



UNIVERSIDADE DE PERNAMBUCO (UPE)  
ESCOLA POLITÉCNICA DE PERNAMBUCO (POLI)  
GRADUAÇÃO EM ENGENHARIA DE COMPUTAÇÃO

ANTONIO MARCOS DE OLIVEIRA JUNIOR

MYCLASS: UM SISTEMA DE COMUNICAÇÃO PARA AGENDA DE AULAS

Recife, Junho de 2019



UNIVERSIDADE DE PERNAMBUCO (UPE)  
ESCOLA POLITÉCNICA DE PERNAMBUCO (POLI)  
GRADUAÇÃO EM ENGENHARIA DE COMPUTAÇÃO

ANTONIO MARCOS DE OLIVEIRA JUNIOR

MYCLASS: UM SISTEMA DE COMUNICAÇÃO PARA AGENDA DE AULAS

Orientador: Prof. Dr. Joabe Bezerra Jesus Júnior

Artigo apresentado à Universidade de  
Pernambuco como parte dos requisitos para a  
obtenção do título de Bacharel em Engenharia de  
Computação

Recife, Junho de 2019

Escola Politécnica de Pernambuco  
Graduação em Engenharia de Computação  
Projeto de Final de Curso



MONOGRAFIA DE FINAL DE CURSO

Avaliação Final (para o presidente da banca)\*

No dia 08/07/2019, às 14h, reuniu-se para deliberar sobre a defesa da monografia de conclusão de curso do(a) discente **ANTONIO MARCOS DE OLIVEIRA JUNIOR**, orientado(a) pelo(a) professor(a) **JOABE BEZERRA DE JESUS JÚNIOR**, sob título MyClass: Um sistema de comunicação para agenda de aulas, a banca composta pelos professores:

**BRUNO JOSÉ TORRES FERNANDES (PRESIDENTE)**  
**JOABE BEZERRA DE JESUS JÚNIOR (ORIENTADOR)**

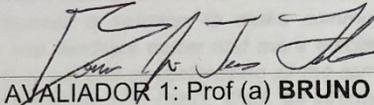
Após a apresentação da monografia e discussão entre os membros da Banca, a mesma foi considerada:

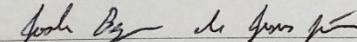
Aprovada       Aprovada com Restrições\*       Reprovada

e foi-lhe atribuída nota: NOVE ( 9,0 )

\*(Obrigatório o preenchimento do campo abaixo com comentários para o autor)

O(A) discente terá 7 dias para entrega da versão final da monografia a contar da data deste documento.

  
AVALIADOR 1: Prof (a) **BRUNO JOSÉ TORRES FERNANDES**

  
AVALIADOR 2: Prof (a) **JOABE BEZERRA DE JESUS JÚNIOR**

AVALIADOR 3: Prof (a)

\* Este documento deverá ser encadernado juntamente com a monografia em versão final.

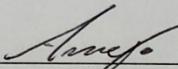
Escola Politécnica de Pernambuco  
Graduação em Engenharia de Computação  
Projeto de Final de Curso

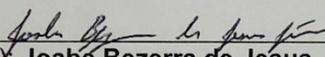


### Autorização de publicação de PFC

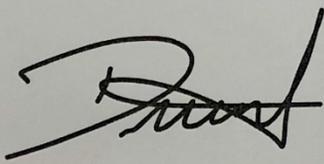
Eu, **Antonio Marcos de Oliveira Junior** autor(a) do projeto de final de curso intitulado: **MyClass: Um sistema de comunicação para agenda de aulas**; autorizo a publicação de seu conteúdo na internet nos portais da Escola Politécnica de Pernambuco e Universidade de Pernambuco.

O conteúdo do projeto de final de curso é de responsabilidade do autor.

  
\_\_\_\_\_  
**Antonio Marcos de Oliveira Junior**

  
\_\_\_\_\_  
Orientador(a): **Joabe Bezerra de Jesus Júnior**

\_\_\_\_\_  
Coorientador(a):

  
\_\_\_\_\_  
Prof, de TCC: **Daniel Augusto Ribeiro Chaves**

\_\_\_\_\_  
Data: 08/07/2019

# MyClass: Um sistema de comunicação para agenda de aulas

Antonio Marcos<sup>1</sup>  [orcid.org/0000-0003-2202-1665](https://orcid.org/0000-0003-2202-1665)

Joabe Jesus<sup>1</sup>  [orcid.org/0000-0002-1518-0572](https://orcid.org/0000-0002-1518-0572)

<sup>1</sup> Escola Politécnica de Pernambuco, Universidade de Pernambuco, Recife, Brasil.

E-mail do autor principal: Antonio Marcos de Oliveira Junior [amoj@ecomp.poli.br](mailto:amoj@ecomp.poli.br)

## Resumo

---

Este trabalho propõe uma solução para tornar a comunicação entre instituições de ensino superior, discentes e docentes mais fácil e eficiente. O objetivo geral é criar um sistema de informação que objetivará a centralização de toda a comunicação de aulas realizadas no contexto da Instituição de Ensino Superior (IES). Assim, toda a informação a respeito métodos de avaliação de disciplinas, dados e modificações de turmas e aulas será compartilhada de forma atômica e instantânea tanto para os docentes quanto para os discentes dentro do sistema.

**Palavras-Chave:** Software; Mobile; Gestão;

## Abstract

---

*This work proposes a solution to make communication between institutions of higher education, students and teachers easier and more efficient. The general objective is to create an information system that will aim at centralizing all the communication of classes held in the context of the Higher Education Institution (HEI). Thus, all information regarding methods of evaluation of disciplines, data and modifications of classes and classes will be shared in an atomic and instantaneous way for both teachers and students within the system.*

**Key-words:** Software; Mobile; Management;

## 1 Introdução

Devido a crescente demanda da população em busca de especialização profissional, como mostra (Pauluze, 2018), o número de universitários cresceu 68% entre 2007 e 2017. Além disso, as Instituições de Ensino Superior (IES) vem expandindo seus campus/unidades e os cursos ofertados, como vem sendo realizado na Universidade de Pernambuco (UPE), que segundo os relatórios anuais (UPE, 2004) e (UPE, 2018) criou 4 novas unidades e 27 novos cursos. Além disso, segundo (UPE, 2018), a UPE possui 11 campus, 15 unidades, 61 cursos e 14.224 alunos.

Neste contexto, um problema tem se tornado relevante: a dificuldade de comunicação entre instituição de ensino (professores e servidores/funcionários) e alunos. Em particular, na UPE, o problema da falta de comunicação é grave devido a grande quantidade de alunos e sua estrutura multicampi. Além disso, mais especificamente na sua unidade de ensino Escola Politécnica de Pernambuco (POLI), o compartilhamento de informações básicas como um cancelamento de aula ou modificações na programação se tornaram tarefas difíceis de acompanhar, gerando transtornos e desgastes desnecessários.

Assim, tarefas envolvendo informações como localização de salas de aula, atrasos e cancelamentos de aula ou acontecimentos de uma turma são difíceis de se obter. Além disso, para conseguir esse tipo de informação pode ser necessário estar presencialmente na universidade.

Numa IES, a Gestão Educacional é fundamental para que os resultados sejam melhores, o processo de ensino e aprendizagem se torne mais eficaz e a IES seja mais reconhecida. Segundo a especialista em educação Heloísa Luck (Nadal, 2009), a Gestão escolar relaciona-se a uma atuação que foca em promover a organização, mobilização e articulação das condições essenciais. Além disso, essa gestão visa garantir o avanço do processo socio educacional das instituições de ensino e possibilitar que elas promovam o aprendizado dos estudantes de forma efetiva.

Uma das formas de otimizar a Gestão Educacional é a utilização de softwares dedicados a realização de tarefas burocráticas do cotidiano, assim toda a informação necessária se torna organizada e de fácil acesso. Existem diversos sistemas no mercado com foco na Gestão Educacional como: o Sig@ da Universidade Federal de Pernambuco (UFPE); o SIGAA da Universidade federal do rio grande do norte (UFRN); o TOTVS Educacional; e o SPONTE. Contudo, visam, em essência, resolver problemas complexos da administração acadêmica, financeira e de pessoal das instituições. Assim, nenhum destes visa facilitar

a comunicação entre instituição e discentes através de uma única plataforma.

O objetivo deste trabalho é o desenvolvimento de uma aplicação móvel que deve sistematizar o envio de notificações de forma automatizada dentro da IES, melhorando e otimizando o processo de transmissão e compartilhamento de informação além de facilitar o consumo das informações por parte dos discentes e docentes.

Este artigo está organizado em 5 seções: na seção 2 é apresentado o background da solução a ser proposta. Na seção 3 são definidos os materiais e métodos para o desenvolvimento deste trabalho. Na seção 4 é apresentada as primeiras etapas da construção do software. Na seção 5 é apresentado os protótipos iniciais e de testes da solução. Finalmente, na seção 6 são apresentadas as conclusões e os impactos esperados do trabalho.

## 2 Referencial Teórico

### 2.1 Engenharia de Software

A engenharia de software consiste na parte técnica e científica dos sistemas, desde a fase de desenvolvimento até a gestão sempre visando organização, produtividade e qualidade.

A ideia surgiu devido a uma crise de software vivida por volta de 1960 e para dar um tratamento de engenharia ao desenvolvimento de sistemas de software complexos, a ciência envolve o uso de modelos abstratos e precisos que permitem ao engenheiro especificar, projetar, implementar e manter softwares garantindo e mantendo sua qualidade, também busca definir alguns métodos e estudos que devem ser realizados antes, durante e após o desenvolvimento de software, para isso algumas atividades são fundamentais, tais como: especificação, projeto, implementação, validação e evolução.

Existem vários modelos de processo de software, contudo três modelos são os mais tradicionais: cascata, incremental e orientado a reuso, todos os modelos sofrem ajustes e adequações a depender do contexto em que são aplicados, pois quando postos em prática sempre nota-se que algum ponto pode ser melhorado ou modificado para que o modelo seja mais eficiente.

## 2.2 Modelo Evolucionário

Para a execução do projeto será utilizada a metodologia de desenvolvimento evolucionária, mais especificamente o modelo de prototipação. Esta decisão reflete o contexto no qual as necessidades do negócio e do produto mudam frequentemente a medida que o projeto vai sendo materializado. Assim, a metodologia evolucionária consiste no desenvolvimento do software à medida em que as necessidades de novas *features* (características) venham surgindo. Com isso é reduzida a necessidade de recodificação e do escopo de utilização resultando assim num software que atende a necessidade da maioria dos usuários que irão manipular o sistema. Os modelos evolucionários possuem dois principais modelos de desenvolvimento: Prototipação e Espiral, sendo o Modelo de Prototipação normalmente utilizado quando não se tem certeza, ou ainda não está muito claro, o escopo e limitações do sistema (Devmedia, 2019).

A prototipação se dá fundamentalmente com a comunicação que ocorre nas reuniões com os envolvidos a fim de se definir os objetivos e identificar os requisitos que até então são conhecidos e também planejar as áreas que precisam ser mais bem projetadas. Cada iteração deve ser planejada rapidamente nos moldes de um projeto de metodologia ágil, focado sempre nos aspectos do software visíveis ao usuário como interface, formatos e iteratividade. Como visto na Fig. 1, Etapas da Prototipação.

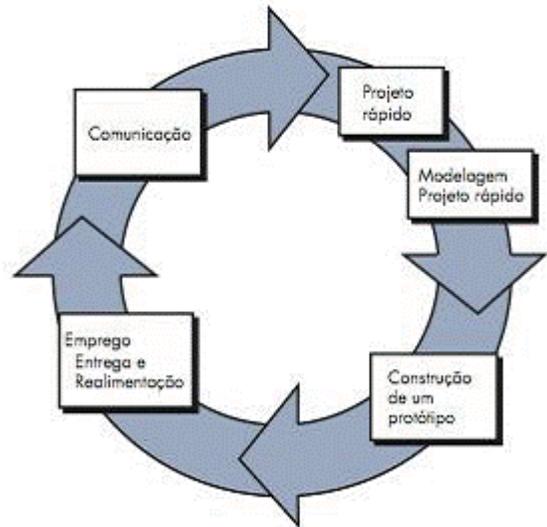


Fig. 1 - Etapas da Prototipação

## 2.3 Plataformas para Aplicativos Móveis

### 2.3.1 A plataforma Android

O Android se tornou a plataforma mais utilizada do mundo. Já em 2013 obteve a maior porcentagem de vendas de plataformas móveis e em 2014 levantou-se que 1 bilhão de usuários utilizam o sistema. O Android se tornou muito comum entre empresas e desenvolvedores que buscam desenvolver softwares de baixo custo, personalizáveis e com alto potencial de popularização entre os usuários. Além disso, o código do sistema é aberto e disponibilizado pela Google.

O sistema operacional (SO) é baseado no núcleo Linux e com interface baseada na manipulação direta, o SO foi lançado oficialmente em sua versão alpha em 2007, e logo em seguida trouxe sua versão comercial na versão 1.0 em meados de setembro de 2008, desde então vem em constante desenvolvimento pela Google, desde seu lançamento o sistema vem trazendo uma série de novidades e melhoramentos com suas atualizações, tais como correções de erros, falhas de segurança e melhoria de desempenho, cada versão foi lançada com o codinome de um doce, hoje o sistema se encontra na versão 9.0 de codinome Pie.

#### 2.3.1.1 Desenvolvimento de Aplicativos para a plataforma Android

Existem duas formas de desenvolvimento de apps para a plataforma, a utilização de linguagem nativa e híbrida, a linguagem nativa como o nome sugere é o desenvolvimento que se dá utilizando ferramentas oficiais criadas e disponibilizadas pela empresa responsável pelo software, no momento em que esse artigo é escrito existem duas linguagens de programação oficiais: JAVA e Kotlin, ficando a escolha do desenvolvedor decidir qual das linguagens utilizar ou até mesmo combinar as duas,

sendo cada arquivo em uma das linguagens, sendo a IDE oficial de desenvolvimento o Android Studio, disponibilizada gratuitamente.

Para o deploy das aplicações em produção a Google oferece a ferramenta "Play Console", com a ferramenta é possível realizar o gerenciamento de versão e distribuição das aplicações, também na plataforma é disponibilizado dados gerenciais de download e utilização, para uso da ferramenta é necessário realizar o cadastro com uma conta Google e efetuar o pagamento de uma taxa de US\$ 25,00 por um acesso de validade vitalícia.

### **2.3.1.2 Distribuição de Aplicativos Android**

A distribuição de aplicativos acontece na loja de aplicativos da Google "Play Store", devido ao grande fluxo de aplicações a fiscalização é muito baixa, fazendo com não exista muita burocracia para publicação de apps, porém como consequência existe uma grande quantidade de aplicativos maliciosos sendo disponibilizados.

### **2.3.2 A Plataforma Apple iOS**

O Sistema Operacional iOS foi desenvolvido e pertence a Apple. Criado inicialmente para o iPhone, foi desenvolvido para possuir exclusividade de hardware, uma das características que faz com que o sistema possua uma fluidez e eficiência conhecidas, sua interface é baseada no conceito de manipulação direta, e sua primeira versão foi lançada em 2007. A Apple lança novas versões do SO periodicamente, em intervalos de aproximadamente um ano. na época atual a versão mais nova é a 12 disponível para versões do iPhone superiores a 5S.

#### **2.3.2.1 Desenvolvimento de Aplicativos para a plataforma iOS**

O desenvolvimento de aplicações iOS é feito através da linguagem Swift ou Objective-C, ficando a escolha do desenvolvedor, podendo inclusive ser utilizado no mesmo arquivo código nas duas linguagens. A IDE oficial de desenvolvimento é o

Xcode disponibilizado gratuitamente através da App Store da Apple.

Para deploy em produção a Apple disponibiliza a plataforma "Apple Developer", onde é possível realizar todo o gerenciamento da aplicação. para uso da ferramenta é necessário realizar o cadastro com uma conta iCloud e realizar o pagamento de uma taxa de US\$ 99,00 por ano.

#### **2.3.2.2 Distribuição de Aplicativos iOS**

A distribuição das aplicações acontece na sua loja de aplicativos "APP Store". Nessa loja online, a Apple analisa todos os aplicativos que disponibiliza para os seus usuários, gerando uma certa burocracia e sendo necessário seguir vários guidelines de desenvolvimento para que a aplicação não seja rejeitada. Por isso, os apps disponibilizados são muito seguros. Além disso, quando não aprovados, os aplicativos são devolvidos ao desenvolvedor para que realize as modificações necessárias.

### **2.4 Armazenamento de Dados e plataformas móveis**

O armazenamento de dados das aplicações pode ser realizado de diversas maneiras a depender da mecânica da aplicação, em regras gerais os apps trabalham armazenando dados de três formas: online, local com sincronia online e local. Para uma maior disponibilidade ao usuário o ideal é que a aplicação armazene dados localmente com sincronia online, pois quando o usuário se encontra com uma conexão de má qualidade ainda assim conseguirá utilizar a aplicação. Esse armazenamento pode ser feito de duas maneiras, em um banco de dados SQL (Relacionais) e NoSQL (Não Relacional), cada uma possui características que a depender do objetivo da aplicação levam a ser a melhor tecnologia a ser empregada.

#### **2.4.1 NoSQL**

As bases de dados NoSQL surgiram como um padrão de armazenamento de dados alternativo ao SQL padrão, e tendo como principal objetivo oferecer maior robustez e escalabilidade que os formatos utilizados até então.

O NoSQL é considerado um banco de dados não relacional e a necessidade de um padrão tão diferente dos modelos convencionais veio da popularização da internet, onde houve uma explosão na quantidade de dados armazenados e em trânsito na rede, fazendo com que gradualmente fosse aumentando o custo e a complexidade do tratamento da informação nos modelos convencionais.

Os primeiros a adotar o NoSQL foram as redes sociais, pois existe uma grande geração de conteúdo pelos dispositivos móveis, e um número cada vez maior de pessoas e dispositivos conectados, fazendo com que o armazenamento e utilização desses dados nas ferramentas de análise de dados enfrente problemas de escalabilidade e manutenção.

#### 2.4.2 Firebase

Firebase é uma plataforma de desenvolvimento mobile e web adquirida pela Google em 2014, que surgiu da necessidade dos desenvolvedores construírem aplicações de forma mais rápida e fácil, possui foco em ser um back-end completo e de fácil usabilidade, envolve uma série de tecnologias com finalidades diferentes, tais como o armazenamento de dados, análises de performance, envio de notificações entre outras, todas essas ferramentas são disponibilizadas para várias linguagens, tais como: Java, Swift, Python e etc, o grande intuito do Firebase é centralizar na plataforma o que antes tinha como ponto de sincronização os servidores, que exigiam altos custos de manutenção e complexidade elaboração de aplicações, assim o Firebase agrega em uma única aplicação funcionalidades de vários serviços de nuvem.

O Firebase oferece serviços fáceis de implementar, sendo uma boa alternativa para o desenvolvimento de aplicações que exigem uma rápida fase de prototipação e que tenham foco na interação com o usuário.

#### 2.4.3 Firestore

O Firestore é um banco de dados escalável e flexível para o desenvolvimento de aplicações para

dispositivos móveis, baseado num banco de dados NoSQL com armazenamento e sincronia de dados na nuvem, esta ferramenta permite a o desenvolvedor enviar dados ao servidor e instantaneamente sincronizar dados com qualquer dispositivo conectado.

Um grande diferencial da ferramenta é oferecer suporte offline, armazenando em cache dados que são usados ativamente pela aplicação, Assim o aplicativo possui a capacidade de ler, escrever e consultar dados mesmo que o dispositivo esteja desconectado, e quando o dispositivo retorna online as alterações são sincronizadas com o Firestore, essa característica permite criar aplicações que funcionam independentemente da latência e qualidade da internet.

Toda a informação armazenada no Firestore é inserida em documentos que possuem mapeamento de campos e valores, e cada documento é armazenado em uma coleção, que são similares a contêineres de documentos criados para organizar os dados e gerar as consultas.

Os dados contidos no documento podem ser de diferentes tipos de dados, desde estruturas primitivas a objetos complexos, também é possível criar subcoleções dentro dos documentos gerando assim estruturas hierárquicas e que podem ser escalonadas à medida que o banco cresce.

Apesar da complexidade que um banco Firestore pode possuir, é possível realizar consultas somente num determinado nível da estrutura e também realizar especificações quantitativas e qualitativas da informação desejada.

Atualmente a ferramenta se encontra na sua versão Beta de desenvolvimento, porém já com um bom grau de estabilidade e amadurecimento.

### 3 Materiais e Métodos

Para a construção do Sistema MyClass é utilizada a metodologia de Prototipação Evolucionária. Em particular, executamos dois ciclos de construção, cada um obedecendo as etapas da Fig. 1 apresentada anteriormente. Assim, a construção do projeto é feita nas 5 etapas do primeiro ciclo e o projeto é ajustado para evoluir ao próximo ciclo. No primeiro ciclo é utilizada a modelagem Entidade-Relacionamento (ER) e o protótipo foi desenvolvido para iOS. Já no segundo ciclo, é feita uma modelagem de Casos de Uso de Negócio para melhor entendimento do contexto da aplicação e o protótipo é evoluído para iOS e para Android.

As linguagens escolhidas para desenvolvimento foram: Java para a plataforma Android e Swift para a plataforma iOS. Além disso, como tecnologia de armazenamento foi utilizado o Firestore.

## 4 Construção do Sistema MyClass

Esta seção apresenta os dois ciclos de construção do Sistema MyClass usando prototipação evolucionária.

### 4.1 Projeto rápido (v1)

Antes de iniciar o desenvolvimento do software procurou-se entender melhor o problema da falta de comunicação entre os discentes e docentes, foi realizado um questionário de validação do problema, para essa validação foi utilizado um questionário do Google Forms para estudo do comportamento de alunos e professores, a pesquisa foi realizada com 13 alunos da universidade, a primeira pergunta realizada foi: "Você utiliza ferramentas para se comunicar com seu professor ?" Como visto na Fig.2, Pergunta 1.

Você utiliza ferramentas para se comunicar com seu professor?  
13 respostas

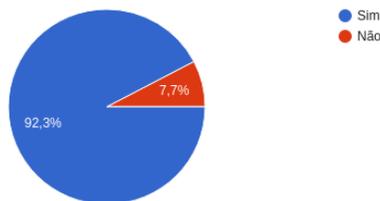


Fig. 2 - Pergunta 1

Como mostrado, mais de 92% dos estudantes que responderam a pesquisa assinalaram que utilizam alguma ferramenta para se comunicar, e aproximadamente 7% não utilizam.

A segunda pergunta realizada foi: "A comunicação feita por aplicativos ajuda no andamento da disciplina?", o resultado é mostrado na Fig.3, Pergunta 2.

A comunicação feita por aplicativos ajuda no andamento da disciplin  
13 respostas

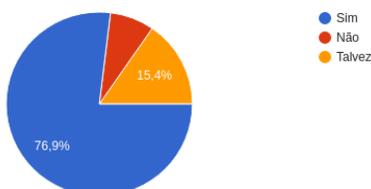


Fig.3, Pergunta 2

Como mostrado, aproximadamente 77% consideram que é benéfica a comunicação via aplicativos, 15,4% são indiferentes, e 7,7% consideram que não ajuda.

A terceira pergunta realizada foi: "Quais ferramentas você utiliza ou utilizou durante a universidade?", o resultado é mostrado na Fig.4, Pergunta 3.

Quais ferramentas você utiliza ou utilizou durante a universidade  
13 respostas

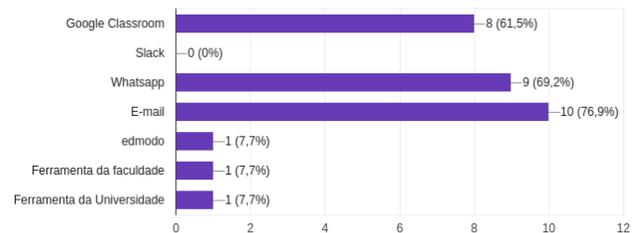


Fig.4, Pergunta 3

Como mostrado, a grande maioria utiliza uma ou mais ferramentas de comunicação, contudo três aplicações aparecem com mais frequência, são elas: E-mail, Whatsapp e Google Classroom, nesta ordem.

A partir da pesquisa realizada e dos dados levantados, concluiu-se que de fato existe dificuldades na comunicação dentro da universidade, uma vez que a grande maioria utiliza ferramentas que não foram criadas com este propósito ou a executam de forma não eficiente ou inapropriada.

### 4.2 Modelagem do projeto rápido (v1)

O sistema foi modelado considerando o contexto da Escola Politécnica de Pernambuco (POLI), na qual temos dois atores principais, os professores e os funcionários técnico-administrativos que controlam as salas de aula. Em particular, a divisão de Apoio Técnico-Pedagógico (DATP) representa esse controle de salas na POLI. Para modelarmos o sistema foi criado um modelo Entidade-Relacionamento (ER) da base de dados (discutida na seção 4.2.1) que trata dos dois tipos de atores, mas vale ressaltar que apenas os funcionários do DATP são usuários do sistema.

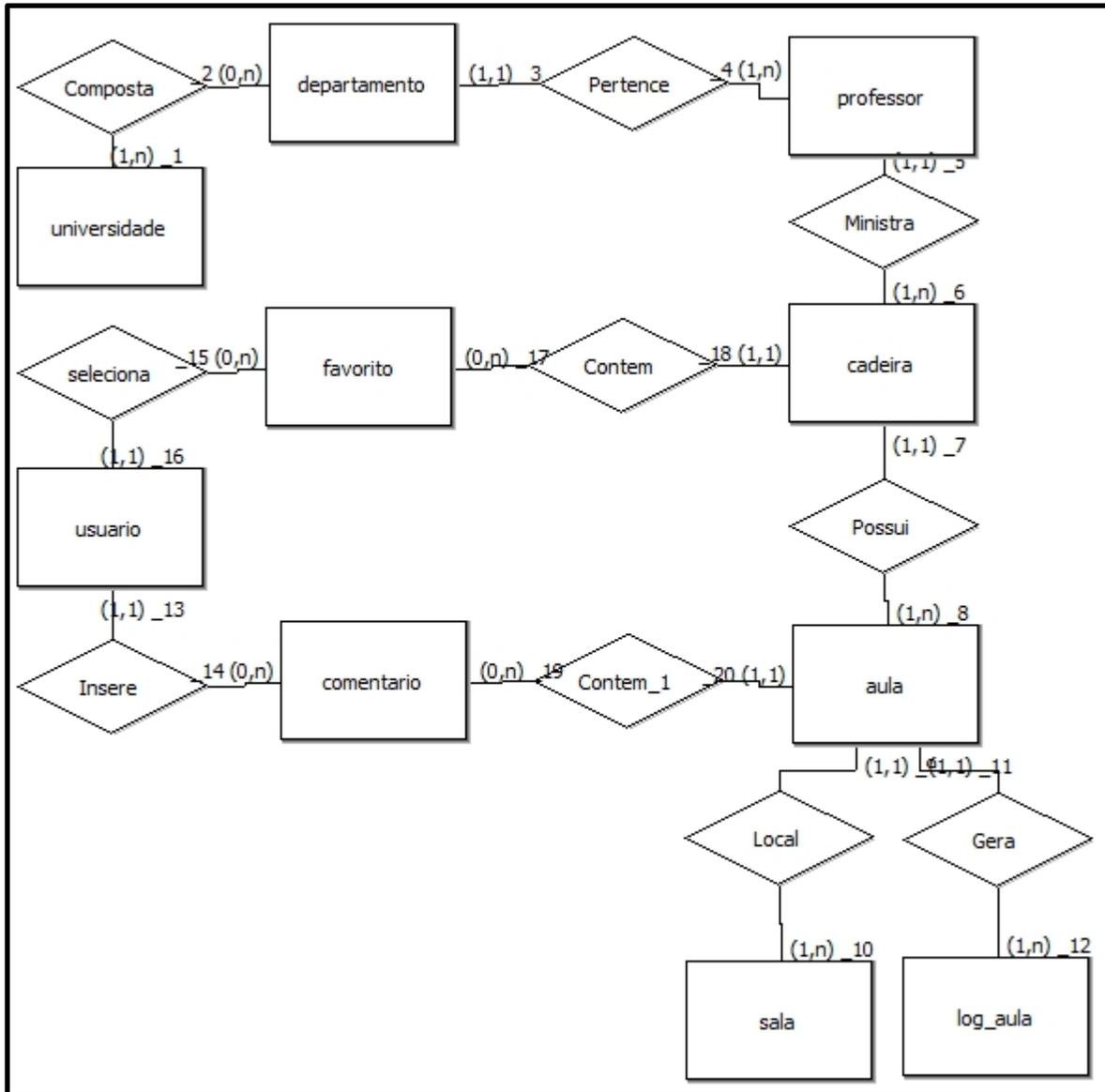


Fig.5 Modelo Entidade-Relacionamento (ER)

#### 4.2.1 Modelagem da Base de Dados

Inicialmente a base de dados da aplicação foi pensada para ser uma base de dados comum do tipo relacional, e para que funcionasse com um software padrão utilizado hoje comercialmente como MySQL, Postgree ou MariaDB, e como uma das primeiras etapas do desenvolvimento de uma base deste tipo é gerar o modelo entidade-relacionamento este foi gerado e o resultado é mostrado na Fig.5. Modelo Entidade-Relacionamento (ER).

Contudo, na etapa inicial de desenvolvimento do software foram detectados diversos problemas e limitações de custo e

implementação que utilizar um software convencional iria trazer ao projeto, por isso foi decidido que a melhor alternativa era buscar uma solução de armazenamento de dados voltado ao desenvolvimento rápido de aplicações e a solução mais viável foi a plataforma Firestore do Firebase, como descrito no início deste artigo, a plataforma apresenta uma série de vantagens com relação a tempo e custo de desenvolvimento, contudo por se tratar de uma solução NoSQL a modelagem apresentada anteriormente não seria aplicada como planejada, por isso foi necessário realizar algumas

modificações para atender ao padrão de armazenamento do Firestore.

### 4.3 Protótipo (v1)

Neste protótipo toda a aplicação foi pensada para ser gerenciada por uma única entidade, o DATP, assim a entidade deveria realizar todo o controle de inserção, edição e deleção de professores, turmas e aulas, através de uma plataforma web, onde ao receber as informações vindas dos docentes ou responsáveis de turma, deveria acessar a plataforma e executar os procedimentos necessários.

Na aplicação mobile, em sua tela inicial, foi proposto apresentar um menu em barra inferior que será mostrada em toda a aplicação para que o usuário identifique rapidamente em que funcionalidade se encontra, além de cumprir os guidelines de desenvolvimento de aplicações Android e IOS modernas. Na barra constará as seguintes opções: "Professores", "Cadeiras" e "Aulas", cada uma dessas opções foram pensadas para realizar as funcionalidades que serão descritas a seguir.

Por padrão a funcionalidade selecionada ao abrir a aplicação é a "Professores", nela são listados os professores que se encontram ativos dentro da universidade para que o usuário localize a turma ou a aula que deseja obter informações filtrando pelo professor que irá ministrar a aula, como mostrado na Fig.6, Tela Professores.



Fig.6, Tela Professores

Na funcionalidade "Cadeiras", é apresentada a lista das cadeiras oferecidas pela instituição, para que o usuário localize a aula ou turma que deseja obter informações, sendo possível filtrar pelo nome da cadeira, e após selecionar, será apresentada a lista de turmas disponíveis, como mostrado na Fig.7, Tela Turma.

Na Funcionalidade "Aulas", é apresentada as próximas aulas que o usuário irá assistir, para que suas aulas sejam mostradas nessa tela é necessário que o aluno acesse a turma que deseja favoritar, e clicar no coração que representa a seleção da turma para suas favoritas, depois disso todas as aulas serão listadas nessa funcionalidade em ordem crescente da data de ocorrência, e após selecionar uma dessas aulas será apresentada todas as informações da aula, bem como sua localização no mapa e um chat para troca de informação dentro da turma, como mostrado na Fig.8 Tela Aula.

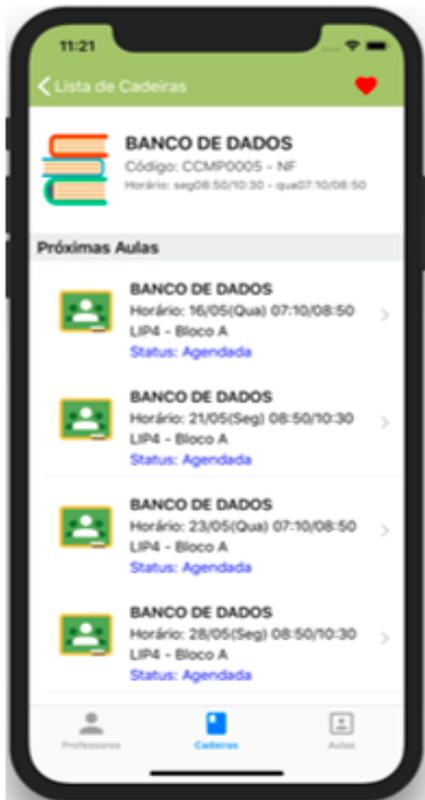


Fig.7, Tela Turma



Fig.8, Tela Aula

#### 4.4 Emprego (Entrega Feedback) e Comunicação (v1)

Diante do contexto em que o problema se encontrava, decidiu-se criar um modelo de negócio para que se compreendesse melhor a mecânica de funcionamento da universidade, e assim poder levantar com mais clareza os requisitos necessários para a criação do sistema, baseado nas necessidades dos usuários.

Baseado nos feedbacks percebeu-se que o controle estava centralizado demais no DATP e conseqüentemente o sobrecarregando, assim foi decidido descentralizar o controle, e conseqüentemente se fez necessário reformular a base de dados.

Além disso alguns layouts e a mecânica de navegação na aplicação foram reformuladas para dar mais fluidez e usabilidade ao aplicativo. Exemplo: o chat entre alunos e professores que estava disponível dentro da área de aulas e foi avaliado que seria mais interessante que os comentários fossem realizados dentro da área de turma.

## 4.5 Projeto Rápido (v2)

Para visualizar com mais clareza o contexto, foi realizado a modelagem de negócio que é apresentado na Fig.5, Modelo de Negócio.

Fig.5, Modelo de Negócio

Essa modelagem fez com que fosse possível visualizar a mecânica de controle da aplicação, e realizar modificações de forma mais padronizada.

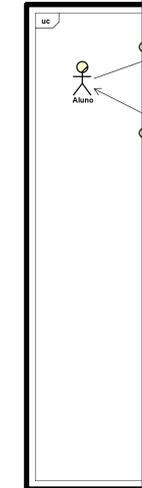
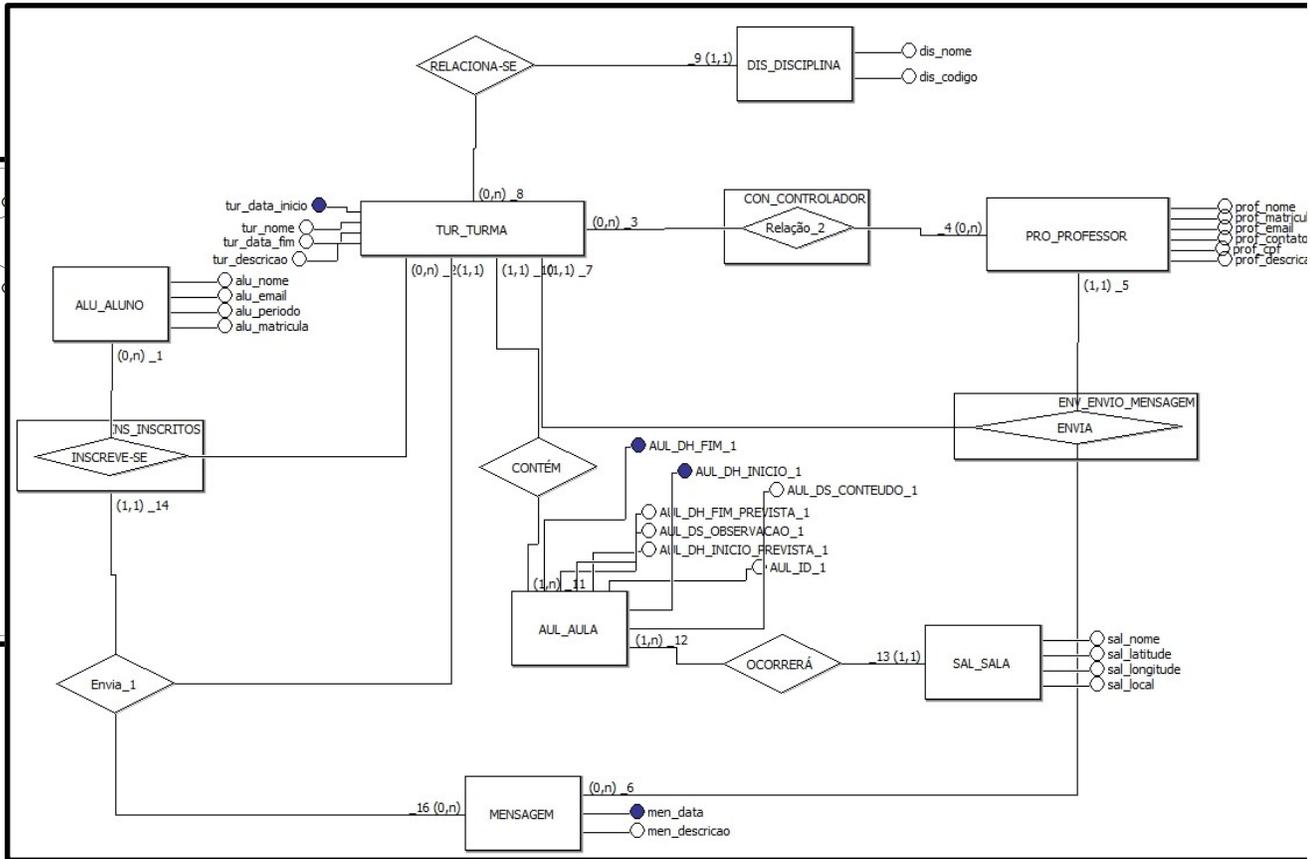


Fig.6 Base de dados v2

## 4.6 Modelagem do Projeto Rápido (v2)

### 4.7.1 Modelagem da Base de Dados

Na segunda versão do protótipo, o controle dos dados foi descentralizado, pois baseado nos feedbacks recebidos, centralizar o controle no DATP iria sobrecarregar a entidade e conseqüentemente tornar a aplicação ineficiente, assim a base de dados foi remodelada para que os usuários com determinadas permissões, atribuídas pelos responsáveis do controle da aplicação, tivessem autonomia para realizar modificações, então foram criados dois perfis de administração: "Administrador" e "Docente", o administrador tem permissão total sobre o sistema, podendo realizar o CRUD de qualquer Turma, Aula ou Docente, já o

perfil "Docente", pode realizar o CRUD somente de turmas e aulas que estiverem sob sua responsabilidade. A modelagem é apresentada na Fig.6 Base de dados v2.

### 4.7 Protótipo (v2)

Dando continuidade ao ciclo evolucionário, iniciou-se de fato o desenvolvimento da segunda versão, que pode ser considerado um melhoramento do protótipo anterior e foi confeccionado com base no feedback dos discentes e docentes que utilizaram a aplicação inicial.

O menu apresentado na barra inferior foi modificado para apresentar melhor usabilidade, nesta versão as funcionalidades apresentadas na

barra são: "Home", "Professores", "Turmas", "Eventos", "Ajustes".

Na tela inicial da aplicação, por padrão a opção do menu selecionada é a "Home", nela são listadas as turmas favoritas do discente/docente e um feed de notícias onde são listadas todas as informações de atualização a respeito de: horário, status e local das aulas cujo usuário tenha favoritado, como mostrado na Fig.8 Tela Home.



Fig.8 Tela Home

Na tela mostrada acima, é possível facilmente localizar uma turma em que se tenha previamente favoritado, e clicando no ícone ser rapidamente redirecionado para a tela da turma em que se deseja obter informações. Além disso é possível acompanhar em tempo real todos os updates das aulas que serão realizadas das turmas favoritadas através do Feed de notícias.

Na funcionalidade "Professores", a mecânica da aplicação permaneceu inalterada, sendo nela listados todos os professores da universidade, com opção de filtragem e após a seleção do professor, a aplicação fornece os dados e as turmas que o docente está responsável como mostrado na Fig.9 Tela Professor.



Fig.9 Tela Professor

Na tela acima, é mostrado um campo "Sobre" que é editável pelo professor e este pode inserir informações sobre sua formação, metodologia de aula, materiais exigidos etc., além disso no campo "Turmas", são listadas as turmas em que o docente é responsável no período corrente.

A terceira funcionalidade do menu é a “Turmas”, nela são listadas todas as turmas da universidade, com opção de filtragem e após a seleção da turma desejada é mostrada a tela da Fig.10 Tela Turma.



Fig.10 Tela Turma

Na tela mostrada, o campo “Próximas Aulas” lista todas as aulas que irão ocorrer para a turma selecionada na ordem crescente de data, o campo “Comentários”, fornece uma área para troca de mensagens instantâneas entre os usuários que desejam se comunicar, seguindo o feedback dado no ciclo 1 para que os comentários sejam a nível de turma.

Caso o usuário deseje que seja mostrado em seu feed os dados a respeito da turma, e notificações quando modificações forem feitas em suas aulas,

pode favoritar a turma através do clique no botão azul com o ícone de coração.

Ao clicar em uma aula, o usuário será direcionado a tela de detalhes da aula como mostrado na Fig.11 Detalhes Aula.

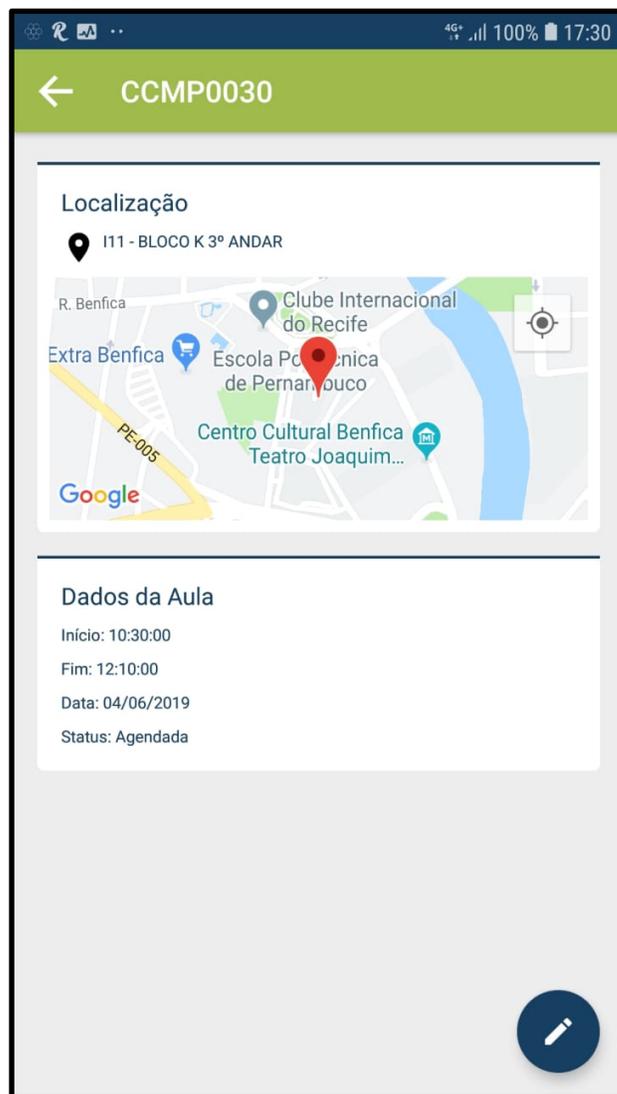


Fig.11 Detalhes Aula

Na tela acima o campo “Localização” exibe um mapa com a localização atual do usuário e da sala onde irá ocorrer a aula, através do mapa o usuário pode verificar seu deslocamento em tempo real para encontrar com mais facilidade sua sala. além disso contém um pequeno descritivo do local da sala como o andar em que se encontra. No campo “Dados da Aula” o usuário de encontrar informações como data, horário e status da aula.

Caso o usuário que se encontra nesta tela seja responsável pela aula, será mostrado um botão

com o ícone de um lápis, esse botão serve para a edição dos dados da aula, ao clicar no botão o usuário será direcionado para a página da Fig.12 Edição de Aula.



Fig.12 Edição de Aula

Na tela acima, no campo "Localização" o usuário pode clicar no spinner e selecionar algumas das salas de aulas possíveis, e assim modificar o local onde a aula será realizada, no campo "Dados da Aula" é possível editar o horário de início e fim da aula, e além disso é possível editar o status da aula, modificando qualquer uma dessas informações, todos os usuários que favoritaram a turma receberão notificação de push em seus smartphones instantaneamente, e a informação passará a constar em seus feeds de notícias.

#### 4.6 Emprego (Entrega Feedback) e Comunicação (v2)

O Protótipo v2 foi concluído e está pronto para testes, foi realizado o release da aplicação em versão Beta de testes na "Play Store" e já se encontra disponível para download como mostrado na Fig.13 Aplicação publicada.

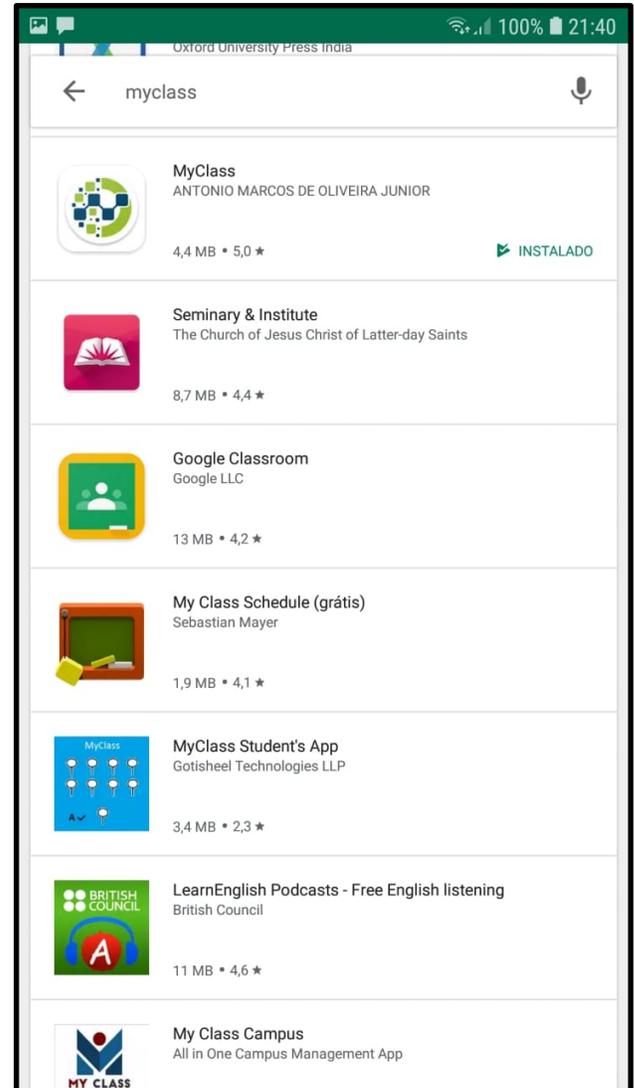


Fig.13 Aplicação Publicada

Espera-se obter novos dados de feedbacks, com essas informações a aplicação deve ser submetida a novos ciclos de evolução para se obter uma aplicação cada vez mais funcional, útil e de boa aparência.

#### 5. Conclusões

Soluções tecnológicas vêm tornando a vida das pessoas mais fácil, e graças a disseminação do acesso a dispositivos eletrônicos estas soluções se tornaram cada vez mais escaláveis.

Este aumento de escalabilidade vem fazendo com que culturas venham sendo criadas e as existentes venham sofrendo modificações, podemos citar como exemplo o UBER e o AIRBNB que revolucionaram o

mercado de transporte de curta distância e o de hospedagem.

Espera-se que com a implantação e utilização da ferramenta seja possível criar uma cultura de comunicação ágil, e de grande escala dentro do ambiente acadêmico que hoje é ineficiente nesse sentido.

Pretende-se ainda executar testes e validações no protótipo atual e colher novos feedbacks, principalmente dos docentes que possuem dificuldades com a manipulação de tecnologia, e assim executar novos ciclos de prototipação, e com isso a ferramenta se torne cada vez mais inclusiva.

Como trabalhos futuros é desejado que novos ciclos de evolução sejam aplicados ao software, com foco principalmente em usabilidade uma vez que o software possui grande dependência de funcionamento por parte dos docentes, que ainda possuem dificuldades na manipulação da tecnologia.

## Referências

[1] Pressman, R. Engenharia de Software. 7ª edição. Editora Bookman

[2] Introdução aos Processos de Software e o Modelo Incremental e Evolucionário. DevMedia. Disponível em: <<https://www.devmedia.com.br/introducao-aos-processos-de-software-e-o-modelo-incremental-e-evolucionario/29839>>. Acesso em: 22 Abr. 2019

[3] UPE EM NÚMEROS. UPE. Disponível em: <[http://www.upe.br/anexos/upeemnumeros/2019/UPE\\_em\\_numeros\\_2019.pdf](http://www.upe.br/anexos/upeemnumeros/2019/UPE_em_numeros_2019.pdf)>. Acesso em: 22 Abr. 2019

[4] UPE EM NÚMEROS. UPE. Disponível em: <[http://www.upe.br/images/industrix/arquivos/institucional/documentos/upe-em-numeros/UPE\\_EM\\_NUMEROS\\_2004\\_2005.pdf](http://www.upe.br/images/industrix/arquivos/institucional/documentos/upe-em-numeros/UPE_EM_NUMEROS_2004_2005.pdf)>. Acesso em: 22 Abr. 2019

[5] Paula Nadal. HELOISA LUCK FALA SOBRE OS DESAFIOS DA LIDERANÇA NAS ESCOLAS. GestaoEscolar. Disponível em: <<https://gestaoescolar.org.br/conteudo/787/heloisa-luck-fala-sobre-os-desafios-da-lideranca-nas-escolas>>. Acesso em: 22 Abr. 2019

[6] MEC ASSINA CONVÊNIOS PARA EXPANDIR UNIVERSIDADES. MEC. Disponível em: <[http://portal.mec.gov.br/index.php?option=com\\_content&view=article&id=5236&catid=212](http://portal.mec.gov.br/index.php?option=com_content&view=article&id=5236&catid=212)>. Acesso em: 22 Abr. 2019

[7] Disponível em: <<https://www1.folha.uol.com.br/educacao/2018/09/ensino-superior-volta-a-crescer-no-pais-mas-so-na-modalidade-a-distancia.shtml>>. Acesso em: 22 Abr. 2019