

## PLANO DE ENSINO

ANO LETIVO	QUADRIMESTRE	TURNO	CAMPUS
2022	Q3	Manhã	Santo André

CÓDIGO	NOME
MCZA010-13	Laboratório de Engenharia de Software
Turma	RECOMENDAÇÃO
DA1MCZA010-13SA	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 10h às 12h | quarta-feira das 08h às 10h

**Sala/Laboratório:** 407-2 (Santo André)

### **Curso no Moodle:**

DA1MCZA010-13SA - Laboratório de Engenharia de Software - Paulo

Meirelles - 2022.3

- <https://moodle.ufabc.edu.br/course/view.php?id=3969>

- Todas as aulas serão presenciais
- A turma será dividida em equipe de desenvolvimento e atuarão em projetos reais a partir da terceira semana.

## **AULAS E ATIVIDADES**

### Semana 1 (19 e 21/09)

- Apresentação da disciplina
- Revisão sobre Engenharia de Software
- Introdução aos Métodos Ágeis

### Semana 2 (26 e 28/09)

- Planejamento ágil
- Apresentação dos projetos
- Disponibilização do formulário para escolha dos projetos e outras informações

### Semana 3 (03 e 05/10)

- Seleção dos projetos
- Definição das equipes
- Organização de ambientes de desenvolvimento e comunicação
- *Quiz (05/10)*

### Semana 4 (10/10)

- Iteração/sprint/ciclo 1
  - planejamento
  - desenvolvimento

### Semana 5 (17 e 19/10)

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

### Semana 6 (24 e 26/10)

- Iteração/sprint/ciclo 2
  - planejamento
  - desenvolvimento

### Semana 7 (31/10)

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

### Semana 8 (07 e 09/11)

- Iteração/sprint/ciclo 3
  - planejamento
  - desenvolvimento

### Semana 9 (16/11)

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

### Semana 10 (21 e 23/11)

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

### Semana 11 (28 e 30/11)

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

### Semana 12 (05 e 07/12)

- Iteração/sprint/ciclo 4
  - desenvolvimento
  - entrega final
  - retrospectiva
  - apresentação do projeto
  - *Prova (07/12)*

### Semana 13 (12, 15 e 16/12)

- Atividade substitutiva e Recuperação
- Finalizar issues/tarefas pendentes

## CRITÉRIOS DE AVALIAÇÃO

A avaliação da disciplina será dividida da seguinte forma:

*Individual = 50%*

- *Presença = 10%*
- *Contribuições individuais = 10%*
- *Auto-avaliação e avaliação do coach (10%)*
- *Quiz/Prova (20%)*

*Grupo = 50%*

- *Fase 1 (5%)*
  - Código: repositório configurado e ambiente de desenvolvimento configurado
  - Equipe: atribuições de papéis, "stand up" e mobs iniciais rolando, ambiente de trabalho informativo criado e definida política de horas fora da aula
  - Projeto: planejamento inicial
- *Fase 2 (15%)*
  - Código: testes e commits frequentes
  - Equipe: bom rodízio de pares e consistência em mobs
  - Projeto: boa comunicação entre a equipe e ter feito a primeira entrega
- *Fase 3 (30%)*
  - 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

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

**O** = Ausência de avaliação ou de frequência (<75%), ou ainda não responder ao formulário de escolha de projeto ou não integrar uma equipe até a terceira semana.

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