Faculdade Estácio Curitiba

**Aplicativo Telegram COOrdBOT**



MSc. Douglas Rocha Mendes, Coordenador do curso de Análise e Desenvolvimetno de Sistemas na Unidade Estácio Curitiba

MSc. Arivonil Matoski Junior, Gestor Acadêmico na Unidade Estácio Curitiba

Automação do esclarecimento de dúvidas referents as questões institucionais e divulgação de vagas de emprego para os alunos de ADS da Unidade Estácio Curitiba. Automação implementada na plataforma do aplicativo Telegram disponível para os celulares Android e IOS/Apple

Este trabalho consiste de um relatório técnico. implementado para agilizar e aperfeiçoar e agilizar a interação entre Coordenação, CSC, Secretaria e alunos durante este período de pandemia, como também se aplicou para ser utilizado como exemplo prático durante as aulas de programação, modelagem de sistemas, gerencia de configuração e banco de dados.

Curiiba, PR.

Setembro/2020

Sumário

[1 Introdução 4](#_Toc51003900)

[2 Justificativa 4](#_Toc51003901)

[3 Objetivo 4](#_Toc51003902)

[4 Metodologia 5](#_Toc51003903)

[5 Histórico da Empresa / Projeto 5](#_Toc51003904)

[6 Fundamentação Teórica 5](#_Toc51003905)

[6.1 Orientada a objetos 6](#_Toc51003906)

[6.1.1 Comparação entre o paradigma da programação estruturada e paradigma da orientação a objetos 6](#_Toc51003907)

[6.2 Disciplina de Banco de Dados 7](#_Toc51003908)

[6.2.1 MER – Modelo de Entidade e Relacionamentos 8](#_Toc51003909)

[6.2.2 Normalização em banco de dados 8](#_Toc51003910)

[6.3 Disciplina de Computação em Nuvem 9](#_Toc51003911)

[6.3.1 Plataform as a Service - PaaS 10](#_Toc51003912)

[6.3.2 Gerência de Configuração 11](#_Toc51003913)

[6.4 Disciplina de Programação 2 13](#_Toc51003914)

[6.4.1 JDBC 13](#_Toc51003915)

[6.4.2 API Telegram 13](#_Toc51003916)

[6.5 Disciplina de Gerência de Projetos 14](#_Toc51003917)

[6.5.1 Modelo Ágil 14](#_Toc51003918)

[6.5.2 Framework Scrum 14](#_Toc51003919)

[7 Apresentação dos Resultados 14](#_Toc51003920)

[7.1 Menu Academico 14](#_Toc51003921)

[7.2 Menu Secretaria 15](#_Toc51003922)

[7.3 Menu Financeiro 15](#_Toc51003923)

[7.4 Dicas do Boleto 16](#_Toc51003924)

[7.5 Dicas da Grade Curricular 16](#_Toc51003925)

[7.6 Biblioteca 17](#_Toc51003926)

[7.7 Contato da Coordenação de ADS 17](#_Toc51003927)

[7.8 Upload do Book Estácio e datas importantes 17](#_Toc51003928)

[7.9 Vagas de emprego 17](#_Toc51003929)

[7.10 Próximas ações 18](#_Toc51003930)

[8 Instalar o BOT 19](#_Toc51003931)

[8.1 Suporte ao COOrdBOT 19](#_Toc51003932)

[8.2 Acesso via WEB 19](#_Toc51003933)

[9 Considerações Finais 20](#_Toc51003934)

[10 Referências 20](#_Toc51003935)

[11 Cronograma do Projeto 22](#_Toc51003936)

# Introdução

* Este trabalho representa a automatição do atendimento do Coordenador de Curso no que se refere a dúvidas dos alunos sobre a utilização dos requerimentos do Sistema SIA, como também, atua como um concentrador dos FAQs (Frequently Asked and Questions) apresentados aos alunos para esclarecimento de dúvidas quanto a datas, meios de acesso a aulas e gravaçãoes de aulas do ensino presencial/digital e EAD. Esta aplicativo também se propõe a faciltar informações sobre formação e ajustes na grade,biblioteca e como contactar as áreas acadêmica, financeira e secretaria.

# Justificativa

* A ideia de implementar o aplicativo COOrdBOT foi para agilizar o esclarecimento de dúvidas dos alunos sem a necessidade de interagir diretamente com o Coordenador,de curso, como também, sem a necessidade de procurar o atendimento presencial, principalmente durante este período que vivemos de isolamento social devido a Pandemia.
* Como segundo ponto citamos a importância de concentrar em um único ponto as informações relacionadas ao dia-a-dia do aluno na unidade que tende a exigir muito esforço da coordenação de curso e das equipes de atendimento presencial ou CSC. Observamos este pico no início de cada semestre, sendo que a dificuldade de obter uma resposta rápida e precisa levava o discente a gerar reclamações e descontentamento.
* Como terceiro ponto citamos também que a aplicação dispõe de muitos recursos tecnologicos que passaram a ser abordados durante as dsiciplinas técnicas, como forma de representar na prática um determinado conceito que antes ficava apenas no campo da teoria.

# Objetivo

* O foco do aplicativo foi oferecer ao aluno um atendimento rapido e preciso sobre todas as suas dúvidas em um modelo 24x7, garantindo ainda que a sua pesquisa seja em um único local. Desta forma, mesmo que a dúvida apareça fora do horário comercial da Unidade o aluno contará com um atendente sorridente e com respostas rápidas e precisas. Um ponto que também levamos em consideração foi que neste caso não existirá o impacto em questões trabalhistas, pois não haverá necessidade de um colaborador atuar fora do seu horário de trabalho.

# Metodologia

* Neste trabalho utilizamos a pesquisa aplicada, pois nosso objetivo foi colocar o aplicativo em prática para agilizar a obtenção da orientação sobre os procedimentos aos discentes da Unidade Estácio Curitiba Com a disponibilização do aplicativo COOrdBOT busamos evitar aglomerações na secretária da Unidade.
* Este trabalho foi realizado com a aplicação da expériência do professor responsável ao longo de 20 anos de desenvolvimento de sistemas. As pesquisas foram ralizadas em sites de Internet especializados na construção de ChatBOTS [30][31] para a plataforma Telegram [2][3][4][5][6][8][19].

# Histórico da Empresa / Projeto

* A Faculdade Estácio Curitiba se propõe a ministrar cursos de graduação, na modalidade bacharelado e tecnológico, licenciatura, pós-graduação, aperfeiçoamento, sequenciais e extensão; a desenvolver pesquisas e programas de extensão, estabelecendo uma filosofia educacional sob a égide da necessária identificação com os problemas que afligem a região de abrangência da IES. Isso conduz à formação de recursos humanos conscientes da realidade socioeconômica do cenário em que certamente irão atuar. Vejamos a trajetória. Em 2001 a instituição foi credenciada junto ao MEC, como Faculdade Pitágoras de Administração Superior de Curitiba. Está credenciada junto ao MEC pela Portaria n°. 1.824 de 15/08/2001, publicada no Diário Oficial da União em 17/08/2001.
* Desde 2001 a entrada de alunos vem crescendo a cada ano. Devido a esta satisfatória entrada de alunos observamos a necessidade de concentrar os procedimentos e contatos da Faculdade para serem consultados pelos alunos em um único ponto digital, como também, oferecer ao aluno que esta consulta ocorra de forma digital através de um aplicativo de celular, garantindo informação precisa sem a necessidade de ir a Unidade presencialmente e ainda tendo a informação em qualquer horário e em qualquer dia da semana.

# Fundamentação Teórica

Neste semestre no modelo AURA temos as disciplinas de Banco de Dados (com foco nos banco de dados relacionais), Computação em Nuvem [32][33][34][35][36] e na grade 118 temos a disciplina de programação 2 que aborda a linguagem Java.

É importante observar que neste trabalho aplicamos também um dos conceitos da disciplina de Paradigmas de linguagens de programação em Python [37][38] (ARA0066) sobre o conceitos de abordagem. Optamos pelo uso da API do Telegram por proporcionar a usabilidade do aplçicativo em celulares de qualquer fabricante sem a necessidade do desenvolvimento de múltiplas versões, gerando economia de tempo e recursos.

## Orientada a objetos

[7] A linguagem Java foi criada seguindo o paradigma da orientação a objetos e, por isso, traz de forma nativa a possibilidade de o programador usar os conceitos de herança, polimorfismo e encapsulamento. O paradigma da orientação a objetos existe desde a década de 70, mas somente após o sucesso da linguagem Java é que o paradigma ganhou credibilidade. O paradigma de orientação a objetos traz um enfoque diferente da programação estruturada, no sentido de adotar formas mais próximas do mecanismo humano para gerenciar a complexidade de um sistema. Nesse paradigma, o mundo real é visto como sendo constituído de objetos autônomos, concorrentes, que interagem entre si, e cada objeto tem seu próprio estado (atributos) e comportamento (métodos), semelhante a seu correspondente no mundo real.

Quando desenvolvemos programas orientados a objetos e estruturados temos dois paradigmas totalmente diferentes. A forma de pensar e escrever o código são diferentes. É importante observar que muitos programadores usam a linguagem Java, mas continuam pensando no formato estruturado. Essa má prática de programação é muito freqüente em estudantes e profissionais que utilizam Java diariamente. Este livro objetiva minimizar esta má prática do uso da linguagem Java. A seguir é apresentada uma comparação entre o desenvolvimento de um sistema bancário usando o paradigma da programação estruturada e o paradigma da programação orientada a objetos.

### Comparação entre o paradigma da programação estruturada e paradigma da orientação a objetos

Considere o exemplo de um sistema bancário com os seguintes requisitos funcionais:

* Manter o cliente (envolvendo inserir cliente, remover cliente, selecionar cliente e atualizar cliente).
* Manter conta (envolvendo abrir conta, fechar conta, alterar conta, pesquisar conta e remover conta).
* Movimentar caixa (envolvendo sacar, depositar e transferir).
* Registrar movimento (envolvendo gerar histórico).
* Emitir relatórios contábeis.

No paradigma utilizado pelo modelo estruturado seria necessário criar um programa para manter clientes, um programa para manter conta, um programa para movimentar caixa e um programa para emitir os relatórios. Cada programa deverá ser executado por um programa principal, ou seja, um programa com o método main(). Cada programa e também o programa principal têm um conjunto de variáveis específicas para suas necessidades. Cada variável que precisar ser compartilhada com outros programas deve ser criada como global e, se estivermos usando a linguagem C, esta deveria ser externalizada (definir a variável como extern em outro programa) para outros programas. Um ambiente estruturado nos limita apenas a identificar os programas que serão envolvidos na implementação do sistema.

Neste modelo de programação, a reutilização é pequena, e a redundância é grande. No paradigma procedural, quando existe alguma modificação, muitas vezes pelo alto acoplamento entre os programas, é necessário alterar vários programas e novamente testar todo o sistema. No modelo estruturado ficamos limitados a identificar os programas necessários e, no máximo, a criar funções que possam ser reutilizadas em outros programas. No paradigma da orientação a objetos temos outras possibilidades, tais como o uso de encapsulamento para oferecer segurança à classe, herança que permite a reutilização de código, polimorfismo e padrões de projeto, todos explorados de forma detalhada neste livro.

No caso do paradigma da orientação a objetos, inicialmente precisamos identificar nossas classes com seus atributos e métodos. Como técnica para identificar classes temos a análise dos substantivos e locuções substantivas (dois substantivos juntos), listados pelos requisitos funcionais (Use Cases/Casos de Uso). Assim, para o exemplo apresentado, teríamos as classes Conta, Pessoa, Cliente, Gerente de Relacionamento, Contabilidade, Movimentação e Histórico, entre outras. Percebam que nesta abordagem identificamos os substantivos e locuções substantivas descritos pelos requisitos funcionais.

É importante observar que no paradigma da programação orientada a objetos não nos preocupamos em definir quantos programas seriam necessários e sim quais seriam os substantivos utilizados pelo sistema. Cada substantivo representa um objeto no mundo real e uma classe em nosso sistema. Cada classe seria representada por um novo programa Java. Outra vantagem do paradigma da orientação a objetos é a possibilidade de usar uma ferramenta de modelagem para a geração de código.

## Disciplina de Banco de Dados

Na discipljna de Banco de Dados (ARA0040) aplicamos os prncipais conceiteos relacionados ao Modelo de Entidade e Relacionamento (MER), abordamos ainda as associações entre tabelas de banco de dados relacionais e damos foco na normalização. Todos estes conceitos foram aplicados ao aplicativo COOrdBOT.

### MER – Modelo de Entidade e Relacionamentos

[9] - Em engenharia de software, um modelo entidade relacionamento (MER) é um modelo de dados para descrever os dados (entidades e suas colunas) ou aspectos de informação de um domínio de negócio, de uma maneira abstrata que em última análise se presta a ser implementada em um banco de dados relacional. Os principais componentes dos Modelos Entidade-Relacionamento (MER) são as entidades (tabelas) e suas relações (chaves primárias e estrangeiras) qe determinam as cardinalidades entre as entidades.

Os Modelos Entidade-Relacionamento são abordados e praticados na disciplina de Banco de Dados do segundo período do modelo AURA.

### Normalização em banco de dados

[10] “Para adequar o banco de dados, é necessário avaliar com base em cinco regras, que recebem o nome de formas normais. Essas correspondem a um conjunto de regras de simplificação e adequação de tabelas. Diz-se que a tabela do banco de dados relacional está numa certa forma normal quando satisfaz as condições exigentes.”

O trabalho original de Edgar F. Codd, definiu três dessas formas, porem existem outras formas normais que apesar de aceitas são utilizadas poucas vezes em projetos de software. “Cada forma normal representa uma condição mais forte das que a precedem na lista. Para a maioria dos efeitos práticos, considera-se que as bases de dados estão normalizadas se aderirem à terceira forma normal.”

A primeira atividade do responsável pelo sistema é identificar as entidades a partir dos modelos de caso de uso. Para cada entidade são definidos seus atributos (colunas) sendo que um destes atributos será eleito para atuar como chave primária. A partir desta definição partimos para analisar a relação entre os atributos, sua chave primária e aplicamos a teoria da normalização. Vejamos as 3 principais formas normais utilizadas:

* “Primeira Forma Normal (ou 1FN). Nesta forma os atributos precisam ser atômicos, o que significa que as tabelas não podem ter valores repetidos e nem atributos possuindo mais de um valor. Exemplo: CLIENTE = {ID + ENDEREÇO + TELEFONES}. Porém, uma pessoa poderá ter mais de um número de telefone, sendo assim o atributo "TELEFONES" é multivalorado. Para normalizar, é necessário:
  + Identificar a chave primária e também a coluna que possui dados repetidos (nesse exemplo "TELEFONES") e removê-los;
  + Construir uma outra tabela com o atributo em questão, no caso "TELEFONES". Mas não se esquecendo de fazer uma relação entre as duas tabelas: CLIENTE = {ID + ENDEREÇO} e TELEFONE (nova tabela) = {CLIENTE\_ID (chave estrangeira) + TELEFONE}.”
* Segunda Forma Normal (ou 2FN). Primeiramente, para estar na 2FN é preciso estar também na 1FN. 2FN define que os atributos normais, ou seja, os não chave, devem depender unicamente da chave primária da tabela. Assim como as colunas da tabela que não são dependentes dessa chave devem ser removidas da tabela principal e cria-se uma nova tabela utilizando esses dados. Exemplo: PROFESSOR\_CURSO = {ID\_PROF + ID\_CURSO + SALARIO + DESCRICAO\_CURSO} Como podemos observar, o atributo "DESCRICAO\_CURSO" não depende unicamente da chave primária "ID\_PROF", mas sim somente da chave "ID\_CURSO". Para normalizar, é necessário:
  + Identificar os dados não dependentes da chave primária (nesse exemplo "DESCRICAO\_CURSO") e removê-los;
  + Construir uma nova tabela com os dados em questão: PROFESSOR\_CURSO = {ID\_PROF + ID\_CURSO + SALARIO} e CURSOS (nova tabela) = {ID\_CURSO + DESCRICAO\_CURSO}.
* Terceira Forma Normal (ou 3FN). Assim como para estar na 2FN é preciso estar na 1FN, para estar na 3FN é preciso estar também na 2FN. 3FN define que todos os atributos dessa tabela devem ser funcionalmente independentes uns dos outros, ao mesmo tempo que devem ser dependentes exclusivamente da chave primária da tabela. 3FN foi projetada para melhorar o desempenho de processamento dos banco de dados e minimizar os custos de armazenamento. Exemplo: FUNCIONARIO = {ID + NOME + VALOR\_SALARIO + VALOR\_FGTS}. Como sabemos o valor do FGTS é proporcional ao salário, logo o atributo normal "VALOR\_FGTS" é dependente do também atributo normal "VALOR\_SALARIO". Para normalizar, é necessário:
  + Identificar os dados dependentes de outros (nesse exemplo "VALOR\_FGTS");
  + Removê-los da tabela. Esses atributos poderiam ser definitivamente excluídos -- e deixando para a camada de negócio a responsabilidade pelo seu cálculo -- ou até ser movidos para uma nova tabela e referenciar a principal ("FUNCIONARIO").”

## Disciplina de Computação em Nuvem

Na disciplina de Computação em Nuvem (ARA0078) exploramos a publização do aplicativo na Plataforma https://www.heroku.com/ que oferece serviços gratuítos e pagos para a hospedagem de aplicações implementadas em Tecnologias com a linguagem de programação Java e o servidor de aplicação [21] Tomcat [22][23][24] versão 9. O site https://www.heroku.com/ atua oferece um serviço denominado PaaS [11] (Plataform as a Service). Dentro deste contexo ainda foi explorado os conceitos de Gerencia de Configuração [20], pelo uso das ferramentas GIT [12] e Maven [13].

### Plataform as a Service - PaaS

“PaaS — Platform as a Service — em computação, consiste no serviço propriamente dito de hospedagem e implementação de hardware e software, que é usado para prover aplicações (software como serviço) por meio da Internet. Oferece a possibilidade de substituir o paradigma de aplicações tradicional, no qual programas de computador são instalados na máquina do usuário e ficam assim muito mais restritos às configurações de hardware e software disponíveis nela, principalmente ao sistema operacional.”

Neste projeto utilizamos a plataforma Heroku que oferece suporte a projeto Java com o servidor de aplicação Tomcat versão 9.

#### Servidor de aplicação

Um servidor de aplicação, é um servidor equipado com software especializado que disponibiliza um ambiente para a instalação e execução de aplicativos implementados em linguagens de programação. Em nosso projeto utilizamos a linguagem Java e por isto nosso servidor de aplicação foi o Tomcat.

O objetivo do servidor de aplicação é disponibilizar uma plataforma que separe do desenvolvedor de software algumas das complexidades de um sistema computacional. No desenvolvimento de aplicações comerciais, por exemplo, o foco dos desenvolvedores deve ser a resolução de problemas relacionados ao negócio da empresa, e não em questões de infraestrutura de software. O servidor de aplicações responde a algumas questões comuns a todas as aplicações, como segurança, garantia de disponibilidade,

#### Vantagens da plataforma PaaS

“O modelo de PaaS é considerado benéfico para desenvolvedores, já que é uma boa maneira de o profissional dedicar-se, de fato, ao trabalho de construção de aplicativos, deixando o restante com o fornecedor, como gerenciamento, licenças de uso, atualização e manutenção da infraestrutura, entre outras agruras, com a vantagem da redução de custos da operação. Para projetos em que a equipe é reduzida, o uso de PaaS é apontado como uma boa solução, pois permite o foco em desenvolvimento, sem a necessidade de gerenciar, fazer a manutenção e garantir a segurança do servidor, optando por uma plataforma pronta. Como é característica de soluções de Computação em Nuvem, nas soluções de PaaS é possível ajustar o consumo de recursos de acordo com a demanda, assim como a cobrança também é feita por demanda.”

#### Desvantagens da plataforam PaaS

Como desvantagens citamos a complexidade para criar scripts de deploy, pouca documentação em português, falta de suporte gratuito em português e aumento de preços quando se necesita de uma ambiente comalta escalábilidade.

### Gerência de Configuração

Gerência de configuração de software é uma área da engenharia de software responsável por fornecer o apoio para o desenvolvimento de software. Suas principais atribuições são o controle de versão (Ex.: GITHUB [25][26]), o controle de mudança [28][29] (Ex.: Remedy Change Management [27]) e a auditoria das configurações. Roger Pressman, em seu livro Software Engineering: A Practitioner's Approach, especifica que a gerência de configuração de software (GCS) é o: conjunto de atividades projetadas para controlar as mudanças pela identificação dos produtos do trabalho que serão alterados, estabelecendo um relacionamento entre eles, definindo o mecanismo para o gerenciamento de diferentes versões destes produtos, controlando as mudanças impostas, e auditando e relatando as mudanças realizadas.

#### Controle de Mudanças

O gerenciamento do controle de mudanças é um processo para garantir que todos os procedimentos definidos pela empresa ou orgão regulador sejam seguidos extamente conforme definido. O objetivo é minimizar o impacto de incidentes relacionados a mudanças e garantir que possa ser auditada. Outro ponto importante é que através do controle de mudanças todas as áreas e pessoas envolvidas possam documentar e registrar suas ações de forma que toda a nova versão de um do software possa ser validada no futuro.

#### GITHUB

Sistema de controle de versões distribuído e gratuíto, usado principalmente no desenvolvimento de software, mas pode ser usado para registrar o histórico de edições de qualquer tipo de arquivo tal como word, PDF ou Excel..

O GITHUB posui ferrametnas simples de integração com a plataforma Eclipse utilizada para o desenvolvimento de sistemas na linguagem de progamação Java.

#### Maven

O software Apache Maven, é uma ferramenta de automação de compilação utilizada primariamente em projetos implementados em linguagem Java. Ela é similar à ferramenta Ant, utilizada na década de 2000.

Um projeto Java equipado no modelo da ferrametna Maven utiliza um arquivo XML denominado pom.xml que descreve todas as bibliotecas (arquivos com extensão .jar) utilizadas pelo software, como também, plugins e as caraterísticas definidas ao projeto Java. O Maven realiza o download automático de bibliotecas Java e seus plug-ins a partir de um repositório local (controlado pela empresa) ou global (disponível na Internet).

Ao final da execução de um script equipado com a ferrametna Maven será um arquivo .WAR ou .EAR que serão utilizados no processo de deploy no servidor de aplicação Tomcat.

#### Script de Deploy do COOrdBOT

Para agilizar a publicação de uma nova versão do aplicativo criamos um script através da linguagem de scripts do Microsoft Windows denominado arquivo de lote. O script foi criado para realizar o processo de deploy no servidor de aplicação chamado de Dyno (servidor de aplicação Apache Tomcat) disponibilizado pela plataforma Heroku (PaaS – Plataform as a Service). Vejamos o script na íntegra.

echo \_\_\_\_\_\_

echo Maven

echo \_\_\_\_\_\_

call mvn clean install

call mvn package

echo \_\_\_\_\_\_

echo Login

echo \_\_\_\_\_\_

call heroku login

echo \_\_\_\_\_\_

echo GIT

echo \_\_\_\_\_\_

call git init

call git config --global http.sslVerify false

call heroku git:remote -a webmavenheroku

call git add .

echo \_\_\_\_\_\_

echo GIT Commit

echo \_\_\_\_\_\_

call git commit -am "versao 1.0"

echo \_\_\_\_\_\_

echo GIT Push

echo \_\_\_\_\_\_

call git push heroku master

echo \_\_\_\_\_\_

echo Deploy FINAL

echo \_\_\_\_\_\_

call heroku plugins:install java

call heroku war:deploy target/webmavenheroku-0.0.1-SNAPSHOT.war --remote origin

echo \_\_\_\_\_\_

echo Fim do Deploy

Neste script observamos o uso das feramentas Maven e GIT. A aplicação Maven é a responsável pela compilação do aplicativo e empacotamento em um arquivo no formato .war. O aplicativo GIT é o responsável pelo versionamento do projeto no repositório da plataforma Heroku [1].

## Disciplina de Programação 2

Na disciplina de Programação 2 (CCT0695) foi posível aplicar a implementação de progamas em conjunto com o Framework [15] da API do Telegram. Aplicamos neste projeto o uso da API JDBC (Java DataBase Conectivity) [14], que nospermitiu colocar em prática as teorias analisadas na disciplina de Banco de Dados no modelo AURA. Atráves da linguagem Java aplicamos algoritmos complexo que tiveram a função de controlar o fluxo da usabilidade do BOT, como também, na montagem dinâmicas dos menus superiores e inferiores. Vejamos sobre o comportamento da API JDBC [39] e a API Telegram que disponibiliza um framework para que desenvolvedores d diferentes linguagens de programação possam rapidametne criar aplicações sobre esta tecnologia.

### JDBC

A API Java Database Connectivity ou JDBC é um conjunto de classes e interfaces Java que fazem o envio de instruções SQL [40] para um banco de dados relacional tal como MySQL [41], SQLServer [43], Postgree [42], Oracle, Sybase entre vários outros;

### API Telegram

https://pt.wikipedia.org/wiki/Telegram\_(aplicativo)

O Telegram é um serviço de mensagens instantâneas baseado em Computação em Nuvem e esta disponível disponível para:

* Celulares ou Tablets equipados com Android, iOS/Apple, Windows Phone, Ubuntu Touch, Firefox OS,
* Computadores equipados com sistema operacional Microsoft Windows, OS X ou Linux.

Os desenvolvedores podem equipar seu aplicativos de ChatBOT com chamadas com vídeo, enviar mensagens e enviar e receber fotos, vídeos, autocolantes e arquivos de qualquer tipo.

#### ChatBOTs

Chatbot são programas de computador programados em uma linguagem de programação que simulam um chat operado por uma pessoa humana, porem de forma automatizada como se fosse um robo, por isto o termo ChatBOT.. Dessa forma, é possível automatizar tarefas repetitivas e burocráticas, como dúvidas frequentes, na forma de diálogo pré-definido entre o usuário e um robô.

#### Conceito de Framework

Um framework representa um conjunto de programas semi prontos que ao serem estendidos facilitam e alavancam a produtividade no desenvolvimento de um software.

Um Framework não se trata de um software pronto para ser utilizado, mas sim de um modelo de dados que ao serem estendidos geram valor a um determinado ramos da economia.

## Disciplina de Gerência de Projetos

A disciplina de gerencimaneto de projetos trouxe ao projeto os conceitos atuais sobre os modelos ágeis e o framework Scrum.

### Modelo Ágil

[16][17] O Modelo Ágil de Software é uma disciplina que estuda um conjunto de comportamentos, processos, práticas e ferramentas utilizados para a criação de produtos e sua subsequente disponibilização para os usuários finais. Estes produtos podem envolver software ou não. O modelo

### Framework Scrum

[18] O Scrum é um framework de gerenciamento de projetos, da organização ao desenvolvimento ágil de produtos complexos e adaptativos com o mais alto valor possível, através de várias técnicas, utilizado desde o início de 1990 e que atualmente é utilizado em mais de 60% dos projetos ágeis em todo o mundo.

# Apresentação dos Resultados

Observamos que no início de cada semestre recebemos dezenas de questionamentos que se repetem a cada ciclo. Com base nas recorrentes respostas percebemos a possibilidade de concentrar e unificar o ponto das consultas em uma aplicação digital. Desta forma reunimos as funcionalidades do aplicativo nas seguintes opções que compõe o menu principal: Todas as funcionalidades principais se dividem em uma ou mais funcionalidades secundárias.

## Menu Academico



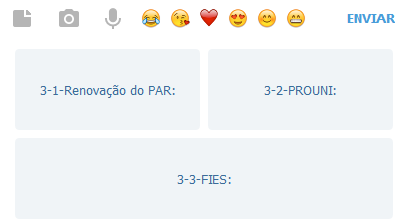
Dicas sobre os procedimentos Acadêmicos, contatos, dicas para o estágio profissional, como configurar senha e utilizar os aplicativos Windows dos laboratórios da Estácio na Nuvem através da ferramenta Citrix via VDI (Virtual Desktop Infrastructure). Neste item principal apresentamos como são distribuídas e dicas sobre as Atividades Acadêmicas Complementares das grades 118 e AURA. Nesta opção os alunos poderão receber os procedimentos sobre como interagir com o ambiente Microsoft Teams em formato PDF através de upload de um arquivo que o BOT realizará ao celular do aluno.

## Menu Secretaria

Contato da Secretaria. Apresentamos os telefones e e-mail para contato. . Poderemos inserir novas funcionalidades em um nova versão.

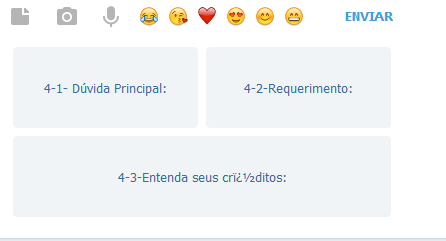
## Menu Financeiro

* Dicas sobre como abrir requerimentos para que alunos fiquem atualizados e tenham suas dúvidas esclarecidas através do CSC. Neste item do menu principal o aluno poderá interagir sobre o programa PAR, PROUNI e FIES.



## Dicas do Boleto

* Dicas sobre o boleto financeiro, inclusive o motivo de valores divergentes do esperado. O aluno poderá ainda avaliar o procedimento para abertura de requerimento ao CSC.



## Dicas da Grade Curricular



Nesta opção o aluno saberá a definição das disciplinas eletivas, poderá avaliar os casos onde a sua grade esteja sem opções para incluir disciplinas, poderá contactar os canais corretos sobre o tratamento da colação de grau, direcionar as dúvidas sobre a documentação da formatura, obter os links para visualizar a grade e ainda obter o link para ajustar a grade deste semestre.

## Biblioteca

Neste item informamos que a Biblioteca encontra-se fechada devido ao período de isolamento social. Poderemos inserir novas funcionalidades em um nova versão.

## Contato da Coordenação de ADS

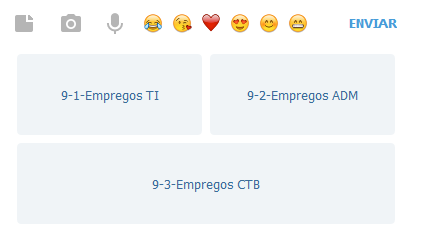
Apresentamos o telefone e e-mail do coordenador. Poderemos inserir novas funcionalidades em um nova versão.

## Upload do Book Estácio e datas importantes

* Neste item o ChatBOT realiza o upload do Book Estácio e ainda apresenta um resumo das datas das provas presenciais e EAD.

## Vagas de emprego

Através desta funcionalidade o BOT realizará o upload de um arquivo PDF ao celular do aluno com dezenas de vagas de emprego nas áreas de TI em Curitiba e região. O BOT ainda apresenta as opções de vagas de emprego aos curso de a Administração e Contabilidade.



Outro ponto que contribui para a implementação do COOrdBOT foi que por mais que façamos o envio dos procedimentos via e-mail, por muitas vezes estes não são lidos, por diferentes motivos, levando o aluno a buscar respostas juto a coordenação ou secretaria. Mesmo com o uso da ferramenta de mensagens Whatsapp não conseguimos alcançar todos os alunos, pois muitos por preferência não participam dos grupos das redes sociais. Outro ponto avaliado foi que mesmo os alunos que estão nos grupos muitos acabam não acompanhando nossas comunicações pelo envio de dezenas de mensagens que rapidamente mudam de assunto e fazem com que informações importantes fiquem distantes das leituras esparsas.

Observamos ainda que a ferramenta poderia ser utilizada para exemplificação de situações reais relacionadas as disciplinas deste semestre, pois muitas vezes utilizamos exemplos com foco acadêmico ou apenas teórico que os distanciam do mundo corporativo. Vejamos os itens positivos avaliados em um mês da utilização do aplicativo:

* Satisfação do aluno em ter a informação centralizada, resumida e disponível 24x7.
* Uso dos recursos aplicados ao aplicativo para as aulas práticas das disciplinas das grades AURA e 118.
* Redução das consultas aos canais de atendimento da Unidade Estácio Curitiba.
* Redução da ida do aluno a Unidade para obtenção de respostas sobre procedimentos simples.

## Próximas ações

Esta planejado para que durante este semestre seja implementado novas funcionalidades relacionadas a gestão do uso do aplicativo com relatórios voltados para análise quantitativa em relação ao uso do aplicativo. Planejamos ainda integrar com o NAC a capacidade da inserção automatizada de vagas de empregos aos cursos da Unidade Estácio Curitiba para todos os cursos em andamento.

Esta planejado a realização de uma pesquisa aos alunos da unidade sobre a qualidade do aplicativo e sobre as sugestões de melhorias e novas funcionalidades.

# Instalar o BOT

Para acessar o aplicativo o aluno deverá baixar e instalar o aplicativo Telegram de sua loja (Google Play ou Apple Store) vinculada ao seu celular. Em seguida deverá clicar no link: https://t.me/cscpr\_bot, e escolher a opção /start que acessará o COOrdBOT.

## Suporte ao COOrdBOT

Caso o BOT não responda fique tranquilo. O BOT pode estar dormindo. :)). Para acordar o Servidor de Aplicação onde o aplicativo esta instalado digitar ou colar esta URL no Browser: <https://webmavenheroku.herokuapp.com/> Ao ser apresentada a tela acessar voltar ao aplicativo do BOT que responderá normalmente.

Para acompanhar a utilização do sistema poderá acessar a URL: <https://webmavenheroku.herokuapp.com/logfaculdade.jsf>. Nesta URL clicar no botão Consultar todas as LOGs para visualizar sobre as consultas já realizadas pelos alunos.

## Acesso via WEB

Como ocorre com o aplicativo Whatsapp o Telegram também pode ser utilizado pela Web a fim de facilitar a interação com o envio e recebimento de mensagens. Para esta facilidade acessar o BOT pela URL: <https://web.telegram.org/#/im?p=@cscpr_bot>. É importante observar que para utilizar o BOT pela Internet o usuário precisa ter o APP do Telegram instalado no celular e configurado para o uso. Recomendamos acessar o BOT via celular para então tentar seu acesso via Web.

# Considerações Finais

Como resultados do trabalho observamos o interesse dos alunos pela aplicação, como também, redução de questionamentos a coordenação e da abertura de requerimentos que muitas vezes posuem período de resposta além da necessidade dos alunos. O aplicativo estão disponível aos alunos a 1 mês, sendo que neste período observamos os seguintes pontos positivos:

* O aluno percebeu que é posível aplicar a ideia do BOT em suas empresas, a fim de alavancar os negócios e ainda alavancar sua imagem e carreira.
* Facilidade para acessar esclarecer suas dúvidas em qualquer horário, sem a necessidade do contato via 0800 ou presencial.
* Correlação entre as técnicas utilizadas no desenvolvimento do BOT durante as discplinas de Paradigmas em Programação em Python, Programação 2 (Java) e Computação em Nuvem.
* Aplicação integrada dos conceitos vistos em disciplinas do semestre integradas em um projeto no mesmo formato dos projetos implementados em um ambiente corporativo.
* Aplicação dos paradigmas de programação, aplicação de frameworks complexos, uso do atendimento digital e contato com tecnologias de mercado.

# Referências

* [1] - https://devcenter.heroku.com/articles/deploying-java
* [2] - https://core.telegram.org/, 05/2020
* [3] - https://core.telegram.org/bots, 05/2020
* [4] - https://core.telegram.org/bots/samples, 05/2020
* [5] - https://core.telegram.org/bots/api, 05/2020
* [6] - <https://www.codota.com/code/java/methods/com.pengrad.telegrambot.TelegramBot>/execute
* [7] - MENDES, Douglas. Programação Java com Ênfase em Orientação a Objetos. São Paulo, Editora Novatec, 2009.
* [8] - <https://medium.com/@michelpf/criando-bots-do-telegram-com-api-java-64e921762b5f>, 05/2020
* [9] - <https://pt.wikipedia.org/wiki/Modelo_entidade_relacionamento>
* [10] - https://pt.wikipedia.org/wiki/Normaliza%C3%A7%C3%A3o\_de\_dados
* [11] - https://pt.wikipedia.org/wiki/Plataforma\_como\_servi%C3%A7o
* [12] - <https://pt.wikipedia.org/wiki/Git>
* [13] - <https://pt.wikipedia.org/wiki/Apache_Maven>
* [14] - <https://pt.wikipedia.org/wiki/JDBC>
* [15] https://pt.wikipedia.org/wiki/Framework
* [16] - https://pt.wikipedia.org/wiki/Manifesto\_%C3%81gil
* [17] - https://pt.wikipedia.org/wiki/Desenvolvimento\_%C3%A1gil\_de\_software
* [18] - <https://pt.wikipedia.org/wiki/Scrum_(desenvolvimento_de_software)>
* [19] - <https://pt.wikipedia.org/wiki/Telegram_(aplicativo)>
* [20] - <https://pt.wikipedia.org/wiki/Ger%C3%AAncia_de_configura%C3%A7%C3%A3o_de_software>
* [21] - <https://pt.wikipedia.org/wiki/Servidor_de_aplica%C3%A7%C3%A3o>
* [22] - <https://www.google.com/search?client=firefox-b-e&q=tomcat>
* [23] - <https://pt.wikipedia.org/wiki/Apache_Tomcat>
* [24] - <https://www.devmedia.com.br/introduzindo-o-servidor-de-aplicacao-apache-tomcat/27939>
* [25] - <https://github.com/>
* [26] - <https://pt.wikipedia.org/wiki/GitHub>
* [27] - <https://www.bmcsoftware.pt/education/courses/remedy-change-management-8-0-using.html?301-bmcsoftware-com-br>
* [28] - https://pt.wikipedia.org/wiki/Change\_Management
* [29] - <https://pt.wikipedia.org/wiki/Gest%C3%A3o_de_mudan%C3%A7a_(engenharia)>
* [30] - <https://pt.wikipedia.org/wiki/Chatterbot>
* [31] - <https://en.wikipedia.org/wiki/Chatbot>
* [32] - https://pt.wikipedia.org/wiki/Computa%C3%A7%C3%A3o\_em\_nuvem
* [33] - NETO, Manuel V de S. Computação Em Nuvem - Nova Arquitetura de TI [BV:PE]. 1Ed. Rio de Janeiro: Brasport, 2015. Disponível em: https://plataforma.bvirtual.com.br/Acervo/Publicacao/160695
* [34] - TANEMBAUM, A.; STEEN, M. Sistemas Distribuídos: Princípios e Paradigmas [BV:PE]. 2 ed.. São Paulo: Pearson Prentice Hall, 2007. Disponível em: https://plataforma.bvirtual.com.br/Acervo/Publicacao/411
* [35] - SOMMERVILLE, Ian. Engenharia de Software [BV:PE]. 10ª Ed.. São Paulo: Pearson Prentice Hall, 2011. Disponível em: https://plataforma.bvirtual.com.br/Leitor/Loader/168127/pdf
* [36] - KUROSE, J. F; ROSSA, Keith W. Redes de Computadores e a Internet: uma abordagem top-down [BV:PE]. 6. ed. São Paulo: Pearson, 2013. Disponível em: <https://plataforma.bvirtual.com.br/Leitor/Loader/3843/pdf>
* [37] - SEBESTA, Robert W. Conceitos de Linguagens de Programação. 11ª Ed. Porto Alegre: Bookkman, 2018. Disponível em: https://integrada.minhabiblioteca.com.br/#/books/9788582604694/
* [38] - <https://python.org.br/introducao/>
* [39] - MENDES, Douglas. Programação Java em Ambiente Distribuído. São Paulo, Editora Novatec, 2011.
* [40] - <https://en.wikipedia.org/wiki/SQL-92>
* [41] - <https://www.mysql.com/>
* [42] - <https://www.postgresql.org/>
* [43] - https://www.devmedia.com.br/primeiros-passos-no-sql-server/40653

# Cronograma do Projeto

|  |  |
| --- | --- |
| Planejamento | 03/2020 |
| Estudo das tecnologias envolvidas | 04/2020 e 05/2020 |
| Implementação do aplicativo | 06/2020 até 09/2020 |