UNIVERSIDADE ESTADUAL DA PARAÍBA - UEPB
CAMPUS VII – GOVERNADOR ANTÔNIO MARIZ
CENTRO DE CIÊNCIAS EXATAS E SOCIAIS APLICADAS
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

ANDERSON BATISTA DE ARAÚJO SANTANA

COMP-CONNECT: DESENVOLVIMENTO DE UM APLICATIVO ANDROID PARA AUXÍLIO DOS ALUNOS DO CURSO DE CIÊNCIA DA COMPUTAÇÃO DA UEPB - CAMPUS PATOS

PATOS - PB
2019

ANDERSON BATISTA DE ARAÚJO SANTANA

COMP-CONNECT: DESENVOLVIMENTO DE UM APLICATIVO ANDROID PARA AUXÍLIO DOS ALUNOS DO CURSO DE CIÊNCIA DA COMPUTAÇÃO DA UEPB - CAMPUS PATOS

Trabalho de Conclusão de Curso em Ciência da Computação da Universidade Estadual da Paraíba, como requisito parcial à obtenção do título de Bacharel em Ciência da Computação.

Área de concentração: Desenvolvimento mobile.

Orientadora: Dra. Jannayna Domingues Barros Filgueira

PATOS - PB

2019

S232c Santana, Anderson Batista de Araujo.
Comp-connect: [manuscrito] : desenvolvimento de um aplicativo android para auxílio dos alunos do curso de ciência da computação da UEPB - campus Patos / Anderson Batista de Araujo Santana. - 2020.
67 p.
Digitado.
Trabalho de Conclusão de Curso (Graduação em Computação) - Universidade Estadual da Paraíba, Centro de Ciências Exatas e Sociais Aplicadas , 2020.
"Orientação : Profa. Dra. Jannayna Domingues Barros Filgueira , Coordenação do Curso de Computação - CCEA."
1. Linguagem de programação. 2. Android. 3. JavaScript. I. Título
21. ed. CDD 006.76

Elaborada por José E. da S. Eugênio - CRB - 15/591 BSC7/UEPB

Anderson Batista de Araújo Santana

COMP-CONNECT: DESENVOLVIMENTO DE UM APLICATIVO ANDROID PARA AUXÍLIO DOS ALUNOS DO CURSO DE CIÊNCIA DA COMPUTAÇÃO DA UEPB - CAMPUS PATOS

Trabalho de Conclusão de Curso apresentado ao Curso de Bacharelado em Ciência da Computação da Universidade Estadual da Paraíba, em cumprimento à exigência para obtenção do grau de Bacharel em Ciência da Computação.

Aprovado em 07/12/2020

BANCA EXAMINADORA

Jannayna Domingues Barros Filgueira

Prof. Dra. Jannayna Domingues Barros Filgueira
(Orientadora)

Rômulo Rodrigues de Morais Bezerra

Prof. Me. Rômulo Rodrigues de Morais Bezerra
(Examinador)

Prof. Me. Pablo Ribeiro Suárez
(Examinador)

RESUMO

Esse projeto tem como premissa a criação de um sistema que auxilie os alunos da Universidade Estadual da Paraíba – Campus VII. Nesse aplicativo o aluno poderá criar seus próprios cronogramas de atividades, visualizar informações referentes ao curso e universidade. O aplicativo foi implementado utilizando a linguagem de programação JavaScript, durante a codificação do sistema foi utilizada o framework React Native, que permite a criação de um sistema funcional na plataforma Android e iOS utilizando apenas uma linguagem de programação. Para o criação do aplicativo foram utilizadas técnicas de metodologias ágeis, o *easYProcess*(YP) foi utilizado para auxiliar no desenvolvimento do aplicativo e organizar as informações sobre o sistema, o aplicativo poderá ser integrado com o banco de dados do controle acadêmico referente a universidade para assim facilitar a utilização do mesmo. A ideia central desse trabalho é desenvolver a interface funcional de um aplicativo utilizando metodologias ágeis durante o processo.

Palavras-Chave: JavaScript. Mobile. YP

ABSTRACT

This project has as premise the creation of a system that helps the students of the State University of Paraíba - Campus VII. In this application the student can create their own activity schedules, view information regarding the course and university. The application was implemented using a JavaScript programming language, during the execution of the system using the React Native framework, which allows the creation of a functional system on the Android and iOS platform using only a programming language. For the creation of the application, agile methodological techniques were used, easYProcess (YP) was used to assist in the development of the application and organize the information about the system, the application can be integrated with the academic control database referring to the university for thus facilitating its use. The central idea of this work is to develop a functional interface for an application using agile methodologies during the process.

Keywords: JavaScript. Mobile. YP

LISTA DE ILUSTRAÇÕES

Figura 1 – Banco de Dados Orientado a Documento..... 13
Figura 2 – Banco de Dados Grafos..... 14
Figura 3 – Colcic..... 16
Figura 4 – UFU-Mobile..... 17
Figura 5 – easYProcess..... 19
Figura 6 – Big Chart..... 22
Figura 7 – Definição de Papéis..... 24
Figura 8 – Requisitos Funcionais..... 26
Figura 9 – Requisitos Não Funcionais..... 27
Figura 10 – Perfil de Usuário..... 28
Figura 11 – Objetivos de Usabilidade..... 29
Figura 12 – Modelo de tarefas – Calculadora..... 30
Figura 13 – Modelo de tarefas – Cronograma de Estudos..... 31
Figura 14 – Modelo de tarefa – Páginas de Informações..... 31
Figura 15 – Modelo de tarefas – Grade Curricular..... 32
Figura 16 – Modelo de tarefas – Horário Acadêmico..... 32
Figura 17 – Modelo de tarefas – Login..... 33
Figura 18 – Modelo de tarefas – Área do Professor..... 33
Figura 19 – Modelo de tarefas – Tela Principal..... 34
Figura 20 – Modelo de tarefas – Tela de Busca..... 35
Figura 21 – User Stories e Testes de Aceitação..... 36
Figura 22 – Protótipo de Interface – Tela Aluno e Página Inicial..... 37
Figura 23 – Protótipo de Interface – Barra Lateral..... 38
Figura 24 – Protótipo de Interface – Horário Acadêmico..... 39
Figura 25 – Protótipo de Interface – Informações Gerais e Área do Professor..... 40

LISTA DE TABELAS

Tabela 1 – Modelo Relacional 12
Tabela 2 – Banco de Dados Orientado a Coluna..... 14
Tabela 3 – Estrutura de Chave/Valor..... 15
Tabela 4 – Plano de Release..... 42
Tabela 5 – Big Chart..... 44
Tabela 6 – Análise de Riscos..... 46

SUMÁRIO

1. INTRODUÇÃO 2
2. OBJETIVOS 2
2.1 Objetivos Gerais 2
2.2 Objetivos Específicos 2
2.3 Justificativa 3
3. REVISÃO BIBLIOGRÁFICA 3
3.1 Desenvolvimento Mobile 3
3.1.1 *Plataformas Nativas* 4
3.1.2 *Plataformas Híbridas* 5
3.2 Linguagens e Tecnologias 6
3.2.1 *React Native* 6
3.2.2 *Redux* 7
3.2.3 *Node.js* 7
3.3 Android 8
3.4 Banco de Dados 11
3.4.1 *Banco de Dados Relacionais* 12
3.4.2 *Banco de Dados não Relacionais* 13
3.5 Trabalhos Relacionados 15
3.5.1 *Colcic* 15
3.5.2 *UFU-Mobile* 16
4. METODOLOGIA 17
4.1 Métodos Ágeis 18
4.1.1 *easYProcess* 18
5. RESULTADOS E DISCUSSOES 22
5.1 Definição de Papéis 22
5.2 Conversa com o Cliente 24
5.2.1 *Descrição do Sistema* 24
5.2.2 *Requisitos Funcionais e não Funcionais* 26
5.2.3 *Perfil de Usuário* 27
5.2.4 *Objetivos de Usabilidade* 29
5.3 Inicialização 30
5.3.1 *Modelagem de Tarefas* 30
5.3.2 *User stories e Testes de Qceitação* 35
5.3.3 *Protótipo de Interface* 37
5.3.4 *Projeto Arquitetural* 40
5.4 Planejamento 41
5.4.1 *Plano de Release* 41
5.4.2 P*lano de Iteração* 44

5.5 Big Chart 44
5.6 Análise de Riscos 45
6. CONCLUSÃO 47
7. PROPOSTA PARA TRABALHOS FUTUROS 47
REFERÊNCIAS 48
APÊNDICE A – Plano de Iterações 50
APÊNDICE B – Interface do Aplicativo 507

1. INTRODUÇÃO

A evolução da tecnologia vem aumentando exponencialmente, os dispositivos móveis passaram por várias mudanças ao longo dos anos. Inicialmente esses dispositivos eram restritos apenas a efetuar funções simples como realizar chamadas e enviar mensagens, ao longo dos anos novos sistemas operacionais e novas tecnologias de hardware foram desenvolvidas, permitindo um grande avanço tecnológico. Dispositivos móveis passaram a ser um objeto bastante presente no dia a dia das pessoas, vários aplicativos foram desenvolvidos com o intuito de melhorar e facilitar a vida do usuário (MORO, TEREZINHA, 2014). Existem aplicativos disponíveis para ajudar estudantes a encontrar informações sobre seus respectivos cursos, ajudando a manter um controle sobre sua vida acadêmica, esses aplicativos fornecem informações sobre professores, restaurante universitários e até mesmo horários referentes ao transporte público. Como por exemplo o aplicativo “Portal aluno UFRJ”, ele é utilizado oficialmente para ajudar os alunos a emitir documentos com maior facilidade, o aplicativo também permite a visualização do boletim, histórico escolar e localização das salas de aula.

Neste aspecto, este trabalho tem o objetivo de desenvolver um aplicativo mobile que auxilia estudantes a encontrar informações sobre professores, grade curricular, horários de aulas e eventos destinado ao curso de Ciência da Computação, com o objetivo de auxiliar o aluno a organizar sua rotina acadêmica.

2. OBJETIVOS

2.1 Objetivos gerais

Este trabalho tem como objetivo desenvolver um aplicativo, proporcionando aos alunos do curso de Ciências da Computação da UEPB - Campus VII uma forma simples e rápida de encontrar informações sobre o curso de modo que ajude a organizar sua rotina acadêmica.

2.2 Objetivos específicos

A criação desse projeto tem como principal função criar a interface de usuário do sistema, os dados armazenados pelo aplicativo devem ser inseridos a partir de um banco de dados fornecido pela UEPB. Para facilitar a utilização do aplicativo é

necessário integrá-lo ao sistema acadêmico da universidade, assim, facilitando o acesso a informações fornecidas pelo aplicativo. Os objetivos específicos desta pesquisa são:

- Realizar revisão bibliográfica sobre aplicativos mobile;
- Realizar pesquisa sobre aplicativos utilizados nas universidades;
- Fazer levantamento dos requisitos necessários para implementar o aplicativo;
- Desenvolver a interface de um aplicativo mobile para auxiliar os alunos do curso de Computação da UEPB.

2.3 Justificativa

O curso de Ciência da Computação do Campus VII da Universidade Estadual da Paraíba hoje conta com cerca de 250 alunos matriculados. Para obter informações sobre o curso os alunos recorrem ao site ou a grupos de redes sociais. Deste modo, o aplicativo Comp-Connect tem a proposta de agrupar todas essas informações em uma única plataforma, proporcionando aos alunos e professores do curso um acesso rápido e fácil às atividades referentes ao curso.

O aplicativo fornece dados sobre o curso de maneira mais rápida e fácil, auxiliando o aluno durante toda sua jornada acadêmica a criar seus próprios horários de estudo e facilitando a busca sobre informações referentes à Universidade.

3. REVISÃO BIBLIOGRÁFICA

Este tópico descreve as características do desenvolvimento mobile, apresentando aplicativos semelhantes ao que foi desenvolvido.

3.1 Desenvolvimento Mobile

Os dispositivos móveis estão presentes cada vez mais no nosso cotidiano, segundo a Agência Nacional de Telecomunicações (Anatel) o Brasil possui 228,6 milhões de celulares em 2019 (Anatel, 2019). Várias plataformas e sistemas operacionais são utilizadas atualmente. Android e iOS são as duas plataformas mais utilizadas para o desenvolvimento de sistemas. Os dispositivos móveis são utilizados de diversas formas, como por exemplo, o Smartphone, ele fornece recursos avançados para o usuário e possibilita a realização das mesmas atividades de um

computador. O *Smartphone* é um tipo de computador portátil de fácil mobilidade e com acesso à internet de forma prática.

Por muitos anos o acesso à internet era consideravelmente maior por computadores, mas em 2014 o acesso à internet mediante o uso de celulares ultrapassou a quantidade de computadores. Atualmente os Smartphones oferecem recursos interativos com tecnologias diferentes ou aprimoradas de smartphones menos evoluídos, tais como Wifi, GPS, câmera e acesso à internet (NASSIF, 2014).

O desenvolvimento de aplicativos mobile pode ser feito de várias formas. Existem dois tipos distintos de aplicações bastante utilizadas para o desenvolvimento mobile: as nativas ou híbridas. Esses tipos de aplicações possuem suas vantagens e desvantagens. Aplicativos nativos são extremamente eficientes, porém, esse tipo de aplicativo pode se tornar bastante caro pelo fato de serem exclusivos de um único sistema operacional. Aplicativos híbridos não são desenvolvidos utilizando uma linguagem nativa do sistema, eles são desenvolvidos utilizando linguagens bastantes utilizadas pela web, são elas: CSS, HTML e JavaScript, permitindo que esses aplicativos possam ser executados em diversos sistemas operacionais (SOUZA et. al, 2017). Sistemas web e mobile utilizam de ferramentas fornecidas pelo modelo cliente-servidor, permitindo a armazenação do sistema de forma eficiente.

O modelo cliente-servidor é uma aplicação de sistemas distribuídos, o cliente é responsável por enviar requisições de dados para o servidor e aguardar a resposta referente ao pedido, o servidor é *host* responsável por executar vários serviços que deve fornecer recursos ao cliente, o modelo cliente-servidor pode ser considerado em um aplicativo como o back-end e front-end, o front-end é responsável por cuidar da parte interativa da aplicação e enviar requisições ao back-end que por sua vez será o servidor (MENDES, 2002).

3.1.1 *Plataformas Nativas*

Aplicações nativas necessitam de bibliotecas, linguagens e *frameworks* que são dependentes da plataforma desejada. No desenvolvimento de aplicativos para dispositivos mobile existem diversas plataformas, sendo elas Android, iOS, Windows Phone e Blackberry. Cada plataforma possui suas próprias restrições de linguagens e tecnologias que podem ser utilizadas. O kit de desenvolvimento de software (SDK) é único para cada plataforma de desenvolvimento (MORO, 2014).

O desenvolvimento de aplicativos nativos exige um conjunto de tecnologias e plataformas específicas para serem executadas (MORO, 2004).

Aplicativos nativos são produzidos utilizando linguagens nativas do sistema operacional escolhido. Estes podem ser feitos especificamente para um sistema operacional escolhido e deve ser utilizado uma linguagem referente a própria plataforma do sistema, no caso do Android, por exemplo, é utilizada a linguagem de programação java, para a plataforma IOS é utilizada a linguagem Objective-C, por essa razão eles são mais eficientes do que sistemas híbridos. (WHITE, 2003).

A plataforma de desenvolvimento para iOS necessita de linguagens e tecnologias específicas para sua utilização, a linguagem Objective-C é utilizada juntamente com a SDK Cocoa Touch para o desenvolvimento de seus sistemas nativos (FERNANDO, 2015).

3.1.2 *Plataformas Híbridas*

Aplicativos híbridos são a junção de sistemas híbridos e web, esses aplicativos são desenvolvidos com o intuito de serem executadas por meio da utilização de browsers a partir de dispositivos móveis. Sistemas híbridos têm seu conteúdo carregado pela internet que se comportam como aplicativos. Esse tipo de sistema é bastante utilizado por serem mais fáceis de desenvolver um sistema que funcione em diversas plataformas, assim tendo como uma das principais características o seu baixo custo de produção (FERNANDO, 2015).

Sistemas desenvolvidos em plataformas híbridas podem utilizar recursos e linguagens presentes no desenvolvimento de sistemas web, eles podem utilizar linguagens de marcação de texto baseados em HTML5 e ao mesmo tempo aproveitar recursos disponíveis no próprio smartphone (VENTEU; PINTO, 2018).

Frameworks são formas de facilitar a criação de códigos específicos ou em grande escala, são códigos-fontes referentes a classes, funções e metodologias que auxilia no desenvolvimento de sistemas (MINETTO, 2007).

O Ionic é um framework utilizado para o desenvolvimento de aplicativos híbridos, esse framework utiliza a linguagem de programação JavaScript juntamente com HTML e CSS para a criação de uma interface gráfica, com ele é possível

desenvolver aplicativos mobile de forma rápida e prática utilizando de ferramentas do próprio framework (CRISTINA; SCOMBATI, 2018).

O PhoneGap e Cordova são frameworks que permitem criar aplicativos híbridos utilizando apenas a linguagem de programação JavaScript, CSS e HTML. A PhoneGap possui várias bibliotecas que facilitam a manipulação de componentes nativos de um sistema, assim permitindo o desenvolvimento de um código que funcione em várias plataformas utilizando apenas um único código (CRISTINA; SCOMBATI, 2018). O Cordova permite a criação de aplicações para plataformas mobile com base em componentes WebView que funciona como um Browser.

3.2 Linguagens e Tecnologias

Várias linguagens, bibliotecas e frameworks podem ser utilizados para o desenvolvimento de aplicativos, a criação de um aplicativo pode ser dividida em 2 partes, são elas front-end e back-end.

O desenvolvedor front-end é responsável por gerenciar todas as tecnologias que deverão ser utilizadas na parte visual do site, o front-end trabalha diretamente com o lado do cliente. O HTML, CSS e Javascript são linguagens utilizadas pelo front-end para manipular imagens, botões, cores, margens e tipografias do site.

O back-end é utilizado do lado do servidor, ele é responsável por controlar e armazenar os dados do sistema. PHP, Python, C# e Java são exemplos de linguagens utilizadas no back-end, essa parte do sistema é responsável por controlar o banco de dados que deve conter todos os dados armazenados do aplicativo (DUARTE, 2017).

3.2.1 *React Native*

O React Native foi desenvolvido pelo Facebook, consiste em várias ferramentas que podem ser utilizadas para o desenvolvimento de aplicativos móveis, esses aplicativos podem ser utilizados na plataforma Android e iOS. O React Native permite a utilização de diversos pacotes e bibliotecas (LEBENSOLD, 2018).

De acordo com Lebensold (2018), o React Native é um framework que utiliza o JavaScript para criar aplicativos híbridos, no momento em que o sistema for compilado o React Native identifica qual plataforma ele foi compilado, caso o sistema seja compilado no Android o React Native vai traduzir o código JavaScript na linguagem principal do sistema operacional que foi compilado. O React Native utiliza pontes, que

são necessárias para realizar a comunicação entre a parte nativa do sistema e a parte desenvolvida em JavaScript.

Algumas vantagens podem ser obtidas ao utilizar esse framework, os sistemas terão requisições mais rápidas tornando a experiencia do usuário melhor, esses sistemas são mais rápidos de serem construídos pelo fato de funcionar em diversas plataformas, assim evitando de desenvolver o sistema em várias plataformas separadas. O React Native possui uma resposta mais rápida com o hardware do dispositivo, pelo fato do sistema ser nativo (MONTEIRO, 2017).

3.2.2 *Redux*

O Redux é uma biblioteca de gerenciamento de estados, ele permite a criação de logins, recarregamento automático, viagem temporal, aplicativos universais e gravações. O Redux consiste em armazenar a lógica de atualização de modelo em um determinado local, esse local é a camada de aplicação, deixando o código livre de alterações, em caso de alterações elas se tornam ações, permitindo assim avaliar o que acontece com determinada parte do código (LEBENSOLD, 2018).

O Redux pode ser dividido em 3 partes, a primeira parte é a store, essa parte é responsável pelos estados da aplicação, sendo responsável por entregar informações que foram requisitadas pelo componente. A segunda são os reducers, eles são identificados pelo nome da Store mais a palavra “user”, são encarregados de tratar todas as ações dos componentes. A terceira parte do Redux são as Actions, o seu objetivo é apenas enviar dados ao Reducer (LEBENSOLD, 2018).

3.2.3 *Node.js*

O Node.js foi criado em 2009, com o passar dos anos essa estrutura de desenvolvimento assíncrono passou a ser bastante utilizada para o desenvolvimento de aplicativos utilizando a linguagem de programação JavaScript. A Linkedln e Walmart são duas grandes empresas que utilizam o Node em seus sistemas (LHRIG, 2014).

Segundo o site oficial do Node.js, ele é um ambiente de execução da linguagem JavaScript voltada para o desenvolvimento back-end, o Node tem como foco o desenvolvimento de aplicações escalonáveis de rede.

O Node.js é utilizado como uma plataforma de desenvolvimento de aplicações JavaScript, essas aplicações são compiladas e otimizados utilizando a plataforma node e sendo interpretados pela máquina virtual V8, fazendo isso é possível utilizar o JavaScript no lado cliente e servidor (LHRIG, 2014).

O desenvolvimento de aplicativos utilizando Node.js é bastante utilizado pelo fato de ser uma tecnologia assíncrona que utiliza apenas um thread de execução, assim, o Node é capaz de atender uma grande quantidade de requisições ao mesmo tempo (LHRIG, 2014).

A linguagem de programação JavaScript foi criada em 1995. Com o passar dos anos o JavaScript se tornou a linguagem padrão para o desenvolvimento de sistemas front-end, durante muito tempo linguagens como o PHP e Java eram as mais utilizadas para desenvolver o back-end de sistemas, mas, com o surgimento do Node.js, o JavaScript passou a ser utilizado tanto no front-end quanto no back-end (LHRIG, 2014).

3.3 ANDROID

O Android é um sistema operacional Open Source, baseado em Linux, a primeira versão do Android foi lançada em 2008, após seu lançamento a maioria dos dispositivos móveis começaram a aderir a esse sistema operacional (DEITEL, 2015). O sistema operacional Android é considerado uma plataforma aberta, vários outros sistemas operacionais podem ser derivados dele, assim possuindo uma grande quantidade de sistemas baseados nesse sistema operacional (BURTON, 2014). O Android possui uma arquitetura dividida em 6 camadas, são elas: *Applications*, *Application Framework, Libraries* e *Runtime* e Linux Kernel (DEITEL, 2015).

O Android foi desenvolvido inicialmente pela empresa Android.inc no ano de 2003, A empresa foi criada em Palo Alto por Andy Rubin, Rich Miner, Nick Sears e Chris White (KELLEN et. al, 2017).

A primeira versão do Android foi desenvolvida para o dispositivo HTC Dream G1, que foi o primeiro celular a receber o sistema Android. O HTC Dream tinha teclado físico deslizável, 256mb de RAM e tela de 3,2 polegadas.

A segunda versão, de acordo com a versão 1.5 do Android, também conhecida como Cupcake, a partir da versão 1.5 todas as versões do Android possuem nome de

sobremesa em ordem alfabética. Essa versão tinha melhorias na câmera, autorrotação de tela e oferecia suporte a teclados virtuais (REVISTA INFO EXAME, 2011, p. 57).

O Android Donut trouxe várias novas funcionalidades para sua versão, segundo Kinast (2019) a versão 1.5 do Android foi lançada em 2009, várias melhorias foram feitas nesta versão do sistema operacional. A caixa de pesquisa rápida utilizando recursos de voz foram adicionados a essa versão do sistema juntamente com outras melhorias na interface e câmera. O sistema começou a oferecer suporte para telas maiores de até 400 x 800 pixels (REVISTA INFO EXAME, 2011, p. 57).

Lançado em 2010, o Android Eclair trouxe várias novidades, essa versão trouxe planos de fundo animado, capacidade de editar a tela inicial e organizar os aplicativos em pastas, navegação no Google Maps, vários novos recursos para a câmera e suporte para Bluetooth 2.1 (MELLO, SGANZERLA, 2013).

A versão 2.2 foi lançada em meados de 2010, essa versão do sistema estava 450% mais rápida que outras versões. Os recursos de voz foram melhorados, sendo capaz de executar ações como pesquisar, definir alarme e procurar rotas, apenas utilizando comando de voz, o Android Froyo também é capaz de compartilhar conexão 3G e possui suporte ao Adobe Flash (MELLO, SGANZERLA, 2013).

A versão 2.3 do Android foi a mais rápida e com maior capacidade de processamento já lançada, essa versão melhorou a utilização de aplicativos de jogos, o Gingerbread oferece mais durabilidade da bateria e suporte ao NFC, que permite realizar transferência de dados aproximando os dispositivos (MELLO, SGANZERLA, 2013).

A versão 3.0 do Android foi desenvolvida com o intuito de aprimorar a experiência de usuários de tablets, o Android Honeycomb aprimorou sua interface facilitando a sua utilização, essa versão utiliza um layout maior, facilitando a utilização de ferramentas de leitura e vídeo (LECHETA, 2019).

O Android 4.0 foi lançado no final de 2011, essa versão unificou as plataformas de desenvolvimento de smartphones e tablets, o Android Ice Cream Sandwich teve grandes atualizações e adições de novas ferramentas, permitindo o desbloqueio da tela por reconhecimento facial, navegação na internet com até 16 abas, várias

atualizações visuais em aplicativos nativos, e a adição do Android Beam, que permitia o compartilhamento de aplicativos, contatos, vídeos e músicas entre dispositivos próximos, utilizando o NFC (MELLO, SGANZERLA, 2013).

O Android 4.1, codinome Jelly Bean, foi lançado em 2012, com vários aprimoramentos no desempenho e com design moderno, nessa versão teve como ponto principal o lançamento do Google Now, esse aplicativo permite ao usuário utilizar a assistência móvel da Google, utilizando recursos que facilitam o dia a dia do usuário (MELLO, SGANZERLA, 2013).

A versão 4.4 do Android trouxe várias otimizações no sistema, várias delas são atualizações visuais, o sistema pode realizar várias ações sempre que o usuário falar “Ok Google”, a partir desse comando é possível utilizar a barra de pesquisa, reproduzir músicas e enviar mensagens (MELLO, SGANZERLA, 2013).

Seguindo para a versão 5.1, segundo Kinast (2019), o Android Lollipop trouxe várias novidades na tela de bloqueio, facilitando seu uso e mostrando notificações centralizadas, essa versão permite um melhor controle de recursos, foram aprimorados o gerenciamento de energia e outros componentes que utilizam bateria. O Android 5.0 é compatível com telas de smartphone, tablets, Android Wear e Android TV.

Já na versão 6.0, de codinome Android Marshmallow, foi lançado em 2015, essa versão trouxe várias novas funcionalidade e otimizações, foram melhoradas as funções de armazenamento de aplicativos nativos, uma nova função que permite utilizar o Google Now a partir da tela de bloqueio, nessa versão o usuário começou a ter um maior controle da utilização de dados no dispositivo, conseguindo visualizar informação de uso de memória RAM para cada aplicativo (KINAST, 2019).

Segundo Kinast (2019) o Android 7.0, de codinome Nougat, aprimorou várias funcionalidades de outras versões do sistema operacional, as barras de notificações passaram a ser fixadas em um único bloco, essa versão do Android possui suporte nativo para realidade virtual, também foi adicionado funcionalidades para facilitar a experiência do usuário, a partir dessa versão o Android puro começa a permitir a repartição de telas.

O Android 8.0 trouxe várias melhorias, dentre elas estão o baixo consumo de energia, grande aumento no desempenho sendo capaz de ligar o aparelho 2 vezes mais rápido que a versão antiga. No Android Oreo foi implementada a funcionalidade picture in picture, essa funcionalidade permite utilizar 2 tarefas ao mesmo tempo. (KINAST, 2019)

Segundo Kinast (2019) O Android 9.0, de codinome Pie, possui várias novas funcionalidades que possibilitam ao usuário realizar um melhor uso de seus aplicativos e evitar problemas relacionados a utilização frequente do smartphone, o Android Pie possui inteligência artificial que permite utilizar melhor a energia do dispositivo, concedendo mais energia para aplicativos frequentemente utilizados, essa versão permite ao usuário gerenciar melhor suas notificações, evitando assim informações em momentos indesejáveis.

3.4 Banco de Dados

Bancos de dados são essenciais no desenvolvimento de qualquer sistema computacional, são responsáveis por armazenar e tratar os dados que possuem relacionamentos com o sistema, bancos de dados representam funções do sistema que são ligados com aspectos do mundo real, uma de suas principais características são o armazenamento e a manipulação de dados do sistema (ELMASRI; NAVATHE, 2011).

Um banco de dados pode ter vários tipos de armazenamento de dados. por exemplo, lista de nomes, endereços, locais, serviços, tipos de usuários e vários outros elementos que estão relacionados com o mundo real (ELMASRI, NAVATHE, 2011).

O Sistema Gerenciador de Banco de Dados (SGBD) é uma junção de várias tecnologias que permitem a criação de um banco de dados, a função do SGBD é definir, construir manipular e realizar o compartilhamento de dados entre usuários e aplicações (ELMASRI, NAVATHE, 2011).

A melhor forma de armazenar dados de um sistema é com a utilização de banco de dados. O controle centralizado de dados ajuda a manter o sistema funcionando corretamente (ROCHA, 2011).

3.4.1 *Banco de Dados Relacionais*

O modelo Entidade-Relacionamento foi desenvolvido em 1976, ele é utilizado para descrever processos de negócio, realizando a criação e tabelas para estruturar o sistema. O diagrama é constituído por entidades que representam alguma “coisa” ou “objeto” relacionado ao mundo real, uma entidade pode se relacionar com uma ou mais entidades com o intuito de formar um diagrama capaz de ser modelado em um banco de dados, uma entidade contém um conjunto de atributos que descrevem aquela tabela, esses atributos representam os tipos de dados que podem ser armazenados nesta tabela (ROCHA, 2011).

O modelo de banco de dados relacionais, dentre todos os outros modelos de implementação, é o mais simples de ser criado, pois o modelo relacional possui uma estrutura mais formal e uniforme (KOTARO et. al, 2005).

A Tabela 1 representa a utilizando do modelo relacional, cada linha representa uma coleção de valores a primeira linha representa um estudante e todos os valores dessa linha representa informações sobre os dados relacionados ao estudante, os valores de uma mesma coluna são chamados de atributos e geralmente possuem valores de um mesmo tipo.

Tabela 1: Modelo Relacional

Estudante	Nome	Matricula	Curso
	Renato	121114352	Física
	Lucas	142425682	Administração
	Pedro	235687213	Matemática

Fonte: Própria

3.4.2 *Banco de Dados não Relacionais*

Bancos de dados relacionais utilizam linhas e colunas em sua criação, por outro lado os não relacionais não utilizam esquemas de tabelas com linhas e colunas. os bancos de dados não relacionais utilizam um sistema de armazenamento com chaves e valores, podendo ser caracterizados também como bancos de dados de grafos, banco de dados de colunas e banco de dados orientados a documentos. (SILVA, 2014)

Banco de dados orientado a documentos são bastantes flexíveis se comparado a outros bancos, são baseados em armazenamento de dados. Esse modelo não possui uma estrutura física de tabelas e colunas igual no modelo relacional, pois utiliza um armazenamento de documentos e em cada um possui um conjunto de campos e seus respectivos valores (SILVA, 2014). A Figura 1 representa uma forma de armazenar dados em um banco de dados orientado a documento.

Figura 1 – Banco de Dados Orientado a Documento

KEY	Documento

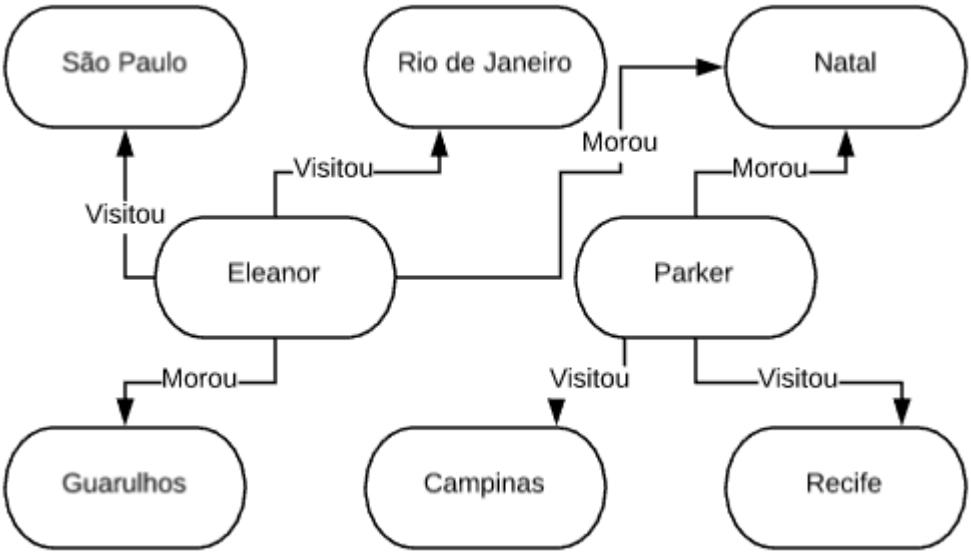
01	{
	“nome” : “Anderson”
	“idade” : “21”
	“texto” : “Local de armazenamento do texto”
	}

02	{
	“nome” : “UEPB”
	“campus” : “VII”
	“endereço” : {
	“CEP” : “58700-070”
	“estado” : “paraiba”
	“cidade” : “patos”
	“bairro” : “salgadinho”
	“rua” : “r.rotary”
	}
	}

Fonte: própria

Banco de dados de grafos possuem uma maior complexidade em relação a outras arquiteturas de armazenamento de dados. Diferente de outros bancos de dados, os modelos de bancos utilizando grafos são responsáveis por guardar objetos, a sua estrutura é composta por 3 componentes: os nós, as arestas e seus atributos (SILVA, 2014). A Figura 2 representa um modelo de banco de dados, a imagem é representada por objetos que possuem alguma ligação, eles são representados por pessoas e objetos realizando várias interações entre os elementos.

Figura 2: Banco de Dados Grafos



Fonte: própria

Banco de dados orientados a colunas permitem realizar recuperações de dados mais rápido que outros bancos, esse tipo de banco possui uma boa performance relacionada ao desempenho de requisições de entrada e saída de dados (SILVA, 2014).

A Tabela 2 demonstra um exemplo de banco de dados orientados a coluna, nesse banco temos linha e colunas que representam seus identificadores, nomes e ano de lançamento de algumas linguagens de programação.

Tabela 2: Banco de Dados Orientado a Coluna

Linguagens_id	Linguagens_nome	Ano_id
103	JavaScript	1995
104	Python	1991
105	C#	2000

Fonte: própria

Banco de dados orientado a Chave/Valor possuem uma baixa complexidade de desenvolvimento, eles armazenam dados de algum objeto e realiza uma indexação

de chaves que permite acessar esses dados, esse modelo permite que os dados sejam acessados rapidamente por meio de chaves (SILVA,2014). A Tabela 3 representa a estrutura de um banco de dados orientado a Chave/Valor, nessa estrutura cada objeto possui uma chave indexada, os objetos da Tabela 3 podem ser acessados pela sua respectiva chave.

Tabela 3: Estrutura de Chave/Valor

Chave	Valor
Chave 1	Valor da chave 1
Chave 2	Valor da chave 2
Chave 3	Valor da chave 3

Fonte: própria

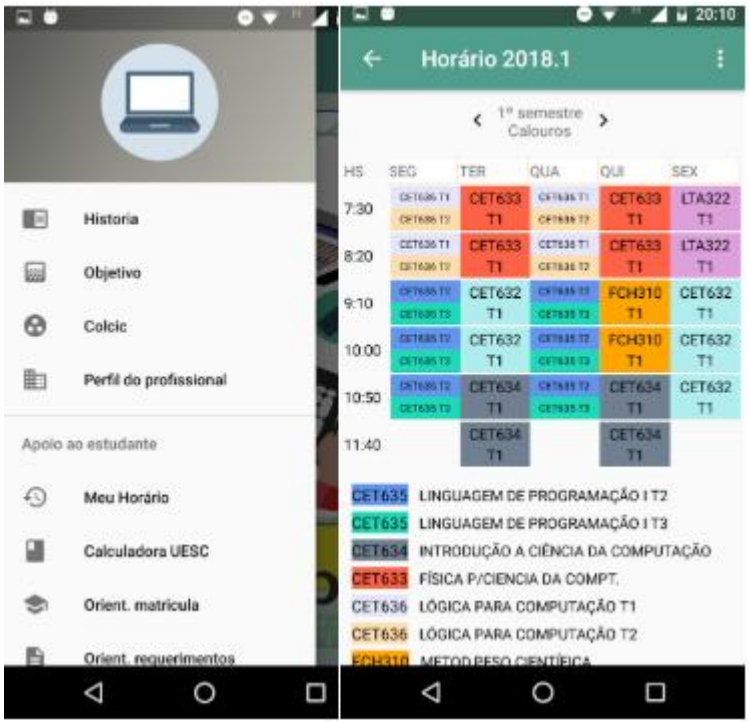
3.5 Trabalhos Relacionados

Uma pesquisa foi realizada utilizando a loja de aplicativos oficiais da Google com o intuito de encontrar aplicativos com tecnologias e funcionalidades semelhantes que estejam disponíveis para outras Universidades. Foram encontrados alguns aplicativos de outras Universidades que utilizam tecnologias ou funcionalidades semelhantes.

3.5.1 *Colcic*

O aplicativo Colcic foi desenvolvido por alunos do curso de Ciência da Computação da UESC, esse aplicativo fornece aos alunos várias funcionalidades relacionadas ao curso, os alunos podem visualizar informações sobre o curso, professores, ementas de cada disciplina, o aplicativo permite que o aluno cadastre o seu horário. A Figura 3 mostra o aplicativo Colcic, nele podemos ver história da universidade, perfis de profissionais, horário do semestre e outros detalhes disponíveis oferecidos pelo aplicativo.

Figura 3: Colcic



Fonte: Play Store (editado)

3.5.2 *UFU-Mobile*

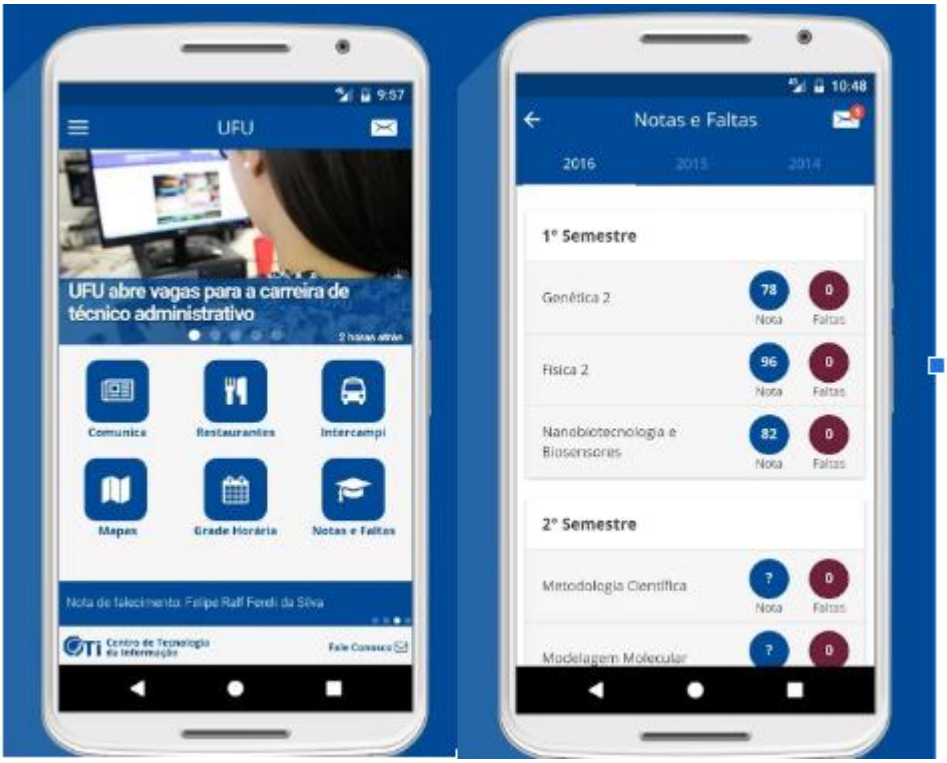
O UFU-Mobile é o aplicativo oficial da UFU Universidade Federal da Uberlândia, o aplicativo possui 3 módulos, o primeiro é o módulo para a comunidade, nesse módulo o usuário tem acesso a informações sobre o Restaurante, Transporte, Comunicação e mapas da Universidade.

O segundo módulo é o módulo para estudantes, que oferece aos alunos informações sobre o seu histórico, grade curricular, e mensagens.

O terceiro módulo é voltado para os docentes, nesse módulo os docentes terão acesso a lista de presença, agenda e mensagens.

Na Figura 4 podemos ver algumas funcionalidades do aplicativo UFU-Mobile, o intuito do aplicativo é facilitar a vida acadêmica de docentes e discentes.

Figura 4: UFU-Mobile



Fonte: Play Store (editado)

4. METODOLOGIA

O processo de desenvolvimento desse trabalho foi baseado na premissa de uma necessidade relacionada aos estudantes da UEPB, após realizar pesquisas envolvendo o cenário foi realizado a prototipação de um aplicativo, assim buscando soluções para o determinado problema.

A partir de dados levantados durante o projeto foi feita a criação de uma interface funcional que deverá fornecer informações e ferramentas para facilitar a vida acadêmica dos estudantes da UEPB, foi escolhido realizar o desenvolvimento de um aplicativo mobile que possibilita aos estudantes buscar informações relacionadas ao curso e planejar sua própria rotina acadêmica. O sistema possui características estáticas, dinâmicas e encapsuladas.

Estáticas: Textos e imagens informativas explicando a história do curso de Ciência da Computação da UEPB, perfil dos profissionais da área, objetivo do curso entre outras informações.

Dinâmicas: Informações geradas com a interação dos usuários, como o horário particular de cada aluno, a lista de matérias, professores e ementas das disciplinas disponibilizadas semestralmente pelo colegiado, entre outras informações.

Encapsuladas: Links de informações disponibilizadas em outros sistemas que são acopladas dentro do Comp-Connect, como Portal Prograd, Controle Acadêmico, Regimento Geral da UEPB, entre outros.

O YP é a metodologia ágil utilizada para organizar e realizar coleta de dados para o desenvolvimento do aplicativo.

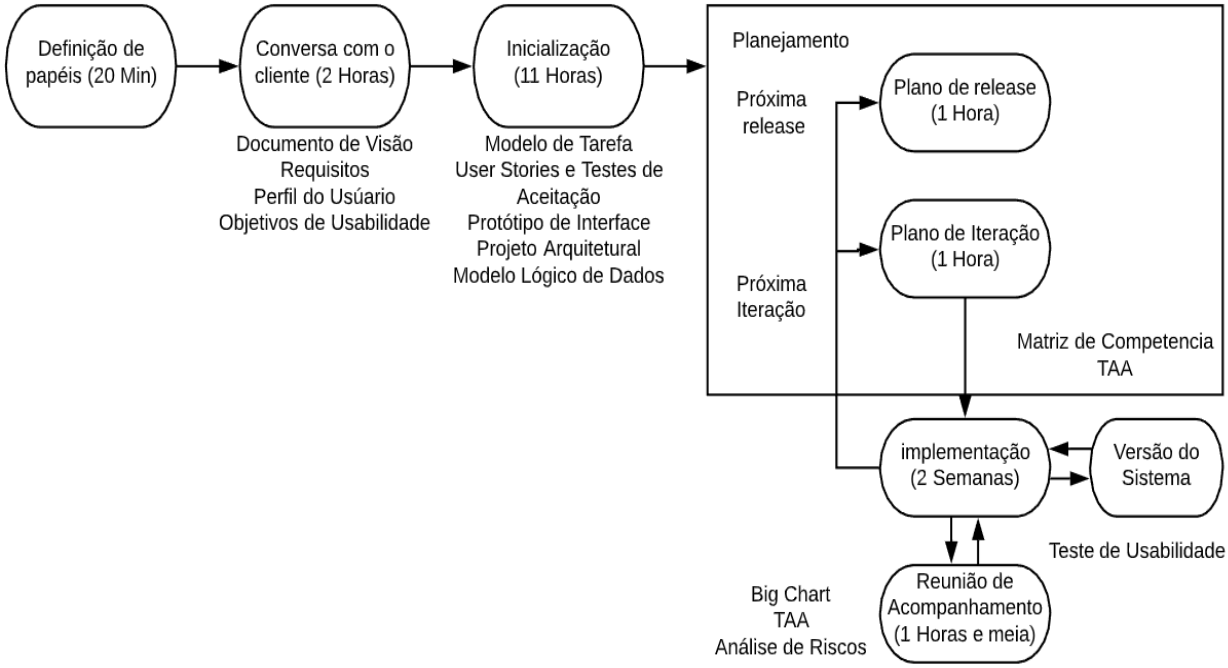
4.1 Métodos ágeis

Métodos ágeis são utilizados para melhorar a qualidade de projetos. Esses métodos são flexíveis e adaptáveis, pois realizam processo incremental e iterativo que possibilitam a equipe a desenvolver um produto que atenda melhor aos requisitos do cliente.

4.1.1 *easYProcess*

O easYProject, também conhecido como YP, é um método ágil desenvolvido pela Universidade Federal de Campina Grande. O YP possui um fluxo de processos capaz de gerar documentações e entrega do produto para auxiliar a equipe do projeto, na Figura 5 podemos observar todas as etapas referentes a modelo YP.

Figura 5: easYProcess



Fonte: Documentação do YP (Editado)

A primeira etapa do YP é a definição de papéis, nessa etapa a equipe deve gerar um documento com os membros da equipe e suas atribuições. O YP recomenda a criação de 5 papéis, são eles: cliente, usuário, gerente, desenvolvedor e testador.

O segundo processo do YP é a realização da conversa inicial com o cliente, essa reunião deve levar até 2 horas, a partir dessa reunião devem ser gerados 4 artefatos, são eles: Documento de visão, requisitos, perfil do usuário e objetivos de usabilidade.

O documento de visão deve ser criado logo após a reunião com o cliente, este artefato deve conter as ideias gerais do sistema, assim, definindo a área do problema de uma forma que seja de fácil entendimento.

Documento de requisitos deve ser produzido para definir características, funcionalidades e limitações do sistema, esse documento é dividido em documento de requisitos funcionais e não funcionais, os requisitos funcionais são aqueles que caracterizam funções que o software deve possuir, por outro lado os requisitos não funcionais são restrições e propriedades que o sistema deve possuir.

O artefato de perfil do usuário deve conter características de possíveis usuários do sistema, dados como idade, habilidades, objetivos e motivações podem ser características de cada perfil de usuário de acordo com sua relevância no projeto.

Objetivos de usabilidade é um artefato que deve conter dados para avaliar a usabilidade dos sistemas, esses objetivos podem ser metas que o sistema deve atingir para melhorar sua eficiência.

O terceiro processo do YP é a inicialização, esse processo dura 11 horas e deve gerar 5 artefatos, sendo eles o modelo de tarefas, User Stories e testes de aceitação, protótipo de interface, projeto arquitetural e modelo lógico de dados.

O documento de modelo de tarefa é uma representação de raízes em ordem hierárquica, cada raiz e sub raiz possuem tarefas que visam facilitar o entendimento de cada funcionalidade do sistema.

User Stories e testes de aceitação são um documento de especificação que o desenvolvedor deve criar as funcionalidades do sistema, sua função é facilitar e organizar o desenvolvimento do sistema, auxiliando o desenvolvedor a analisar o modelo de tarefa.

O protótipo de interface deve conter partes do sistema, seja ele funcional ou não, esse protótipo deve facilitar o entendimento do sistema a partir de representações gráficas, desta forma temos um maior controle sobre as especificações do sistema.

O projeto arquitetural tem como função principal descrever todo um funcionamento do sistema, tornando fácil o entendimento do funcionamento de cada parte do sistema e como cada parte dele funciona, esse artefato deve mostrar a estrutura do sistema dividido por partes, e o como cada parte se relaciona com outras.

O modelo lógico de dados é um artefato que descreve o funcionamento do banco de dados do sistema, mostrando toda sua estrutura lógica e descrevendo o relacionamento de cada objeto presente no banco de dados.

O quarto processo do YP é o planejamento e execução de cada release. Cada release deve conter iterações, cada um deles deve estar associada a User Stories e testes de aceitação. O plano de iteração consiste em dividir User Stories em

subtarefas menores, cada tarefa deve conter o tempo necessário para sua realização, esse tempo deve ser fornecido pelos desenvolvedores responsáveis por cada User Stories. O plano de release deve conter 2 iterações, cada iteração deve conter um conjunto de User Stories e seus respectivos testes de aceitação, cada atividade deve ser feita por membros da equipe pré-definidos.

O quinto processo atribuído pelo YP é implementação de cada iteração que deve ser realizado por cada membro da equipe da qual a tarefa foi atribuída, esse processo tem como objetivo conter uma versão do sistema funcional, o tempo necessário de implementação de todo o sistema deve ser representado por 3 releases, cada release possui 2 iterações, cada iteração deve ser feita em até duas semanas, assim, o tempo necessário para o desenvolvimento do sistema não deve ultrapassar 3 meses.

A reunião de acompanhamento é o último processo fornecido pelo YP, esse processo deve gerar 3 artefatos: Big Chart, Tabela de Alocação de Atividades (TAA) e análise de riscos. O Big Chart deve conter o acompanhamento da coleta de métricas de User Stories contendo observações sobre cada uma sempre que necessário, caso uma User Stories não possa ser realizada, deve se constar no Big Chart para uma melhor avaliação do projeto. A Figura 6 contém uma demonstração fictícia de um modelo de Big Char, nesse modelo podemos visualizar todos os dados necessários para a o desenvolvimento da Big Char referente ao sistema.

Figura 6: Big Chart

Data	Classes	Scripts Bd	Testes de Aceitação	Testes de Unidade	User Stories	Observações
19/12	0	0	0	0	0	Dia do inicio.
04/01	0	0	1	0	1	Periodo de análise
2/1	0	0	3	0	2	Periodo disponibilizado para estudo
10/1	2	0	4	0	3	Houve um problema relacionado a adaptação das tecnologias por parte da equipe
25/2	4	0	5	0	3	Não foi feita a US4.
01/11	8	0	6	0	4	Houve um problema tecnico
7/02/20	15	6	17	0	9	
16/02	21	9	23	0	11	
02/03	22	9	23	0	11	Houve um problema tecnico

Fonte: própria

A Tabela de alocação de atividades é onde utilizamos os dados do Big Chart e da Tabela de Alocação de Atividades, utilizamos esses dados para analisar informações referentes ao andamento do projeto, os resultados dessa tabela devem ser analisados pelo gerente de projeto, ele deve analisar as falhas e pensar em soluções para resolver os possíveis problemas analisados esses dados pelo TAA.

5. RESULTADOS E DISCUSSÕES

Nessa sessão será discutida todos os passos que foram feitos para o desenvolvimento do sistema, utilizaremos de artefatos referentes ao YP para demonstrar toda a criação do projeto.

5.1 Definição de papéis

De acordo com a documentação oficial do YP, para iniciarmos o processo de desenvolvimento é necessário montar uma equipe para que cada membro assuma um papel durante o projeto. No YP temos 5 papeis: Cliente, usuário, gerente, desenvolvedor e testador. Podemos analisar que vários papeis podem ser atribuídos a um mesmo individuo caso seja viável, a alocação dos papeis são dinâmicas podendo ser alteradas e substituídas dependendo da viabilidade do projeto.

O cliente é responsável por descrever como o sistema deve funcionar, e quais as características que o sistema deve ter, assim, fornecendo informações de suma importância para o desenvolvimento do projeto. É responsabilidade do cliente especificar quais partes do sistema devem ser consideradas como urgência, para que sejam implementadas assim que possível. O cliente deve fornecer informações no decorrer de todo o projeto e participar das seguintes etapas: Plano de release, testes de aceitação, perfil de usabilidade, objetivos de usabilidade, protótipo de interface, projeto arquitetural e dedicar o máximo de tempo possível para interagir com os membros da equipe para auxiliar na criação de um sistema que atenda a todos os seus requisitos.

É dever do gerente coordenar todas os processos que envolvem o projeto, ele deve ser capaz de tomar decisões de riscos e analisar cada membro da equipe com o intuito de verificar os prazos e eficácia de todas as atividades. O gerente é responsável por: Elaborar artefatos relacionados a release e iteração, administrar documentos de análise de riscos, manter todos os artefatos em versões atuais e acessíveis, alocar papeis, administrar reuniões de acompanhamento e resolver quais problemas que envolva o projeto e seus membros.

O desenvolvedor é responsável por administrar os requisitos referentes ao sistema, assim como criar o código referente ao projeto e analisar todos as características do sistema e como podem ser manipuladas. É responsabilidade do desenvolvedor: criar os artefatos de requisitos funcionais e não funcionais juntamente com o cliente para conseguir entregar um produto o mais próximo possível do desejado, analisar e criar o modelo de tarefas, criar protótipos de interface, identificar objetivos de usabilidade, produzir testes de unidade, construir o modelo logico de dados e criar projeto arquitetural.

O testador é responsável por analisar o código dos desenvolvedores e realizar seus devidos testes de aceitação, é de suma importância que o testador realize as seguintes tarefas: Desenvolver testes de aceitação, criar testes para o código de outros desenvolvedores, analisar código fonte da equipe, realizar testes de usabilidade. A Figura 7 representa o primeiro artefato referente ao YP, nele podemos verificar toda a equipe envolvida no projeto e seus respectivos papeis.

Figura 7: Artefato de Definição de Papéis

EQUIPE	PAPÉIS
Jannayna	Cliente
Alunos e Coordenação	Usuários
Anderson	Desenvolvedor, Gerente, Testador

Fonte: Própria

5.2 Conversa com o cliente

O segundo processo do YP é responsável por criar 4 artefatos, nessa etapa reunimos informações sobre o sistema a partir de uma reunião feita com o cliente, assim, adquirimos um maior entendimento sobre as necessidades dos usuários e clientes, toda a equipe deve aproveitar esse momento para iniciar o processo de desenvolvimento de vários artefatos, como por exemplo os requisitos funcionais e não funcionais, perfil de usuário, objetivos de usabilidade e vários outros.

Para analisar se o primeiro contato com o cliente foi eficiente é necessário que a equipe tenha uma ideia base de como será elaborado os artefatos de: documento de visão, perfil de usuário, requisitos funcionais e não funcionais.

5.2.1 *Descrição do sistema*

Na criação do primeiro artefato referente a etapa de conversa com o cliente, é realizado uma descrição das funcionalidades do sistema.

O aplicativo Comp-Connect, foi desenvolvido com o intuito de ajudar os alunos do curso de ciência da computação. Os alunos de computação possuem muitas vezes dificuldade em buscar informações referentes ao curso, percebemos então que há uma grande necessidade de um sistema que possa auxiliar os alunos durante o período acadêmico.

O Comp-Connect será utilizado pelos alunos da UEPB Campus VII, o aplicativo contará com um sistema de login onde será utilizado dados de login da própria instituição, informando o seu número de matrícula e uma senha, a matricula do usuário será validada a partir do banco de dados da UEPB que contém o número de matricula de todos os estudantes de computação da UEPB Campus VII.

O aplicativo possui duas categorias, a primeira é direcionada para o aluno com informações privadas, e a segunda com informações sobre toda a UEPB, o aplicativo possui apenas um tipo de usuário, esse usuário é o aluno que deve possuir acesso a todas as funcionalidades do aplicativo, o aluno poderá utilizar as seguintes ferramentas:

- Apresentação do Curso: Contém informações básicas disponibilizadas pela UEPB sobre o curso de Ciência da Computação.
- Perfil do Egresso: Contém informações sobre o perfil de um profissional da área de Computação.
- Atividades: Essa parte do sistema é responsável por informar aos alunos todos os projetos de extensão, monitorias, Estágio, TCC e outra atividades referentes ao curso.
- Calculadora: Os alunos poderão calcular notas das duas unidades, informando sobre a média, e possíveis notas requeridas na prova final.
- Cronograma de Estudos: O aluno poderá criar seu horário de estudos particular, podendo cadastrar atividades e utilizar de alarmes para um maior rendimento escolar.
- História: Nessa etapa deverá conter a história da UEPB campus VII.
- Calendário Acadêmico: Contém um link que permite ao usuário visualizar o calendário acadêmico do período letivo
- Horário: Permite ao usuário criar o seu horário de aulas, adicionando disciplinas nos horários da manhã e noite.
- Professores: Contém dados de todos os professores de computação, informando sobre o e-mail currículo lates e outas informações.
- Regimento: Aqui o aluno poderá visualizar o documento do regimento da UEPB.

O Comp-Connect terá os dados armazenados em uma base de dados, que será integrada ao sistema de controle acadêmico da UEPB, para que seja possível ter acesso ao login de cada aluno.

5.2.2 *Requisitos funcionais e não funcionais*

O documento de requisitos é feito antes da etapa de desenvolvimento, a partir dele podemos analisar como o sistema deve se comportar, o que ele deve ou não fazer e quais elementos devem ser buscados para chegar no objetivo do sistema.

A partir do artefato de requisitos funcionais é possível observar o que o sistema deve fazer e como deve se comportar, na Figura 8 podemos analisar o artefato de requisitos funcionais.

Figura 8: Requisitos Funcionais

TIPO	DESCRIÇÃO
Componente 1 Cadastro	Realizar cadastro referente a operações realizadas pelos usuários e administradores do sistema.
Componente 2 Visualizar Arquivos	O sistema deve mostrar ao usuário documentos criados pela UEPB, esses documentos são: Perfil de Egresso, Apresentação do curso, Atividades, História, Calendário Acadêmico, Horário das aulas, Regimento.
Componente 3 Informações Pessoais	O Sistema deve oferecer ao aluno ferramentas que possa ser utilizado para auxiliar na sua vida acadêmica, o aluno deve ter acesso ao cronograma particular de estudos, calculadora de notas e horário acadêmico personalizado.
Componente 4 Alterar Informações	O sistema deve possuir uma área editavel onde será possivel alterar e adicionar informações.
Componente 5 Banco de Dados	O sistema deve realizar a coleta de dados referentes ao login a partir do banco de dados do controle acadêmico dos alunos.
Componente 6 Área do Aluno	A área do aluno deve conter os seguintes componentes: Apresentação do curso, perfil de egresso, atividades, calculadora, cronograma de estudos e horário do período
Componente 7 Área do Campus	A área do campus é destinada a informações fornecidas pela UEPB, os componentes que devem fazer parte dessa área são: Calendário acadêmico, história, horário geral, professores, regimento, mapa do campus

Fonte: Própria

Os requisitos não funcionais representam funcionalidades e aspectos importantes do sistema, tais requisitos são importantes para manter a qualidade do

software e analisar quais fatores devem ser levados em conta para a construção do sistema. Na Figura 9 podemos visualizar o artefato de requisitos não funcionais.

Figura 9: Requisitos Não Funcionais

REQUISITO	DESCRIÇÃO
Portabilidade	Smartphone com sistema Android ou IOS
Confiabilidade	O sistema deve oferecer ao usuario segurança e confiabilidade.
Privacidade	Os dados privados do aluno devem ser visiveis apenas ao proprio usuário, os dados de login devem ser utilizados a partir do banco de dados da instiruição e usados com cautela.
Desempenho	O sistema deve ser capaz de realizar um comando em até 5 Segundos.
Banco de Dados	O banco de dados utilizado seria utilizado a partir de um banco de dados da UEPB e caso seja necessário, deverá ser criado um banco de dados separado.
Implementação	O sistema devera utilizar a linguagem JavaScript, React Native, bibliotecas JS e outras tecnologias que possam ser necessárias durante o desenvolvimento do sistema.
Segurança	Apenas alunos poderam utilizar o sistema, mediante a uma validação do seu login acadêmico.
Ferramenta Auxiliar	Google Drive.
Interface	Imagens devem ser feitas no Photoshop e deverá utilizar a logo da UEPB.

Fonte: Própria

5.2.3 *Perfil de usuário*

O artefato de perfis trata-se de um conjunto de informações relacionadas a usuários do sistema, esse documento tem o intuito de analisar dados do usuário para definir quais de suas características são importantes para o projeto, nessa etapa é realizado uma conversa com o cliente para analisar os dados e criar o artefato de perfil do usuário. A Figura 10 representa o artefato de perfil de usuário, o mesmo artefato refere-se ao usuário aluno e ao administrador do sistema. O artefato é dividido em três partes, a primeira é responsável por mostrar características gerais dos alunos e do administrador do sistema, na segunda parte é responsável por mostrar conhecimentos

conceituais do usuário, na terceira parte podemos visualizar conhecimentos cognitivos do usuário.

Figura 10: Perfil de Usuário

PERFIL DE USUÁRIO	
Perfis de possiveis usuário do aplicativo descrito anteriormente	
Parte I: Cracterísticas Gerais	
Faixa etária:	16-50 Anos
Sexo:	Indiferente
Habilidades necessárias para utilizar o APP:	Conhecimento básico de informática
Frequência de execução das tarefas	Diária
Habilidades motoras	Normal
Níveis de percepção	Normal
Grau de instrução	Nenhum
Objetivos	Auxiliar na rotina acadêmica
Motivações	Praticidade e organização de conteúdos
Preferências	Uso de um Smarthphone
Frequência de execução das tarefas	Diária
Parte II: Conhecimento Conceitual	
Conhecimento Semântico	Nível de experiência
Função	Baixo
Método	Baixo
Tarefa	Baixo
Computadores	Baixo
Ferramenta utilizada na execução das tarefa	Baixo
Conhecimento Sintático	Nível de experiência
Uso do Smarthphone	Baixo
Parte III: Estilo Cognitivo	
Aprendizado	Treinamento
Capacidade de solucionar problemas	Com ajuda
Capacidade de reter aprendizado	Alta
Personalidade	
Nível de curiosidade: Alto	
Nível de persistência: Alto	
Nível de inivação: Elevado	
Inovador [x] Impulsivo [] Conservador [] Reflexivo []	

Fonte: Própria

5.2.4 *Objetivos de usabilidade*

A usabilidade de um sistema deve atingir determinados objetivos, tais como, interface simples, obter medida de satisfação, facilitar o aprendizado, ter documentos claros e objetivos, ser eficiente e eficaz.

A interface do sistema deve ser otimizada para facilitar a utilização deste aplicativo. Em relação a medida de satisfação, o aplicativo deve ter uma área voltada par ajudar o usuário a tirar suas dúvidas sobre a utilização do aplicativo. Para facilitar o aprendizado, o aplicativo deve utilizar lembretes para auxiliar o aluno.

No âmbito da documentação, o aplicativo deverá oferecer todos os documentos da universidade necessários para os alunos de modo simples e objetivos. Por fim, o aplicativo deve ser eficiente eficaz, de modo que ao utilizar o aplicativo, o aluno possa melhorar seu rendimento escolar além de planejar seu próprio horário de estudo pessoal.

O artefato referente a objetivos de usabilidade é descrito baseado em na eficácia, eficiência, segurança, *learnability* e *memorability,* a eficácia. Na Figura 11 podemos analisar o artefato descrevendo todas essas características.

Figura 11: Objetivos de Usabilidade

Objetivo	Descrição
Interface Simples	O sistema possui uma interface otimizada para facilitar a utilização do aplicatico
Medida de Satisfação	Área voltada para ajudar em duvidas sobre a utilização do aplicativo
Facilitar o Aprendizado	O aplicativo utiliza lembretes para ajudar na eficiência do aluno
Documentos claros e objetivos	O sistema oferece vários doscumento da universidade de forma simples.
Eficiência	Ao utilizar o aplicativo o aluno poderá melhorar o seu rendimento escolar e acessar informaçoes disponiveis peloa UEPB de forma simples
Eficácia	O aluno poderá planejas seu proprio hórario pessoal

Fonte: Própria

5.3 Inicialização

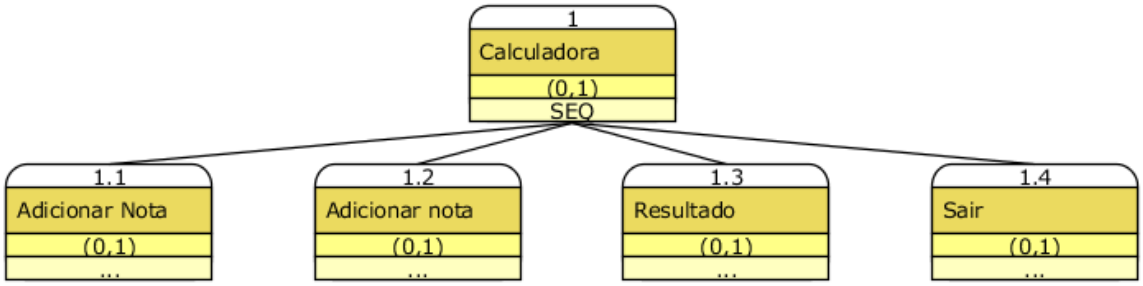
No início do desenvolvimento do YP foram realizadas as etapas de definições de papeis e conversa com o cliente que gerou outros 4 artefatos, a terceira etapa é responsável por analisar os dados até então coletados e gerar os artefatos de modelo de tarefa, User stories e testes de aceitação, protótipos e projeto arquitetural.

5.3.1 *Modelagem de tarefas*

O processo de modelagem é responsável por organizar e analisar todas as tarefas que serão utilizadas pelo usuário, a partir desse processo os desenvolvedores passam a ter maior facilidade para analisar todos os requisitos e tarefas que devem ser feitas no projeto.

A modelagem das tarefas do sistema fora feita utilizando o formalismo *Task and Action Oriented System* (TAOS). Na Figura 12 podemos analisar a modelagem referente a calculadora de notas disponibilizada pelo sistema.

Figura 12: Modelo de Tarefas - Calculadora

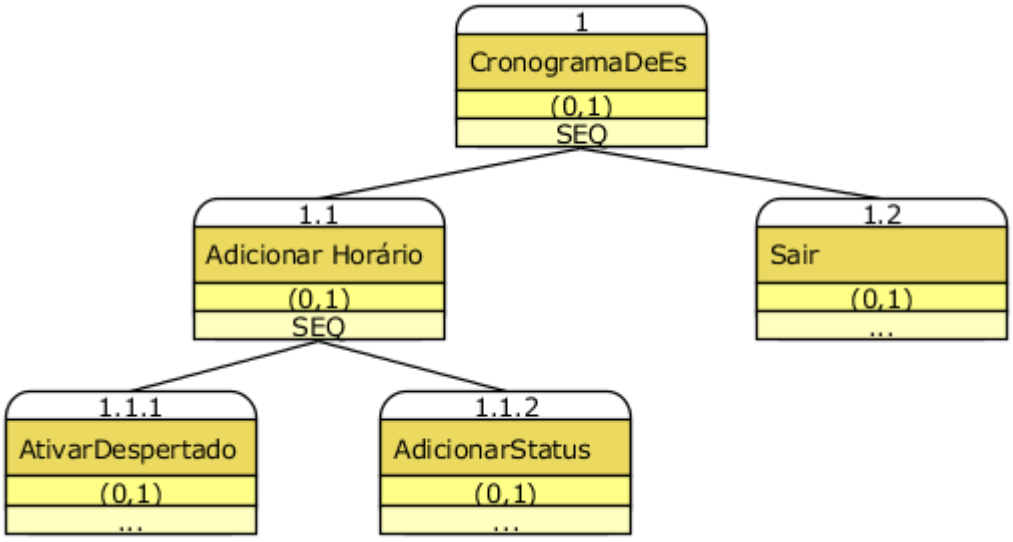


Fonte: Própria

O aplicativo possui uma área reservada para o aluno administrar suas atividades durante o dia, o usuário pode criar atividades com despertadores em horários específicos e marcar cada tarefa como concluída ou não.

A Figura 13 representa a modelo de tarefas referente a criação de atividades do aluno.

Figura 13: Modelo de tarefas – Cronograma de estudos

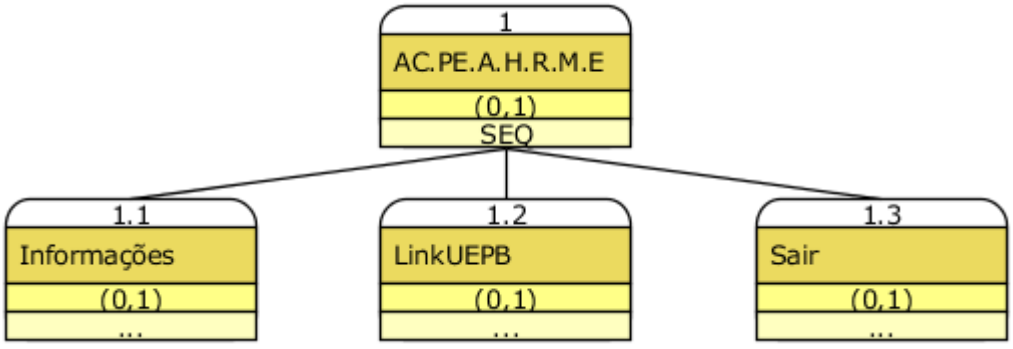


Fonte: Própria

O sistema possui algumas informações que possuem uma mesma estrutura de tela. O perfil de egresso, apresentação do curso, atividades, história, regimento e mapa possuem a mesma estrutura e o mesmo modelo de tarefas.

A Figura 14 representa o modelo de tarefas das informações mostradas anteriormente, essas informações são fornecidas pela UEPB com links para o site oficial da instituição e descrições no próprio aplicativo.

Figura 14: Modelo de tarefa – páginas de informações

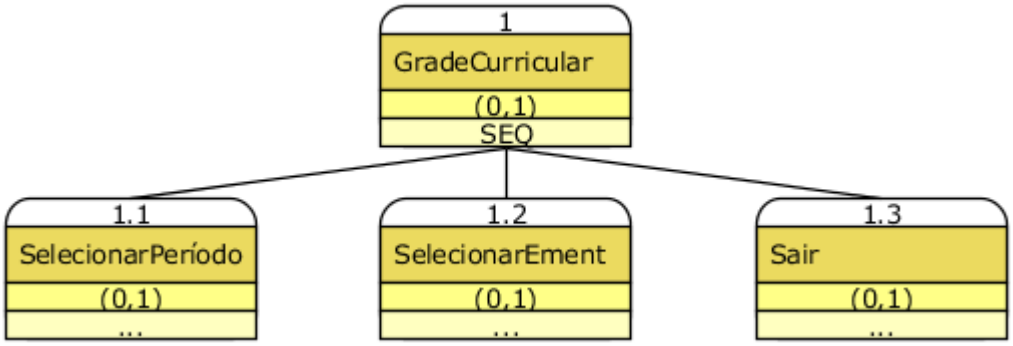


Fonte: própria

O aluno possui uma área reservada para informações sobre cada período do curso, o usuário pode selecionar o período e após isso visualizar todas as disciplinas

do período e suas respectivas ementas. A Figura 15 representa o modelo de tarefas referente a grade curricular.

Figura 15: Modelo de tarefas – grade curricular

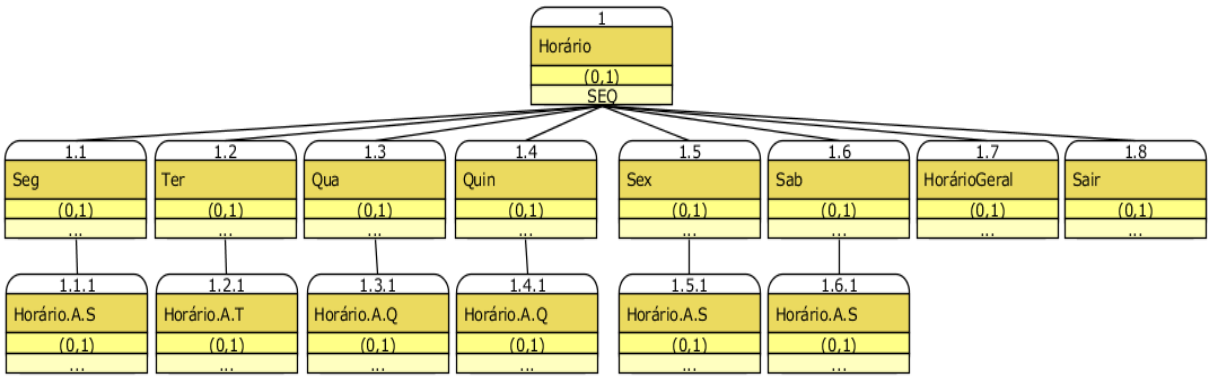


Fonte: própria

O sistema possui uma área que disponibiliza a criação de seu horário acadêmico, referente apenas a criação do horário das aulas.

Na Figura 16 podemos analisar o modelo de tarefas referente a criação do horário acadêmico.

Figura 16: Modelo de tarefas – horário acadêmico



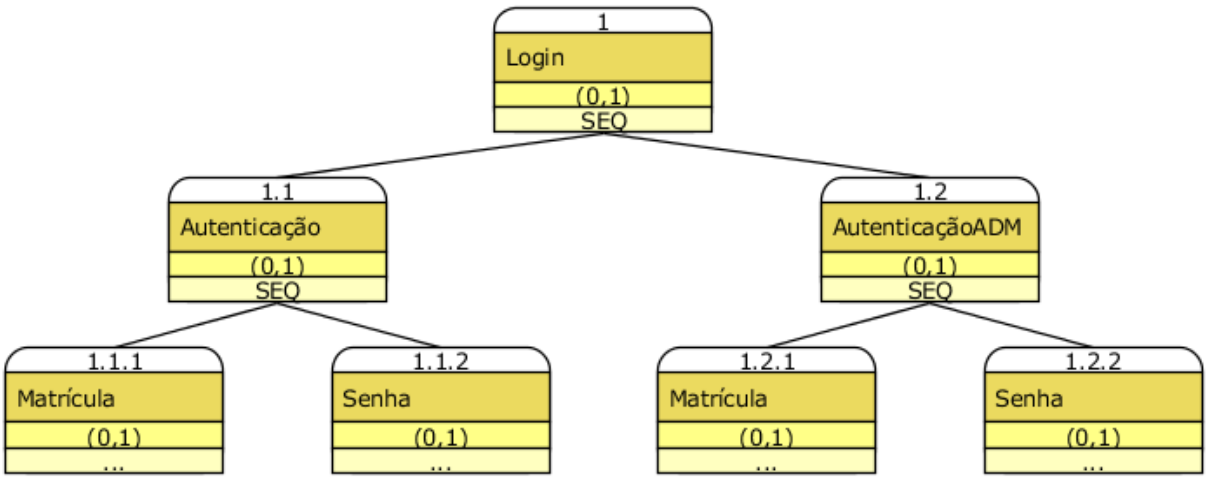
Fonte: própria

O projeto possui planos de mudanças para trabalhos futuros, visando a adaptação do aplicativo com o controle acadêmico da UEPB, foi criada o modelo de tarefas referente ao login, para realizar login no aplicativo é necessário validar os dados do controle acadêmico, caso o aluno esteja matriculado na instituição ele

deverá utilizar os mesmos dados do controle acadêmico, e assim o login será efetuado com sucesso.

Na Figura 17 podemos analisar o modelo de tarefas que demonstra como deve ser feito esse login.

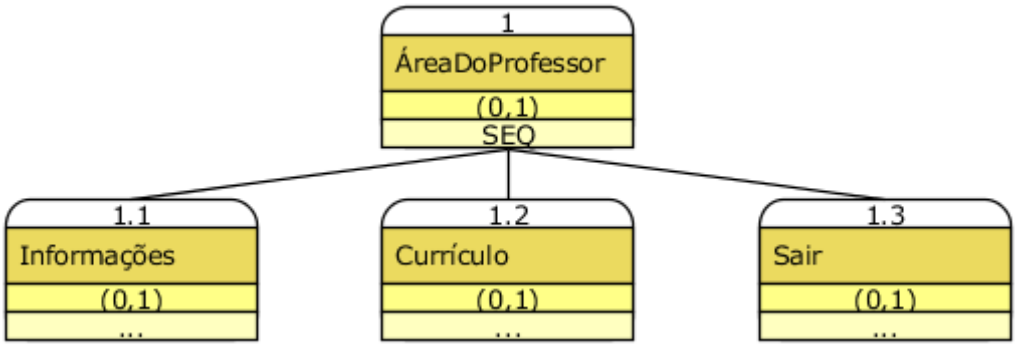
Figura 17: Modelo de tarefas - login



Fonte: própria

O sistema possui uma área que contém informações sobre todos os professores, na Figura 18 podemos analisar a modelagem de tarefa dessa parte do sistema.

Figura 18: Modelo de tarefas – área do professor

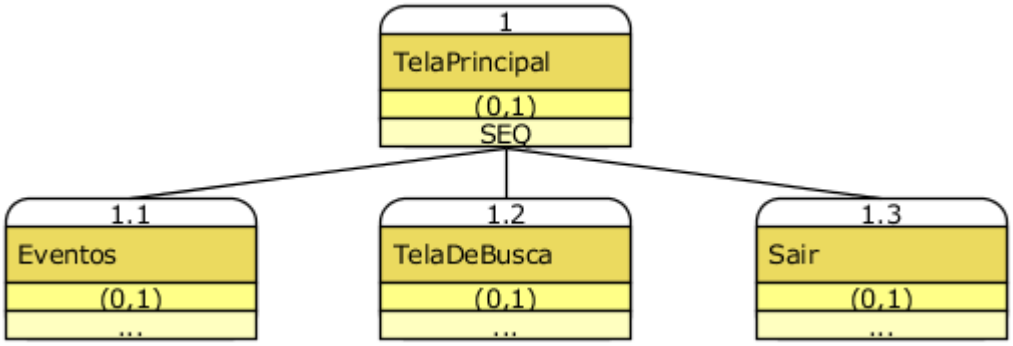


Fonte: própria

A tela principal do aplicativo possui informações sobre eventos, e uma aba destinada a opções referentes a área do aluno e da universidade.

Na Figura 19 podemos analisar a tela principal do aplicativo e suas funcionalidades.

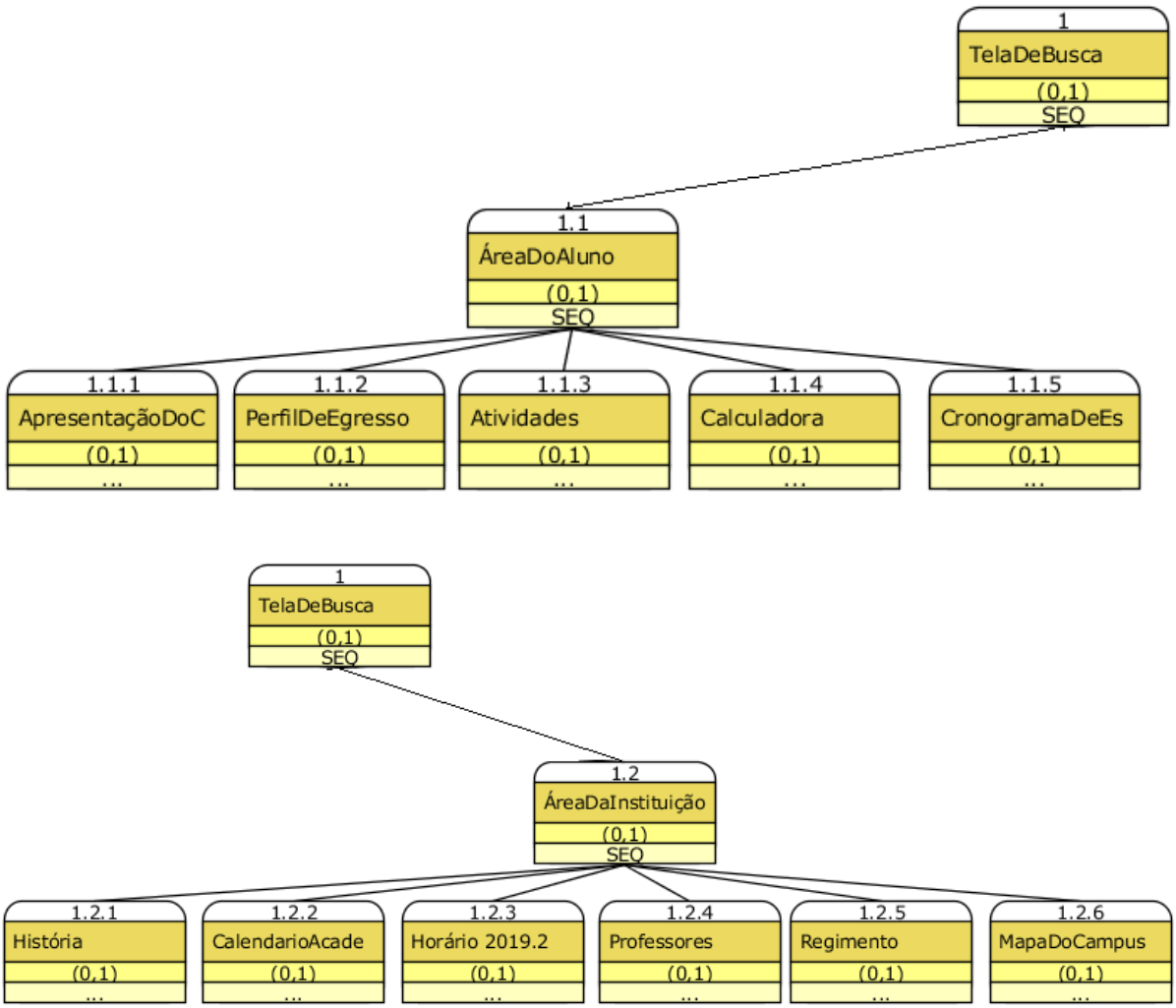
Figura 19: Modelo de tarefas – tela principal



Fonte: própria

A tela de busca contem informações sobre o curso, universidade e campus assim como uma area destinada exclusivamente ao aluno, na Figura 20 temos o modelo de tarefas dividido em duas partes, a primeira refere-se a area destida ao aluno e a segunda contem a estrutura de tarefas da area destinada a informações sobre o campus.

Figura 20: Modelo de tarefas – tela de busca



Fonte: própria

5.3.2 *User stories e testes de aceitação*

O artefato de user stories é responsável por planejar as funções que o sistema deve fazer, assim como definir partes do processo. Esse documento trata-se do planejamento inicial das user stories e dos seus respectivos testes de aceitação, esse artefato pode ser alterado no decorrer do projeto caso seja requisitado pelo cliente ou pela equipe de desenvolvimento.

A Figura 21 demonstra o planejamento de cada user storie com seus respectivos testes de aceitação.

Figura 21: User Stories e Testes de Aceitação

	User Stories e Testes de Aceitação	
US01	Analisar artefatos.	Estimativa: 10h
US02	Estudar tecnologias.	Estimativa: 50h
TA2.1	Verificar se as tecnologias satisfazem as necessidades do sistema.	
US03	Criar protótipos e validar os requisitos do sistema.	Estimativa: 10h
TA3.1	Verificar se os protótipos estão de acordo com as necessitades do projeto.	
TA3.2	Analisar protótipos gerados e artefatos relacionados.	
TA4.3	Avaliar protótipo com o cliente.	
US04	Elaborar banco de dados.	Estimativa: 17h
TA4.1	Armazenar dados de login do alunos e administradores.	
US05	Implementar tela inicial do sistema.	Estimativa: 12h
TA5.1	Verificar dados de evento e criar slidshow.	
US06	Implementar área barra lateral, botão de saída e estrutura de informações do aluno e instituição.	Estimativa: 7h
TA6.1	Analisar eficiência da barra lateral e posicionamento de informações.	
US07	Criar horário acadêmico do aluno.	Estimativa: 14h
TA7.1	Adicionar horário de segunda a sábado.	
TA7.2	Visualizar horário geral do período.	
TA7.3	Adicionar disciplina referente ao horário escolhido.	
US08	Implementar área de conteúdos gerais.	Estimativa: 6h
TA8.1	Apresentação do curso.	
TA8.2	Perfil de egresso.	
TA8.3	Atividades gerais.	
TA8.4	História do curso.	
TA8.5	Regimento.	
TA8.6	Ementa.	
US09	Implementar área do professor.	Estimativa: 9h
TA9.1	Visualizar informações sobre os professores e verificar o currículo selecionado.	
US10	Criar área de login.	Estimativa: 5h
TA10.1	Adicionar número de matricula e senha do aluno e administradores do site.	
US11	Validar dados e analisar documentação	Estimativa: 10h
TA11.1	Alterar artefatos	
US12	Criar área de cronograma pessoal do aluno.	Estimativa: 17h
TA.1	Adicionar atividade.	
TA.2	Verificar funcionalidade do alarme.	
TA.3	Adicionar checklist.	

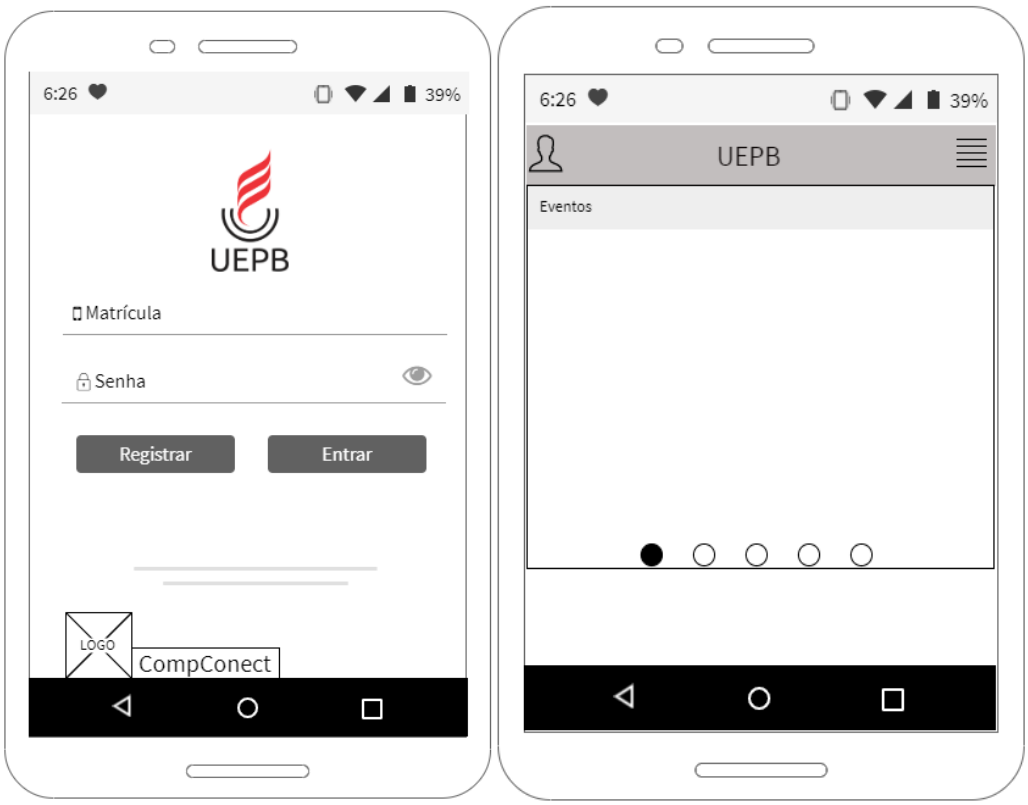
Fonte: própria

5.3.3 *Protótipo de interface*

Protótipos são representações de funções do sistema através de imagens, o YP recomenda a construção de telas não funcionais e simples, esse artefato proporciona ao cliente e usuários descrições visuais que facilitam o entendimento dos requisitos do sistema. Os protótipos de interface não representam todas as características do sistema final, os requisitos podem ser alterados o que acarretaria em mudança no sistema que não foram previstas no protótipo.

Na Figura 22 podemos analisar o login do aluno e administrador do sistema, onde deve ser colocado informações referentes ao login do controle acadêmico. O segundo protótipo representa a tela inicial do sistema onde temos informações sobre eventos e a barra latera do aluno.

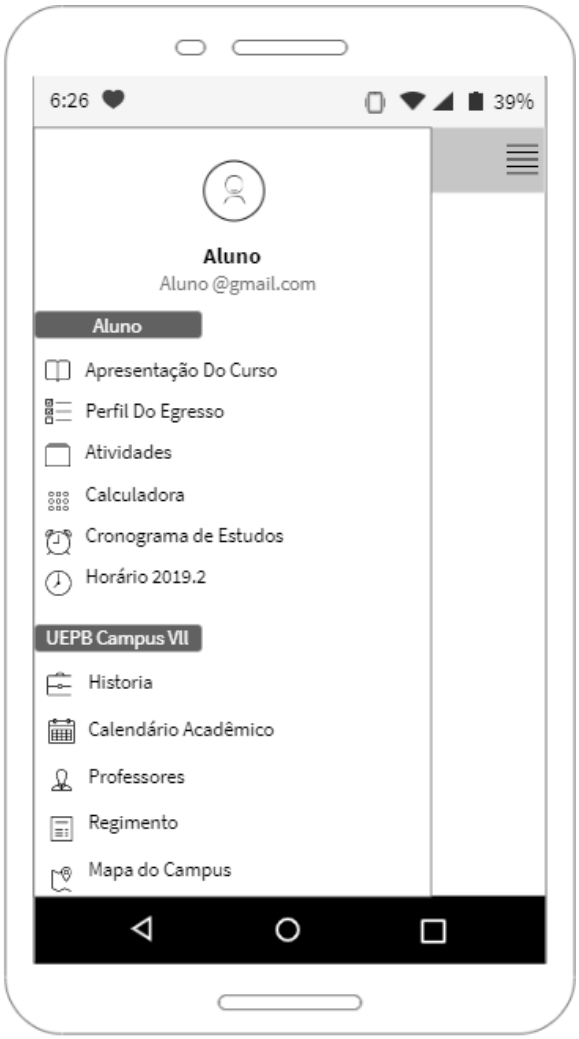
Figura 22: Protótipo de Interface – Tela Aluno e Página Inicial



Fonte: própria

A Figura 23 representa a barra lateral do usuário, podemos analisar duas áreas distintas, um referente ao aluno e outro com informações gerais sobre a universidade.

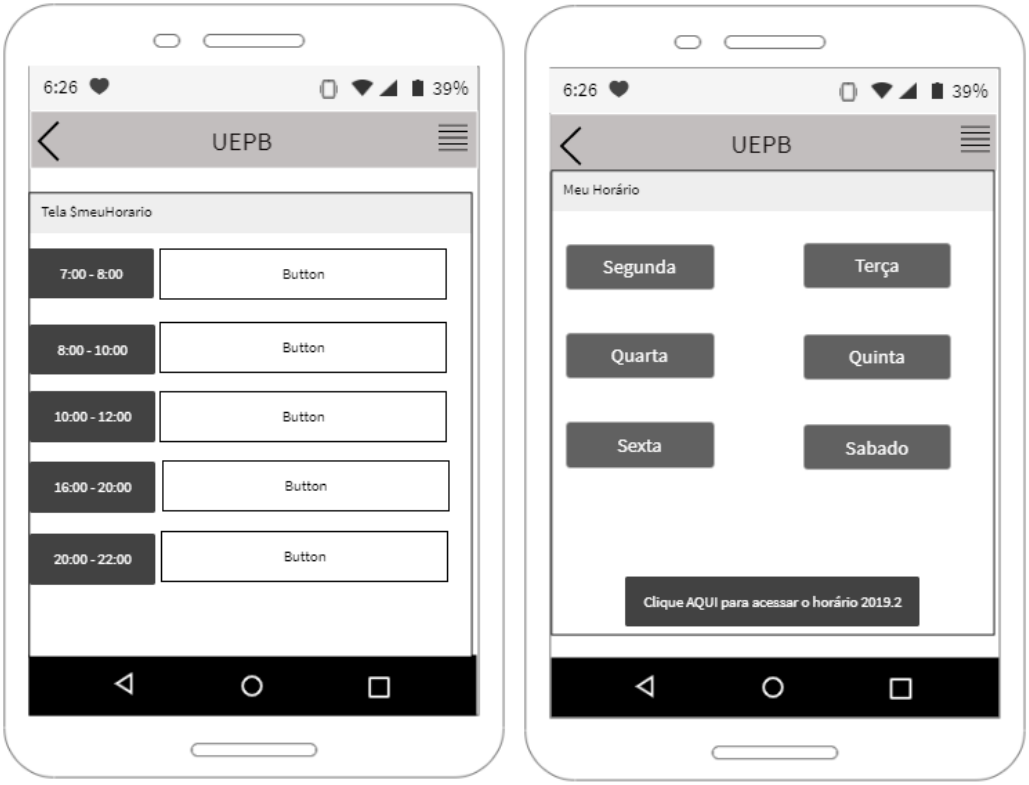
Figura 23: Protótipo de Interface – Barra Lateral



Fonte: Própria

A Figura 24 representa o protótipo relacionado a área de horário do período, nele podemos selecionar 1 dia da semana de segunda a sábado e depois, após isso a tela será substituída para uma área onde o aluno pode selecionar o horário da aula e sua respectiva disciplina.

Figura 24: Protótipo de Interface – Horário Acadêmico

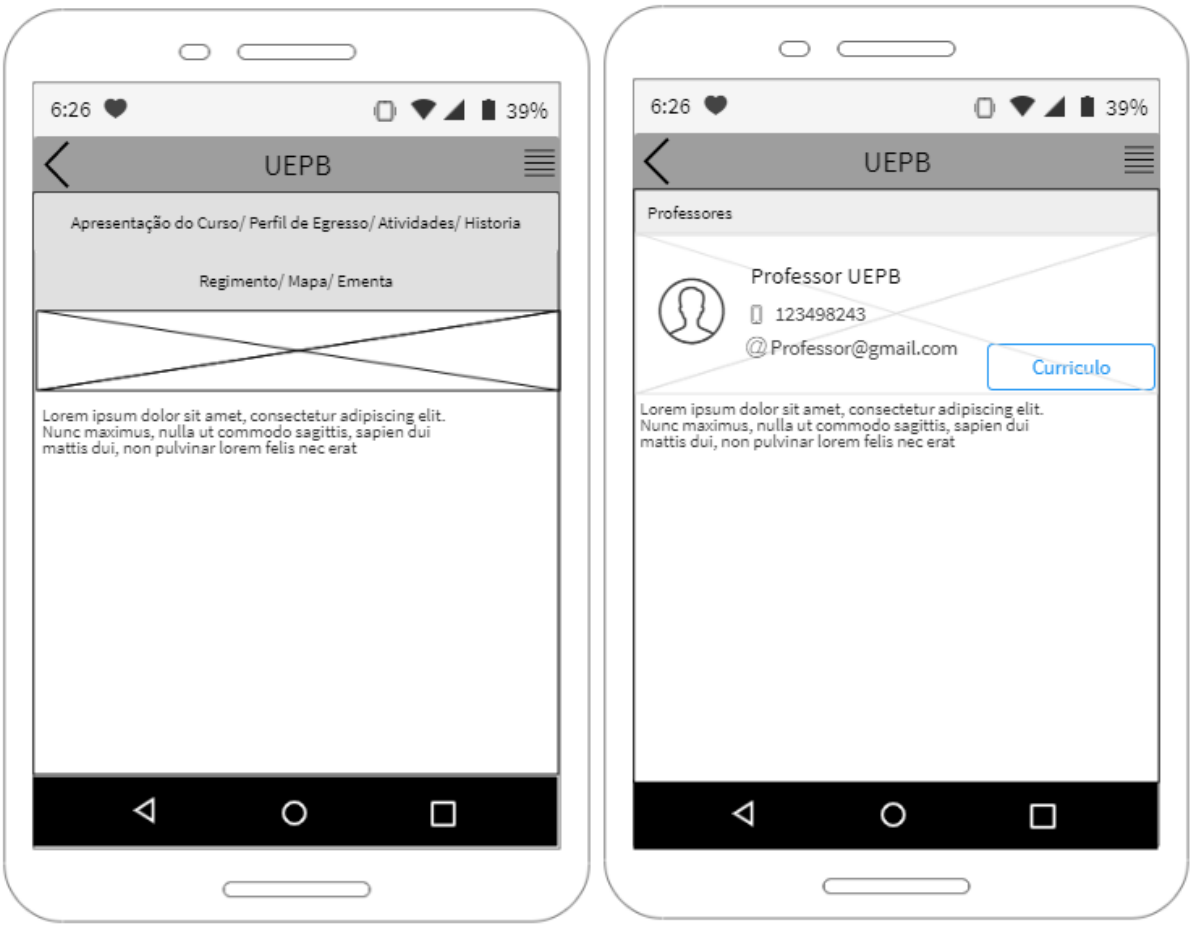


Fonte: própria

O sistema possui uma tela para informações gerais, onde deve ser possível acessar informações sobre a universidade, essas informações são relacionadas a monitorias, projetos de extensão e vários outros, na área do professor pode ver informações sobre seu telefone, e-mail, currículo e uma breve descrição da sua área de estudos e cadeiras ministradas.

Na Figura 25 podemos analisar todas as características descritas anteriormente.

Figura 25: Protótipo de Interface – Informações Gerais e Área do Professor



Fonte: própria

5.3.4 *Projeto arquitetural*

O último artefato referente a etapa de inicialização do YP, é a criação de um documento onde deve ser especificado descrições do sistema, funcionalidades e tecnologias necessárias, esse documento pode sofrer alterações dependendo da necessidade da equipe, o projeto arquitetural deve ser criado por toda a equipe de desenvolvimento do sistema e após criado deve ser validado com o cliente.

O aplicativo deve acessar arquivos externos fornecidos pela universidade estadual da Paraíba, deve ser utilizado links para visualizar informações e documentos contidos no site da universidade.

Para o desenvolvimento do projeto foram utilizadas algumas tecnologias para a criação de artefatos e código fonte, o React Native foi utilizado a partir do uso do Android Studio. O JavaScript foi utilizado como a linguagem padrão para o

desenvolvimento do sistema. Foram utilizadas várias ferramentas e frameworks durante o projeto, tais como: CorelDraw, Google Drive, ITAOS, Visual Studio Code, Android Studio, Genymotion.

5.4 Planejamento

Seguindo o cronograma de atividades descritas pelo YP, a etapa de planejamento deve ser feita junto com o cliente.

O primeiro documento que deve ser criado é a matriz de competências, que deve descrever o que cada membro da equipe deve fazer durante essa etapa. A matriz de competência não se faz nesse caso, pelo fato da quantidade de membros da equipe, podemos considerar que o único desenvolvedor deverá efetuar todas as atividades e criação de documentos, assumindo assim o papel de gerente e desenvolvedor.

O planejamento de implementação do sistema é definido a partir de releases e iterações, para criar os releases é preciso usar como base as User Stories descritas anteriormente, cada release deve ser dividido em iterações, cada iteração deve realizar a implementação da User Stories escolhida.

5.4.1 *Plano de release*

Para cria o aplicativo de forma pratica e organizada é necessário organizar todas as User Stories, para isso usaremos o plano de release, nele será adicionado 2 ou mais iterações e cada uma delas estará diretamente ligada a uma ou mais User Stories, cada iteração deve possuir o tempo de 2 semanas para concluir o desenvolvimento da User Stories e seus respectivos testes de aceitação.

Na Tabela 4 podemos analisar os releases e suas respectivas iterações.

Tabela 4: Plano de Release

Release 01: 09/07 ~ 29/07	Gerente - Anderson	
Iteração	User Story	Período
Iteração 01	US01	09/07 ~ 13/07
Iteração 02	US02	14/07 ~ 29/07
Release 02: 01/08 ~ 15/08	Gerente - Anderson	
Iteração	User Story	Período
Iteração 03	US02	01/08 ~ 09/08
Iteração 04	US03	09/08 ~ 15/08
Release 03: 16/08 ~ 01/09	Gerente - Anderson	
Iteração	User Story	Período
Iteração 05	US 04	16/08 ~ 01/09
Release 04: 02/09 ~ 14/09	Gerente - Anderson	
Iteração	User Story	Período

Iteração 06	US 05	02/09 ~ 09/09
Iteração 07	US 06	09/09 ~14/09
Release 05: 14/09 ~ 01/10	Gerente - Anderson	
Iteração	User Story	Período
Iteração 08	US 07	14/09 ~ 21/09
Iteração 09	US 08	21/09 ~ 01/10
Release 06: 03/10 ~ 20/10	Gerente - Anderson	
Iteração	User Story	Período
Iteração 10	US09 / US10	03/10 ~ 20/10
Release 07: 20/10 ~ 09/11	Gerente - Anderson	
Iteração	User Story	Período
Iteração 11	US11	20/10 ~ 25/10
Iteração 12	US12	26/10 ~ 09/11

Fonte: Própria

5.4.2 P*lano de iteração*

A iteração é formada a partir de User Stories, cada iteração possui várias User Stories que devem ser alocadas a cada membro da equipe responsável que deve ser responsável por realizar todas as tarefas descritas, após selecionar o membro responsável deve ser feito uma estimativa de tempo que o mesmo deverá gastar para finalizar sua tarefa.

A tabela de alocação de atividades é um artefato que possui todas as atividades referentes a iteração, no apêndice A podemos analisar a TAA referente a todas as iterações do projeto.

5.5 Big Chart

Para realizar análises dos acontecimentos referentes a cada release e iteração é criado o Big Chart, o gerente do projeto deve avaliar o que aconteceu em cada etapa do projeto e adicionar métricas a esse artefato.

O Big Chart é responsável por selecionar a data a partir do dia inicial de cada User Stories, a partir disso devemos preencher dados referentes a cada etapa do projeto.

Na Tabela 5 podemos analisar o Big Chart.

Tabela 5: Big Chart

Data	Scripts	Testes de Aceitação	User Stories	Observações
09/07	0	0	01	Analise de Artefatos
14/07	0	1	02	Semanas de estudo
01/08	0	1	02	Semanas de estudo
09/08	0	3	03	Prototipagem do aplicativo

16/08	7	0	04	Não foi possível realizar essa etapa
02/09	4	1	05	
09/09	7	1	06	
14/09	15	3	07	
21/09	8	6	08	
03/10	13	2	09/10	Foram realizadas duas User Stories na mesma iteração
20/10	0	1	11	Alterar documentação
26/10	9	1	12	Não foi possível concluir todos os testes de aceitação

Fonte: Própria

5.6 Análise de riscos

O artefato de análise de riscos deve avaliar o projeto e descrever situações que não estavam previstas durante o desenvolvimento, os riscos precisam ser identidades a avaliados, deve-se procurar uma solução para cada um dos riscos encontrados e analisados durante essa etapa, o projeto deve seguir um cronograma que pode ser alterado em caso de riscos considerados sem soluções ou que precisem ser realizadas alterações na documentação ou código fonte.

Para realizar a criação desse artefato é necessário descrever o problema encontrado e identifica-lo em 3 níveis de prioridade: Baixo, médio e alto.

Na Tabela 6 podemos analisar o artefato de análise de riscos, nessa tabela podemos analisar o status de cada risco encontrado, o status possui 3 níveis: Superado, vigente e abortado.

Tabela 6: Análise de riscos

Data	Risco	Prioridade	Status	Solução
01/08	Tecnologias sem conhecimento	Alta	Superado	Estudar linguagens, frameworks e tecnologias necessárias
16/08	Criar banco de dados baseado no controle acadêmico	Alta	Abortado	Não pode ser desenvolvido um banco de dados do início, mas poderá ser feito uma integração futura com o banco de dados da UEPB
09/09	Erro na criação da barra lateral	Media	Superado	Erros foram encontrados na implementação da barra lateral
14/09	Falta de conhecimento em tecnologias especificas	Media	Superado	Foram utilizadas novas ferramentas
20/10	Erros em artefatos relacionados a requisitos	Alta	Superado	Foram alterados vários artefatos
10/11	Atraso na finalização da User Storie 12	Alta	Superado	

Fonte: Própria

6. CONCLUSÃO

Este projeto teve o intuito de propor a criação de um aplicativo de auxílio para a Universidade Estadual da Paraíba Campus Vll, durante o desenvolvimento foi utilizado o React Native, esse framework possibilita o desenvolvimento de aplicativos utilizando a linguagem JavaScript, foram analisados dados referentes às necessidades dos alunos, esses dados serviram de ponto inicial para o desenvolvimento da interface funcional, a interface foi criada usando o ambiente de desenvolvimento Visual Studio Code e o Android Studio.

O aplicativo tem como principal funcionalidade auxiliar os alunos do curso de Ciência da Computação da UEPB, no aplicativo o aluno pode encontrar todas as informações relacionadas ao curso de forma rápida e eficiente, o sistema oferece ao aluno a possibilidade de criar o seu próprio cronograma de estudos, organizar notas, criar horário acadêmico, visualizar informações referentes aos professores e suas respectivas áreas. A interface do sistema pode ser visualizada no Apêndice B

Para o desenvolvimento do sistema foi necessário a utilização da metodologia ágil YP, baseado nessa ferramenta foi possível criar vários artefatos para facilitar o entendimento do projeto e o desenvolvimento do aplicativo.

7. PROPOSTA PARA TRABALHOS FUTUROS

A implementação do sistema com o banco de dados da UEPB não foi possível ser feita por motivos externos, mas o sistema possui suporte para integração com banco de dados relacionais, assim possibilitando o desenvolvimento e integração a uma base de dados própria do aplicativo.

Como possibilidades futuras para a continuação desse projeto fica o desenvolvimento de um banco de dados dedicado ao sistema, fornecendo dados relacionados ao histórico de todas as disciplinas, histórico de notas, e organização geral dos dados armazenados.

REFERÊNCIAS

BURTON, M; FELKER, D. **DESENVOLVIMENTO DE APLICATIVOS ANDROID PARA LEIGOS.** Rio de janeiro, Editora: Alta Books. 2014.

CRISTINA, K; SCOMBATTI, G. **Desenvolvimento móvel hibrido**. São Paulo: Interface Tecnológica, 2018.

DEITEL, P; DEITEL, H; DEITEL, A. **ANDROID COMO PROGRAMAR:** Porto Alegre: Bookman, 20015.

Duarte, L; **Programação Web com Node.js:** Gravataí: LuizTools, 2017.

ELMASRI, R. NAVATHE, S. **SISTEMAS DE BANCO DE DADOS. 6ed.** São Paulo: Addison Wesley, 2011.

FERNANDO, M. **APLICATIVO PARA CAPTURA DE DADOS MÓVEIS DE POLUENTES ATMOSFÉRICOS EM AMBIENTES URBANOS UTILIZANDO WEB APIS.** 2015. Trabalho de conclusão de curso (Monografia) - Curso de Pós-graduação em desenvolvimento Web, Universidade Federal do Paraná - UTFPR, Londrina, 2015.

LECHETA, R. **GOOGLE ANDROID PARA TABLETS**. São Paulo Editora: Novatec. 2012.

LEBENSOLD, J. React Native Cookbook - Bringing the Web to Native Platforms. EUA, 1ª Edição, março 2018.

MENDES, António. **Arquitetura de Software: desenvolvimento orientado para arquitetura**. Editora Campus. Rio de Janeiro - RJ, 2002.

MELLO, C; SGANZERLA, M. **APLICATIVO ANDROID PARA AUXILIAR NO DESENVOLVIMENTO DA COMUNICAÇÃO DE AUTISTAS.** Gravataí, 2013.

MORO, M; TEREZINHA, M. **OS PARADIGMAS DE DESENVOLVIMENTO DE APLICATIVOS PARA CELULARES**. 2015. 9F. Departamento de Computação - Universidade Federal de São Carlos - UFSCar, São Carlos , 2014.

MONTEIRO, F. React native Webview ou realmente nativo?, 2017. Disponível em<**https://medium.com/nutripad/react-native-webview-ou-realmente-nativo-4e30a37ae020**> Acesso em: 02/05/2019.

NASSIF, T. **APLICATIVO MOBILE PARA RESTAURANTE UNIVERSITÁRIO DA UFPR.** 2014. 68F. Trabalho de conclusão de curso (Monografia) - Curso de Tecnologia em Design Gráfico e Artes Gráficas - Universidade Tecnológica federal do paraná - UTFPR, Curitiba, 2014.

REVISTA INFO EXAME. **UMA INVASÃO ANDROID**. São Paulo: Editora Abril, nº 310, dez. 2011, 57 p.

ROCHA, E. **BANCO DE DADOS RELACIONAIS**. 2011. 63F. Trabalho de conclusão de curso (Monografia) - Curso Tecnológico em processamento de dados - Faculdade de Tecnologia de São Paulo - FATEC, São Paulo, 2011.

SILVA, S. **BANCO DE DADOS NÃO-RELACIONAIS: UM NOVO PARADIGMA PARA ARMAZENAMENTO DE DADOS EM SISTEMAS DE ENSINO COLABORATIVO**, Macapá. 2014.

SOUZA, et al. Estratégia Inteligentes para Desenvolvimento de Aplicativos Mobile Multiplataforma. In XIV SEGeT SIMPÓSIO DE EXCELÊNCIA EM GESTÃO E TECNOLOGIA, Rio de Janeiro,2017.

VENTEU, K. C.; PINTO, G. S. Desenvolvimento Mobile Híbrido. Revista Interface Tecnológica, São Paulo, v. 15, n. 1, p. 86-96, ISSN On-line 2447-0864, 2018.

KELLEN G. **ANDROID E A INFLUÊNCIA DO SISTEMA OPERACIONAL LINUX.** 2017.

KINAST, P. A história do Android: das versões 1.0 a 9.0 Pie. oficinadanet, 2019.**https://www.oficinadanet.com.br/android/24807-a-historia-do-android-das-versoes-10-ao-90-pie**.> Acesso em: 05/05/2019.

APÊNDICE A – PLANO DE ITERAÇÕES

Iteração 01 - 09/07 - 13/07					
US01 – Analisar artefatos.					
Atividade	**Descrição**	**Responsável**	**Estimativa de tempo**	**Tempo real**	**Status**
A1.1	Analisar artefatos referentes a etapa de conversa com o cliente	Anderson	3h	4h	
A1.2	Analisar artefatos referentes a etapa de inicialização	Anderson	4h	5h	
A1.3	Analisar artefatos referentes a etapa de planejamento	Anderson	3h	4h	

Iteração 02 - 14/07 - 29/07					
US02 – Estudar tecnologias.					
Testes de Aceitação					**Status**
TA2.1	Verificar se as tecnologias satisfazem as necessidades do sistema				
Atividade	**Descrição**	**Responsável**	**Estimativa de tempo**	**Tempo real**	**Status**
A2.1	Estudar ferramentas	Anderson	5h	2h	
A2.2	Estudar linguagens	Anderson	30h	32h	

A2.3	Estudar frameworks	Anderson	15h	13h	

Iteração 03 - 01/08 - 09/08					
US02 – Estudar tecnologias.					
Testes de Aceitação					**Status**
TA2.1	Verificar se as tecnologias satisfazem as necessidades do sistema				
Atividade	**Descrição**	**Responsável**	**Estimativa de tempo**	**Tempo real**	**Status**
A2.4	Estudar linguagens	Anderson	20h	17h	

Iteração 04 - 09/08 - 15/08					
US03 – Criar protótipos e validar requisitos do sistema.					
Testes de Aceitação					**Status**
TA3.1	Verificar se os protótipos estão de acordo com as necessidades do projeto.				
TA3.2	Analisar protótipos gerados e artefatos relacionados.				
TA3.3	Avaliar protótipo com o cliente.				
Atividade	**Descrição**	**Responsável**	**Estimativa de tempo**	**Tempo real**	**Status**
A3.1	Criar protótipos da pagina de login e pagina inicial	Anderson	3h	2h	
A3.2	Criar protótipos da barra lateral e conteúdos gerais	Anderson	3h	4h	

3.3	Criar protótipos da página de calculadora e horários	Anderson	4h	4h	

Iteração 05 - 16/08 - 01/09					
US04 – Elaborar banco de dados					
Testes de Aceitação					**Status**
TA4.1	Armazenar dados de login do aluno e administradores				
Atividade	**Descrição**	**Responsável**	**Estimativa de tempo**	**Tempo real**	**Status**
A4.1	Criar banco de dados para armazenas o login dos usuários	Anderson	10h	20h	

Iteração 06 - 02/09 - 09/09					
US05 – Implementar tela inicial					
Testes de Aceitação					**Status**
TA5.1	Verificar dados de eventos e criar slideshow				
Atividade	**Descrição**	**Responsável**	**Estimativa de tempo**	**Tempo real**	**Status**
A5.1	Criar tela inicial	Anderson	12h	25h	

Iteração 07 - 09/09 - 14/09					
US06 – Implementar barra lateral, botão de saída e estrutura de informações do aluno e instituição.					
Testes de Aceitação					**Status**
TA6.1	Analisar eficiência da barra lateral e posicionamento de informações.				
Atividade	**Descrição**	**Responsável**	**Estimativa de tempo**	**Tempo real**	**Status**
A6.1	Implementar funcionalidades do sistema referentes a barra lateral	Anderson	7h	15h	

Iteração 08 - 14/09 - 21/09					
US07 – Criar horário acadêmico do aluno.					
Testes de Aceitação					**Status**
TA7.1	Adicionar horário de segunda a sábado.				
TA7.2	Visualizar horário geral do período.				
TA7.3	Adicionar disciplina referente ao horário escolhido.				
Atividade	**Descrição**	**Responsável**	**Estimativa de tempo**	**Tempo real**	**Status**
A7.1	Criar horário em dias selecionados	Anderson	4h	5h	

A7.2	Visualizar horário do 5° período	Anderson	4h	5h	
A7.3	Adicionar disciplina	Anderson	5h	5h	

Iteração 09 - 21/09 - 01/10

US08 – Implementar área de conteúdo gerais.

Testes de Aceitação		Status
TA8.1	Criar área referente a apresentação do curso.	
TA8.2	Criar área referente ao perfil de egresso.	
TA8.3	Criar área referente a atividades gerais.	
TA8.4	Criar área referente a história do curso.	
TA8.5	Criar área referente ao regimento.	
TA8.6	Criar área referente a ementa.	

Atividade	Descrição	Responsável	Estimativa de tempo	Tempo real	Status
A8.1	Desenvolver área para disponibilizar acesso a informações da universidade	Anderson	6h	10h	

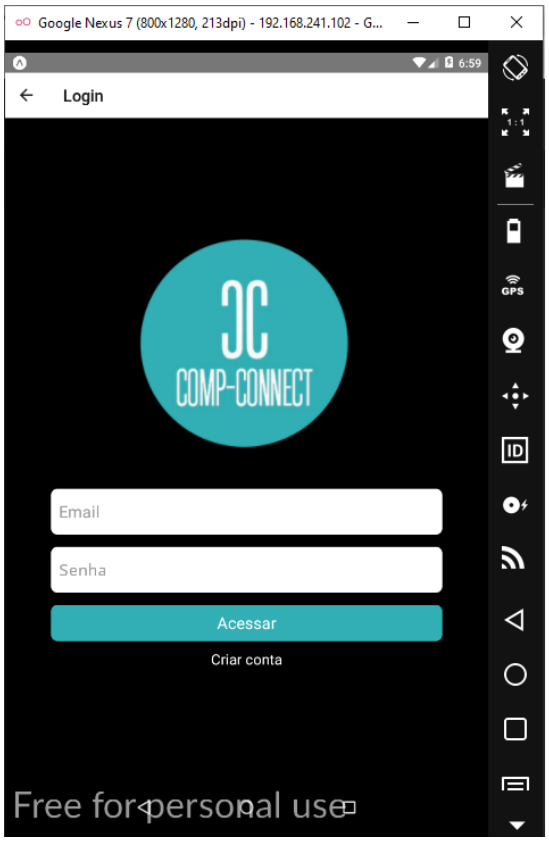
Iteração 10 - 03/10 - 20/10					
US09, US10 – Implementar área do professor e login					
Testes de Aceitação					**Status**
TA9.1	Visualizar informações sobre os professores e verificar o currículo selecionado.				
TA10.1	Adicionar número de matricula e senha do aluno e administradores do site.				
Atividade	**Descrição**	**Responsável**	**Estimativa de tempo**	**Tempo real**	**Status**
A9.1	Criar tela referente a informações sobre os professores	Anderson	9h	12h	
A10.1	Criar tela de login	Anderson	5h	5h	

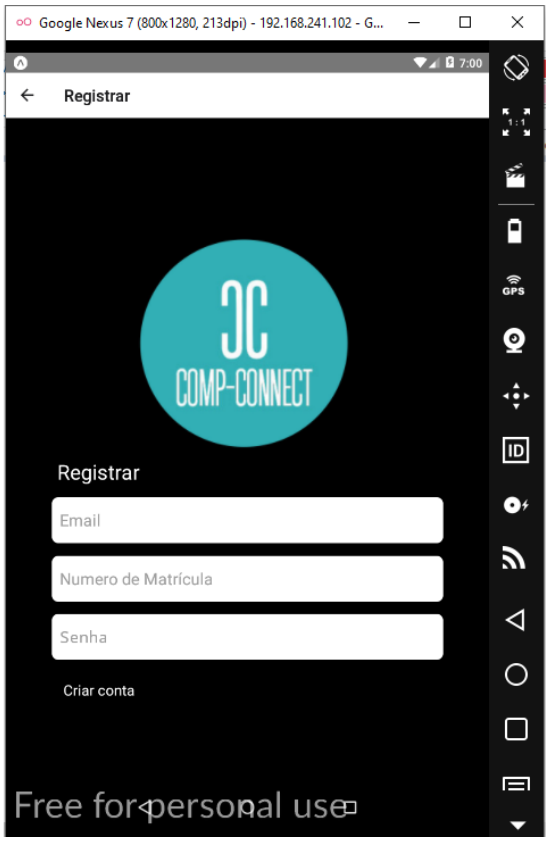
Iteração 11 - 20/10 - 25/10					
US11 – Validar dados e analisar documentação					
Testes de Aceitação					**Status**
TA11.1	Alterar artefatos				
Atividade	**Descrição**	**Responsável**	**Estimativa de tempo**	**Tempo real**	**Status**
A11.1	Analisar artefatos da etapa de planejamento e inicialização	Anderson	10h	12h	

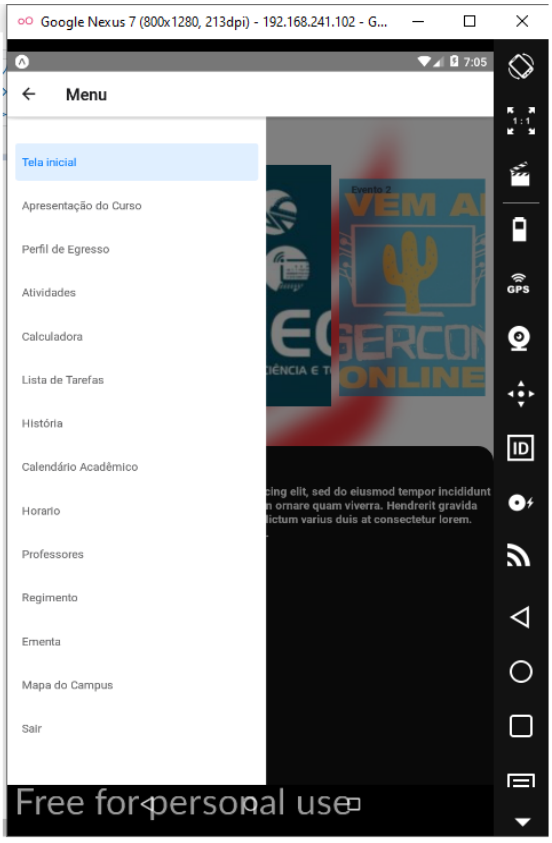
Iteração 12 - 26/10 - 09/11		
US12 - Criar área de cronograma pessoal do aluno.		
Testes de Aceitação		**Status**
TA12.1	Adicionar atividade.	
TA12.2	Verificar funcionalidade do alarme.	
TA12.3	Adicionar checklist.	

Atividade	Descrição	Responsável	Estimativa de tempo	Tempo real	Status
A12.1	Criar área de cronograma pessoal do aluno.	Anderson	4h	4h	
A12.2	Adicionar atividade.	Anderson	4h	5h	
A12.3	Verificar funcionalidade do alarme.	Anderson	5h	7h	
A12.4	Adicionar checklist.	Anderson	4h	5h	

APÊNDICE B – INTERFACE DO APLICATIVO





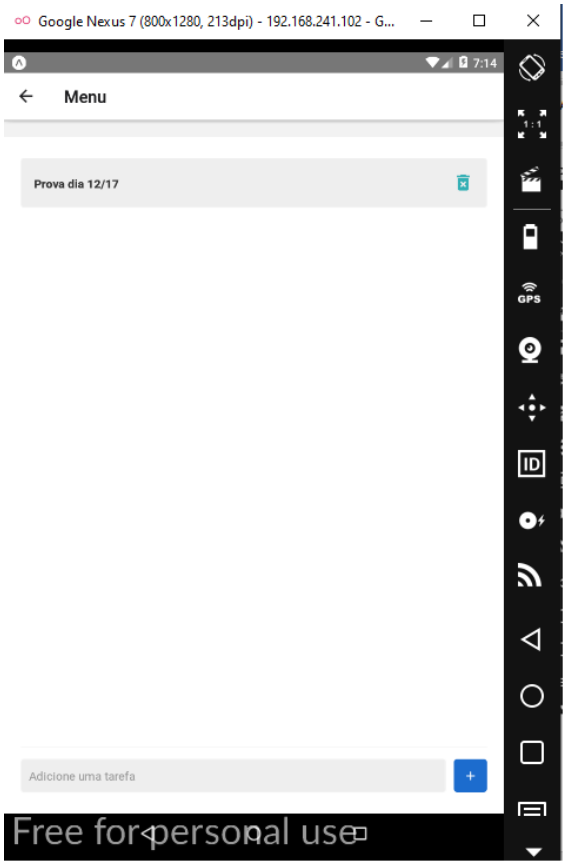


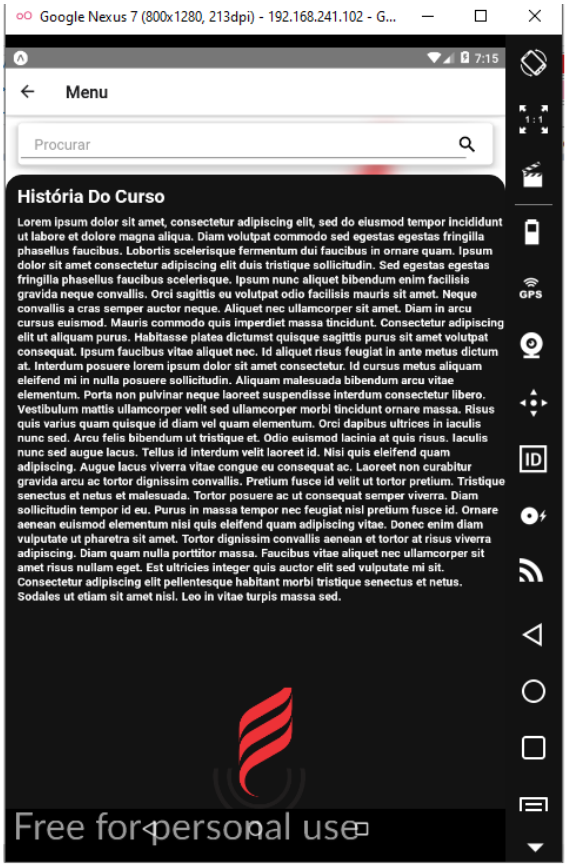

Google Nexus 7 (800x1280, 213dpi) - 192.168.241.102 - G...
7:09
Menu
Procurar
Apresentação Do Curso
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Diam volutpat commodo sed egestas egestas fringilla phasellus faucibus. Lobortis scelerisque fermentum dui faucibus in ornare quam. Ipsum dolor sit amet consectetur adipiscing elit duis tristique sollicitudin. Sed egestas egestas fringilla phasellus faucibus scelerisque. Ipsum nunc aliquet bibendum enim facilisis gravida neque convallis. Orci sagittis eu volutpat odio facilisis mauris sit amet. Neque convallis a cras semper auctor neque. Aliquet nec ullamcorper sit amet. Diam in arcu cursus euismod. Mauris commodo quis imperdiet massa tincidunt. Consectetur adipiscing elit ut aliquam purus. Habitasse platea dictumst quisque sagittis purus sit amet volutpat consequat. Ipsum faucibus vitae aliquet nec. Id aliquet risus feugiat in ante metus dictum at. Interdum posuere lorem ipsum dolor sit amet consectetur. Id cursus metus aliquam eleifend mi in nulla posuere sollicitudin. Aliquam malesuada bibendum arcu vitae elementum. Porta non pulvinar neque laoreet suspendisse interdum consectetur libero. Vestibulum mattis ullamcorper velit sed ullamcorper morbi tincidunt ornare massa. Risus quis varius quam quisque id diam vel quam elementum. Orci dapibus ultrices in iaculis nunc sed. Arcu felis bibendum ut tristique et. Odio euismod lacinia at quis risus. Iaculis nunc sed augue lacus. Tellus id interdum velit laoreet id. Nisi quis eleifend quam adipiscing. Augue lacus viverra vitae congue eu consequat ac. Laoreet non curabitur gravida arcu ac tortor dignissim convallis. Pretium fusce id velit ut tortor pretium. Tristique senectus et netus et malesuada. Tortor posuere ac ut consequat semper viverra. Diam sollicitudin tempor id eu. Purus in massa tempor nec feugiat nisl pretium fusce id. Ornare aenean euismod elementum nisi quis eleifend quam adipiscing vitae. Donec enim diam vulputate ut pharetra sit amet. Tortor dignissim convallis aenean et tortor at risus viverra adipiscing. Diam quam nulla porttitor massa. Faucibus vitae aliquet nec ullamcorper sit amet risus nullam eget. Est ultricies integer quis auctor elit sed vulputate mi sit. Consectetur adipiscing elit pellentesque habitant morbi tristique senectus et netus. Sodales ut etiam sit amet nisl. Leo in vitae turpis massa sed.
Free for personal use


Google Nexus 7 (800x1280, 213dpi) - 192.168.241.102 - G...
7:10
Menu
Procurar
Perfil De Egresso
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Diam volutpat commodo sed egestas egestas fringilla phasellus faucibus. Lobortis scelerisque fermentum dui faucibus in ornare quam. Ipsum dolor sit amet consectetur adipiscing elit duis tristique sollicitudin. Sed egestas egestas fringilla phasellus faucibus scelerisque. Ipsum nunc aliquet bibendum enim facilisis gravida neque convallis. Orci sagittis eu volutpat odio facilisis mauris sit amet. Neque convallis a cras semper auctor neque. Aliquet nec ullamcorper sit amet. Diam in arcu cursus euismod. Mauris commodo quis imperdiet massa tincidunt. Consectetur adipiscing elit ut aliquam purus. Habitasse platea dictumst quisque sagittis purus sit amet volutpat consequat. Ipsum faucibus vitae aliquet nec. Id aliquet risus feugiat in ante metus dictum at. Interdum posuere lorem ipsum dolor sit amet consectetur. Id cursus metus aliquam eleifend mi in nulla posuere sollicitudin. Aliquam malesuada bibendum arcu vitae elementum. Porta non pulvinar neque laoreet suspendisse interdum consectetur libero. Vestibulum mattis ullamcorper velit sed ullamcorper morbi tincidunt ornare massa. Risus quis varius quam quisque id diam vel quam elementum. Orci dapibus ultrices in iaculis nunc sed. Arcu felis bibendum ut tristique et. Odio euismod lacinia at quis risus. Iaculis nunc sed augue lacus. Tellus id interdum velit laoreet id. Nisi quis eleifend quam adipiscing. Augue lacus viverra vitae congue eu consequat ac. Laoreet non curabitur gravida arcu ac tortor dignissim convallis. Pretium fusce id velit ut tortor pretium. Tristique senectus et netus et malesuada. Tortor posuere ac ut consequat semper viverra. Diam sollicitudin tempor id eu. Purus in massa tempor nec feugiat nisl pretium fusce id. Ornare aenean euismod elementum nisi quis eleifend quam adipiscing vitae. Donec enim diam vulputate ut pharetra sit amet. Tortor dignissim convallis aenean et tortor at risus viverra adipiscing. Diam quam nulla porttitor massa. Faucibus vitae aliquet nec ullamcorper sit amet risus nullam eget. Est ultricies integer quis auctor elit sed vulputate mi sit. Consectetur adipiscing elit pellentesque habitant morbi tristique senectus et netus. Sodales ut etiam sit amet nisl. Leo in vitae turpis massa sed.
Free for personal use

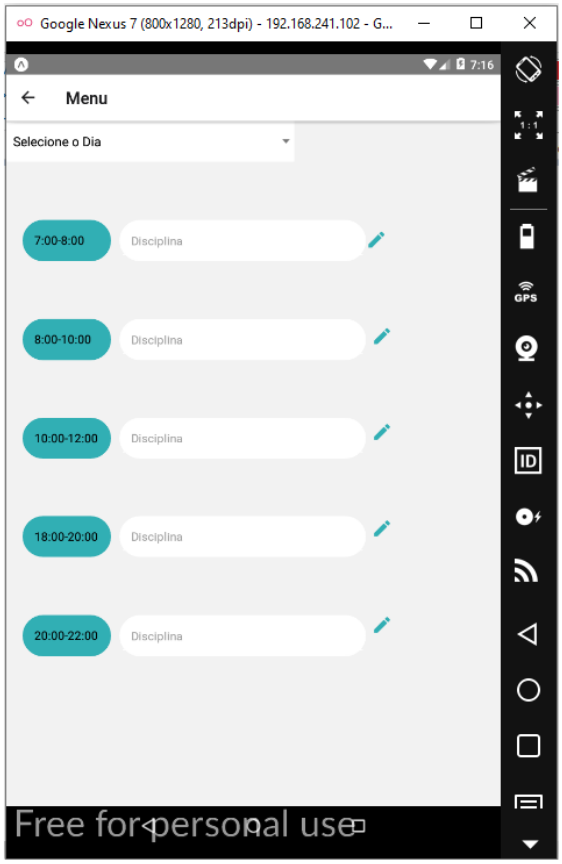

Google Nexus 7 (800x1280, 213dpi) - 192.168.241.102 - G...
7:12
Menu
Procurar
Atividades
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Diam volutpat commodo sed egestas egestas fringilla phasellus faucibus. Lobortis scelerisque fermentum dui faucibus in ornare quam. Ipsum dolor sit amet consectetur adipiscing elit duis tristique sollicitudin. Sed egestas egestas fringilla phasellus faucibus scelerisque. Ipsum nunc aliquet bibendum enim facilisis gravida neque convallis. Orci sagittis eu volutpat odio facilisis mauris sit amet. Neque convallis a cras semper auctor neque. Aliquet nec ullamcorper sit amet. Diam in arcu cursus euismod. Mauris commodo quis imperdiet massa tincidunt. Consectetur adipiscing elit ut aliquam purus. Habitasse platea dictumst quisque sagittis purus sit amet volutpat consequat. Ipsum faucibus vitae aliquet nec. Id aliquet risus feugiat in ante metus dictum at. Interdum posuere lorem ipsum dolor sit amet consectetur. Id cursus metus aliquam eleifend mi in nulla posuere sollicitudin. Aliquam malesuada bibendum arcu vitae elementum. Porta non pulvinar neque laoreet suspendisse interdum consectetur libero. Vestibulum mattis ullamcorper velit sed ullamcorper morbi tincidunt ornare massa. Risus quis varius quam quisque id diam vel quam elementum. Orci dapibus ultrices in iaculis nunc sed. Arcu felis bibendum ut tristique et. Odio euismod lacinia at quis risus. Iaculis nunc sed augue lacus. Tellus id interdum velit laoreet id. Nisi quis eleifend quam adipiscing. Augue lacus viverra vitae congue eu consequat ac. Laoreet non curabitur gravida arcu ac tortor dignissim convallis. Pretium fusce id velit ut tortor pretium. Tristique senectus et netus et malesuada. Tortor posuere ac ut consequat semper viverra. Diam sollicitudin tempor id eu. Purus in massa tempor nec feugiat nisl pretium fusce id. Ornare aenean euismod elementum nisi quis eleifend quam adipiscing vitae. Donec enim diam vulputate ut pharetra sit amet. Tortor dignissim convallis aenean et tortor at risus viverra adipiscing. Diam quam nulla porttitor massa. Faucibus vitae aliquet nec ullamcorper sit amet risus nullam eget. Est ultricies integer quis auctor elit sed vulputate mi sit. Consectetur adipiscing elit pellentesque habitant morbi tristique senectus et netus. Sodales ut etiam sit amet nisl. Leo in vitae turpis massa sed.
Free for personal use

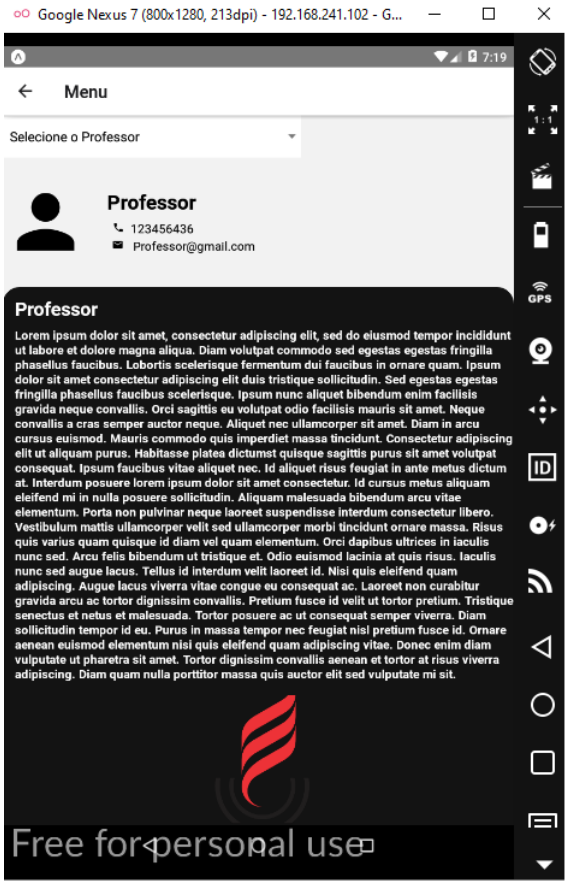

Google Nexus 7 (800x1280, 213dpi) - 192.168.241.102 - G...
7:12
Menu
Grade Curricular
1° Unidade
Enviar
2° Unidade
Enviar
Calcular Média Final
Free for personal use


Google Nexus 7 (800x1280, 213dpi) - 192.168.241.102 - G...
7:14
Menu
Prova dia 12/17
Adicione uma tarefa
Free for personal use

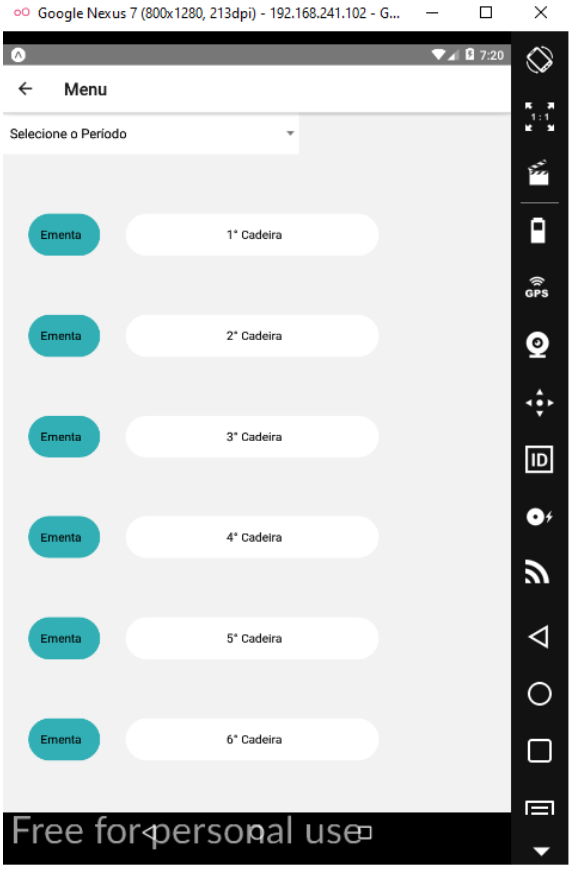

Google Nexus 7 (800x1280, 213dpi) - 192.168.241.102 - G...
7:15
Menu
Procurar
História Do Curso
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Diam volutpat commodo sed egestas egestas fringilla phasellus faucibus. Lobortis scelerisque fermentum dui faucibus in ornare quam. Ipsum dolor sit amet consectetur adipiscing elit duis tristique sollicitudin. Sed egestas egestas fringilla phasellus faucibus scelerisque. Ipsum nunc aliquet bibendum enim facilisis gravida neque convallis. Orci sagittis eu volutpat odio facilisis mauris sit amet. Neque convallis a cras semper auctor neque. Aliquet nec ullamcorper sit amet. Diam in arcu cursus euismod. Mauris commodo quis imperdiet massa tincidunt. Consectetur adipiscing elit ut aliquam purus. Habitasse platea dictumst quisque sagittis purus sit amet volutpat consequat. Ipsum faucibus vitae aliquet nec. Id aliquet risus feugiat in ante metus dictum at. Interdum posuere lorem ipsum dolor sit amet consectetur. Id cursus metus aliquam eleifend mi in nulla posuere sollicitudin. Aliquam malesuada bibendum arcu vitae elementum. Porta non pulvinar neque laoreet suspendisse interdum consectetur libero. Vestibulum mattis ullamcorper velit sed ullamcorper morbi tincidunt ornare massa. Risus quis varius quam quisque id diam vel quam elementum. Orci dapibus ultrices in iaculis nunc sed. Arcu felis bibendum ut tristique et. Odio euismod lacinia at quis risus. Iaculis nunc sed augue lacus. Tellus id interdum velit laoreet id. Nisi quis eleifend quam adipiscing. Augue lacus viverra vitae congue eu consequat ac. Laoreet non curabitur gravida arcu ac tortor dignissim convallis. Pretium fusce id velit ut tortor pretium. Tristique senectus et netus et malesuada. Tortor posuere ac ut consequat semper viverra. Diam sollicitudin tempor id eu. Purus in massa tempor nec feugiat nisl pretium fusce id. Ornare aenean euismod elementum nisi quis eleifend quam adipiscing vitae. Donec enim diam vulputate ut pharetra sit amet. Tortor dignissim convallis aenean et tortor at risus viverra adipiscing. Diam quam nulla porttitor massa. Faucibus vitae aliquet nec ullamcorper sit amet risus nullam eget. Est ultricies integer quis auctor elit sed vulputate mi sit. Consectetur adipiscing elit pellentesque habitant morbi tristique senectus et netus. Sodales ut etiam sit amet nisl. Leo in vitae turpis massa sed.
Free for personal use


Google Nexus 7 (800x1280, 213dpi) - 192.168.241.102 - G...
7:16
Menu
Procurar
Calendário Acadêmico
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Diam volutpat commodo sed egestas egestas fringilla phasellus faucibus. Lobortis scelerisque fermentum dui faucibus in ornare quam. Ipsum dolor sit amet consectetur adipiscing elit duis tristique sollicitudin. Sed egestas egestas fringilla phasellus faucibus scelerisque. Ipsum nunc aliquet bibendum enim facilisis gravida neque convallis. Orci sagittis eu volutpat odio facilisis mauris sit amet. Neque convallis a cras semper auctor neque. Aliquet nec ullamcorper sit amet. Diam in arcu cursus euismod. Mauris commodo quis imperdiet massa tincidunt. Consectetur adipiscing elit ut aliquam purus. Habitasse platea dictumst quisque sagittis purus sit amet volutpat consequat. Ipsum faucibus vitae aliquet nec. Id aliquet risus feugiat in ante metus dictum at. Interdum posuere lorem ipsum dolor sit amet consectetur. Id cursus metus aliquam eleifend mi in nulla posuere sollicitudin. Aliquam malesuada bibendum arcu vitae elementum. Porta non pulvinar neque laoreet suspendisse interdum consectetur libero. Vestibulum mattis ullamcorper velit sed ullamcorper morbi tincidunt ornare massa. Risus quis varius quam quisque id diam vel quam elementum. Orci dapibus ultrices in iaculis nunc sed. Arcu felis bibendum ut tristique et. Odio euismod lacinia at quis risus. Iaculis nunc sed augue lacus. Tellus id interdum velit laoreet id. Nisi quis eleifend quam adipiscing. Augue lacus viverra vitae congue eu consequat ac. Laoreet non curabitur gravida arcu ac tortor dignissim convallis. Pretium fusce id velit ut tortor pretium. Tristique senectus et netus et malesuada. Tortor posuere ac ut consequat semper viverra. Diam sollicitudin tempor id eu. Purus in massa tempor nec feugiat nisl pretium fusce id. Ornare aenean euismod elementum nisi quis eleifend quam adipiscing vitae. Donec enim diam vulputate ut pharetra sit amet. Tortor dignissim convallis aenean et tortor at risus viverra adipiscing. Diam quam nulla porttitor massa. Faucibus vitae aliquet nec ullamcorper sit amet risus nullam eget. Est ultricies integer quis auctor elit sed vulputate mi sit. Consectetur adipiscing elit pellentesque habitant morbi tristique senectus et netus. Sodales ut etiam sit amet nisl. Leo in vitae turpis massa sed.
Free for personal use


Google Nexus 7 (800x1280, 213dpi) - 192.168.241.102 - G...
7:16
Menu
Selecione o Dia
7:00-8:00
Disciplina
8:00-10:00
Disciplina
10:00-12:00
Disciplina
18:00-20:00
Disciplina
20:00-22:00
Disciplina
Free for personal use


Google Nexus 7 (800x1280, 213dpi) - 192.168.241.102 - G...
7:19
Menu
Selecione o Professor
Professor
123456436
Professor@gmail.com
Professor
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Diam volutpat commodo sed egestas egestas fringilla phasellus faucibus. Lobortis scelerisque fermentum dui faucibus in ornare quam. Ipsum dolor sit amet consectetur adipiscing elit duis tristique sollicitudin. Sed egestas egestas fringilla phasellus faucibus scelerisque. Ipsum nunc aliquet bibendum enim facilisis gravida neque convallis. Orci sagittis eu volutpat odio facilisis mauris sit amet. Neque convallis a cras semper auctor neque. Aliquet nec ullamcorper sit amet. Diam in arcu cursus euismod. Mauris commodo quis imperdiet massa tincidunt. Consectetur adipiscing elit ut aliquam purus. Habitasse platea dictumst quisque sagittis purus sit amet volutpat consequat. Ipsum faucibus vitae aliquet nec. Id aliquet risus feugiat in ante metus dictum at. Interdum posuere lorem ipsum dolor sit amet consectetur. Id cursus metus aliquam eleifend mi in nulla posuere sollicitudin. Aliquam malesuada bibendum arcu vitae elementum. Porta non pulvinar neque laoreet suspendisse interdum consectetur libero. Vestibulum mattis ullamcorper velit sed ullamcorper morbi tincidunt ornare massa. Risus quis varius quam quisque id diam vel quam elementum. Orci dapibus ultrices in iaculis nunc sed. Arcu felis bibendum ut tristique et. Odio euismod lacinia at quis risus. Iaculis nunc sed augue lacus. Tellus id interdum velit laoreet id. Nisi quis eleifend quam adipiscing. Augue lacus viverra vitae congue eu consequat ac. Laoreet non curabitur gravida arcu ac tortor dignissim convallis. Pretium fusce id velit ut tortor pretium. Tristique senectus et netus et malesuada. Tortor posuere ac ut consequat semper viverra. Diam sollicitudin tempor id eu. Purus in massa tempor nec feugiat nisl pretium fusce id. Ornare aenean euismod elementum nisi quis eleifend quam adipiscing vitae. Donec enim diam vulputate ut pharetra sit amet. Tortor dignissim convallis aenean et tortor at risus viverra adipiscing. Diam quam nulla porttitor massa quis auctor elit sed vulputate mi sit.
Free for personal use


Google Nexus 7 (800x1280, 213dpi) - 192.168.241.102 - G...
7:19
Menu
Procurar
Regimento
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Diam volutpat commodo sed egestas egestas fringilla phasellus faucibus. Lobortis scelerisque fermentum dui faucibus in ornare quam. Ipsum dolor sit amet consectetur adipiscing elit duis tristique sollicitudin. Sed egestas egestas fringilla phasellus faucibus scelerisque. Ipsum nunc aliquet bibendum enim facilisis gravida neque convallis. Orci sagittis eu volutpat odio facilisis mauris sit amet. Neque convallis a cras semper auctor neque. Aliquet nec ullamcorper sit amet. Diam in arcu cursus euismod. Mauris commodo quis imperdiet massa tincidunt. Consectetur adipiscing elit ut aliquam purus. Habitasse platea dictumst quisque sagittis purus sit amet volutpat consequat. Ipsum faucibus vitae aliquet nec. Id aliquet risus feugiat in ante metus dictum at. Interdum posuere lorem ipsum dolor sit amet consectetur. Id cursus metus aliquam eleifend mi in nulla posuere sollicitudin. Aliquam malesuada bibendum arcu vitae elementum. Porta non pulvinar neque laoreet suspendisse interdum consectetur libero. Vestibulum mattis ullamcorper velit sed ullamcorper morbi tincidunt ornare massa. Risus quis varius quam quisque id diam vel quam elementum. Orci dapibus ultrices in iaculis nunc sed. Arcu felis bibendum ut tristique et. Odio euismod lacinia at quis risus. Iaculis nunc sed augue lacus. Tellus id interdum velit laoreet id. Nisi quis eleifend quam adipiscing. Augue lacus viverra vitae congue eu consequat ac. Laoreet non curabitur gravida arcu ac tortor dignissim convallis. Pretium fusce id velit ut tortor pretium. Tristique senectus et netus et malesuada. Tortor posuere ac ut consequat semper viverra. Diam sollicitudin tempor id eu. Purus in massa tempor nec feugiat nisl pretium fusce id. Ornare aenean euismod elementum nisi quis eleifend quam adipiscing vitae. Donec enim diam vulputate ut pharetra sit amet. Tortor dignissim convallis aenean et tortor at risus viverra adipiscing. Diam quam nulla porttitor massa. Faucibus vitae aliquet nec ullamcorper sit amet risus nullam eget. Est ultricies integer quis auctor elit sed vulputate mi sit. Consectetur adipiscing elit pellentesque habitant morbi tristique senectus et netus. Sodales ut etiam sit amet nisl. Leo in vitae turpis massa sed.
Free for personal use


Google Nexus 7 (800x1280, 213dpi) - 192.168.241.102 - G...
7:20
Menu
Selecione o Período
Ementa
1° Cadeira
Ementa
2° Cadeira
Ementa
3° Cadeira
Ementa
4° Cadeira
Ementa
5° Cadeira
Ementa
6° Cadeira
Free for personal use