****

**ESPECIFICAÇÃO técnica FUncional**

**track vision**

**Gestão e Finanças**

**Grupo 04**

Gabriel Cavalcanti

Julia Barbosa

Leandro Vieira

Michelly Mendes

Pedro Moretti

+55-114412-4412  
Trackvision@gmail.com

**ÍNDICE**

[1 Identificação 3](#_Toc119062097)

[2 Descrição Geral da Demanda 3](#_Toc119062098)

[3 Publico Alvo 3](#_Toc119062099)

[4 requisitos Funcionais 4](#_Toc119062100)

[5 Parecer Quanto a Viabilidade Técnica da Solução 6](#_Toc119062101)

[6 Descrição da Análise Técnica 6](#_Toc119062102)

[7 Arquitetura 6](#_Toc119062103)

[7.1 Diagrama 6](#_Toc119062104)

[7.2 Detalhamento 6](#_Toc119062105)

[8 (Modelo de Dados) 6](#_Toc119062106)

[8.1 Diagrama Relacional 6](#_Toc119062107)

[8.2 Detalhamento do Diagrama Relacional – Dicionario de Dados 6](#_Toc119062108)

[9 Interfaces de Integração 6](#_Toc119062109)

[10 Requisitos não Funcionais 8](#_Toc119062110)

[10.1 Detalhamentos requisitos não funcionais 8](#_Toc119062113)

[10.2 Considerações Quanto a Segurança 9](#_Toc119062114)

[10.3 Método de Autenticação 9](#_Toc119062115)

[10.4 Características de Plataforma – Requisitos não funcionais 9](#_Toc119062116)

[10.5 Considerações sobre Ambientes Necessários 9](#_Toc119062117)

[11 Referências 9](#_Toc119062118)

[12 Histórico de Modificações 9](#_Toc119062119)

[13 Anexos 10](#_Toc119062120)

[13.1 Modelo de Entidades 10](#_Toc119062121)

[12.1.1 Diagrama de Entidades 10](#_Toc119062135)

[12.1.2 Detalhamento do Diagrama de Entidades 10](#_Toc119062136)

[13.2 Diagrama de Caso de USO, Classes e/ou Diagrama de Seqüência 10](#_Toc119062137)

# 1 Identificação

|  |  |
| --- | --- |
| Sistema | TRACK VISION |
| Descrição da Solicitação | Monitoramento de Hardware |
| Número da Solicitação | ID 001 |
| Analista Responsável | Grupo 04 |
| Versão | 1.08.9 |

# 

# 2 Descrição Geral da Demanda

Monitoramento de processamento de uso de hardware

# 3 Público Alvo

|  |  |
| --- | --- |
| **Perfil** | **Localização e Quantidade** |
| Gerente de TI | 200 personas, setor Governança |
| Técnico de TI | 500 personas, setor Técnico |

# 4 Requisitos Funcionais

**Requisito:** RF-01 - O sistema deve gerar gráficos com os dados captados das APIs para que *o técnico de TI* consiga *monitorar o sistema*

**Regras de negócio:** *O sistema deverá gerar gráficos em tempo real, todos os dias com o apoio de uma equipe técnica, para maior satisfação do cliente.*

**Requisito:** RF-02 - O sistema deve emitir alertas para que uma equipe técnica possa *prevenir a falha do sistema, invasão e falha no hardware* para *poupar gastos*

**Regras de negócio:** *Através de alertas configuradas no código sobre os componentes do hardware, o cliente poderá prever falhas e invasões em seu sistema, evitando gastos para reparação e ter seus sistema fora de serviço*

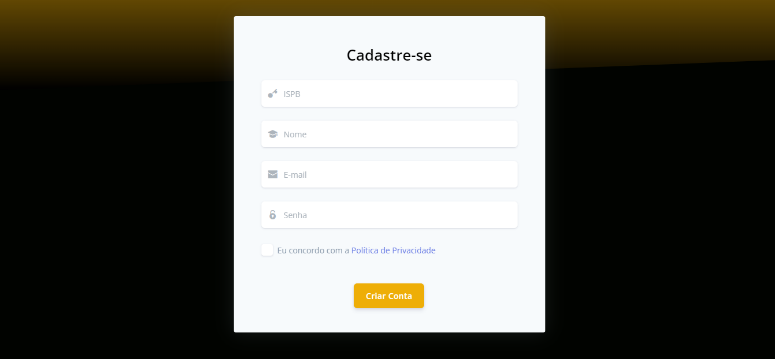
**Requisito:** RF -03 - O sistema *deverá enviar alertas via Slack quando um chamado for aberto por falhas no hardware* *para que a equipe técnica* possa *reparar danos* para *o caixa continuar serviço*

**Regras de negócio:** *O sistema deverá ter a opção de abertura de chamados da parte do cliente para que ele informe a falha e consiga o recovery o quanto antes*

**Requisito:** RF -04 - O sistema deverá emitir avisos caso o cadastro ou login falhe para o cliente que estiver sendo logado ou se cadastrando possa ter uma visão do que está fazendo errado *para auto-intuição e conseguir entrar no site*

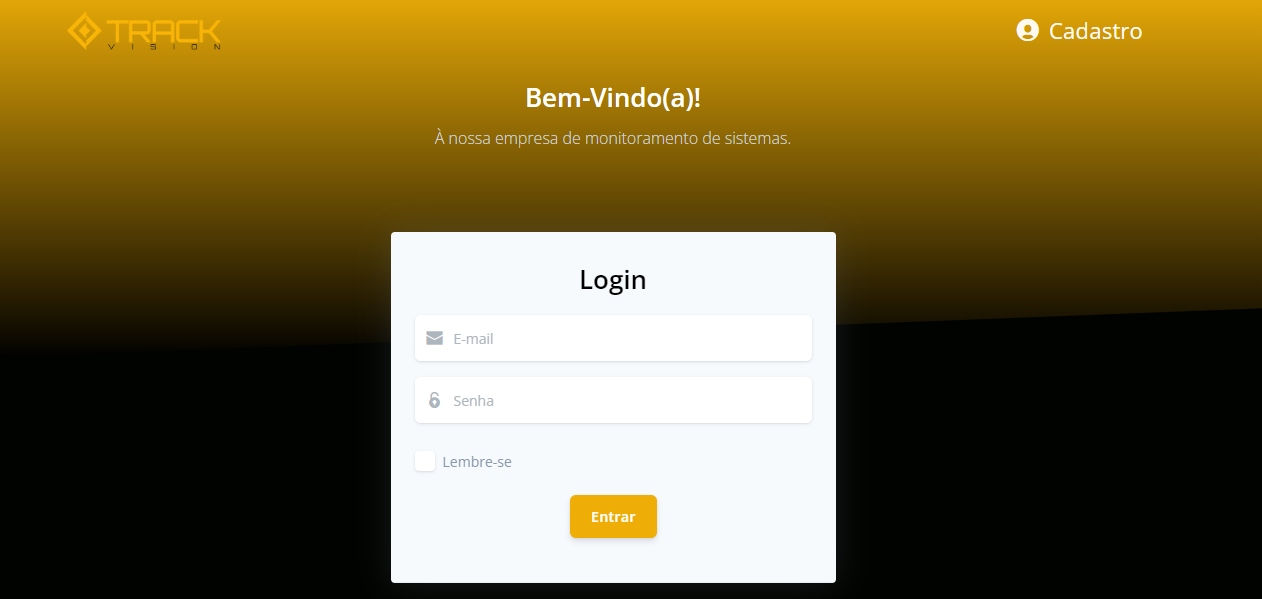
**Regras de negócio:** *O sistema terá uma série de avisos para a parte de cadastro e login, pegando as informações inseridas e verificando no banco de dados para conferir se na hora do cadastro tem e-mail duplicado, campos vazios ou confirmação de senha errada, ou também se na hora do login haja campos vazios e informações erradas como e-mail e senha*

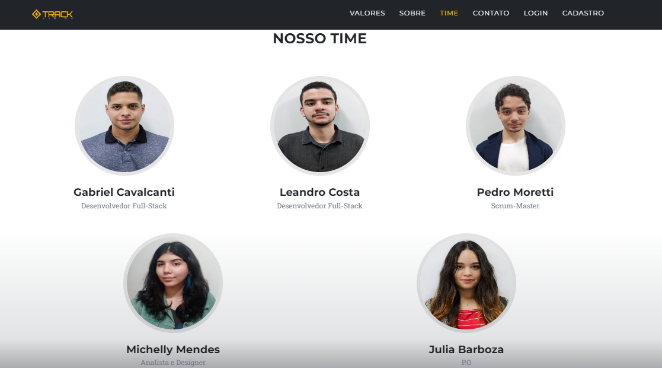
**Telas:**











Dependências: *RF01, RNF02, RF03, RNF 05*

# 5 Parecer Quanto a Viabilidade Técnica da Solução



# 6 Descrição da Análise Técnica

Os caixas eletrônicos têm um fluxo de uso em horário comercial bastante elevado, com picos de uso de bancos ao decorrer do dia, o que demanda mais processamento e transições de dados que o sistema operacional da ATM deve exercer. Para tanto, nosso projeto visa estabelecer a otimização deste hardware, buscando alternativas para que o equipamento possa se manter efetivo na maior parte do tempo de uso, evitando quedas do sistema (manutenção) e diminuir o fluxo de espera nos bancos.

- Diminuir fluxo de espera;

- Atuação imediata do suporte;

- Uptime do sistema em cenário de altas solicitações.

Temos por objetivo otimizar e minimizar o fluxo de espera para utilização de caixas eletrônicos, tendo em vista sua disponibilidade e necessidade de atualização de hardware e/ou software utilizado, o que impacta na utilização dos serviços e altas solicitações em horário comercial.

O desenvolvimento conta com uma equipe técnica para a implementação do projeto, dividindo em planejamento, produção e entrega, sempre acompanhando a evolução junto aos clientes.

# 7 Arquitetura

## 7.1 Diagrama

## 7.2 ALTERAÇÃO DIAGRAMA

|  |  |
| --- | --- |
| **Componente** | **Descrição** |
| Agência | Local de hospedagem dos caixas eletrônicos a serem monitorados |
| Caixas eletrônicos | Caixas a serem implementados uma API para monitoramento de hardware |
| Disco | Componente de Hardware |
| CPU | Componente de Hardware |
| RAM | Componente de Hardware |
| Looca | Biblioteca de Kotlin usada para obtenção de dados |
| Kotlin | Linguagem usada no desenvolvimento do projeto, dependente de java |
| Slack | Ferramenta de help desk integrada para abertura de chamados em nosso site |
| AWS | Amazon Web Services, plataforma de serviços de computação em nuvem |
| EC2 | Elastic Compute Cloud, plataforma de cloud computing da Amazon que permite alugueis de computadores virtuais para aplicações. |
| Python | Linguagem usada no desenvolvimento do projeto |
| Ubuntu | Dependência linux |
| Linux | Sistema Operacional |
| MySQL | Banco de Dados |
| Azure | Banco de Dados |
| Node JS | Linguagem usada no desenvolvimento do projeto |
| CSS | Linguagem de Front end usada no desenvolvimento do projeto |
| Java Script | Linguagem usada no desenvolvimento do projeto |
| Navegadores | Navegadores mais comuns na internet |

# 8 (Modelo de Dados)

## 8.1 Diagrama Relacional

## 8.2 Detalhamento do Diagrama Relacional – Dicionario de Dados

|  |  |
| --- | --- |
| **Objeto** | **Descrição** |
| tempCpu | Campo adicionado para obtenção de Temperatura da CPU |
| Processos ativos | Campo adicionado para obtenção lista de processos ativos |
| servicosAtivos | Campo adicionado para obtenção lista de serviços ativos |
| momento | Campo adicionado para obtenção data e hora |
| processosPerigosos | Campo adicionado para obtenção lista de processos de risco |
| servicosPerigosos | Campo adicionado para obtenção lista de serviços de risco |
| Projeto\_Julia | Tabela de projeto individual |

# 9 Interfaces de Integração

|  |  |  |  |
| --- | --- | --- | --- |
| **Integração** | **Mecanismo** | **Método** | **Classificação (Consumidor/Provedor)** |
|  |  |  |  |
|  |  |  |  |

# 

# 10 Requisitos não Funcionais

**10.1 DETALHAMENTOS REQUISITOS NÃO FUNCIONAIS**

# RNF 07 – O sistema deve rodar com banco de dados mysql, azure e aws

# Regra de negócio: Deve ser implementado os bancos de dados solicitados

# RNF 12 – O sistema deve ser desenvolvido em python

# Regra de negócio: Deve ser implementado API’S em linguagem python

# RNF 04 – O sistema deve ter uma autenticação, ligado com banco de dados e com alertas

# Regra de negócio: Deve ser implementado um método de autenticação por meio de validação dentro do site

# RNF 09 – O sistema operacional deve ser linux ou windows

# Regra de negócio: O Projeto deverá contar apenas com os sistemas operacionais linux ou windows

# RNF 18 – O sistema deve ser seguro contra ataques/invasões

# Regra de negócio: O Projeto deve contar com uma empresa ou equipe parceira que cuidará da segurança de dados de nossa rede

## 10.2 Considerações Quanto a Segurança

O Sistema terá implementado um método de validação por meio do back end com o banco de dados, para conferir se o usuário é cadastrado, campos errados e duplos emails

## 10.3 Método de Autenticação

Autenticação através de Login, Cadastro, com alertas caso não bata as informações com o banco (dados incorretos), caso já haja um usuário com o mesmo e-mail na parte do cadastro(cadastro duplicado, e-mails já existente), alertas caso tenha um campo não preenchido.

# 12 Anexos



## 12.1 Diagrama UML de Caso de USO, Classes e/ou Diagrama de Sequência

