

## INFORMAÇÕES GERAIS DA DISCIPLINA

*(esse documento pode sofrer alterações, acesse através do Moodle para garantir que está vendo a versão mais recente)*

### Objetivos Gerais

- Conhecer e praticar as principais técnicas da Engenharia de Software.

### Objetivos Específicos

- Conhecer uma visão sistemática da Engenharia de Software;
- Conhecer as principais disciplinas da Engenharia de Software;
- Praticar técnicas de Engenharia de Software através do desenvolvimento de um projeto de software.

### Canais e Atendimento

- Moodle
  - Material gravado e didático, entrega de listas, entrega do trabalho
  - <https://moodle2.quixada.ufc.br/course/view.php?id=680>
- Google Meet (ao vivo)
  - Dúvidas gerais, discussões complementares
  - Link da videochamada: <https://meet.google.com/cje-skaf-ykk>
- Telegram
  - Dúvidas gerais, discussões complementares
  - Mensagem direta para dúvidas ou assuntos individuais
  - Observação: a resposta pode demorar.
  - Link de entrada: <https://t.me/joinchat/bgOKNfe3RO1IODZh>

### Estimativa de tempo semanal (3 horas)

- Leitura do material didático: 50 min
- Assistir aula pré-gravada: 50 min
- Lista de exercícios: 30 min
- Aula ao vivo/trabalho prático: 50 min

### Avaliação

- Listas de exercícios = nota AP1
- Trabalho prático = nota AP2
- Média =  $(AP1 + AP2) / 2$
- Avaliação final para quem não atingir média 7,0

## CALENDÁRIO (\*) = vale frequência

	Data	Tópico	Formato
1	30/09	Apresentação da disciplina	Ao vivo (*)
2	01/10	Cap. 1: Introdução a Engenharia de Software	Leitura & Vídeo
3	07/10	Resolução da lista (Cap 1)	Ao vivo
4	08/10	Cap. 2: Processos	Leitura & Vídeo
5	14/10	Cap. 3: Requisitos <b>Trabalho – Entrega 0</b>	Leitura & Vídeo Moodle
6	21/10	Resolução das listas (Cap 2 e Cap 3) Oficina 1 – Escrita de histórias de usuário	Ao vivo (*)
7	22/10	Cap. 4: Modelos	Leitura & Vídeo
8	28/10	Resolução da lista (Cap 4) Tira-dúvidas sobre Trabalho	Ao vivo
9	29/10	Cap. 5: Princípios de Projeto	Leitura & Vídeo
10	04/11	Resolução da lista (Cap 5) Oficina 2 – Organização de backlog	Ao vivo (*)
11	05/11	Cap. 7: Arquitetura	Leitura & Vídeo
12	11/11	<b>Trabalho – Entrega 1</b>	Ao vivo (*)
13	12/11	Controle de versão com Git	Leitura & Prática
14	18/11	Resolução da lista (Cap 7) Tira-dúvidas sobre controle de versão	Ao vivo
15	19/11	Cap. 8: Testes	Leitura
16	25/11	Resolução da lista (Cap 8)	Ao vivo
17	26/11	Extra: Especificação de Testes Funcionais	Leitura & Vídeo
18	02/12	Oficina 3 – Especificação de testes	Ao vivo (*)
19	03/12	Cap. 9: Refatoração	Leitura & Vídeo
20	09/12	<b>Trabalho – Entrega 2</b>	Ao vivo (*)
21	10/12	Cap. 10: DevOps	Leitura & Vídeo
22	16/12	Resolução da lista (Cap 9 e Cap 10)	Ao vivo
23	17/12	Acompanhamento do trabalho	Agendado
24	23/12	Acompanhamento do trabalho	Agendado
<b>RECESSO NATALINO</b>			
25	13/01	Acompanhamento do trabalho	Agendado
26	14/01	Acompanhamento do trabalho	Agendado
27	20/01	Acompanhamento do trabalho	Agendado
28	21/01	Acompanhamento do trabalho	Agendado
29	27/01	Acompanhamento do trabalho	Agendado
30	28/01	<b>Trabalho – Entrega 3</b>	Ao vivo (*)
31	03/02	Retrospectiva	Ao vivo
32	04/02	Avaliação Final	Moodle

Lista	Data	Assunto
C1	06/10	Cap. 1: Introdução a Engenharia de Software
C2	13/10	Cap. 2: Processos
C3	20/10	Cap. 3: Requisitos
C4	27/10	Cap. 4: Modelos
C5	03/11	Cap. 5: Princípios de Projeto
C7	17/11	Cap. 7: Arquitetura
C8	24/11	Cap. 8: Testes
C9	08/12	Cap. 9: Refatoração
C10	15/12	Cap. 10: DevOps

Horário limite de entrega  
em todas as datas: 23h59m

## Frequência

- Aulas ao vivo com indicação (\*) valem 2 presenças (7x2=14) [síncrono]
- Lista de exercícios valem 2 presenças (9x2=18) [assíncrono]
- Total = 32 presenças (18 assíncronas e 14 síncronas)
- Mínimo para aprovação = 24 presenças (75%) de qualquer tipo

## Listas de Exercícios (AP1)

- Respostas entregues via Moodle
- Cada lista vale 10,0 pontos
- AP1 = média das listas

## Trabalho Prático (AP2)

- Equipes de 3 e 5 integrantes
- Nota única ao final da disciplina (até 10,0)
- Após cada entrega, a equipe recebe feedback com erros/problemas nos artefatos
  - Se a equipe corrigir até a entrega final, não perde pontos
  - Se a equipe não fizer uma entrega, ou não entregar todo o escopo, perde 2,0 pontos na nota final
- Descrição detalhada abaixo

## TRABALHO PRÁTICO

### Objetivo

- Desenvolver um projeto de software em equipe
- Qual o sistema a ser desenvolvido?
  - Cada equipe escolhe o sistema a ser desenvolvido
  - Deve conter interface gráfica de usuário (desktop, web ou móvel)
  - Deve conter funções que exijam uso de persistência de dados
- Plataforma/linguagens
  - Livre escolha da equipe
  - Recomenda-se escolher linguagens/plataforma que já sejam de conhecimento da equipe
  - Não serão aceitos projetos baseados em plataformas como Office, Sharepoint, e similares

### Entrega 0. Formação da equipe e tema

- Apenas um(a) membro da equipe envia através do Moodle as informações:
  - Lista completa de integrantes
  - Nome do projeto (pode mudar posteriormente)
  - Descrição do tema do projeto
- O tema do projeto será aprovado pelo professor.

### Entrega 1. Requisitos

- Documento de visão (seções 1, 2, 3):  
[https://drive.google.com/file/d/1DrZwQ11khy5E4EjJOR8FF-9Hgwz\\_bLNb/view?usp=sharing](https://drive.google.com/file/d/1DrZwQ11khy5E4EjJOR8FF-9Hgwz_bLNb/view?usp=sharing)
- Histórias de usuário organizados em ferramenta de backlog
- Documento de visão deve estar versionado (git ou similar)

### Entrega 2: Modelagem de dados e Testes

- Modelo de banco de dados (obrigatório uso de ferramenta)
  - Entidades, atributos e relacionamentos que serão persistidos
  - Incluir na modelagem todo o escopo dos requisitos
- Especificação de testes funcionais (template em breve)
  - Especificar casos de testes para 50% dos requisitos
- Apresentar esboço de arquitetura ou primeira versão do sistema
- Uso de controle de versão para todos os artefatos

### Entrega 3: Projeto completo

- Implementação do sistema (mín. 4 histórias)
- Todos os demais artefatos atualizados
- Uso de controle de versão para artefatos

### Entregas ao vivo

- Apresentação ao vivo da equipe
- Apresentar os artefatos produzidos

- Enviar via Moodle os artefatos ou links para ferramentas

### **Critérios de avaliação**

*Os critérios de avaliação serão checados após a entrega final;*

*Não-conformidades encontradas nas entregas anteriores podem ser resolvidas pela equipe.*

- Requisitos = 2,0
  - Qualidade da elaboração da visão do sistema
  - Qualidade da especificação de requisitos
- Persistência de dados = 2,0
  - Qualidade da modelagem de dados
- Testes = 2,0
  - Qualidade da especificação de testes
- Implementação = 3,0
  - Sistema deve funcionar
  - Pelo menos 4 histórias implementadas
- Configuração = 1,0
  - Uso de controle de versão ao longo do projeto

### **Sugestões de Ferramentas de Gerenciamento**

- Gerenciamento de tarefas
  - Trello + Bitbucket <https://bitbucket.org/product/features/trello-boards>
  - Trello <https://trello.com>
  - Jira <https://www.atlassian.com/software/jira>
- Requisitos
  - Planilha eletrônica  
<https://drive.google.com/file/d/18XrXS667wax5OdwMmCtohmUwsoeEi71/view?usp=sharing>
  - Jira <https://www.atlassian.com/software/jira>
  - Trello <https://trello.com>
- Modelagem
  - diagrams.net <https://app.diagrams.net/>
- Configuração (controle de versão)
  - Github <http://github.com>
  - Gitlab <https://about.gitlab.com/>
  - Atlassian <https://bitbucket.org/>