

## PLANO DE ENSINO

para o Quadrimestre Suplementar (QS) – Resolução CONSEPE 240/2020

ANO LETIVO	QUADRIMESTRE	TURNO	CAMPUS
2021	Q3	Noturno	Santo André

CÓDIGO	NOME
MCZA010-13	Laboratório de Engenharia de Software
Turma	RECOMENDAÇÃO
NA1MCZA010-13SA	Engenharia de Software

## OBJETIVO

A disciplina tem por objetivo o desenvolvimento prático de sistemas de software segundo os paradigmas da Engenharia de Software. O estudo e aplicação das metodologias em todas as etapas do ciclo de vida do software visam desenvolver a capacidade de avaliação das metodologias para diferentes tipos de aplicações.

## Conteúdo

Revisão dos conceitos fundamentais de engenharia de software. Metodologias de desenvolvimento de software. Metodologias para desenvolvimento de sistemas orientados a objetos. Desenvolvimento Ágil. Estudo de casos reais utilizando as metodologias de desenvolvimento. Projetos a serem desenvolvidos utilizando as metodologias (ferramentas/ambientes serão utilizados na prática de tais estudos). Análise comparativa entre metodologias de desenvolvimento.

## SOBRE A DISCIPLINA

**Docente:** Paulo Roberto Miranda Meirelles

**Horário semanal:** segunda-feira das 21h às 23h | quarta-feira das 19h às 21h

**Curso no Moodle:**

TNA1MCZA010-13SA - Laboratório de Engenharia de Software - Paulo Meirelles - 2021.3

- <https://moodle.ufabc.edu.br/course/view.php?id=2439>

**Link do Google Meet:**

- <https://meet.google.com/joc-puet-fee>

- Todas as aulas serão síncronas: apenas as aulas teóricas serão gravadas
- A turma será dividida em equipes de desenvolvimento e atuarão em projetos reais a partir da terceira semana.
- Fique atento aos comunicados realizados no Moodle

## Aulas e Atividades

### Semana 1

- Apresentação da disciplina
- Introdução aos métodos ágeis

### Semana 2

- Planejamento ágil
- Apresentação dos projetos
- participação de alguns clientes e convidados
- Disponibilização do formulário para escolha dos projetos e outras informações

### Semana 3

- Seleção dos projetos
- Definição das equipes
- Organização de ambientes de desenvolvimento e comunicação

### Semana 4

- Iteração/sprint/ciclo 1
  - planejamento
  - desenvolvimento

### Semana 5

- Iteração/sprint/ciclo 1
  - desenvolvimento
  - entrega/revisão

### Semana 6

- Iteração/sprint/ciclo 2
  - planejamento
  - desenvolvimento

## Semana 7

- Iteração/sprint/ciclo 2
  - desenvolvimento
  - entrega/revisão
  - retrospectiva (1)

## Semana 8

- Iteração/sprint/ciclo 3
  - planejamento
  - desenvolvimento

## Semana 9

- Iteração/sprint/ciclo 3
  - desenvolvimento
  - entrega/revisão
  - “show me the code”

## Semana 10

- Iteração/sprint/ciclo 4
  - planejamento
  - mapeamento de dívida técnica
  - desenvolvimento

## Semana 11

- Iteração/sprint/ciclo 4
  - desenvolvimento
    - foco em refactoring e/ou testes

## Semana 12

- Iteração/sprint/ciclo 4
  - desenvolvimento
  - entrega final
  - retrospectiva
  - apresentação do projeto

## Semana 13

- Atividade substitutiva e Recuperação
- Finalizar issues/tarefas pendentes

## Critérios de avaliação

A avaliação da disciplina será dividida da seguinte forma:

- *Etapa “Iniciante” ( 20%)*
  - Código: repositório configurado e ambiente de desenvolvimento configurado
  - Equipe: atribuições de papéis, “standup” e mobs iniciais rolando, ambiente de trabalho informativo criado e definida política de horas fora da aula
  - Projeto: meio de comunicação com cliente/comunidade definido
- *Etapa “Saudável” (30%)*
  - Código: testes e commits frequentes
  - Equipe: bom rodízio de pares e consistência em mobs
  - Projeto: boa comunicação com o cliente/comunidade e ter feito a primeira entrega

- *Etapa “Maduro” (40%)*
  - Código: tracking, integração contínua, cobertura de testes e artefatos para continuidade do projeto
  - Equipe: auto organização ("coach invisível") e propriedade coletiva do código
  - Projeto: entregas aos cliente/comunidade
- *Auto-avaliação e avaliação do coach (10%)*
  - Individual: Auto-avaliação de cada membro
  - Coach: Avaliação do Coach sobre cada membro (professor e meta-coaches/monitoras avaliam os coaches)
  -

### **Conceito final**

A nota de cada etapa listada anteriormente será somada e convertida para uma escala de 0 a 10. O valor final será convertido para conceitos, conforme abaixo:

**A** =  $N \geq 9,0$

**B** =  $9,0 > N \geq 8,0$

**C** =  $8,0 > N \geq 7,0$

**D** =  $7,0 > MF \geq 6,0$

**F** =  $N < 6,0$

### **Avaliação Substitutiva (SUB)**

Estarão habilitados para a avaliação substitutiva os/as alunos/as que tiverem algum problema em um dos ciclos de desenvolvimento no período regular e contemplados pelo benefício de acordo com a Resolução CONSEPE no. 227, de 23 de abril de 2018. A cada fim de ciclo de desenvolvimento os casos e justificativas serão avaliadas.

### **Mecanismo de Recuperação (REC)**

Estarão habilitados para a avaliação de recuperação os alunos que obtiverem conceito final D ou F na conclusão de todas as avaliações aplicadas no período letivo regular, obedecendo às regras indicadas na Resolução CONSEPE no. 182, de 23 de outubro de 2014. O conceito final após a recuperação será calculado da mesma forma que para o período regular, com conceito máximo limitado a C.

### **Ferramentas de apoio**

1. GMeet + Discord: desenvolvimento remoto
2. Git (Github, GitLab): código e kanban
3. MetroRetro: retrospectivas online
4. Live Share do VSCode: compartilhamento do ambiente de desenvolvimento e pareamento/mobbing remoto
5. GNU Screen ou Tmate: compartilhar o terminal para pareamento remoto
6. Docker: facilita trocar entre ambientes (Linux, Mac, Windows) para cada pessoa ter seu setup