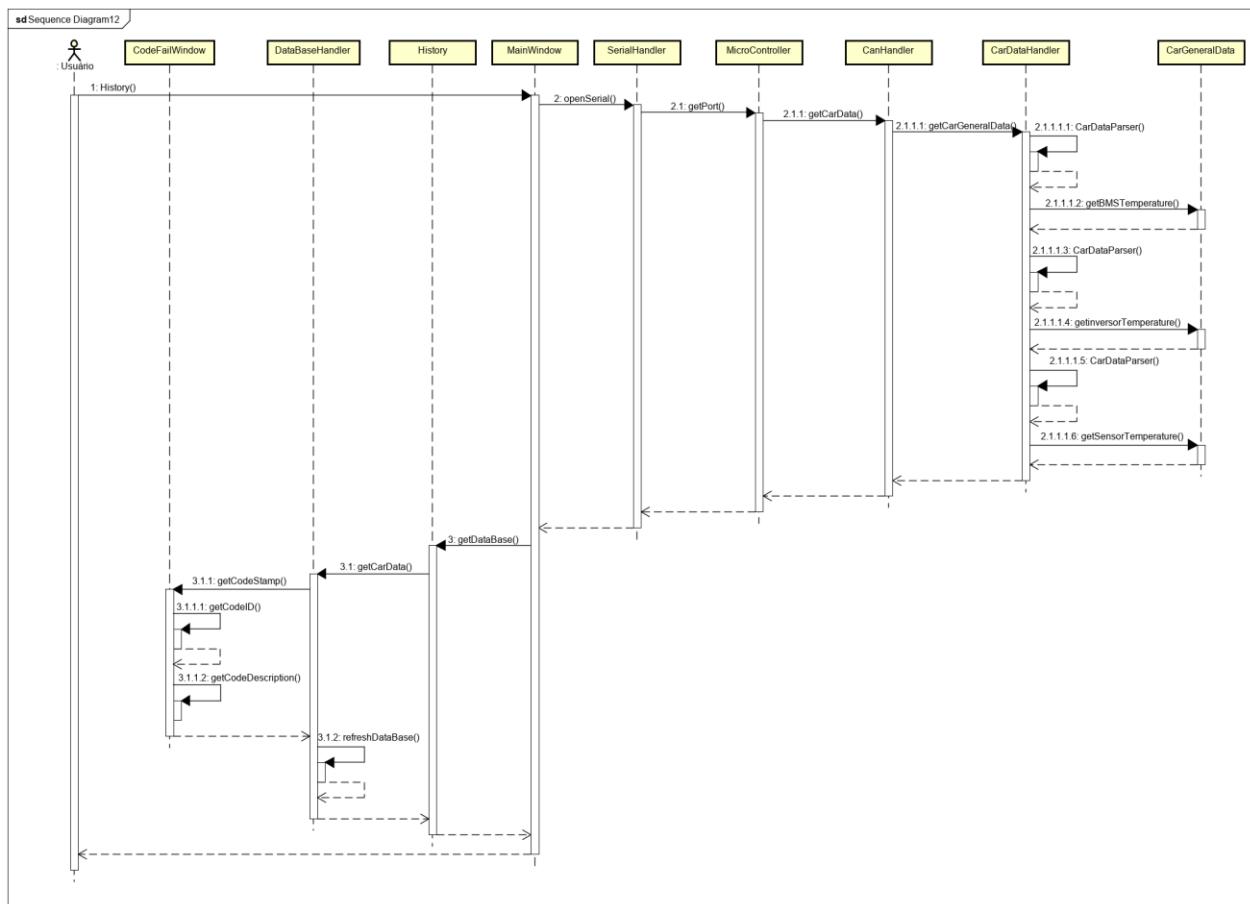


Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Use Case 11:

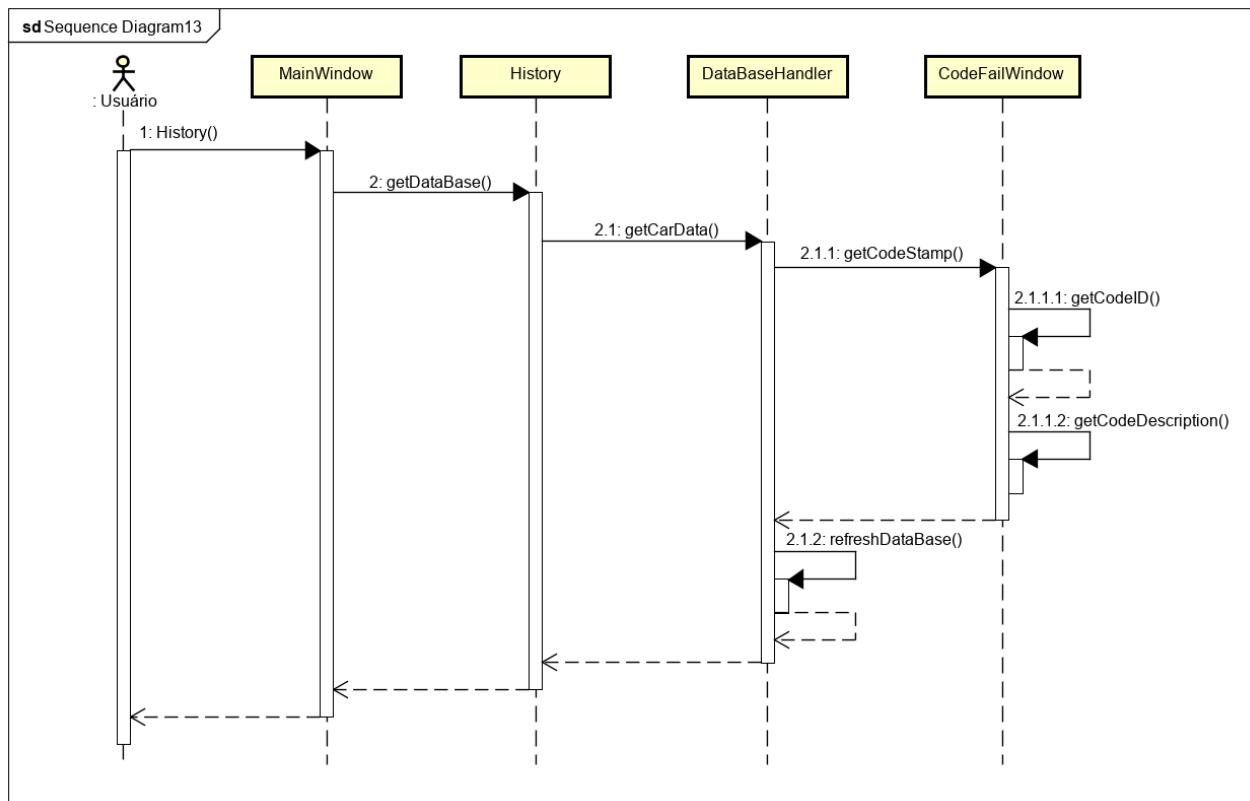


Use Case 12:

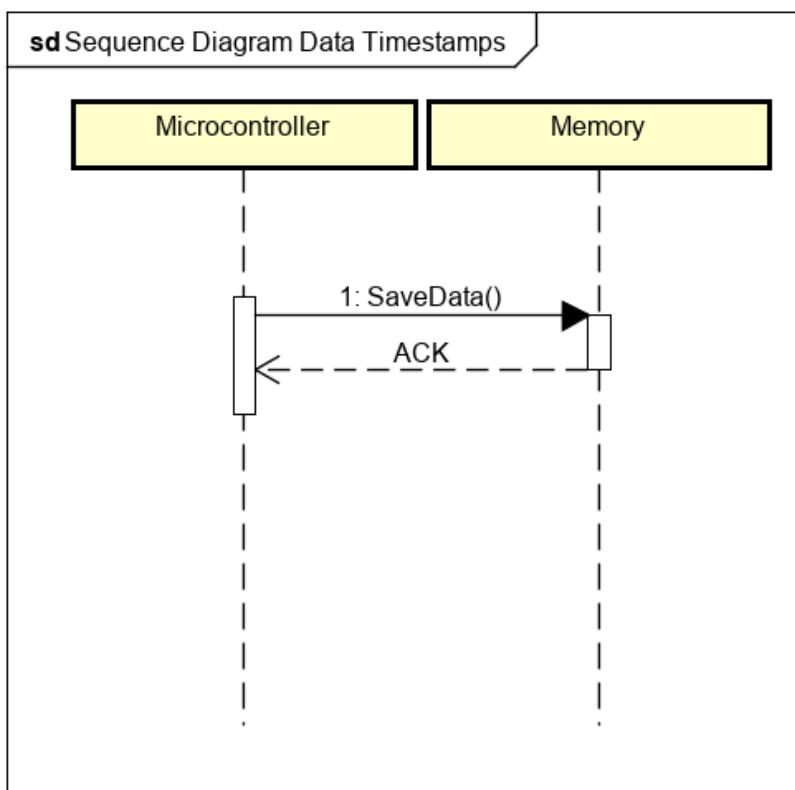


Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Use Case 13:

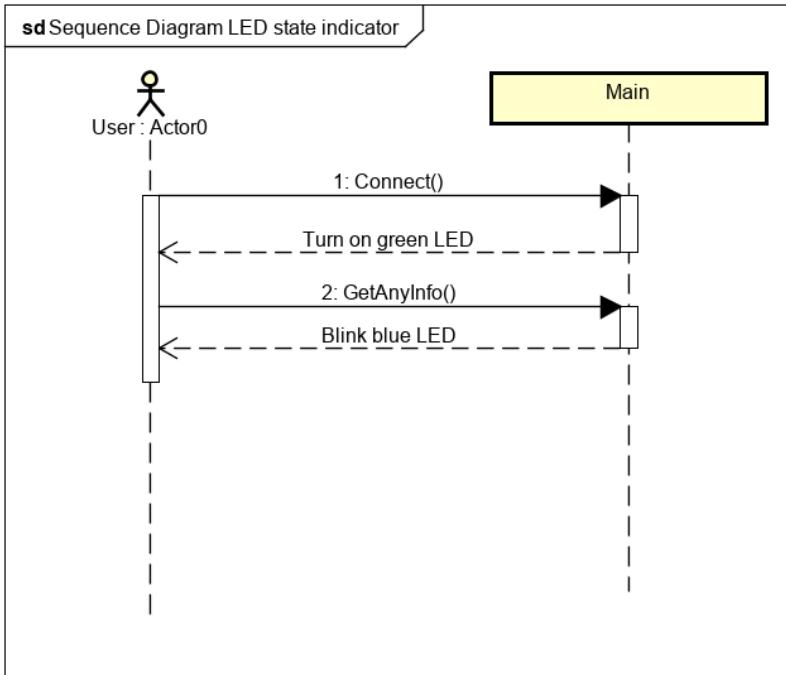


Use Case 14:

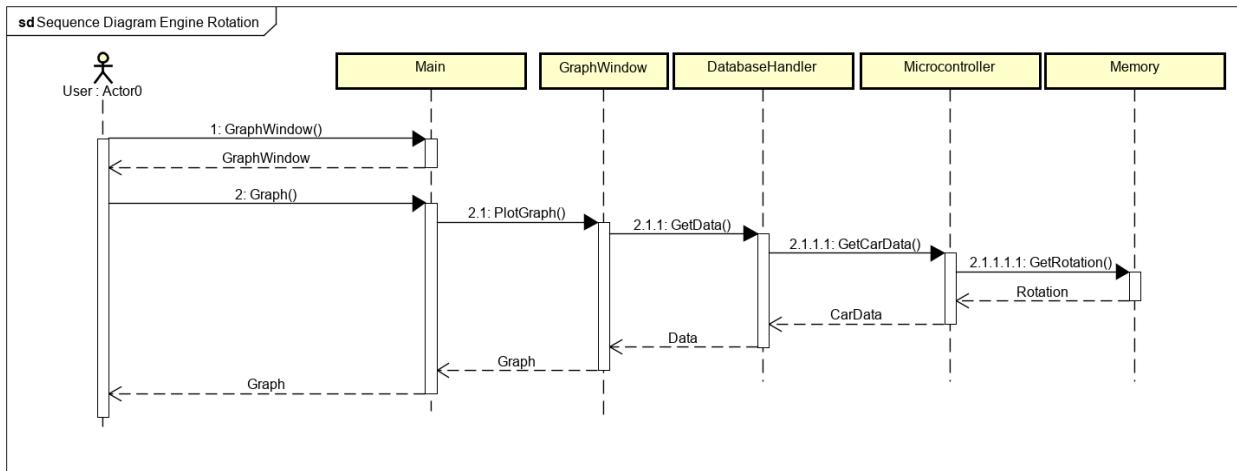


Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Use Case 15:

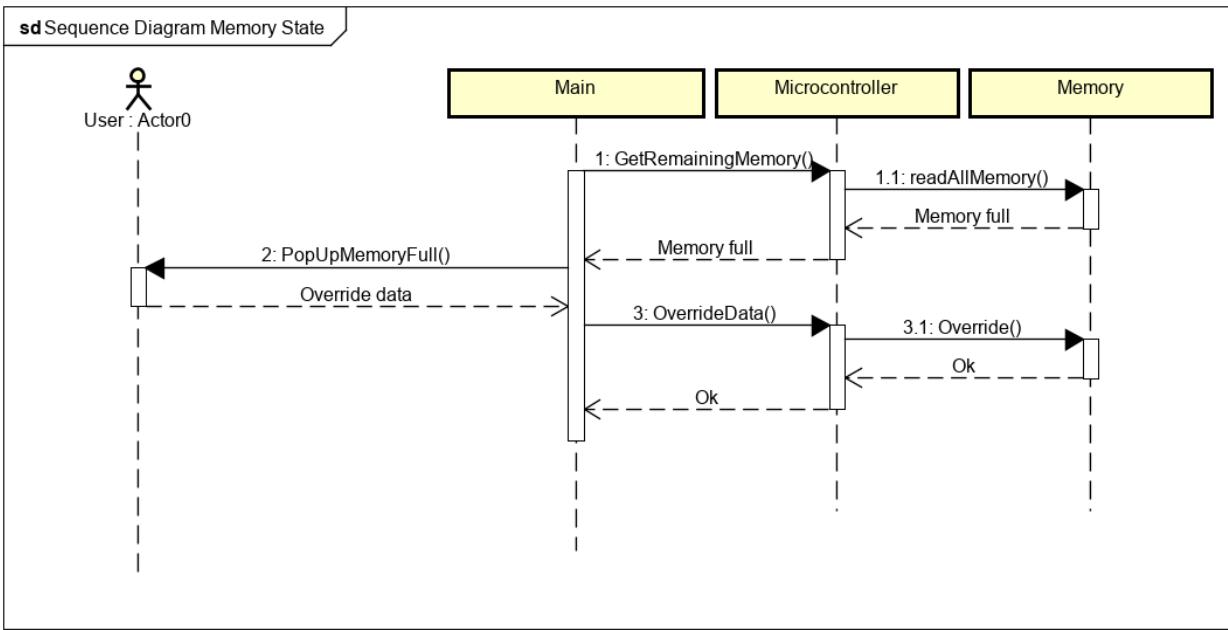


Use Case 16:

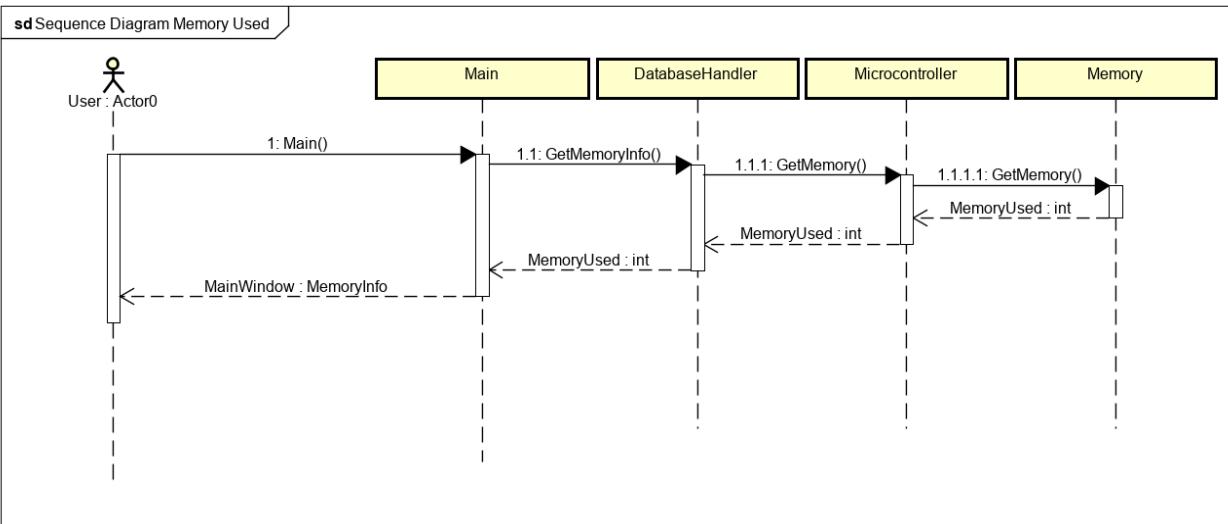


Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Use Case 17:

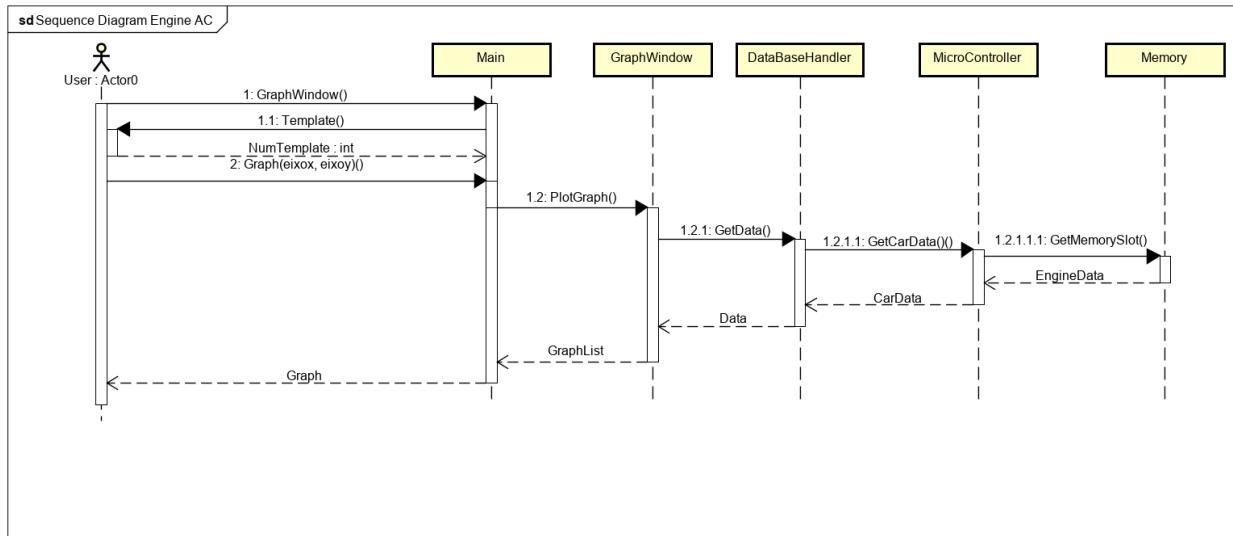


Use Case 18:



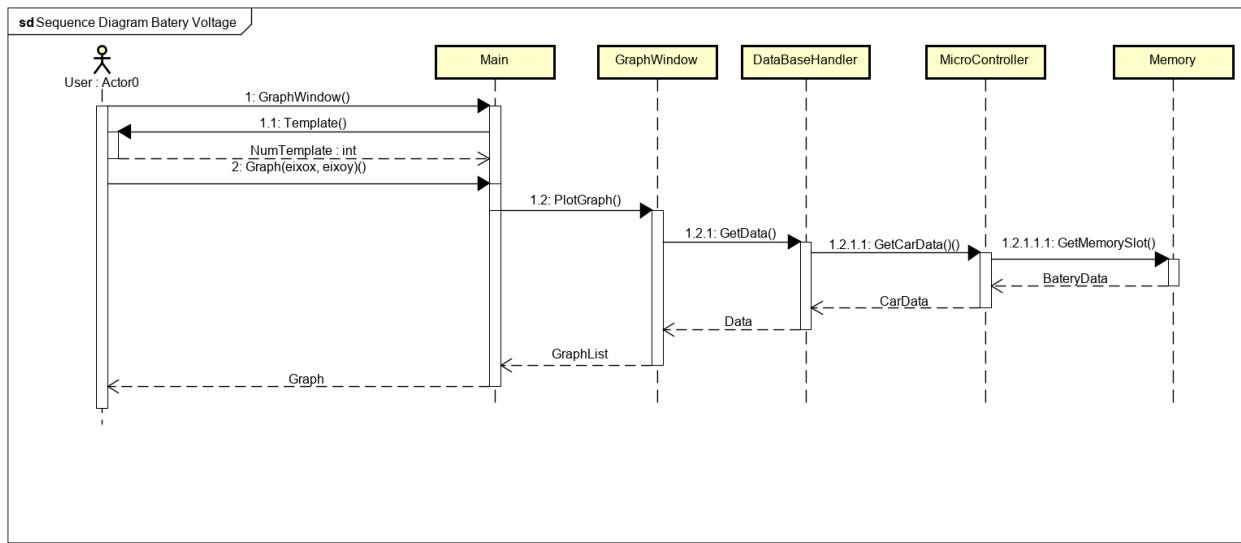
Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Use Case 19:

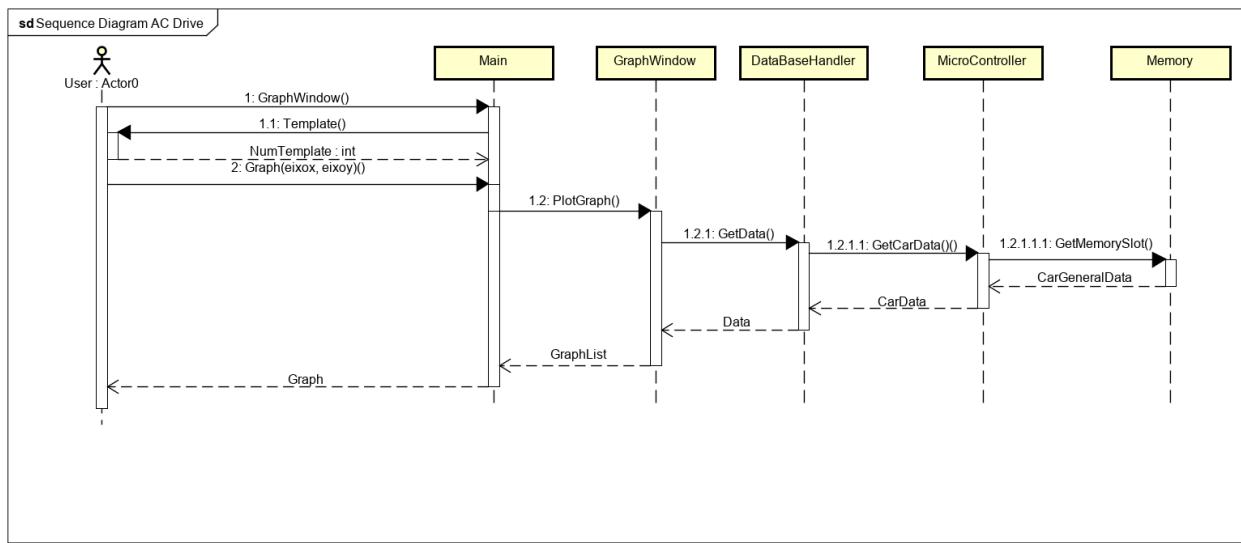


Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Use Case 20:

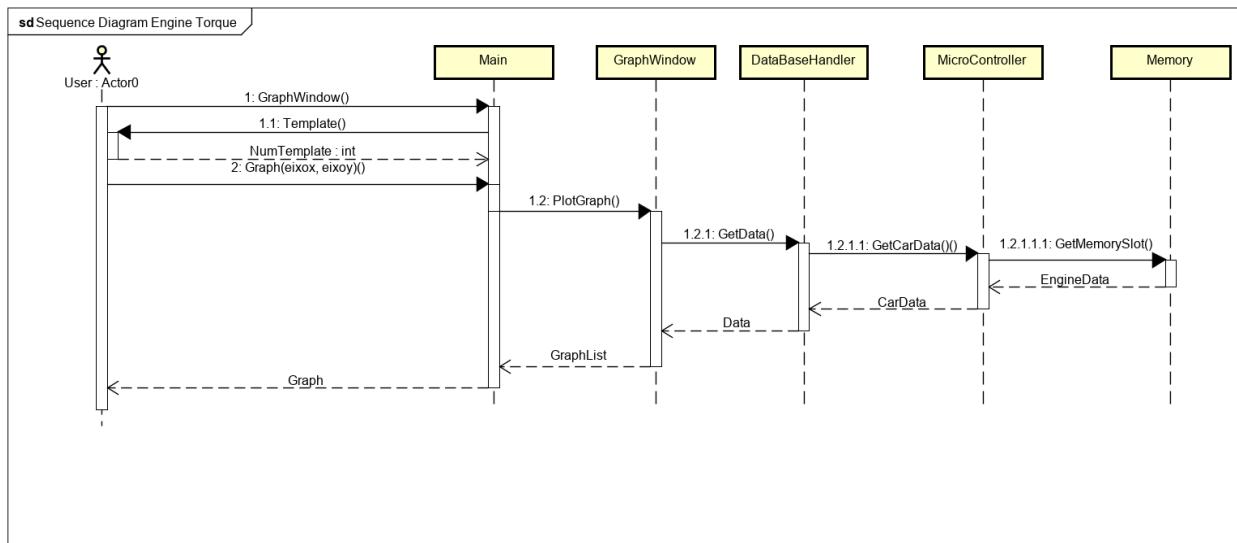


Use Case 21:

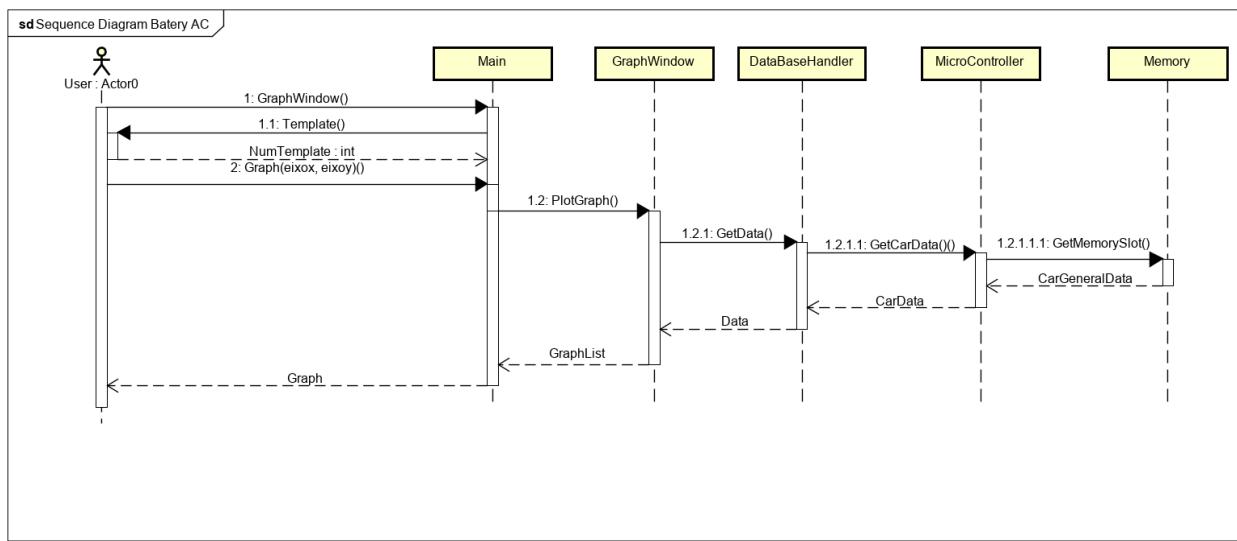


Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Use Case 22:



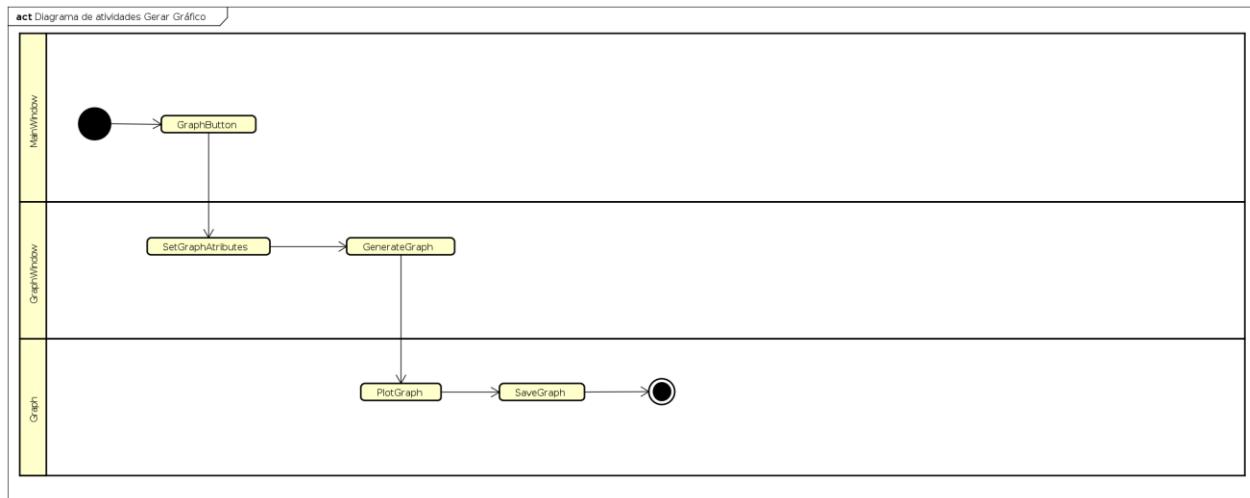
Use Case 23:



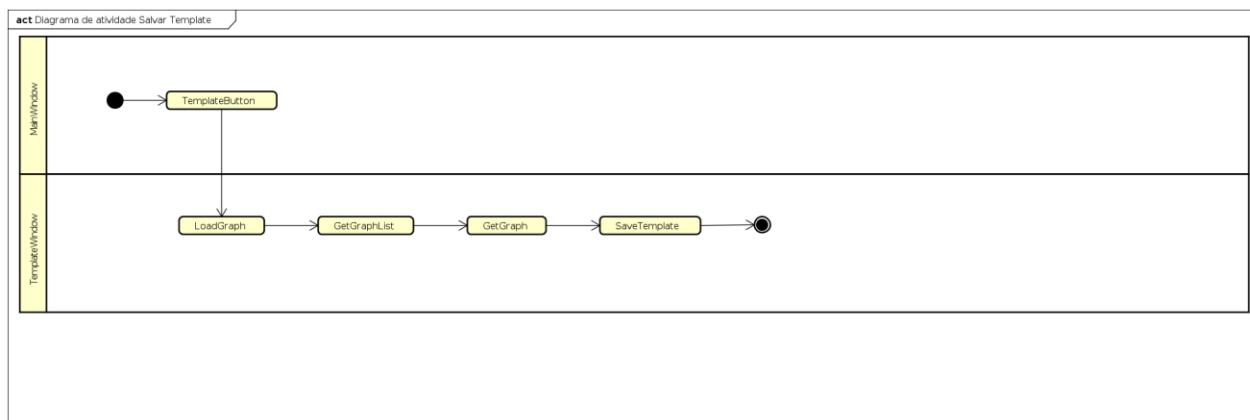
Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

4.5 Diagrama de Atividades

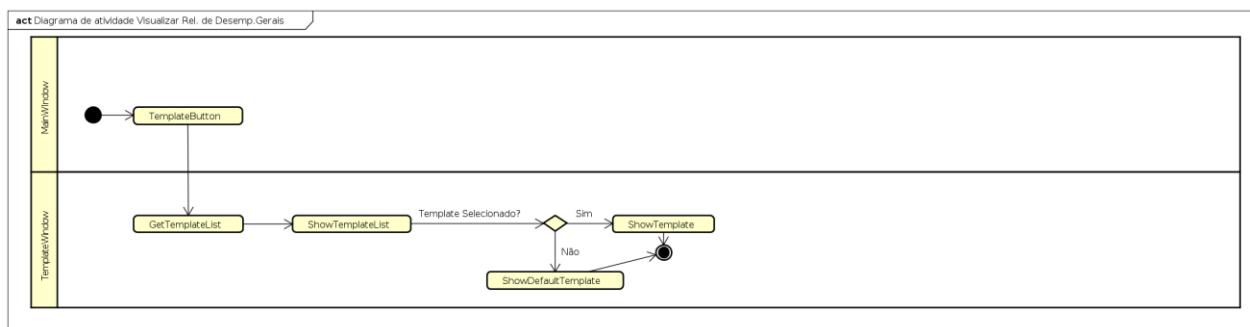
Use Case 1:



Use Case 2:

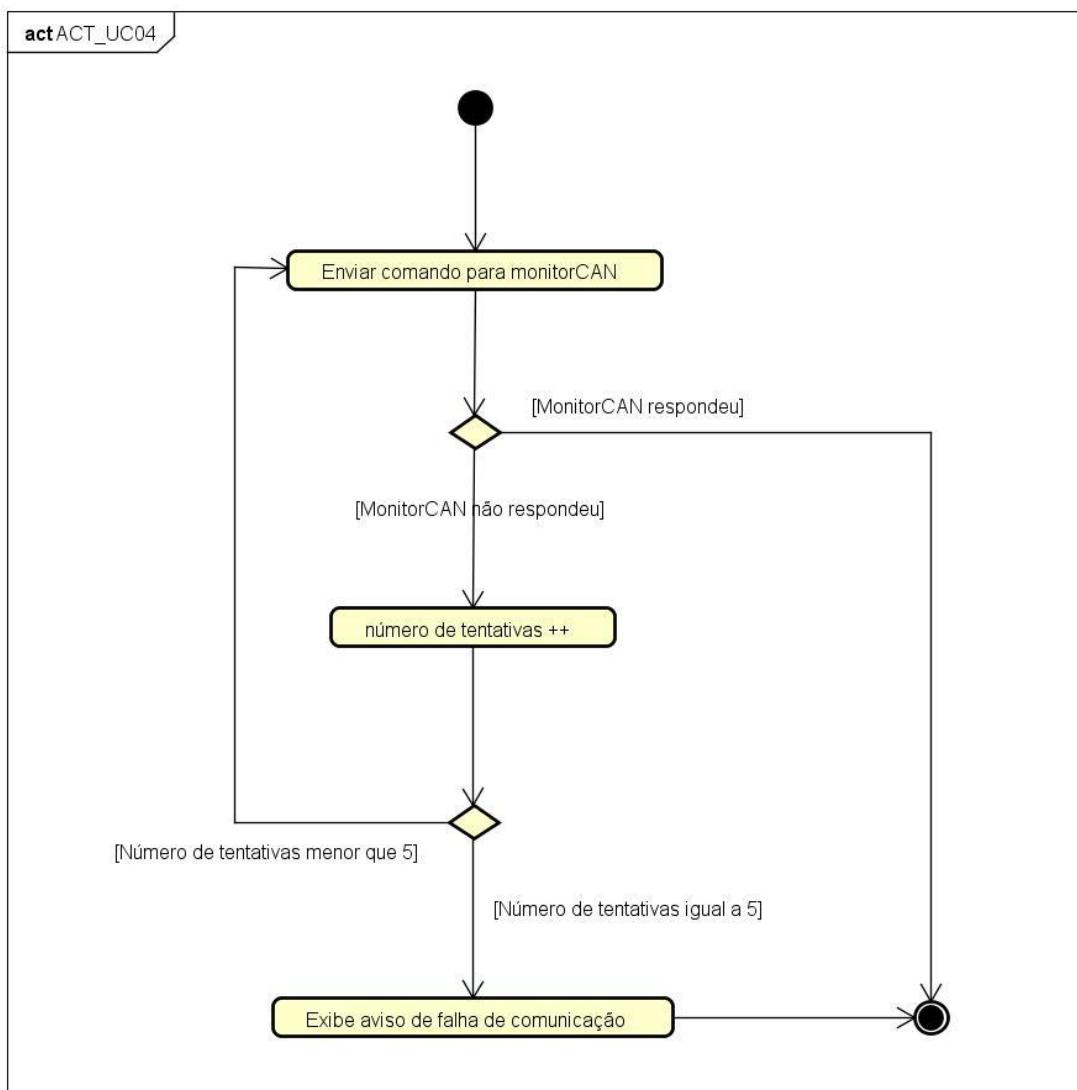


Use Case 3:



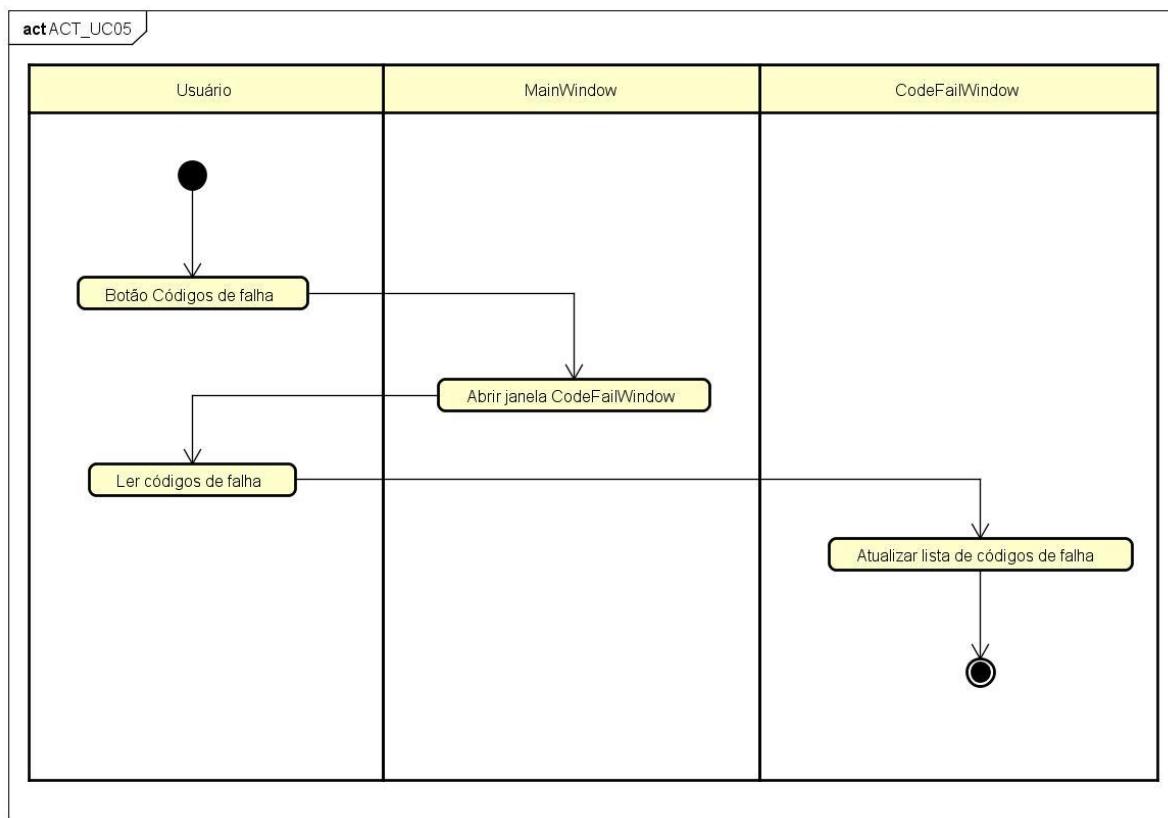
Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Use Case 4:



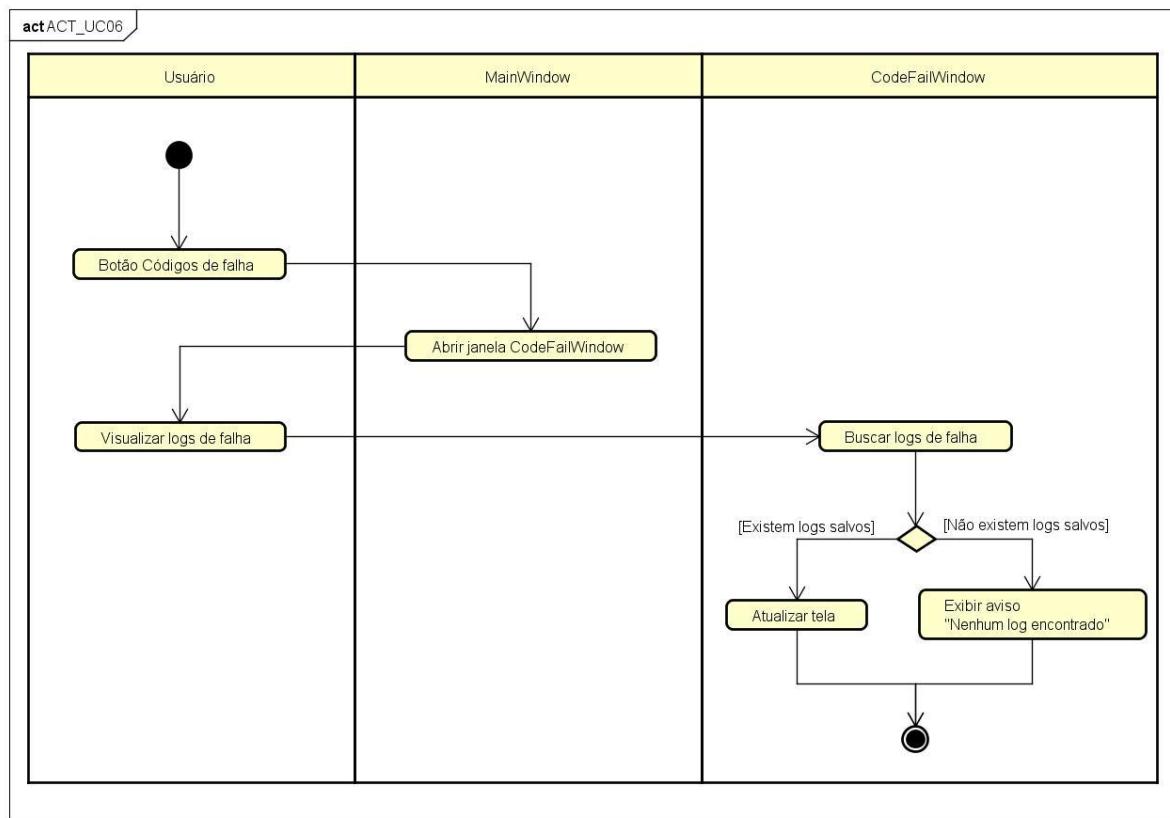
Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Use Case 5:



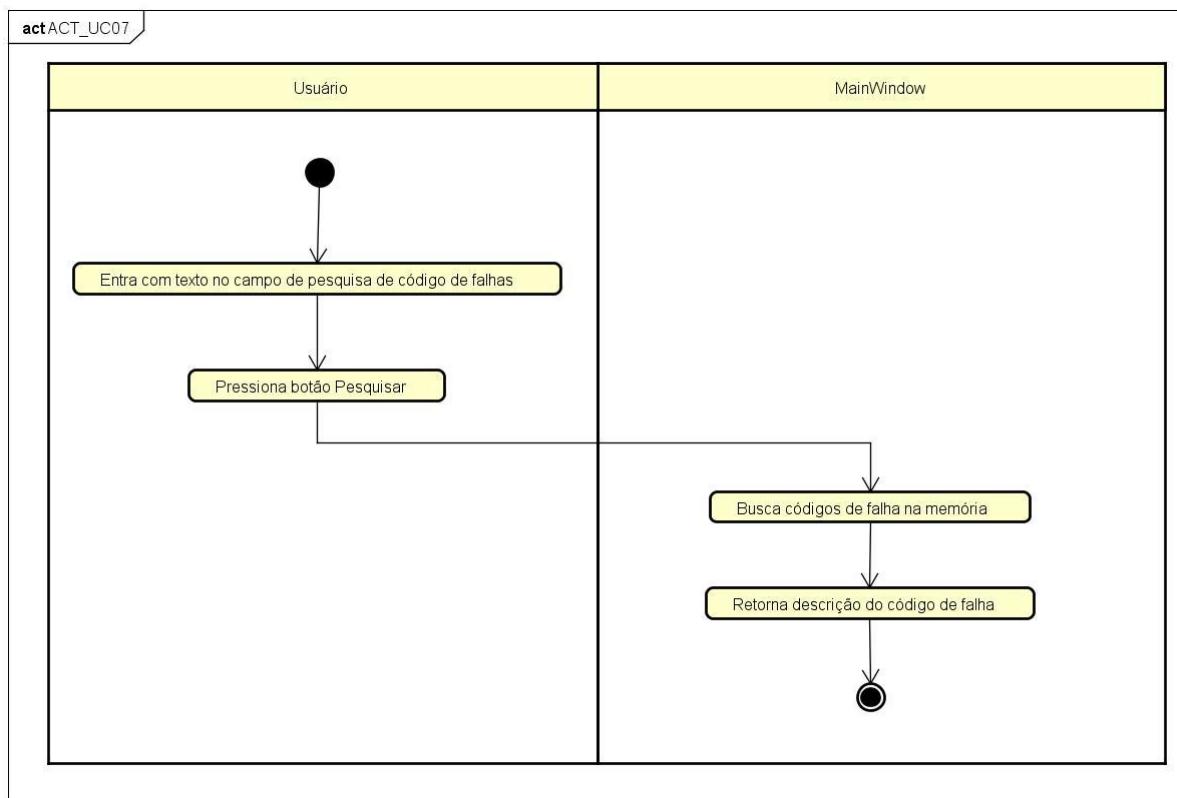
Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Use Case 6:



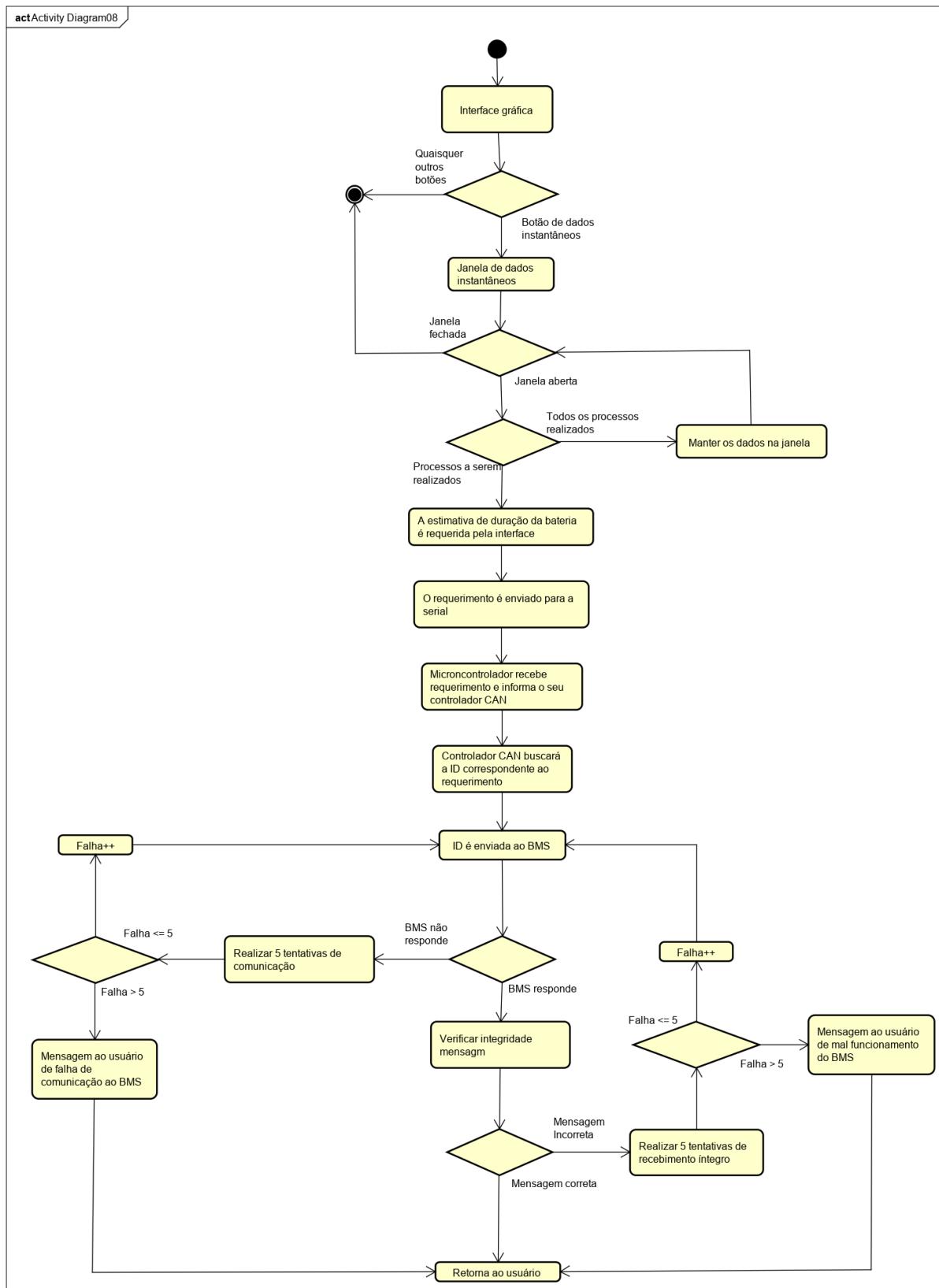
Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Use Case 7:



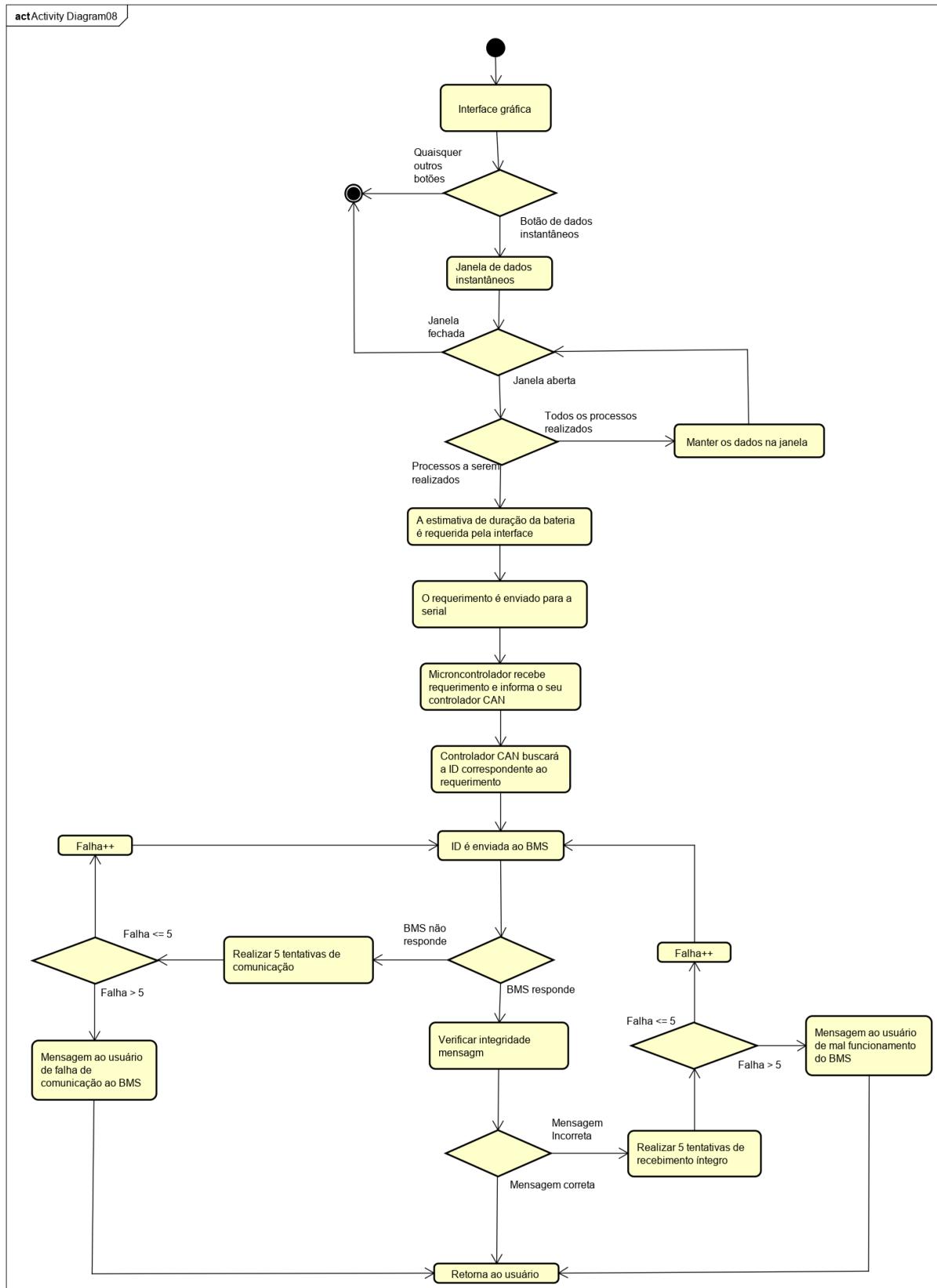
Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Use Case 8:



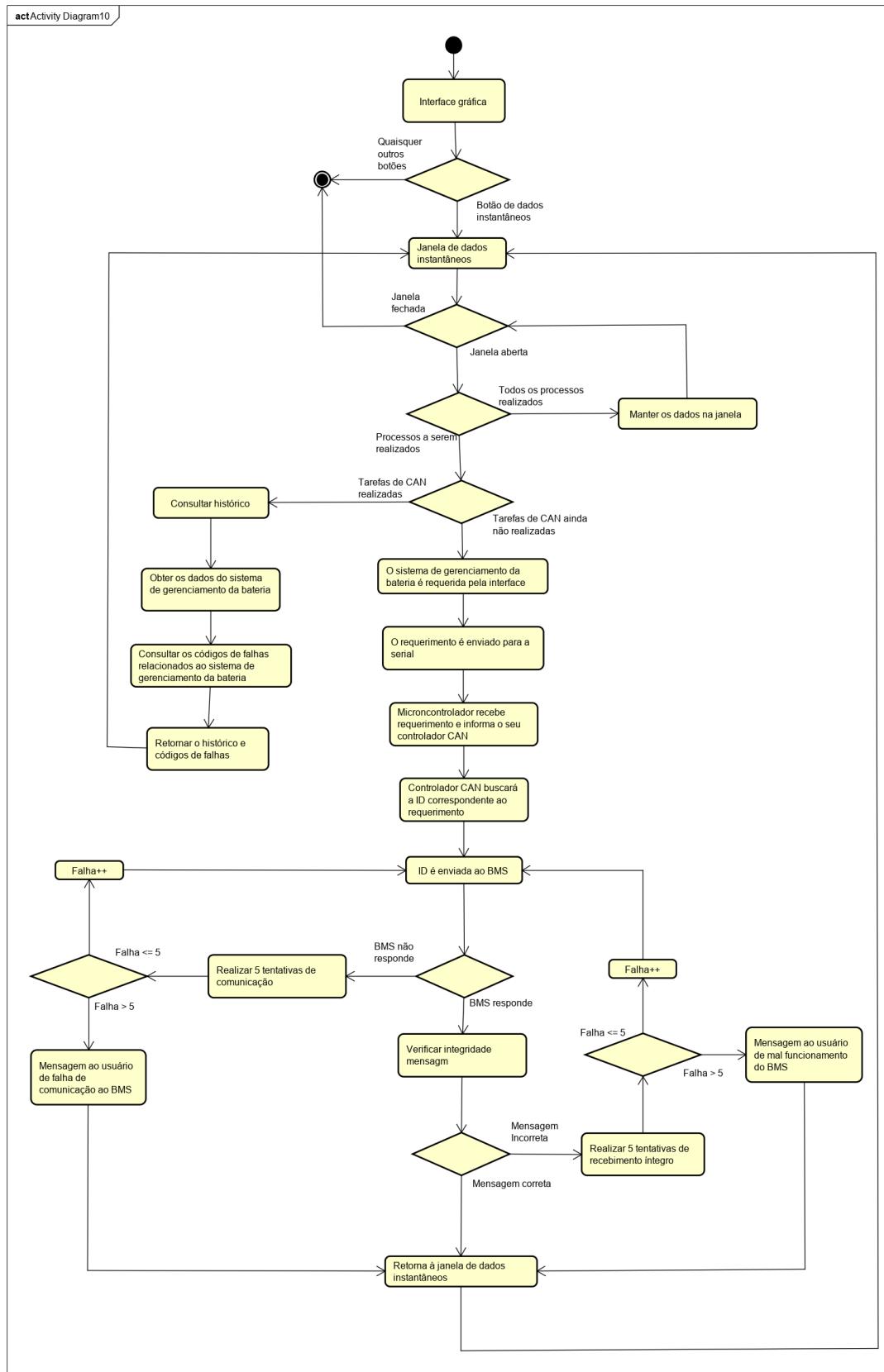
Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Use Case 9:



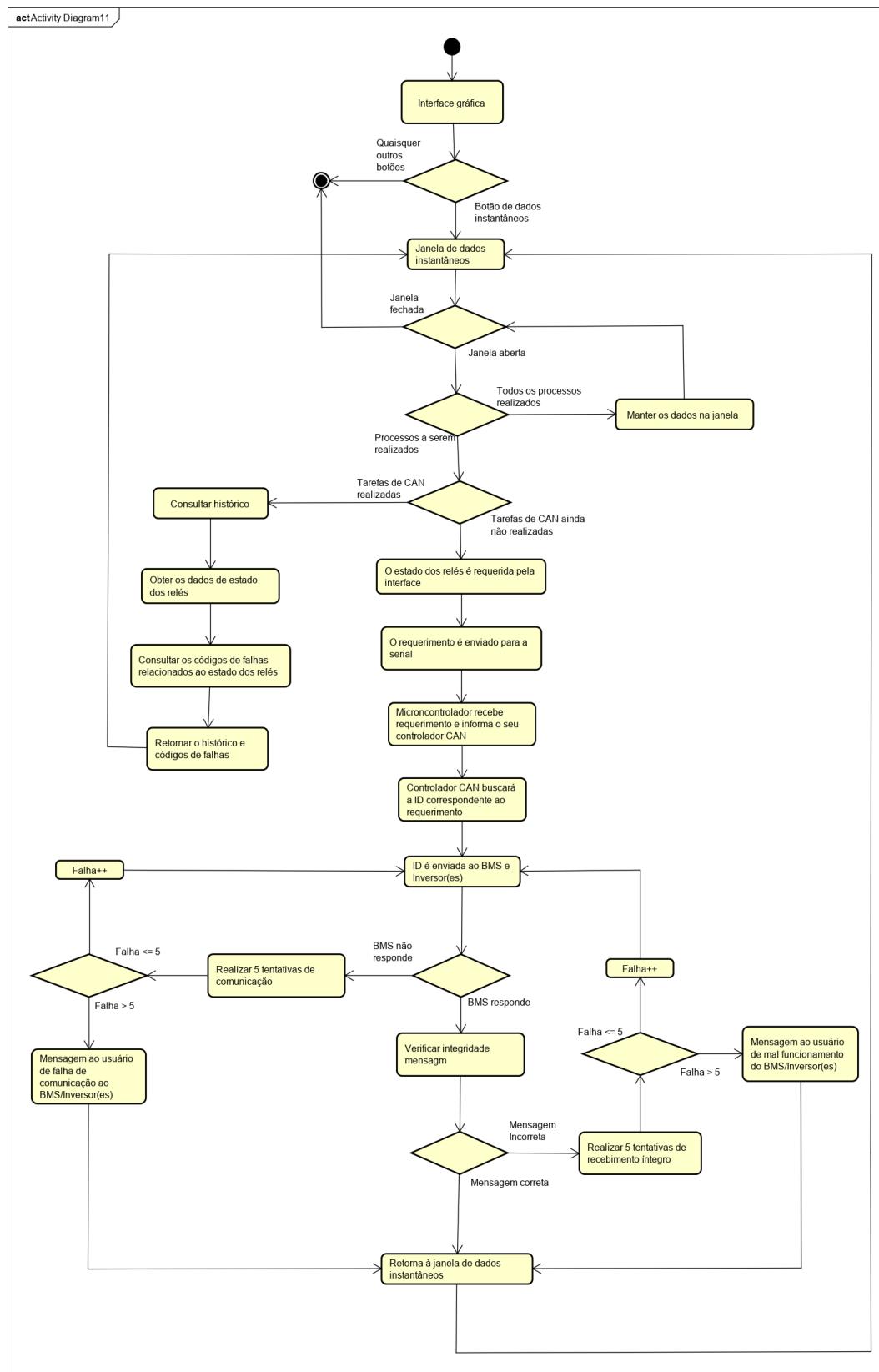
Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Use Case 10:



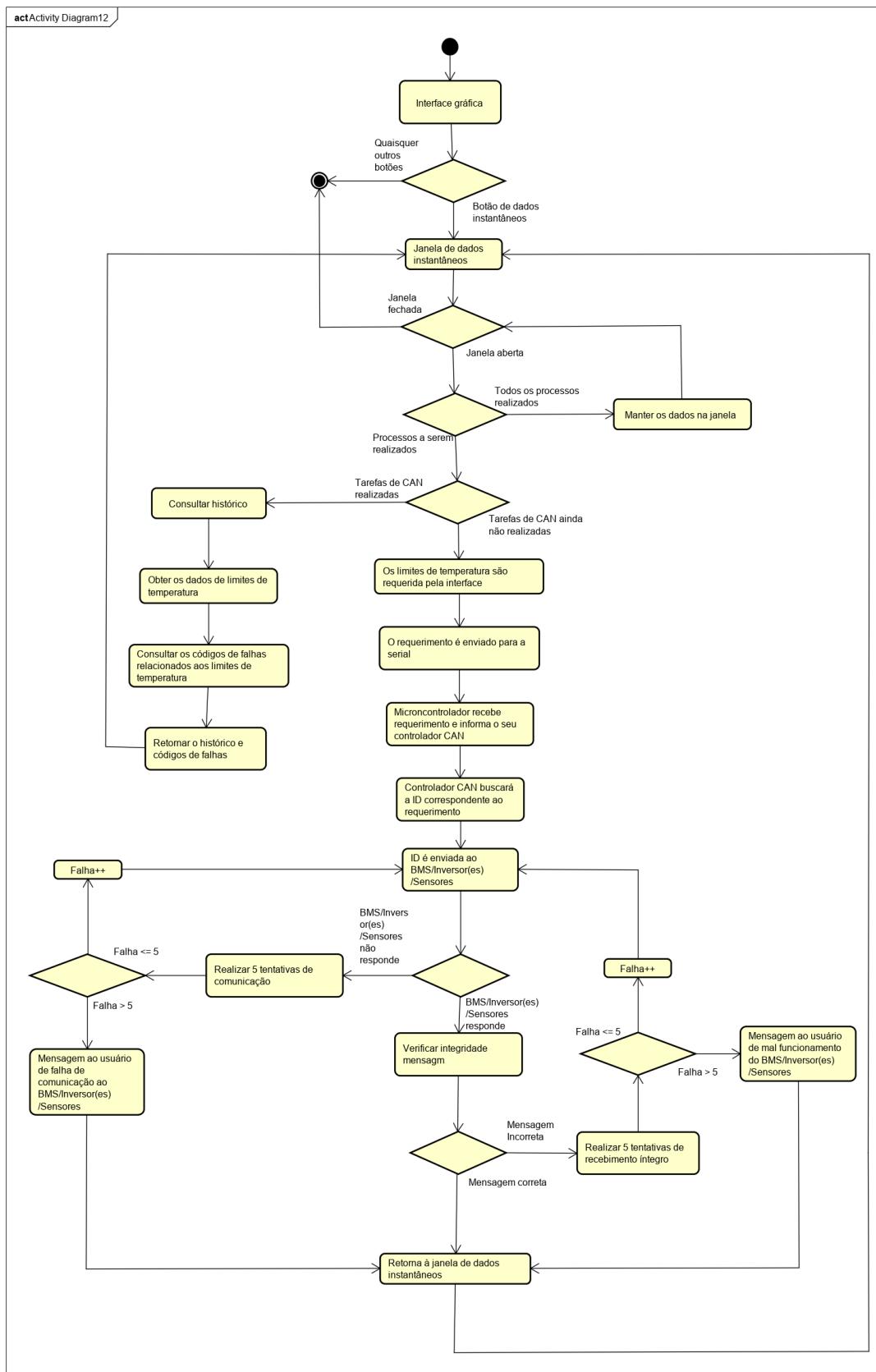
Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Use Case 11:

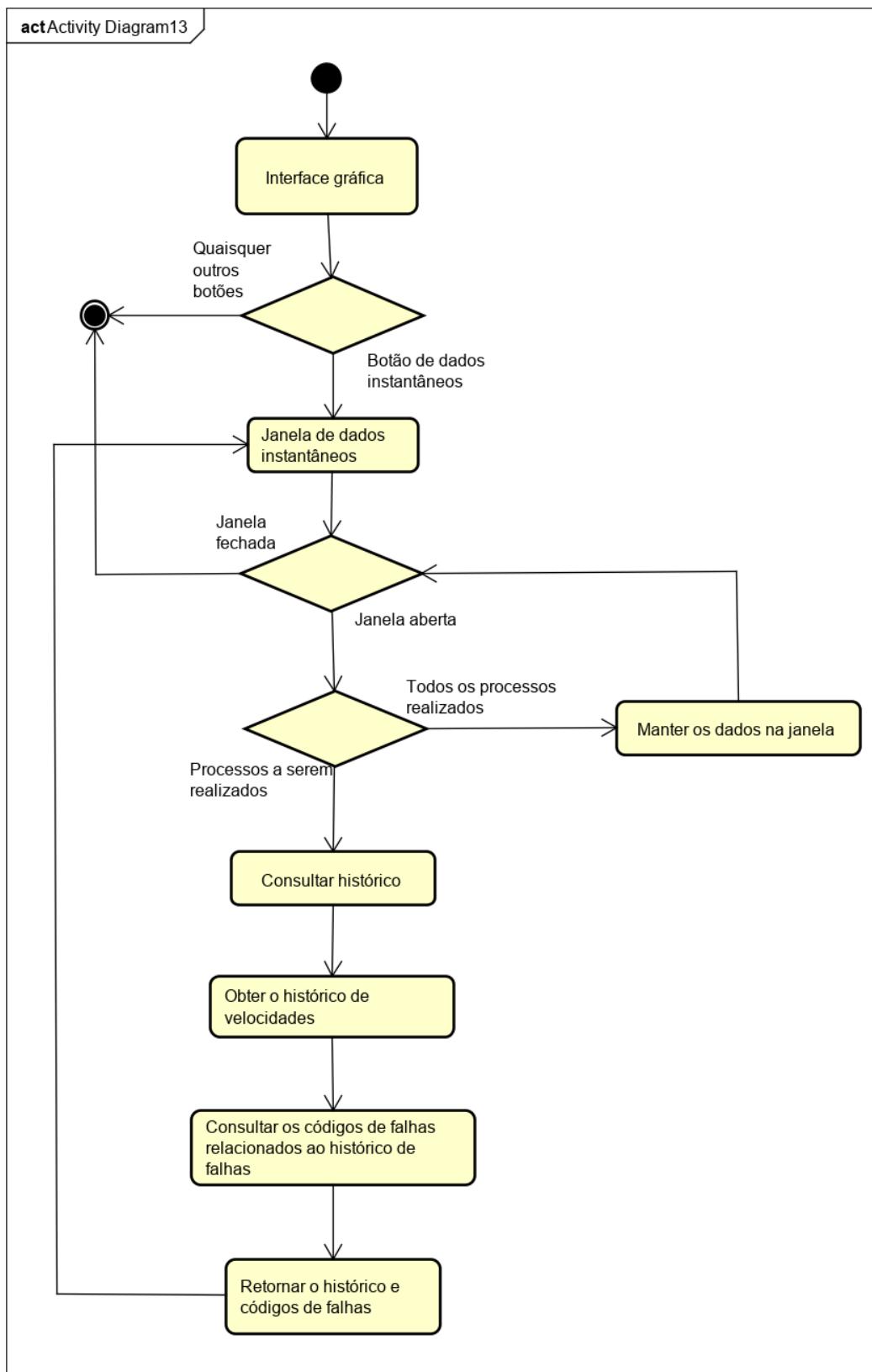


Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Use Case 12:

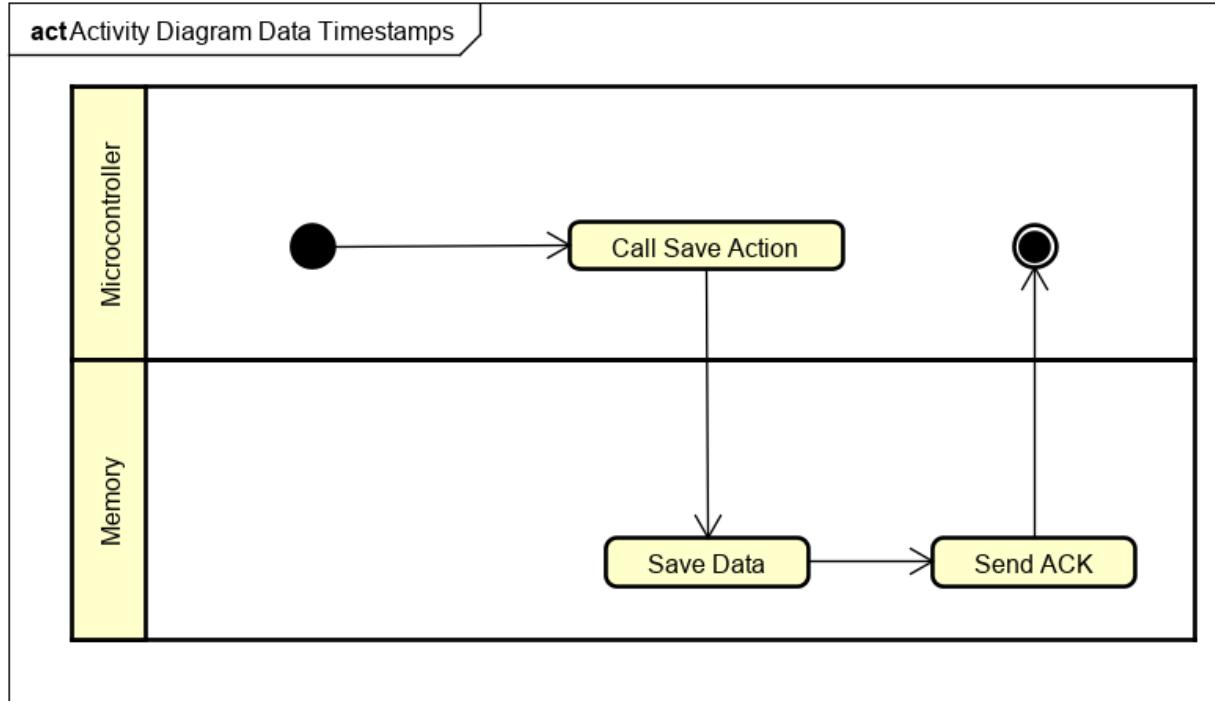


Use Case 13:

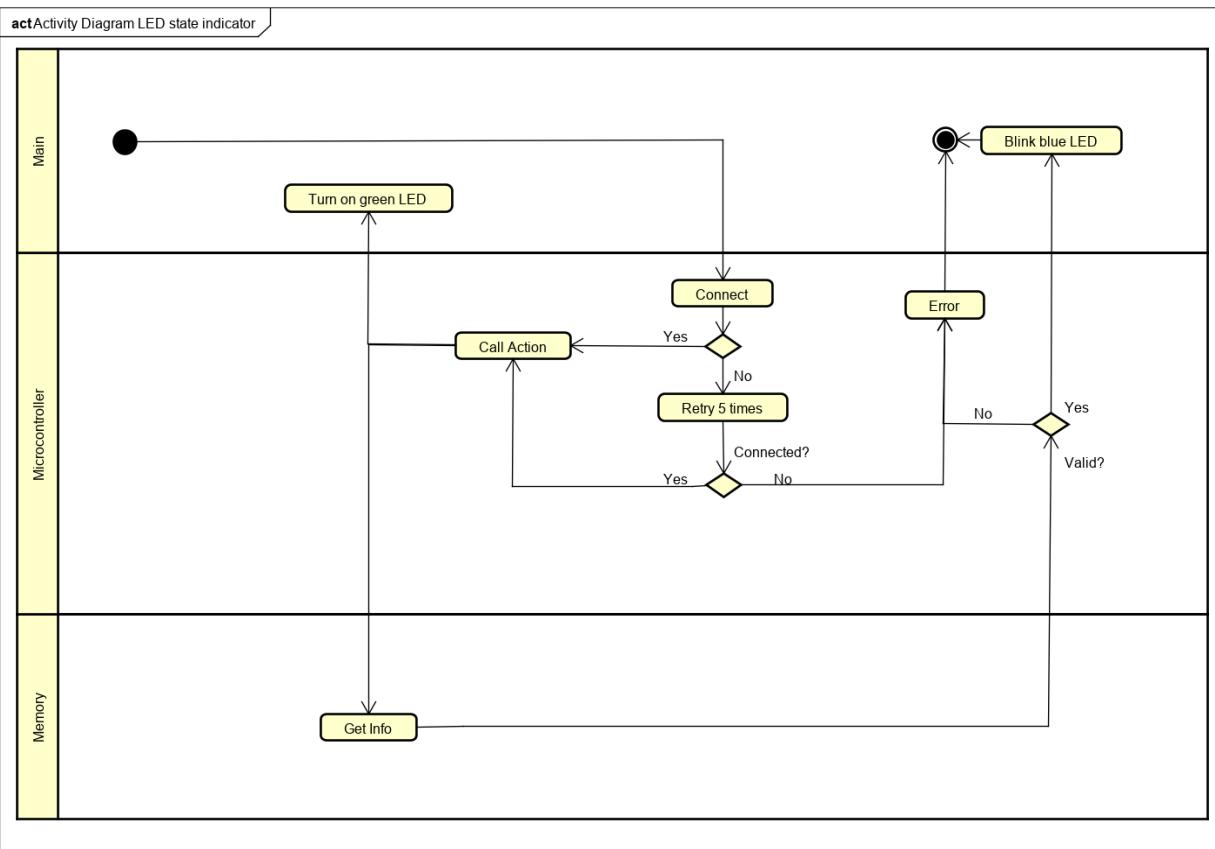


Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Use Case 14:

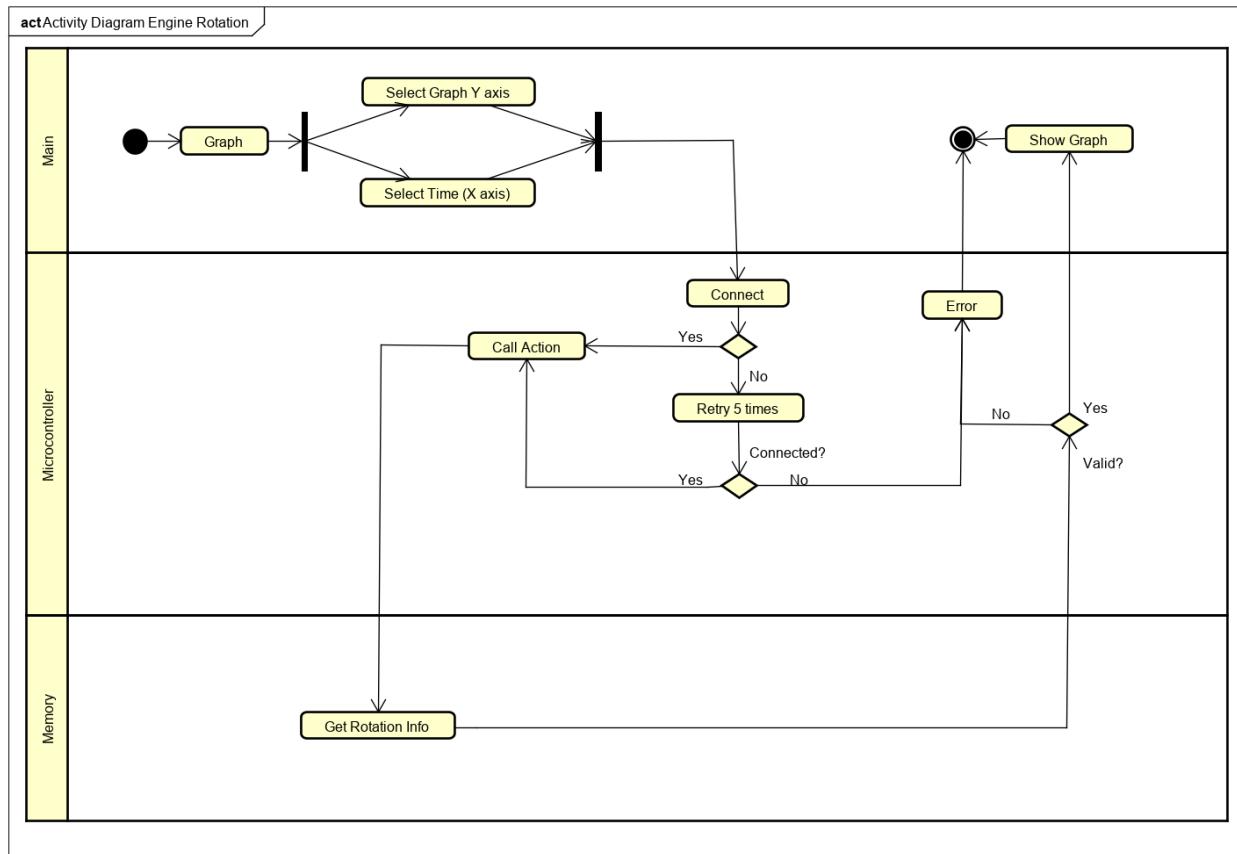


Use Case 15:



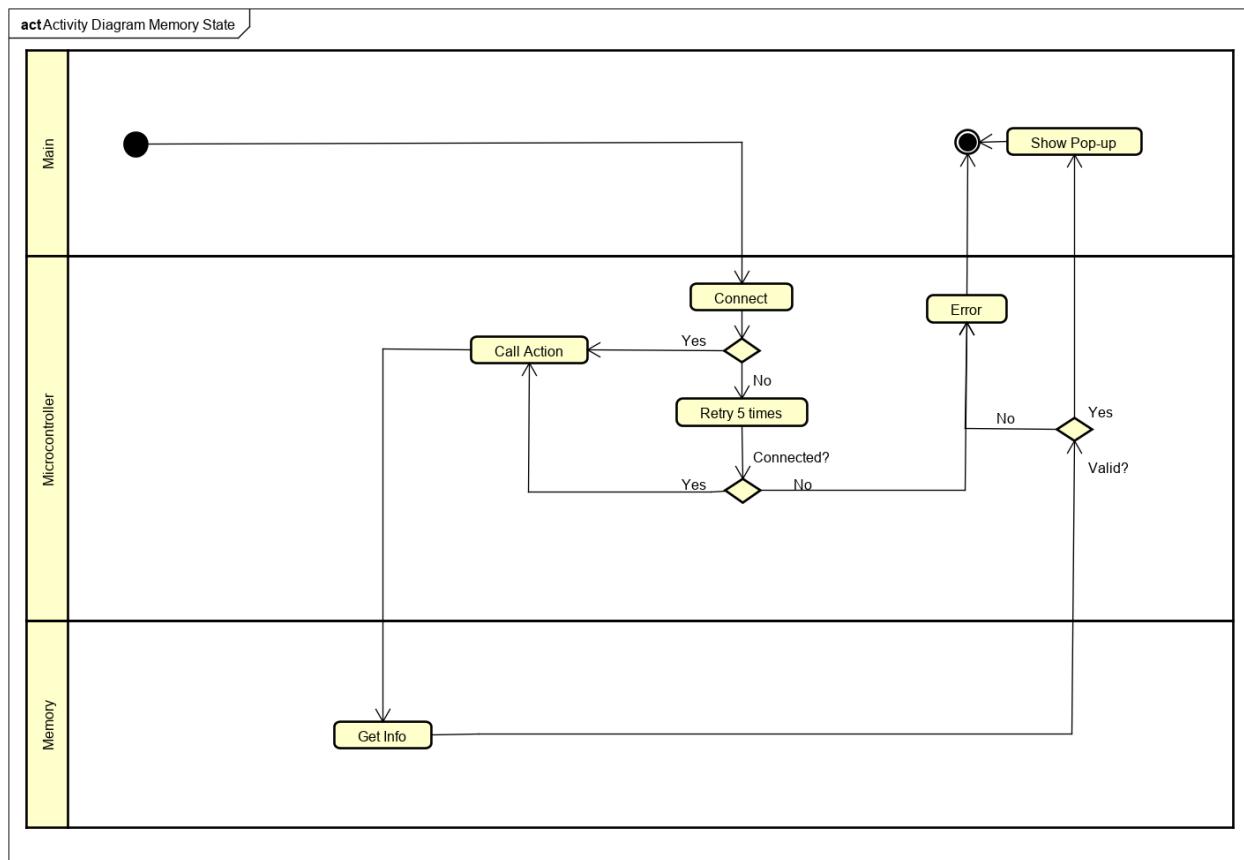
Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Use Case 16:



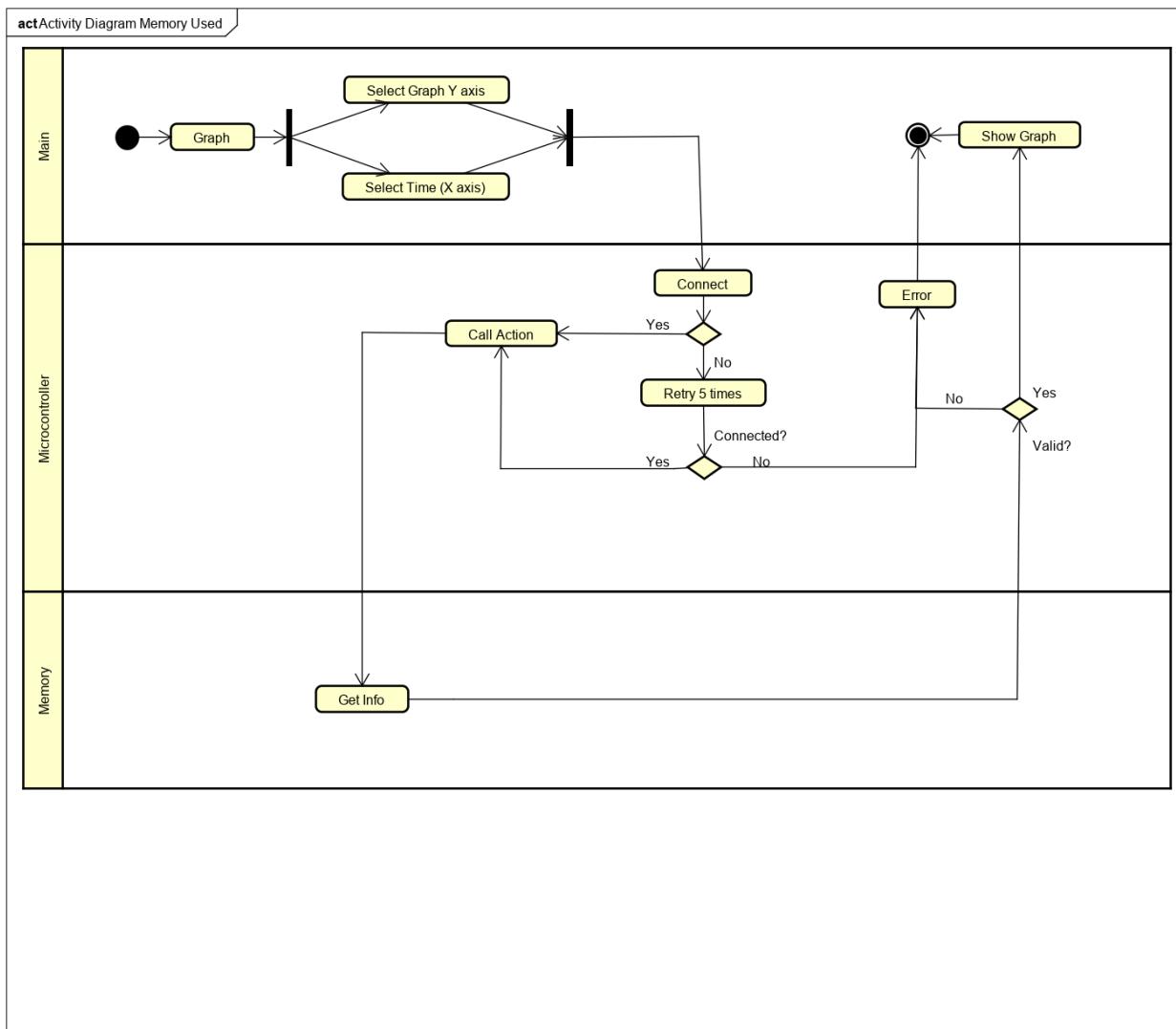
Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Use Case 17:



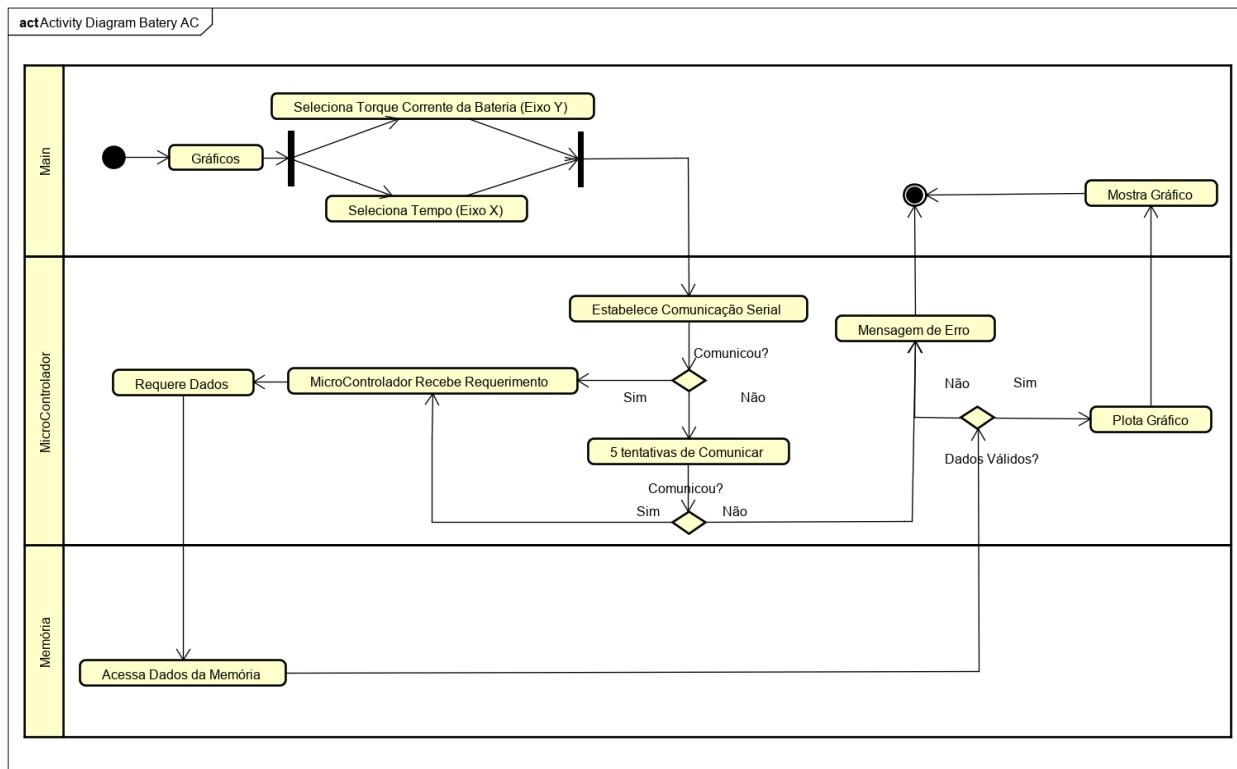
Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Use Case 18:

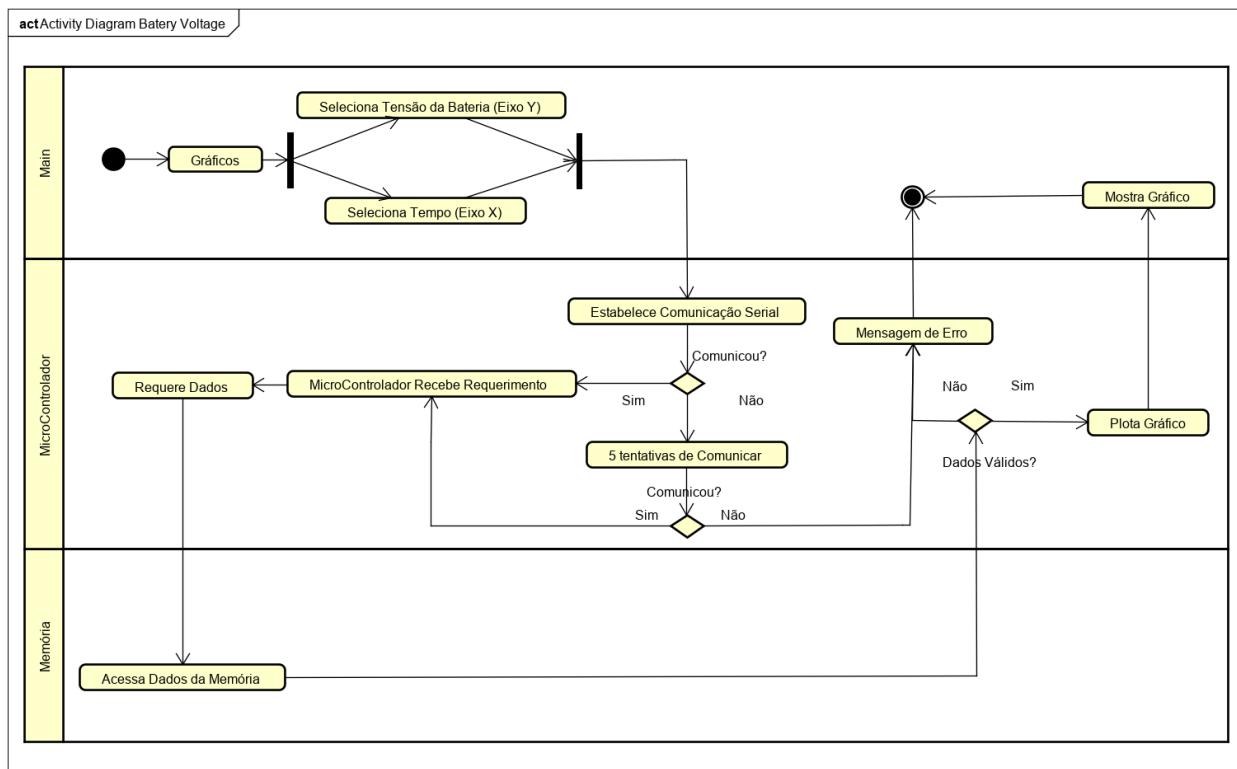


Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Use Case 19:

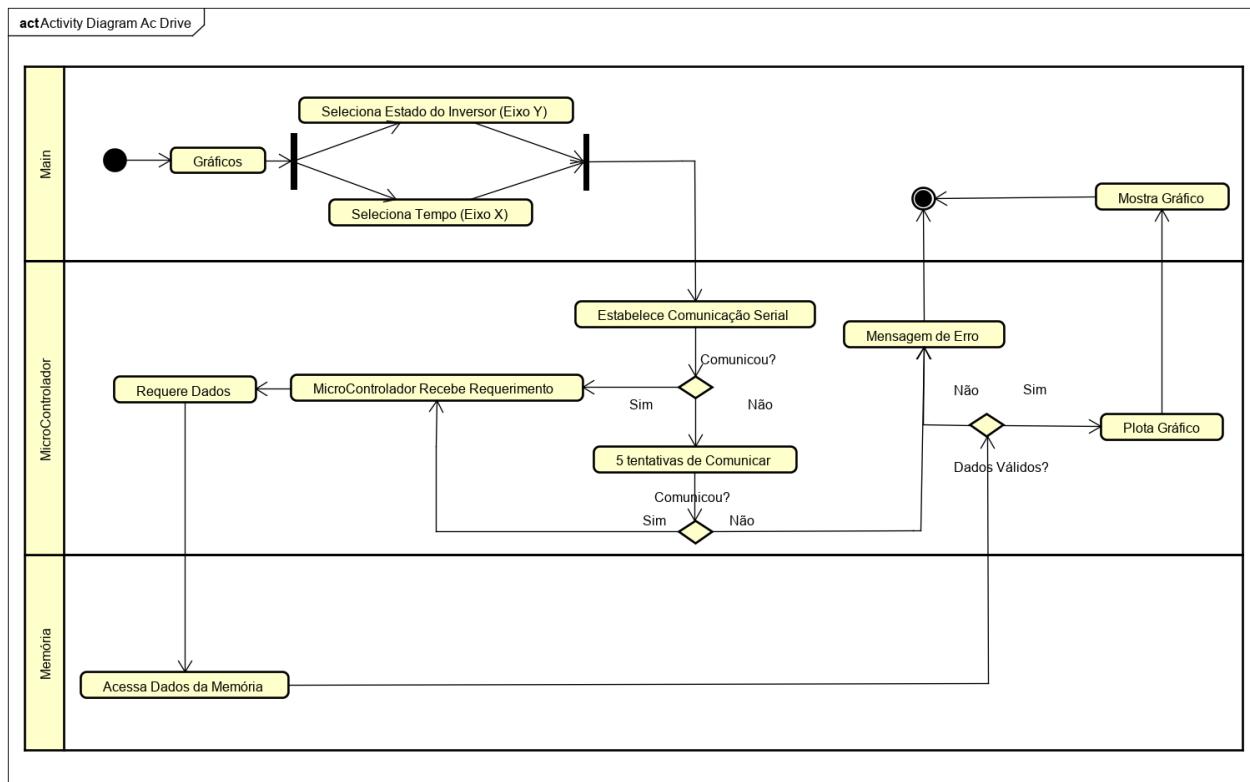


Use Case 20:

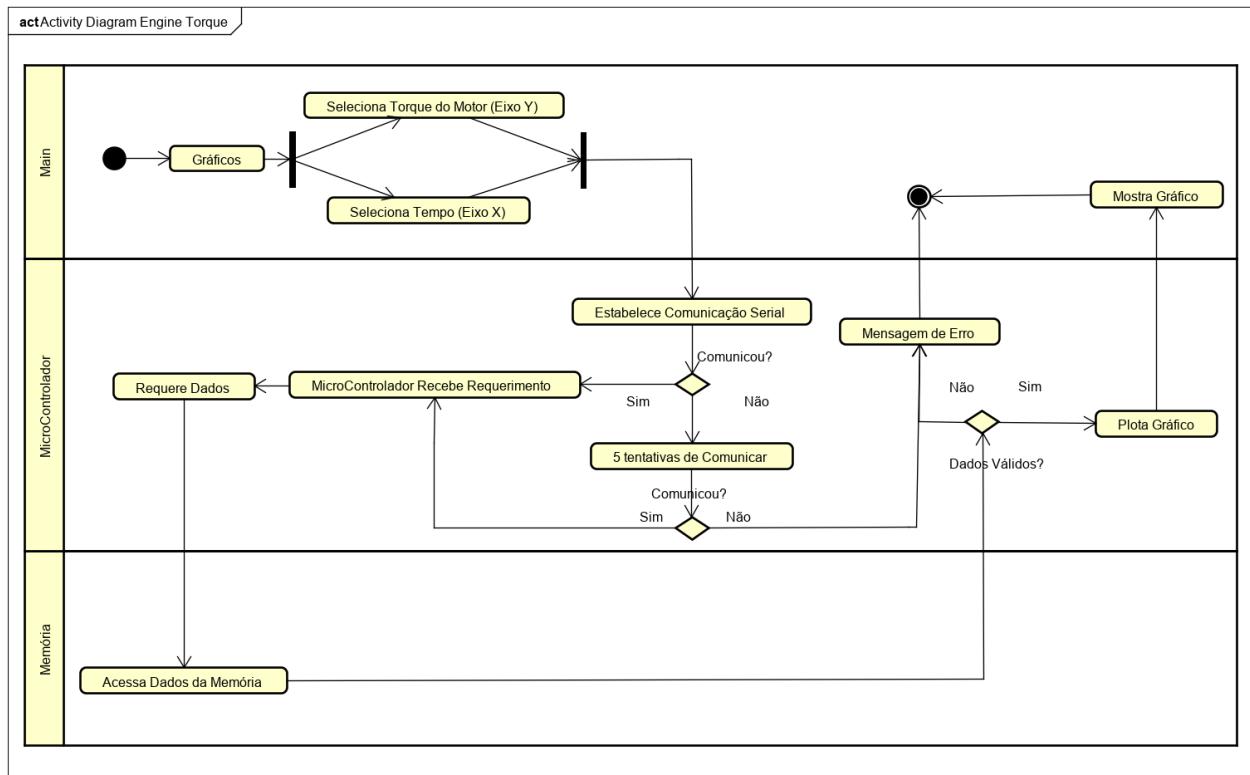


Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Use Case 21:

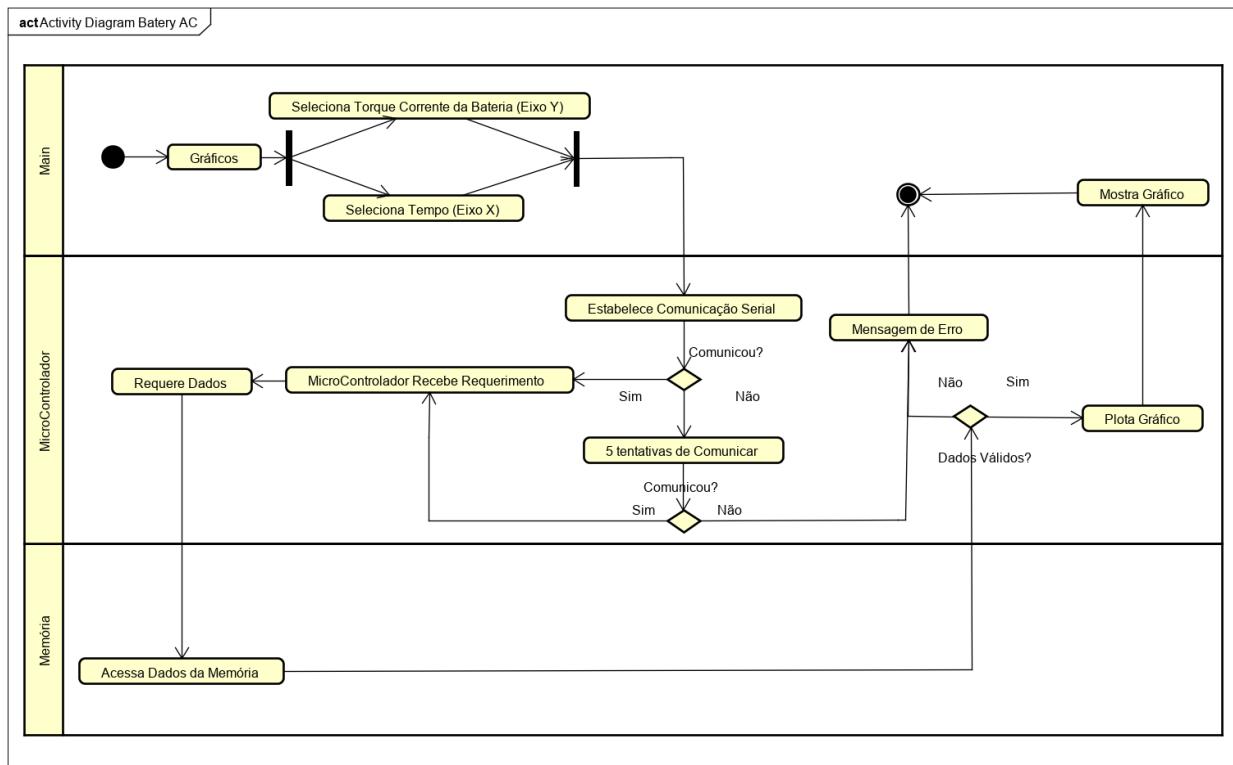


Use Case 22:



Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Use Case 23:

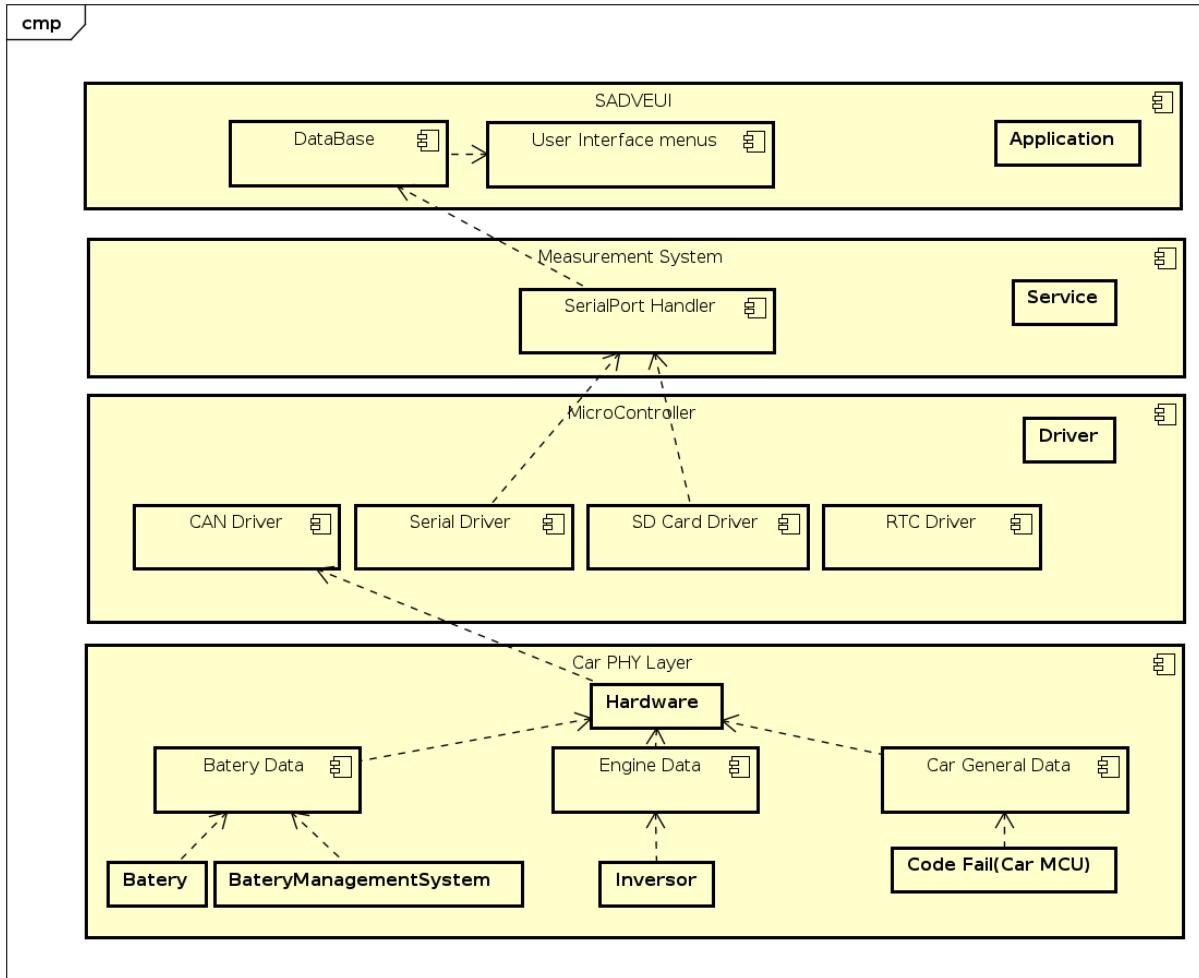


Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

5 ARQUITETURA DE COMPONENTES

A organização do SADVE pode ser observada em sua arquitetura, o qual é uma abstração da estrutura funcional do sistema, permitindo a decomposição dos componentes lógicos e permitindo que cada parte possa ser bem definida por meio dos seus métodos, atributos e objetos.

5.1 Padrão de arquitetura



Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

6 PROTÓTIPO

Manual do Usuário:

Agradecemos a preferência pelo SADVE!

Para inicializar adequadamente, siga os seguintes passos:

1. Instale em seu computador o nosso software SADVE. O instalador do software será disponibilizado junto com a compra dele. Futuramente, o instalador estará disponível no link: www.sadveenterprises.com/downloads.
2. Conecte o dispositivo SADVE na porta OBD2 de seu veículo. Caso não saiba a localização da porta OBD2 de seu veículo, consulte o fabricante.
3. Conecte o dispositivo SADVE na porta serial de seu computador.
4. Execute o SADVE.exe em seu computador. Abrirá o programa conforme a figura 1 abaixo:

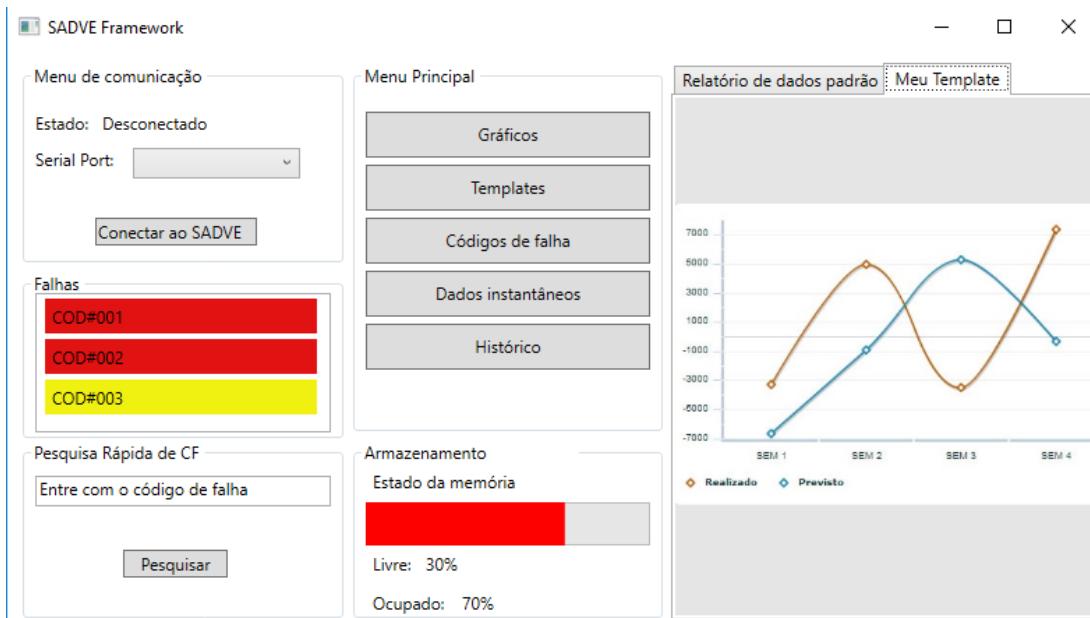


Figura 1

- Comunicando via porta serial

Na janela **Menu de comunicação** conforme a figura 2, selecione a porta serial em que o dispositivo SADVE está conectado (verifique se o dispositivo aparece em “Serial Port”, no Gerenciador de Dispositivos do seu computador), e solicite a comunicação clicando no botão **Conectar ao SADVE**. O software agora tem acesso a todos os dados registrados na memória.

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

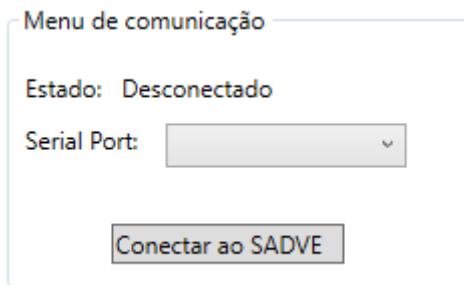


Figura 2

- Falhas

A partir do momento que houver conexão do dispositivo com o software, serão exibidas as falhas detectadas durante o uso de seu veículo elétrico. Todas serão listadas na caixa **Falhas** com os códigos de acordo com documento fornecido pelo fabricante do veículo conforme a figura 3. Para a documentação detalhada de cada falha, digite o código da falha na janela **Pesquisa Rápida de CF**.

Caso deseje mais dados referentes à falha, selecione a opção **Códigos de falha** no **Menu principal**.

A documentação poderá ser visualizada futuramente acessando o link:
www.sadveenterprises.com/documentação (em desenvolvimento).

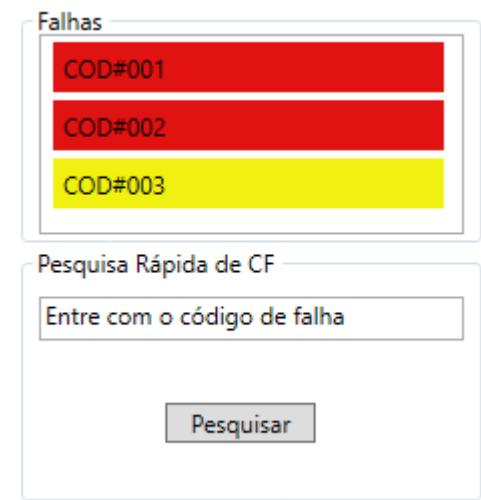


Figura 3

- Gráficos

O software SADVE oferece a opção de visualização dos dados armazenados em gráficos. Siga os passos a seguir

1. Selecione a opção “Gráficos” no Menu Principal mostrado na figura 4

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19



Figura 4

2. Configure qual Template será utilizado para representar os gráficos. Aperte OK
3. Selecione quais dados serão mostrados na janela **Eixo Y** de acordo com o Template escolhido e o espaço de tempo em que eles foram lidos no **Eixo X** mostrados na figura 5.

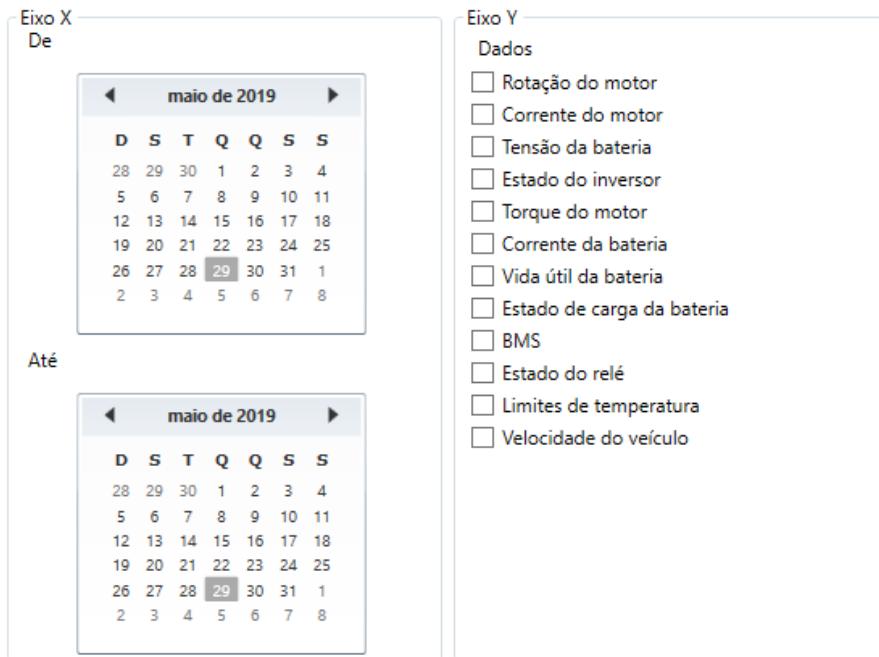


Figura 5

4. O gráfico será gerado como na figura 6
 - a. Após o uso do gráfico, selecione **Salvar Gráfico** caso deseje visualizar depois.
 - b. Caso deseje usar o gráfico em outras plataformas, selecione **Exportar para PDF**.

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

- c. Caso os dados não são pertinentes, selecione a opção **Descartar dados** para remove-los da memória
- d. Se deseja gerar um novo gráfico alterando alguns parâmetros, selecione a opção **Redefinir Gráfico**.

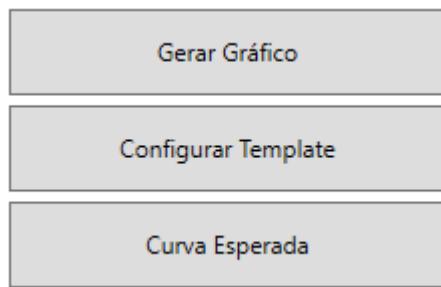


Figura 6

O Software SADVE possui a opção **Curva Esperada** mostrada na figura 6, fornecida pelos testes dos fabricantes, na qual oferece comparativo visual entre os dados lidos e os dados esperados

- Templates

Na opção **Templates** é possível definir e gravar modelos de apresentação dos gráficos. É possível selecionar gráficos salvos na opção **Carregar Gráficos** e dispor conforme a sua necessidade. A opção **Salvar Template** grava em memória para serem acessados e modificados depois com a opção **Carregar Template**. Para visualização em PDF, é só selecionar a opção **Exportar Template**.

- Dados Instantâneos

O software SADVE oferece a análise dos dados em tempo real. Ao selecionar a opção **Dados Instantâneos**, é apresentado em tela todos os dados momentâneos ao seu veículo assim como na figura 7.

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Dados Instantâneos	
Rotação do motor	Estado do inversor
1200 RPM	Label
Corrente do motor	Tensão da bateria
300A	64.1V
Corrente da bateria	BMS
Label	Label
Torque do motor	Vida útil da bateria
Label	Label
Temperatura do motor	Carga da bateria
Label	Label
Temperatura dos inversores	Estado do relé
Label	Label

Figura 7

- Armazenamento

O dispositivo SADVE é compatível com cartões microSD cujo estado da memória será mostrado na janela **Armazenamento**, conforme figura 8. Quando a memória estiver cheia, o software irá sinalizar ao usuário solicitando a exclusão de dados ou a troca do cartão microSD.

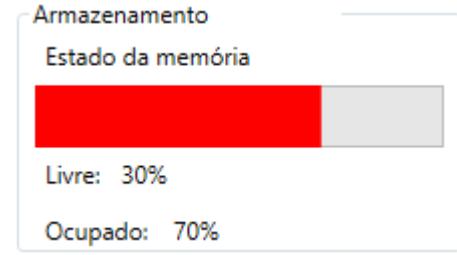


Figura 8

- Histórico

Na opção **Histórico**, será disponibilizado ao usuário uma tabela com todos os dados armazenados na memória do dispositivo SADVE. Selecione a opção **Excluir dados** para remover da memória os dados selecionados.

- Relatório de dados padrão

A aba **Relatório de dados Padrão** mostrada na figura 9, segue todas as métricas estabelecidas pelos fabricantes para visualização dos dados do veículo elétrico, a qual adequa os gráficos salvos pelo usuário e os dispõe para visualização.

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

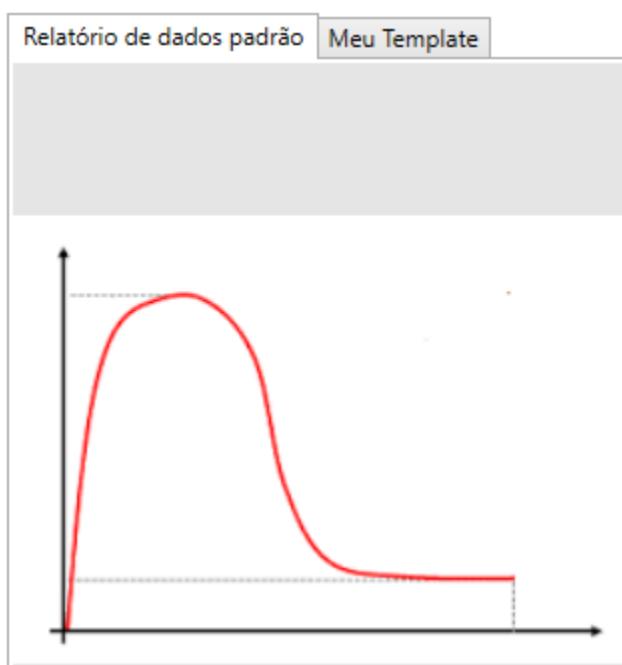


Figura 9

- Meus Templates

Será listado na aba **Meu Template** demonstrado na figura 10, todos os Templates criados pelo usuário. Caso deseja remover algum Template, vá na opção **Templates > Remover Template**.



Figura 10

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Agradecemos a leitura do manual de uso do SADVE, e qualquer dúvida poderá ser sanada nos
contatando via:
www.sadveenterprises.com/contactus

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

7 ITEM DE CRIATIVIDADE

O logo, o instalador do SADVE e a utilização do GitHub foram os itens de criatividade utilizados na elaboração desta fase do projeto.

GitHub:

GitHub é um sistema de controle de versão de arquivos. Através deles podemos desenvolver projetos na qual diversas pessoas podem contribuir simultaneamente no mesmo, editando e criando arquivos e permitindo que eles possam existir sem o risco de suas alterações serem sobreescritas. Nossa equipe realizou todo o versionamento do código e do projeto através do GitHub, uma das plataformas online mais conhecidas para versionamento e o repositório está público para possíveis contribuições da comunidade e manutenção do projeto. As imagens abaixo mostram o mapeamento de todos os commits desde a criação do projeto (figura 11) e o repositório da equipe utilizado (figura12).

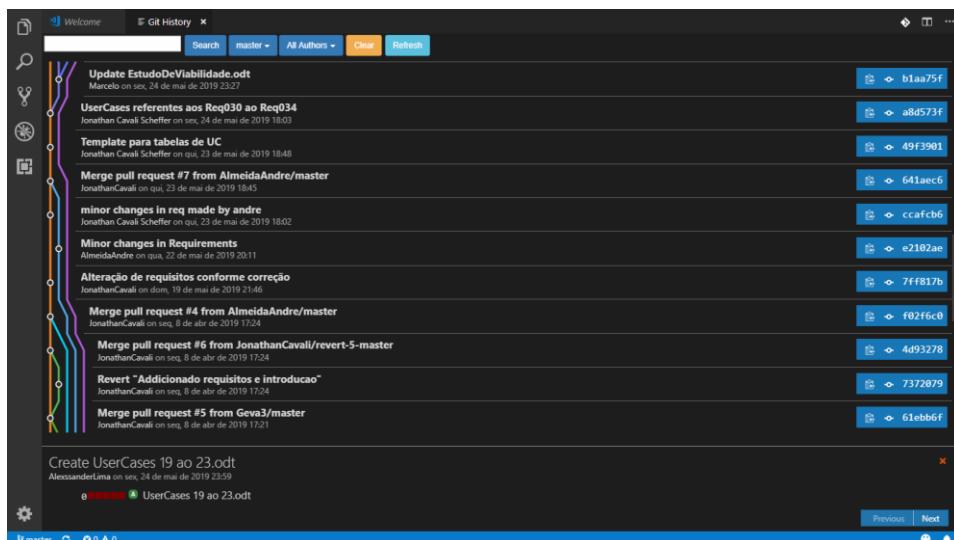


Figura 11 – Mapeamento de todos os commits

The screenshot shows the GitHub repository 'EngenhariaDeSoftware'. At the top, there are buttons for 'Branch: master', 'New pull request', 'Create new file', 'Upload files', 'Find File', and 'Clone or download'. Below this is a table of recent commits:

Commit	Description	Time Ago
Added Action Diagram	Added Action Diagram	18 hours ago
Create gerar grafico.PNG	Create gerar grafico.PNG	22 hours ago
Diagrams.pdf	Diagrams de atividade e sequencia casos 19 ao 23	a day ago
EstudoDeViabilidade.doc	Add files via upload	2 months ago
EstudoDeViabilidade.odt	Update EstudoDeViabilidade.odt	5 days ago
Orcamento_Scanner.ods	Criação de Alternativas 1&2	2 months ago
README.md	Add files via upload	2 months ago
UserCases Diagram.asta	Adding the user Classes Diagram	22 hours ago
UserCases 19 ao 23.odt	Correção de user cases	a day ago
UserCasesTables.odt	Changes on UserCases	19 hours ago

Below the commits is the 'README.md' file content:

```
EngenhariaDeSoftware

Diretório de versionamento para ser usado na disciplina de Engenharia de SW.

Arquivos:
• README.md: Overview do diretório.
• EstudoDeViabilidade.doc: Artefato que descreve o Estudo de Viabilidade do projeto.
```

Figura 12 – Repositório da equipe

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

Para acessar todos os artefatos e as versões basta acessar o link https://github.com/JonathanCavali/Engenharia_De_Software.git

Grandes empresas como a companhia automotiva Tesla, Google, Tweeter, Facebook, entre outras, possuem seus códigos fonte disponibilizados para a comunidade. Esse é um grande avanço que surgiu com a Open Source Initiative e traz muitos benefícios para o desenvolvimento ágil do projeto.

Logo:

O design da logo do SADVE (figura 13) foi criada em formato de selo postal com o intuito de passar uma primeira impressão ao usuário de que o produto é voltado para a comunicação, o carro ao centro indica que o produto é voltado para veículos elétricos e a pilha dentro dele mostrando que o produto é especificamente para automóveis elétricos. A parte verde do selo representa a visão da empresa sobre a nova era ecológica que estamos testemunhando atualmente e o veículo em azul representa a energia elétrica



Figura 13 – Logo SADVE

Instalador:

Junto com o protótipo desenvolvido, foi criado um instalador para facilitar a instalação do software pelo usuário do sistema. O instalador está disponível somente para o sistema operacional Windows. A seguir estão descritos os passos para a realizar a instalação do SADVE UI, junto com imagens que demonstram os passos.

Passo 1: Executar o instalador SADVEInstaller.msi. Ao abrir o instalador, aparecerá uma janela, como a da figura 14. Clique em “Avançar”, para continuar.

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

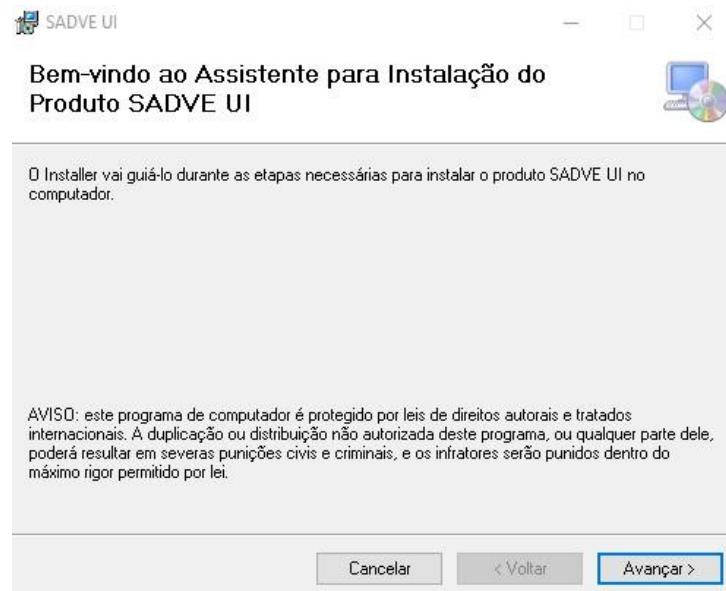


Figura 14 – Instalador

Passo 2: Aparecerá a opção para selecionar o diretório onde será instalado o SADVE UI, como na figura 15. Selecione o diretório desejado e clique em “Avançar”.

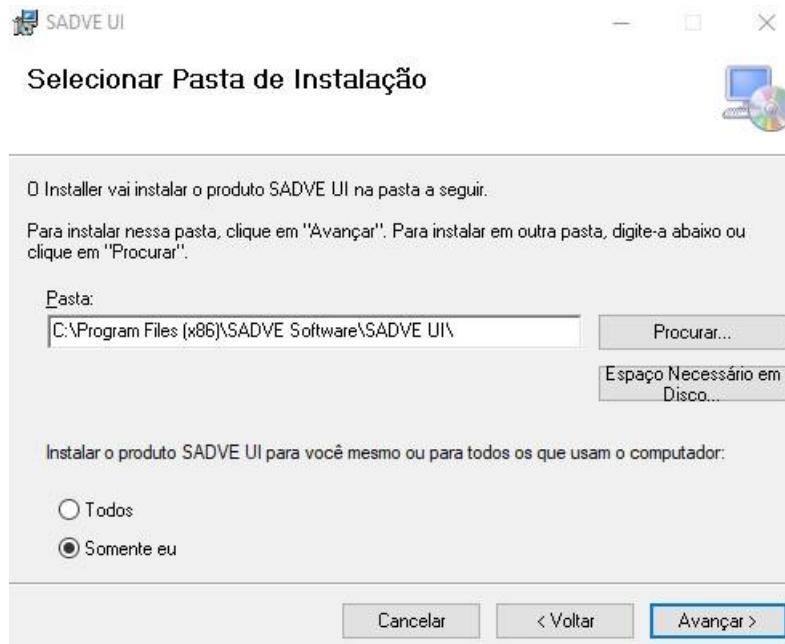


Figura 15 - Instalador

Passo 3: Será requisitado uma confirmação do usuário para finalizar a instalação, como na figura 16. Para confirmar clique em “Avançar”.

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

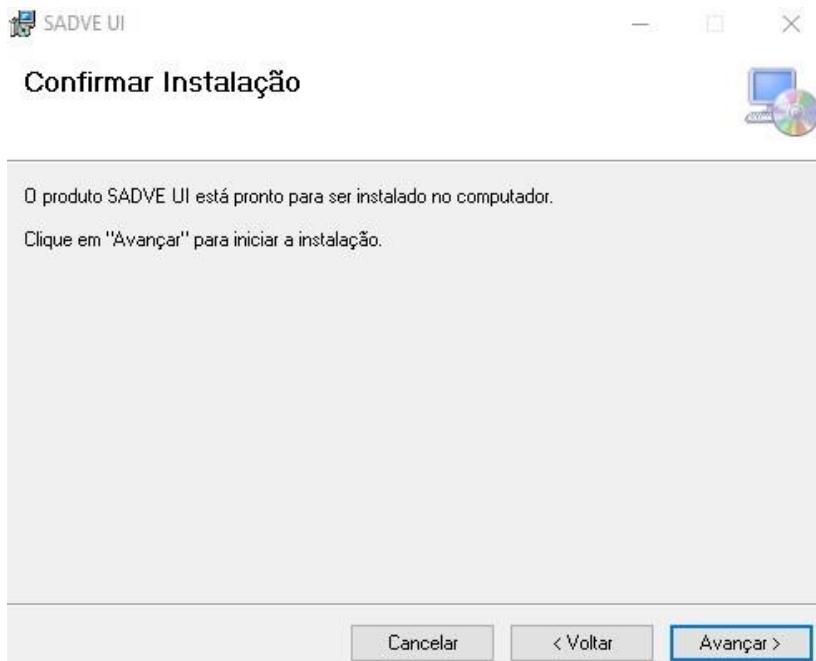


Figura 16 - Instalador

Passo 4: A última tela será o progresso da instalação do software, como na figura 17. Para finalizar a instalação, clique em “Avançar”. Após instalado, aparecerá um ícone de atalho na Área de Trabalho para executar o software.

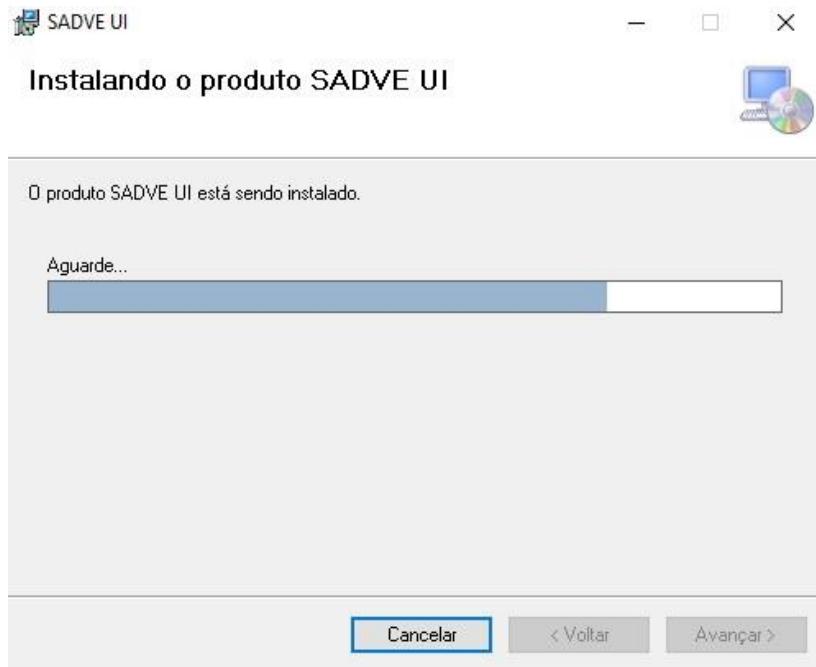


Figura 17 – Instalador

Sistema de Análise de Veículos Elétricos	Versão: 1.0
Documento de Arquitetura de Software	Data: 29/05/19

8 RESPONSABILIDADES

Documento assinado por:

Andre Luiz de Almeida Camargo

Alexssander Siqueira Lima

Guilherme Gustavo Gevard

Jonathan Cavali Scheffer

Marcelo Rudolf Junior

Na data de 29 de maio de 2019