

HelpDesk

Christian de Jesus Gutierrez, Diogo Costa, Giovani Heineck Chiele

Instituto Federal do Paraná – IFPR

Campus Foz do Iguaçu – Foz do Iguaçu – PR – Brasil

gutichristian1@gmail.com, diogoferreira@gmail.com,

chgiovani5@gmail.com

Resumo. *Esse trabalho aborda a criação e desenvolvimento de um Aplicativo para Dispositivos Mobile com Sistema Operacional Android cujo objetivo é a possibilidade de criação de chamados de TI e resolução dos mesmos. O Aplicativo foi construído utilizando Dart com o Framework Flutter. Para construção do Back-End foi utilizado Java e como Framework foi escolhido o Spring Boot. Por fim, o Sistema Gerenciador de Banco de Dados escolhido foi o MySQL. Utilizando padrão MVC e Paradigma de Orientação a Objetos.*

Abstract. *This project deals with the creation and development of an Application for Mobile Devices that use Android as an Operational System. Its objective is the possibility of creating TI Orders and their resolution. The App was constructed using Dart with the Flutter Framework. Java and Spring Boot were chosen for the creation of the Back-End. The Database Sistem Manager chosen was MySQL and the project was built using MVC pattern and Object Oriented Paradigm.*

1. Justificativa

Muitas microempresas e empresas de pequeno porte ainda enfrentam dificuldade em comunicação em casos de problemas com dispositivos de informática e sistemas. O projeto HelpDesk é um aplicativo de gerenciamento de chamados de TI que será feito para mitigar estas adversidades. Já existem sistemas/aplicativos no mercado que cumprem sua função com excelência, mas em sua maioria possuem diversas funcionalidades que nem sempre são utilizadas, possuem layouts confusos/complexos e custam caro para empresas menores.

Com este aplicativo, será possível reduzir o ruído de comunicação entre cliente e técnicos de informática, garantindo com que o tempo de resposta de um chamado seja reduzido custos e o problema seja solucionado o mais rápido possível, assim otimizando o tempo e deixando o ambiente de trabalho o mais produtivo possível.

2. Objetivos do Projeto

Desenvolvimento de um aplicativo de gerenciamento de chamados de problemas de TI, com uma interface amigável e de simples utilização. Haverá um sistema de distinções de funcionalidades do aplicativo de acordo com o tipo de usuário, os usuários do tipo “Cliente” realizaram a abertura do chamado de TI, inserindo as informações a respeito do problema, o local e podendo inserir uma foto do problema seja diretamente da câmera de seu celular bem como da galeria de fotos, também poderá acompanhar como está o andamento dos chamados que o mesmo abriu e caso já tenha sido resolvido podendo visualizar qual foi a resolução que o técnico apresentou ao seu chamado.

Já os usuários do tipo “Técnico” visualizaram os chamados criados por todos os Clientes na aba “Criados” o técnico poderá visualizar a descrição do chamado e podendo iniciar o atendimento, desta forma o chamado será movido para a aba “Em progresso”, e após o técnico solucionar o problema o mesmo inserirá no chamado que está “Em progresso” a resolução, ao resolver o chamado será movido para a aba “Resolvidos”. O aplicativo será disponibilizado até o fim de julho de 2021.

3. Escopo

O resultado desejável no fim do projeto é a entrega do aplicativo consumindo os dados de um sistema e executando todas as funções esperadas. Além destes, será entregue uma documentação completa contendo diagramas, requisitos funcionais e regras de negócio do aplicativo.

4. Não-Escopo

Não conta com um gerenciamento de ativos de hardware (ex: computadores, impressoras, tablets...); Não conta com gerenciamento de ativos de software; Não conta com um gerenciamento de níveis de prioridade para o atendimento do chamado, não possuindo um controle de SLA (Service Level Agreement) ou (Acordo de Nível de Serviço); Não conta com log de serviço nos chamados, único campo onde o usuário técnico poderá descrever o que foi realizado para a resolução será no campo “Resolução”.

5. Restrições

Devido a pandemia de Covid 19 e suas medidas de prevenção não será possível ir ao IFPR, limitando que o desenvolvimento seja realizado com hardware próprio que pode não ter um desempenho satisfatório; O não aprendizado a respeito das ferramentas que serão utilizadas no desenvolvimento implicam na não entrega do aplicativo.

6. Projetos Inter-Relacionados

Para produzir e gerir dados para o aplicativo que será desenvolvido utilizando o Framework Flutter que utiliza a linguagem Dart, será construído uma API REST utilizando a linguagem Java com Spring Boot.

7. Riscos Iniciais

Dúvidas e problemas com as ferramentas para desenvolvimento do aplicativo; Eventuais mudanças no planejamento do software.

8. Tempo Estimado

O prazo estimado para a realização e entrega do aplicativo será até o fim de julho de 2021.

9. Requisitos Funcionais

Tabela 1. Tabela de Requisitos Funcionais

Código	Requisito Funcional
RF 01	O aplicativo deve permitir com que o usuário se cadastre e gerencie com os seguintes dados: Nome, sobrenome, CPF, telefone, foto, e-mail, senha e tipo de usuário(podendo escolher entre ser cliente ou técnico).
RF 02	O aplicativo deve permitir com que os usuários possam fazer login utilizando e-mail e senha.
RF 03	O aplicativo deve permitir com que os usuários possam recuperar sua senha.
RF 04	O aplicativo deverá permitir que o usuário tire uma foto ou selecione uma da galeria do dispositivo, na abertura de um chamado.
RF 05	O aplicativo deve permitir com que o usuário do tipo cliente possa criar e gerenciar chamados criados por ele utilizando os seguintes dados: Identificação, título, descrição, imagem e local.
RF 06	O aplicativo deve permitir com que usuários do tipo técnico possam escolher um chamado a ser iniciado.
RF 07	O aplicativo deve permitir com que usuários do tipo técnico possam no término do chamado especificar no campo solução a resolução do problema.
RF 08	O aplicativo deverá permitir com que usuários do tipo cliente e técnico possam pesquisar por chamados por situação ou palavras contidas no título.
RF 09	O aplicativo deve ordenar a listagem dos chamados na ordem de criação do mais antigo para o mais novo nos status “Criado” e “Em progresso”.
RF 10	O aplicativo deve permitir que o campo imagem no chamado seja facultativo para preenchimento.
RF 11	O aplicativo deve permitir que o técnico só possa visualizar os detalhes de um chamado por vez.

RF 12	O aplicativo deve permitir que os chamados deverão poder estar nas seguintes situações: (“Criado”, “Em progresso” e “Resolvido”).
--------------	---

10. Requisitos Não-Funcionais

Tabela 2. Tabela de Requisitos Não-Funcionais

Código	Requisito Não-Funcional
RNF 01	O back-end deve criptografar a senha no banco de dados utilizando criptografia SHA-1.
RNF 02	O back-end deve utilizar o sistema de banco de dados MySQL.
RNF 03	O back-end deve utilizar o protocolo REST para comunicação entre servidor e cliente utilizando JSON.
RNF 04	O back-end deve ser feito em Java com Spring Boot.
RNF 05	O Aplicativo deve ser feito em Dart utilizando o Flutter Framework.

11. Regras de Negócio

Tabela 3. Tabela de Regras de Negócio

Código	Regra de Negócio
RN 01	Campos CPF e E-mail serão únicos para cada usuário.
RN 02	Tipos de usuário, local e situação deverão ser previamente cadastrados no sistema.
RN 03	Um chamado só poderá ter o campo “situação” alterado na seguinte ordem: “Criado” -> “Em progresso” -> “Resolvido”.
RN 04	Tipos de usuário, local e situação deverão ser previamente cadastrados no sistema.
RN 05	Quando um chamado for criado a data de criação do chamado será salva.
RN 06	Quando um chamado for iniciado o valor do campo “situação” mudará para “Em progresso”.
RN 07	Quando um chamado for resolvido o valor do campo “situação” mudará para “Resolvido”.

RN 08	Quando um chamado for resolvido a data de resolução do chamado será salva.
RN 09	Cada usuário do tipo “Técnico” pode ter no máximo 03(três) chamados com a situação “Em progresso” simultaneamente em sua propriedade.

12. Telas

The image displays two side-by-side screenshots of a mobile application's user registration screen, titled "Cadastro".

Left Screenshot (11:43): This view shows the initial registration form. At the top, there is a large grey placeholder for a profile picture. Below it are two blue buttons: "Camera" and "Galeria". The form contains four input fields: "Nome", "Sobrenome", "CPF", and "Telefone".

Right Screenshot (16:38): This view shows the registration form with an additional field. It includes the same fields as the left screenshot, plus an "E-mail" field. Below the "E-mail" field is a toggle switch labeled "Cliente" (which is currently selected) and "Técnico". At the bottom of the form is a large blue button labeled "Cadastrar".

Figura 1. Telas de Cadastro de Usuário

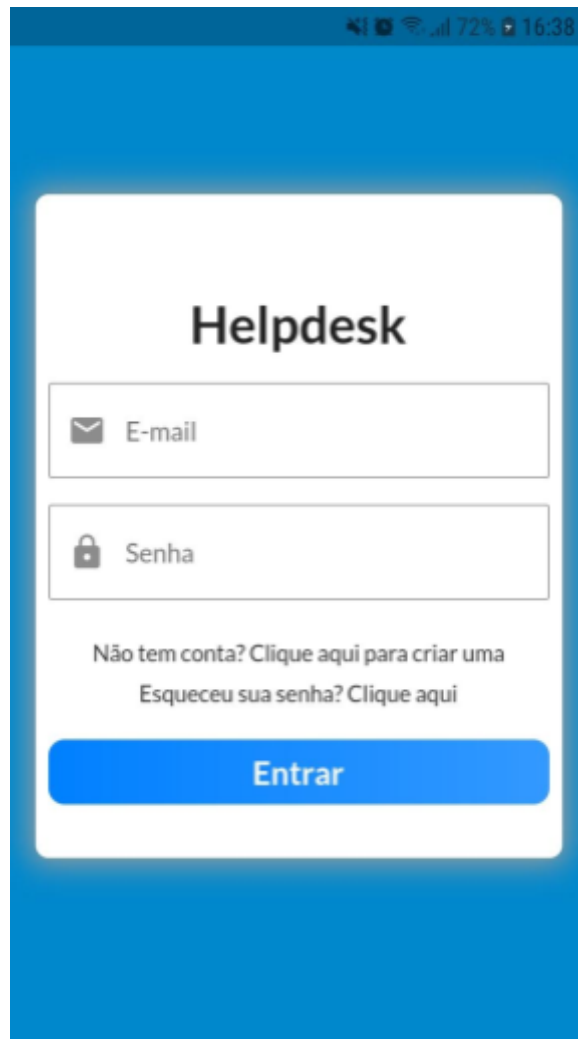


Figura 2. Tela de Login



The image shows a mobile application interface for password recovery. At the top, there is a blue header bar with a white back arrow icon on the left and the text "Esqueci minha senha" in white. Below the header, the main background is a solid blue color. In the center, there is a white rounded rectangle containing the following elements: a title "Insira CPF e Email para a geração de uma nova senha" in black; a text input field with a light gray border, a small ID card icon on the left, and the label "CPF" in gray; another text input field with a light gray border, a small envelope icon on the left, and the label "E-mail" in gray; and a blue rounded button with the white text "Recuperar Senha". The top status bar of the phone is visible at the very top, showing icons for signal, battery, and time (16:45).

Esqueci minha senha

Insira CPF e Email para a geração de uma nova senha

CPF

E-mail

Recuperar Senha

Figura 3. Tela de Esqueci minha senha

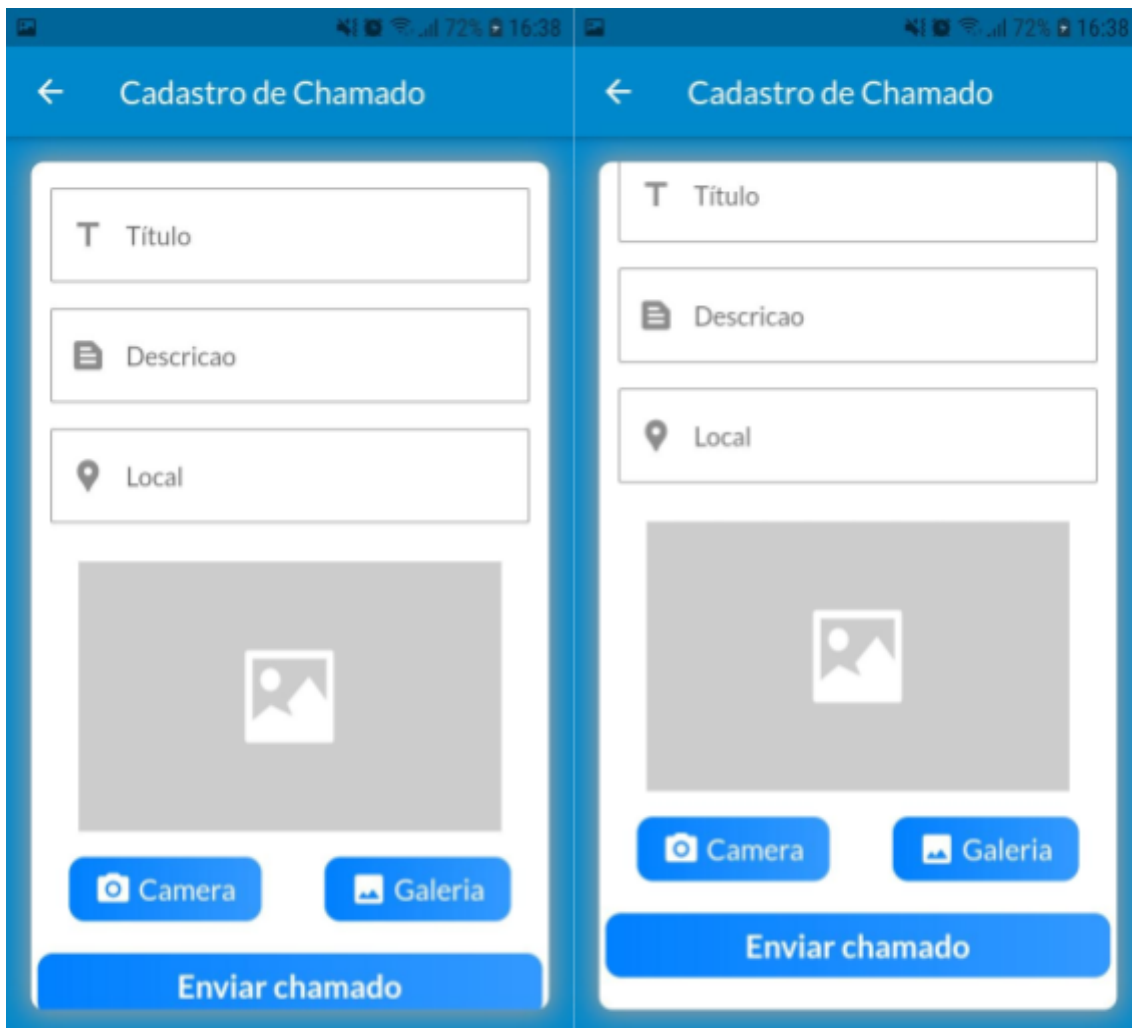


Figura 4. Telas de Cadastro de Chamado

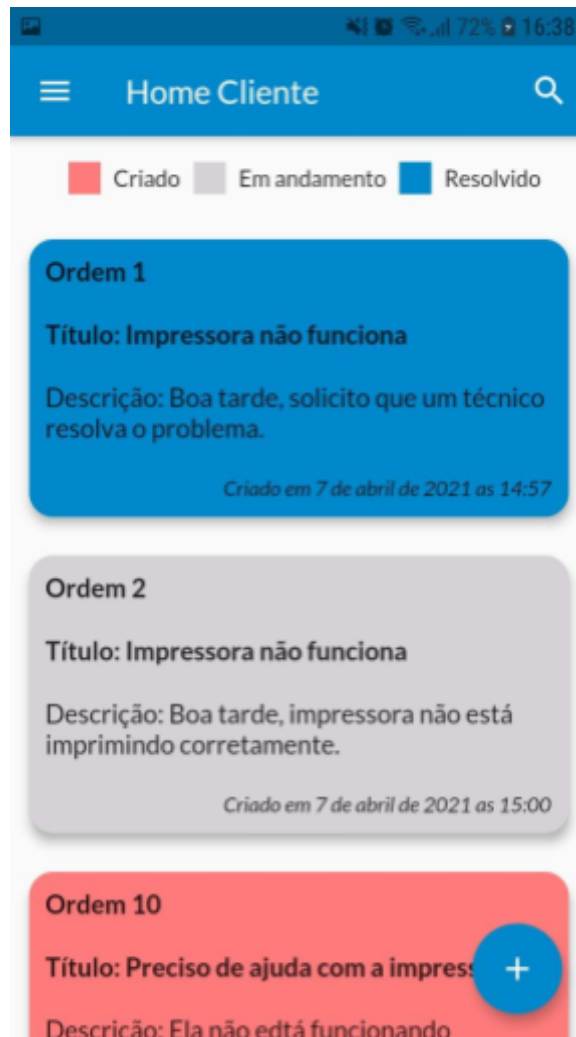


Figura 5. Tela Inicial



Figura 6. Telas de Detalhe do Chamado

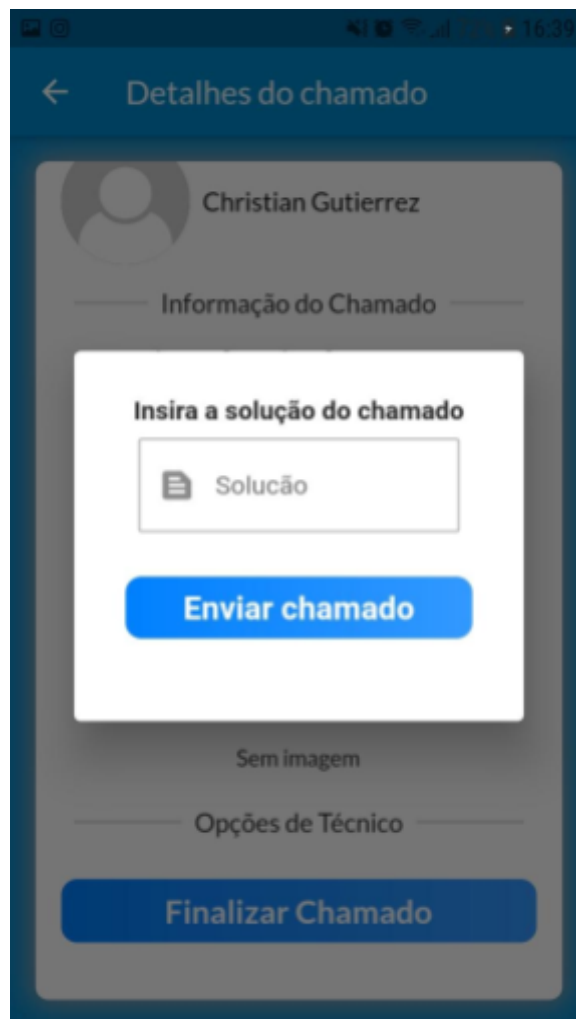


Figura 7. Tela de Inserção de Solução

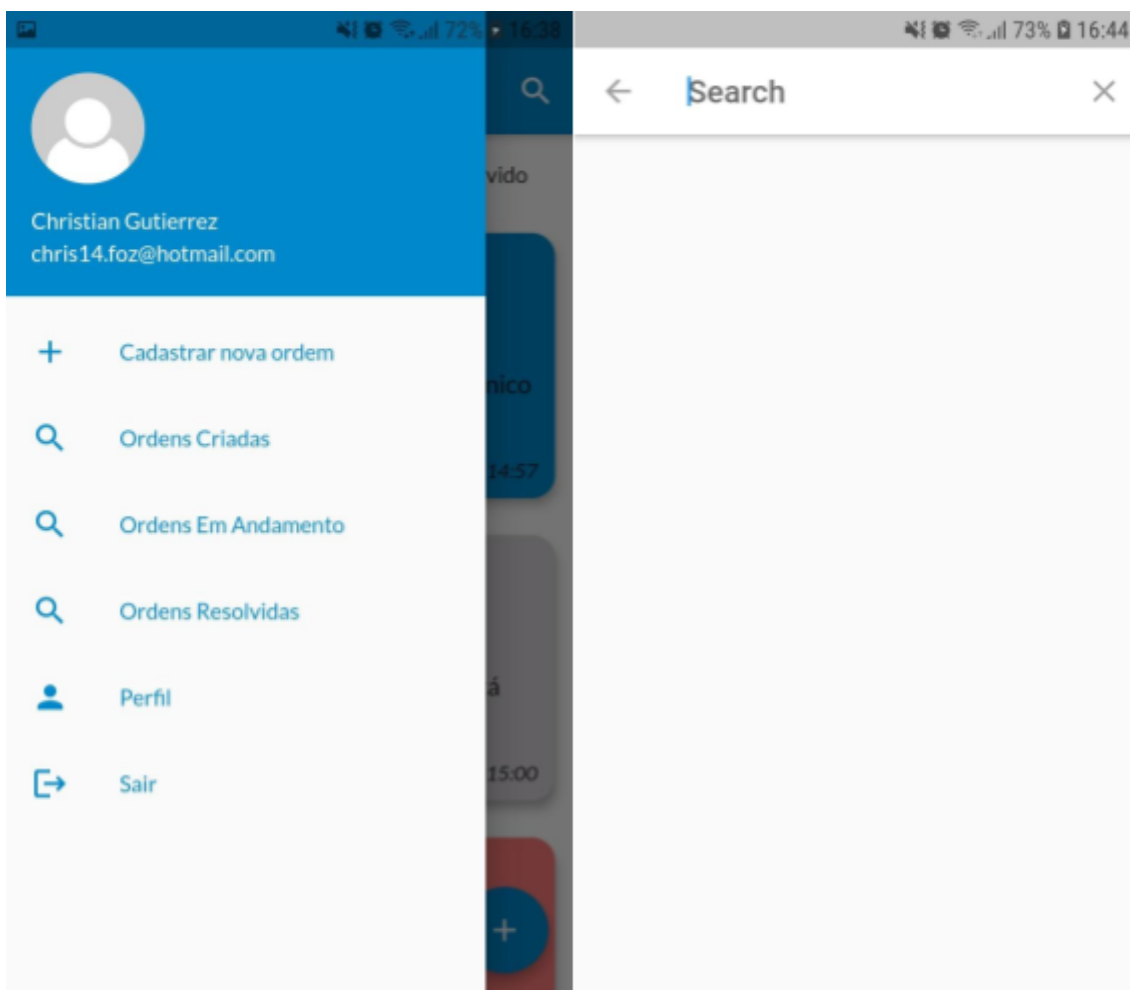


Figura 8. Telas de Pesquisa por Chamados