

FACULDADE DE INFORMÁTICA E ADMINISTRAÇÃO PAULISTA

DIGITAL BUSINESS ENABLEMENT  
3º ENTREGA CHALLENGE

Grupo: TAGE Integrantes:

Arthur Henrique Toshiriro Sakamoto Santos - rm86832

Gustavo de Cássio Vassellai - rm88264

Izadora Amorim Santana - rm88494

João Armando Alves Neto - rm88255

Nathan Santana da Silva - rm88517

## Link do Repositório no GitHub.

Link: <https://github.com/ArthurSakamoto/challengeDigital.git>

### Escopo de projeto – TAGE

#### Justificativa

Com Base no desafio proposto pela Plusoft e CPDQD, procuramos decidir em qual área dos negócios iríamos desenvolver nosso projeto. Após entrarmos em um consenso decidimos trabalhar na área da educação. Tendo isso definido fizemos uma pesquisa sobre o ramo da educação e descobrimos que um estudo realizado pela Secretaria da Educação do Estado de São Paulo, com 642 mil alunos do 5º e 9º ano do ensino fundamental e da 3ª série do ensino médio. Onde um em cada três alunos relatou dificuldade de concentração em nível alto ou moderado, Além de mostrarem defasagem de educação de pelo menos 4 anos.

#### Finalidade

O objetivo desse desafio consiste em apresentar um uso da Inteligência artificial com interfaces de voz como diferencial no relacionamento entre pessoas nos canais digitais.

#### Objetivos

Desenvolver uma ferramenta capaz de auxiliar nos estudos desses estudantes e assim tentar diminuir essa defasagem na educação.

Para isso desenvolvemos uma IA capaz de apontar quais as principais dificuldades dos alunos em determinada matéria e encaminhá-lo materiais de apoio da própria bibliografia do usada pelo professor para sanar as suas dúvidas. E para descobrir as dúvidas dos alunos nossa IA retira esses dados através das respostas dada pelos alunos nas tarefas atribuídas. E retornar para o professor quais são as maiores dificuldades dos alunos. Junto também desenvolvemos uma IA com voz capaz de auxiliá-los na navegação da ferramenta.

#### Descrição do Produto

Nossa ferramenta será disponível por assinatura, onde após as entidades escolares terem contratado nossos serviços eles devem primeiramente cadastrar fazer o cadastro tanto dos professores quanto dos alunos. Uma vez feito isso ambos poderão utilizar da nossa ferramenta e de suas funcionalidades ao realizar o login. Quando um professor realiza o login em nossa ferramenta ele poderá postar para os seus alunos materiais didáticos e gerar uma lista de exercícios relacionada a matéria que leciona, ele pode tanto usar exercícios próprios ou usar exercícios registrado em nosso banco de dados. Uma vez postado os exercícios o professor poderá receber quais alunos não fizeram e uma análise feita por um bot quais são as maiores dificuldades dos alunos e o seu progresso. Agora quando um aluno realiza o login ele poderá ver quais disciplinas ele está cursando, uma vez que ele entre em uma das disciplinas ele poderá ter acesso a todos os materiais didáticos disponibilizados pelo professor da disciplina e as listas exercícios do professor. O aluno também poderá ver quais listas de exercícios ele já fez e quais estão com a entrega pendente. Uma vez que o aluno fez a lista de exercícios ele poderá ver quais questões ele acertou e errou, onde através das respostas erradas o bot poderá descobrir quais são as maiores dificuldades do aluno e então encaminhar materiais de apoio para sanar suas dúvidas, esses materiais de apoio podem ser tanto dos já postado pelo professor quanto de terceiros. Existe também uma interface de voz onde tanto o aluno quanto o professor podem usar para se locomover pela ferramenta.

## Equipes envolvidas

Cliente do projeto – Plusoft e CPDQD

Supervisores do projeto – Henrique Ferreira e Thiago Yamamoto

Supervisor de Compliance & Quality Assurance – Renato Jardim Parducci

Supervisor de Database Application & Data Science – Marcel Thomé Filho

Supervisor de DevOps Tools & Cloud Computing – João Carlos Menk

Supervisor de Digital Bussiness Enablement – João Carlos Lima e Silva

Supervisor de Disruptive Architectures: IT, IoB & IA – Arnaldo Alves Viana Júnior

Supervisor de Enterprise Application Development – Elias Romero Belinello

Supervisor de Hybrid Mobile App Development – Alexandre dos Santos Mignon

Equipe do projeto – Arthur Henrique, Gustavo de Cássio, João Armando, Izadora Amorim e Nathan Santana

## Critérios de aceitação

Compliance & Quality Assurance

- Desenvolva o plano de release do seu projeto com Scrum, usando AZURE para planejar o escopo a ser desenvolvido na sua solução.

Database Application & Data Science

- Criar um **procedimento** e uma **função** no Oracle Database. Cada um dos grupos deve definir a funcionalidade de cada um dos programas. A funcionalidade deve ser pertinente ao projeto desenvolvido pelo grupo.
- Um **procedimento** que faça INSERT de dados em uma entidade **associativa** do projeto desenvolvido pelo grupo. Não use hard code, o procedimento deve ser feito pensando que o usuário o usará para inserir novos dados na tabela em momento futuro.
- Deverá ser feito, em cada um dos programas, o tratamento de exceção WHEN OTHERS e, pelo menos, mais duas exceções o grupo julgar pertinente.

DevOps Tools & Cloud Computing

- Utilizando as tecnologias que aprendemos nos Deploys da Azure, desenvolva uma aplicação Web ou API, ou utilize a criada na disciplina de DBE, que realize um CRUD (Inclusão, Alteração, Eliminação e Consulta) sobre uma tabela simples de seu projeto da Plusoft ou outra tabela caso você não tenha definido em sua solução persistência de dados. Utilize um típico SGBDR de sua preferência. Pode ser H2, Oracle da FIAP, SQL da Azure etc.
- Após concluir a etapa anterior, faça o deploy como Web App ( Serviços de Aplicativos) na plataforma Azure. Aqui, se o grupo desejar, poderá ser feito também como alternativa um Deploy no Azure Functions.
- Inclua e movimente no mínimo 6 linhas com conteúdo significativo (inclua essas ações em sua documentação).

Digital Bussiness Enablement

Documento no formato pdf ou readme.md contendo:

- Objetivo e escopo do projeto: descrever a solução proposta pelo grupo de forma textual.
- Arquitetura da solução atualizada;
- Tabela dos *endpoints*.
- Breve descrição das funcionalidades do *backend*.

Backend contendo:

- Implementação de uma API RESTful com principais funcionalidades da solução proposta. Pode ser implementada em uma tecnologia abordada no curso de TDS (Java, C#).

Disruptive Architectures: IT, IoB & IA

- Entregar a solução de voz/chatbot **integrado** a proposta de solução final do projeto (website, aplicativo, software...).
- Fluxo de conversacional voz/chatbot **completo** (apresentar fluxograma) entrega esperada: documentação PDF ou link do canva / visio / mural / Draw.io....
- Demonstração da **implementação completa** do fluxo conversacional de acordo com o fluxograma (vídeo da solução). entrega esperada: link do vídeo no youtube.

Enterprise Application Development

- Arquitetura da solução atualizada.
- Implementação do **backend** de uma funcionalidade principal da aplicação, pode ser implementada em uma tecnologia abordada no curso de TDS (Java, C# ou mobile).
- Descrição da funcionalidade principal que foi implementada.

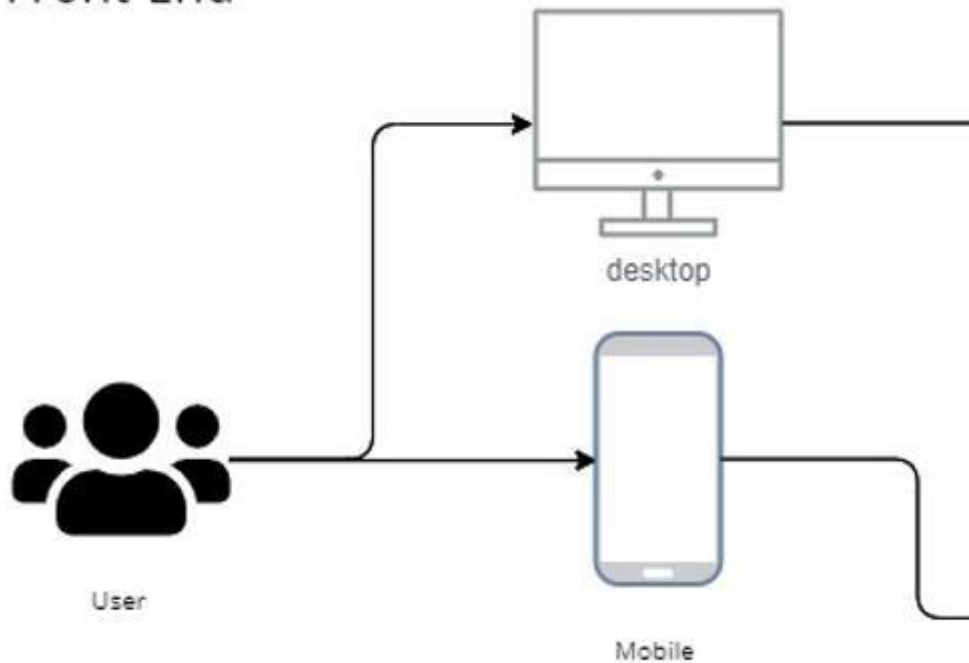
Hybrid Mobile App Development

- Escolher uma das funcionalidades que o aplicativo possui e implementar um CRUD completo.

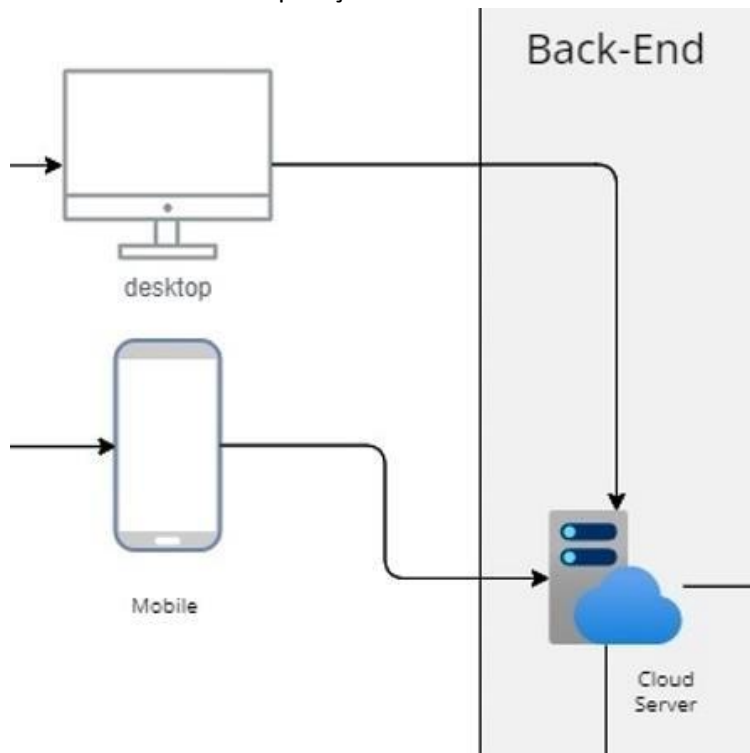
## Arquitetura do projeto

Primeiramente o Usuário irá acessar a nossa aplicação através de um aparelho Mobile ou Desktop através de um login ou um cadastro.

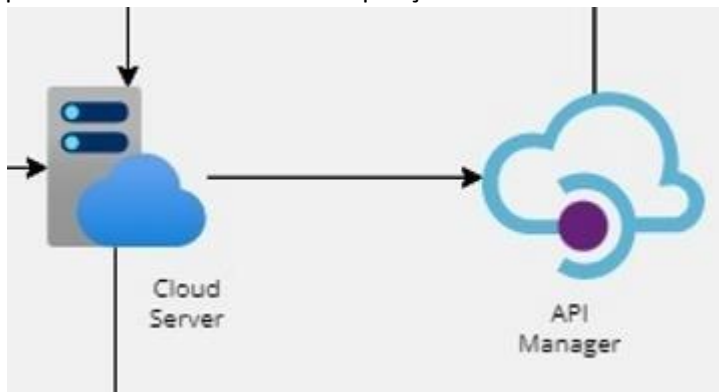
### Front-End



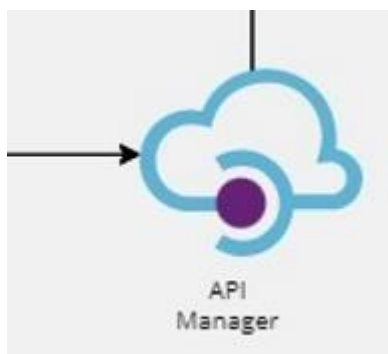
Depois através de um Json os Aparelhos Mobile e Desktop irão se comunicar com um servidor na nuvem onde a nossa aplicação está localizada.



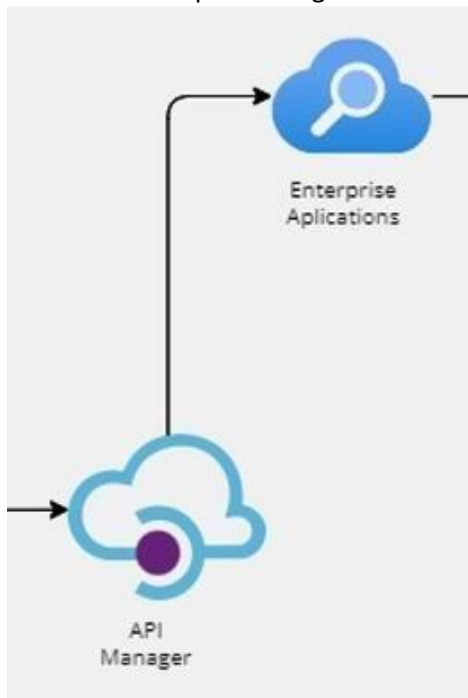
Uma vez que nossa aplicação na nuvem tenha recebido o Json ela executara duas tarefas a primeira é se fazendo uma requisição com o nosso API Manager.



O API Manager por sua vez tem a função de lidar com análises de cada requisição, Controle do ciclo de vida das APIs e versionamento das APIs.



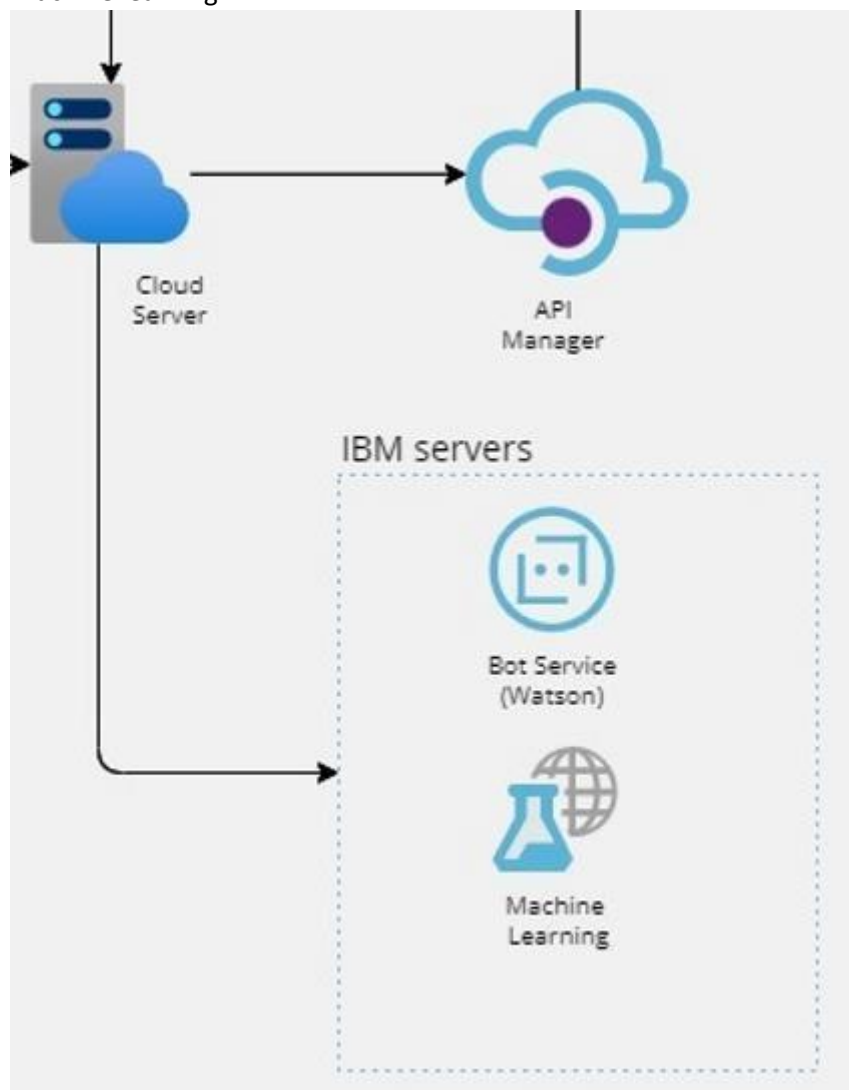
Uma vez que a requisição foi aceita o nosso API Manager irá se comunicar através de um Json com nossa API que no diagrama chamamos de Enterprise Applications.



Ele por sua vez irá se comunicar com nosso Banco de Dados onde ele irá verificar se os dados informados pelos usuários estão corretos ao comparar os dados informados com os registrados em nosso banco.



Uma vez que os dados estejam corretos o usuário poderá utilizar a nossa aplicação, e então nossa aplicação na nuvem irá realizar sua segunda tarefa. Através de uma conexão no Node-RED ele irá se comunicar com os servidores da IBM e chamará o nosso chatbot e a nossa IA de machine learning.



Nosso chatbot por sua vez auxiliará o usuário a navegar pela aplicação, através de comandos de voz.





Já nossa IA ela terá a função de prever quais são as maiores dificuldades dos alunos, através de exercícios e na pesquisa feita nos materiais didáticos em nossa aplicação.



## Tabela de endpoints

Listar Alunos

GET localhost:8080/api/aluno?page=0&size=5

Cadastrar Alunos

POST localhost:8080/api/aluno

Apagar Alunos

DELETE localhost:8080/api/aluno/1

Alterar Alunos

PUT localhost:8080/api/aluno/2

Listar Professor

GET localhost:8080/api/professor

Cadastrar Professor

POST localhost:8080/api/professor

Apagar Professor

DELETE localhost:8080/api/professor/4

Alterar Professor

PUT localhost:8080/api/professor/5

## Descrição da funcionalidade Backend.

Para a entrega da sprint 3 desenvolvemos para teste uma API Rest capaz de cadastrar Alunos e Professores. Foi feita essa escolha para poder criar uma classe com 7 professores e 40 alunos e assim nesse ambiente também podemos separar os papéis de cada usuário (ROLE) e verificar se funcionam como devido. Onde o aluno com o Role de user poderá apenas visualizar as suas informações tanto quanto, tarefas aplicadas e os materiais didáticos disponibilizados pelo professor para sua classe.

Já o próprio professor irá poder usar com o Role para poder publicar uma tarefa para a classe em que está lecionando, ele também poderá cadastrar novas tarefas, tanto quanto atualizá-las ou removê-las. Para essa primeira entrega, decidimos não implementar as tabelas de classe nem de tarefas na API por conta do tempo e de nosso conhecimento então optamos por essas duas tabelas para assim podemos ter o mínimo para testar a funcionalidade de nossa ferramenta, para isso adicionamos um atributo no professor chamado tarefa assim ele poderá simular nessa entrega essa Tabela, também não conseguimos implementar com sucesso o spring security por conta de um atraso nas aulas por causa disso não conseguimos aplicar os Roles corretamente aos usuários então nessa entrega tecnicamente o professor é o administrador que tem o direito de atualizar e remover um aluno.