

**CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA**  
**FACULDADE DE TECNOLOGIA DE MAUÁ**  
**CURSO SUPERIOR DE TECNOLOGIA EM DESENVOLVIMENTO DE**  
**SOFTWARE MULTIPLATAFORMA**

**GABRIEL MESQUITA DE PAULA**  
**HIAGO GABRIEL OLIVEIRA PINTO**

**APPFATEC: SISTEMA MULTIPLATAFORMA DE GESTÃO**  
**ACADÊMICA MÓVEL INTEGRADO COM INTELIGÊNCIA**  
**ARTIFICIAL**

MAUÁ  
2026

**GABRIEL MESQUITA DE PAULA**  
**HIAGO GABRIEL OLIVEIRA PINTO**

**APPFATEC: SISTEMA MULTIPLATAFORMA DE GESTÃO  
ACADÊMICA MÓVEL INTEGRADO COM INTELIGÊNCIA  
ARTIFICIAL**

Relatório técnico-científico apresentado à  
Faculdade de Tecnologia de Mauá, como  
exigência parcial para obtenção do título de  
Tecnólogo em Desenvolvimento de Software  
Multiplataforma.

Orientadora: Profa. Me. Luana Lourenço.

MAUÁ  
2026

## RESUMO

O presente relatório técnico-científico analisa o desenvolvimento do AppFatec, um sistema de gestão acadêmica multiplataforma voltado à melhoria do acesso às informações discentes por meio de dispositivos móveis, caracterizando-se como pesquisa aplicada de abordagem exploratória centrada na identificação de limitações dos sistemas acadêmicos tradicionais e na proposição de uma solução tecnológica mais eficiente, responsiva e acessível. A metodologia adotada envolveu tecnologias modernas — React Native no frontend e Node.js no backend —, testes de software para validação de desempenho, segurança e funcionalidade, análises de acessibilidade fundamentadas nas diretrizes WCAG 2.1 e pesquisa de campo com 94 discentes da Faculdade de Tecnologia de Mauá para embasar empiricamente as decisões de design e funcionalidade. Os resultados demonstram boa performance, interface intuitiva e melhorias significativas na experiência do usuário em comparação aos modelos convencionais, evidenciando que a adoção de arquiteturas modernas e de práticas de desenvolvimento centradas no usuário contribui de forma relevante para a criação de soluções mais eficientes e acessíveis no contexto educacional.

**Palavras-chave:** desenvolvimento de software; acessibilidade; aplicativos móveis; gestão acadêmica; experiência do usuário.

## SUMÁRIO

|                                                                      |    |
|----------------------------------------------------------------------|----|
| 1 INTRODUÇÃO .....                                                   | 6  |
| 1.1 Contextualização .....                                           | 6  |
| 1.2 Problema de pesquisa .....                                       | 6  |
| 1.3 Objetivos.....                                                   | 6  |
| 1.3.1 Objetivo Geral .....                                           | 6  |
| 1.3.2 Objetivos Específicos .....                                    | 7  |
| 1.4 Justificativa.....                                               | 8  |
| 1.5 Estrutura do relatório .....                                     | 8  |
| 2 DESENVOLVIMENTO.....                                               | 9  |
| 2.1 Fundamentação teórica .....                                      | 9  |
| 2.1.1 Revisão de literatura .....                                    | 9  |
| 2.1.2 Técnicas de desenvolvimento de software .....                  | 10 |
| 2.1.3 Linguagens de programação e frameworks .....                   | 10 |
| 2.1.4 Metodologias de teste de software .....                        | 11 |
| 2.2 Metodologia.....                                                 | 11 |
| 2.2.1 Tipo de pesquisa .....                                         | 11 |
| 2.2.2 Método de coleta de dados .....                                | 12 |
| 2.2.3 Procedimentos de análise.....                                  | 12 |
| 2.2.4 Critérios de avaliação .....                                   | 12 |
| 2.2.5 Pesquisa de campo com discentes .....                          | 13 |
| Perfil dos respondentes .....                                        | 13 |
| Avaliação do sistema SIGA vigente .....                              | 14 |
| Demandas e expectativas para um aplicativo móvel acadêmico .....     | 16 |
| 2.3 Ficha técnica .....                                              | 19 |
| 2.4 Técnicas e testes de software e avaliação da acessibilidade..... | 19 |
| 2.4.1 Testes unitários .....                                         | 19 |

|                                                                    |    |
|--------------------------------------------------------------------|----|
| 2.4.2 Testes de integração.....                                    | 20 |
| 2.4.3 Testes de performance .....                                  | 20 |
| 2.4.4 Testes de segurança .....                                    | 21 |
| 2.4.5 Metodologias de desenvolvimento e qualidade.....             | 21 |
| 2.4.6 Avaliação de acessibilidade.....                             | 21 |
| 2.5 Funcionalidades complementares implementadas.....              | 22 |
| 2.5.1 Autenticação biométrica por hardware.....                    | 23 |
| 2.5.2 Suporte offline e persistência de cache.....                 | 24 |
| 2.5.3 Sistema de mensagens internas (aluno ↔ professor) .....      | 25 |
| 2.5.4 Notificações push e gerenciamento de tokens.....             | 25 |
| 2.5.5 Calendário acadêmico interativo .....                        | 26 |
| 2.5.6 Interface responsiva adaptativa (Bottom Tabs e Sidebar)..... | 26 |
| 2.5.7 Painel de detalhes da disciplina.....                        | 27 |
| 2.6 Interfaces gráficas do sistema.....                            | 27 |
| 3 CONSIDERAÇÕES FINAIS .....                                       | 34 |
| 3.1 Principais conclusões.....                                     | 34 |
| 3.2 Recomendações .....                                            | 34 |
| 3.3 Limitações do estudo .....                                     | 34 |
| 3.4 Perspectivas futuras .....                                     | 35 |
| REFERÊNCIAS .....                                                  | 36 |
| APÊNDICE A – QUESTIONÁRIO APLICADO AOS DISCENTES DA FATEC MAUÁ ..  | 37 |
| FORMULÁRIO DE IDENTIFICAÇÃO .....                                  | 43 |

# **1 INTRODUÇÃO**

## **1.1 Contextualização**

Atualmente, a acessibilidade proporcionada pelos dispositivos móveis consolidou-se como o principal canal de acesso universal à informação, especialmente no meio educacional. Instituições de ensino superior em âmbito global têm implementado atualizações sistêmicas com o intuito de prover dados acadêmicos em tempo real a seus discentes, frequentemente por meio de ecossistemas digitais que unificam informações essenciais de gestão. Neste cenário, a modernização do acesso a históricos escolares, mensuração de frequências e lançamento de avaliações tornou-se imperativa para manter o estudante engajado em seu ciclo de aprendizado, exigindo infraestruturas mais ágeis do que os portais tradicionais baseados exclusivamente em plataformas web antigas.

## **1.2 Problema de pesquisa**

Como uma arquitetura multiplataforma baseada em React Native e Node.js pode superar as limitações de usabilidade mobile identificadas no SIGA, ampliando a autonomia informacional do discente da FATEC?

## **1.3 Objetivos**

O desenvolvimento do AppFatec fundamenta-se na necessidade de modernizar a experiência de consulta e gestão acadêmica discente no âmbito da Faculdade de Tecnologia. Desta forma, este trabalho delinea seus propósitos estruturados entre um objetivo geral e um conjunto de objetivos específicos orientados a guiar o ciclo completo de engenharia da solução.

### **1.3.1 Objetivo Geral**

Desenvolver, avaliar e implementar o AppFatec, um sistema multiplataforma de gestão acadêmica móvel integrado com Inteligência Artificial, visando proporcionar aos discentes da instituição um acesso dinâmico, ágil e acessível às suas informações educacionais, aperfeiçoando a usabilidade e a experiência de navegação em comparação aos portais legados tradicionais.

### 1.3.2 Objetivos Específicos

Para a consecução do objetivo geral estabelecido, definiram-se os seguintes objetivos específicos:

- **Projetar e arquitetar uma solução multiplataforma:** desenvolver uma aplicação cliente utilizando *React Native* e *Expo* voltada para dispositivos móveis (Android e iOS) a partir de uma base de código única, otimizando o ciclo de desenvolvimento e simplificando a compilação por meio de ferramentas como o *Expo Application Services* (EAS).
- **Estruturar um backend escalável e seguro:** construir uma Interface de Programação de Aplicações (API) RESTful em ambiente *Node.js* com o micro-framework *Express.js*, integrada a um banco de dados NoSQL (*MongoDB Atlas*), sob padrões rigorosos de segurança e criptografia baseados em *JSON Web Tokens* (JWT) e *bcrypt*.
- **Integrar Inteligência Artificial pedagógica:** acoplar o modelo de linguagem de grande escala *Google Gemini* por meio de seu SDK oficial para fornecer um assistente virtual de aprendizagem (Chat IA) capaz de auxiliar na orientação pedagógica do aluno, utilizando como contexto os dados acadêmicos reais do discente.
- **Implementar recursos nativos e tolerância a falhas:** prover suporte a recursos modernos de usabilidade móvel, tais como autenticação biométrica local (*Face ID* e *Touch ID*), notificações *push* assíncronas em tempo real sobre atualizações acadêmicas e persistência offline de cache utilizando *TanStack Query* e *AsyncStorage*.
- **Assegurar acessibilidade e inclusão digital:** avaliar e adequar as interfaces visuais do aplicativo baseando-se nas diretrizes de acessibilidade WCAG 2.1, garantindo conformidade ergonômica, legibilidade tipográfica superior e contraste cromático adequado em múltiplos temas (claro, escuro e automático).
- **Validar a qualidade do ecossistema de software:** estabelecer um robusto arcabouço de testes automatizados, compreendendo testes de unidade no frontend com *Jest* e testes de integração de rotas críticas no backend com *Supertest*, além de avaliações de desempenho de renderização a taxas elevadas de quadros por segundo (60 FPS).
- **Realizar validação empírica com o público-alvo:** conduzir pesquisa de campo com discentes para fundamentar requisitos, analisar o nível de aceitação, usabilidade e expectativas em relação à solução desenvolvida, balizando decisões de design centradas no usuário.

## **1.4 Justificativa**

O Sistema Integrado de Gestão Acadêmica (SIGA) tradicionalmente utilizado pela instituição baseia-se em uma arquitetura legada cuja interface não possui suporte responsivo otimizado para a exibição em telas de dispositivos móveis. Essa limitação arquitetônica resulta em uma navegação complexa, carente de notificações assíncronas em tempo real e de difícil consulta acadêmica por parte do aluno. O desenvolvimento de um ecossistema com um aplicativo móvel atuando de forma integrada como cliente, conectado a uma Interface de Programação de Aplicações (API) robusta, permite não apenas mitigar tais limitações de usabilidade, como também introduzir ferramentas computacionais de última geração. A inclusão de modelos de Inteligência Artificial para suporte pedagógico oferece uma solução tecnologicamente superior no contexto contemporâneo das tecnologias educacionais.

## **1.5 Estrutura do relatório**

O presente relatório organiza-se em três eixos centrais de análise. A etapa introdutória explora o cenário educacional e os objetivos da solução arquitetada. A seção de desenvolvimento detalha o arcabouço teórico que alicerça o projeto, além das tecnologias fundamentais e das técnicas metodológicas aplicadas na estruturação do produto base, contemplando adicionalmente as avaliações de engenharia e acessibilidade efetuadas. Por fim, as considerações finais concluem este estudo ao expor uma síntese dos aprendizados obtidos, as limitações intrínsecas à presente iteração do sistema e os apontamentos cruciais para a continuada escalabilidade do software.

## 2 DESENVOLVIMENTO

### 2.1 Fundamentação teórica

#### 2.1.1 Revisão de literatura

A expansão do uso de dispositivos móveis na educação superior tem sido objeto de crescente investigação acadêmica nas últimas décadas. A adoção de aplicativos como suporte ao processo de ensino e aprendizagem, denominada na literatura como Mobile Learning (m-learning), configura-se como uma tendência consolidada em instituições de ensino que buscam ampliar a autonomia discente e reduzir barreiras de acesso à informação acadêmica (SACCOL; REINHARD; SCHLEMMER, 2013). Nesse contexto, a disponibilização de dados de frequência, notas e calendário por meio de interfaces nativas para smartphones representa um salto qualitativo em relação aos portais web legados, que historicamente foram concebidos para acesso exclusivo em desktops e que carecem de responsividade adequada ao padrão contemporâneo de navegação em telas compactas.

Entre as ferramentas voltadas ao desenvolvimento de software multiplataforma, o React Native ocupa posição de destaque como um dos frameworks mais amplamente utilizados no mundo, permitindo a construção de aplicativos com comportamento nativo em Android e iOS a partir de um único código-fonte escrito em JavaScript/TypeScript (SILVA, 2020). Conforme aponta a documentação oficial (REACT NATIVE, 2026), o diferencial do framework reside em “combinar os melhores aspectos do desenvolvimento nativo com o React, uma biblioteca JavaScript de ponta para construção de interfaces”. Sua integração ao ecossistema Expo viabiliza o ciclo de desenvolvimento acelerado, permitindo a compilação e distribuição do aplicativo por meio de ferramentas como o Expo Application Services (EAS) sem a necessidade de ambientes de build complexos. Essa característica é particularmente relevante em projetos acadêmicos de pequenas equipes, onde a produtividade e a facilidade de manutenção são fatores críticos.

Sob a perspectiva da arquitetura de sistemas, o estilo arquitetural REST (Representational State Transfer) consolidou-se como a abordagem dominante no projeto de APIs voltadas à comunicação com clientes móveis, graças à sua simplicidade, à ausência de estado entre requisições e à plena aderência ao protocolo HTTP. A combinação de Node.js com Express.js no servidor (NODE.JS, 2026) e MongoDB Atlas como banco de dados não relacional (MONGODB, 2026) oferece um conjunto tecnológico especialmente alinhado ao perfil de sistemas acadêmicos que demandam escalabilidade horizontal e flexibilidade de esquema, dado

que o modelo de dados de um estudante — contendo matérias, notas e frequências — é inerentemente variável ao longo dos semestres.

A incorporação de IA generativa (incluindo LLMs e GANs) em aplicações educacionais é apresentada como altamente promissora para personalização, criação de conteúdo e novos formatos de interação entre estudantes e sistemas de ensino (MITTAL et al., 2023). A utilização do modelo Gemini, desenvolvido pelo Google (GOOGLE, 2026), como assistente pedagógico personalizado — tendo como contexto os dados acadêmicos reais do discente — vai ao encontro das recomendações da literatura especializada, que aponta a personalização como fator determinante para o engajamento do estudante com ferramentas de suporte cognitivo. Por fim, as diretrizes WCAG 2.1 (Web Content Accessibility Guidelines), publicadas pelo World Wide Web Consortium (W3C), constituem o referencial normativo adotado neste projeto para assegurar que a interface seja utilizável pelo mais amplo espectro possível de usuários, incluindo aqueles com deficiências visuais ou motoras (W3C, 2018).

### **2.1.2 Técnicas de desenvolvimento de software**

O ciclo de engenharia adotado na estruturação do AppFatec fundamentou-se em um modelo claro de separação de responsabilidades lógicas entre a aplicação cliente (frontend) e o servidor (backend). Para conciliar ambos, instituiu-se o padrão de arquitetura RESTful (Representational State Transfer), permitindo o trânsito veloz de informações sob o protocolo HTTP. O projeto cliente aplicou enfaticamente o princípio da modularidade e componentização reativa, enquanto a lógica de manipulação e as tratativas de persistência de informações orientaram-se sob os paradigmas do desenvolvimento modelado por rotas, aderentes às práticas modernas de segurança de tráfego que empregam esquemas de autenticação criptografada à base de JSON Web Tokens (JWT).

### **2.1.3 Linguagens de programação e frameworks**

No que concerne ao desenvolvimento da camada cliente da aplicação móvel, a pilha tecnológica consistiu no uso primário da linguagem TypeScript, com tipagem estática, em complemento ao JavaScript. Tal ambiente foi estruturado a partir do framework React Native (versão 0.81.4), contando com o gerenciamento auxiliar de configuração da plataforma Expo (versão 54). O pré-processador de estilos NativeWind (versão 4.2) foi implementado no encapsulamento estético nativo dos elementos, ao passo que as bibliotecas React Native Reanimated (versão 4.1.1) e Moti (versão 0.30) ficaram responsáveis pelos cálculos de

transição que viabilizam uma experiência tátil fluida para o usuário. Para o gerenciamento eficiente de estado assíncrono e cache de dados remotos, adotou-se o TanStack Query v5, biblioteca de sincronização de dados entre cliente e servidor.

Analogamente, o domínio do servidor estrutura-se no ambiente de execução de código aberto Node.js (versão 20.x). O ecossistema backend orquestra os métodos e roteamentos de requisições por meio do micro-framework Express.js (versão 4.18). Adotou-se o modelo não relacional (NoSQL) providenciado pelo MongoDB Atlas para o salvamento dos dados agregados, com o esquema de relações abstraído pela camada do Mongoose Object Document Mapper (ODM), versão 7.0. A integração com o modelo Google Gemini é realizada por meio do SDK oficial @google/genai (versão 1.46), executado no cliente móvel com comunicação direta à API do provedor.

### **2.1.4 Metodologias de teste de software**

A concepção sistêmica prevê o emprego de validações metodológicas direcionadas a constatar eventuais falhas assíncronas ou exceções nulas. No módulo frontend, os testes incidem sistematicamente no estado de carregamento, com as simulações e assertivas criadas por intermédio de utilitários padrões do ecossistema JavaScript, notavelmente o framework Jest. Em relação ao backend, instituiu-se um protocolo de validação de fluxo integrativo para verificar a resolutividade dos endpoints. Estes métodos garantem fundamentalmente o bloqueio de requisições sem cabedal de segurança JWT adequado, a precisão aritmética imposta nos cálculos de média e faltas por disciplina, bem como o correto processamento no upload em nuvem das imagens de perfil.

Com base nesse referencial, a próxima seção descreve os procedimentos metodológicos adotados para o desenvolvimento e validação do sistema.

## **2.2 Metodologia**

### **2.2.1 Tipo de pesquisa**

Do ponto de vista científico-tecnológico, a atual iniciativa qualifica-se metodologicamente como uma pesquisa aplicada focada na prototipação e desenvolvimento experimental. Ela sustenta também uma postura de cunho exploratório ao mapear os déficits interativos do modelo vigente em portais legados universitários e assume caráter propositivo através da disponibilização fática de uma aplicação completa que atue como infraestrutura modelo de nova geração para a consulta discente.

### **2.2.2 Método de coleta de dados**

A modelagem dos requisitos do sistema partiu da observação das limitações de usabilidade do SIGA em dispositivos móveis, identificadas pelos próprios autores durante o uso cotidiano da plataforma. Essas observações foram sistematizadas em requisitos funcionais e não funcionais, traduzindo necessidades reais dos discentes — como consulta ágil de notas, frequências e notificações — em especificações parametrizadas para o desenvolvimento do banco de dados e das interfaces do sistema.

Complementarmente à análise observacional, e com o objetivo de fundamentar empiricamente as decisões de design e funcionalidade do sistema, foi aplicado um questionário estruturado (reproduzido integralmente no Apêndice A) junto a discentes da FATEC, no mês de maio de 2026. O instrumento foi disponibilizado em formato digital por meio da plataforma Google Forms, compondo-se de 25 questões objetivas agrupadas em três eixos temáticos: (i) perfil sociodemográfico e tecnológico do respondente; (ii) avaliação da experiência atual com o Sistema Integrado de Gestão Acadêmica (SIGA); e (iii) demandas e expectativas quanto a um aplicativo móvel acadêmico. A amostra, obtida de forma não probabilística por conveniência, reuniu 94 respondentes de diferentes cursos e semestres, cujos dados consolidados são analisados na subseção 2.2.5

### **2.2.3 Procedimentos de análise**

Para certificar o funcionamento do sistema desenvolvido, adotou-se um fluxo de avaliação ponta a ponta (End-to-End) em ambiente de desenvolvimento local. O procedimento de homologação consistiu em testes locais integrados, simulando o fluxo de dados desde a inicialização de containers Docker da API backend até o consumo dos serviços pelo cliente emulador Android e iOS por meio do Expo Go. Esse fluxo validou a autenticação de credenciais e a consistência da transmissão de dados entre as camadas da aplicação.

### **2.2.4 Critérios de avaliação**

O sistema foi avaliado com base em três critérios principais. O primeiro analisou o desempenho global da aplicação em requisições assíncronas, garantindo tempos de carregamento curtos e boa usabilidade. O segundo critério focou na segurança da informação acadêmica, exigindo o bloqueio de acessos não autorizados por meio da autenticação baseada em token JWT. O terceiro critério avaliou a consistência visual da interface, assegurando a aderência ao Design System proposto e a correta exibição dos dados estatísticos do estudante.

### 2.2.5 Pesquisa de campo com discentes

Os resultados obtidos por meio do questionário aplicado (ver Apêndice A) são apresentados a seguir em três blocos analíticos: perfil dos respondentes, avaliação do sistema SIGA vigente e demandas para um aplicativo móvel acadêmico. Todos os dados referem-se a uma amostra de N = 94 respondentes, salvo indicação contrária.

#### *Perfil dos respondentes*

A distribuição etária dos participantes revelou predominância de estudantes jovens adultos. Conforme evidenciado na Tabela 1, a faixa etária de 17 a 20 anos concentrou a maior parcela dos respondentes (30,9%), seguida pela faixa de 21 a 25 anos (27,7%), indicando que o público-alvo da solução é composto majoritariamente por nativos digitais familiarizados com dispositivos móveis.

**Tabela 1 – Distribuição dos respondentes por faixa etária**

| <b>Faixa etária</b> | <b>Frequência absoluta</b> | <b>Frequência relativa (%)</b> |
|---------------------|----------------------------|--------------------------------|
| 17 a 20 anos        | 29                         | 30,9                           |
| 21 a 25 anos        | 26                         | 27,7                           |
| 26 a 30 anos        | 17                         | 18,1                           |
| Acima de 30 anos    | 22                         | 23,4                           |
| Total               | 94                         | 100,0                          |

Fonte: Elaborado pelos autores (2026).

No que se refere ao dispositivo de acesso predominante, a Tabela 2 aponta que a maioria dos respondentes (69,1%) utiliza o smartphone como principal meio de acesso a informações acadêmicas, dado que corrobora diretamente a pertinência do desenvolvimento de uma solução nativa para dispositivos móveis.

**Tabela 2 – Dispositivo predominantemente utilizado para acesso a informações acadêmicas**

| Dispositivo                                              | Frequência absoluta | Frequência relativa (%) |
|----------------------------------------------------------|---------------------|-------------------------|
| Smartphone (celular)                                     | 65                  | 69,1                    |
| Notebook ou computador de mesa                           | 15                  | 16,0                    |
| Utilizo todos os dispositivos com frequência equivalente | 13                  | 13,8                    |
| Tablet                                                   | 1                   | 1,1                     |
| Total                                                    | 94                  | 100,0                   |

Fonte: Elaborado pelos autores (2026).

*Avaliação do sistema SIGA vigente*

A percepção dos discentes quanto à usabilidade do SIGA em dispositivos móveis foi avaliada por meio de três indicadores distintos: facilidade de uso, velocidade de carregamento e qualidade do design visual. Os resultados consolidados estão apresentados na Tabela 3.

**Tabela 3 – Avaliação dos discentes quanto aos atributos de qualidade do SIGA em dispositivos móveis**

| Atributo avaliado                 | Excelente (%) | Bom (%) | Regular (%) | Ruim (%) | Péssimo (%) |
|-----------------------------------|---------------|---------|-------------|----------|-------------|
| Facilidade de uso em mobile       | 6,4           | 41,5    | 38,3        | 9,6      | 4,3         |
| Velocidade de carregamento        | 4,3           | 46,8    | 39,4        | 7,4      | 2,1         |
| Design visual e interface gráfica | 4,3           | 36,2    | 27,7        | 23,4     | 8,5         |

Fonte: Elaborado pelos autores (2026).

Os dados da Tabela 3 revelam uma avaliação majoritariamente intermediária quanto à usabilidade do SIGA em dispositivos móveis, com o design visual concentrando a maior insatisfação: enquanto 47,9% dos respondentes classificaram a facilidade de uso em mobile como "Bom" ou "Excelente" e 38,3% como "Regular", o design visual e a interface gráfica reuniram 31,9% de avaliações "Ruim" ou "Péssimo" — o pior desempenho entre os três atributos. Ainda que a percepção geral não seja predominantemente negativa, as fragilidades

específicas tornam-se evidentes nas dificuldades pontuais relatadas (Tabela 5), em consonância com a justificativa elencada na subseção 1.3 deste relatório. Tais atritos ganham relevância quando considerados em conjunto com a frequência de acesso declarada: conforme a Tabela 4, a maioria dos estudantes (81,9%) acessa o sistema de uma a várias vezes por semana, o que torna qualquer obstáculo de usabilidade um problema recorrente.

**Tabela 4 – Frequência de acesso ao SIGA e dispositivo preferencialmente utilizado**

| <b>Frequência de acesso</b>       | <b>Frequência relativa (%)</b> |
|-----------------------------------|--------------------------------|
| De duas a quatro vezes por semana | 45,7                           |
| Uma vez por semana                | 27,7                           |
| De duas a três vezes por mês      | 14,9                           |
| Diariamente                       | 8,5                            |
| Raramente ou nunca                | 3,2                            |
| Total                             | 100,0                          |

Fonte: Elaborado pelos autores (2026).

Quanto às dificuldades enfrentadas, a Tabela 5 sintetiza as principais barreiras de uso identificadas pelos respondentes. Ressalta-se que a questão permitia a seleção de até três alternativas, razão pela qual o somatório das frequências relativas ultrapassa 100%.

**Tabela 5 – Principais dificuldades enfrentadas pelos discentes ao utilizar o SIGA**

| <b>Dificuldade</b>                                       | <b>Frequência absoluta</b> | <b>% em relação ao total de respondentes</b> |
|----------------------------------------------------------|----------------------------|----------------------------------------------|
| Necessidade frequente de reautenticação                  | 52                         | 55,3                                         |
| Lentidão ou travamento do sistema                        | 44                         | 46,8                                         |
| Dificuldade em localizar informações (navegação confusa) | 28                         | 29,8                                         |
| Aparência visual ultrapassada                            | 27                         | 28,7                                         |
| Ausência de notificações automáticas                     | 24                         | 25,5                                         |
| Interface não adaptada a dispositivos móveis             | 22                         | 23,4                                         |
| Não encontro dificuldades significativas                 | 9                          | 9,6                                          |

Fonte: Elaborado pelos autores (2026).

A extensão do impacto dessas limitações é reforçada pelos dados das Tabelas 6 e 7. Conforme demonstrado, 56,4% dos respondentes declararam que já deixaram de consultar informações acadêmicas em algumas ou diversas ocasiões em razão de dificuldades de uso do

sistema (Tabela 6). Além disso, 56,4% dos alunos confirmaram que já perderam prazos ou comunicados importantes em algumas ou diversas ocasiões pela ausência de notificações automáticas — funcionalidade inexistente no SIGA atual (Tabela 7).

**Tabela 6 – Ocorrência de omissão de consulta acadêmica em razão de dificuldades no SIGA**

| <b>Resposta</b>           | <b>Frequência absoluta</b> | <b>Frequência relativa (%)</b> |
|---------------------------|----------------------------|--------------------------------|
| Não, jamais ocorreu       | 14                         | 14,9                           |
| Sim, em raras ocasiões    | 27                         | 28,7                           |
| Sim, em algumas ocasiões  | 37                         | 39,4                           |
| Sim, em diversas ocasiões | 16                         | 17,0                           |
| Total                     | 94                         | 100,0                          |

Fonte: Elaborado pelos autores (2026).

**Tabela 7 – Ocorrência de perda de prazos ou comunicados por ausência de notificação do SIGA**

| <b>Resposta</b>           | <b>Frequência absoluta</b> | <b>Frequência relativa (%)</b> |
|---------------------------|----------------------------|--------------------------------|
| Não, jamais ocorreu       | 22                         | 23,4                           |
| Sim, em raras ocasiões    | 19                         | 20,2                           |
| Sim, em algumas ocasiões  | 31                         | 33,0                           |
| Sim, em diversas ocasiões | 22                         | 23,4                           |
| Total                     | 94                         | 100,0                          |

Fonte: Elaborado pelos autores (2026).

#### *Demandas e expectativas para um aplicativo móvel acadêmico*

Na terceira dimensão da pesquisa, os respondentes foram consultados acerca das funcionalidades que consideram mais relevantes em um aplicativo acadêmico móvel, bem como sobre a receptividade a recursos específicos implementados no AppFatec. A Tabela 8 apresenta as funcionalidades mais demandadas, em questão de múltipla escolha com até três alternativas.

**Tabela 8 – Funcionalidades consideradas mais relevantes em um aplicativo acadêmico móvel**

| Funcionalidade                                        | Frequência absoluta | % em relação ao total de respondentes |
|-------------------------------------------------------|---------------------|---------------------------------------|
| Consulta ágil de notas e faltas                       | 86                  | 91,5                                  |
| Notificações automáticas de novas notas e comunicados | 69                  | 73,4                                  |
| Alerta de risco de reprovação por excesso de faltas   | 65                  | 69,1                                  |
| Acesso a documentos e declarações                     | 43                  | 45,7                                  |
| Visualização do horário diário de aulas               | 40                  | 42,6                                  |
| Calculadora de média acadêmica                        | 38                  | 40,4                                  |
| Assistente virtual com inteligência artificial        | 25                  | 26,6                                  |
| Alternância entre temas claro e escuro                | 18                  | 19,1                                  |

Fonte: Elaborado pelos autores (2026).

A recepção à funcionalidade de notificações automáticas de notas e faltas, já implementada no AppFatec, foi avaliada por meio de escala Likert de cinco pontos, cujos resultados estão consolidados na Tabela 9. Constatou-se que 94,7% dos discentes manifestaram intenção de uso positiva (concordando ou concordando parcialmente com a utilidade do recurso).

**Tabela 9 – Nível de concordância dos respondentes com a utilidade de notificações automáticas de notas e faltas**

| Nível de concordância | Frequência absoluta | Frequência relativa (%) |
|-----------------------|---------------------|-------------------------|
| Concordo              | 80                  | 85,1                    |
| Concordo parcialmente | 9                   | 9,6                     |
| Indiferente           | 4                   | 4,3                     |
| Discordo parcialmente | 1                   | 1,1                     |
| Discordo              | 0                   | 0,0                     |
| Total                 | 94                  | 100,0                   |

Fonte: Elaborado pelos autores (2026).

De forma análoga, a Tabela 10 registra a percepção dos discentes quanto à utilidade de um assistente de inteligência artificial integrado ao aplicativo — funcionalidade incorporada ao AppFatec por meio da integração com o modelo Google Gemini. O resultado aponta que 73,4%

dos estudantes consideram que o assistente virtual traria benefícios ao seu desempenho acadêmico.

**Tabela 10 – Nível de concordância quanto à utilidade de assistente de IA integrado ao aplicativo acadêmico**

| Nível de concordância | Frequência absoluta | Frequência relativa (%) |
|-----------------------|---------------------|-------------------------|
| Concordo              | 48                  | 51,1                    |
| Concordo parcialmente | 21                  | 22,3                    |
| Indiferente           | 16                  | 17,0                    |
| Discordo              | 6                   | 6,4                     |
| Discordo parcialmente | 3                   | 3,2                     |
| Total                 | 94                  | 100,0                   |

Fonte: Elaborado pelos autores (2026).

Por fim, a Tabela 11 sintetiza a expectativa de melhoria da experiência acadêmica proporcionada por um aplicativo móvel dedicado em comparação ao SIGA atual. Os dados apontam um forte consenso sobre o impacto positivo da solução, com 96,8% dos respondentes projetando melhorias significativas (enormes ou moderadas).

**Tabela 11 – Expectativa de melhoria da experiência acadêmica com um aplicativo móvel dedicado**

| Percepção de melhoria esperada | Frequência absoluta | Frequência relativa (%) |
|--------------------------------|---------------------|-------------------------|
| Traria melhoria enorme         | 60                  | 63,8                    |
| Traria melhoria moderada       | 31                  | 33,0                    |
| Não traria qualquer melhoria   | 3                   | 3,2                     |
| Total                          | 94                  | 100,0                   |

Fonte: Elaborado pelos autores (2026).

Os dados coletados evidenciam que os estudantes reconhecem de forma abrangente as limitações do SIGA para uso mobile e demonstram forte demanda pelas funcionalidades centrais implementadas no AppFatec — notadamente a consulta ágil de notas e faltas, as notificações automáticas e o assistente inteligente. Esses resultados legitimam as escolhas arquiteturais e funcionais adotadas no desenvolvimento do sistema e qualificam a iniciativa como uma resposta tecnologicamente adequada às necessidades reais do corpo discente da instituição.

2.3 Ficha técnica

Conforme demandado pelas especificações projetuais, as restrições e ferramentas componentes deste modelo arquitetônico se encontram sintetizadas no Quadro 1.

Quadro 1 – Ficha técnica do sistema AppFatec

|                                |                                                                                                                                                                                                                                                                           |
|--------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Nome do sistema/software       | App FATEC                                                                                                                                                                                                                                                                 |
| Versão                         | 2.0                                                                                                                                                                                                                                                                       |
| Desenvolvedores                | Daniel Pereira, Gabriel Mesquita, Hiago Gabriel, Nauã da Silva                                                                                                                                                                                                            |
| Data de criação                | 10 de fevereiro de 2025                                                                                                                                                                                                                                                   |
| Plataformas                    | Android, Ios, WEB                                                                                                                                                                                                                                                         |
| Linguagens utilizadas          | JavaScript, TypeScript                                                                                                                                                                                                                                                    |
| Frameworks                     | React Native (v0.81.4), Expo (v54), Node.js (v20.x) e Express.js (v4.18)                                                                                                                                                                                                  |
| Banco de dados                 | MongoDB Atlas (Mongoose v7.0)                                                                                                                                                                                                                                             |
| Requisitos de hardware         | Dispositivos móveis inteligentes em compilações operacionais atuais, atendentes aos requisitos base estabelecidos pelos canais Expo Developer                                                                                                                             |
| Requisitos de software         | Ambiente do lado do cliente estruturado sobre empacotamento nativo da App Store ou Play Store; na infraestrutura distribuída, servidores pautados no ambiente Runtime do Node.js (v20+), passíveis de orquestramento através de containers virtualizados da Docker Engine |
| Ferramentas de desenvolvimento | Visual Studio Code, Git, Docker Compose e clientes auxiliares CLI do Expo (EAS)                                                                                                                                                                                           |
| Objetivo                       | Oferecer ao discente acesso ágil e centralizado a informações acadêmicas (notas, faltas, horários e calendário) por meio de um aplicativo móvel nativo, com suporte de assistente pedagógico baseado em Inteligência Artificial.                                          |

Fonte: Estrutura adaptada pelos autores (2026).

2.4 Técnicas e testes de software e avaliação da acessibilidade

2.4.1 Testes unitários

No panorama do ecossistema front-end, foi conferida ênfase particular aos componentes visuais da infraestrutura e aos comportamentos isolados contidos em ganchos abstratos denominados "Custom Hooks" — notadamente as estruturas lógicas useStudent e useActivities. A execução automatizada, implementada com o framework Jest, garantiu comportamentos previsíveis e estritos em situações-limite, como cenários em que há esvaziamento de

preenchimento ou devolução nula da API, precavendo instabilidades na renderização da árvore de documentos do display. Os testes em `useStudent.test.ts` e `useActivities.test.ts` cobrem os estados de carregamento (`loading: true`), sucesso no recebimento de dados da API, tratativas de erros de rede e comportamento do método `refetch()`.

#### **2.4.2 Testes de integração**

Dada a arquitetura modular em camadas do servidor, organizada por domínios funcionais (autenticação, estudante, atividades e upload), o projeto baseou-se categoricamente na eficácia coesa dessas vias. Comprovou-se satisfatoriamente o acoplamento do sistema a utilitários de terceiros de armazenamento e cognição semântica externa. As rotas internas encarregadas do manuseio processaram os envios de dados no formato multipart/form-data para provedores de nuvem adequadamente — exemplificado pela alocação de arquivos no bucket R2 hospedado na plataforma Cloudflare, em harmonia total com as diretivas seguras de autenticação via `PutObjectCommand` do SDK AWS S3 compatível, bem como a implementação sintática da API de streaming oferecida ao cliente pela IA integrada do Google Gemini. O conjunto de testes de integração do backend cobre três domínios críticos: segurança e controle de acesso JWT (`auth-security.test.js`), cálculo de médias e dados de alunos (`student.test.js`) e gerenciamento de atividades acadêmicas (`activity.test.js`).

#### **2.4.3 Testes de performance**

A avaliação de desempenho do sistema foi conduzida sob duas perspectivas complementares: a fluidez da camada de renderização no cliente móvel e a capacidade de resposta da API no servidor.

No frontend, o desempenho de animação é viabilizado pela biblioteca `React Native Reanimated` (versão 4.1.1), que transfere os cálculos de interpolação e transformações visuais do thread principal do JavaScript para threads nativas de UI dedicadas (`UI Thread` e `Render Thread`), disponíveis nas CPUs dos dispositivos Android e iOS. Essa arquitetura permite que as transições de tela e microinterações operem de forma independente do estado de carregamento de dados da aplicação, alcançando taxas de renderização próximas a 60 quadros por segundo (60 FPS) de forma ininterrupta nas animações de navegação, mesmo sob carga assíncrona. A biblioteca `Moti` (versão 0.30), construída sobre o `Reanimated`, complementa esse conjunto ao fornecer primitivas de animação declarativas que simplificam a composição de efeitos de entrada e saída de componentes.

No que diz respeito à resiliência da API backend, o uso de containers Docker orquestrados pelo Docker Compose permitiu reproduzir ambientes de execução equivalentes ao de produção de forma local e controlada. Os testes de integração com Supertest simularam múltiplas requisições simultâneas aos endpoints críticos de leitura (GET /api/student) e escrita (POST /api/activities), validando a consistência das respostas e o correto isolamento por token JWT. O cache de dados implementado no cliente por meio do TanStack Query — com staleTime de 5 minutos e gcTime de 24 horas, persistido via AsyncStorage — reduz significativamente o volume de requisições redundantes ao servidor, contribuindo para a economia de banda e a melhoria da experiência do usuário em condições de conectividade intermitente.

#### **2.4.4 Testes de segurança**

A segurança das credenciais é garantida por hashing com bcrypt, que impede o armazenamento de senhas em texto puro no banco de dados. O controle de acesso às rotas protegidas é feito pelo middleware middlewares/auth.js, que valida o JSON Web Token (JWT) em cada requisição antes de encaminhá-la às camadas de controle — bloqueando qualquer acesso desprovido de token válido. Tokens são gerados com prazo de expiração restrito, mitigando riscos de sessões abertas indefinidamente.

#### **2.4.5 Metodologias de desenvolvimento e qualidade**

O ciclo de desenvolvimento iterativo contou com o recurso de Fast Refresh do React Native, que recarrega apenas os componentes modificados sem reiniciar a aplicação por completo. Isso permitiu validar cada alteração de forma imediata, reduzindo o tempo entre implementação e verificação visual e funcionando como barreira preventiva contra a introdução de erros antes da consolidação no repositório principal.

#### **2.4.6 Avaliação de acessibilidade**

O sistema foi minuciosamente verificado para atestar sua usabilidade por pessoas com diferentes perfis de acesso, com o foco em promover a maior abrangência possível no uso da plataforma.

A avaliação de acessibilidade do sistema teve como principal objetivo garantir que a interface fosse utilizável por diferentes perfis de usuários, incluindo aqueles com limitações visuais e dificuldades de interação. Buscou-se assegurar que os elementos visuais e interativos

proporcionassem uma experiência intuitiva, inclusiva e eficiente, alinhada às boas práticas de usabilidade em aplicações modernas.

A análise foi conduzida com base em princípios de design contemporâneo, especialmente aqueles oriundos da filosofia Clean Material, priorizando organização visual, consistência e clareza na interface. Foram realizadas verificações práticas de navegação e interação, considerando diferentes condições de uso, como variações de luminosidade e uso em dispositivos móveis. A avaliação concentrou-se na estrutura dos elementos, responsividade e comportamento dos componentes interativos.

Como referência normativa, a interface foi analisada com base nos critérios estabelecidos pelas diretrizes WCAG 2.1, especialmente no que se refere ao contraste de cores e legibilidade. Foram realizados ajustes na colorimetria para garantir níveis adequados de contraste entre texto e fundo, respeitando os padrões recomendados. Também foi considerada a utilização em diferentes modos de exibição — tema claro, tema escuro e tema automático (sincronizado com as configurações do sistema operacional) —, assegurando consistência visual e legibilidade em todos os cenários.

Durante a avaliação, não foram identificadas falhas críticas de acessibilidade. No entanto, como medida preventiva, destacou-se a necessidade de manter uniformidade no contraste entre todos os elementos da interface e garantir que componentes interativos permaneçam claramente visíveis. Como proposta de melhoria, recomenda-se a utilização futura de ferramentas automatizadas, como Lighthouse e WAVE, além da realização de testes com tecnologias assistivas, como leitores de tela.

Os resultados obtidos indicam que o sistema apresenta um nível satisfatório de acessibilidade, proporcionando uma experiência visual estável e de fácil interpretação. A tipografia adotada, com maior espessura e boa proporção, contribui para a legibilidade do conteúdo. Além disso, os elementos interativos, como botões, foram projetados com dimensões ampliadas e bordas arredondadas, respeitando princípios ergonômicos que reduzem erros de toque e aumentam a precisão na interação. Dessa forma, a aplicação demonstra alinhamento com os conceitos de usabilidade e acessibilidade discutidos na fundamentação teórica, reforçando a importância do desenvolvimento de sistemas inclusivos.

## **2.5 Funcionalidades complementares implementadas**

Esta seção documenta funcionalidades do sistema AppFatec que, embora integralmente implementadas na base de código, não haviam sido abordadas em profundidade nas seções

anteriores. Cada subseção descreve o objetivo da funcionalidade, os atores e fluxos envolvidos, bem como os componentes técnicos responsáveis pela sua execução.

### 2.5.1 Autenticação biométrica por hardware

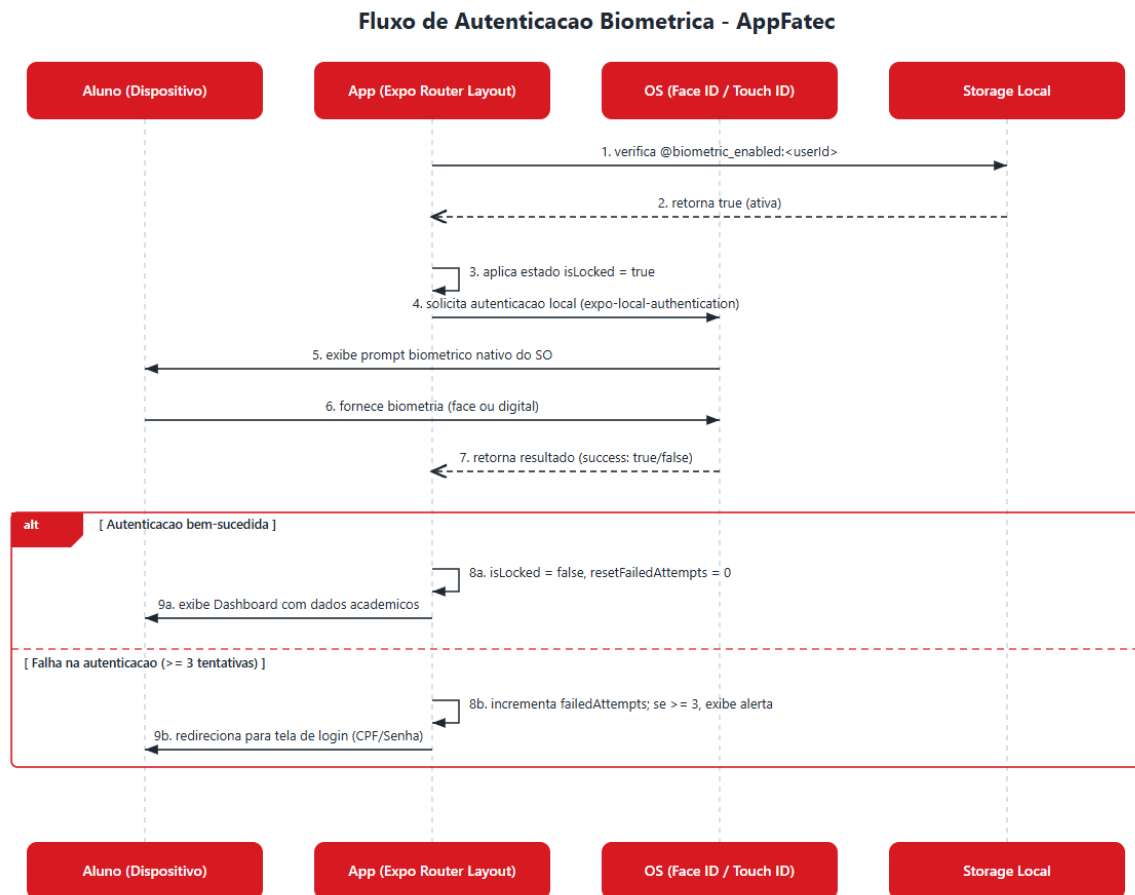
O AppFatec implementa uma camada de segurança adicional no acesso aos dados acadêmicos por meio da autenticação biométrica nativa dos dispositivos móveis — Face ID (iOS) e leitor de impressão digital (Android). Essa funcionalidade é gerenciada pela biblioteca expo-local-authentication, que abstrai as APIs biométricas dos sistemas operacionais, possibilitando uma implementação unificada para ambas as plataformas.

O fluxo de segurança opera da seguinte forma: ao realizar o login inicial com CPF e senha, o discente que possuir hardware biométrico disponível e credenciais enrolladas no sistema operacional é convidado a ativar o recurso nas configurações de perfil. Uma vez ativado, a cada nova inicialização do aplicativo, o layout autenticado (`app/(app)/_layout.tsx`) bloqueia imediatamente a exibição de dados sensíveis e aciona o prompt biométrico nativo do sistema operacional. Em caso de autenticação bem-sucedida, o estado de bloqueio é encerrado e o Dashboard é exibido normalmente.

O sistema implementa um limite de três tentativas biométricas malsucedidas consecutivas. Atingido esse limite, o aplicativo exibe um alerta informativo e redireciona o usuário para a tela de login com CPF e senha, garantindo que o acesso legítimo não seja permanentemente bloqueado por falhas de hardware ou envelhecimento biométrico. As preferências de ativação são armazenadas de forma isolada por usuário no AsyncStorage, utilizando chaves prefixadas pelo identificador único do discente (`@biometric_enabled:<userId>`), o que assegura a independência de configurações em dispositivos compartilhados.

O fluxo completo de autenticação biométrica, do acionamento ao desbloqueio da sessão, é representado no diagrama de sequência da Figura 8.

Figura 8 – Diagrama de sequência do fluxo de autenticação biométrica



Fonte: Elaborado pelos autores (2026).

### 2.5.2 Suporte offline e persistência de cache

O AppFatec foi projetado para manter a usabilidade em condições de conectividade intermitente ou ausente, característica de grande relevância em contextos de mobilidade. Essa resiliência é viabilizada por duas camadas complementares: a detecção em tempo real do estado da rede e a persistência local do cache de dados.

A detecção de conectividade é implementada pelo hook `useNetworkStatus`, que encapsula a biblioteca `@react-native-community/netinfo`. Ao subscrever eventos de mudança de estado de rede, o hook mantém atualizado o flag `isConnected` ao longo do ciclo de vida da aplicação. Quando a ausência de conexão é detectada, a interface exibe um banner visual de estado offline, informando o discente sem bloquear o acesso às informações previamente carregadas.

A persistência dos dados acadêmicos é gerenciada pelo TanStack Query em conjunto com o `PersistQueryClientProvider`, configurado em `lib/queryClient.ts`. Os dados recuperados da API são serializados e armazenados no `AsyncStorage` sob o prefixo

@appfatec\_query\_cache, com tempo de revalidação (staleTime) de cinco minutos e tempo de retenção em cache (gcTime) de 24 horas. Dessa forma, um discente que acessou o aplicativo enquanto conectado pode consultar suas notas, frequências e disciplinas sem qualquer degradação de experiência mesmo em ambientes sem sinal de internet, como metrô ou zonas de cobertura deficiente.

### **2.5.3 Sistema de mensagens internas (aluno ↔ professor)**

O AppFatec incorpora um canal de comunicação direta entre discentes e os professores responsáveis pelas disciplinas em que estão matriculados. Esse módulo permite a troca assíncrona de mensagens de texto, agregando um histórico estruturado por remetente e ordenado cronologicamente.

No frontend, o módulo é composto pela tela de listagem de conversas (app/(app)/mensagens.tsx) e pela tela de conversa individual, acessada por rota dinâmica (app/(app)/mensagens/[userId].tsx), onde userId corresponde ao identificador único do professor. O hook useMessages abstrai as chamadas à API e o gerenciamento de estado das mensagens. No backend, o messageController.js é responsável pela persistência das mensagens no modelo Message do MongoDB, garantindo o correto mapeamento entre remetente, destinatário e conteúdo de cada interação.

Essa funcionalidade responde a uma das demandas identificadas na pesquisa de campo (seção 2.2.5): a ausência de canais de comunicação direta no SIGA atual obriga estudantes a recorrer a meios externos e informais para contato com docentes, o que prejudica a rastreabilidade e a formalidade das interações acadêmicas.

### **2.5.4 Notificações push e gerenciamento de tokens**

O AppFatec implementa notificações push para alertar proativamente o discente sobre eventos acadêmicos relevantes, como o lançamento de novas notas ou comunicados institucionais. A funcionalidade é construída sobre o serviço Expo Push Notifications, que gerencia o roteamento de notificações para os serviços nativos FCM (Firebase Cloud Messaging, Android) e APNs (Apple Push Notification Service, iOS).

O ciclo de vida do token de notificação é orquestrado pelo módulo lib/notifications.ts. Ao inicializar a sessão autenticada, o aplicativo solicita permissão ao usuário, registra o dispositivo junto à plataforma Expo para obtenção do token, e envia esse token ao backend por meio de requisição autenticada. O backend persiste o token no modelo NotificationToken,

vinculado ao identificador do usuário, permitindo que notificações direcionadas sejam disparadas para dispositivos específicos. Ao realizar logout, o token é removido tanto do armazenamento local quanto do backend — prevenindo o recebimento de notificações após o encerramento de sessão.

O listener de notificações, configurado no layout autenticado (`app/(app)/_layout.tsx`), captura eventos de toque em notificações recebidas e realiza o roteamento imediato para a tela correspondente no aplicativo, por meio do `router.push` do Expo Router, garantindo uma experiência coerente de navegação por notificação.

### **2.5.5 Calendário acadêmico interativo**

O módulo de calendário do AppFatec oferece ao discente uma visualização estruturada das datas letivas relevantes ao seu semestre, incluindo prazos de avaliação, feriados e eventos institucionais. A funcionalidade é composta por dois pontos de acesso: um painel resumido (`CalendarWidget`) exibido na tela de Dashboard para apresentar os próximos eventos de forma compacta, e uma tela dedicada de calendário mensal (`app/(app)/calendario.tsx`) acessível por meio da barra lateral de navegação (`Sidebar`).

Essa organização cronológica permite que o estudante planeje sua rotina de estudos com antecedência, visualizando de forma integrada os compromissos acadêmicos de todas as disciplinas em que está matriculado sem a necessidade de consultar cada disciplina individualmente.

### **2.5.6 Interface responsiva adaptativa (Bottom Tabs e Sidebar)**

A arquitetura de navegação do AppFatec foi projetada para se adaptar ao tamanho da tela do dispositivo em uso, oferecendo experiências de navegação otimizadas para celulares, tablets e desktops — inclusive quando executado como aplicação web via Expo Web.

A lógica de adaptação está centralizada no layout autenticado (`app/(app)/_layout.tsx`), que monitora continuamente a largura da janela por meio do hook `useWindowDimensions`. Quando a largura detectada é inferior a 768 pixels — característica de smartphones em orientação retrato —, o sistema exibe uma barra de navegação inferior (`Bottom Tabs`), padrão amplamente estabelecido em interfaces mobile. Quando a largura é igual ou superior a 768 pixels — indicativa de tablets em modo paisagem, iPads ou desktops —, as abas inferiores são ocultadas e substituídas pelo componente `Sidebar`, uma barra lateral fixa de navegação que aproveita o maior espaço disponível em telas largas para exibir rótulos completos e acesso a

seções adicionais, como Calendário e Perfil, que ficam ocultas da barra de abas mobile por limitações de espaço.

Essa estratégia de design responsivo garante que o aplicativo seja utilizável em um amplo espectro de dispositivos sem a necessidade de builds separados ou adaptações manuais de layout por plataforma.

### **2.5.7 Painel de detalhes da disciplina**

Além da tela consolidada de notas e frequências acessível pela navegação principal, o AppFatec disponibiliza um painel individualizado para cada disciplina matriculada. Essa funcionalidade é acessada ao tocar em qualquer cadeira acadêmica na listagem de notas (`app/(app)/notas.tsx`) ou na tela de disciplinas (`app/(app)/disciplinas.tsx`), redirecionando o estudante para a rota dinâmica `app/(app)/disciplina/[id].tsx`, onde `id` é o identificador único da matéria no banco de dados.

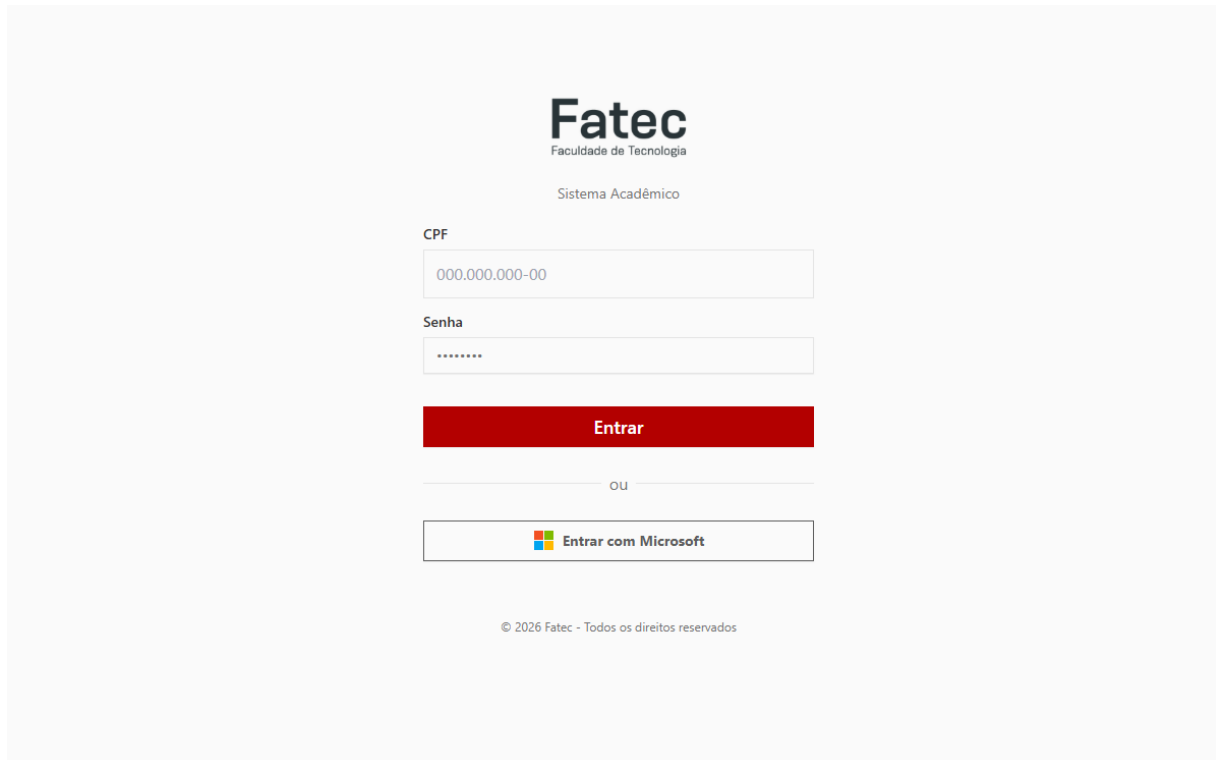
O painel de detalhes exibe um conjunto completo de informações da cadeira acadêmica: as notas individuais fracionadas nas modalidades P1, P2, Trabalho (T) e P3; o percentual exato de frequência calculado pelo backend em tempo real a partir de  $((\text{totalAulas} - \text{faltas}) / \text{totalAulas}) \times 100$ ; a contagem absoluta de faltas acumuladas com indicação visual de risco de reprovação quando o limite institucional é ultrapassado; a ementa ou plano de ensino da disciplina; e os dados de contato do professor responsável. Essa visão granular capacita o discente a identificar com precisão em quais avaliações seu desempenho está abaixo do esperado e qual é sua margem de segurança em termos de frequência, favorecendo decisões de estudo mais direcionadas.

## **2.6 Interfaces gráficas do sistema**

Com o objetivo de validar visualmente a usabilidade, a responsividade e a aderência aos padrões estéticos modernos da filosofia Clean Material descritos anteriormente, apresenta-se nesta seção a demonstração gráfica das principais telas que compõem o ecossistema do AppFatec.

O acesso inicial e a segurança de dados no primeiro contato do estudante com o aplicativo ocorrem por meio da interface de login, conforme apresentado na Figura 1.

Figura 1 – Tela de Login e Identificação Discente



A imagem mostra a interface de login do sistema acadêmico do Fatec. No topo, o logo "Fatec" é exibido em uma fonte grande e preta, com "Faculdade de Tecnologia" em uma fonte menor abaixo dele. Abaixo do logo, o texto "Sistema Acadêmico" é centralizado. O formulário de login contém dois campos de entrada: "CPF" com o exemplo "000.000.000-00" e "Senha" com caracteres ocultos por pontos. Abaixo dos campos, há um botão vermelho com o texto "Entrar" em branco. Segue-se a palavra "OU" centralizada. Abaixo disso, há um botão branco com o ícone da Microsoft e o texto "Entrar com Microsoft". No rodapé, o texto "© 2026 Fatec - Todos os direitos reservados" é exibido em uma fonte pequena.

Fonte: Elaborado pelos autores (2026).

A tela de login (Figura 1) exige o preenchimento do CPF e senha, servindo como o portal de entrada para a autenticação e posterior geração do token JWT.

Após a realização do login inicial, caso o discente opte por ativar a autenticação local nas configurações, o aplicativo passa a exigir validação biométrica a cada inicialização, conforme exibido na Figura 2.

Figura 2 – Tela de Desbloqueio e Autenticação Biométrica

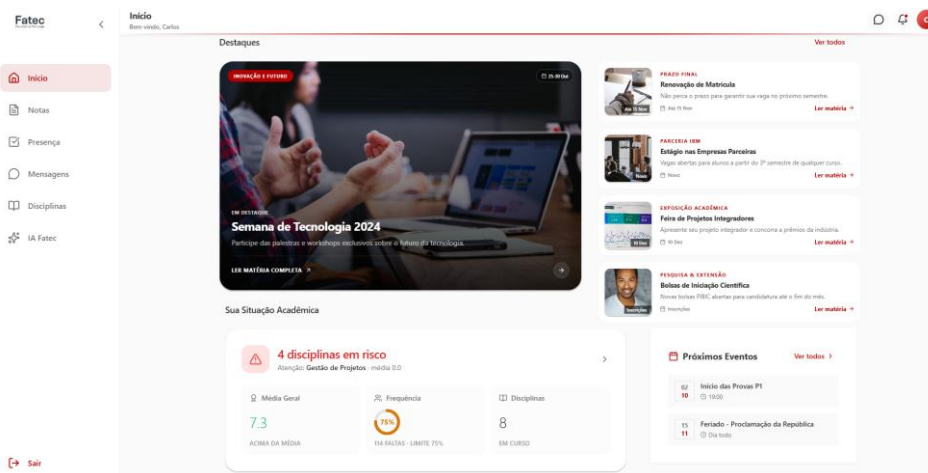


Fonte: Elaborado pelos autores (2026).

A interface de bloqueio (Figura 2) previne acessos não autorizados aos dados acadêmicos do aluno e oferece um mecanismo rápido de desbloqueio por digital ou reconhecimento facial, contando com um botão de alternativa para digitação da senha caso o hardware biométrico falhe.

Uma vez autenticado, o estudante é direcionado à tela de início (Dashboard). A disposição dos elementos interativos nesta tela é ilustrada na Figura 3.

Figura 3 – Painel Principal (Dashboard)

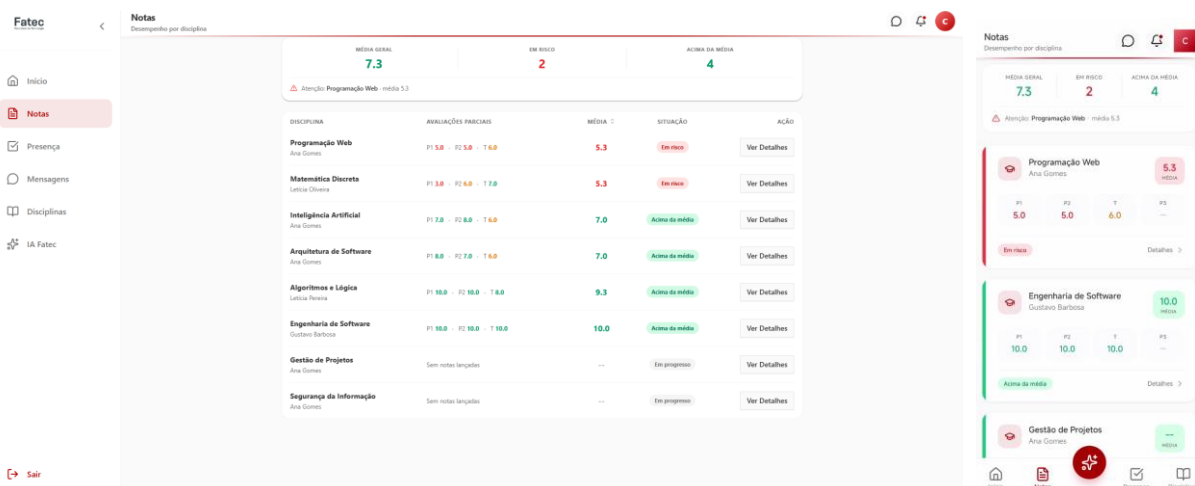


Fonte: Elaborado pelos autores (2026).

O Dashboard (Figura 3) funciona como o centro operacional do aluno, exibindo o carrossel de notícias institucionais, atalhos rápidos para informações acadêmicas relevantes e um painel de calendário com eventos próximos.

Para o acompanhamento detalhado de rendimento, o discente pode acessar a tela de Notas, ilustrada na Figura 4.

Figura 4 – Tela de Histórico de Notas

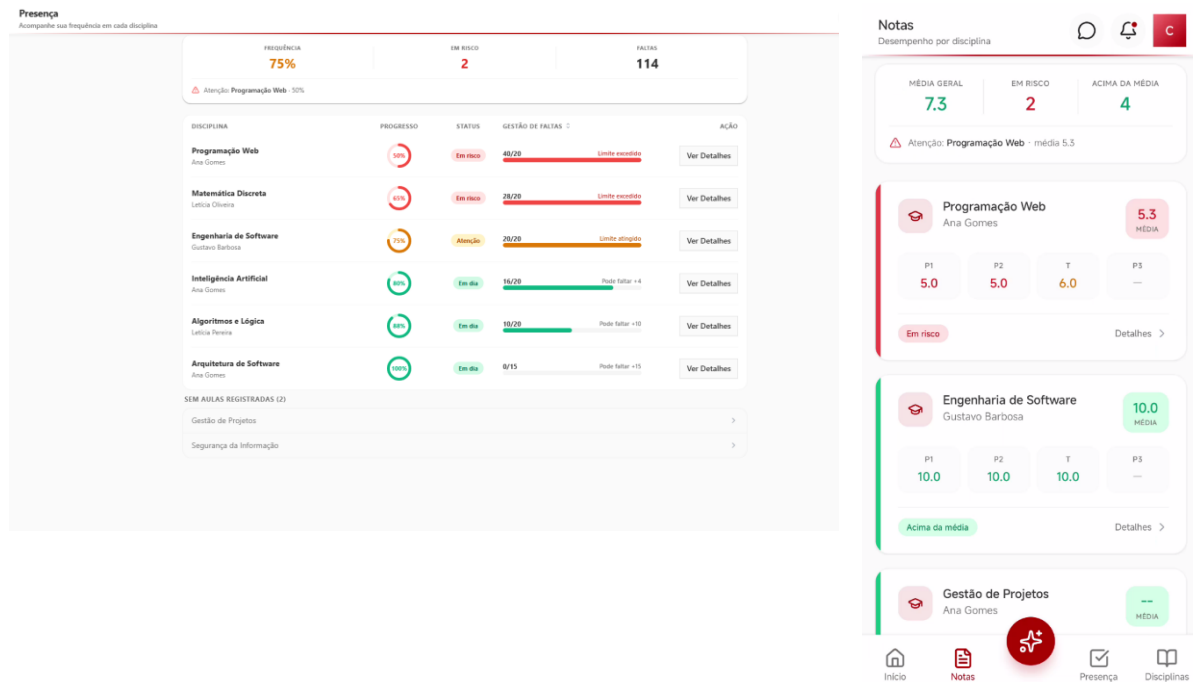


Fonte: Elaborado pelos autores (2026).

Na tela de Notas (Figura 4), são listadas todas as matérias ativas e concluídas no semestre corrente, juntamente com a média final calculada dinamicamente pelo backend e o status situacional da matrícula.

Ao tocar em uma disciplina específica na listagem, o aplicativo direciona o aluno para a tela de detalhes, ilustrada na Figura 5.

Figura 5 – Tela de Detalhes da Disciplina e Frequência: versão web (esquerda) e mobile (direita)

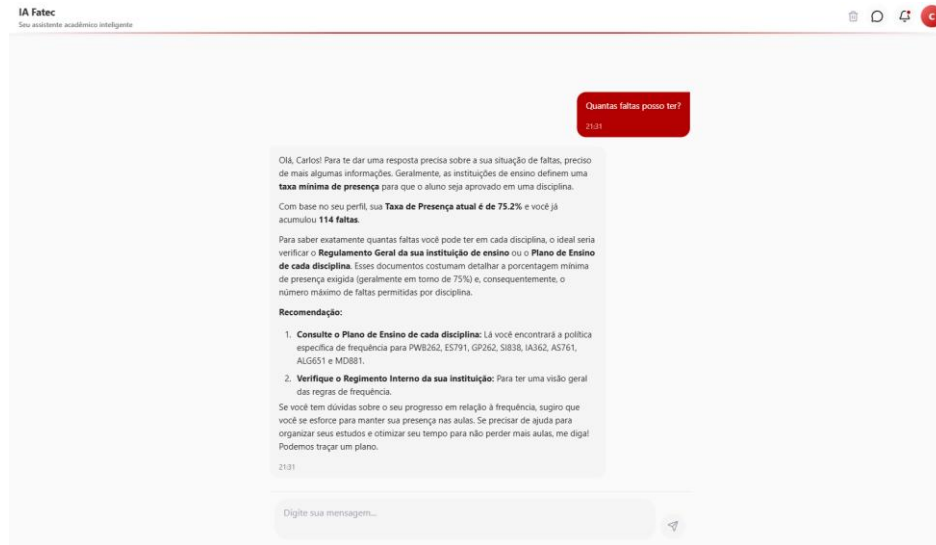


Fonte: Elaborado pelos autores (2026).

A interface de detalhes (Figura 5) apresenta o detalhamento de notas fracionadas (P1, P2, T e P3), a frequência calculada e a ementa de aulas com contatos diretos do professor responsável pela disciplina.

Para o suporte pedagógico interativo e orientação de estudos personalizada, o discente dispõe da ferramenta de chat inteligente baseada na IA do Google Gemini, exibida na Figura 6.

Figura 6 – Assistente Virtual de Aprendizagem (Chat IA)

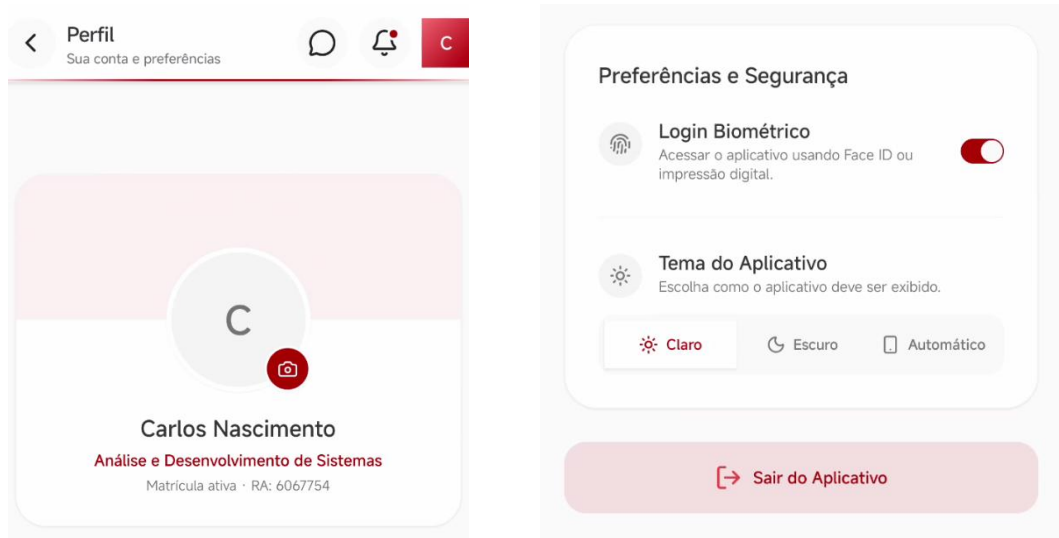


Fonte: Elaborado pelos autores (2026).

O módulo de Chat (Figura 6) exibe a interface de conversação por streaming, onde o estudante pode sanar dúvidas e receber resumos sobre o conteúdo de suas matérias a partir de prompts enriquecidos com seu histórico acadêmico personalizado.

Por fim, o controle de preferências estéticas — incluindo a seleção de tema claro, escuro ou automático — e os metadados de identificação discente encontram-se na tela de Perfil, ilustrada na Figura 7.

Figura 7 – Tela de Perfil e Configurações de Acessibilidade



Fonte: Elaborado pelos autores (2026).

A tela de Perfil (Figura 7) permite o upload e alteração da foto do usuário — integrada com o Cloudflare R2 —, a alternância do tema do aplicativo (Claro, Escuro ou Automático, seguindo as configurações do sistema operacional) e a ativação do login biométrico.

### **3 CONSIDERAÇÕES FINAIS**

#### **3.1 Principais conclusões**

Conclui-se que o projeto AppFatec cumpriu seus objetivos ao disponibilizar um aplicativo móvel eficiente para a gestão acadêmica. A aplicação soluciona problemas históricos de usabilidade dos portais institucionais tradicionais em telas de smartphones, oferecendo acesso rápido a notas, faltas e horários.

A integração do modelo de linguagem em larga escala Google Gemini agregou valor pedagógico à aplicação, permitindo que os alunos tirem dúvidas acadêmicas contextualizadas por seus dados em tempo real. A fluidez da interface, auxiliada pelo uso do React Native Reanimated para processamento de animações no thread nativo de renderização, garante uma navegação ágil e ergonômica. Além disso, a pesquisa de campo realizada com 94 discentes confirmou a relevância do projeto: a consulta de notas e faltas foi apontada por 91,5% dos respondentes como a funcionalidade mais importante, validando a priorização do desenvolvimento dessas interfaces. A alta receptividade a notificações automáticas de notas (94,7% de concordância positiva) e ao assistente virtual (73,4% de concordância positiva) reforça o alinhamento do produto de software construído com as necessidades do público-alvo.

#### **3.2 Recomendações**

Com vistas a refinar as iterações posteriores e preservar a integridade do software, sugere-se a adoção de pipelines integrados de Entrega Contínua (CI/CD), orquestrados sob provedores de automação como GitHub Actions, abarcando tanto a API Express quanto os mecanismos do Expo Application Services (EAS). O fomento dessas infraestruturas visa resguardar o sistema de erros em produção por meio de execução automatizada de testes a cada build. Adicionalmente, recomenda-se a formalização de um processo de integração com as interfaces oficiais do SIGA, que depende de autorização institucional, a fim de eliminar a necessidade de alimentação manual do banco de dados e viabilizar a sincronização de dados discentes de forma orgânica.

#### **3.3 Limitações do estudo**

A principal limitação operacional do estudo reside na ausência de integração com os dados reais do SIGA institucional. Por se tratar de um sistema legado sem API pública disponível, o banco de dados do sistema foi populado de forma simulada no MongoDB. Esta

restrição impede a validação da ferramenta em ambiente real de produção com o volume e a diversidade de dados da base oficial de discentes.

Outra limitação diz respeito à metodologia de pesquisa de campo, que utilizou uma amostragem por conveniência (não probabilística), restringindo a generalização dos resultados. Adicionalmente, recursos de notificação push e biometria dependem de serviços específicos de infraestrutura do sistema operacional do dispositivo do usuário (como APNs e Google Cloud Messaging) e de chaves de API restritas ao ambiente de desenvolvimento acadêmico, exigindo parametrizações adicionais para distribuição em escala corporativa.

### **3.4 Perspectivas futuras**

Para iterações futuras do ecossistema AppFatec, planeja-se a implementação de serviços em segundo plano que disparem alertas inteligentes quando a frequência acumulada do discente se aproximar do limite de reprovação institucional. Também se projeta a geração direta de relatórios de desempenho acadêmico em formato PDF e a otimização contínua da navegação responsiva em telas de grandes dimensões (desktops).

O horizonte mais importante de evolução tecnológica é a homologação do aplicativo junto à administração acadêmica para permitir a integração oficial com a API do SIGA, eliminando os processos manuais de cadastro e tornando o sistema plenamente viável para adoção em larga escala na instituição.

## REFERÊNCIAS

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. ABNT NBR 6023: informação e documentação: referências: elaboração. Rio de Janeiro: ABNT, 2018.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. ABNT NBR 10520: informação e documentação: citações em documentos: apresentação. Rio de Janeiro: ABNT, 2023.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. ABNT NBR 10719: informação e documentação: relatório técnico e/ou científico: apresentação. Rio de Janeiro: ABNT, 2015.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. ABNT NBR 14724: informação e documentação: trabalhos acadêmicos: apresentação. Rio de Janeiro: ABNT, 2011.

GOOGLE. Gemini API Documentation. Disponível em: <https://ai.google.dev/>. Acesso em: 7 abr. 2026.

MITTAL, A. et al. A Comprehensive Review on Generative AI for Education. IEEE Access, 2023.

MONGODB. MongoDB Atlas Documentation. Disponível em: <https://www.mongodb.com/docs/>. Acesso em: 7 abr. 2026.

NODE.JS. Node.js Documentation. Disponível em: <https://nodejs.org/>. Acesso em: 7 abr. 2026.

REACT NATIVE. React Native · A framework for building native apps using React. Meta Platforms Inc., 2026. Disponível em: <https://reactnative.dev>. Acesso em: 7 abr. 2026.

SACCOL, A.; REINHARD, N.; SCHLEMMER, E. M-learning e U-learning: novas perspectivas da aprendizagem móvel e ubíqua. São Paulo: Pearson Education, 2013.

SILVA, A. E. de S. Análise comparativa entre os frameworks de desenvolvimento de aplicativos móveis multiplataforma. 2020. 60 f. Trabalho de Conclusão de Curso (Graduação em Sistema de Informação)- Universidade Federal do Ceará, Campus de Quixadá, Quixadá, 2020.

W3C. Web Content Accessibility Guidelines (WCAG) 2.1. 2018. Disponível em: <https://www.w3.org/TR/WCAG21/>. Acesso em: 7 abr. 2026.

## **APÊNDICE A – QUESTIONÁRIO APLICADO AOS DISCENTES DA FATEC MAUÁ**

O instrumento de coleta de dados apresentado a seguir foi elaborado pelos autores e disponibilizado em formato digital por meio da plataforma Google Forms, no mês de maio de 2026, com o objetivo de levantar a percepção dos discentes da Faculdade de Tecnologia de Mauá acerca do sistema SIGA vigente e de suas demandas quanto a um aplicativo móvel acadêmico.

### **SEÇÃO I – IDENTIFICAÇÃO E PERFIL DO RESPONDENTE**

**Questão 1.** Insira seu nome. (Campo de texto aberto)

**Questão 2.** Qual o seu curso de graduação na FATEC?

- ☐ Desenvolvimento de Software Multiplataforma (DSM)
- ☐ Gestão Empresarial
- ☐ Outro: \_\_\_\_\_

**Questão 3.** Em qual semestre do curso você se encontra matriculado?

- ☐ 1º semestre
- ☐ 2º semestre
- ☐ 3º semestre
- ☐ 4º semestre
- ☐ 5º semestre
- ☐ 6º semestre

**Questão 4.** Qual a sua faixa etária?

- ☐ 17 a 20 anos
- ☐ 21 a 25 anos
- ☐ 26 a 30 anos
- ☐ Acima de 30 anos

**Questão 5.** Qual o dispositivo predominantemente utilizado para acessar informações acadêmicas?

- ☐ Smartphone (celular)
- ☐ Notebook ou computador de mesa
- ☐ Tablet
- ☐ Utilizo todos os dispositivos com frequência equivalente

## SEÇÃO II – AVALIAÇÃO DO SISTEMA SIGA ATUAL

**Questão 6.** Com que frequência você acessa o SIGA?

- ☐ Diariamente
- ☐ De duas a quatro vezes por semana
- ☐ Uma vez por semana
- ☐ De duas a três vezes por mês
- ☐ Raramente ou nunca

**Questão 7.** Ao acessar o SIGA, qual dispositivo é utilizado preferencialmente?

- ☐ Sempre o celular
- ☐ Geralmente o celular
- ☐ Geralmente o computador
- ☐ Sempre o computador
- ☐ Indiferente

**Questão 8.** Quais informações você consulta com maior frequência no SIGA? (Selecione até 3 alternativas)

- ☐ Notas e médias parciais
- ☐ Faltas e frequência
- ☐ Horário de aulas
- ☐ Calendário acadêmico
- ☐ Documentos e declarações
- ☐ Comunicados institucionais
- ☐ Outras informações

**Questão 9.** Como você avalia a facilidade de uso do SIGA em dispositivos móveis?

- ☐ Excelente
- ☐ Bom
- ☐ Regular
- ☐ Ruim
- ☐ Péssimo

**Questão 10.** Como você avalia o tempo de resposta e a velocidade de carregamento do SIGA?

- ☐ Excelente
- ☐ Bom
- ☐ Regular
- ☐ Ruim
- ☐ Péssimo

**Questão 11.** Como você avalia o design visual e a interface gráfica do SIGA?

- ☐ Excelente
- ☐ Bom
- ☐ Regular
- ☐ Ruim
- ☐ Péssimo

**Questão 12.** "O SIGA atende satisfatoriamente às minhas necessidades acadêmicas cotidianas."

- ☐ Concordo plenamente
- ☐ Concordo
- ☐ Indiferente
- ☐ Discordo parcialmente
- ☐ Discordo plenamente

**Questão 13.** "Consigo localizar com facilidade e rapidez as informações que procuro no SIGA."

- ☐ Concordo plenamente
- ☐ Concordo
- ☐ Indiferente
- ☐ Discordo parcialmente
- ☐ Discordo plenamente

**Questão 14.** "O SIGA apresenta funcionamento adequado quando acessado por meio de dispositivos móveis."

- ☐ Concordo plenamente
- ☐ Concordo
- ☐ Indiferente
- ☐ Discordo parcialmente
- ☐ Discordo plenamente

**Questão 15.** Qual a principal dificuldade enfrentada por você ao utilizar o SIGA? (Selecione todas as que se aplicam)

- ☐ Interface não adaptada adequadamente a dispositivos móveis
- ☐ Dificuldade em localizar informações (navegação confusa)
- ☐ Necessidade frequente de reautenticação
- ☐ Ausência de notificações automáticas
- ☐ Lentidão ou travamento do sistema
- ☐ Aparência visual ultrapassada
- ☐ Não enfrento dificuldades significativas

**Questão 16.** Você já deixou de consultar informação acadêmica relevante em razão de dificuldades de uso do sistema?

- ☐ Não, jamais ocorreu
- ☐ Sim, em raras ocasiões
- ☐ Sim, em algumas ocasiões
- ☐ Sim, em diversas ocasiões

**Questão 17.** Você já deixou de tomar conhecimento de prazos ou comunicados importantes por ausência de notificação do sistema?

- ☐ Não, jamais ocorreu
- ☐ Sim, em raras ocasiões
- ☐ Sim, em algumas ocasiões
- ☐ Sim, em diversas ocasiões

### SEÇÃO III – DEMANDAS E EXPECTATIVAS PARA UM APLICATIVO MÓVEL ACADÊMICO

**Questão 18.** Caso houvesse um aplicativo oficial da FATEC para dispositivos móveis, com que frequência você o utilizaria?

- ☐ Diariamente
- ☐ Algumas vezes por semana
- ☐ Algumas vezes por mês
- ☐ Raramente
- ☐ Não utilizaria

**Questão 19.** Quais funcionalidades você considera mais relevantes em um aplicativo acadêmico? (Selecione até 3 alternativas)

- ☐ Consulta ágil de notas e faltas
- ☐ Notificações automáticas de novas notas e comunicados
- ☐ Alerta de risco de reprovação por excesso de faltas
- ☐ Visualização do horário diário de aulas
- ☐ Calculadora de média acadêmica
- ☐ Acesso a documentos e declarações
- ☐ Assistente virtual com inteligência artificial para esclarecimento de dúvidas
- ☐ Alternância entre temas claro e escuro

**Questão 20.** "Utilizaria com frequência um aplicativo que notificasse automaticamente sobre novas notas e faltas registradas."

- ☐ Concordo
- ☐ Concordo parcialmente
- ☐ Indiferente
- ☐ Discordo parcialmente
- ☐ Discordo

Questão 21. "A existência de um assistente de inteligência artificial integrado ao aplicativo, voltado ao esclarecimento de dúvidas acadêmicas, seria útil para o meu desempenho."

- ☐ Concordo
- ☐ Concordo parcialmente
- ☐ Indiferente
- ☐ Discordo parcialmente
- ☐ Discordo

**Questão 22.** Em comparação ao SIGA atual, em que medida um aplicativo móvel dedicado aprimoraria a sua experiência acadêmica?

- ☐ Traria melhoria enorme
- ☐ Traria melhoria moderada
- ☐ Não traria qualquer melhoria

**Questão 23.** Você possui alguma condição que afete a utilização de aplicativos móveis?  
(Resposta confidencial)

- ☐ Deficiência visual (baixa visão ou cegueira)
- ☐ Deficiência motora que afeta o uso da tela
- ☐ Dificuldade de leitura (dislexia ou similar)
- ☐ Nenhuma das condições anteriores

**Questão 24.** Você costuma utilizar o celular em condições ambientais adversas (luminosidade intensa, baixa luminosidade ou em deslocamento)?

- ☐ Sim, com frequência
- ☐ Sim, ocasionalmente
- ☐ Não

**Questão 25.** Qual o seu modo de exibição preferido em aplicativos móveis?

- ☐ Tema claro (sempre)
- ☐ Tema escuro (sempre)
- ☐ Automático (segue as configurações do sistema)
- ☐ Indiferente

## FORMULÁRIO DE IDENTIFICAÇÃO

| <b>DADOS DO RELATÓRIO TÉCNICO E/OU CIENTÍFICO</b> |               |                    |                       |
|---------------------------------------------------|---------------|--------------------|-----------------------|
| Título e subtítulo:                               |               | Tipo de relatório: |                       |
| Instituição:                                      |               | Data:              |                       |
| Curso:                                            |               | Nota:              |                       |
| Autor(es):                                        |               |                    |                       |
| Orientador:                                       |               |                    |                       |
| Banca avaliadora:                                 |               |                    |                       |
| Produto desenvolvido:                             |               |                    |                       |
| Especificações técnicas:                          |               |                    |                       |
| Palavras-chave/Descritores:                       |               |                    |                       |
| Edição                                            | Nº de páginas | Nº do volume/parte | Área do conhecimento: |
| Observações/notas                                 |               |                    |                       |

Fonte: Estrutura adaptada pelos autores com base na norma ABNT NBR 10719:2015.