

Desenvolvimento de um aplicativo para agendar tarefas acadêmicas em dispositivos móveis

Andrews do Amaral MELO¹; Jorge Yoshio KANDA¹

¹Universidade Federal do Amazonas

andrewsamara1@gmail.com; jorge.kanda@gmail.com

Resumo

Geralmente, a vida acadêmica para alguns alunos é bastante corrida, pois eles precisam realizar várias tarefas solicitadas pelo professor em sala de aula, além de estudar para as provas. Lembrar de todas essas atividades torna mais fácil para os alunos se eles possuírem uma agenda na qual podem armazenar todas as informações necessárias. Também, seria muito mais prático se essa agenda estivesse instalada em seu aparelho celular, visto que nos dias atuais é uma tecnologia bastante popular. Portanto, este projeto propõe o desenvolvimento de um aplicativo denominado Agenda Acadêmica para dispositivos móveis com a finalidade de facilitar o gerenciamento das atividades pertinentes na vida universitária.

Palavras-Chave: Agenda Acadêmica; Aluno; Atividades; Plataforma Android

Lista de abreviaturas esiglas

ADT: *Android Development Tools*

API: *Application Programming Interface*

AVD: *Android Virtual Device*

ED: Estrutura de Dados

ER: Entidade e Relacionamento

FA: Fluxo Alternativo

IDE: *Integrated Development Environment*

JDK: *Java Development Kit*

RF: Requisito Funcional

RN: Regras de Negócio

RNF: Requisito Não Funcional

SDK: *Software Development Kit*

UC: *Use Case*

UML: *Unified Modeling Language*

1. Introdução

A proposta desta pesquisa consiste no desenvolvimento de um aplicativo que auxilie na vida acadêmica dos alunos. Em muitas universidades, públicas ou particulares, a vida acadêmica para a maioria dos alunos é bastante agitada, independente do curso no qual eles estão matriculados.

Na maioria das vezes, eles precisam realizar diversas tarefas simultaneamente solicitadas pelos professores em sala de aula, isso sem considerar as atividades que os mesmos já desempenham em sua vida cotidiana ou pessoal (Grispino 2014).

Para conseguir lembrar-se de todas as atividades e executá-las, torna-se fundamental que estas estejam armazenadas em algum lugar seguro, pois caso sejam perdidas e não realizadas, certamente o aluno poderá ser prejudicado.

Um modo de armazenar todas as informações importantes é fazer uso de aparelhos digitais, onde existem aplicativos que podem cumprir essa função, como por exemplo: Agendas e Blocos de textos eletrônicos (Meier 2009; Deitelet *al.* 2014). No entanto, tais aplicativos podem conter muitas informações ou funções desnecessárias que só dificultam a realização do procedimento mais importante, que é o agendamento de tarefas.

Nesse contexto, a presente abordagem propõe o desenvolvimento de uma Agenda Acadêmica para dispositivos móveis a fim de contribuir de forma eficaz no auxílio dessa atividade bastante comum ao longo da vida acadêmica do aluno.

1.1 Objetivos

Para uma melhor compreensão dos objetivos deste trabalho, os mesmos foram divididos em Geral e Específicos.

1.1.1 Objetivo Geral

Desenvolver um aplicativo para agendar tarefas acadêmicas para dispositivos móveis de modo que auxilie o planejamento das atividades a serem cumpridas pelos alunos.

1.1.2 Objetivos Específicos

- ❖ Realizar levantamento bibliográfico das principais técnicas relacionadas ao desenvolvimento da aplicação;
- ❖ Realizar levantamento bibliográfico dos principais padrões de projetos para aplicações móveis;
- ❖ Implementar a abordagem proposta em computador pessoal;
- ❖ Embarcar o projeto em dispositivo móvel;
- ❖ Testar o aplicativo proposto;
- ❖ Avaliar os resultados obtidos através da abordagem proposta.

1.2 Justificativa

Com o avanço das mídias tecnológicas, ficou mais fácil realizar tarefas previstas para cada dia. Os aparelhos como os smartphones são atualmente capazes de desempenhar diversas funções, auxiliando o usuário para a realização dessas tarefas. Usando tais aparelhos, o usuário pode navegar na internet, enviar e receber e-mails, ler vários tipos de arquivos, dentre outras funções (Lecheta 2010; Gaertner 2012).

Entretanto, mesmo com as diversas ferramentas disponíveis nesses aparelhos para auxiliar o aluno em sua vida acadêmica, ainda é necessário um aplicativo que seja simples para o usuário, de fácil interação e que atenda a real necessidade de um acadêmico no que diz respeito ao agendamento de suas tarefas (Lee 2011; Burnette 2008).

Esta pesquisa propõe o desenvolvimento de uma aplicação, mais especificamente uma agenda acadêmica, para contribuir no agendamento de tarefas em dispositivos móveis, proporcionando mais praticidade e segurança para o acadêmico na realização dessa atividade.

2. Material e Métodos

Nesta seção serão apresentadas as ferramentas utilizadas no desenvolvimento da aplicação, como os requisitos: funcionais e não funcionais; as regras de negócio; o modelo de dados; e diagramas baseados na notação UML.

2.1 Ferramentas Utilizadas

O processo de desenvolvimento do projeto utiliza os seguintes equipamentos:

- ❖ Celular Samsung *Galaxy Grand Duos*, sistema operacional Android 4.2.2;
- ❖ *Notebook SEMP TOSHIBA*, com processador *Intel Dual Core (2.20 GHz)* e 2 GB de memória RAM.

Para a implementação da aplicação foram utilizados os seguintes *softwares*:

- ❖ Sistema Operacional *Windows 7 Ultimate* para processadores de 32 bits;
- ❖ *IDE Eclipse* versão: 4.2.1,(c) *Copyright Eclipse*;
- ❖ *ADT Android Build: v22.3.0-887826, Copyright (c) The Android Open Source Project* - (Java SDK, SQLite).

2.2 Requisitos

O protótipo que está sendo desenvolvido deverá ser portado em um dispositivo móvel, *smartphones*, compatíveis com o sistema Android. O sistema proposto deverá agendar e notificar atividades acadêmicas ao usuário.

O usuário deverá cadastrar alguma tarefa no sistema da aplicação, que ficará responsável por lembrá-lo quando a atividade estiver próxima emitindo um alerta na

barra de notificações do aparelho. Ao clicar na notificação recebida, o usuário abrirá a tela, que irá mostrar a atividade agendada e as informações especificadas por ele no momento do cadastro.

A Figura 1 exibe a estrutura inicial do protótipo do sistema com as suas principais telas:

- a): Tela de Calendário da aplicação;
- b): Tela de Cadastro da Atividade;
- c): Tela de Atividades Agendadas.

A aplicação deve ser compatível com dispositivos móveis com a plataforma Android 3.2 ou superior, permitindo que usuários com aparelhos com essas configurações possam utilizar sem problemas os recursos oferecidos pela aplicação.

2.2.1 Requisitos Funcionais

Nesta fase, é feito um levantamento de requisitos necessários para a implementação do sistema. Esse processo consiste em entender a natureza do software a ser construído. Além disso, o domínio do problema deve ser compreendido, bem como as funcionalidades e o comportamento esperado (Falbo 2005).

Na Tabela 1 são listados os requisitos funcionais do sistema. Os requisitos foram selecionados e descrevem as funcionalidades e quais atributos devem ser implementados no protótipo utilizado neste projeto.

Os requisitos são categorizados com níveis de prioridades, a saber:

Alta: A falha em implementar este tipo de requisito pode fazer com que o sistema não atenda às necessidades dos *stakeholders* (pessoas envolvidas no projeto).

Média: Requisitos importantes para a eficiência de muitas das funcionalidades da aplicação.

Baixa: Requisito não tão importante, e de utilização menos frequente pelo usuário.

2.2.2 Requisitos Não Funcionais

Esse tipo de requisito é caracterizado pela restrição das funções que o sistema deve apresentar, tais como restrições de tempo, de uso de recursos entre outras. Dependendo do cenário do projeto, os requisitos não funcionais podem ser classificados de diferentes maneiras, tais como requisitos de desempenho, requisitos de portabilidade, requisitos legais, requisitos de conformidade etc. (Sommerville 2003).

A Tabela 2 descreve as restrições necessárias para o desenvolvimento do sistema em questão, esses requisitos também utilizam os critérios de prioridade citados na seção anterior.

2.2.3 Regras de Negócio

Na tabela 3 são mostradas as regras de negócio do sistema, que servem para definir o conjunto de instruções que os usuários já seguem e a aplicação deve seguir.

2.3 Modelagem de Dados

A atividade da Análise Essencial de Sistemas é sem dúvida primordial para o processo de modelagem de dados e, para tal, o mais utilizado é o modelo de Entidades e Relacionamento (ER). O modelo organiza os dados a serem armazenados em um sistema, tendo sido criada para dar suporte a projetos que utilizam banco de dados (Koscianski 2007).

O diagrama de Entidades e Relacionamentos representa as entidades do mundo real, que pode ser a representação de um ser, um objeto, um organismo social etc., e os relacionamentos entre elas, que um sistema de informação precisa simular internamente (Falbo 2005). As Figuras 2 e 3 mostram a estrutura do sistema por meio do modelo ER.

Essa estrutura do sistema apresenta um total de oito entidades, sendo as principais para a elaboração do projeto as entidades *Aluno* e *Atividade*.

A entidade *Aluno*, assim como seu nome sugere, representa a base de dados do aluno, sendo este, o usuário responsável em agendar tarefas acadêmicas na aplicação. Essa entidade se relaciona com outras duas entidades, a entidade *Instituição* e a entidade *Curso*. A Instituição representa a universidade que o aluno pertence, e o Curso é o nome do curso no qual ele está matriculado.

Outra entidade principal é a *Atividade*, que representa a estrutura de armazenamento das atividades agendadas na base de dados. Essa entidade se relaciona com outras quatro entidades e os nomes são bem sugestivos quanto à função de cada uma delas, a saber:

Disciplina: Representa a disciplina referente àquela atividade;

Tipo_Atividade: Descreve o tipo de cada atividade, podendo ser um exercício, prova, trabalho entre outras;

Status_Atividade: A atividade agendada pode estar ativa, cancelada, ou já ter acontecido.

Agenda_Notificação: Representa os dados como data e hora que o aluno escolhe para ser lembrado sobre alguma atividade agendada.

2.4 Diagrama UML – Casos de Uso

De acordo com Koscianski (2007), a UML (Unified Modeling Language) é uma linguagem gráfica de modelagem, semiformal e orientada a objetos muito bem aceita na indústria de software. É um padrão mundial que permite a modelagem em várias fases de desenvolvimento de um sistema, desde a maior abstração, como os requisitos, até a geração de códigos.

O diagrama de Casos de Uso permite descrever a visão externa do sistema e suas interações com o mundo exterior. Conforme ilustra a Figura 4, são apresentados os principais requisitos funcionais, os atores que interagem com o sistema, podendo ser pessoas, máquinas ou outros sistemas, e os relacionamentos, que são mensagens trocadas entre os atores (Pressman 2002; Koscianski 2007).

2.4.1 Especificações de Casos de Uso

Esta seção é uma descrição típica do Caso de Uso do sistema, e contém a identificação do ator que iniciou o caso de uso, o pré-requisito do caso de uso, caso haja algum, e a descrição textual do fluxo normal ou fluxos alternativos, se houver. A seguir é descrita a estrutura do caso de uso:

Nome

Agenda Acadêmica.

Descrição sucinta

O aluno agenda uma ou mais atividades.

Atores

Aluno.

Pré-condições

Na primeira execução da aplicação o Aluno deve ter feito o cadastro.

Fluxo básico – Cadastrar Atividade

1. O Aluno seleciona a opção “Agendar Atividade”;
2. O sistema exibe a tela “Calendário”;
3. O Aluno escolhe a data da Atividade;
4. O sistema exibe a tela “Agendar Atividade” com as possíveis atividades;
 - 4.1 O Aluno escolhe a categoria da atividade, (Prova, Trabalho, Exercício, Evento ou Outra Atividade);
 - 4.2 A disciplina a qual a atividade é referente;
 - 4.3 A data de notificação para ser lembrado;
 - 4.4 A descrição da atividade.
5. O Aluno seleciona a opção “Confirmar”;

6. O sistema exibe a Tela de Atividades Agendadas;
7. O caso de uso é encerrado.

Fluxos alternativos

FA1. Alternativo ao passo 5 – Informações incorretas

- a.1 O Aluno não preenche todas as informações de agendamento da atividade;
- a.2 O sistema retorna ao Passo 4;

Fluxo básico - Editar cadastro

1. O Aluno seleciona a opção 'Meu Perfil';
2. O sistema exibe a tela 'Meu Perfil';
3. O Aluno confirma cadastro;
4. O caso de uso é encerrado.

Fluxos alternativos

FA2. Alternativo ao passo 3 – Informações incorretas

- a.1 O Aluno não preenche todos os campos necessários;
- a.2 O sistema retorna ao passo 2;

Fluxo básico - Ver Agenda

1. O Aluno seleciona a opção 'Minha Agenda';
2. O sistema exibe a tela 'Minha Agenda';
 - 2.1 O Aluno escolhe a opção visualizar por 'Atividade', 'Disciplina' ou 'Data'.
3. O caso de uso é encerrado

Estruturas de dados

(ED1 – Atividade Agendada)

- 1.1 Tipo da atividade
- 1.2 Descrição da atividade
- 1.3 Disciplina da atividade
- 1.4 Data da atividade
- 1.5 Hora da atividade
- 1.6 Data para ser lembrado
- 1.7 Hora para ser lembrado

Regras de negócio

RN001 O usuário deve se cadastrar para acessar a aplicação;

RN002 Para ser notificado o usuário deve agendar a tarefa;

RN003 O sistema deve atualizar a lista de tarefas, para cada tarefa notificada;

RN004 O usuário pode escolher ser lembrado da atividade, por exemplo, momentos antes de a atividade acontecer.

2.5 Arquitetura do Sistema

Nesta seção será apresentada a arquitetura do protótipo para agendar tarefas acadêmicas e atender as funcionalidades citadas nas seções anteriores. A Figura 5 ilustra a arquitetura do projeto.

A aplicação apresenta uma estrutura bem simples, sendo utilizados recursos para armazenar dados do aluno, dados das atividades e recursos de notificação, a estrutura foi dividida em usuário e uma estrutura responsável em realizar a tarefa de persistência de dados no dispositivo móvel. Dessa forma, para a tarefa de armazenamento de dados foi utilizado o *SQLite*, um gerenciador de banco de dados nativo do *Android*.

O usuário realiza a tarefa de inserir e manipular as informações na base de dados do sistema, que por sua vez é responsável em retornar os eventos necessários e exibir a notificação ao usuário.

2.6 Agenda Acadêmica – Estrutura do Projeto

O projeto está organizado em pacotes, dessa forma, facilita a visualização da função de cada pacote na estrutura do protótipo. Até o momento foram desenvolvidos os seguintes métodos e seus pacotes:

br.com.agenda.activity: Nesse pacote estão as classes referentes as telas da aplicação, e auxiliam na interação do usuário com a aplicação.

br.com.agenda.modelo: Contém os objetos que recebem os valores referentes a base de dados da aplicação.

br.com.agenda.dao: Esse pacote contém a estrutura do banco de dados SQLite do projeto, nele estão contidos os métodos de criação e manipulação do banco de dados.

2.7 Resultados Parciais

Nesta seção serão apresentadas as funcionalidades desenvolvidas até o momento da aplicação, assim como os recursos utilizados pelo protótipo.

2.7.1 Banco de dados SQLite

Inicialmente, as classes referentes à criação do banco de dados foram implementadas, uma vez que será necessário o armazenamento de dados no dispositivo móvel. Esses métodos são responsáveis em gerenciar as operações de inserção, atualização, pesquisa e exclusão de dados na base de dados.

2.7.2 Modelo de dados

As classes de modelagem de dados também foram implementadas, sendo as responsáveis pela manipulação do valor das variáveis que serão persistidos no banco de dados.

2.7.3 Tela de cadastro de atividades

A prototipação da tela de agendamento de atividades é primordial, pois é a partir dela serão refinados e aprimorados os requisitos da aplicação. O protótipo pode ser observado na Figura 6.

O usuário será capaz de agendar tarefas clicando em uma data na tela do calendário da aplicação, que o redirecionará para a tela de cadastro de atividades. A tela de cadastro apresentará a data em que ocorrerá a atividade, as opções de atividades, assim como as

opções de agendar hora e data para ser lembrado, a disciplina pertencente àquela atividade e a descrição da atividade.

Para concluir o cadastramento da atividade, o usuário escolherá a opção de salvar, esses dados serão salvos na base de dados do dispositivo, ou ele poderá escolher a opção de cancelar cadastramento.

2.7.4 Sistema de notificação

A plataforma Android disponibiliza alguns recursos, um deles é o sistema de notificações, que são eventos que disparam uma mensagem na tela do dispositivo com o intuito de chamar a atenção do usuário para aquele evento. Um exemplo de notificação é quando chegam mensagens no aparelho móvel e o usuário decide visualizar ou ignorar.

Ao ser visualizada, a *Intent* configurada pode uma abrir uma nova *Activity* ou pode ser usada para iniciar um serviço (MEIER 2009).

A Figura 7 mostra a notificação de um evento agendado em um determinado horário, a mensagem aparece na barra superior do dispositivo, chamando a atenção do usuário.

O sistema ainda está na fase inicial de desenvolvimento, por isso não foram realizados testes com usuários. A aplicação foi validada primeiramente no emulador ou AVD, disponibilizado pelo Android SDK, que é um sistema capaz de simular um dispositivo real, porém com algumas limitações, por exemplo, otimização de tempo no processo de emular a aplicação.

2.8 Considerações Finais

O projeto corresponde a uma alternativa para auxiliar os universitários a organizar seus horários de atividades ou não esquecer na realização de uma tarefa importante. O projeto está na primeira fase de implementação do sistema. Após a conclusão dessa etapa serão realizados testes funcionais do sistema com os usuários.

A realização dos testes irá consistir por meio da análise da aplicação. Para isso será elaborado um questionário com perguntas referentes ao sistema, a fim de que seja possível realizar um levantamento de informações da aceitação ou rejeição por parte dos usuários.

A aplicação de questionário é importante, pois é através dela que será analisado o grau de satisfação da experiência do usuário com a manipulação da aplicação no aparelho móvel. Portanto, a partir dos resultados obtidos serão construídos gráficos que servirão como base para análise e assim concluir os testes para apresentação no relatório final.

Agradecimentos

Agradecimentos ao professor Jorge Y. Kanda e a FAPEAM pelo incentivo a iniciação científica e suporte financeiro ao desenvolvimento da presente projeto.

Bibliografia

- Burnette, Ed. Hello, **Android: Introducing Google's Mobile Development Platform**. Pragmatic Bookshelf, 2008.
- Deitel, Harvey; Deitel, Abbey; Deitel, Paul J; Morgano, Michael. **Android Para Programadores - Uma Abordagem Baseada Em Aplicativos**. Editora: Bookman, 2012.
- Falbo, Ricardo de Almeida, Engenharia de Software**, UFES - Universidade Federal do Espírito Santo, 2005.
- I. Sommerville, **Engenharia de Software**, São Paulo: Addison-Wesley, 6ª edição, 2003.
- Koscianski, André, **Qualidade de Software: aprenda as metodologias e técnicas mais modernas para o desenvolvimento de software/** André Koscianski, Michel dos Santos Soares. – 2 ed. São Paulo, Novatec Editora, 2007.
- Lecheta, R. R. **Google Android: Aprenda a criar aplicações para dispositivos móveis com o Android SDK**. Novatec, 2ª edição, 2010.
- Lee, Wei-Meng. **Introdução ao Desenvolvimento de Aplicativos para o Android** Editora: Ciência Moderna, 2011.
- Meier, Reto. **Professional Android Application Development**. Indianapolis: Wiley Publishing, 2009, 1ª. ed.
- R.S. Pressman, **Engenharia de Software**, Rio de Janeiro: McGraw Hill, 5ª edição, 2002.
- Gartner. **Smartphone Operating System Market by Year-End 2012** (<http://www.gartner.com/it/page.jsp?id=1622614>). Acesso em 14/04/2014
- Grispino, Izabel Sandalla. **Crianças com sobrecarga de atividades**. (http://www.izabelsadallagrispine.com.br/index.php?option=com_content&view=article&id=1159). Acesso em 13/04/2014.

TABELAS

Tabela 1. Descrição dos Requisitos Funcionais da aplicação.

Identificador	Descrição	Prioridade	Requisitos Relacionados
RF001	O usuário deve cadastrar-se antes de usar a aplicação	Alta	
RF002	A aplicação deve prover mecanismos para agendar atividades, trabalhos, provas e eventos.	Alta	
RF003	Disponibilizar um mecanismo de armazenamento de dados	Alta	
RF004	Exibir uma lista com as tarefas agendadas	Alta	RF002
RF005	A lista de tarefas agendadas deve estar discriminada por: Atividade, Disciplina, Data.	Alta	RF002, RF004
RF006	Conter a função de busca para as atividades agendadas	Alta	
RF007	A aplicação deve notificar o usuário das tarefas agendadas	Alta	
RF008	O usuário contará com os recursos de data e hora para agendar as tarefas	Alta	RF007

*RF: Requisito Funcional

Tabela 2. Descrição dos Requisitos Não Funcionais da aplicação.

Identificador	Descrição	Categoria	Prioridade	Requisitos Relacionados
RNF001	Fornecer uma interface de fácil manipulação pelo usuário destinado à aplicação (Alunos).	Usabilidade	Alta	RNF006
RNF002	Tempo de resposta do aplicativo não deve ultrapassar o tempo de 2 segundos.	Desempenho	Alta	RNF001
RNF003	Deve rodar na plataforma Android, versão 3.2 ou superior.	Interoperabilidade	Alta	RNF004
RNF004	A aplicação deve contar com o uso do banco de dados nativo do sistema operacional Android o SQLite.	Interface	Média	RNF003
RNF005	Fornecer um sistema de autenticação do usuário	Segurança	Média	
RNF006	Idioma em Português-BR	Usabilidade	Média	

*RNF: Requisito não Funcional

Tabela 3. Descrição das Regras de Negócio da aplicação.

Identificador	Descrição
RN001	O usuário deve se cadastrar para acessar a aplicação.
RN002	Para ser notificado o usuário deve agendar a tarefa.
RN003	O sistema deve atualizar a lista de tarefas, para cada tarefa notificada.
RN004	O usuário pode escolher ser lembrado da atividade, por exemplo, momentos antes de a atividade acontecer.

*RN: Regras de Negócio

FIGURAS

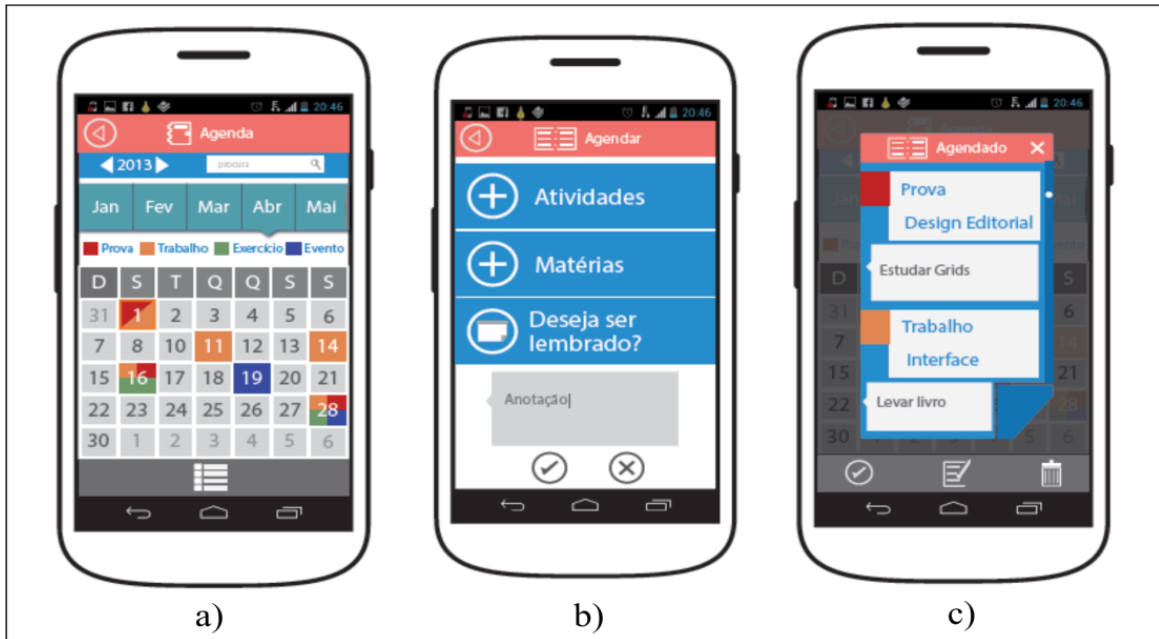


Figura 1. Principais telas da aplicação: **a)** Tela de Calendário; **b)** Tela de Cadastro da Atividade; e **c)** Tela de Atividades Cadastradas.

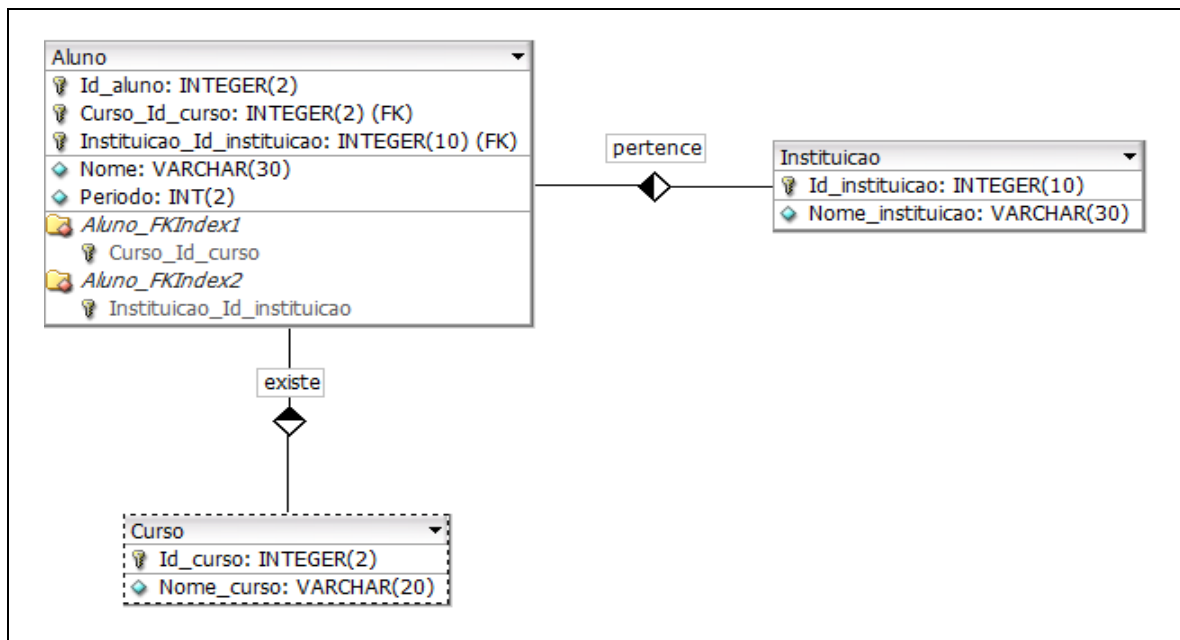


Figura 2. Diagrama de Entidade e Relacionamento - Aluno

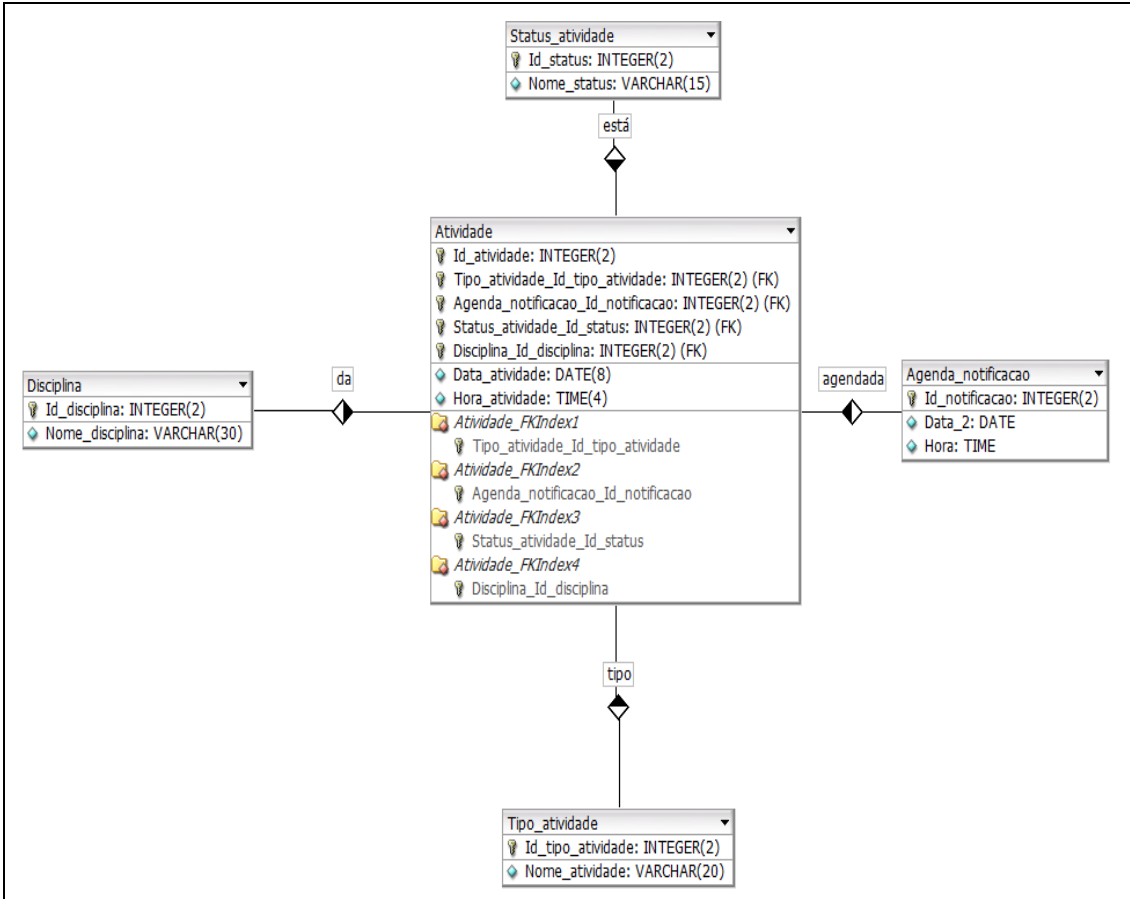


Figura 3. Diagrama de Entidade e Relacionamento - Atividade

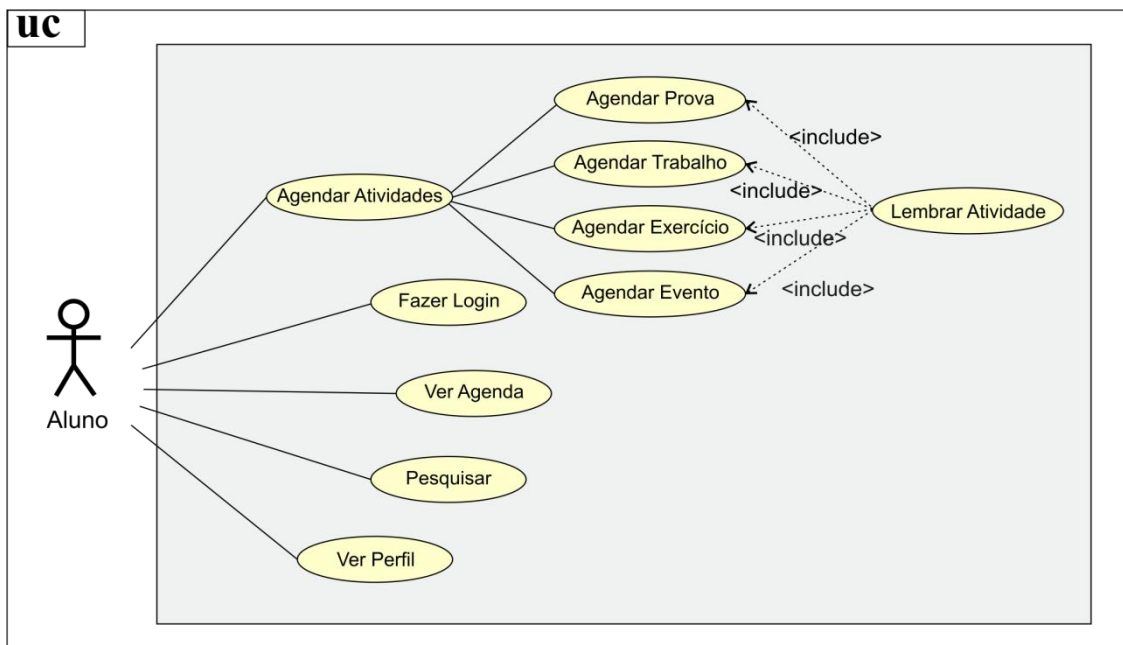


Figura 4. Diagrama de Casos de Uso do sistema da Agenda Acadêmica

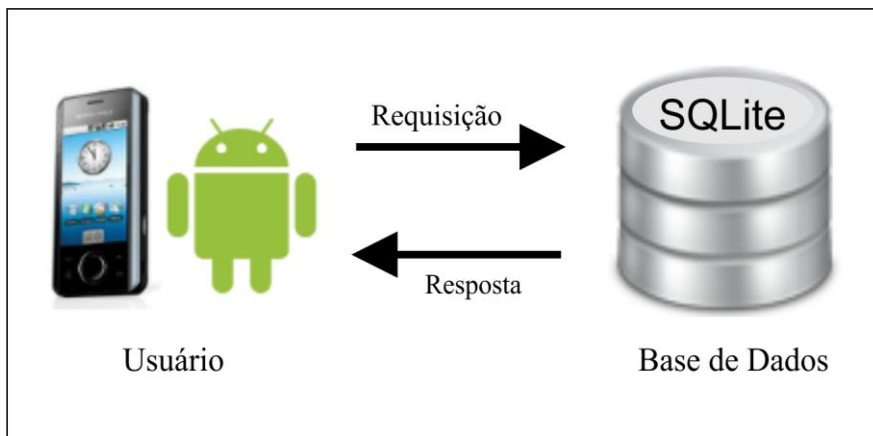


Figura 5. Arquitetura do sistema Agenda Acadêmica



Figura 6. Tela de Cadastro de Atividade



Figura 7. Exemplo de notificação do sistema ao usuário

Cronograma de Atividades

Nº	Descrição	Ago	Set	Out	Nov	Dez	Jan	Fev	Mar	Abr	Mai	Jun	Jul
		2014					2015						
1	Análise de abordagens para desenvolvimento de aplicativos móveis	X	X										
2	Elaboração de diagramas para a aplicação		X	X									
3	Análise de padrões de projeto em dispositivos móveis				X	X							
4	Implementação do protótipo da aplicação						X	X	X				
5	Integração da agenda acadêmica em dispositivos móveis									X			
6	Realização de testes práticos									X	X		
7	Análise dos resultados											X	
8	- Elaboração do Resumo e Relatório Final (atividade obrigatória)												X