



**ALINE DOS SANTOS SCHMIDT**  
**ROBSON DE SOUZA FIGUEIREDO**

**DESENVOLVIMENTO DE APLICATIVO *HELP DESK* NA EMPRESA EVOLUÇÃO  
SISTEMAS.**

Ji-Paraná  
2021

**ALINE DOS SANTOS SCHMIDT  
ROBSON DE SOUZA FIGUEIREDO**

**DESENVOLVIMENTO DE APLICATIVO *HELP DESK* NA EMPRESA EVOLUÇÃO  
SISTEMAS.**

Monografia apresentada à disciplina de trabalho de conclusão de Curso, do 8º período do Curso de Sistemas de Informação, do Grupo Educacional São Lucas Ji-Paraná, como parte dos requisitos para obtenção do título de Bacharel.

Orientador: Prof. Willian Alves de Oliveira Fachetti

Ji-Paraná  
2021

**Dados Internacionais de Catalogação na Publicação - CIP**

S349d Schmidt, Aline dos Santos.

Desenvolvimento de aplicativo Help Desk na empresa Evolução Sistemas. / Aline dos Santos Schmidt ; Robson de Souza Figueiredo. – Ji-Paraná, 2021.  
37 p., il.

Monografia (Curso de Sistemas de Informação) – Centro Universitário São Lucas Ji-Paraná, 2021.

Orientador: Prof. Willian Alves de Oliveira Fachetti

1. Desenvolvimento. 2. Aplicativo Help Desk. 3. Metodologia RAD. 4. Serviço de atendimento ao cliente. I. Figueiredo, Robson de Souza. II. Fachetti, Willian Alves de Oliveira. III. Título.

CDU 004:658

**ALINE DOS SANTOS SCHMIDT  
ROBSON DE SOUZA FIGUEIREDO**

**DESENVOLVIMENTO DE APLICATIVO *HELP DESK* NA EMPRESA EVOLUÇÃO  
SISTEMAS.**

Monografia apresentada à Banca Examinadora do Centro Universitário São Lucas Ji-Paraná, como requisito de aprovação para obtenção do Título de Bacharel em Sistemas de Informação.

Orientador Prof. Willian Alves de Oliveira Fachetti

Ji-Paraná, 23 de 06 de 2021

Avaliação/Nota:

BANCA EXAMINADORA

---

Prof. Me. Maigon Nacib Pontchuska

São Lucas Ji-Paraná

---

Prof. Esp. Hailton Alves dos Reis

São Lucas Ji-Paraná

---

Prof. Me. Ana Flavia Moreira Camargo

São Lucas Ji-Paraná

“Comece fazendo o que é necessário, depois o que é possível, e de repente você estará fazendo o impossível.”

(São Francisco de Assis)

## RESUMO

Percebendo a dificuldade de alguns clientes da empresa Evolução Sistemas para comunicar-se com o suporte e tirar suas dúvidas e ainda considerando obter uma opção em que técnico e cliente consigam ver protocolos já fechados anteriormente, foi analisada a necessidade de desenvolver uma aplicação de atendimento *Help Desk* com o intuito de alcançar uma comunicação com todos os usuários de forma simples e por fim integrar ao sistema ERP *eCenterManager* já existente da empresa. A proposta dessa aplicação foi baseada na metodologia RAD (Desenvolvimento Rápido de Aplicação) utilizando como linguagem de programação o *Delphi* e os bancos de dados *PostgreSQL* e *FirebirdSQL*, este último apenas para fazer a identificação do cliente, pois o sistema comercial usado por ele é esse. Na fase de testes, ocorreram bons resultados tendo apenas contratempos quanto aos antivírus instalados nos computadores dos clientes, porém foram cumpridos todos os requisitos apresentados. A aplicação facilitará para os usuários que poderão tirar as eventuais dúvidas sobre os sistemas comerciais, evitando perda de tempo e possíveis contratempos, proporcionando qualidade no serviço prestado pela empresa Evolução e conseqüentemente um melhor uso do cliente no sistema comercial.

**Palavras-chave:** Desenvolvimento. Aplicativo *Help Desk*. Metodologia RAD.

## **ABSTRACT**

Realizing the difficulty of some customers of the company Evolução Sistemas to communication with the support for answer their doubts and still obtain an option in which technician and customer can see protocols already closed previously, was analyzed the need to develop a service application with in order to achieve communication with all users in a simple way and finally integrate into the company existing to ecenterManeger ERP system. The proposal for this application was based on the RAD (Rapid Application Development) methodology using Delphi for programming language and the PostgreSQL and FirebirdSQL for databases, the latter only for identifying the client, as the commercial system used by him is this FirebirdSQL. In the testing phase, there were good results with only setbacks regarding antivirus on the installed computers of the customers, but all necessary requirements were met. The application facilitated for users who take the doubts about the system utilizing, avoiding the loss of time and possible setbacks, providing quality in the service provided by the company Evolução and consequently a better use of the customer in the commercial system.

**Keywords:** Development. Application Help Desk. Methodology RAD

## LISTA DE ILUSTRAÇÕES

Figura 1 - Diagrama de caso de uso.....	21
Figura 2 - Diagrama de classes.....	22
Figura 3 - Diagrama de atividades.....	23
Figura 4 - Diagrama de entidade e relacionamento.....	26
Figura 5 - meios do cliente acessar o aplicativo.....	27
Figura 6 - Tela inicial.....	28
Figura 7 - Protocolo em andamento.....	29
Figura 8 - Tela inicial do ERP EcenterManeger.....	30
Figura 9 - Tela inicial do chat.....	30
Figura 10 - exemplo de notificação.....	31
Figura 11 - exemplo filtros de atendimento.....	31
Figura 12 - chats aberto, fechado e em andamento.....	32
Figura 13 - Guia aberta de um protocolo.....	33

## LISTA DE QUADROS

Quadro 1 - Requisitos Funcionais.....	19
Quadro 2 - Requisitos Não Funcionais.....	20
Quadro 3 - Dicionário de dados.....	24

## LISTA DE ABREVIATURAS E SIGLAS

ERP	<i>Enterprise Resource Planning</i> (Sistema Integrado de Gestão Empresarial)
RAD	<i>Rapid Application Development</i> (Desenvolvimento Rápido de Aplicações)
TI	Tecnologia da Informação
RF	Requisitos Funcionais
RNF	Requisitos Não Funcionais
NFe	Nota Fiscal Eletrônica
NFCe	Nota Fiscal do Consumidor Eletrônica
GUI	<i>Graphical User Interface</i> (Interface Gráfica do Utilizador)
SGBDR	Sistema de Gerenciamento de Banco de Dados Relacional
SGBD	Sistema de Gerenciamento de Banco de Dados
SQL	<i>Structured Query Language</i> (Linguagem de Consulta Estruturada)
IBM	<i>International Business Machines Corporation</i>

## SUMÁRIO

1. INTRODUÇÃO	10
1.1 JUSTIFICATIVA	10
1.2 OBJETIVO GERAL	11
1.2.1 Objetivos Específicos	11
2. REFERENCIAL TEÓRICO	12
2.1 HELP DESK	12
2.2 METODOLOGIA RAD	12
2.2.1 Fases da Metodologia RAD	13
2.3 FERRAMENTAS	14
2.3.1 Software Evolução Sistemas	14
2.3.2 Linguagem de programação Delphi	14
2.3.3 Banco de dados FirebirdSQL E PostgreSQL	15
3. MATERIAIS E MÉTODOS	16
3.1 MÉTODOS	16
3.1.1 Objeto de estudo	17
3.1.2 Metodologia de desenvolvimento	17
4. DESENVOLVIMENTO	18
4.1 DESCRIÇÃO DO SISTEMA	18
4.2 MODELAGEM DO SISTEMA	19
4.3 MODELAGEM DO BANCO	24
4.4 RESULTADOS	27
4.4.1 Sistema do lado do cliente	27
4.4.2 Sistema do lado do técnico	29
4.5 DISCUSSÕES	33
5. CONCLUSÃO	34
REFERÊNCIAS	36

## 1. INTRODUÇÃO

Frequentemente as empresas tendem a buscar uma melhoria contínua na qualidade do atendimento ao cliente, pois este é um fator fundamental para o sucesso das organizações. Porém, a satisfação do consumidor não depende apenas de um fator em específico, mas na combinação da qualidade do produto e na qualidade do atendimento prestado (FARIA, 2017). Diante disso as empresas além de buscar obter colaboradores capacitados e treinados buscam softwares que facilitam e auxiliam na qualidade do relacionamento entre colaborador e consumidor.

A Evolução Sistemas é uma empresa de desenvolvimento de softwares comerciais, no qual atende diversos segmentos. Atualmente possui um aplicativo de atendimento *Help Desk* terceirizado e o considera muito pesado, que exige grande banda de internet ao cliente contratante, tomando muito o tempo em atendimentos simples ou ainda correndo o risco de não ser atendido, por não conseguir abrir o aplicativo. “Uma internet de má qualidade, além de deixar a prestação desse serviço mais vagarosa, aumenta o tempo de solução do problema” (ROSARIO; PEREIRA; SILVEIRA, 2017, p.17).

Observa-se também que os colaboradores possuem muitas ferramentas para utilizarem durante o dia de trabalho que acarreta em muitos cliques impactando em sua produtividade.

Dessa forma, pensando na produtividade dos colaboradores e visto que a empresa possui muitos clientes que carecem de uma boa internet, como fazer para que a comunicação entre a central Evolução Sistemas e seus clientes se torne simples e ainda possuir um prático gerenciamento de protocolos acessível para ambas as partes?

### 1.1 JUSTIFICATIVA

Devido a empresa Evolução Sistemas possuir muitos clientes que se situam em distritos e cidades pequenas que carecem de uma conexão de internet estável e até mesmo alguns distritos com carência de linha telefônica, tornando-se difícil obter uma boa comunicação, faz-se necessário um aplicativo de comunicação que se torne possível o contato do cliente com a central de atendimento. "No que diz respeito às

empresas, os sistemas *Help Desk* atuam como um elo entre as mesmas e os clientes"(LINKE; SILVEIRA, 2015, p.5).

Dessa forma a empresa terá um próprio aplicativo de comunicação e como possui um ERP para gerenciamento de atendimentos, será integrado a esse ERP, diminuindo assim a necessidade dos colaboradores de terem vários aplicativos em aberto. A unificação desta ferramenta ao sistema interno da empresa facilitará o trabalho dos colaboradores, conseqüentemente aumentando sua produtividade. Para o cliente a aplicação se iniciará com o sistema operacional *Windows*, facilitando quando precisar abrir um protocolo.

Com a finalidade de aprimorar a experiência do cliente, pretende-se desenvolver a funcionalidade de visualização de histórico de protocolos, com o objetivo de supervisionar ou apenas consultar dúvidas anteriores já respondidas.

## 1.2 OBJETIVO GERAL

Desenvolver um aplicativo de gerenciamento de atendimento *Help Desk*, unificando o ERP *EcenterManeger* na empresa Evolução Sistemas.

### 1.2.1 Objetivos Específicos

Desenvolver um aplicativo *desktop*.

Unificar ao aplicativo ERP *EcenterManeger*.

Armazenar o histórico de conversas.

Utilizar a metodologia RAD.

## 2. REFERENCIAL TEÓRICO

### 2.1 HELP DESK

Em empresas de TI, são usados os sistemas *Help Desk*, a tradução para o português seria o mesmo que mesa de ajuda, para Cavalari e Costa (2015) tem por finalidade ser um facilitador de informações ao cliente. Sendo composto por três básicos componentes que são o software, equipe e metodologia de serviço.

O software controlará o fluxo de atendimentos e o tempo médio dos atendimentos; A equipe que atuará com a resolução dos problemas, ou identificando e mapeando os mesmos; A metodologia de serviço que é a parte que será definido as estratégias de ações, apresentando os indicadores de desempenho e identificando novas oportunidades para melhoria (CAVALARI e COSTA, 2015).

O *help desk* é a central de atendimento que possui o objetivo de solucionar simples imprevistos que ocorrem com os clientes. A finalidade de um *Help Desk* é tratar de forma rápida e com eficiência o problema ou dúvida do usuário (ROSARIO; PEREIRA; SILVEIRA, 2017).

Um Sistema Help-Desk possui a função de contribuir para aprimorar o trabalho da equipe de suporte das empresas, auxiliando na coordenação e solução de dificuldades trazidas pelos usuários, com o objetivo de atender e resolver estes incidentes com eficácia e eficiência. (LINKE; SILVEIRA, 2015, p.5).

### 2.2 METODOLOGIA RAD

A metodologia RAD surgiu nos anos 70, época em que as aplicações levavam muito tempo de desenvolvimento, que muitas das vezes os requerimentos eram mudados antes mesmo que a aplicação estivesse pronta. Foi em 1991 com o livro "*Rapid Application Development*", que foi escrito por James Martin que a metodologia RAD se formalizou como prática de desenvolvimento (PISKE e SEIDEL, 2006).

“O RAD encoraja a participação do usuário no processo de análise e *design*, o produto final tende a ter custos menores de manutenção e menor tempo de

desenvolvimento, em geral comprometendo a escalabilidade e o desempenho” (PISKE e SEIDEL, 2006, p.2).

Martin define os principais objetivos da RAD como: sistemas de alta qualidade, rápido desenvolvimento e entrega e baixos custos. Esses objetivos podem ser resumidos em uma frase: A necessidade comercial de entregar aplicativos de negócios funcionais em prazos mais curtos e com menos investimento. (apud DAVIES et al, 1999, p.2, nossa tradução).

Pelo fato das ferramentas RAD estarem introduzidas nas plataformas de sistemas proprietários, como exemplo o sistema operacional *Windows*, pode-se destacar duas ferramentas que são bem conhecidas no mercado como a *Microsoft Visual Basic* e o *Borland Delphi*. Pelo *Delphi* pode-se obter uma utilização de uma linguagem prática e direta e pelo fato de as construções da linguagem serem simples ganha-se no aprendizado mais rápido, obtendo assim um ganho de produtividade (PISKE e SEIDEL, 2006, p.2).

### 2.2.1 Fases da Metodologia RAD

Segundo Sbrocco e Macedo (2012) a metodologia RAD é composta por 5 fases, são elas:

- Modelagem de negócios - Nessa fase o principal objetivo é entender o processo, pois ela é voltada a responder às seguintes perguntas: Quais são as informações que dirigem o processo? Qual informação é gerada? Quem gera? Para onde ela vai? Quem processa a informação? (SBROCCO, MACEDO, 2012).
- Modelagem de dados - É nessa fase que os processos e os requisitos serão identificados para serem desenvolvidos e utilizados. É nela que o fluxo de informações que foi definido, será refinado em um conjunto de objetos de dados, que são essenciais para dar apoio ao negócio (SBROCCO, MACEDO, 2012).
- Modelagem de processos - É nessa fase que será montado o fluxo para ser implementado. Os objetos de dados são transformados com a finalidade de conseguir o fluxo de informação para implementação da função dos negócios (SBROCCO, MACEDO, 2012).
- Geração de aplicativos - É nessa fase que as informações vão virar código. O modelo RAD destaca o reuso, assim é reusado os componentes quando

preciso ou criado componentes novos quando inevitável (SBROCCO, MACEDO, 2012).

- Testes e entrega - Por destacar o reuso muitos dos componentes já foram testados, no qual não precisará testar tão exaustivamente quanto os novos componentes, dessa forma reduzindo o tempo de testes (SBROCCO, MACEDO, 2012).

## 2.3 FERRAMENTAS

### 2.3.1 Software Evolução Sistemas

A Evolução Sistemas<sup>1</sup> é uma empresa de desenvolvimento de softwares, atualmente contando com 12 sistemas atuando no mercado em diversos segmentos, atendendo 4 estados brasileiros, com mais de 700 empresas sendo atendidas, como restaurantes, lanchonetes, óticas, mercados e supermercados, casas agropecuárias, farmácias, oficinas de móveis e decorações, lojas de confecções entre outras. Os sistemas são de gerenciamento comercial. Algumas de suas funções incluem o controle de caixa, vendas, orçamentos, relatórios, controle de promoções, envio de NFe e NFCe, manifestações de notas, controle de estoque, entre muitas outras funções.

A empresa dispõe de um software interno chamado *EcenterManager* ao qual será integrado o aplicativo proposto neste trabalho, trata-se de um ERP que possui o monitoramento dos atendimentos realizados pelos técnicos, gerencia as liberações dos clientes, encaminha os boletos, realiza a emissão de relatórios entre muitas outras funções.

### 2.3.2 Linguagem de programação Delphi

Em 1994 a Borland lançou a ferramenta *Delphi*, derivada da linguagem de programação *Object Pascal* que é orientada a objetos. Desde o ano de seu nascimento o *Delphi* teve diversas versões cada uma com novas implementações e novidades, de 1994 até 2003 com as versões de 1 a 8, após tendo suas versões

---

<sup>1</sup> Empresa desenvolvedora de softwares comerciais. Disponível em: <<http://portalevolucao.com/>> Acesso em: 30 de set. de 2020.

renomeadas com seu ano de lançamento como as versões *Delphi 2005* e *2006*, no mesmo ano de 2006 foi lançada a versão que foi batizada com o nome *Turbo Delphi*, ela era a versão reduzida do *Delphi 2006* com preços populares (LEITE, 2008).

Atualmente as versões estão sendo renomeadas como *Delphi XE* pela empresa Embarcadero<sup>2</sup>.

"O *Delphi* tem o estilo de ferramenta faz-tudo, pois com ela é possível construir desde aplicações comuns até complexos sistemas de banco de dados, plugados ou não na internet." (FEDELI, 2010, p. 79).

Acima de tudo o *Delphi* possui uma IDE (Ambiente de Desenvolvimento Integrado) que facilita a criação de aplicações de forma rápida pela seleção de componentes na paleta. É uma ferramenta muito poderosa, tendo como característica a metodologia RAD, desta maneira facilitando o desenvolvimento de softwares GUI (Interface Gráfica do Usuário) (FEDELI, 2010).

### **2.3.3 Banco de dados *FirebirdSQL* E *PostgreSQL***

#### **2.3.3.1 *FirebirdSQL***

Segundo Cantú (2010), o *Firebird* é um sistema de gerenciamento de banco de dados (SGBD) de código aberto e não possui licença dupla, ou seja, pode-se usar em qualquer tipo de aplicação gratuitamente em qualquer finalidade.

É um poderoso sistema de gerenciamento de banco de dados relacional SQL (SGBDR) cliente / servidor compacto que pode ser executado em uma variedade de plataformas de sistema operacional cliente e servidor, incluindo Windows, Linux e diversas outras plataformas UNIX, incluindo FreeBSD e Mac OS X. É um SGBDR de força industrial que apresenta um alto nível de conformidade com os padrões SQL, enquanto implementa alguns recursos de linguagem poderosas na esfera específica do fornecedor de programação de procedimento (Borrie, 2004, p.xxiii, tradução nossa).

Por volta dos anos 2000 a *Borland* que é a proprietária do banco de dados *InterBase*, resolveu abrir seu código fonte na versão 6.0, foi nessa abertura que nasceu o *Firebird 1.0* com alguns desenvolvedores que resolveram identificar e corrigir defeitos. Mesmo tendo origens na *interbase*, o *firebird* evoluiu com características e

---

<sup>2</sup> Empresa desenvolvedora de ferramentas para desenvolvedores de software. Disponível em: <<https://www.embarcadero.com/br/products/delphi>> Acesso em: 25 de set. de 2020

ferramentas próprias e aspectos distintos, sendo assim, não se deve considerá-lo igual ou uma extensão do *Interbase* (NETO, 2011).

Vale destacar que será usado *FirebirdSQL* apenas para fazer a identificação do cliente, pois, o sistema comercial que o cliente faz uso e este o banco de dados utilizado, já o *PostgreSQL* será o banco de dados definido para a aplicação *Help Desk*.

### 2.3.3.2 PostgreSQL

Segundo a documentação do *PostgreSQL* (2020), pode-se conceituar o mesmo como um sistema de gerenciamento de banco de dados (SGBD) Relacional de objeto. É de código aberto, ou seja, pode ser usado, modificado e distribuído por qualquer pessoa gratuitamente, para qualquer finalidade.

“Utilizando os conceitos mais básicos, o *PostgreSQL* é uma ferramenta encarregada de armazenar dados e gerenciar o acesso de cada informação de acordo com as regras previamente definidas.” (MILANI, 2008, p.26).

De acordo com Milani, (2008) o *PostgreSQL* teve seu nascimento em um projeto na universidade Berkeley, na Califórnia (EUA) no ano de 1986. O intuito era criar um modelo e as regras de um novo sistema de armazenamento de dados, tinha como apoiadores desse projeto os órgãos *Army Research Office* como também a *National Science Foundation*. Em 1987 a primeira versão ficou pronta, e em 1989 a primeira versão estável. Mas em torno de 1991, o código foi adquirido pela empresa *Illustra Information Technologies* que se fundiu com a empresa *Informix* que hoje pertence a IBM. Em 1994 o projeto foi encerrado e foi dado origem ao *Postgres95* que foi incorporando a linguagem SQL e compatibilizado com o padrão ANSI C. Foi no ano de 1996 que teve mais melhorias e o nome teve outra mudança para *PostgreSQL*, que é o que conhecemos hoje.

## 3. MATERIAIS E MÉTODOS

### 3.1 MÉTODOS

Optou-se por utilizar para esse projeto a pesquisa exploratória, pois permitirá maior proximidade com o tema e o procedimento estudo de caso, para um amplo e aprofundado conhecimento. “As pesquisas exploratórias têm como propósito

proporcionar maior familiaridade com o problema, com vistas a torná-lo mais explícito ou a construir hipóteses” (GIL, 2018, p.25).

O estudo de caso por sua vez é definido como uma “modalidade de pesquisa amplamente utilizada nas ciências sociais. Consiste no estudo profundo e exaustivo de um ou poucos casos, de maneira que permita seu amplo e detalhado conhecimento” (GIL, 2018, p.33).

### **3.1.1 Objeto de estudo**

O objeto estudado será a empresa Evolução Desenvolvimento de Sistemas, situada em Ouro Preto do Oeste/RO, na avenida Daniel Comboni nº 2069, no bairro União. Está no mercado há 14 anos contando com mais de 700 clientes, sendo distribuídos em 4 estados brasileiros. Além dos aplicativos comerciais, a empresa possui alguns aplicativos para uso interno, sendo um deles o ERP *EcenterManager*, ao qual a aplicação proposta neste trabalho será integrada como um módulo desse sistema.

### **3.1.2 Metodologia de desenvolvimento**

A metodologia escolhida para o desenvolvimento do aplicativo é o modelo RAD. Esse modelo possui 5 fases, detalhamos aqui como foi desenvolvido o aplicativo seguindo esse modelo.

Fase 1 - Modelagem de negócios - Nessa fase com uma comunicação direta com os responsáveis da empresa Evolução foi buscado entender o processo, e definir as características do fluxo de informações. Respondendo às perguntas necessárias para essa fase, como o que é gerado? seria os protocolos entre técnicos e clientes, para onde a informação vai? Os dados irão para o banco de dados *PostgreSQL*.

Fase 2 - Modelagem dos dados - Nessa fase, foram analisadas as informações levantadas na fase anterior, foram filtrados os dados relevantes e foi levantado os requisitos.

Fase 3 - Modelagem de processos - Nessa fase foram recolhidas as informações que foram filtradas na fase anterior e foi convertida em informações necessárias para a montagem do fluxo da implementação. Como por exemplo as funções para adicionar, modificar, descartar ou recuperar um elemento qualquer.

Fase 4 - Geração de aplicativos - Essa é a fase da implementação, no qual foi criado os códigos quando necessário e reusado os códigos quando disponíveis na biblioteca que a empresa dispõe.

Fase 5 - Testes e modificações - Por fim foi a fase de testes e possíveis modificações em que foi assegurado o funcionamento adequado do aplicativo. Cada protótipo foi testado separadamente, porém alguns não tanto quanto outros pois, como muitos já haviam sido testados anteriormente em outras aplicações, não foram testados tão abrangentemente quanto a componentes novos, mas não foram deixados de serem testados também. Com os testes sendo feitos separadamente pode-se conseguir identificar as melhorias e alterações para o bom funcionamento do aplicativo. Após seguimos para os testes com alguns clientes que foram selecionados pelos técnicos.

## **4. DESENVOLVIMENTO**

### **4.1 DESCRIÇÃO DO SISTEMA**

O aplicativo terá o objetivo de fazer a comunicação da empresa Evolução Sistemas com os seus clientes, com o intuito de resolver possíveis dúvidas ou eventuais erros.

Haverá a possibilidade de visualização de histórico de conversas, tanto para o cliente, quanto para os técnicos, no qual terá o filtro pelo período no lado do cliente, e no lado do técnico será filtrado pelo período e pelo número de protocolo.

O sistema do lado do cliente será instalado ao abrir o aplicativo evolução, mas isso somente se caso já não estiver instalado, ou poderá também simplesmente colocar o aplicativo na pasta bin do sistema. Terá a funcionalidade de serviço do Windows, assim quando o cliente ligar o computador o aplicativo será aberto, facilitando na hora que o cliente for chamar no suporte.

Terá uma identificação por cor nos protocolos, em que azul será para atendimentos que ainda não foram respondidos, verde para os atendimentos que estão em andamento e roxo para atendimentos já finalizados. Nas conversas em andamento a legenda de cor será azul para mensagens que o cliente ou o técnico ainda não visualizou, verde para as mensagens visualizadas que foi enviada e verde claro para mensagens que foram recebidas.

No lado do técnico ele poderá filtrar todos os clientes podendo pesquisar pelo nome ou id, como também ele poderá filtrar apenas pelos atendimentos dele em aberto, como também os atendimentos sem respostas e todos os atendimentos que estão em aberto tanto dele quanto dos outros técnicos. Ao filtrar pelo cliente, o técnico poderá selecionar pelos computadores que o cliente venha a ter, e nesse terá a identificação dos nomes do computador como também se a mensagem já foi lida ou não, e quantos protocolos já foram abertos em cada computador.

#### 4.2 MODELAGEM DO SISTEMA

No quadro 1 a seguir serão apresentados os requisitos funcionais, onde RF significa Requisitos funcionais.

Quadro 1 - Requisitos Funcionais

	Requisitos Funcionais
RF01	O sistema deverá notificar aos técnicos os chamados ainda não respondidos.
RF02	O sistema deverá mostrar aos técnicos os protocolos em aberto.
RF03	O sistema deverá permitir que o cliente abra um novo protocolo.
RF04	O sistema deverá permitir que o cliente coloque o nome dele no protocolo.
RF05	O sistema deverá permitir o envio e o recebimento de mensagens dos técnicos e clientes ao ser aberto um protocolo.
RF06	O sistema deverá salvar as mensagens e protocolos em tempo real.
RF07	O sistema deve permitir que clientes e técnicos vejam seus históricos arquivados.

Fonte: Próprio autor com base na ABNT, 2020

No quadro 2, serão apresentados os requisitos não funcionais, onde RNF significa Requisito não Funcional.

Quadro 2 - Requisitos Não Funcionais

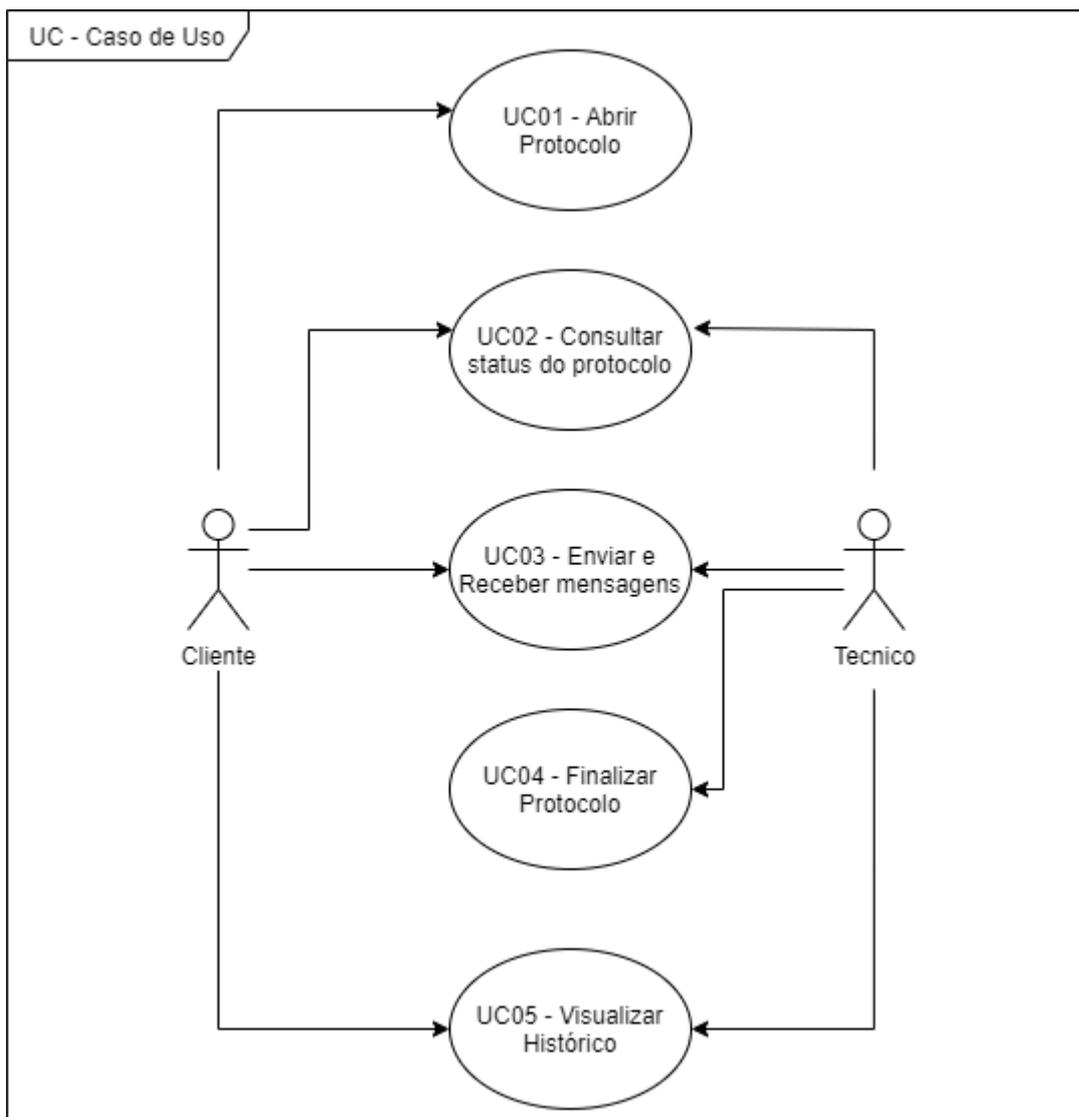
<b>Identificador</b>	RNF01	<b>Categoria</b>	Implementação	<b>RF</b>	Global
<b>Nome</b>	O sistema deverá ser implementado na linguagem <i>Delphi</i> .				
<b>Prioridade</b>	Essencial				
<b>Descrição</b>	O sistema deverá ser desenvolvido na linguagem de programação <i>Delphi Community</i> na versão 10.3.				
<b>Identificador</b>	RFN02	<b>Categoria</b>	Identificação	<b>RF</b>	Global
<b>Nome</b>	O sistema deverá utilizar a base de dados da Evolução Sistemas e a do cliente para identificá-lo.				
<b>Prioridade</b>	Essencial				
<b>Descrição</b>	O sistema deverá utilizar o banco de dados <i>PostgreSQL</i> 9.6 do lado da Evolução Sistemas e <i>firebirdSQL</i> 2.5 no lado do cliente.				
<b>Identificador</b>	RFN03	<b>Categoria</b>	Disponibilidade	<b>RF</b>	RF03
<b>Nome</b>	O sistema deverá estar disponível, em 99% do tempo, todos os dias da semana.				
<b>Prioridade</b>	Importante				
<b>Descrição</b>	Do lado do cliente o aplicativo funcionará com serviço, assim estará disponível a todo o momento, se caso o cliente finalizar o serviço, bastará clicar no ícone para funcionar novamente.				
<b>Identificador</b>	RFN04	<b>Categoria</b>	Segurança	<b>RF</b>	RF07
<b>Nome</b>	Visualização de histórico de conversas				
<b>Prioridade</b>	Importante				

<b>Descrição</b>	Somente irá visualizar o histórico de protocolos os funcionários autorizados da Evolução Sistemas e os clientes envolvidos no protocolo, ou seja, o cliente que solicitou ajuda.
------------------	--

Fonte: Próprio autor com base na ABNT, 2020

Na figura 1, será apresentado o diagrama de caso de uso.

Figura 1 - Diagrama de caso de uso



Fonte: Próprio autor com base na ABNT, 2020

Breve explanação de cada caso de uso e onde os atores farão as interações:

UC01 - Abrir Protocolo: O sistema deve permitir que o cliente abra protocolos para conseguir tirar suas dúvidas.

UC02 - Consulta status do protocolo: O sistema mostrará ao técnico e aos clientes os protocolos em aberto, os em andamento e os finalizados todos com diferenciação de cor. Para o lado do técnico o sistema deve notificar os que estão em aberto sem resposta.

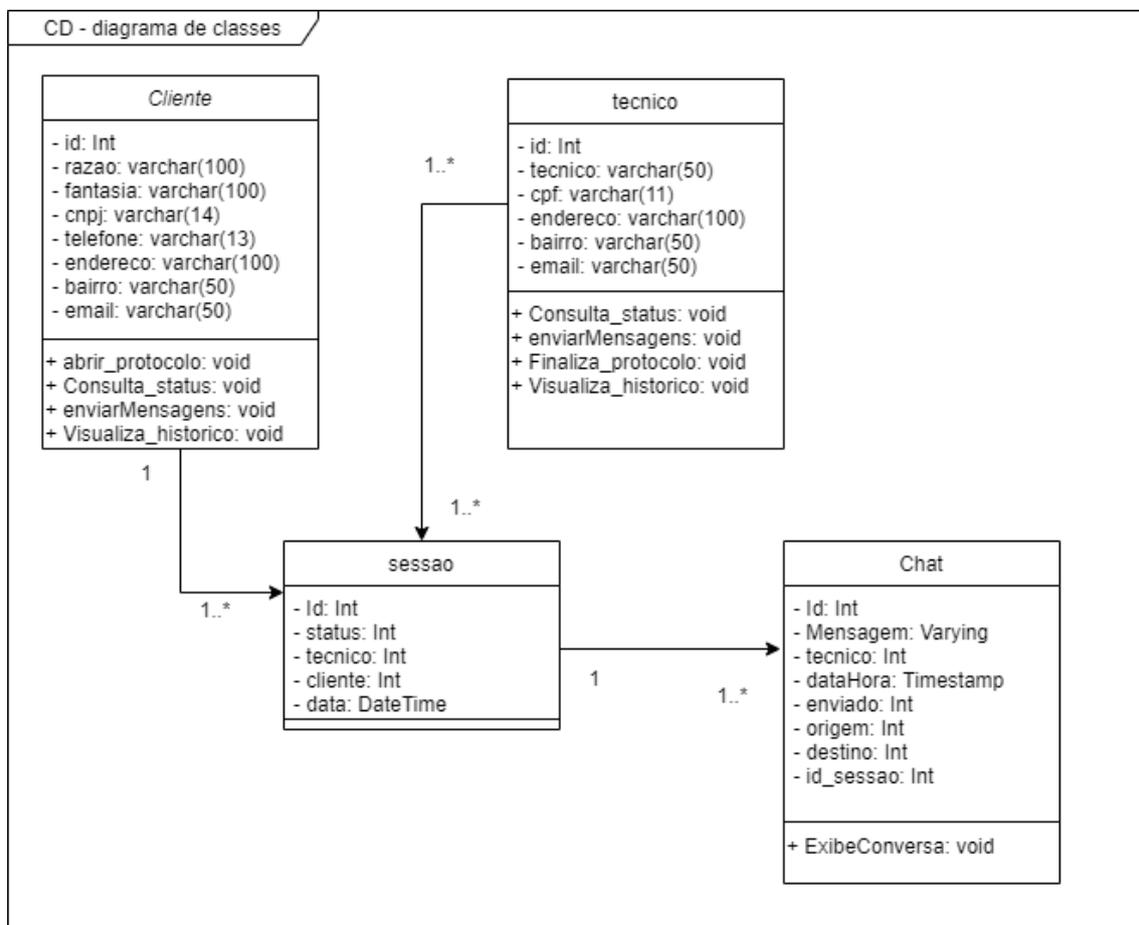
UC03 - Enviar e receber mensagens: Técnico e Cliente conseguirão enviar mensagens para se comunicarem.

UC04 - Finalizar protocolo: O sistema deve permitir que apenas o técnico finalize os protocolos.

UC05 - Visualizar histórico: Técnico e cliente poderão visualizar o seu protocolo, consultando pelo seu id ou pela data do mesmo.

Na figura 2 podemos analisar o diagrama de classes, no qual será apresentado os métodos de cada classe.

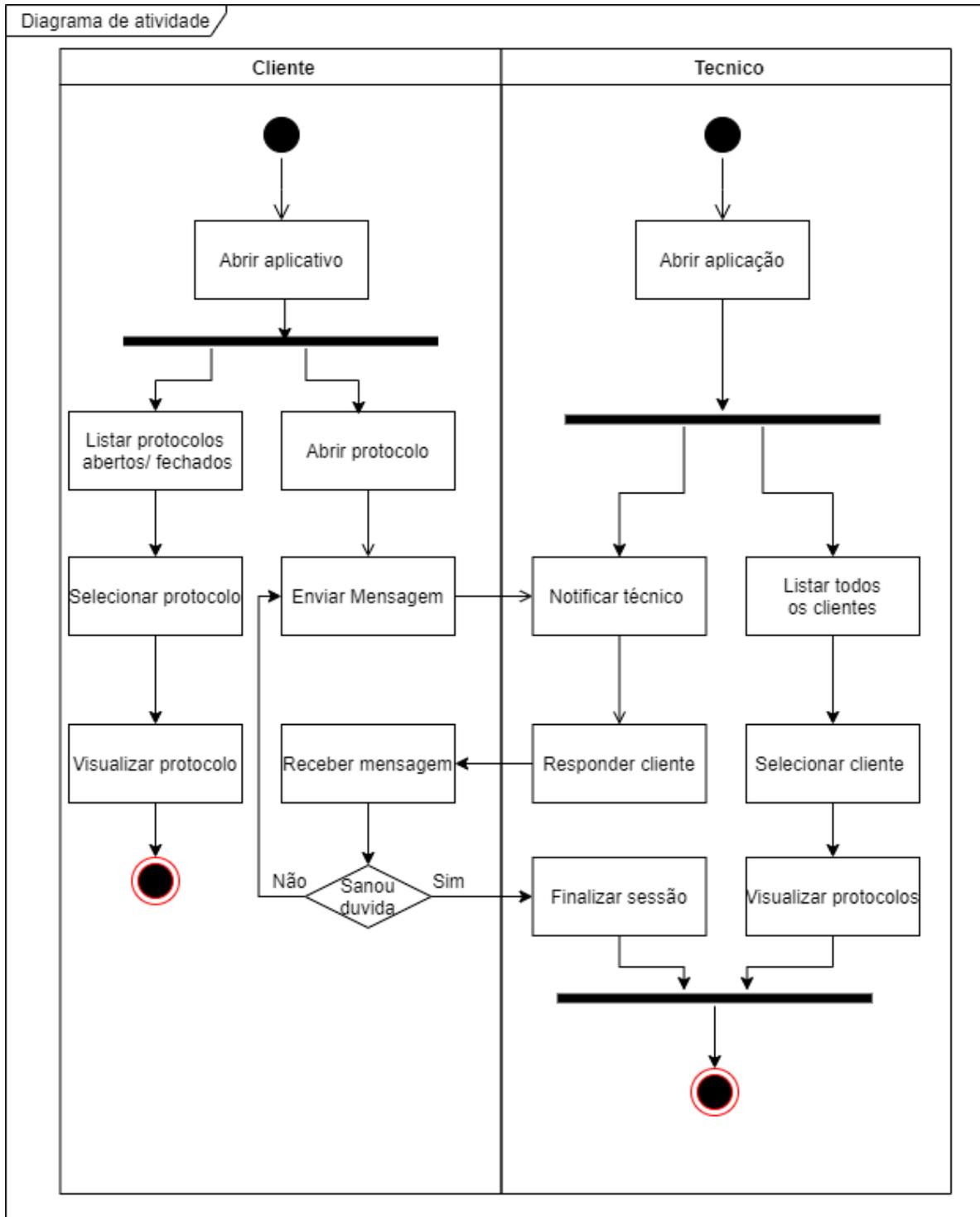
Figura 2 - Diagrama de classes



Fonte: Próprio autor com base na ABNT, 2020

Na figura 3, será apresentado o diagrama de atividades, que representará o fluxo das atividades.

Figura 3 - Diagrama de atividades



Fonte: Próprio autor com base na ABNT, 2020

### 4.3 MODELAGEM DO BANCO

No quadro 3, será apresentado o dicionário de dados. Vale ressaltar que o *firebirdSQL* será usado apenas para poder identificar o cliente, pois o sistema usado no software comercial é este banco, assim o dicionário de dados abaixo é voltado para o *PostgreSQL*.

Quadro 3 - Dicionário de dados

Entidade	Tb_sessao			
Descrição	Responsável por armazenar o protocolo.			
Atributo	Tipo	Domínio	Descrição	Restrições
Id	Int	ss_id	Código de identificação de tabela	PK
Status	Int	ss_idstatus	Identificação do protocolo	Não nulo
Técnico	Int	ss_idtec	Identificação do técnico	
Cliente	Int	ss_idcliente	Identificação do cliente	Não nulo
Data	DateTime	ss_histcadastro	Data do protocolo	Não nulo
Entidade	Tb_chat			
Descrição	Responsável por armazenar as mensagens dos protocolos			
Atributo	Tipo	Domínio	Descrição	Restrições
Id	Int	ch_id	Código de identificação de tabela	PK
Mensagem	Char varying	ch_mensagem	Conversas entre técnico e cliente	Não nulo
Nome cliente	Varchar(50)	ch_nomecli	Identificação do nome do cliente	

Técnico	Int	ch_idtec	Identificação do técnico	Não nulo
Cliente	Int	ch_idcliente	Identificação da empresa	Não nulo
Data e hora	DateTime	ch_dthrmsg	Data e hora da mensagem recebida	Não nulo
Enviado	Int	ch_enviado	Status da mensagem	
Origem	Int	ch_origem	Origem da mensagem	Não nulo
Destino	Int	ch_destino	Destino da mensagem	Não nulo
Id sessao	Int	ch_idsessao	Identificação do protocolo	Não nulo

Entidade	Tb_cliente			
Descrição	Armazena as informações dos clientes.			
Atributo	Tipo	Domínio	Descrição	Restrições
Id	Int	cl_id	Código de identificação de tabela	PK
Razão	Varchar(100)	cl_nome	Razão social do cliente	Não nulo
Fantasia	Varchar(100)	cl_fantasia	Nome fantasia do cliente	Não nulo
CNPJ	Varchar(14)	cl_cnpj	CNPJ do cliente	Não nulo
Telefone	Varchar(13)	cl_telefone	Telefone do cliente	
Endereço	Varchar(100)	cl_endereco	Endereço do cliente	
Bairro	Varchar(50)	cl_bairro	Bairro do cliente	
Email	Varchar(50)	cl_email	Email do cliente	

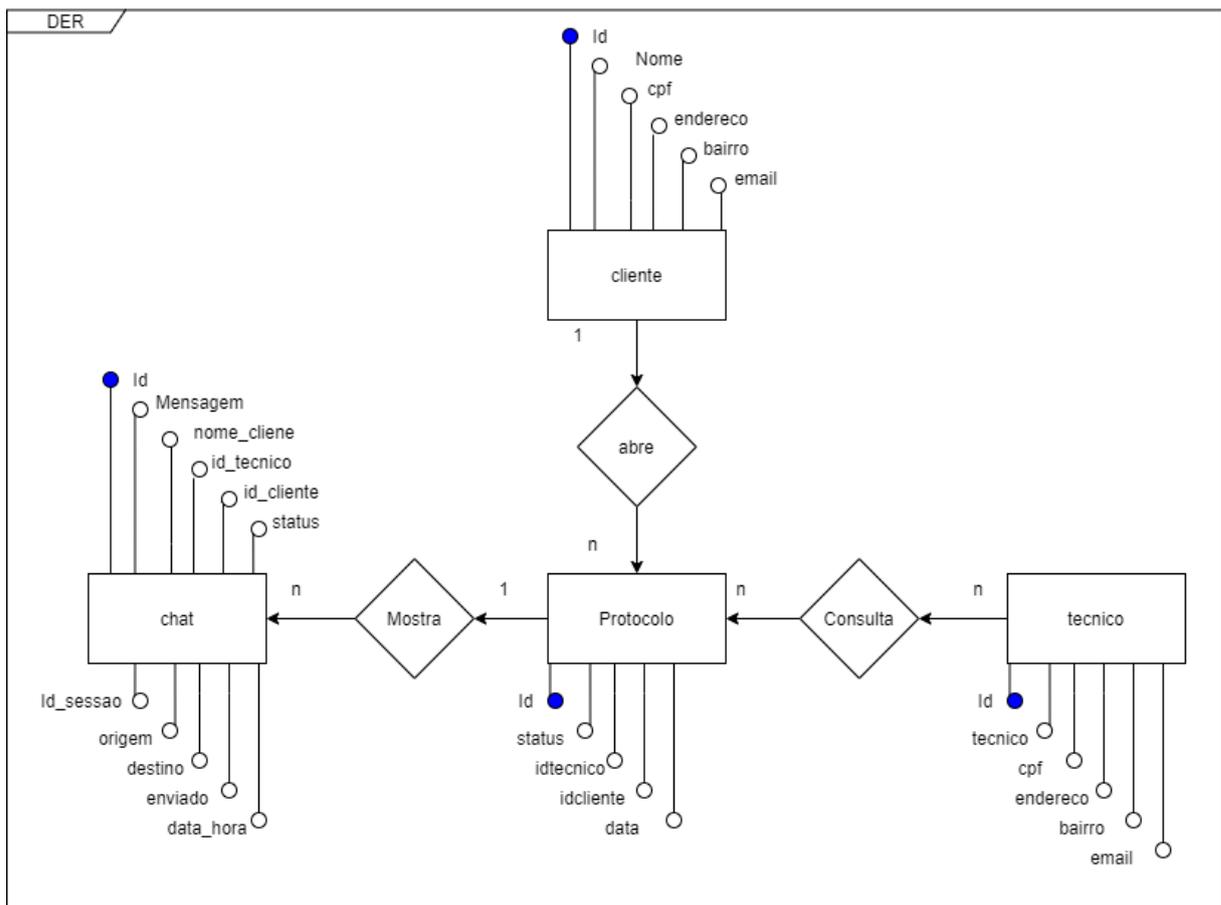
Entidade	Tb_tecnico
----------	------------

Descrição	Armazena as informações dos técnicos			
Atributo	Tipo	Domínio	Descrição	Restrições
Id	Int	tc_id	Código de identificação de tabela	PK
Tecnico	Varchar(50)	tc_tec	Nome do técnico	Não nulo
CPF	Varchar(11)	tc_cpf	CPF do técnico	Não nulo
Endereço	Varchar(100)	tc_endereco	Endereço do técnico	
Bairro	Varchat(50)	tc_bairro	Bairro do técnico	
Email	Varchat(50)	tc_email	Email do técnico	Não nulo

Fonte: Próprio autor com base na ABNT, 2020

Na figura 4, será apresentado o diagrama de entidade e relacionamento.

Figura 4 - Diagrama de entidade e relacionamento



Fonte: Próprio autor com base na ABNT, 2020

Neste diagrama temos os relacionamentos definidos, como a entidade cliente pode abrir várias sessões, e uma sessão pode ser aberta apenas por um cliente. A entidade técnico consulta várias sessões, e a sessão pode ser vista por vários técnicos. Por último a entidade sessão pode ter vários chats e o chat tem sua própria sessão.

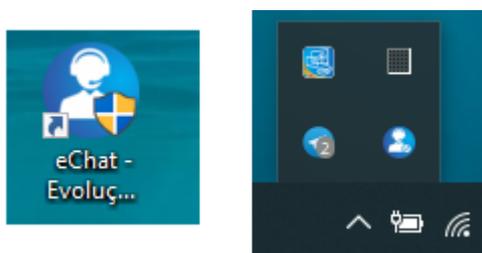
#### 4.4 RESULTADOS

A seguir como resultados obtidos pelo desenvolvimento será apresentado as telas que compõem a aplicação, tanto do lado do cliente quanto do lado do técnico e uma breve explicação sobre o funcionamento.

##### 4.4.1 Sistema do lado do cliente

Para acessar o aplicativo o cliente poderá fazer de duas maneiras: a primeira é clicando no ícone da aplicação na área de trabalho do computador, ou indo nos ícones ocultos do *Windows*. Na figura 5, será apresentado os meios que o cliente tem para acessar o aplicativo.

Figura 5 - meios do cliente acessar o aplicativo



Fonte: Próprio autor com base na ABNT, 2021

Na tela principal do aplicativo o cliente poderá visualizar o histórico de chamados já realizados anteriormente, os que estiverem na cor roxa, significa que já foi finalizado, na cor verde os que estão sendo atendidos, e na cor azul claro os que ainda não foram respondidos. Na figura 6 será apresentada a tela inicial do aplicativo.

Figura 6 - Tela inicial

**EVOLUÇÃO SISTEMAS**

## Histórico de Conversas

Filtro de Conversas

Período das Conversas

Período: Este Mês

De: 16/04/2021

Até: 16/05/2021

2021.4.1.3

**Novo Chat**

[ESTE COMPUTADOR]

[20256] 17/04/2021 19:46  
TEC: Aline dos Santos

[20255] 17/04/2021 19:43  
TEC: Aline dos Santos

[20254] 17/04/2021 19:40  
TEC: Aline dos Santos

Em Aberto    Em Andamento    Finalizado

Fonte: Próprio autor com base na ABNT, 2021

Ao clicar em Novo chat abrirá uma tela, na qual o cliente iniciará o protocolo e assim esclarecer suas dúvidas. Na figura 7 será apresentado um exemplo de protocolo em aberto.

Figura 7 - Protocolo em andamento

The screenshot shows a chat window for 'EVOLUÇÃO SISTEMAS' with protocol number [20256]. The user 'Robson' is logged in with phone number 99.523.183/2491-23. A red box prompts the user to enter their name. The chat history shows Robson's message at 19:46 and a technician's response at 19:47. At the bottom, a status bar shows 'Você' (You) as 'Visualizado' (Viewed), and a text input field with a blue arrow button.

**EVOLUÇÃO SISTEMAS**  
Protocolo: [20256]

Suas Credenciais

**Robson**  
99.523.183/2491-23

Para um melhor atendimento informe seu nome no campo abaixo:  
Digite Seu Primeiro Nome  Salvar Nome

✓ Robson:  
[19:46] • Olá, pode me ajudar ?

sábado, 17 de abril de 2021

TEC: Aline dos Santos  
[19:47] • Ola, Boa noite, em que posso ajudar?

Você ✓ Visualizado Técnico Atenção Sistema

Digite Aqui

Fonte: Próprio autor com base na ABNT, 2021

#### 4.4.2 Sistema do lado do técnico

Na página inicial do ERP EcenterManeger o técnico poderá acessar a aplicação clicando no ícone do chat, que fica ao lado da foto do usuário. Na figura 8 será apresentado a tela inicial do EcenterManeger.

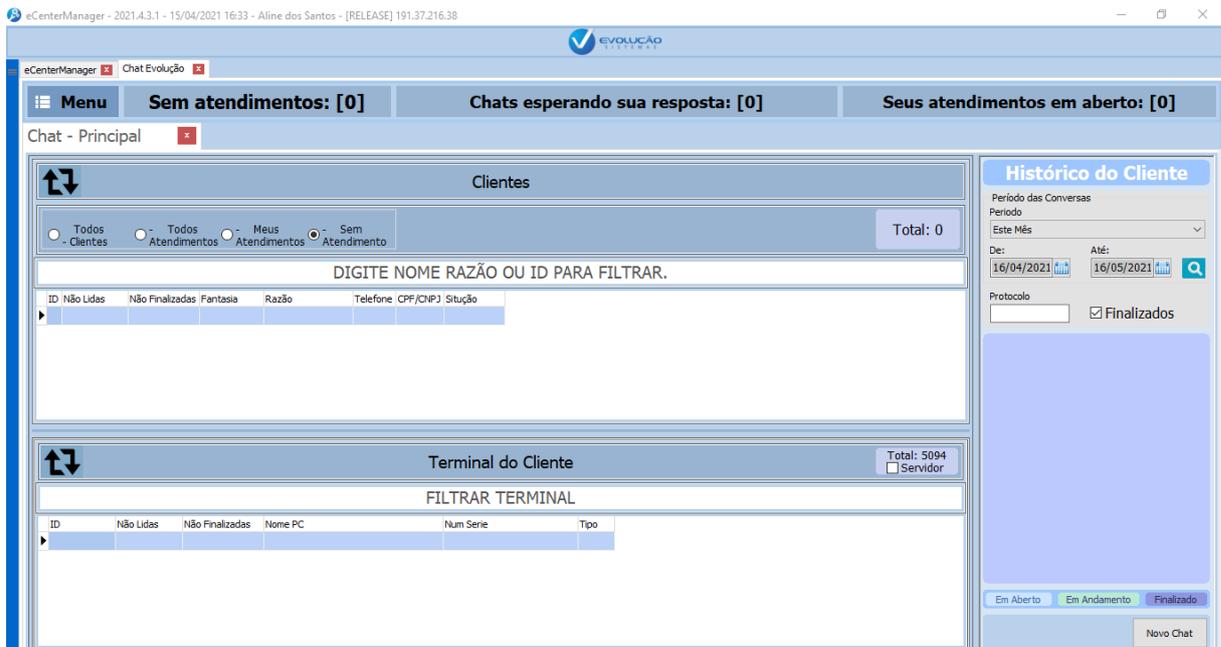
Figura 8 - Tela inicial do ERP EcenterManeger



Fonte: Próprio autor com base na ABNT, 2021

Na figura 9 será apresentado a tela inicial do chat, logo depois será explicada cada ponto da tela.

Figura 9 - Tela inicial do chat



Fonte: Próprio autor com base na ABNT, 2021

Na parte superior é a parte em que o técnico será notificado, se possui algum cliente sem atendimento, se tem algum atendimento dele esperando por resposta e quantos atendimentos ele tem em aberto. Na figura 10, apresentará um exemplo.

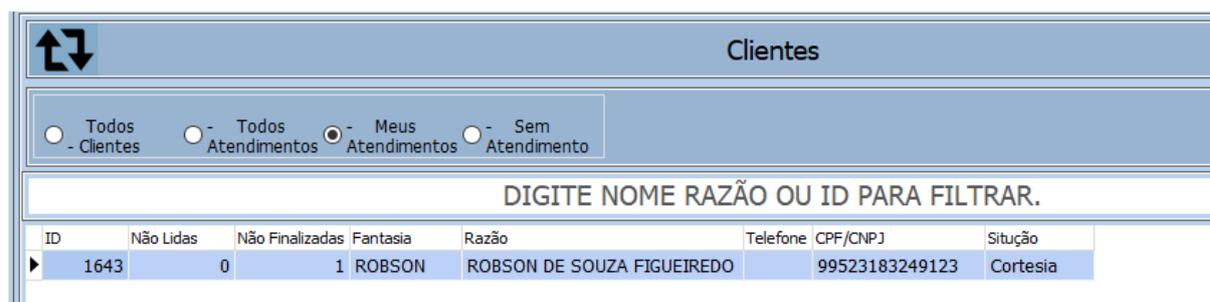
Figura 10 - exemplo de notificação



Fonte: Próprio autor com base na ABNT, 2021

Um pouco abaixo é a parte dos filtros, no qual o técnico poderá filtrar por todos os clientes e fazer sua pesquisa por um cliente em específico, onde poderá ver o histórico deste cliente, como também poderá filtrar por todos os atendimentos que estão em aberto, tem a opção de filtrar apenas pelos seus atendimentos e poderá filtrar apenas pelos que ainda não foram respondidos. Na figura 11, apresentará um exemplo.

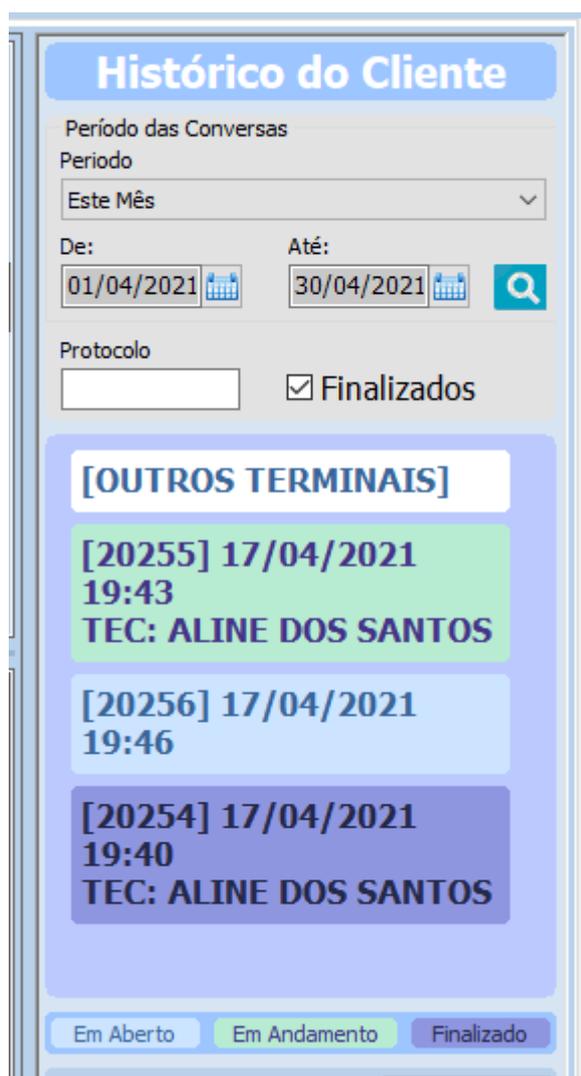
Figura 11 - exemplo filtros de atendimento



Fonte: Próprio autor com base na ABNT, 2021

Ao selecionar um cliente, ao lado esquerdo terá a opção de ver o histórico ou de selecionar um chat ainda não respondido para assim poder atender, tendo a opção de cor conforme seu estado, roxo para protocolos já finalizados, verde para atendimentos aberto e azul para protocolos ainda sem respostas, conforme será mostrado na figura 12.

Figura 12 - chats aberto, fechado e em andamento



Fonte: Próprio autor com base na ABNT, 2021

Ao selecionar um protocolo será aberto uma nova guia dentro do aplicativo, em que o técnico poderá ver o histórico ou responder o cliente. Na figura 13, será apresentado uma guia aberta.

Figura 13 - Guia aberta de um protocolo



Fonte: Próprio autor com base na ABNT, 2021

#### 4.5 DISCUSSÕES

Ao perceber a dificuldade de alguns clientes de se comunicar com o suporte do sistema, surgiu a ideia de desenvolver a aplicação com o intuito de ajudar esses clientes. Para a implementação foi utilizado a metodologia RAD que possui 5 fases, sendo a primeira na qual buscamos entender o processo, na segunda foi levantado os requisitos, a terceira foi montando o fluxo de implementação, a quarta fase é a de desenvolvimento e pôr fim a fase de testes, que por sua vez foi dividida em duas partes, sendo a primeira que foi testado cada protótipo separadamente e feito as devidas modificações, a segunda foi os testes com alguns clientes.

Na fase de testes tivemos alguns contratemplos com alguns antivírus que os clientes possuíam instalados em seus computadores, que apagavam ou colocavam em quarentena o aplicativo, no qual para resolver foi necessário colocar o aplicativo na lista de exceções do antivírus.

Após o desenvolvimento da aplicação, a princípio foram escolhidos dois clientes que foi constatado pelos técnicos que tinham dificuldade para se comunicarem e dois clientes que não tinham essa dificuldade, assim pode-se fazer uma análise dos resultados obtidos com a aplicação.

Os clientes que não tinham problemas para se comunicar com a central obteve resultados melhores, foi aberto o aplicativo rapidamente, e as mensagens eram enviadas e recebidas instantaneamente, já os clientes que tinham dificuldades de se comunicar, tiveram uma pequena demora ao abrir o sistema, porém nada muito exagerado, enquanto um teve uma demora de 6 segundos, o outro de 7 segundos, e ao enviar e receber as mensagens também na mesma faixa, entre 5 à 8 segundos, após foram realizados mais testes com outros clientes. Dessa forma conseguindo bons resultados com a aplicação.

## 5. CONCLUSÃO

Constatou-se, que após a realização deste trabalho, que a empresa Evolução Sistemas tinha uma dificuldade para se comunicar com alguns de seus clientes que careciam de uma internet estável, pois os colaboradores possuíam muitos aplicativos para abrir durante o dia de trabalho, diminuindo, assim a produtividade. Dessa forma era importante desenvolver um aplicativo *Help Desk* para a empresa.

Diante disso, o objetivo geral proposto foi desenvolver um aplicativo de gerenciamento de atendimento *Help Desk*, unificando o ERP EcenterManeger na empresa Evolução Sistemas. Constatou-se que o objetivo geral e os objetivos específicos foram atendidos porque efetivamente o trabalho conseguiu demonstrar que foi possível o desenvolvimento do protótipo desktop funcional, conforme estava previsto no escopo do projeto.

Pela limitação de tempo foi escolhida a metodologia RAD, pela sua vantagem de desenvolvimento rápido. Com o intuito de validação da aplicação foram realizados testes com alguns clientes, nos quais foram obtidos resultados consideravelmente bons tanto para as conexões estáveis quanto para as instáveis. De acordo com a

gerência da empresa, a aplicação foi bastante útil para o relacionamento com seus clientes, com o qual os testes feitos não houve demora para o cliente abrir o sistema como também para mandar e receber mensagens, e com a unificação do aplicativo ao ERP, já existente da empresa, o técnico não precisará mais abrir várias telas, dessa forma aumentando sua produtividade. Diante disso a empresa aprovou a aplicação e já está colocando em prática o seu uso.

O aplicativo desenvolvido possibilita ao cliente a visualização de protocolos, no qual poderá acessar a qualquer momento sem precisar solicitar antes, deste modo garantindo uma melhor transparência e até mesmo sendo possível consultar uma dúvida já esclarecida anteriormente. Sendo assim, o aplicativo contribuirá para que os clientes da empresa tirem as dúvidas as quais venham a ter, evitando assim possíveis contratemplos oferecendo qualidade no serviço prestado pela empresa Evolução Sistemas.

## REFERÊNCIAS

BORRIE, Helen; **The Firebird Book: A Reference for Database Developers**. Apress, 2004. Disponível em: <<https://books.google.com.br/books?hl=pt-BR&lr=&id=ENt06WRFvvgC&oi=fnd&pg=PR17&dq=firebird&ots=G0P46G1qsW&sig=8EkANuSKd386wdME7mnjTvJF7nU#v=onepage&q=firebird&f=false>> Acesso em: 25 de set. de 2020.

CANTU, Carlos H. Conheça o Firebird em 2 minutos. **Firebird**, 2010. Disponível em: <[https://www.firebirdnews.org/docs/fb2min\\_ptbr.html](https://www.firebirdnews.org/docs/fb2min_ptbr.html)> Acesso em: 25 de set. de 2020.

CAVALARI, Gabriel O. T.; COSTA, Heitor A. X. Modelagem e Desenvolvimento de um Sistema Help-Desk para a Prefeitura Municipal de Lavras - MG. **Revista Eletrônica de Sistema de Informação**, v.4, p. 1-18, 2004. Disponível em: <<http://www.periodicosibepes.org.br/index.php/reinfo/article/view/158/50>> Acesso em: 14 de set. de 2020.

DAVIES, Paul Beynon; et al. Rapid application development (RAD): an empirical review, **European Journal of Information Systems**. 1999. Disponível em: <[https://www.researchgate.net/profile/Paul\\_Beynon-Davies/publication/31978101\\_Rapid\\_application\\_development\\_RAD\\_An\\_empirical\\_review/links/546f48cc0cf2d67fc0310860/Rapid-application-development-RAD-An-empirical-review.pdf](https://www.researchgate.net/profile/Paul_Beynon-Davies/publication/31978101_Rapid_application_development_RAD_An_empirical_review/links/546f48cc0cf2d67fc0310860/Rapid-application-development-RAD-An-empirical-review.pdf)> Acesso em: 24 de set. de 2020.

Delphi. **Embarcadero**, c2020. Disponível em: <<https://www.embarcadero.com/br/products/delphi>> Acesso em: 25 de set 2020.

Documentação do PostgreSQL 9.6.20: O que é PostgreSQL?. **PostgreSQL**, c1996 - 2020. Disponível em: <<https://www.postgresql.org/docs/9.6/intro-whatIs.html>> Acesso em: 30 de set. de 2020.

EVOLUÇÃO SISTEMAS, **Portal Evolução**, c2020. Página inicial. Disponível em: <<http://www.portalevolucao.com/>> Acesso em: 30 de set. de 2020.

FARIA, Joana. **Qualidade no Atendimento ao Cliente: Impacto na Satisfação**. Escola Superior de Comunicação Social, 2017 Disponível em: <[https://repositorio.ipl.pt/bitstream/10400.21/8343/1/Disserta%c3%a7%c3%a3o\\_PM\\_JoanaFaria\\_9608.pdf](https://repositorio.ipl.pt/bitstream/10400.21/8343/1/Disserta%c3%a7%c3%a3o_PM_JoanaFaria_9608.pdf)> Acesso em: 07 de jun de 2021.

FEDELI, Daniel R.; POLLONI, Enrico G. F.; PERES, Fernando E. Introdução à ciência da computação. 2 ed. São Paulo: Cengage Learning, 2010.

GIL, Antonio Carlos. **Como elaborar projetos de pesquisa**. 6. ed. São Paulo: Atlas, 2018.

LEITE, Mario; **Acessando banco de dados com ferramentas RAD: aplicações em Delphi**. Rio de Janeiro: Brasport, 2008. Disponível em : <<https://books.google.com.br/books?hl=pt-BR&lr=&id=elbxWBmdl34C&oi=fnd&pg=PA1&dq=linguagem+de+programa%C3%A7>>

%C3%A3o+delphi&ots=bmsr4z-  
 auX&sig=F26bfyQ4VvFg0164oh8iciZqvlS#v=onepage&q=linguagem%20de%20progr  
 ama%C3%A7%C3%A3o%20delphi&f=false> Acesso em: 24 de set de 2020.

LINKE, Leonardo L.; SILVEIRA, Sidnei R. **Implantação de um Sistema Help-Desk: um estudo de caso na Exatus Soluções Estratégicas**, 2015. Disponível em: <[https://repositorio.ufsm.br/bitstream/handle/1/12839/TCCG\\_SIFW\\_2015\\_LINKE\\_LEONARDO.pdf?sequence=1&isAllowed=y](https://repositorio.ufsm.br/bitstream/handle/1/12839/TCCG_SIFW_2015_LINKE_LEONARDO.pdf?sequence=1&isAllowed=y)> Acesso em: 09 de jun de 2021.

MILANI, André; **PostgreSQL: guia do programador**. São Paulo: Novatec Editora, 2008. Disponível em: <[https://books.google.com.br/books?hl=pt-BR&lr=&id=eb7fXbM70F4C&oi=fnd&pg=PA19&dq=banco+de+dados+postgres&ots=FW9d\\_vWKv-&sig=WdUb2L-OF0Opeqtn2CyTWC7mJs8#v=onepage&q=banco%20de%20dados%20postgres&f=false](https://books.google.com.br/books?hl=pt-BR&lr=&id=eb7fXbM70F4C&oi=fnd&pg=PA19&dq=banco+de+dados+postgres&ots=FW9d_vWKv-&sig=WdUb2L-OF0Opeqtn2CyTWC7mJs8#v=onepage&q=banco%20de%20dados%20postgres&f=false)> Acesso em: 27 de set. de 2020.

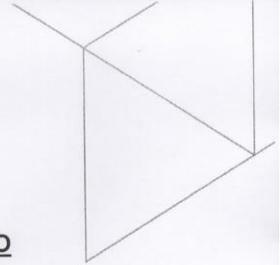
NETO, Sylvio V. B. **MySQL X Firebird: Uma Análise Comparativa de Desenvolvimento**. Trabalho de Conclusão de Curso – Faculdade de Tecnologia de Guaratinguetá, Guaratinguetá, 2011. Disponível em: <[https://www.warmboot.com.br/fb/imgdocs/tcc\\_fbmysql.pdf](https://www.warmboot.com.br/fb/imgdocs/tcc_fbmysql.pdf)> Acesso em: 26 de set. de 2020.

PISKE, Otavio R.; SEIDEL, Fábio A. **Rapid Application Development**, 2006. Disponível em: <<http://www.angusyong.org/arquivos/artigos/rad.pdf>> Acesso em 24 de set. de 2020.

ROSARIO, Dyeison P. V.; PEREIRA, Adriana S.; SILVEIRA, Sidnei R.; **ESTUDO DE CASO SOBRE O ATENDIMENTO REMOTO AOS CLIENTES DE UMA EMPRESA REPRESENTANTE DE EQUIPAMENTOS DE AUTOMAÇÃO**. Universidade Federal de Santa Maria, RS, 2017. Disponível em: <[https://repositorio.ufsm.br/bitstream/handle/1/12848/TCCG\\_SIFW\\_2017\\_ROSARIO\\_DYEISON.pdf?sequence=1&isAllowed=y](https://repositorio.ufsm.br/bitstream/handle/1/12848/TCCG_SIFW_2017_ROSARIO_DYEISON.pdf?sequence=1&isAllowed=y)> Acesso em: 09 jun 2021.

SBROCCO, JOSÉ HENRIQUE T. C.; MACEDO, PAULO C. **Metodologias ágeis: engenharia de software sob medida**. 1. ed. São Paulo: Érica, 2012.

## Anexo A – Ata de Trabalho de Conclusão de Curso



### ATA DE TRABALHO DE CONCLUSÃO DE CURSO

#### ATA Nº 06/2020 DE TRABALHO DE CONCLUSÃO DE CURSO

No vigésimo terceiro dia do mês de junho de 2021, no horário das 18h às reuniram-se o(a) Orientador(a) professor(a) Me. Maigon Nacib Pontuschka e os(as) professores(as) Prof. Me. Ana Flávia Moreira Camargo e Prof. Esp. Hailton Alves dos Reis para comporem Banca Examinadora de Trabalho de Conclusão de Curso, sob a presidência do(a) primeiro(a), para analisarem a apresentação do trabalho “**Desenvolvimento de Aplicativo Help Desk na empresa Evolução Sistemas**”. Após arguições e apreciação sobre o trabalho exposto foi atribuída à menção como nota do Trabalho de Conclusão de Curso dos(a) acadêmicos(a): Aline dos Santos Schmidt e Robson de Souza Figueiredo

**Obs:** Trabalho de Conclusão de Curso ( X ) aprovado ou ( ) reprovado com nota total de 8,6 (oito virgula seis) pontos, sendo atribuídos o valor 8,2 (oito virgula dois) ao trabalho escrito e 9,0 (nove) à apresentação oral.

*Aline dos Santos Schmidt*      *Robson S. Figueiredo*

ALINE DOS SANTOS SCHMIDT E ROBSON DE SOUZA FIGUEIREDO

*Maigon N. Pontuschka*

Prof. Me. Maigon Nacib Pontuschka  
Orientador

*Hailton Alves dos Reis*

Prof. Esp. Hailton Alves dos Reis

*Ana Flávia Moreira Camargo*

Prof. Me. Ana Flávia Moreira Camargo

*Thyago Bohrer Borges*

Prof. Me. Thyago Bohrer Borges  
Coord. Sistemas de Informação