



**CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E
DESENVOLVIMENTO DE SISTEMAS**

**CÁSSIA YAISA DIAS DE SOUZA
KEVIN RICKELME MARQUES DA SILVA**

DESENVOLVIMENTO DE UMA REDE SOCIAL ACADÊMICA

Guarulhos

2023

CÁSSIA YAISA DIAS DE SOUZA
KEVIN RICKELME MARQUES DA SILVA

DESENVOLVIMENTO DE UMA REDE SOCIAL ACADÊMICA

Trabalho de Graduação do Curso de Análise e Desenvolvimento de Sistemas, apresentado como requisito parcial para obtenção do Título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientadora: Profa. Esp. Jane Maria dos Santos Eberson.

Guarulhos

2023

CÁSSIA YAISA DIAS DE SOUZA
KEVIN RICKELME MARQUES DA SILVA

DESENVOLVIMENTO DE UMA REDE SOCIAL ACADÊMICA

Trabalho de Graduação apresentado ao Curso de Análise e Desenvolvimento de Sistemas como requisito parcial para obtenção do **Título de Tecnólogo em Análise e Desenvolvimento de Sistemas**.

Banca Examinadora

Orientador: _____

Profa. Esp. Jane Maria dos Santos Ebersson
Faculdade de Tecnologia de Guarulhos

Orientador: _____

Orientador: _____

Guarulhos, 23/06/2023

AGRADECIMENTOS

Agradecemos em primeiro lugar, a Deus, que fez com que nossos objetivos fossem alcançados. Por sempre nos acompanhar e ajudar nessa difícil jornada e por não nos fazer desistir no meio do caminho.

Reconhecemos que se não fosse por Deus e pelo apoio de nossas famílias, que nos incentivaram nos momentos mais difíceis e que compreenderam nossa ausência ao dedicar nosso tempo à realização deste trabalho, não chegaríamos até aqui.

Somos gratos a todos os professores que nos acompanharam neste curso e que nos inspiraram, cada um de uma forma diferente, a sermos profissionais e principalmente seres humanos melhores, em particular aos professores Jean Menossi, Euclides Junior e Rodrigo Campos. Nossos agradecimentos especiais à nossa orientadora, Profa. Esp. Jane Maria dos Santos Eberson, por nos acompanhar e orientar neste trabalho, pelos conselhos e por sempre estar presente e disposta a fazer, junto a nós, com que este trabalho chegasse à sua conclusão.

Agradecemos a todos aqueles que contribuíram, de alguma forma, para a realização deste trabalho, e aos nossos caros colegas de curso, pelo companheirismo e troca de experiências que nos permitiram crescer pessoal e profissionalmente.

RESUMO

Redes sociais causaram uma disruptura na forma com que consumimos informações, uma vez que trouxeram para as pessoas novas formas de se comunicarem e trocarem informações. Na era da internet a desinformação em redes sociais têm crescido de maneira vertiginosa, pois, a informação encontra-se segmentada de forma que o indivíduo não consiga encontrá-la por completo em uma única fonte. Em uma instituição de ensino, observou-se uma dificuldade em relação a criação de relacionamentos entre os alunos de diferentes cursos, levando a pessoas que passam muitas vezes todo seu período na instituição, do início do curso até o fim, sem interagir com outras pessoas além das quais estão presentes em sua turma. Tendo em vista esse problema, esse trabalho teve como objetivo analisar e desenvolver uma rede social para uso interno a uma instituição de ensino, de maneira que a informação seja disponibilizada de forma simples e prática, além de, fazer os usuários interagirem entre si trocando experiências e compartilhando ideias. A metodologia utilizada foi a pesquisa bibliográfica, onde procurou-se entender quais as melhores formas de se produzir uma rede social e a pesquisa experimental, onde buscou-se elementos em redes sociais já existentes para o sistema a ser desenvolvido. Como resultado, a rede social foi desenvolvida utilizando as melhores práticas em desenvolvimento de sistemas e foi capaz de trazer um ambiente onde estudantes possam criar conexões entre si, divulgar suas experiências profissionais e pessoais enquanto discentes desta instituição de ensino. A partir do resultado, concluímos que o sistema desenvolvido, se implementado, poderá trazer benefícios para a instituição no que diz respeito ao engajamento entre os estudantes de diferentes cursos.

Palavras-chave: Comunicação; Desenvolvimento; Fatec; Informação; Rede social.

ABSTRACT

Social networks have disrupted the way we consume information, as they have brought people new ways of communicating and exchanging information. In the age of the internet, disinformation on social media platforms has been growing exponentially, as the information is segmented in such a way that the individual could not find it completely in an only source. In an educational institution, there was a difficulty in creating networking between students in different courses, leading people at this institution, from the beginning of the course to the end, without interaction with each other than those present in your class. In view of this problem, this work had as objective analyse and develop a social network for internal use in an educational institution, so that information is made available in a simple and practical way, in addition to making users interact with each other by exchanging experiences and sharing ideas. The methodology used was bibliographical research, where we sought to understand the best way to produce a social network and experimental research, where we sought elements in existing social networks for the system to be developed. As a result, the social network was developed using the best practices in systems development and was able to bring an environment where students can create connections with each other, share their professional and personal experiences as students of this educational institution. From the result, we can conclude that the developed system, if implemented, could bring benefits to the institution in terms of engagement between students from different courses.

Keywords: Communication; Development; Fatec; Information; Social network;

LISTA DE FIGURAS

Figura 1 - Gráfico do número de usuários de mídias sociais no mundo.....	12
Figura 2 - Diagrama de Caso de Uso - Interação dos usuários com o sistema	22
Figura 3 - Diagrama de Entidade-Relacionamento da rede social	23
Figura 4 - Desenho da Arquitetura Cliente – Servidor.....	24
Figura 5- Tela Inicial da rede social.....	25
Figura 6 - Continuação da tela inicial da rede social	26
Figura 7 – Tela de cadastro.....	27
Figura 8 - Tela do perfil do usuário.....	28
Figura 9 - Tela de configuração do Perfil.	29
Figura 10 - Tela de amigos.....	29
Figura 11 - Feed.....	30
Figura 12 - Chat	31

LISTA DE ABREVIATURAS E SIGLAS

API	<i>Application Programming Interface</i>
FATEC	Faculdade de tecnologia
FATEC GRU	Faculdade de tecnologia – Campus Guarulhos
FORTRAN	IBM Mathematical FORmula TRANslation System
HTML	<i>Hyper Text Markup Language</i>
IBM	International Business Machines
IoC	<i>Inversion of Control</i>
IU	Interface do Usuário
LGPD	Lei Geral de Proteção de Dados
LINQ	<i>Language Integrated Query</i>
LTS	<i>Long Term Support</i>
MVC	<i>Model-View-Controller</i>
OO	Orientado a Objetos
Perl	<i>Practical Extraction and Reporting Language</i>
PHP	<i>Hypertext Preprocessor</i>
POO	Programação Orientada a Objetos
SIGA	Sistema Integrado de Gestão Acadêmica
Simula-67	<i>Simulation Language</i>
SOLID	<i>Single-responsibility Principle Open-closed Principle Liskov Substitution Principle Interface Segregation Principle Dependency Inversion Principle</i>
SQL	<i>Structured Query Language</i>
STS	<i>Short Term Support</i>
OAuth	<i>Open-Authorization</i>
WWW	<i>World Wide Web</i>

SUMÁRIO

1. INTRODUÇÃO	9
2. FUNDAMENTAÇÃO TEÓRICA	10
2.1. REDES SOCIAIS	10
2.2. LINGUAGEM DE PROGRAMAÇÃO	12
2.3. ARQUITETURA DE SOFTWARE	16
2.4. BANCO DE DADOS.....	18
2.5. FRONTEND E BACKEND.....	19
3. METODOLOGIA.....	20
5. RESULTADOS E DISCUSSÕES	24
6. CONSIDERAÇÕES FINAIS	31
REFERÊNCIAS.....	33

1. INTRODUÇÃO

Observa-se na Faculdade de Tecnologia - Campus Guarulhos (FATEC GRU) certa dificuldade no acesso de informações institucionais e problemas de inter-relacionamentos entre discentes de cursos de graduações distintos. Redes sociais e canais de comunicação da instituição não conseguem atrair a uma parte do público interessado – os discentes – e, com isso, certas informações institucionais, muitas vezes, passam despercebidas pelos estudantes. Nota-se também a criação de bolhas estudantis quanto a relacionamentos entre discentes, pois, a partir do momento em que o estudante está inserido num grupo de indivíduos, não há incentivos para que ele crie relacionamentos com outros alunos de turmas, ou até cursos, diferentes.

O objetivo geral do trabalho é desenvolver um software para o ambiente *WEB* (*World Wide Web*) – de Rede Social, para integrar informações institucionais, conectar os discentes de todos os cursos disponíveis na Faculdade de Tecnologia (FATEC) a participarem mais de eventos institucionais, a estarem mais inteirados do que acontece na instituição, divulgar trabalhos, cursos livres, eventos, congressos e aumentar o *networking* entre os usuários, inicialmente focado na FATEC Guarulhos.

“A inserção em rede é determinante para o compartilhamento da informação e do conhecimento. Isto porque as redes são espaços valorizados para o compartilhamento da informação e para a construção do conhecimento.” (TOMAÉL; ALCARÁ; DI CHIARA, 2005, p. 93).

Para atingir o objetivo, o trabalho foi dividido em 4 partes. A primeira parte apresenta o referencial teórico sobre as tecnologias utilizadas durante o desenvolvimento do software e em cada subtópico, apresentamos as motivações para a escolha de cada tecnologia presente no projeto desenvolvido. A segunda parte apresenta o desenvolvimento do projeto, com a implementação dos módulos do sistema e o banco de dados. A terceira parte aborda a implementação do protótipo de software apresentado. E por fim, as considerações finais aborda os problemas encontrados durante o processo de desenvolvimento do software e as propostas para futuras melhorias do projeto.

2. FUNDAMENTAÇÃO TEÓRICA

Nesta seção, são apresentadas as definições e conceitos de elementos tecnológicos que contextualizam para com o tema, o problema e a solução proposta. O objetivo é introduzir os conceitos de cada elemento fundamental do projeto.

2.1. REDES SOCIAIS

Nos anos 2000, foram de suma importância para o desenvolvimento da computação, pois surgiu a *World Wide Web* (WWW) que rapidamente se estabeleceu como principal serviço da internet, a partir disso começaram a surgir novos modelos de negócio e blogs. A evolução cultural desse novo período chegou com as redes sociais, que se tornaram viáveis com maior conectividade entre as pessoas conquistadas nos anos 1990 (Wazlawick, 2021).

Ainda segundo Wazlawick (2021), a sociedade começou a sofrer mudanças causadas pela tecnologia nos anos 2000, com a disponibilização de *smartphones*, e a criação das redes sociais como o *Facebook*, Orkut, MSN e entre outras. A disseminação das redes sociais de maneira rápida e crescente só foram possíveis pela disponibilização de *smartphones* mais poderosos e baratos. Em 2004, Mark Zuckerberg cria sua própria rede social, a primeira rede social a atingir a marca de 1 bilhão de usuários. Inicialmente só alunos de Harvard poderiam ser membros do site, porém, com a fama da rede social logo alunos de outras universidades tiveram a permissão de participar. Para entrar no *Facebook* a pessoa teria que receber um convite de um usuário. Mas em setembro de 2006, a empresa decidiu que qualquer pessoa acima de 13 anos com um *e-mail* válido poderia criar um perfil na rede.

No mesmo ano, o Twitter foi lançado. Proporcionando um sistema simples para envio de mensagens curtas, na qual cresceu rapidamente. O que fez do Twitter um sucesso é o sistema em que as pessoas falam sobre assuntos políticos, comerciais e entretenimento, além de, seguir outros usuários da rede (Wazlawick, 2021).

Em 2009, outra rede social surgiu, o WhatsApp, o crescimento dessa rede se deu pelo fato de não ter nenhuma propaganda ou jogos, poder enviar mensagens curtas a contatos sem pagar nada a mais, como era o caso dos SMS. O aplicativo não exigia informações do usuário, era necessário apenas um número de telefone, tornou-

se tão popular que em 2013 atingiu a marca de aplicativo mais utilizado do mundo (Wazlawick, 2021).

As redes sociais mencionadas acima não apenas mudaram a forma como os indivíduos socializam, mas impactaram também produtos e serviços. O *Facebook* criou uma comunidade global, o WhatsApp liquidou com as mensagens tarifadas de SMS, o Youtube acabou com as videolocadoras. As redes estabeleceram a nova estrutura social em nossa sociedade globalizada, e a disseminação da mentalidade em redes tem causado alterações significativas nos processos de produção, experiência, poder e cultura. Apesar da organização social em redes já tenha existido em épocas e lugares distintos, a tecnologia da informação proporciona material para uma ampla expansão em todas as áreas da estrutura social (Wazlawick, 2021).

Essa evolução tecnológica levou à afirmação de Manuel Castells (2003), de que a nova sociedade é uma sociedade em rede, isso porque as redes sociais se tornaram o principal meio de comunicação das últimas duas décadas. Segundo o *Digital 2022 Global Overview Report* 62,5% da população mundial tem acesso à *internet* e as mídias sociais ficam em segundo lugar do *ranking* dos tipos de *websites* mais visitados e aplicações mais usadas na *internet*, ficando apenas atrás de *chats* e aplicativos de mensagem. Podemos observar na Figura 1 o crescimento do número de usuários de mídias sociais no mundo, esse gráfico demonstra como as pessoas estão cada vez mais conectadas as mídias sociais e como essas comunidades globais vêm crescendo de maneira exponencial, principalmente com a popularização da internet.

Figura 1 - Gráfico do número de usuários de mídias sociais no mundo.



Fonte: Adaptada de Digital 2022 Global Overview Report (2022)

As primeiras redes sociais surgiram antes da popularização da *internet* e eram desenvolvidas em formatos diferentes e com recursos limitados. Não há documentação sobre quais linguagens de programação foram usadas para desenvolver as primeiras redes sociais, mas as linguagens mais usadas em 1997, ano de lançamento da primeira rede social, chamada SixDegrees, eram PHP (*Hypertext Preprocessor*), Perl (*Practical Extraction and Reporting Language*) e HTML (*Hyper Text Markup Language*). Com os avanços da tecnologia e com o crescente acesso à *internet*, as redes sociais se tornaram uma plataforma poderosa para a disseminação de informações. O número de pessoas que recorrem a canais digitais (incluindo redes sociais) como fonte de informação é de 4 em cada 5 adultos (DataReportal, 2020). A popularização das redes sociais só foi possível devido às mudanças tecnológicas que contribuíram para moldar a maneira com que as pessoas se informam e interagem.

2.2. LINGUAGEM DE PROGRAMAÇÃO

Ada Byron, mais conhecida como Ada Lovelace, filha de Lord Byron e Annabella Milbanke, é considerada a primeira programadora da história. Ada publicou,

em 1843, uma tradução do artigo de Luigi Menabrea sobre a Máquina Analítica de Charles Babbage, a primeira máquina automática, de uso geral – ou seja, para a construção de outros programas – do mundo e durante o processo de tradução, adicionou diversas notas, onde descreveu os processos necessários para calcular uma série de figuras, chamadas de números de Bernoulli e, com isso, temos o primeiro programa idealizado (Kim e Toole, 1999).

Backus (1978) verificou que, antes de 1954, a programação era feita apenas no que conhecemos como programação de baixo nível, utilizando linguagem de máquina ou a linguagem *assembly*. Linguagens de programação de baixo nível fazem a comunicação de instruções diretamente para o processador de um computador e são chamadas assim, pois, possuem um baixo nível de abstração, ou seja, cada comando em uma linguagem de baixo nível representa apenas um comando. A linguagem de máquina é aquela na qual o computador é capaz de entender, e todo código de computador – independente da linguagem de programação – que necessita executar comandos, têm, em algum momento, sua escrita **traduzida** para a linguagem de máquina, que corresponde a uma linguagem binária (zeros e uns), na qual os processadores conseguem se comunicar, mesmo atualmente, um código precisa ser traduzido para linguagem de máquina para ser executado. A linguagem *assembly* foi a primeira linguagem a fazer um nível abstração da linguagem de máquina, e para executar seu código, é preciso utilizar ferramentas capazes de traduzir o código *assembly* para a linguagem de máquina, esse tipo de ferramenta é chamado de *Assembler*.

Backus (1978) observando a necessidade dos programadores da década de 1950, que consideravam seus trabalhos como complexo – por ter de lidar com linguagens de programação difíceis de entender, como são as linguagens de baixo nível – cria, em 1954, a primeira linguagem de programação de alto nível, o *FORmula TRANslation System* (FORTRAN), criado pela empresa *International Business Machines* (IBM), da qual o objetivo foi auxiliar os programadores na década de 50 a lidarem com problemas técnicos e científicos cuja solução requeria a utilização de computadores eletrônicos.

A programação de alto nível, como o próprio nome sugere, possui um nível de abstração alto, ou seja, em apenas um comando conseguem representar diversos comandos de uma linguagem de baixo nível (Bertolini et al., 2019). Para Ditzel e

Patterson (1998), as linguagens de alto nível possuem diversas justificativas de sua importância, que auxiliam os programadores, entre elas:

- Redução da complexidade da criação de compiladores;
- Redução do custo total de sistemas;
- Redução de custos de desenvolvimento de softwares;
- Redução de problemas semânticos entre o programador e a linguagem de máquina;
- Escrita de programas mais eficientes;
- Melhora na compactação do código.

A justificativa principal é que a criação de linguagens de programação com semânticas mais próximas das quais seres humanos estão acostumados a lidar diminui sua complexidade, aumenta o nível de compreensão dos programadores que, por sua vez, são capazes de construir programas mais eficientes, que executam mais tarefas consumindo menos recursos e com tempos de execução menores.

Para Khoshafian e Abnous (1990) a orientação a objetos surge para resolver um grande problema enfrentado nos anos 90: o surgimento do multitarefa e interface gráfica. Usuários demandavam cada vez mais por sistemas fáceis de compreensão e mais funcionalidades de seus computadores pessoais. Para atender tais necessidades, mais funcionalidades teriam de ser implementadas, e mais complexo um sistema se tornaria e assim, surge o paradigma da programação orientada a objetos.

Podemos descrever o conceito de orientação a objetos como modelagem e desenvolvimento de software que torna mais fácil a construção de sistemas complexos a partir de componentes individuais (Khoshafian, Abnous, 1990).

Ainda segundo Khoshafian e Abnous (1990) a orientação a objetos provém uma melhor abordagem e ferramentas para:

- Modelar o mundo real o mais próximo possível da perspectiva de usuário;
- Maior facilidade na interação com o ambiente computacional, com a utilização de metáforas familiares;
- Construção de componentes de softwares reutilizáveis;
- Maior facilidade na modificação e extensão na hora de implementar novos componentes, sem a necessidade de escrever todo o código novamente.

O principal fundamento da orientação a objetos é fornecer conceitos necessários para a representação do mundo real de maneira mais próxima possível. Existem diversas vantagens na utilização deste paradigma de programação, entre eles, a maior facilidade em representar mais diretamente modelos do mundo real no código do software e “[...] o resultado disto é que a transformação radical normal dos requisitos de sistema (definida em termos do usuário) para especificações de sistema (definida em termos do computador) é muito reduzida” (KHOSHAFIAN E ABNOUS, 1990, p. 7).

Simula-67 (Simulation Langage) é considerada a primeira linguagem de programação de alto nível que introduziu o conceito de orientação a objetos, os objetos tinham uma existência própria, ou seja, cada objeto criado tinha suas próprias características e podiam se comunicar entre si durante a simulação, processo pelo qual a linguagem chama o estado de execução de um código, de forma simplificada, o objeto contém dados e operações para manipular esses dados e a essas operações chamamos de métodos. O conceito de classe também foi introduzido pelo *Simula-67*, e que era utilizado a fim de descrever estruturas e comportamentos de objetos, outro ponto abordado pela linguagem foi o conceito de herança de classe, no qual as classes eram organizadas em hierarquias, possibilitando o compartilhamento de suas implementações e estrutura (Khoshafian, Abnous, 1990).

Após a introdução dos conceitos de Programação Orientada a Objetos (POO) feita pelo *Simula-67*, outras diversas linguagens surgiram implementando, cada uma com suas características, o conceito de POO como o C++, Java, Python, C# entre diversas outras. A linguagem de programação C#, lê-se C **sharp**, foi lançada nos anos 2000, como uma evolução da linguagem C++ – que foi a primeira da família C a implementar os conceitos de orientação a objetos. Inicialmente houve muitas críticas devido a sua semelhança com a linguagem Java, sendo considerado como cópia da linguagem Java pelos funcionários da SUN, empresa que criou o Java, e com todos os rumores o criador da linguagem, Anders Heljsberg, explicou que a linguagem sim, teve inspirações no Java, mas também em outras linguagens como *Modula 2* e C++ (Wazlawick, 2016).

O C# chegou para fazer parte da plataforma *.NET*, que é uma plataforma de desenvolvimento, de código fonte aberto e gratuito que permite a criação de vários tipos de aplicativos e “o *.NET* é criado em um *runtime* de alto desempenho que é usado em produção por muitos aplicativos de alta escala” (MICROSOFT, 2022).

Desde seu lançamento, a linguagem tem se tornado cada vez menos parecida com o Java e a cada versão, uma série de melhorias e adições são feitas para a linguagem, tornando-a moderna e atualizada.

No ano de 2022 a linguagem C# foi atualizada para a versão 9.0 e a plataforma .NET se encontra na versão 7 como sendo uma versão *Short Term Support* e o .NET 6 é a última versão *Long Term Service* do .NET até o ano de 2023 e a partir do .NET 6 se unificou o que era conhecido como .NET Core e .NET Framework em uma plataforma que fornece desenvolvimento simplificado, melhor performance, e uma produtividade para os desenvolvedores. Os termos *Short Term Support* (STS) e *Long Term Support* (LTS) referem-se ao tempo de suporte que a versão receberá da empresa que a mantém, sendo versões STS as que recebem atualizações por menos tempo e que geralmente recebem suporte por cerca de um ano e meio, enquanto versões LTS recebem suporte por cerca de três anos (MICROSOFT, 2023).

A linguagem C# é uma das linguagens mais difundidas no mercado, aparecendo muitas vezes no topo das linguagens mais utilizadas para programação, ela permite a criação dos mais diversos tipos de aplicativos, desde os mais simples, até os mais complexos e tudo isso equilibrando desempenho e segurança. Um dos motivos do bom desempenho da linguagem é seu coletor de lixo, ou *garbage collector*, que permite a recuperação automática de memória ocupada por objetos que não estão sendo utilizados e que estão inacessíveis em algum momento da execução da aplicação. Outra vantagem são as consultas LINQ (*Language Integrated Query*), que cria um padrão para trabalhar com dados de maneira muito mais eficiente dentro do código e é um recurso muito importante para manipulação de dados em bancos de dados *Structured Query Language* (SQL), (MICROSOFT, 2023).

2.3. ARQUITETURA DE SOFTWARE

Segundo Bass et al. (1998), arquitetura de software são estruturas que incluem componentes, suas propriedades externas e o relacionamento entre eles, que compreende o sistema.

De acordo com Varoto (2002, p.15), “a fase de engenharia de requisitos é um aporte para a definição da arquitetura, pois, modela o processo de negócio, realiza o

planejamento estratégico das versões e faz o levantamento de requisitos de cada uma delas”. Ainda, “a arquitetura deve ser construída objetivando o sistema como um todo, caso a arquitetura tenha foco nas funcionalidades a incorporação de mudanças ou novas funcionalidades pode exigir que a arquitetura seja refeita” Varoto (2002, p.16).

Surge então a importância da arquitetura limpa que representa decisões de um software realizado em seu desenvolvimento resultando em sua estrutura atual, como sua divisão de camadas e organização do código, com o propósito de facilitar seu desenvolvimento, implementação, operação e manutenção (MARTIN, 2019).

Para a arquitetura ser considerada como limpa, as decisões tomadas em estruturas devem facilitar os testes no software e ser independentes de estruturas, Interface de Usuário (IU), banco de dados e de quaisquer fatores externos, garantindo, assim, que todas as regras de negócio sejam imparciais as tecnologias escolhidas (MARTIN, 2009).

“Uma importante prática para garantir uma arquitetura limpa, é a refatoração, pois ela altera o código fonte, visando melhorar o entendimento e a manutenibilidade sem alterar suas funções externas” Barrozo et al. (2013, p.2).

A ideia de uma arquitetura bem estruturada e independente de fatores externos pode ser relacionada com a perspectiva de Christopher Alexander (1977) sobre padrões de arquitetura, segundo ele um padrão descreve um problema e sua solução, da forma com que essa solução possa ser utilizada por mais de uma vez, de várias maneiras diferentes.

Um padrão de projeto nomeia, abstrai e identifica os aspectos-chave de uma estrutura de projeto comum para torná-la útil para a criação de um projeto orientado a objetos reutilizável. O padrão de projeto identifica as classes e instâncias participantes, seus papéis, colaborações e a distribuição de responsabilidades. Cada padrão de projeto focaliza um problema ou tópico particular de projeto orientado a objetos. (GAMMA et al., 2000, p.20)

Ainda segundo Gamma et al. (2000), o padrão de arquitetura *Model-View-Controller* (MVC) separa a aplicação em três principais tipos: Modelo, Visão e Controlador. Essa arquitetura garante a visualização do estado dos modelos. A Visão deve refletir o estado do modelo, enquanto o Modelo é a representação do domínio, ou regra de negócio. Podem existir diversas visões para cada modelo e o Controlador – como o próprio nome sugere – controla o fluxo de processamento e acesso aos dados.

Já o padrão *Abstract Factory* é um padrão de criação que tem como objetivo “fornecer uma interface para criação de famílias de objetos relacionados ou dependentes sem especificar suas classes concretas” (GAMMA et al., 2000, p.95). Em outras palavras, fornecem uma abstração de processos de instanciação de classes.

“Um padrão de criação de classe usa a herança para variar a classe que é instanciada, enquanto um padrão de criação de objeto delegará a instanciação para outro objeto” (GAMMA et al., 2000, p.95)

A distinção de padrões de criação de objeto citado por Gamma pode ser relacionada com o trabalho de Robert C. Martin, que definiu um esboço do que seriam os princípios do modelo de orientação a objetos, idealizado como “os dez mandamentos da programação orientada a objetos”.

Esses princípios expõem os aspectos de gerenciamento de dependência do *design* orientado a objetos em oposição aos aspectos de conceituação e modelagem. Isso não quer dizer que a OO (Orientação a Objetos) seja uma ferramenta ruim para a conceituação do espaço do problema ou que não seja um bom local para a criação de modelos. Certamente muitas pessoas obtêm valor desses aspectos da OO. Os princípios, no entanto, concentram-se fortemente no gerenciamento de dependências (MARTIN, 2005, n.p.).

Alguns desses princípios ficaram conhecidos pelo acrônimo de SOLID (*Single-responsibility Principle Open-closed Principle Liskov Substitution Principle Interface Segregation Principle Dependency Inversion Principle*), que são designados principalmente ao *design* de classes (TROQUETE, 2019).

2.4. BANCO DE DADOS

Segundo Date (2004), “um sistema de banco de dados é um sistema computadorizado de manutenção de registros, ou seja, ele é um repositório ou recipiente para uma coleção de arquivos de dados computadorizados”.

Dentro do sistema o usuário pode solicitar que o mesmo realize operações envolvendo arquivos, como: acrescentar arquivos ao banco de dados, inserir dados em arquivos existentes, buscar dados de arquivos, alterar dados de arquivos e remover arquivos (DATE, 2004).

SQL significa *Structured Query Language* – ou Linguagem de Consulta Estruturada. Prescott (2015) define SQL como uma linguagem de consulta de interação com banco de dados relacionais. A linguagem SQL é utilizada para manipular dados em um banco de dados, como a inclusão, leitura, alteração e remoção dos dados. O *SQL Server* é de propriedade da Microsoft e é uma implementação da linguagem de banco de dados SQL que trabalha em conjunto com o *Transact SQL* para fornecer funcionalidades de acesso a banco de dados para aplicações.

Por outro lado, a Inversão de Controle (IoC), foi originalmente proposto por Ralph Johnson e Brian Foote (Johnson & Foote, 1996), e desempenha um papel fundamental nos frameworks utilizados no desenvolvimento de software. A IoC possui como principal objetivo permitir que desenvolvedores de software se concentrem na especificação e codificação lógica de negócio, deixando o controle do fluxo de ações executadas pelo sistema a cargo dos *frameworks* utilizados no processo de desenvolvimento.

2.5. FRONTEND E BACKEND

O *frontend* é a interface gráfica usada por usuários, que interagem atribuindo dados através de botões, programas, sites e outros recursos. Esses recursos são projetados para serem acessíveis, agradáveis e fácil de usar (Souza et al., 2022)

Por outro lado, o *backend* refere-se a parte de um código de um programa que permite que ele funcione, nele está a maioria dos dados e da sintaxe operacional armazenada de um software. É chamada de camada de acesso a dados de software qualquer funcionalidade que precise ser acessada e navegada por meios digitais. Os usuários não conseguem ter acesso e nem interagir com a camada de acesso a dados (Souza et al.,2022).

Uma maneira de gerenciar o acesso a recursos protegidos no *backend* é através do protocolo *OAuth* que permite que um usuário conceda acesso limitado a seus recursos protegidos através do servidor de recursos que emite *tokens* de entrada para acessar os recursos protegidos e hospedados pelo servidor. Esse protocolo separa a função do cliente do proprietário do recurso (Microsoft, 2023).

3. METODOLOGIA

O trabalho está estruturado em duas vertentes: a pesquisa bibliográfica e a pesquisa experimental. Conforme Gil (2007, p.44) a pesquisa bibliográfica explica os exemplos mais característicos desse tipo de pesquisa que são as investigações sobre ideologias ou aquelas que se propõe à análise das diversas posições acerca de um problema. É a busca do conhecimento de temas específicos, nesse projeto foi feita a busca de materiais que apontaram as ferramentas nas quais seriam pertinentes ao desenvolvido de um software para o ambiente WEB.

Conforme Gil (2007), a pesquisa experimental objetiva selecionar as variáveis que seriam capazes de influenciar o objetivo. A pesquisa experimental foi realizada seguindo parâmetros de redes sociais já existentes e aplicando o que funciona nelas, e personalizando para uma instituição de ensino.

4. DESENVOLVIMENTO

Observou-se que na FATEC GRU existem problemas de falta de engajamento dos discentes em relação aos eventos da instituição, as informações dos cursos, festas e materiais, que se encontram dispersos e estes problemas são as motivações para o desenvolvimento desse projeto.

A princípio, o projeto foi desenvolvido em C# com base no padrão MVC, mas durante o processo de desenvolvimento, foi identificada a necessidade de refatoração do código, pois, o padrão MVC não seria escalável o suficiente em caso de aumento do número de usuários ou recursos do projeto. Assim, foi decidido adotar a arquitetura limpa, que segue os conceitos SOLID e que consiste em dividir o projeto em vários projetos menores.

Na arquitetura limpa, há uma camada de domínio para gerenciar o banco de dados e uma camada de infraestrutura para gerenciar o acesso aos dados e a conexão com o banco de dados. Para garantir que as camadas mais internas não dependam das camadas mais externas, foi utilizado o princípio da inversão de dependências, implementado por meio de um projeto para a inversão de controle (IoC).

A camada de aplicação é responsável por intermediar a comunicação entre a camada de *backend* e *frontend*. Quando a camada de *frontend* solicita informações

do banco de dados, o controlador solicita essas informações à aplicação, que, por sua vez, solicita à camada de infraestrutura. A camada de infraestrutura retorna os dados necessários para a aplicação, que, em seguida, os repassa para a camada de *frontend*.

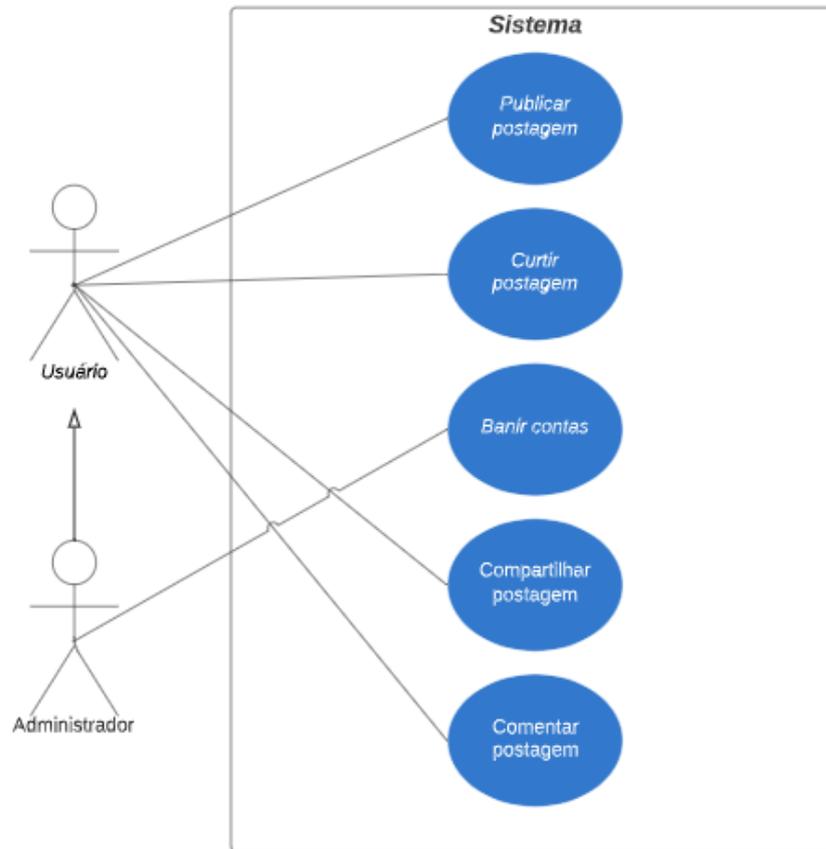
Para entrar no sistema, o usuário deve se autenticar, e ao invés de criar e gerenciar usuários no projeto, a autenticação dos usuários será feita através do Microsoft OAuth 2.0. Ao fazer *login*, o usuário é redirecionado para a plataforma da Microsoft, onde realiza a autenticação através da sua conta institucional. Só serão criados perfis de usuários que tenham o *e-mail* institucional da FATEC. A Microsoft valida a autenticidade do usuário e envia um *token* para o sistema, que é utilizado para verificar a autenticidade em cada acesso subsequente. O sistema armazena um registro para verificar se é o primeiro acesso do usuário e, em caso positivo, redireciona para uma página de criação de perfil personalizado. Dessa forma, o sistema não precisa lidar com a criação e exclusão de usuários, tornando o processo mais simples e seguro.

O perfil do usuário terá um banco de dados associado ao registro de usuários, mas ele contém apenas as informações fornecidas pela Microsoft, e não as informações fornecidas pelos próprios usuários. Dentro do perfil, há um *menu* de configurações que permite que os usuários alterem suas informações de perfil, bem como adicionar amigos.

Existe uma seção no sistema onde os usuários podem fazer publicações, incluindo fotos, textos ou vídeos, além disso, é possível interagir com as publicações de outros usuários.

Há uma diferença entre a conta de um usuário aluno e o usuário administrador. O usuário administrador pode banir contas, mas para ter o usuário administrador deve ter um vínculo empregatício com o Centro Paula Souza. Quando um usuário for banido, o sistema deverá permitir que ele tenha acesso às publicações e comentários, no entanto, será impedido de interagir na rede. O sistema deve permitir que o usuário exclua suas próprias publicações e comentários. Na figura 2 podemos observar as ações que os usuários podem realizar na rede e qual o privilégio do administrador.

Figura 2 - Diagrama de Caso de Uso - Interação dos usuários com o sistema



Fonte: elaborado pelos autores (2023)

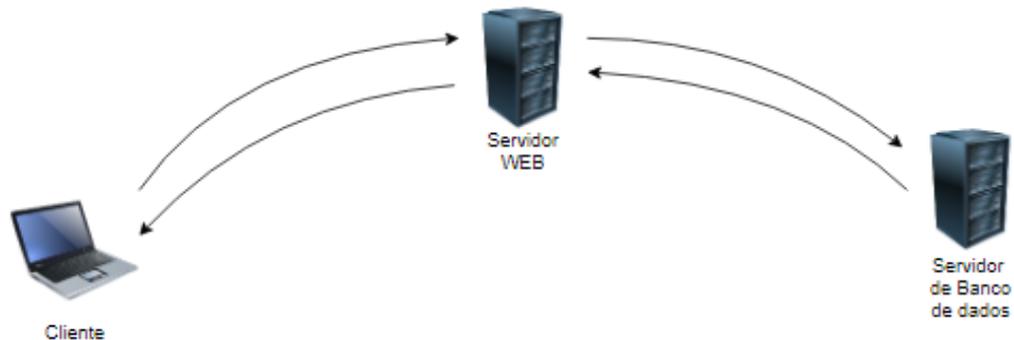
A cada seis meses o banco de dados apaga todos os dados de publicações, mas não exclui perfis de usuários. Os perfis só são excluídos quando o usuário não tiver mais o vínculo com a instituição.

Os requisitos de qualidade estabelecem o que os usuários esperam da solução proposta em termos de qualidade. Já os requisitos de funcionalidade estabelecem a conformidade de acordo com a Lei Geral de Proteção de Dados (LGPD), Lei nº 13.709/2018, a segurança de acesso no que diz respeito ao grande porte, o ambiente é estável e seguro para garantir o acesso restrito aos dados, tendo a infraestrutura de segurança adotada no ambiente e a disponibilidade recomendada de 24 horas por dia, inclusive aos sábados, domingos e feriados.

As informações internas não poderão ser compartilhadas em outras redes sociais. Não serão realizados cadastros de usuários que não tenham matrícula ativa

São utilizados servidores de aplicação e banco de dados, como podemos observar na figura 4. Com relação ao controle de acesso, apenas os administradores terão acesso à infraestrutura.

Figura 4 - Desenho da Arquitetura Cliente – Servidor.



Fonte: elaborado pelo autor (2023)

A qualidade do software será avaliada através de testes aplicados de maneira isolada. Serão aplicadas duas abordagens diferentes o teste caixa branca e o teste caixa preta. O teste caixa branca visa testar o código do sistema, e analisar internamente a estrutura do software, diferente do teste caixa preta que só testará o software do ponto de vista do usuário. Serão verificados usando um *check list* para conferir se todas as funções descritas na documentação conferem com as funções que estão no software.

5. RESULTADOS E DISCUSSÕES

O resultado foi uma rede social funcional, na qual foi possível implementar todas as funcionalidades necessárias para que os usuários possam interagir, curtir, publicar e postar conteúdos na plataforma. Além de poder se conectar com os outros usuários.

O logo do sistema é representado por SI³, pois, significa Sistema Integrado de Informação Interna.

Nas figuras 5 e 6 podemos observar qual será a primeira tela da rede social que o usuário terá contato. Na figura 5, observamos que há três ícones, além do logo do sistema, o primeiro é representado por um ponto de interrogação, na qual fala sobre os desenvolvedores do projeto, no segundo ícone representado por um **i**, é como os usuários poderão entrar em contato com os administradores, por último se encontra botão com a palavra **entrar**, ao clicar no botão o usuário será direcionado para a tela de autenticação da Microsoft e então ao cadastro de usuário.

Figura 5 - Tela Inicial da rede social.



Fonte: elaborado pelo autor (2023)

Figura 6 - Continuação da tela inicial da rede social



Fonte: elaborado pelo autor (2023)

Na figura 7 é apresentada a tela de cadastro, onde o usuário irá completar as informações do seu perfil como nome, sobrenome, data de nascimento, telefone, sexo, curso e semestre em que se encontra. As outras informações são fornecidas pela Microsoft.

Figura 7 – Tela de cadastro

SI³

Ajude-nos Sobre nós Contato

Entrar

Olá, Kevin!

Notamos que este é seu primeiro acesso.

Para acessar a plataforma, preencha alguns dados pessoais primeiro.

Nome*

João Da Silva

Primeiro nome Último nome

Email**

kevin.silva15@fatec.sp.gov.br

Telefone Data de nascimento*

11 90000-0000

Curso* Semestre Atual*

Selecione

Sexo*

Selecione

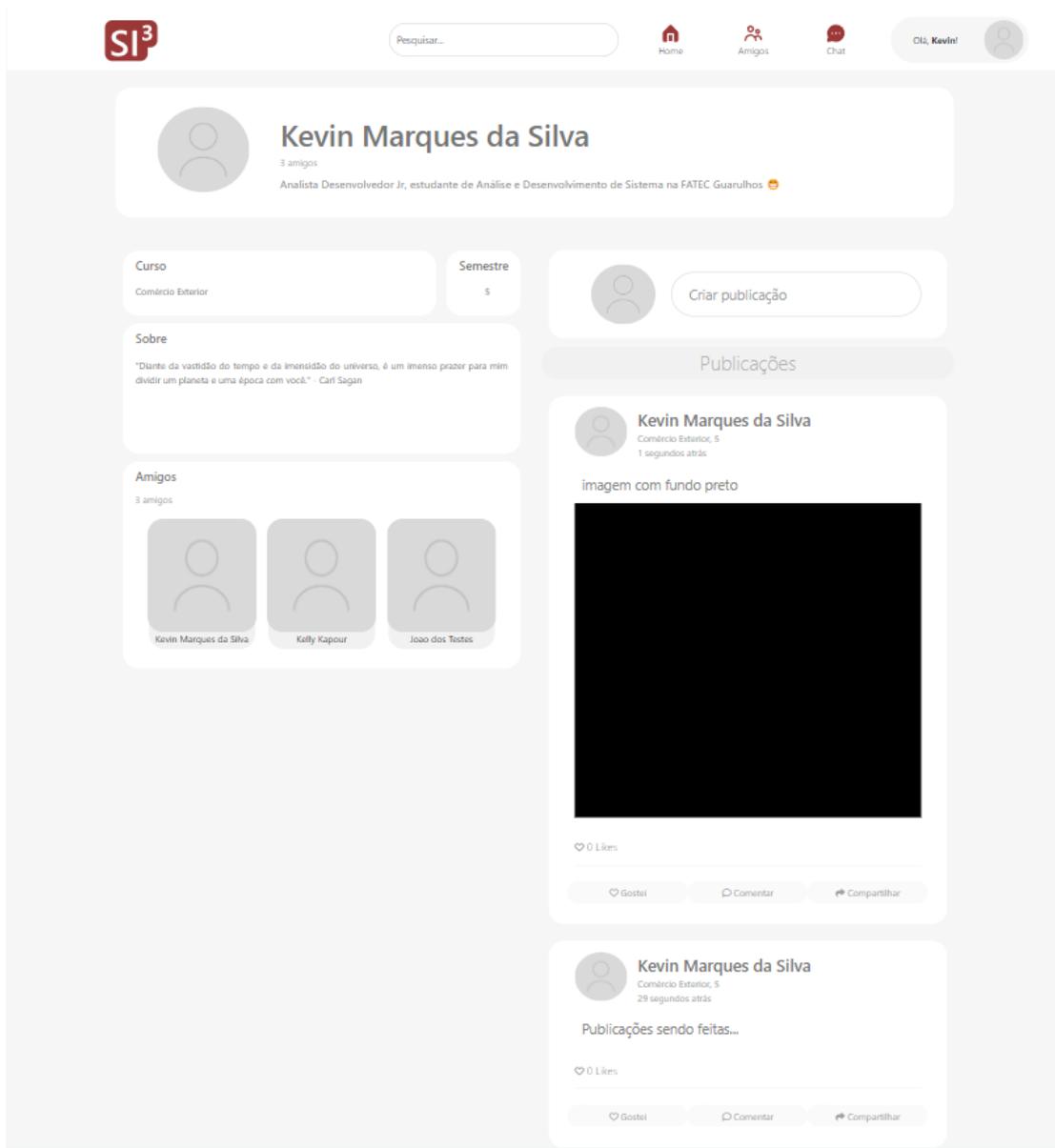
Continuar

* Obrigatório
** Não editável

Fonte: elaborado pelos autores (2023).

A figura 8 mostra o perfil do usuário onde se encontra as informações de quais perfis o usuário está conectado, ou seja, seus amigos, suas publicações, um espaço no qual o usuário pode falar sobre si mesmo, o semestre e o curso que está matriculado. Nessa tela ainda é possível ver um ícone de balão onde os usuários usam para se comunicar, ícone de amigos que ao clicar pode mandar ou aceitar solicitação de amizade e o ícone de *home* que seria o *feed* do usuário.

Figura 8 - Tela do perfil do usuário.



Fonte: elaborado pelos autores (2023).

Na figura 9 apresenta a tela de configurações de perfil do usuário, onde pode ser modificado os campos como nome, sobrenome, telefone, data de nascimento, sexo, curso, semestre atual, biografia e sobre.

Figura 9 - Tela de configuração do Perfil.

SI³ Pesquisar... Home Amigos Chat Olá, Kevin!

Olá, Kevin!
Essas são as informações de seu perfil.
Para alterar, clique no campo e coloque a informação desejada.

Primeiro Nome* Sobrenome
Kevin Silva

Email
kevin.silva15@fatec.sp.gov.br

Telefone Data de Nascimento* Sexo*
11 11978488265 02/12/2001 Masculino

Curso Semestre Atual*
Análise e Desenvolvimento de Sistemas 6

Biografia
Teste

Sobre
123

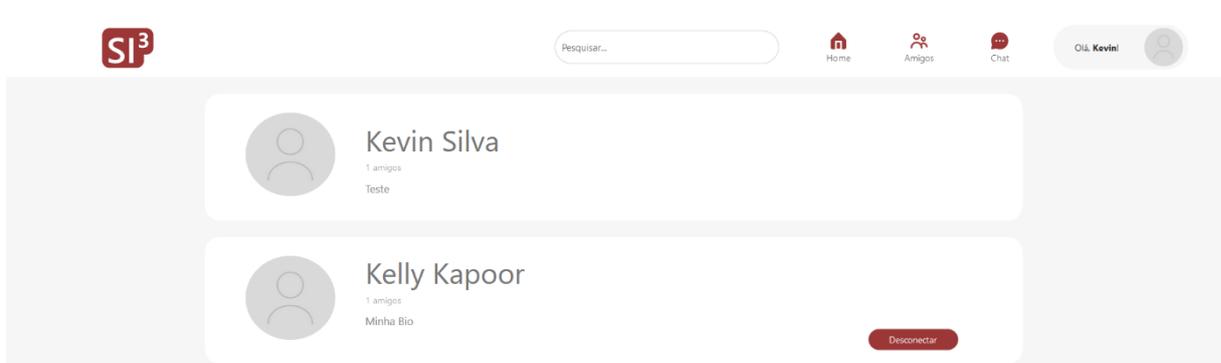
Legenda
* Obrigatório

Continuar

Fonte: elaborado pelos autores (2023).

A figura 10 mostra como é a tela de busca de usuários, onde o usuário pode se conectar com outros usuários da plataforma.

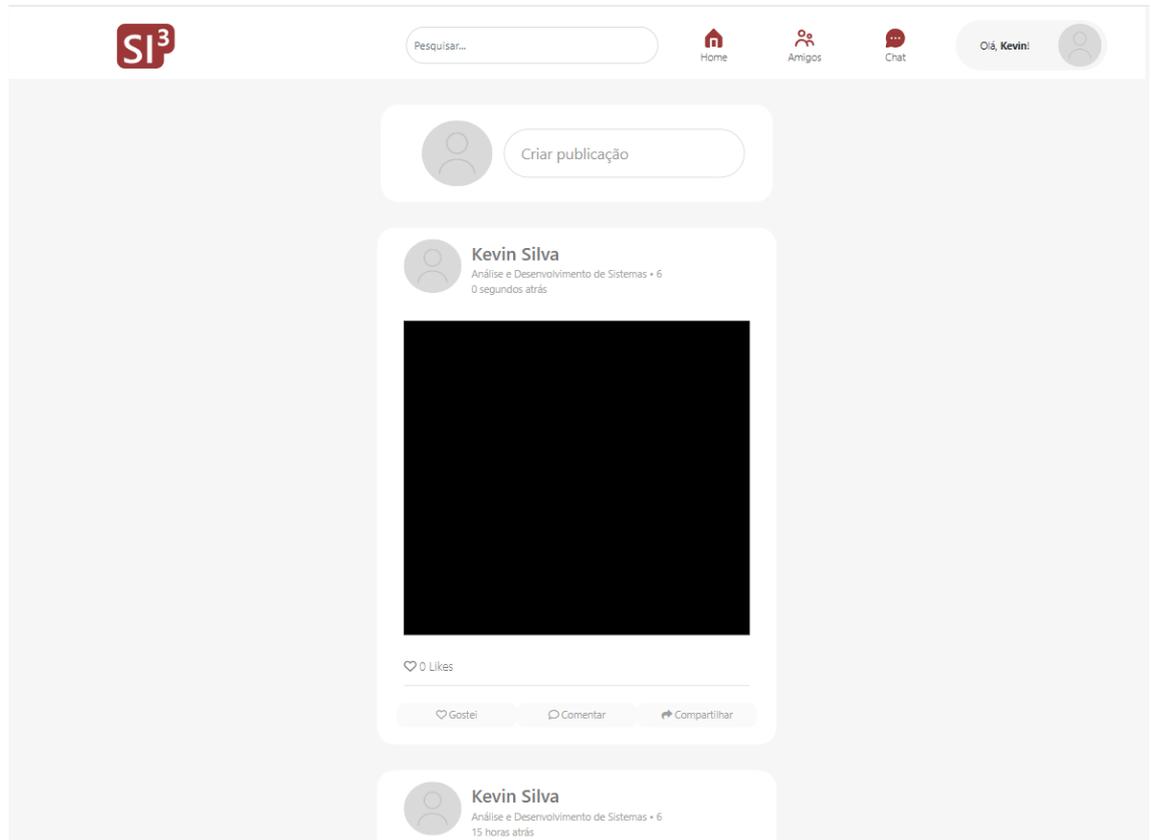
Figura 10 - Tela de amigos



Fonte: elaborado pelos autores (2023).

A figura 11 mostra o *feed* do usuário, que será composto pelas publicações do próprio usuário e da rede dele.

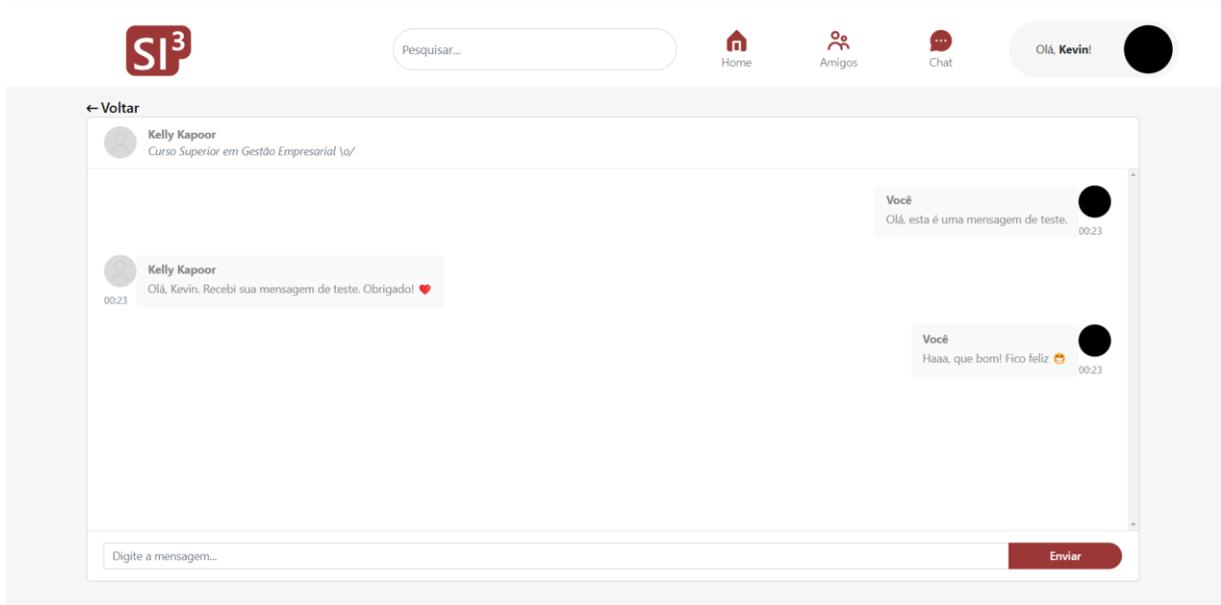
Figura 11 - *Feed*



Fonte: elaborado pelos autores (2023).

A figura 12 apresenta a tela do *chat*, local onde é feita a comunicação por mensagens direto com outro usuário da rede social.

Figura 12 - Chat



Fonte: elaborado pelos autores (2023).

As funcionalidades implementadas no sistema estão funcionais. A implantação do sistema poderia ajudar a instituição em relação aos problemas que se observou na FATEC GRU em relação a falta de engajamento dos discentes aos eventos da instituição, as informações dos cursos, festas e materiais, que se encontram dispersos, já que, segundo o *Digital 2022 Global Overview Report* pessoas passam em média 2 horas e 27 minutos nas plataformas de redes sociais.

6. CONSIDERAÇÕES FINAIS

O objetivo do trabalho é melhorar a comunicação dos discentes, colaboradores administrativos e docentes da FATEC. Para tanto é apresentado no corpo do texto os elementos fundamentais para a construção da rede social. A metodologia utilizada foi a pesquisa bibliográfica e a pesquisa experimental.

O código fonte desse projeto ficará aberto para que os outros estudantes possam melhorar a rede e readequá-la conforme a necessidade da instituição.

Sugere-se que para trabalhos futuros seja pensado em uma integração desta rede social com o SIGA (Sistema Integrado de Gestão Acadêmica), a fim de obter dados para completar o perfil do aluno, dados esses que seriam visíveis apenas para o proprietário do perfil.

Recomenda-se para trabalhos futuros o uso de Inteligência Artificial como moderador das publicações de usuários para evitar conteúdos inadequados que possam ofender a comunidade ou a grupos minoritários.

REFERÊNCIAS

BARROZO, Gracielle Castilho; VINHAS, Hingridi Marques; REIS, José Cláudio de Souza. Refatoração: aperfeiçoando um código existente. In: **RE3C – Revista Eletrônica Científica de Ciência da Computação**, v. 7 n. 1 (2012), Artigos. Disponível em: <https://revistas.unifenas.br/index.php/RE3C/article/view/12>. Acesso em: 27 maio 2023.

BASS, Len; CLEMENTS, Paul; KAZMAN, Rick. **Software Architecture in Practice**. 3rd ed. Boston: Addison Wesley, 1998.

BERTOLINI, Cristiano et al. **Linguagem de Programação I**. 1. ed. Rio Grande do Sul: [s. n.], 2019. 114 p. ISBN 9788583412465. Ebook.

BRASIL. **Lei nº 13.709, de 14 de agosto de 2018**. Dispõe sobre a proteção de dados pessoais e altera a Lei nº 12.965, de 23 de abril de 2014 (Marco Civil da Internet). Diário Oficial da União, Brasília, DF, 15 ago. 2018. Disponível em: <https://www.planalto.gov.br/ccivil_03/_ato2015-2018/2018/lei/l13709.htm> Acesso em: 25 maio de 2023.

DataReportal. (2020). **More than half of the people on earth now use social media**. We Are Social. Disponível em: <https://wearesocial.com/us/blog/2020/07/more-than-half-of-the-people-on-earth-now-use-social-media/>. Acesso em: 24 maio 2022.

DATE, C. J. **Introdução a sistemas de bancos de dados**. Tradução: Daniel Vieira. 8. ed. Rio de Janeiro: Campus, 2004. 865 p. ISBN 9788535212730.

DITZEL, David R.; PATTERSON, David A. **Retrospective on high-level language computer architecture**. ISCA '80: Proceedings of the 7th annual symposium on Computer Architecture, [s. l.], p. 97-104, 6 maio 1980.

EUA (EUA). Microsoft Corporation. Conceitos básicos do .NET: Novidades no .NET 6. In: EUA (EUA). Microsoft Corporation. **Conceitos básicos do .NET: Novidades no .NET 6**. 1. 1. ed. EUA: Microsoft Corporation, 5 out. 2022. Disponível em: <https://learn.microsoft.com/pt-br/dotnet/core/whats-new/dotnet-6>. Acesso em: 28 out. 2022.

GAMMA, Erich *et al.* **Padrões de projetos: soluções reutilizáveis de software orientado a objetos**. 1. ed. Porto Alegre: Bookman, 2000. 368 p. v. 1. ISBN 0 - 201-63361- 2.

GRANDI, Gilberto. **Metodologia Para Especificação de Sistemas em Ambiente Cliente/Servidor Orientada a Objetos**. Orientador: Elizabeth S. Specialski, M Sc. 1996. 145 p. Dissertação (Curso de Pós-Graduação em Ciências da Computação) - Universidade Federal de Santa Catarina - UFSC, FLORIANÓPOLIS, 1996. Disponível em: <https://repositorio.ufsc.br/xmlui/handle/123456789/158069>. Acesso em: 31 maio 2023.

HISTORY OF PROGRAMMING LANGUAGES CONFERENCE, 1978, Los Angeles, CA. **History of programming languages** [...]. New York: Academic Press, 1981. 758 p. Tema: The history of Fortran I, II, and III.

KHOSHAFIAN, Setrag; ABNOUS, Razmik. **Object Orientation: Concepts, Languages, Databases, User Interface: Concepts, Languages, Data Bases, User Interfaces**. 1. ed. New York; Wiley: John Wiley & Sons, 1990. 448 p. ISBN 0471518026 9780471518020 0471518018 9780471518013.

KIM, Eugene Eric; TOOLE, Betty Alexandra. **Ada and the First Computer**. Scientific American, USA, v. 280, n. 5, p. 76-81, maio 1999.

MARTIN, Robert C. **Código limpo: Habilidades práticas do Agile Software**. 1. ed. [S. l.]: Alta Books, 2009. 425 p. ISBN 978-8576082675.

MARTIN, Robert C. The Principles of OOD. *In: Principles of OOD*. [S. l.], 11 maio 2005. Disponível em: <http://butunclebob.com/ArticleS.UncleBob.PrinciplesOfOod>. Acesso em: 10 jun. 2023. MARTIN, Robert C. Principles of OOD. **Principles of OOD.**, [s. l.], 11 maio 2005. Disponível em: <http://butunclebob.com/ArticleS.UncleBob.PrinciplesOfOod>. Acesso em: 27 maio 2023.

MICROSOFT (EUA). Autenticação OAuth 2.0 com o Azure Active Directory. *In: Autenticação OAuth 2.0 com o Azure Active Directory*. [S. l.], 20 mar. 2023. Disponível em: <https://learn.microsoft.com/pt-br/azure/active-directory/fundamentals/auth-oauth2>. Acesso em: 31 maio 2023.

MICROSOFT CORPORATION (EUA). Conceitos básicos do .NET: O que é o .NET? Introdução e visão geral. *In: EUA (EUA). Microsoft Corporation. Conceitos básicos do .NET: O que é o .NET? Introdução e visão geral*. 1. 1. ed. EUA: Microsoft Corporation, 5 out. 2022. Disponível em: <https://learn.microsoft.com/pt-br/dotnet/core/introduction>. Acesso em: 28 out. 2022.

MICROSOFT. Documentação do C#. *In: MICROSOFT. Um tour pela linguagem C#*. [S. l.], 15 fev. 2023. Disponível em: <https://dotnet.microsoft.com/pt-br/platform/support/policy>. Acesso em: 22 maio 2023.

MICROSOFT. **Políticas de suporte para as várias tecnologias que compõem a plataforma .NET**. [S. l.], [ca. 2023]. Disponível em: <https://dotnet.microsoft.com/pt-br/platform/support/policy>. Acesso em: 3 maio 2023.

PRESCOTT, Preston. **SQL para Iniciantes**. 1. ed. [S. l.]: Babelcube Inc., 2015. 30 p. v. 1. ISBN 1507113374, 9781507113370. Disponível em: https://books.google.com.br/booksid=q64cCgAAQBAJ&lpg=PT4&ots=7gtKUn1mjH&dq=sq&lr=lang_pt&hl=pt-BR&pg=PT4#v=onepage&q&f=false. Acesso em: 28 out. 2022.

ROBERTS, D.; JOHNSON, R. **Evolving Frameworks: A Pattern Language for Developing Object-Oriented Frameworks**. *In: Proceedings of Pattern Languages of Programs*. Illinois, 1996. Disponível em: <https://www.semanticscholar.org/paper/Evolving-Frameworks-A-Pattern-Language->

for-Roberts-Johnson/726529e42b243295f902d2297e52b8183e01a2b4. Acesso em: 31 mai. 2023.

SOUZA, Francisco Moreira Calado; LIMA, Edilson Carlos Silva; CARIDADE, Elda Regina de Sena. **Criando sistema escalável de agendamentos utilizando typescript com nestJS no backend e nextJS no frontend**, São Paulo, v. 8, n. 12, p. 43 - 57, 30 dez. 2022. DOI 10.51891/rease.v8i12.7986. Disponível em: <https://www.semanticscholar.org/paper/CRIANDO-SISTEMA-ESCAL%C3%81VEL-DE-AGENDAMENTOS-COM-NO-E-Souza-Lima/5a1babf13419bb95c8f5b3917afe079d4c3ae61c>. Acesso em: 31 maio 2023.

Tomaél, Maria Inês, Alcará, Adriana Rosecler e Di Chiara, Ivone Guerreiro. **Das redes sociais à inovação**. Ciência da Informação [online]. 2005, v. 34, n. 2 [Acessado 4 outubro 2022], pp. 93-104. Disponível em: <<https://doi.org/10.1590/S0100-19652005000200010>>. Epub 12 Abr 2006. ISSN 1518-8353. <https://doi.org/10.1590/S0100-19652005000200010>.

TROQUETE, Rafael. Monografias Lato Sensu: Especialização e MBA. In: TROQUETE, Rafael. **Programação orientada a objetos**: uma visão conceitual dos elementos de modelagem. Orientador: Daniel Couto Gatti. 2019. Monografia de especialização (Especialista em Engenharia de Software) - PUC-SP, Brasil, 2019. p. 52. Disponível em: <https://repositorio.pucsp.br/jspui/handle/handle/26820>. Acesso em: 27 maio 2023.

VAROTO, Ane Cristina. **Visões em arquitetura de software**. 2002. 144 f. Dissertação (Mestrado em Ciência da Computação) - Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2002. Disponível em: <https://www.teses.usp.br/teses/disponiveis/45/45134/tde-20220712-120029/publico/VarotoAneCristina.pdf>. Acesso em: 25 maio de 2023.

WAZLAWICK, Raul Sidnei. **História da Computação**. 1. Ed. [S. I.]: GEN LTC, 2021. 581 p. ISBN 8535285458.