



FIAP GRADUAÇÃO

GLOBAL SOLUTION 2025

2º ANO

ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

Turmas de agosto

DESAFIO FIAP: EVENTOS EXTREMOS

A natureza está cheia de surpresas. Ventos poderosos, chuvas intensas, tremores inesperados, calor de derreter ideias e frio de congelar pensamentos. Os eventos extremos estão aí e fazem parte do nosso novo cenário global. Mas em vez de temê-los, que tal enxergar tudo isso como uma grande oportunidade de criar algo incrível?

Na FIAP, acreditamos que cada desafio é uma chance de inovar, aprender e transformar. E quando a tecnologia se junta à criatividade, surgem soluções que fazem a diferença de verdade.

DESAFIO FIAP: EVENTOS EXTREMOS

Agora é a sua vez. Imagine um cenário impactado por um evento extremo da natureza. Seu papel é propor uma solução que use tecnologia, inovação e boas ideias para ajudar pessoas, proteger o meio ambiente ou prevenir problemas ainda maiores.

Vale usar tudo o que você sabe e explorar o que ainda está aprendendo. Crie protótipos, pesquise, conecte dados, crie narrativas e pense fora da caixa. O que você criaria para enfrentar um super calor? Ou uma cidade isolada por um terremoto? Como a tecnologia pode ser usada de forma prática e surpreendente?

Você está no comando. Solte a criatividade e mostre como grandes ideias nascem em momentos de grandes desafios.

DESAFIO FIAP: EVENTOS EXTREMOS

A inovação pode surgir no design de soluções acessíveis, no uso de sensores conectados à internet das coisas, na análise de grandes volumes de dados em tempo real ou na criação de plataformas que integram pessoas, serviços e recursos de forma ágil e eficaz.

Com criatividade, é possível simular cenários, treinar comunidades por meio de experiências imersivas, construir modelos de negócio sustentáveis e fortalecer a resiliência das cidades e das pessoas. Áreas como tecnologia, design, engenharia e gestão se conectam para enfrentar desafios reais e criar respostas que protegem, aproximam e salvam vidas.

Neste desafio, o mais importante não é apenas o que você sabe, mas o que você é capaz de imaginar e colocar em prática. Afinal, quando ideias ganham propósito, elas têm o poder de mudar o mundo.



VEJA ALGUMAS SUGESTÕES PARA VOCÊ SE INSPIRAR:

- Sistemas de alerta antecipado para desastres naturais
- Aplicativos de evacuação e rotas seguras em situações de risco
- Drones para busca, resgate e entrega de suprimentos em áreas isoladas
- Estações móveis de energia limpa para abrigos temporários
- Plataformas de comunicação offline para comunidades sem acesso à internet
- Monitoramento de encostas, barragens e zonas de risco com sensores IoT
- Reutilização de água da chuva e soluções portáteis de purificação
- Tecnologias para reconstrução rápida e sustentável de moradias
- Gamificação para educação sobre prevenção de desastres



VEJA ALGUMAS SUGESTÕES PARA VOCÊ SE

INSPIRAR:

- Inteligência artificial para prever padrões de eventos extremos
- Gestão de abrigos e recursos em tempo real com uso de dashboards
- Apoio psicológico remoto via apps em contextos de calamidade
- Impressão 3D de estruturas de emergência para zonas afetadas
- Sistemas de energia solar para manter hospitais móveis funcionando
- Mapeamento colaborativo de áreas atingidas usando dados da comunidade



INDICAÇÃO DE FONTE

<https://disasterscharter.org/>

<https://www.youtube.com/watch?v=ZvExM-Z3E2w&t=5s>

A Carta Internacional Space and Major Disasters é uma iniciativa global que reúne agências espaciais de diversos países com o objetivo de fornecer dados de satélite gratuitos e em tempo real para apoiar ações de resposta a desastres naturais.

Sempre que ocorre um grande desastre, como enchentes, terremotos, furacões ou incêndios florestais, a Carta pode ser acionada por governos e instituições autorizadas. A partir disso, satélites são mobilizados para capturar imagens da região afetada, permitindo o monitoramento, a análise dos impactos e a tomada rápida de decisões em situações de emergência.



INDICAÇÃO DE FONTE

No site, é possível acessar:

- Casos recentes de desastres monitorados
- Mapas e imagens de satélite com áreas impactadas
- Tipos de desastres acompanhados
- Ferramentas de visualização e relatórios técnicos

É uma ferramenta poderosa para pesquisadores, estudantes e desenvolvedores que queiram trabalhar com dados reais e atuais em projetos voltados à gestão de crises, cidades resilientes, tecnologias emergenciais e soluções para eventos extremos.



PROGRAMAÇÃO

DATA	EVENTO
26/05	LIVE DE ABERTURA
26/05	CONTEÚDO PARA ALUNOS
06/06 até 23h55	ENTREGA NO PORTAL FIAP
Até 13/06	CORREÇÃO DOS PROFESSORES



REGRAS GERAIS

- Equipe: o desafio pode ser realizado individualmente ou em grupo de até 3 integrantes (sem exceções). Os grupos podem ser formados com alunos de turmas e turnos diferentes.
- Haverá chamada nos dias de aula para todas as disciplinas (mantendo os dias presenciais e remotos).



ENTREGAS

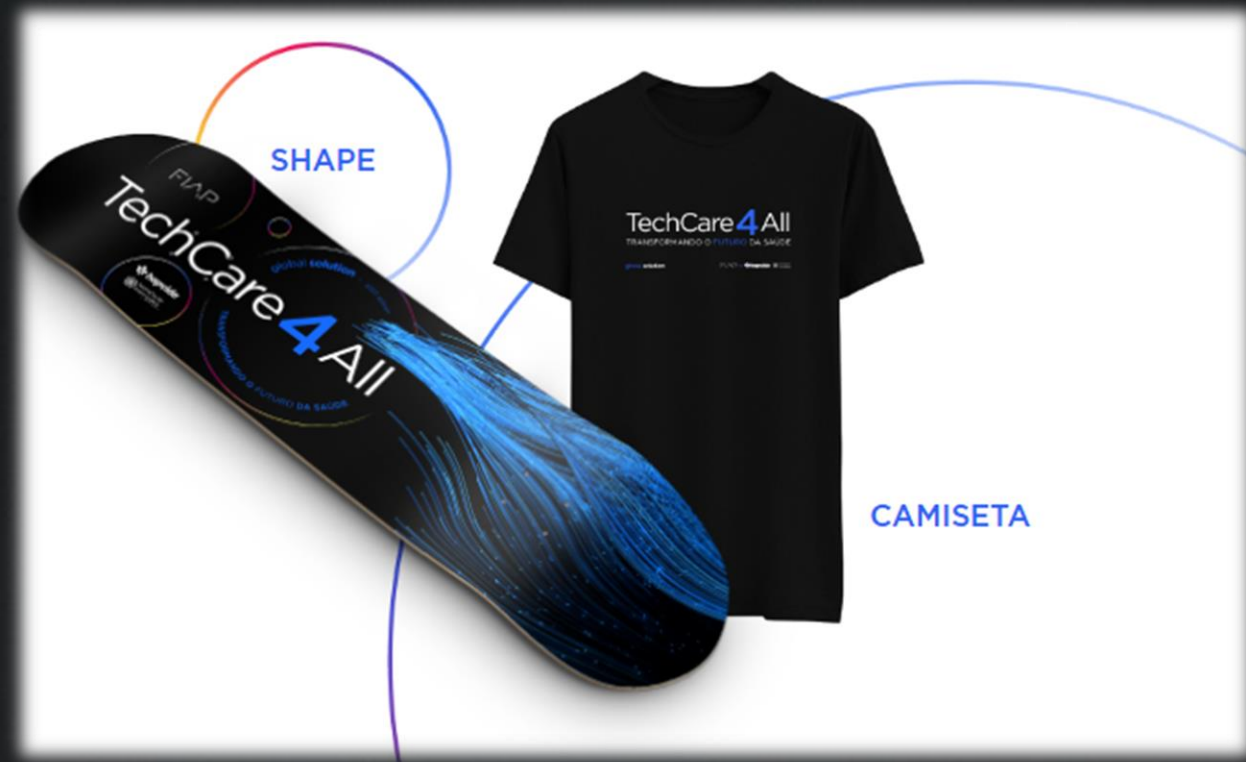
- As entregas de todas as disciplinas serão realizadas até o dia **06/06/25 até às 23h55** na área de trabalhos, no portal do aluno. *(Instruções de entrega estão no final do documento).*
- Entrega de um txt. no .zip de cada entrega: Esse txt deve ter o RM, nome do aluno e turma de cada integrante.



Cadastro dos grupos (prazo até : 26/05): <https://forms.gle/XphT9RV1aAFT6swL8>



- Os grupos serão avaliados além das notas por uma comissão de professores;
- O grupo que obter as melhores notas em todas as disciplinas, junto com a melhor avaliação do vídeo Pitch será o grande vencedor (shape e camisetas exclusivas);
- O grupo que obter nota igual ou maior que nove em todas as disciplinas, junto com a nota do vídeo Pitch da matéria de **Java Advanced**, que também tem que ser uma nota igual ou maior que nove, então esses grupos serão analisados pelos Scrum Master, Professores e o Coordenador, para a escolha de um único grupo vencedor.



ENTREGAS

Descrição das entregas por disciplina



DESAFIO

- O Grupo deve propor uma solução para o **tema da Global Solution**.
- Essa solução deve ser utilizada para desenvolver as entregas de **todas as disciplinas**.

ADVANCED BUSINESS DEVELOPMENT WITH .NET (1/2)

- **Desenvolvimento de uma API em .NET** (Web API / Minimal API) hospedada no Github com as seguintes especificações: (80 pts)
 - CRUD de pelo menos 3 entidades
 - Implementação de boas práticas de design de API's Restful (Hateoas, Rate Limit)
 - Apresentar documentação via Swagger
 - Cobertura de testes do projeto utilizando Xunit
 - Incluir microsserviços, de preferência com RabbitMQ
 - Uso de ML.NET dentro do projeto

ADVANCED BUSINESS DEVELOPMENT WITH .NET (2/2)

- **ReadMe do projeto:** Documentar o projeto com as boas práticas na abordagem de criação do ReadMe contendo: **(20 pts)**
 - Descrição do Projeto
 - Tecnologias utilizadas
 - Como Executar o Projeto
 - Documentação dos Endpoints
 - Instruções de testes

OBS: a organização e formatação do trabalho terá impacto na diminuição da nota



COMPLIANCE, QUALITY ASSURANCE & TESTS

1|2

- Descreva o seu projeto na forma de um Pitch (**10 pontos**):
- Descreva qual o problema e solução que está sendo tratado no projeto de vocês

O Pitch deverá ser feito na forma de documento, não precisa ser feito um vídeo para o pitch.

O Pitch deverá ser descritivo, claro e consiso e demonstrar a viabilidade da proposta colocada no projeto.

- Utilizando a ferramenta de planejamento de projeto ágil em nuvem (Ex.: Azure Boards), a qual foi apresentada em suas aulas, aplicando SCRUM, faça:
- Criação do backlog de produto, organizado em Épicos (Epic), Funcionalidades (Feature) e Itens de entrega de produto (Product backlog itens) - **10 pontos**
- Descrição de cada item do planejamento (Epic, Feature, Product backlog itens) - **10 pontos**
- Critérios de aceite de cada item do planejamento (Epic, Feature, Product backlog itens) - **10 pontos**
- Organização do backlog na ordem de prioridade de execução/realização do desenvolvimento por SPRINT– **10 pontos**



COMPLIANCE, QUALITY ASSURANCE & TESTS

2|2

Crie o plano de testes manuais, de validação no nível de sistema para seu projeto:

- Liste os testes planejados – 10 pontos
- Descreva os dados de entrada para cada teste e descreva os dados de saída para cada teste – 10 pontos
- Descreva o procedimento (passos) de teste aplicado – 10 pontos
- Aponte o status de cada teste (se pendente ou realizado e se realizado, indique quem executou o teste e se passou ou não na avaliação) - 20 pontos

Obs: trabalhe com dados controlados nos seus testes, predefinindo os valores de variáveis para input e output esperado.

- Entrega:

Gere um arquivo (**PDF**) com RM e nome dos integrantes, e o Link do seu projeto no Azure



DEVOPS TOOLS & CLOUD COMPUTING 1|3



Objetivo: Realizar a implementação de um dos projetos solicitados nas disciplinas:

- ADVANCED BUSINESS DEVELOPMENT WITH .NET
- JAVA ADVANCED
- MOBILE APP DEVELOPMENT

O grupo **TEM até TRÊS** opções para a entrega (e a nota pode valer de 0 a 100, **dependendo da qualidade das entregas**);

1º Realizar a implementação em um Serviço de Aplicativo e Banco em nuvem (Banco de Dados em Nuvem Pública MS Azure); • O que será avaliado nessa opção?

- Descrever brevemente o projeto e seus objetivos; **(0 PONTOS) – Caso não tenha, serão descontados 5 pontos.**
- Desenhar a Arquitetura do projeto de DevOps (Draw.io, Visual Paradigm ou Excalidraw); **(20 PONTOS)**
- Criar os serviços necessários: Plano de Serviço, Serviço de Aplicativo e Banco de Dados em nuvem (Banco de Dados em • Nuvem Pública MS Azure); **(20 pontos)**
- Link do projeto no GitHub com fonte e apresentação do Projeto (readme); **(5 pontos)**
- Realizar testes EM NUVEM de todo projeto, mostrando a criação dos recursos, a persistência de dados no Banco, após cada operação do CRUD, apresentando em uma gravação de vídeo; **(55 Pontos) – O vídeo deve ter explicação em voz ou legenda de cada passo, deve ser legível e audível em com qualidade de no mínimo 740p. Não é permitido executar em localhost, deve ser em nuvem**

OBSERVAÇÕES:

- Entregar Arquivo PDF com capa incluindo o nome do Grupo (Solução), integrantes e RM de cada aluno;
- No caso de projeto com API, informar o JSON do GET, PUT, POST e DELETE (na falta desse quesito, para soluções com API, haverá desconto na nota);



DEVOPS TOOLS & CLOUD COMPUTING ^{2|3}

2º Realizar a implementação em um [Azure Container Instance\(ACI\)](#) e Banco em nuvem (Banco de Dados em Nuvem Pública MS Azure);

O que será avaliado nessa opção?

- a) Descrever brevemente o projeto e seus objetivos; **(0 PONTOS) – Caso não tenha, serão descontados 5 pontos.**
- b) Desenhar a Arquitetura do projeto de DevOps (Draw.io, Visual Paradigm ou Excalidraw); **(20 PONTOS)**
- c) Criar os Serviços necessários: [ACR, ACI e Banco de Dados em nuvem \(Banco de Dados em Nuvem Pública MS Azure\)](#); **(20 pontos)**
- d) Link do projeto no GitHub com fonte, Dockerfile (**criação da imagem personalizada**) e apresentação do Projeto (README.md); **(5 pontos)**
- e) Realizar testes EM NUVEM de todo projeto, mostrando a criação dos recursos, a persistência de dados no Banco, após cada operação do CRUD, apresentando em uma gravação de vídeo; **(55 Pontos) – O vídeo deve ter explicação em voz ou legenda de cada passo, deve ser legível e audível em com qualidade de no mínimo 740p. Não é permitido executar em localhost, deve ser em nuvem**

OBSERVAÇÕES:

- Entregar Arquivo PDF com capa incluindo o nome do Grupo (Solução), integrantes e RM de cada aluno;
- Serão aceitos projetos com FRONT ou API. No caso de projeto com API, informar o JSON do GET, PUT, POST e DELETE (Informativo: na falta desse quesito, no caso de API, haverá desconto na nota);



DEVOPS TOOLS & CLOUD COMPUTING 3|3

3º Realizar a implementação no **Azure Pipelines** e **Banco em nuvem (Banco de Dados em Nuvem Pública MS Azure)**;



O que será avaliado nessa opção?

- a) Descrever brevemente o projeto e seus objetivos; **(0 PONTOS) – Caso não tenha, serão descontados 5 pontos.**
- b) Desenhar a Arquitetura do projeto de DevOps (Draw.io, Visual Paradigm ou Excalidraw); **(20 PONTOS)**
- c) Criar a **Pipeline com Build e Deploy**, e o Banco de Dados em nuvem **(Banco de Dados em Nuvem Pública MS Azure)**; **(20 pontos)**
- d) Link do projeto no GitHub com fonte, modo clássico ou arquivo **YML (YAML do Azure Pipelines)** e apresentação do Projeto (README.md); **(5 pontos)**
- e) Realizar testes EM NUVEM de todo projeto, mostrando como foi criada a estrutura, a execução da Pipeline e a persistência de dados no Banco, após cada operação do CRUD, em uma gravação de vídeo; **(55 Pontos) – O vídeo deve ter explicação em voz ou legenda de cada passo, deve ser legível e audível em com qualidade de no mínimo 740p. Não é permitido executar em localhost, deve ser em nuvem**

OBSERVAÇÕES:

- Entregar Arquivo PDF com capa incluindo o nome do Grupo (Solução), integrantes e RM de cada aluno;
- Serão aceitos projetos com FRONT ou API. No caso de projeto com API, informar o JSON do GET, PUT, POST e DELETE (Informativo: na falta desse quesito, no caso de API, **haverá desconto na nota**);



DISRUPTIVE ARCHITECTURES: IOT, IOB & GENERATIVE IA

DESAFIO: SOLUÇÃO BASEADA EM IOT

Objetivo: Integrar a sua solução proposta a um sistema IoT que integre sensores e/ou atuadores para coletar, processar e exibir os dados em tempo real, utilizando plataformas e protocolos adequados para a comunicação entre os dispositivos.



DISRUPTIVE ARCHITECTURES: IOT, IOB & GENERATIVE IA

REQUISITOS TÉCNICOS MÍNIMOS:

- 1. Sistema IoT:** O sistema IoT deve possuir, no mínimo, três dispositivos (sensores, atuadores ou ambos), que devem coletar, processar ou interagir com base nos dados coletados.
- 2. Hardware:** A plataforma utilizada deve ser uma das abordadas durante as aulas (Arduino/ESP32). A solução pode ser desenvolvida e testada no simulador Wokwi ou em hardware real.
- 3. Dashboard:** Implementação de um painel de controle para monitorar e/ou controlar os dispositivos em tempo real.
- 4. Gateway:** Utilizar Node-RED, Thinger.io, ThingSpeak ou plataformas similares como gateway. O gateway deve gerenciar a comunicação entre os dispositivos IoT e o dashboard, incluindo fluxos de processamento, análise e visualização de dados.
- 5. Protocolos:** A comunicação deve ser feita utilizando o protocolo MQTT e/ou HTTP no padrão JSON para a troca de mensagens entre os dispositivos e o gateway. Utilize tópicos personalizados e únicos.



DISRUPTIVE ARCHITECTURES: IOT, IOB & GENERATIVE IA

ENTREGÁVEIS:

A entrega da solução IoT deve ser composta por um conjunto de elementos que comprovem a funcionalidade, a aplicabilidade e o impacto da proposta desenvolvida. Cada item entregue será avaliado considerando a clareza das instruções, a organização dos arquivos e a apresentação das funcionalidades implementadas, destacando o potencial de aplicação em situações reais de desastres ou emergências urbanas.

Para documentar e demonstrar a solução proposta, os seguintes elementos devem ser entregues, garantindo clareza, organização e replicabilidade dos resultados.



DISRUPTIVE ARCHITECTURES: IOT, IOB & GENERATIVE IA

1. Protótipo Funcional (até 50 pontos):

- Apresentação do protótipo operacional que demonstre a funcionalidade da solução IoT, incluindo a integração dos sensores e/ou atuadores e a comunicação entre dispositivos.
- A solução deve ser testada e validada em um ambiente simulado (Wokwi) ou em hardware real, comprovando a eficácia na coleta e transmissão de dados.
- É fundamental que o protótipo apresente uma aplicação prática alinhada e integrada ao contexto da sua solução proposta.

2. Documentação (até 30 pontos):

- A entrega deve incluir um repositório GitHub bem estruturado, contendo todos os arquivos do projeto e o código-fonte devidamente comentado e organizado, com estrutura clara para replicação. A documentação deve conter:
 - ⑩ Descrição completa da solução, com textos explicativos e imagens ilustrativas;
 - ⑩ Instruções didáticas no README para configuração, execução e testes da aplicação;
 - ⑩ Detalhamento dos fluxos implementados no Node-RED, quando aplicável, evidenciando a comunicação entre dispositivos e gateway;
 - ⑩ Apresentação clara do código-fonte.



DISRUPTIVE ARCHITECTURES: IOT, IOB & GENERATIVE IA

3. Vídeo de Apresentação (até 20 pontos):

- Criação de um vídeo de até 3 minutos, apresentando de forma clara e objetiva o funcionamento da solução IoT, destacando as funcionalidades implementadas, a motivação do projeto e os benefícios previstos em cenários de risco ou emergência.

4. Arquivo de Entrega:

Um arquivo .txt contendo:

- Nomes completos dos integrantes do grupo;
- Link do repositório no GitHub;
- Link do vídeo no YouTube.

O arquivo deve ser organizado de forma clara para facilitar a identificação dos links e dos responsáveis pelo projeto.

Observação:

Atenção com os anexos. Entregas com arquivos corrompidos não serão considerados. Não serão considerados arquivos enviados pelo Teams



JAVA ADVANCED 1/2

Considerando o desafio proposto, você deve apresentar uma proposta de solução que utilize tecnologia e inovação para resolver ou mitigar os impactos dos eventos extremos na população. Utilizando Spring MVC com Thymeleaf, você e sua equipe devem construir uma aplicação web seguindo os padrões de princípios apresentados nas aulas do segundo semestre. A entrega deve contemplar os seguintes requisitos técnicos:

- Aplicação Web Spring MVC + Thymeleaf
- Templates dinâmicos
- OAuth 2 para autenticação
- CRUD completo com validação
- Internacionalização
- Spring AI
- RabbitMQ (pelo menos um produtor e um consumidor)
- Testes (unitários e de integração)



JAVA ADVANCED 2/2

A entrega será **avaliada** da seguinte forma:

- Cumprimento dos requisitos técnicos e boas práticas (70 pontos)
- Viabilidade e Inovação (10 pontos)
- Documentação e Apresentação (20 pontos)

O documento de **entrega** deve conter os seguintes artefatos:

- Link dos repositórios com os códigos fontes
- Link dos deploys
- Instruções para acesso e testes
- Vídeo demonstração da solução completa (máx 10 minutos)
- **Vídeo pitch (máx. 3 minutos)**



MASTERING RELATIONAL AND NON-RELATIONAL DATABASE

1 Modelagem Relacional (3FN) – 20 pontos

- Entidades bem definidas (ex: Usuário, Ocorrência, Área de Risco, Alerta, Campanha)
- Normalização adequada

2 Criação e Inserção de Dados – 15 pontos

- Tabelas com PKs, FKs, constraints
- Ao menos 5 registros por tabela

3 Desenvolvimento de Package em PL/SQL – 30 pontos

Toda a lógica programada deve estar dentro de uma única **PACKAGE** no Oracle

Incluindo obrigatoriamente:

- **Procedures** (inserção, alteração, exclusão)
- **Funções** (ex: retorno de pontuação, ranking, risco)
- **Triggers** (validação automática)
- **Cursors** (com LOOPS)
- **Blocos anônimos com IF/ELSE** e controle de fluxo
- **Relatórios complexos com JOINS e agregações**



MASTERING RELATIONAL AND NON-RELATIONAL DATABASE

4 Consultas SQL Avançadas – 15 pontos

- Múltiplos joins
- Agrupamentos e filtros com funções (COUNT, AVG, MAX, etc.)
- Exibição de relatórios simulando uso real do sistema

5 Integração com MongoDB – 10 pontos

- Armazenamento de dados não estruturados: logs, relatos livres, metadados de imagens
- Demonstração da integração no vídeo

6 Entrega PDF + Vídeo Explicativo– 10 pontos

- PDF contendo:
 - Modelo relacional
 - Package completa (código e prints de execução)
 - Exemplo de documentos no MongoDB
- Vídeo demonstrando:
 - Execução dos recursos em Oracle
 - Integração com MongoDB
 - Integração com Front e Back-end



MOBILE APPLICATION DEVELOPMENT

A partir da solução escolhida na Global Solution, seu desafio é desenvolver um aplicativo usando React Native.

Entregáveis:

- Arquivo texto com o endereço do repositório no GitHub Classroom
- Inclua o nomes dos integrantes do grupo e a descrição da solução da Global Solution no arquivo README.md do repositório.

Requisitos para Avaliação

- (10 pts) Criar no mínimo 5 telas e utilizar navegação entre elas (React Navigation ou Expo Router).
- (40 pts) Implementar um CRUD utilizando a API criada na disciplina JAVA ou .NET (Axios ou Fetch).
- (10 pts) Tela de login integrada ao Firebase
- (10 pts) Estilização do aplicativo, incluindo cores, fontes, e imagens personalizadas.
- (10 pts) Arquitetura: será avaliada a organização dos arquivos, nomes das variáveis, funções e componentes utilizados.
- (20 pts) Elabore uma gravação de um vídeo de todas as funcionalidades do App



MOBILE APPLICATION DEVELOPMENT

1. Telas e Navegação (React Navigation ou Expo Router) **10 pontos**

Requisitos:

- O aplicativo deve conter, no mínimo, 5 telas distintas com navegação fluida entre elas.
- A navegação deve seguir boas práticas de usabilidade e acessibilidade.

Avaliação:

- Pontos por navegação sem bugs e bem integrada com as funcionalidades.
- Avaliação da clareza e consistência da experiência de navegação.



MOBILE APPLICATION DEVELOPMENT

2. CRUD com API (Java/.NET) usando Axios ou Fetch **40 pontos**

Requisitos:

- Implementar operações de **Create, Read, Update, Delete** usando uma API RESTful desenvolvida nas disciplinas de backend (Java ou .NET).
- Utilização de bibliotecas adequadas (Axios ou Fetch) com tratamento de erros e feedback visual.

Avaliação:

- Os dados devem ser armazenados e manipulados via API, nunca apenas no dispositivo.
- Interface deve fornecer feedbacks visuais em caso de erro (mensagens, loaders, alertas).
- Avaliação da qualidade e semântica da integração com a API.



MOBILE APPLICATION DEVELOPMENT

3. Tela de Login Integrada ao Firebase **10 pontos**

Requisitos:

- Permitir **criação de nova conta** com e-mail e senha.
- Permitir **login** com conta existente.
- Permitir **logout**.
- A autenticação deve ser feita usando **Firebase Authentication**.
- Acesso às demais telas só deve ocorrer após login bem-sucedido.

Avaliação:

- Funcionalidade implementada corretamente com Firebase.
- Validação de entrada (e-mail e senha).
- Feedback visual ao usuário (erros, carregamento, sucesso).
- Usabilidade e clareza na interface.
- Fluxo seguro e bem definido de autenticação.



MOBILE APPLICATION DEVELOPMENT

4. Estilização com identidade visual personalizada **10 pontos**

Requisitos:

- Personalização de cores, fontes e imagens de acordo com o tema do app.
- Uso consistente do design, respeitando padrões de usabilidade.

Avaliação:

- Aplicação das guidelines da Apple/Google.
- Criatividade e identidade visual clara e funcional



MOBILE APPLICATION DEVELOPMENT

4. Arquitetura do Código **10 pontos**

Requisitos:

- Organização lógica de arquivos, pastas e componentes.
- Nomeação clara e padronizada de variáveis, funções e componentes.
- Separação adequada de responsabilidades (componentes, serviços, estilos, rotas, etc.).

Avaliação:

- Código limpo, legível e bem estruturado.
- Indentação correta e formatação padronizada (uso consistente de espaçamentos, quebras de linha, nomes descritivos, etc.).
- Adoção de boas práticas de desenvolvimento com React Native.
- Uso adequado de ferramentas de lint/formatter (como ESLint, Prettier ou configurador do próprio editor).
- Código mal formatado, com indentação irregular ou difícil leitura poderá resultar em perda de pontos.
- Bibliotecas utilizadas são relevantes, atualizadas e realmente necessárias. Pontos podem ser descontados por uso excessivo, desatualizado ou sem justificativa clara.



MOBILE APPLICATION DEVELOPMENT

5. Gravação de vídeo com todas as funcionalidades **20 pontos**

Requisitos:

- Apresentar o app em funcionamento real (emulador ou dispositivo).
- Demonstrar claramente as principais funcionalidades: navegação, CRUD, usabilidade, etc.
- Narração opcional, mas a apresentação deve ser clara.
- Duração máxima de 5 minutos.

Avaliação:

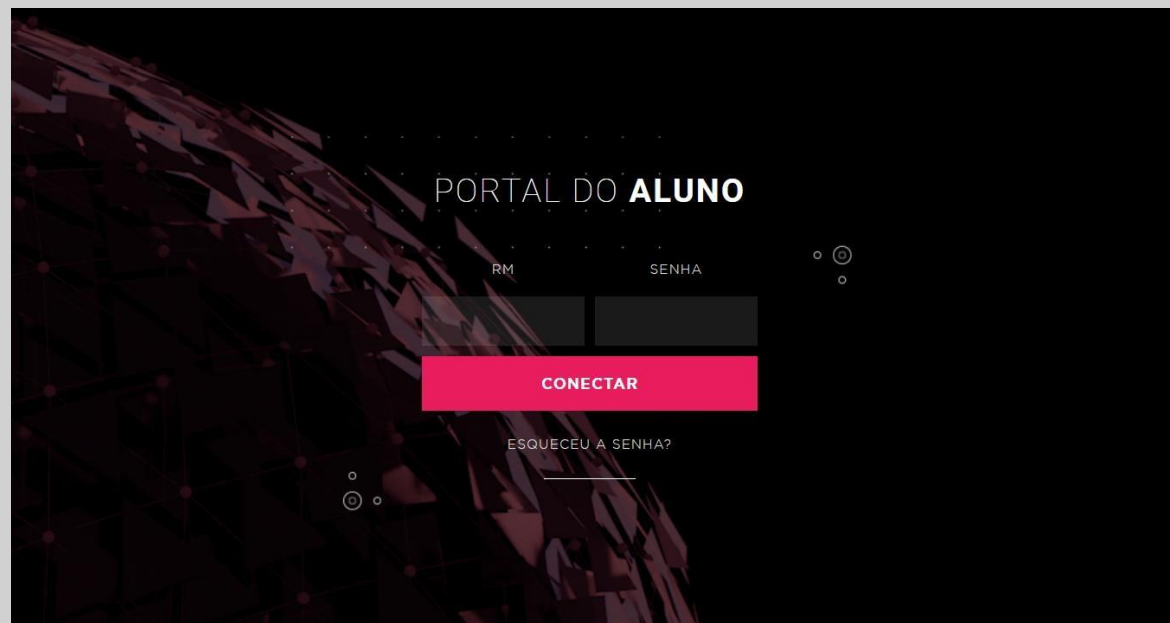
- Clareza na apresentação das funcionalidades.
- Demonstração da criatividade na resolução do problema proposto.
- Qualidade do vídeo e da demonstração geral do produto.

ENTREGA (PASSO A PASSO)

Como fazer as entregas da Global Solution?

1.0 Acesse o Portal do Aluno FIAP

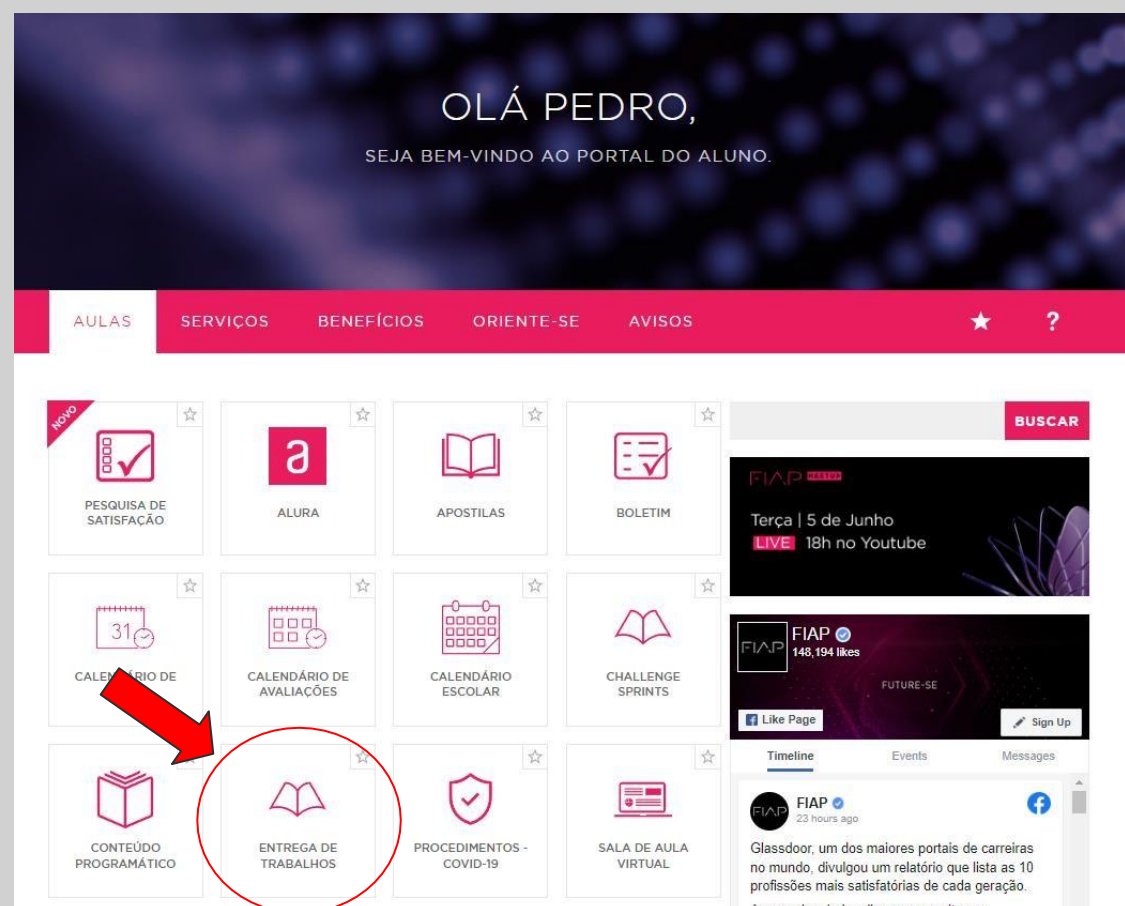
www2.fiap.com.br





ENTREGA (PASSO A PASSO)

2.0 Em Aulas, Clique na opção “Entrega de Trabalhos”





ENTREGA (PASSO A PASSO)

3.0 Clique em um trabalho referente a Global Solution


LISTA DE TRABALHOS		
1TDSB		
<div>07/06/2023 23:55:00</div> <div>NÃO ENTREGUE</div>	<div>BUILDING RELATIONAL DATABASE</div> <div>GLOBAL SOLUTION - BUILDING RELATIONAL DATABASE</div>	
<div>07/06/2023 23:55:00</div> <div>NÃO ENTREGUE</div>	<div>COMPUTATIONAL THINKING USING PYTHON</div> <div>GLOBAL SOLUTION - OBJETIVO DE DESENVOLVIMENTO SUSTENTÁVEL</div>	
<div>07/06/2023 23:55:00</div> <div>NÃO ENTREGUE</div>	<div>DOMAIN DRIVEN DESIGN</div> <div>GLOBAL SOLUTIONS - DDD</div>	
<div>07/06/2023 23:55:00</div> <div>NÃO ENTREGUE</div>	<div>RESPONSIVE WEB DEVELOPMENT</div> <div>GLOBAL SOLUTION 2023 - RESPONSIVE WEB DEVELOPMENT</div>	
<div>07/06/2023 23:55:00</div> <div>NÃO ENTREGUE</div>	<div>SOFTWARE DESIGN & TOTAL EXPERIENCE</div> <div>GLOBAL SOLUTION - SOFTWARE DESIGN & TOTAL EXPERIENCE</div>	



ENTREGA (PASSO A PASSO)

4.0 Anexe o arquivo do seu projeto referente a entrega escolhida

Na página de entrega, você pode conferir o seu grupo, a data de vencimento, e a descrição da entrega.

 ENTREGA DE TRABALHOS

INFORMAÇÕES DO TRABALHO

ANO	TURMA	DISCIPLINA
2023	ITDSB	SOFTWARE DESIGN & TOTAL EXPERIENCE

TEMA

GLOBAL SOLUTION - SOFTWARE DESIGN & TOTAL EXPERIENCE

DATA DE ENTREGA

07/06/2023 23:55

DESCRIÇÃO

GLOBAL SOLUTION - SOFTWARE DESIGN & TOTAL EXPERIENCE

INTEGRANTES

98043 - PEDRO CARVALHO PACHECO

COMENTÁRIOS

[Anexar Arquivos](#)

CADASTRAR COMENTÁRIO

ENTREGA DO TRABALHO

ARQUIVO

Tamanho Máximo: 50 MB.

Escolher arquivo

Nenhum arquivo escolhido

LINK DO ARQUIVO

Caso prefira ou o arquivo tenha mais que 50 MB, você pode fazer o upload do trabalho em um site de hospedagem de arquivos e enviar o link dele.

ENVIAR

Repita este mesmo processo para todas as outras entregas referentes que constam como **Global** Solution.



ENTREGA (PASSO A PASSO)

5.0 Confira o seu arquivo anexado.

O arquivo que você enviar na entrega fica registrado, você pode conferi-lo depois do envio.

ARQUIVOS ANEXADOS

- [Global Solution - Software Design & TX](#)

ENTREGA DO TRABALHO

ARQUIVO

[52ED5F5B-71FE-48CB-A3DC-D294B435F3E3.zip](#) (Entregue pelo(a) aluno(a) PEDRO CARVALHO PACHECO no dia 04/06/2023 às 07:28)

