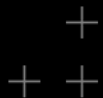




# FIAP GRADUAÇÃO



# ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

## FULL STACK, APPS & ARTIFICIAL INTELLIGENCE

Disciplina: SOFTWARE DESIGN & TX

45697056



# ++ | Apresentações

FIAP

Nome?

Hobby?

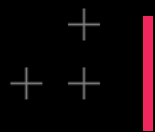
Curiosidade?

O que espera da disciplina?

O que você traz de experiência?

Trabalha na área?

...



# Apresentação da Disciplina

45697056



1º ANO 2022

## SOFTWARE DESIGN & TX

Aprenda a explorar requisitos para um projeto de software, criar modelos de arquitetura e especificações técnicas de engenharia que aproveitam componentes de software existentes e criam componentes e sistemas integrados. Gerencie projetos de forma ágil, mobilizando equipes full stack para a realização dos trabalhos do ciclo de vida de planejamento, desenvolvimento, avaliação e entrega de software. Tudo integrado à Total Experience (TX) que engloba a experiência completa da empresa, do funcionário ao cliente e ao usuário, fornecendo um ambiente de excelência para eles.

Estamos passando por um período diferente, mas utilizando a inteligência, será apenas mais uma vitória.

## NOSSA META

### COMPETÊNCIAS

- ❖ Identificar e selecionar os requisitos que o software deve atender.
- ❖ Especificar o sistema de informação a ser construído, detalhando sua arquitetura e funcionalidades por meio de uma linguagem de modelagem de sistemas (UML).
- ❖ Planejar, acompanhar e monitorar o desenvolvimento de software e gerenciar os produtos de software gerados.
- ❖ Todo o processo será conduzido de forma ágil, permitindo, dessa forma, alterações no projeto durante sua produção e sem causar grandes impactos e riscos ao projeto.

### OBJETIVO

- ❖ Compreender a cultura ágil e aplicar as principais técnicas de modelagem no desenvolvimento de software, determinando atividades a serem cumpridas dentro de um projeto de software, de forma a garantir a sua qualidade.

## DETALHES DO CONTEÚDO

- Projetos tradicionais e ágeis
- **Gestão de projetos ágil**
- Ciclo de vida, processo de software e modelagem de sistemas
- Etapas de criação de software
- Escopo do projeto e do produto
- Detalhamento do escopo do projeto
- Técnicas para coletar corretamente os requisitos
- Criação do projeto e documentação
- Ferramentas de projetos
- Ciclos de vida do software
- Diagrama de caso de uso
- Diagrama de classes
- Visão geral da UML e seus modelos dinâmicos e estáticos
- Identificação de Objetos e Classes a partir de Casos de Uso
- Classes, atributos, métodos e responsabilidade
- Relacionamentos entre classes (Componentização, Agregação, Classes Associativas)
- Diagramas

\*\*\*Existem outros tópicos, porém esses são os principais.



- ❖ Ciclo de vida, processo de software e modelagem de sistemas
- ❖ Análise e projeto de sistemas
- ❖ Conceitos de metodologia ágil
- ❖ Requisito de software
- ❖ Representação dos requisitos no modelo de sistemas
- ❖ Técnicas de levantamento de requisitos
- ❖ Modelagem de negócios aplicando UML
- ❖ Diagramas de caso de uso
- ❖ Modelagem de classe
- ❖ Análise orientada a objeto com UML
- ❖ Modelagem de objeto

## 1º SEMESTRE – Checkpoints (CK)

- ✓ Primeiro CK – Semana do dia 22/03
- ✓ Feedback e correção do CK – Semana do dia 29/03
- ✓ Segundo CK – Semana do dia 26/04
- ✓ Feedback e correção do CK – Semana do dia 03/05
- ✓ Terceiro CK – Semana do dia 24/05
- ✓ Feedback e correção do CK – Semana do dia 24/05

## 2° SEMESTRE

- ❖ Refinamento de atributos
- ❖ Refinamento de métodos
- ❖ Diagramas comportamentais para refinamento de projeto O.O.
- ❖ Diagrama de atividade (aplicado aos métodos)
- ❖ Diagrama de sequência
- ❖ Diagrama de estados
- ❖ CASE aplicado
- ❖ Cronograma
- ❖ Gerenciamento de configuração e mudança

## ■ Dúvidas...



**FIM**