



**Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"**

JÔNATAS FELIPE SANTOS GONÇALVES

**APLICATIVO ANDROID PARA ACESSO A INFORMAÇÕES DA
INSTITUIÇÃO**

**Assis/SP
2018**



**Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"**

JÔNATAS FELIPE SANTOS GONÇALVES

**APLICATIVO ANDROID PARA ACESSO A INFORMAÇÕES DA
INSTITUIÇÃO**

Projeto de pesquisa apresentado ao curso de Análise e Desenvolvimento de Sistemas do Instituto Municipal de Ensino Superior de Assis – IMESA e a Fundação Educacional do Município de Assis – FEMA, como requisito parcial à obtenção do Certificado de Conclusão.

**Orientando (a): Jônatas Felipe Santos Gonçalves
Orientador (a): Diomara Martins Reigato Barros**

**Assis/SP
2018**

FICHA CATALOGRÁFICA

G635a GONÇALVES, Jônatas Felipe Santos
Aplicativo android para acesso a informações da instituição /
Jônatas Felipe Santos Gonçalves. – Assis, 2018.

49p.

Trabalho de conclusão do curso (Análise e Desenvolvimento de
Sistemas). – Fundação Educacional do Município de Assis-FEMA

Orientadora: Ms. Diomara Martins Reigato Barros

1.Android 2.Mobile

CDD 005.133

APLICATIVO PARA ACESSO A INFORMAÇÕES DA INSTITUIÇÃO

JÔNATAS FELIPE SANTOS GONÇALVES

Trabalho de Conclusão de Curso apresentado ao Instituto Municipal de Ensino Superior de Assis, como requisito do Curso de Graduação, avaliado pela seguinte comissão examinadora:

Orientador: _____
Diomara Martins Reigato Barros

Examinador: _____
Alex Sandro Romeo de Souza Poletto

AGRADECIMENTOS

Agradeço acima de tudo Deus por me dar Disposição para superar todas as dificuldades.

Agradeço à Fundação Educacional do Município de Assis, por proporcionar um ensino de qualidade, na qual pude usufruir durante o curso de Análise e Desenvolvimento de Sistemas.

Agradeço à minha família, pelo apoio, carinho, amizade e ajuda nos momentos mais difíceis que enfrentei antes, durante e agora nestes momentos finais do curso para obtenção do grau de Bacharel.

Agradeço aos meus colegas de sala, pelos anos de companheirismo que foram compartilhados entre inúmeras situações, desde o início da faculdade até os dias de hoje.

Agradeço à minha namorada Leticia Trevelim, que, além do carinho incondicional, sempre me apoiou e me ajudou, do início ao término do trabalho.

Agradeço a Professora Diomara Martins Reigato Barros, por toda orientação prestada e por estar sempre à disposição nos momentos de dúvidas e dificuldades.

RESUMO

Este projeto consiste no desenvolvimento de um aplicativo para dispositivos móveis utilizando a plataforma Google Android, baseada na linguagem de programação Java. A necessidade da implantação deste aplicativo originou-se pela necessidade dos alunos da instituição obter o acesso a informações de forma mais ágil e simples. Além da plataforma Google Android foi utilizado às linguagens PHP e o banco de dados MySQL para o armazenamento dos dados. Portanto ao realizar este projeto visamos ampliar com mais facilidade o contato dos alunos com a instituição utilizando uma nova plataforma.

Palavras chave: Java, MySQL, Google Android

ABSTRACT

This project consists of the development of a mobile application using the Google Android platform, based on the Java programming language. The need to implement this application was due to the need of the students of the institution to obtain access to information in a more agile and simple way. In addition to the Google Android platform, the PHP languages and the MySQL database were used for data storage. Therefore, in carrying out this project, we aim to make it easier for students to contact the institution using a new platform.

Keywords: Java, MySQL, Google Android

LISTA DE ILUSTRAÇÕES

Figura 1: Mapa Mental	16
Figura 2: Caso de uso geral.....	18
Figura 3: Caso de Uso Professor.....	19
Figura 4: Caso de Uso Aluno.....	24
Figura 5: Caso de Uso da Instituição	31
Figura 6: Diagrama E-R.....	41
Figura 7: Diagrama de classe	42
Figura 8: EAP	43
Figura 9: Cronograma.....	Erro! Indicador não definido.
Figura 10: Tela Inicial.....	44
Figura 11: Tela Login	45
Figura 12: Tela Menu Aluno.....	46

LISTA DE ABREVIATURAS E SIGLAS

DER	ENTIDADE RELACIONAMENTO
FEMA	FUNDAÇÃO EDUCACIONAL DO MUNICIPIO DE ASSIS
IDE	AMBIENTE INTEGRADO DE DESENVOLVIMENTO
UML	UNIFIED MODELING LANGUAGE
APK	ANDROID PACKAGE
EAP	ESTRUTURA ANALITICA DO PROJETO
SGBD	SISTEMA GERENCIADOR DE BANCO DE DADOS
TI	TECNOLOGIA DA INFORMAÇÃO

SUMÁRIO

1. INTRODUÇÃO	10
1.1. OBJETIVO.....	11
1.2. JUSTIFICATIVAS.....	11
1.3. PUBLICO ALVO	12
2. TECNOLOGIAS UTILIZADAS PARA O DESENVOLVIMENTO DO PROJETO	13
2.1. JAVA	13
2.2. ANDROID STUDIO	13
2.3. MYSQL.....	14
2.4. UML.....	14
3. ANÁLISE E ESPECIFICAÇÃO DO SISTEMA	16
3.1. MAPA MENTAL.....	16
3.2. MODELAGEM DO USUÁRIO	17
3.2.1. Lista de eventos	17
3.2.2. Diagrama de Caso de Uso	18
3.2.3. Diagrama de Entidade e Relacionamento	41
3.2.4. Diagrama de Classe	42
4. ESTRUTURA DO PROJETO	43
4.1. ESTRUTURA ANALITICA DO PROJETO	43
5. INTERFACES DO SISTEMA.....	44
6. CONCLUSÃO	47
6.1 TRABALHOS FUTUROS	47
REFERÊNCIAS.....	48

1. INTRODUÇÃO

Com o aumento na quantidade de informações que circulam segundo a segundo em nosso cotidiano, as pessoas passaram a ser mais exigentes em relação a tudo que consomem, adquirem, utilizam etc.; com isso a área de TI (Tecnologia da Informação) vem se expandindo cada vez mais de forma a tentar suprir as necessidades dos usuários que só desejam encontrar tudo da forma mais fácil e pratica possível.

Embora a maioria das aplicações seja desenvolvida com uma finalidade específica ou para atender uma determinada empresa/instituição, muitos dos programadores compartilham grande parte de seus projetos com outros programadores de forma que essas aplicações sejam “reaproveitadas” não somente para um lugar específico e sim para vários lugares.

Todos os aplicativos sempre tem a função de facilitar alguma atividade dos usuários e hoje possuímos uma gama enorme de aplicativos de diversas categorias seja ela voltada a educação, alimentação, esporte, entre outras.

De inicio a Instituição escolhida para a criação e implantação do aplicativo foi a FEMA, mas futuramente poderá ser reaproveitado seu código e aplicada em qualquer outra instituição que não possua sua própria aplicação.

Para o desenvolvimento da aplicação, seria necessário o acesso às informações que a instituição possui, mas são dados sigilosos por conterem informações pessoais de cada aluno e professor; com isso será criado um modelo que representará o sistema da faculdade e com base no mesmo será desenvolvido a aplicação de forma a demonstrar a facilidade que a aplicação trará a todos.

1.1. Objetivo

O objetivo do trabalho é desenvolver uma aplicação mobile que facilite a disponibilização de informação por parte da instituição e dos professores. Facilitar o acesso das informações por parte dos alunos de forma que consiga ver suas notas, faltas, horários de aula, entre outras informações que a Instituição fornecer.

Se o processo de desenvolvimento da aplicação não acarretar em nenhum imprevisto, será incluso na aplicação um sistema de localização de setores da instituição para visitantes, facilitando assim a comunicação e movimentação do mesmo entre os setores. Fazendo assim com que a instituição passe a ter um melhor relacionamento com seus alunos e visitantes.

1.2. Justificativas

Devido à necessidade de consultar várias informações e com isso encontrar dificuldades e demora o aplicativo será desenvolvido para resolver isso, de forma que os alunos com alguns cliques tenham uma melhor experiência e agilidade com o sistema da instituição.

No sistema atual da instituição alguns documentos são disponibilizados somente de forma presencial, o que para muitos alunos fica inviável devido à correria e imprevistos do dia a dia. O aplicativo junto com a disponibilização da instituição disponibilizaria essas informações de forma on-line para o aluno eliminando assim o comparecimento do mesmo a instituição e também os gastos da instituição com papeis e impressão.

1.3. Publico Alvo

O publico a que se destina o aplicativo são todos os membros que se relacionam com a Instituição seja ele aluno, professor ou visitante de forma a proporcionar que obtenham praticidade e agilidade.

Os conhecimentos aqui apresentados poderão ajudar e contribuir em pesquisas e aplicações futuras para aqueles que desejam aprender sobre desenvolvimento Android.

2. TECNOLOGIAS UTILIZADAS PARA O DESENVOLVIMENTO DO PROJETO

Para a realização deste trabalho destacamos as tecnologias que foram utilizadas, o JAVA por ser uma linguagem robusta e dinâmica, o Banco de dados MySql por ser um Banco de dados universal e que nos fornece diversos recursos para a transferência de dados entre a instituição e a aplicação.

Na análise, a ferramenta Astah Community, que gera diagramas de caso de uso, diagramas de classe, visto que esta ferramenta auxiliará na modelagem dos dados UML.

2.1. JAVA

Java é uma linguagem de programação e uma plataforma de computação lançada pela primeira vez pela Sun Microsystems em 1995. O Java é a base para praticamente todos os tipos de aplicações em rede e é o padrão global para o desenvolvimento e distribuição de aplicações móveis e incorporadas, jogos, conteúdo baseado na Web e softwares corporativos. Com mais de nove milhões de desenvolvedores em todo o mundo, de forma eficiente, o Java permite que você desenvolva implante e use aplicações e serviços estimulantes.

(https://www.java.com/pt_BR/download/faq/whatis_java.xml)

2.2. ANDROID STUDIO

O Android Studio é o ambiente de desenvolvimento integrado (IDE) oficial para o desenvolvimento de aplicativos Android e é baseado no IntelliJ IDEA . Além do editor de código e das ferramentas de desenvolvedor avançados do IntelliJ, o Android Studio oferece ainda mais recursos para aumentar sua produtividade na criação de aplicativos Android, como:

- Um emulador rápido com inúmeros recursos.
- Um ambiente unificado para você poder desenvolver para todos os dispositivos Android.
- InstantRun para aplicar alterações a aplicativos em execução sem precisar compilar um novo APK.
- Ferramentas e estruturas de teste cheias de possibilidades
- Compatibilidade com C++ e NDK.
- E muitas outras funcionalidades.

(<https://developer.android.com/studio/intro/index.html>).

2.3. MySQL

O MySQL é um SGBD (Sistema de Gerenciamento de Banco de Dados) de código aberto mais popular do mundo. Com o seu desempenho comprovado, confiabilidade e facilidade de uso, o MySQL tornou-se a principal escolha de banco de dados para aplicativos baseados na web.

(<https://www.mysql.com/about/>)

2.4. UML

A UML (Unified Modeling Language ou Linguagem de Modelagem Unificada) é uma linguagem visual utilizada para modelar sistemas computacionais por meio do paradigma de Orientação a Objetos. Esta linguagem tornou-se, nos últimos anos, à linguagem-padrão de modelagem de software adotada internacionalmente pela indústria de Engenharia de Software.

No entanto, deve ficar bem claro que a UML não é uma linguagem de programação, e sim uma linguagem de modelagem, cujo objetivo é auxiliar o responsável pelo desenvolvimento a definirem as características do software, tais como seus requisitos,

seu comportamento, sua estrutura lógica, a dinâmica de seus processos e inclusive suas necessidades físicas em relação ao equipamento sobre o qual o sistema deverá ser implantado. Tais características são definidas por meio da UML antes do software começar a ser realmente desenvolvido. (GUEDES, 2004)

3. ANÁLISE E ESPECIFICAÇÃO DO SISTEMA

Para o desenvolvimento da aplicação proposta, foi utilizada a tecnologia UML (Linguagem de Modelagem Unificada) para análise e validação dos requisitos, onde toda a parte gráfica foi criada na ferramenta Astah. Já para a criação do DER (Diagrama Entidade Relacionamento) foi utilizado o DBDesigner.

3.1. Mapa Mental

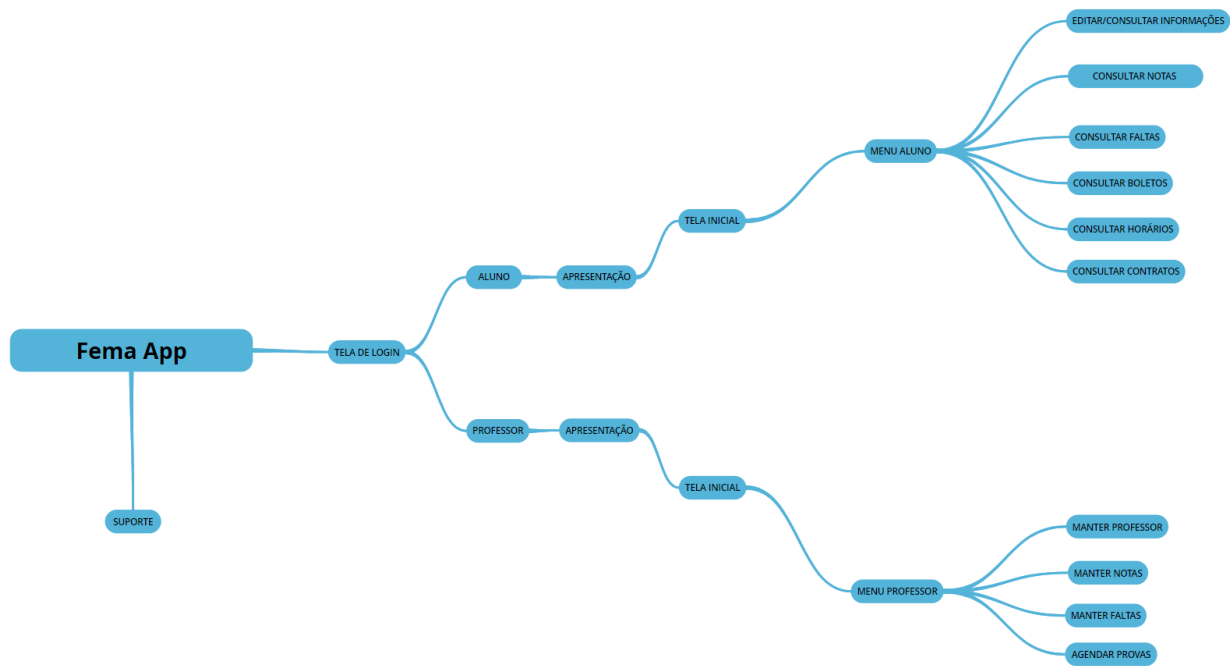


Figura 1: Mapa Mental

3.2. Modelagem Do Usuário

A modelagem do usuário especifica todas as funcionalidades que o aplicativo oferecerá ao usuário do aplicativo. Visto que o objetivo do trabalho é o desenvolvimento de uma aplicação android e o banco de dados será fornecido pela instituição, o que justifica que alguns cadastros não fazem parte desse aplicativo; esse aplicativo conterà apenas o front-end com os dados do aluno/professor.

Os únicos dados do banco que sofrerão alterações serão o manter alunos, manter professor, manter notas e faltas, agendar prova.

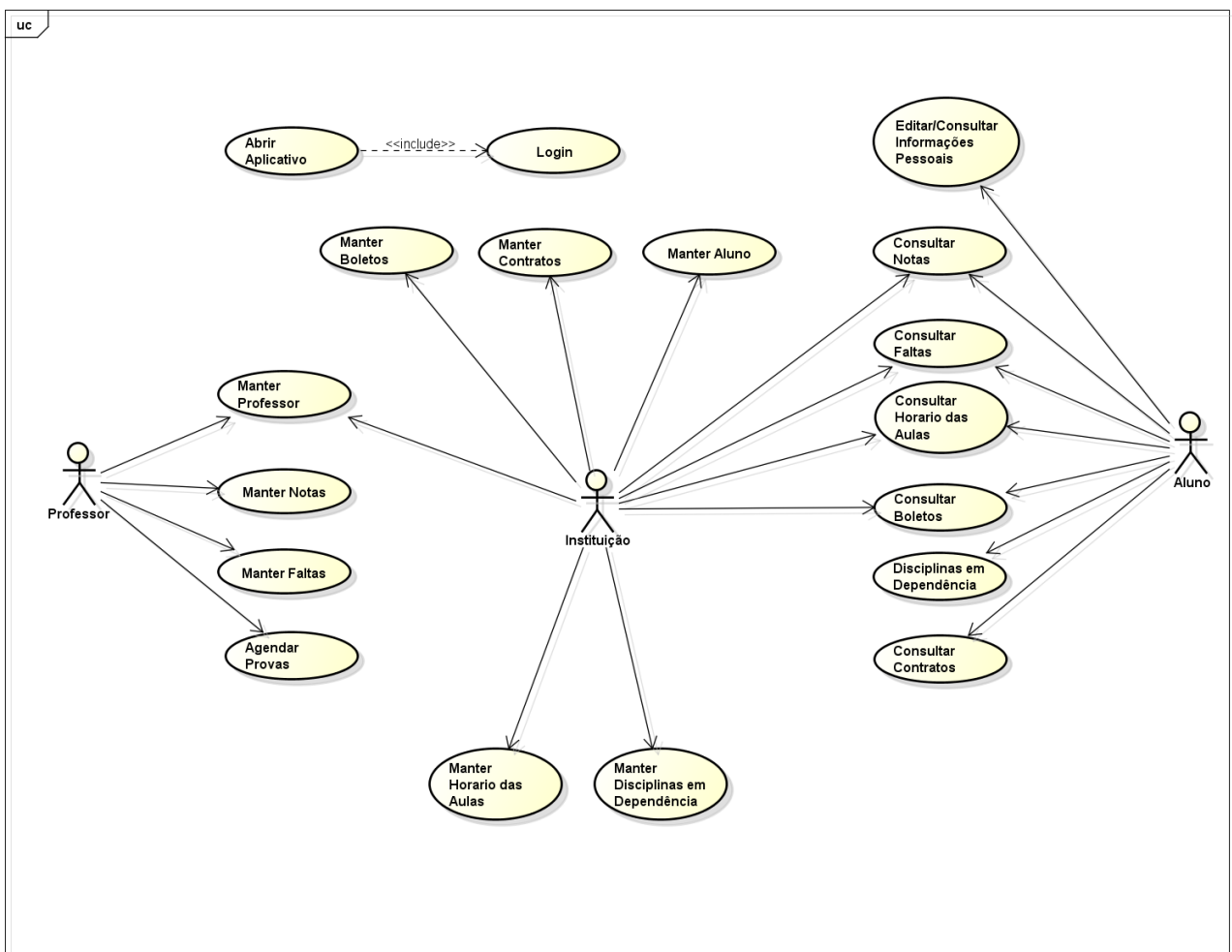
Os outros dados, por exemplo, consultar horários, consultar contratos, consultar notas e faltas, serão apenas uma consulta ao banco de dados já existente, pois não sofrerão nenhuma alteração.

3.2.1. Lista de eventos

1. Efetuar Acesso
2. Editar/Consultar Informações Pessoais
3. Consultar Notas
4. Consultar Faltas
5. Consultar Horário das Aulas
6. Consultar Boletos
7. Consultar Contratos
8. Consultar Disciplina em Dependência
9. Manter Professor
10. Manter Notas
11. Manter Faltas
12. Agendar Provas
13. Manter Boletos
14. Manter Contratos
15. Manter Aluno

- 16. Manter Disciplinas em Dependência
- 17. Manter Horário das Aulas

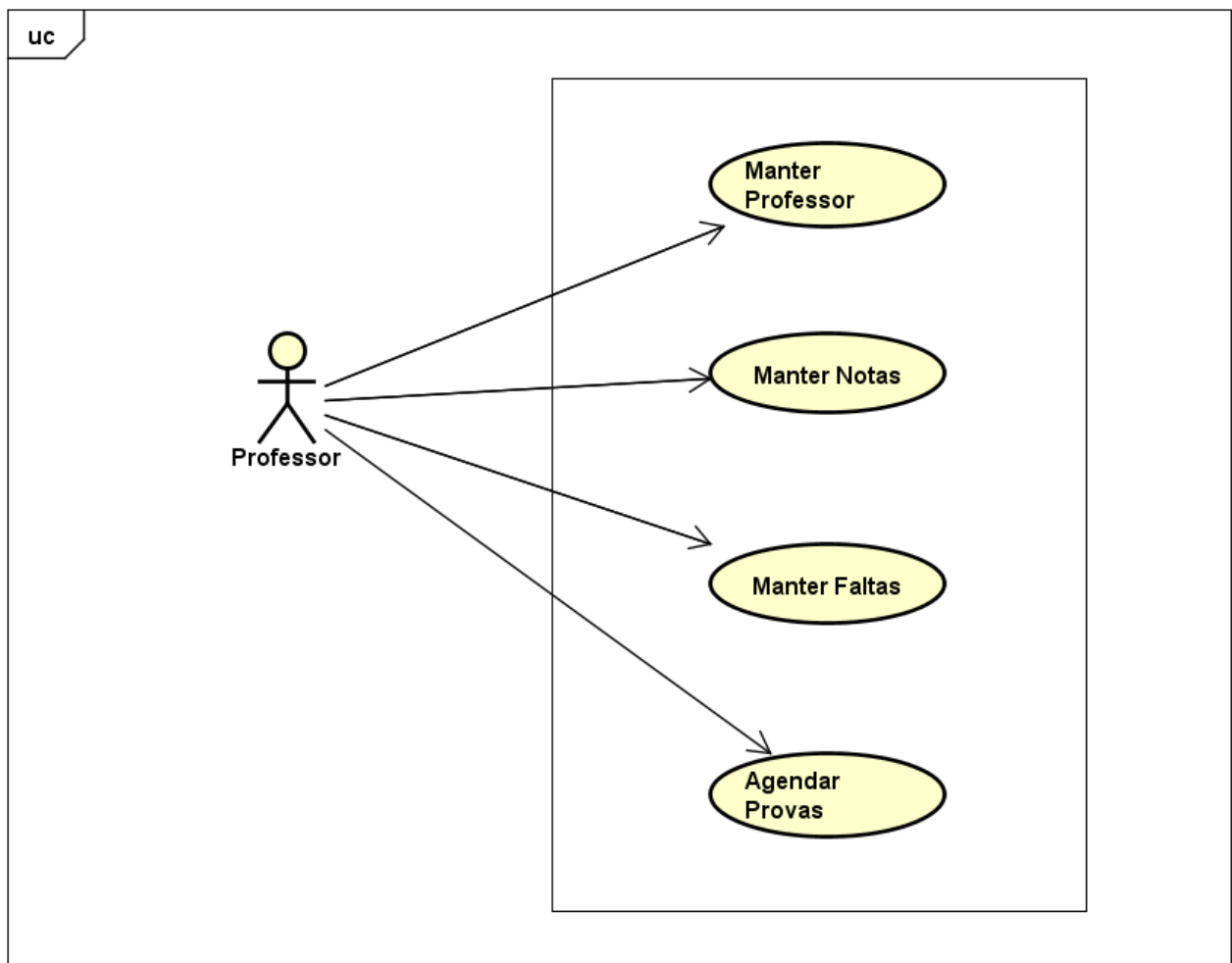
3.2.2. Diagrama de Caso de Uso



powered by Astah

Figura 2: Caso de Uso Geral

3.2.2.1 Descrição dos Casos de Usos do Ator Professor



powered by Astah

Figura 3: Caso de Uso Professor

Caso de Uso Manter Professor

Atores

Professor;

Fluxo Normal:

1. O professor inicia o caso de uso selecionando manter professor.
2. O aplicativo oferece a interface com as opções disponíveis. [A1, A2]
3. O professor escolhe fazer o logout. [A3]
4. O professor escolhe menu.
5. O aplicativo oferece a interface de menu.

Fluxo Alternativo:

A.1 – O professor escolhe alterar.

A.1.1 – O professor informa os dados a serem alterados.

A.1.2 – O aplicativo emite mensagem que a alteração foi efetuada com sucesso.

A.2 – O professor escolhe logout.

A.2.1 – O professor confirma o logout.

A.2.2 – O aplicativo encerra a sessão e fecha a aplicação.

Caso de Uso Manter Notas

Fluxo Normal:

1. O professor inicia o caso de uso selecionando manter notas.
2. O aplicativo oferece a interface com as opções disponíveis. [A1, A2]
3. O professor escolhe fazer o logout. [A3]
4. O professor escolhe menu.
5. O aplicativo oferece a interface de menu.

Fluxo Alternativo:

A.1 – O professor escolhe alterar notas.

A.1.1 – O aplicativo exibe a interface solicitando qual o curso e serie desejado.

A.1.2 – O professor insere as informações necessárias.

A.1.3 – O aplicativo exibe a lista de alunos para a alteração das notas.

A.1.4 – O professor faz as alterações nas notas desejadas.

A.1.5 – O aplicativo emite a mensagem que as notas foram alteradas com sucesso.

A.2 – O professor escolhe inserir notas.

A.2.1 – O aplicativo exibe a interface solicitando qual o curso e serie desejado.

A.2.2 – O professor insere as informações necessárias.

A.2.3 – O aplicativo exibe a lista de alunos para a inserção das notas.

A.2.4 – O professor insere as notas.

A.2.5 – O aplicativo emite a mensagem que as notas foram inseridas com sucesso.

A.3 – O professor escolhe logout.

A.3.1 – O professor confirma o logout.

A.3.2 – O aplicativo encerra a sessão e fecha a aplicação.

Caso de Uso Manter Faltas

Fluxo Normal:

1. O professor inicia o caso de uso selecionando manter faltas.
2. O aplicativo oferece a interface com as opções disponíveis. [A1, A2]
3. O professor escolhe fazer o logout. [A3]
4. O professor escolhe menu.
5. O aplicativo oferece a interface de menu.

Fluxo Alternativo:

A.1 – O professor escolhe alterar faltas.

A.1.1 – O aplicativo exibe a interface solicitando qual o curso e serie desejado.

A.1.2 – O professor insere as informações necessárias.

A.1.3 – O aplicativo exibe a lista de alunos para a alteração das faltas.

A.1.4 – O professor faz as alterações nas faltas desejadas.

A.1.5 – O aplicativo emite a mensagem que as faltas foram alteradas com sucesso.

A.2 – O professor escolhe inserir faltas.

A.2.1 – O aplicativo exibe a interface solicitando qual o curso e serie desejado.

A.2.2 – O professor insere as informações necessárias.

A.2.3 – O aplicativo exibe a lista de alunos para a inserção das faltas.

A.2.4 – O professor insere as faltas.

A.2.5 – O aplicativo emite a mensagem que as faltas foram inseridas com sucesso.

A.3 – O professor escolhe logout.

A.3.1 – O professor confirma o logout.

A.3.2 – O aplicativo encerra a sessão e fecha a aplicação.

Caso de Uso Agendar Provas

Fluxo Normal:

1. O professor inicia o caso de uso selecionando agendar provas.
2. O aplicativo oferece a interface com as opções disponíveis. [A1, A2]
3. O professor escolhe fazer o logout. [A3]
4. O professor escolhe menu.

5. O aplicativo oferece a interface de menu.

Fluxo Alternativo:

A.1 – O professor escolhe agendar data da prova.

A.1.1 – O aplicativo exibe a interface solicitando qual o curso e serie desejado.

A.1.2 – O professor insere as informações necessárias.

A.1.3 – O aplicativo exibe o calendário para escolha da data de agendamento.

A.1.4 – O professor escolhe o dia que deseja aplicar a prova.

A.1.5 – O aplicativo emite a mensagem que a prova foi agendada com sucesso.

A.2 – O professor escolhe alterar data da prova.

A.2.1 – O aplicativo exibe a interface solicitando qual o curso e serie desejado.

A.2.2 – O professor insere as informações necessárias.

A.2.3 – O aplicativo exibe o calendário com todas as provas agendadas.

A.2.4 – O professor escolhe a outra data de aplicação da prova.

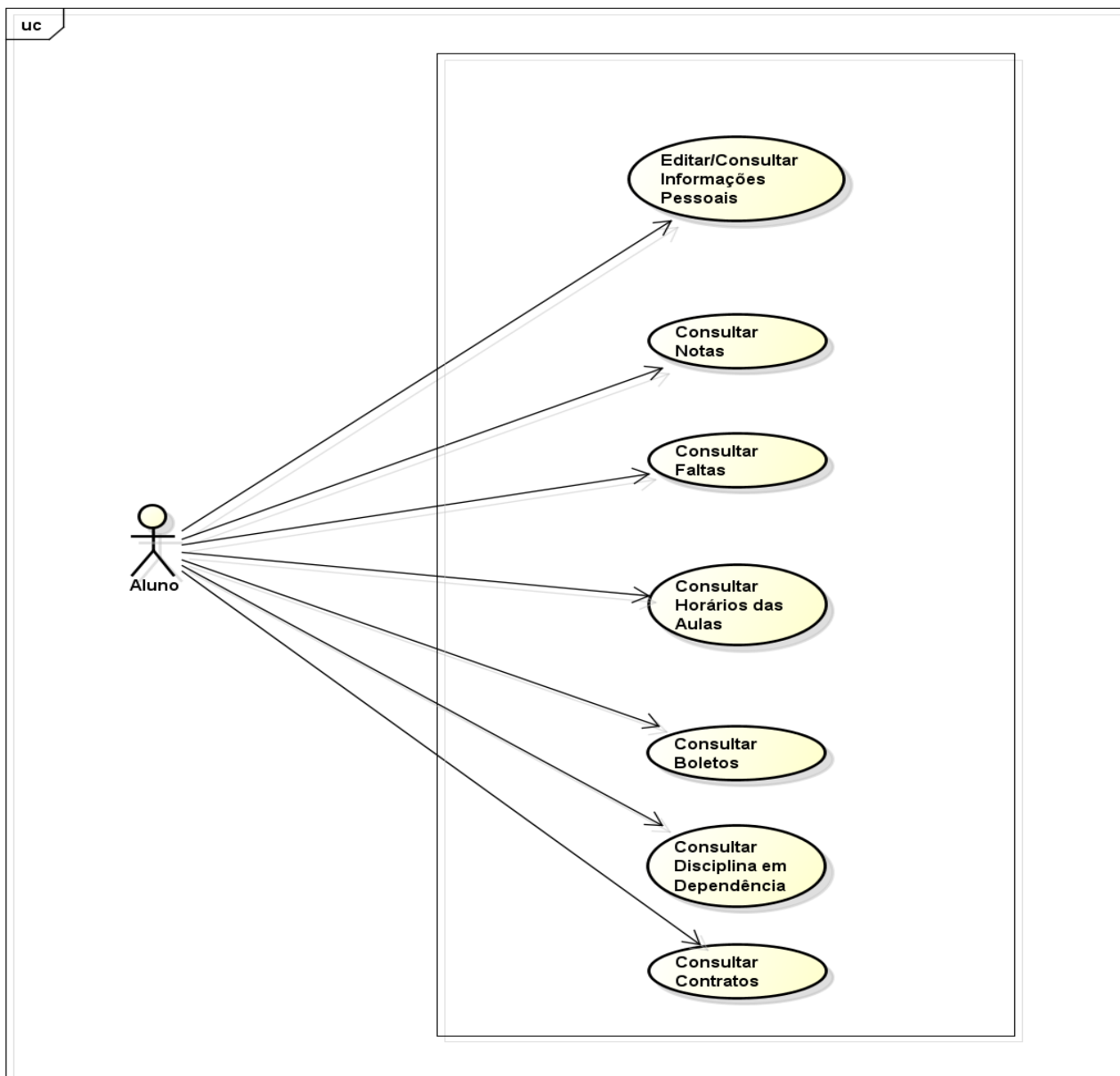
A.2.5 – O aplicativo altera a data e emite a mensagem de confirmação de alteração.

A.3 – O professor escolhe logout.

A.3.1 – O professor confirma o logout.

A.3.2 – O aplicativo encerra a sessão e fecha a aplicação.

3.2.2.1 Descrição dos Casos de Usos do Ator Aluno



powered by Astah

Figura 4: Caso de Uso Aluno

Caso de Uso Editar/Consultar informações pessoais.

Atores

Aluno;

Fluxo Normal:

1. O aluno inicia o caso de uso selecionando editar/consultar informações pessoais.
2. O aplicativo oferece a interface com as informações do aluno.
3. O aluno escolhe editar suas informações pessoais. [A1]
4. O aluno escolhe fazer o logout.[A2]
5. O aluno escolhe menu.
6. O aplicativo oferece a interface de menu.

Fluxo Alternativo:

A.1 – O aluno escolhe editar.

A.1.1 – O aplicativo libera os campos para alteração.

A.1.2 – O aluno insere as informações a serem alteradas.

A.1.3 – O aplicativo emite a mensagem de alterações realizadas com sucesso.

A.2 – O aluno escolhe logout.

A.2.1 – O professor confirma o logout.

A.2.2 – O aplicativo encerra a sessão e fecha a aplicação.

Caso de Uso Consultar notas.

Fluxo Normal:

1. O aluno inicia o caso de uso selecionando consultar notas.
2. O aplicativo confirma a existência de notas lançadas. [A1, A2]

3. O aluno escolhe fazer o logout. [A3]
4. O aluno escolhe menu.
5. O aplicativo oferece a interface de menu.

Fluxo Alternativo:

A.1 – O aplicativo consultara existência de notas lançadas.

A.1.1 – O aplicativo consulta a existência de notas lançadas para o aluno.

A.1.2 – Se e o aluno já tiver notas disponíveis o aplicativo exibe todas as notas disponíveis.

A.2 – O aplicativo consultar a existência de notas lançadas.

A.1.1 – O aplicativo consulta a existência de notas lançadas para o aluno.

A.1.2 – Se e o aluno não tiver notas disponíveis o aplicativo emite a mensagem que não existem notas disponíveis.

A.3 – O aluno escolhe logout.

A.3.1 – O aluno confirma o logout.

A.3.2 – O aplicativo encerra a sessão e fecha a aplicação.

Caso de Uso Consultar faltas.

Fluxo Normal:

1. O aluno inicia o caso de uso selecionando consultar faltas.
2. O aplicativo confirma a existência de faltas lançadas. [A1, A2]
3. O aluno escolhe fazer o logout. [A3]
4. O aluno escolhe menu.
5. O aplicativo oferece a interface de menu.

Fluxo Alternativo:

A.1 – O aplicativo consultar a existência de faltas lançadas.

A.1.1 – O aplicativo consulta a existência de faltas lançadas para o aluno.

A.1.2 – Se e o aluno tiver faltas lançadas o aplicativo exibe todas as faltas lançadas.

A.2 – O aplicativo consultar a existência de faltas lançadas.

A.1.1 – O aplicativo consulta a existência de faltas lançadas para o aluno.

A.1.2 – Se e o aluno não tiver faltas lançadas o aplicativo emite a mensagem você não possui nenhuma falta.

A.3 – O aluno escolhe logout.

A.3.1 – O aluno confirma o logout.

A.3.2 – O aplicativo encerra a sessão e fecha a aplicação.

Caso de Uso Consultar horários das aulas.

Fluxo Normal:

1. O aluno inicia o caso de uso selecionando consultar horários das aulas.
2. O aplicativo oferece a interface com o calendário de horários do aluno.
3. O aluno escolhe fazer o logout. [A1]
4. O aluno escolhe menu.
5. O aplicativo oferece a interface de menu.

Fluxo Alternativo:

A.1 – O aluno escolhe logout.

A.1.1 – O professor confirma o logout.

A.1.2 – O aplicativo encerra a sessão e fecha a aplicação.

Caso de Uso Consultar boletos.

Fluxo Normal:

1. O aluno inicia o caso de uso selecionando consultar boletos.
2. O aplicativo confirma a existência de boletos lançados. [A1, A2]
3. O aluno escolhe fazer o logout. [A3]
4. O aluno escolhe menu.
5. O aplicativo oferece a interface de menu.

Fluxo Alternativo:

A.1 – O aplicativo consultar a existência de boletos lançados.

A.1.1 – O aplicativo consulta a existência de boletos lançados para o aluno.

A.1.2 – Se e o aluno tiver boletos lançados o aplicativo exibe todos os boletos lançados.

A.2 – O aplicativo consultar a existência de boletos lançados.

A.1.1 – O aplicativo consulta a existência de boletos lançados para o aluno.

A.1.2 – Se e o aluno não tiver boletos lançados o aplicativo emite a mensagem você não possui nenhum boleto.

A.3 – O aluno escolhe logout.

A.3.1 – O aluno confirma o logout.

A.3.2 – O aplicativo encerra a sessão e fecha a aplicação.

Caso de Uso Consultar disciplinas em dependência.

Fluxo Normal:

1. O aluno inicia o caso de uso selecionando consultar disciplinas em dependência.
2. O aplicativo confirma a existência de matérias em dependência lançadas. [A1, A2]
3. O aluno escolhe fazer o logout. [A3]
4. O aluno escolhe menu.
5. O aplicativo oferece a interface de menu.

Fluxo Alternativo:

A.1 – O aplicativo consultar a existência de matérias em dependência lançadas.

A.1.1 – O aplicativo consulta a existência de matérias em dependência lançadas para o aluno.

A.1.2 – Se e o aluno tiver matérias em dependência lançadas o aplicativo exibe todos os detalhes sobre as matérias.

A.2 – O aplicativo consultar a existência de matérias em dependência lançadas.

A.1.1 – O aplicativo consulta a existência de matérias em dependência lançadas para o aluno.

A.1.2 – Se e o aluno não tiver matérias em dependência o aplicativo emite a mensagem você não possui nenhuma matéria em dependência parabéns.

A.3 – O aluno escolhe logout.

A.3.1 – O aluno confirma o logout.

A.3.2 – O aplicativo encerra a sessão e fecha a aplicação.

Caso de Uso Consultar contratos.

Fluxo Normal:

1. O aluno inicia o caso de uso selecionando consultar contratos.
2. O aplicativo oferece a interface com os contratos do aluno.
3. O aluno escolhe fazer o logout. [A1]
4. O aluno escolhe menu.
5. O aplicativo oferece a interface de menu.

Fluxo Alternativo:

A.1 – O aluno escolhe logout.

A.1.1 – O professor confirma o logout.

A.1.2 – O aplicativo encerra a sessão e fecha a aplicação.

3.2.2.3 Descrição dos Casos de Usos do Ator Instituição

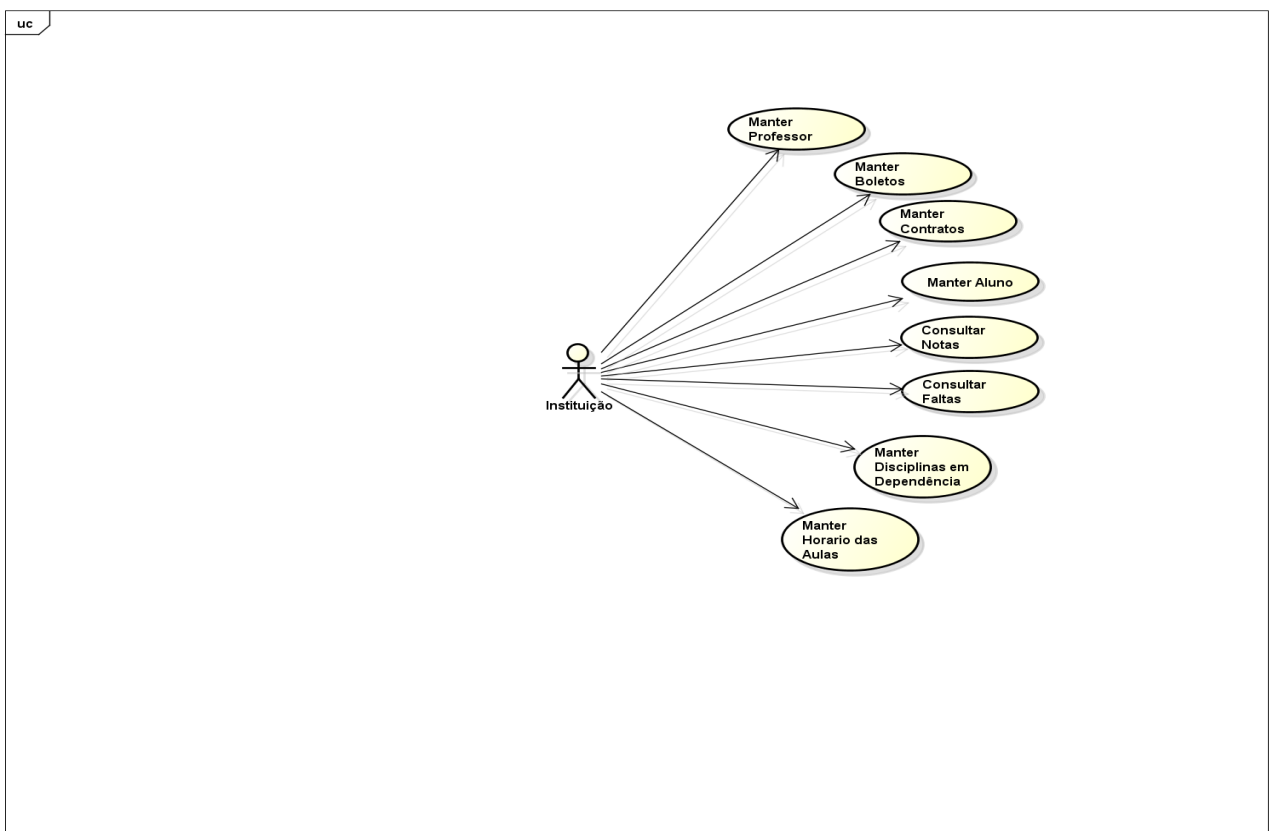


Figura 5: Caso de Uso da Instituição

Caso de Uso Manter professor.

Atores

Instituição;

Fluxo Normal:

1. A instituição inicia o caso de uso selecionando manter professor.
2. O aplicativo oferece a interface com as opções disponíveis. [A1, A2, A3]
3. A instituição escolhe fazer o logout. [A4]
4. A instituição escolhe menu.
5. O aplicativo oferece a interface de menu.

Fluxo Alternativo:

A.1 – A instituição escolhe alterar.

A.1.1 – O aplicativo exibe uma lista com os dados de todos os professores.

A.1.2 – A instituição escolhe qual professor deseja alterar os dados.

A.1.3 – O aplicativo exibe todos os dados do professor escolhido.

A.1.4 – A instituição insere os dados a serem alterados.

A.1.5 – O aplicativo emite a mensagem de dados alterados com sucesso.

A.2 – A instituição escolhe excluir.

A.2.1 – O aplicativo exibe uma lista com os dados de todos os professores.

A.2.2 – A instituição escolhe qual professor deseja excluir.

A.2.3 – O aplicativo emite uma mensagem de confirmação de exclusão.

A.2.4 – A instituição confirma a exclusão.

A.2.5 – O aplicativo emite a mensagem de professor excluído com sucesso.

A.3 – A instituição escolhe inserir.

A.3.1 – O aplicativo exibe a interface para inserção dos dados do professor.

A.3.2 – A instituição insere os dados.

A.3.3 – O aplicativo emite a mensagem de professor cadastrado com sucesso.

A.4 – A instituição escolhe logout.

A.4.1 – A instituição confirma o logout.

A.4.2 – O aplicativo encerra a sessão e fecha a aplicação.

Caso de Uso Manter boletos.

Fluxo Normal:

1. A instituição inicia o caso de uso selecionando manter boletos.
2. O aplicativo exibe as opções disponíveis. [A1, A2]
3. A instituição escolhe fazer o logout. [A3]
4. A instituição escolhe menu.
5. O aplicativo oferece a interface de menu.

Fluxo Alternativo:

A.1 – A instituição escolhe lançar novo boleto.

A.1.1 – O aplicativo exibe a interface para consulta do RA do aluno.

A.1.2 – A instituição informa os dados.

A.1.3 – O aplicativo emite a mensagem boleto adicionado com sucesso.

A.2 – A instituição escolhe consultar a existência de boletos lançados.

A.1.1 – O aplicativo consulta a existência de boletos lançados para o aluno.

A.1.2 – O aplicativo mostra todos os boletos disponíveis.

A.3 – A instituição escolhe alterar de boletos lançados.

A.3.1 – O aplicativo consulta a existência de boletos lançados para o aluno.

A.3.2 – A instituição insere os dados a serem alterados.

A.3.3 – O aplicativo emite a mensagem que o boleto foi alterado com sucesso.

A.3 – A instituição escolhe logout.

A.3.1 – O aluno confirma o logout.

A.3.2 – O aplicativo encerra a sessão e fecha a aplicação.

Caso de Uso Manter contratos.

Fluxo Normal:

1. A instituição inicia o caso de uso selecionando manter contratos.
2. O aplicativo oferece a interface com as opções disponíveis. [A1, A2]
3. A instituição escolhe fazer o logout. [A3]
4. A instituição escolhe menu.
5. O aplicativo oferece a interface de menu.

Fluxo Alternativo:

A.1 – A instituição escolhe alterar.

A.1.1 – O aplicativo exibe a interface para consulta do RA do aluno.

A.1.2 – A instituição informa os dados.

A.1.3 – O aplicativo exibe todos os contratos do aluno escolhido.

A.1.4 – A instituição faz a reimpressão do contrato já com as alterações.

A.1.5 – O aluno assina o contrato.

A.1.6 – A instituição digitaliza o novo contrato e anexa aos dados do aluno.

A.1.7 – O aplicativo emite uma mensagem de contrato atualizado com sucesso.

A.2 – A instituição escolhe inserir.

A.2.1 – O aplicativo exibe uma interface para coleta dos dados do aluno.

A.2.2 – A instituição insere os dados.

A.2.3 – O aplicativo confirma se todos os dados estão corretos.

A.2.4 – O aplicativo emite a mensagem de aluno cadastrado com sucesso.

A.2.5 – O aplicativo gera um numero de RA e Senha para futuro acesso do aluno.

A.2.6 – O aplicativo salva o contrato com os dados do aluno.

A.3 – A instituição escolhe logout.

A.3.1 – A instituição confirma o logout.

A.3.2 – O aplicativo encerra a sessão e fecha a aplicação.

Caso de Uso Manter aluno.

Fluxo Normal:

1. A instituição inicia o caso de uso selecionando manter aluno.
2. O aplicativo oferece a interface com as opções disponíveis. [A1, A2, A3]
3. A instituição escolhe fazer o logout. [A4]
4. A instituição escolhe menu.
5. O aplicativo oferece a interface de menu.

Fluxo Alternativo:

A.1 – A instituição escolhe alterar.

A.1.1 – O aplicativo exibe uma lista com os dados de todos os alunos.

A.1.2 – A instituição escolhe qual aluno deseja alterar os dados.

A.1.3 – O aplicativo exibe todos os dados do aluno escolhido.

A.1.4 – A instituição insere os dados a serem alterados.

A.1.5 – O aplicativo emite a mensagem de dados alterados com sucesso.

A.2 – A instituição escolhe excluir.

A.2.1 – O aplicativo exibe uma lista com os dados de todos os alunos.

A.2.2 – A instituição escolhe qual aluno deseja excluir.

A.2.3 – O aplicativo emite uma mensagem de confirmação de exclusão.

A.2.4 – A instituição confirma a exclusão.

A.2.5 – O aplicativo emite a mensagem de aluno excluído com sucesso.

A.3 – A instituição escolhe inserir.

A.3.1 – O aplicativo exibe a interface para inserção dos dados do aluno.

A.3.2 – A instituição insere os dados.

A.3.3 – O aplicativo emite a mensagem de professor cadastrado com sucesso.

A.4 – A instituição escolhe logout.

A.4.1 – A instituição confirma o logout.

A.4.2 – O aplicativo encerra a sessão e fecha a aplicação.

Caso de Uso Consultar notas.

Fluxo Normal:

1. A instituição inicia o caso de uso selecionando consultar notas.
2. O aplicativo confirma a existência de notas lançadas. [A1, A2]
3. A instituição escolhe fazer o logout. [A3]
4. A instituição escolhe menu.
5. O aplicativo oferece a interface de menu.

Fluxo Alternativo:

A.1 – Se houver notas lançadas.

A.1.1 – O aplicativo consulta a existência de notas lançadas para o aluno.

A.1.2 – Se e o aluno já tiver notas disponíveis o aplicativo exibe todas as notas disponíveis.

A.2 – Se não houver notas lançadas.

A.1.1 – O aplicativo consulta a existência de notas lançadas para o aluno.

A.1.2 – Se e o aluno não tiver notas disponíveis o aplicativo emite a mensagem que não existem notas disponíveis.

A.3 – A instituição escolhe logout.

A.3.1 – A instituição confirma o logout.

A.3.2 – O aplicativo encerra a sessão e fecha a aplicação.

Caso de Uso Consultar faltas.

Fluxo Normal:

1. A instituição inicia o caso de uso selecionando consultar faltas.
2. O aplicativo confirma a existência de faltas lançadas. [A1, A2]
3. A instituição escolhe fazer o logout. [A3]
4. A instituição escolhe menu.
5. O aplicativo oferece a interface de menu.

Fluxo Alternativo:

A.1 – Se houver faltas lançadas.

A.1.1 – O aplicativo consulta a existência de faltas lançadas para o aluno.

A.1.2 – Se o aluno tiver faltas lançadas, o aplicativo exibe todas as faltas lançadas.

A.2 – Se não houver faltas lançadas.

A.1.1 – O aplicativo consulta a existência de faltas lançadas para o aluno.

A.1.2 – Se o aluno não tiver faltas lançadas, o aplicativo emite a mensagem não possui nenhuma falta.

A.3 – A instituição escolhe logout.

A.3.1 – A instituição confirma o logout.

A.3.2 – O aplicativo encerra a sessão e fecha a aplicação.

Caso de Uso Manter disciplinas em dependência.

Fluxo Normal:

1. A instituição inicia o caso de uso selecionando manter disciplinas em dependência.
2. O aplicativo confirma a existência de matérias em dependência lançadas. [A1, A2]
3. A instituição escolhe fazer o logout. [A3]
4. A instituição escolhe menu.
5. O aplicativo oferece a interface de menu.

Fluxo Alternativo:

A.1 – A instituição escolhe inserir matérias em dependência.

A.1.1 – O aplicativo exibe a interface para consulta do aluno.

A.1.2 – A instituição insere os dados do aluno.

A.1.3 – O aplicativo exibe a grade de aulas do aluno.

A.1.4 – A instituição insere a disciplina em que o aluno ficou em dependência.

A.1.5 – O aplicativo emite a mensagem que a disciplina foi adicionada com sucesso.

A.2 – A instituição escolhe excluir as matérias em dependência.

A.2.1 – O aplicativo exibe a interface para consulta do aluno.

A.2.2 – A instituição insere os dados do aluno.

A.2.3 – O aplicativo exibe a grade de aulas do aluno.

A.2.4 – A instituição seleciona a disciplina a ser removida.

A.2.5 – O aplicativo emite a mensagem que a disciplina foi removida com sucesso.

A.3 – O aluno escolhe logout.

A.3.1 – O aluno confirma o logout.

A.3.2 – O aplicativo encerra a sessão e fecha a aplicação.

Caso de Uso Manter horários das aulas.

Fluxo Normal:

1. A instituição inicia o caso de uso selecionando manter horários das aulas.
2. O aplicativo oferece a interface com o calendário de horários do aluno. [A1, A2]
3. A instituição escolhe fazer o logout. [A3]
4. A instituição escolhe menu.
5. O aplicativo oferece a interface de menu.

Fluxo Alternativo:

A.1 – A instituição escolhe inserir horário.

A.1.1 – O aplicativo exibe a interface para consulta do curso e serie desejada.

A.1.2 – A instituição insere os dados.

A.1.3 – O aplicativo exibe a interface com os horários já existentes.

A.1.4 – A instituição insere os novos horários.

A.1.5 – O aplicativo emite a mensagem de horário inserido com sucesso.

A.2 – A Instituição escolhe alterar horário.

A.2.1 – O aplicativo exibe a interface para consulta do curso e serie desejada.

A.2.2 – A instituição insere os dados.

A.1.3 – O aplicativo exibe a interface com os horários já existentes.

A.1.4 – A instituição insere os horários a serem alterados.

A.1.5 – O aplicativo emite a mensagem de horários alterados com sucesso.

A.3 – A instituição escolhe logout.

A.3.1 – A instituição confirma o logout.

A.3.2 – O aplicativo encerra a sessão e fecha a aplicação.

3.2.3. Diagrama de Entidade e Relacionamento

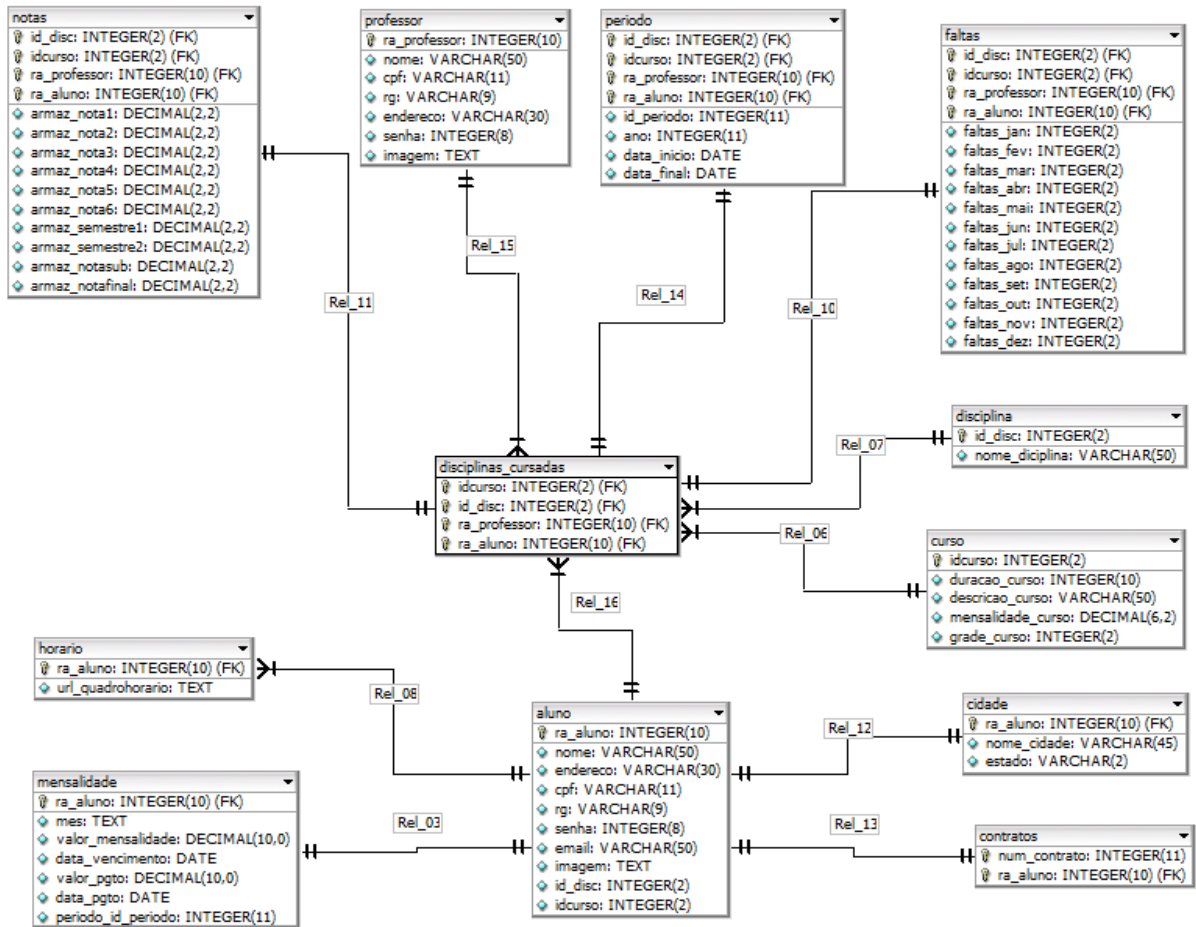


Figura 6: Diagrama E-R

3.2.4. Diagrama de Classe

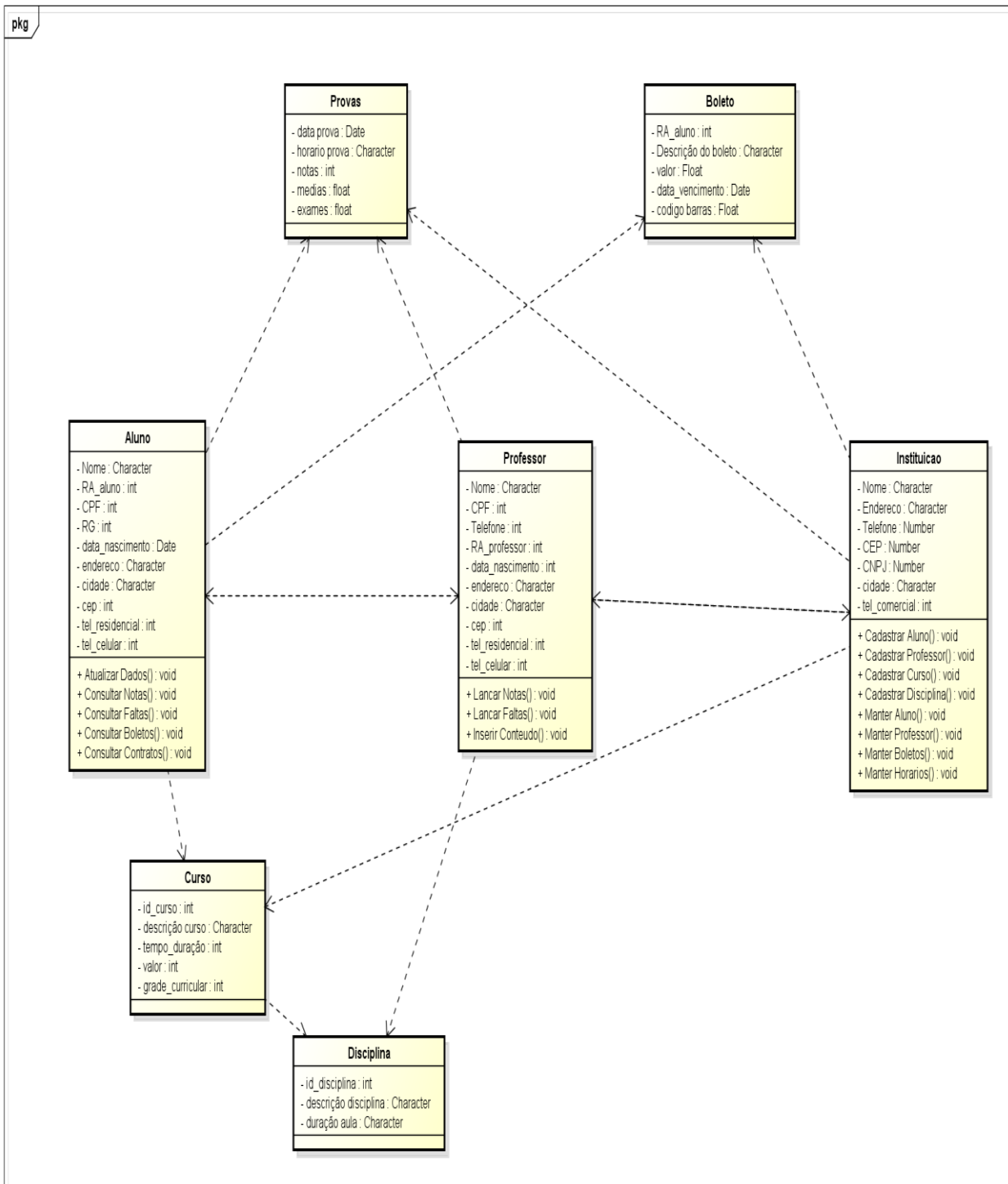


Figura 7: Diagrama de classe

4. ESTRUTURA DO PROJETO

4.1. ESTRUTURA ANALITICA DO PROJETO

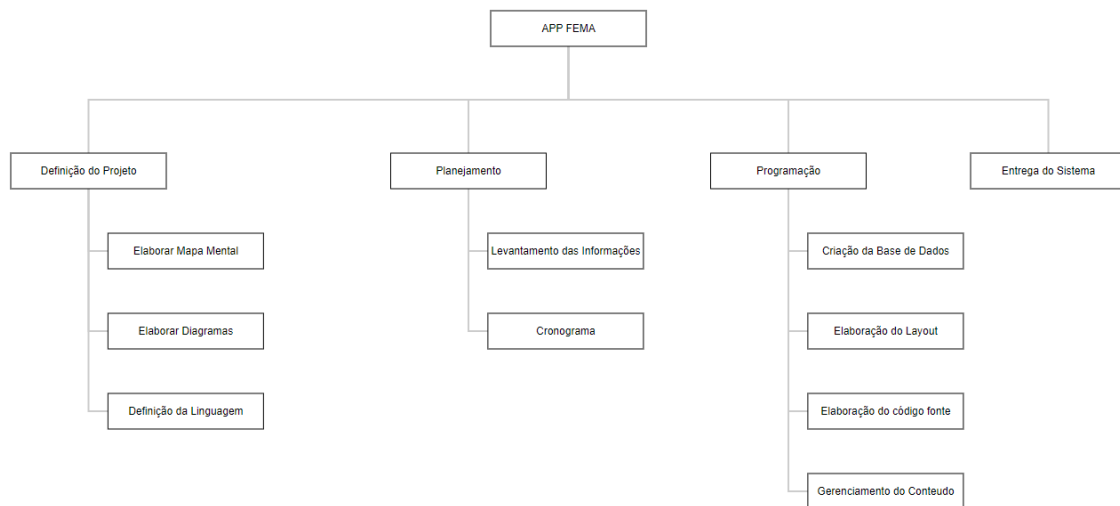


Figura 8: EAP

5. INTERFACES DO SISTEMA



Figura 9: Tela Inicial



Figura 10: Tela Login



Figura 11: Tela Menu Aluno

6. CONCLUSÃO

O projeto FemaApp surge com o intuito de auxiliar e facilitar o dia a dia dos alunos e professores que compõem a instituição de forma que o acesso às informações seja mais simples e direto, através do aplicativo os alunos poderão acessar seus dados, notas, faltas, contratos e do outro lado os professores com a possibilidade da inserção de notas, faltas.

Para que isso pudesse ser realizado houve a necessidade da utilização de diversas bibliotecas previamente desenvolvidas pelo próprio Google Android, de forma que possibilitara o acesso a BD externos com as informações requisitadas, cada etapa que era concluída no desenvolvimento foi proporcionando uma enorme experiência com a linguagem e plataforma do projeto de forma a incentivar para a conclusão do projeto.

6.1 TRABALHOS FUTUROS

Após a conclusão do projeto, pretende-se desenvolver uma atualização para que o aplicativo proporcione aos usuários visitantes uma interação com API's disponibilizadas pela Google Android, tal como a utilização do GOOGLE MAPS, desta forma possibilitando aos visitantes rotas de acesso a cada setor da instituição.

Além disso, pretende-se disponibilizar este projeto no Market do Google de forma que alunos e visitantes tenham acesso a esse aplicativo.

REFERÊNCIAS

CLARO, D. B.; SOBRAL, J. B. M. faeterj. **Faculdade de Educação Tecnológica do Rio de Janeiro**. Disponível em: <<http://www.faeterj-rio.edu.br/downloads/bbv/0031.pdf>>. Acesso em: 14 jul. 2018.

COMPUTER português. **ptcomputador**. Disponível em: <<http://ptcomputador.com/P/php-mysql-programming/91975.html>>. Acesso em: 07 jul. 2018.

CONHEÇA o Android Studio. **developer android**. Disponível em: <<https://developer.android.com/studio/intro/index.html>>. Acesso em: 04 fev. 2018.

GUEDES, G. T. A. **Uml uma Abordagem Prática**. [S.l.]: Novatec, 2004.

MARTINEZ, M. UML. **infoescola**. Disponível em: <<https://www.infoescola.com/engenharia-de-software/uml/>>. Acesso em: 19 maio 2018.

MILANI, A. **MySQL GUIA DO PROGRAMADOR**. São Paulo: Novatec, 2006.

O que é a Tecnologia Java e porque preciso dela? **Java**. Disponível em: <https://www.java.com/pt_BR/download/faq/whatis_java.xml>. Acesso em: 14 jan. 2018.

PIMENTEL, A. R. **slideshare**, 2013. Disponível em: <<https://pt.slideshare.net/fernando.palma/apostila-uml-17297299>>. Acesso em: 19 maio 2018.

SANTANA, O. G. D. devmedia, 2011. Disponível em: <<https://www.devmedia.com.br/porque-java/20384>>. Acesso em: 9 jun. 2018.

SOBRE o MySQL. **mysql**. Disponível em: <<https://www.mysql.com/about/>>. Acesso em: 10 mar. 2018.