

## 1. Identificação

Curso:	SISTEMAS PARA INTERNET 2020/01		
Período letivo:	1º SEM/2023	Turno:	NOTURNO
Unidade Curricular:	TESTES AUTOMATIZADOS		
Unidade:	CAMPUS UBERLÂNDIA CENTRO		
Tipo:	TEÓRICA/PRÁTICA		
Professor(es)	BRUNO QUEIROZ PINTO	(PRINCIPAL)	
Carga horária:	83.00		
Duração aula:	50 minutos		

## 2. Ementa

Introdução e contextualização de testes automatizados. Testes de Unidade. Testes de Integração. Testes de Aceitação. Testes Funcionais (end-to-end). Métricas para aferir cobertura de testes. Visão geral de desenvolvimento orientado a testes (TDD).

## 3. Objetivos

Criação de testes automatizados nos sistemas, para que cada unidade do sistema seja testada individualmente. Construção de testes de integração, para que módulos completos do sistema sejam testados, como por exemplo serviços que serão disponibilizados para outras aplicações. Construção de testes de aceitação para que as interfaces do sistema sejam testadas automaticamente. Utilização de métricas para aferir cobertura de testes.

## 4. Conteúdo Programático

- Fundamentos de testes automatizados
  - Tipos de testes
  - Benefícios
  - Boas práticas e padrões
  - TDD - Test Driven Development
- JUnit
  - Básico (vanilla)
  - Spring Boot
- Mockito
  - @Mock
  - @InjectMocks
  - Mockito.when / thenReturn / doNothing / doThrow
  - ArgumentMatchers
  - Mockito.verify
  - @MockBean
  - @MockMvc
- Testes de Integração
- Testes de Aceitação com Selenium IDE e Selenium WebDriver
- Métricas para aferir cobertura de testes

## 5. Metodologia

- Aulas expositivas
- Atividades práticas
- Exercícios e desafios propostos
- Criação dos cenários de testes utilizando a ferramenta STS
- Utilização do JUnit e Mockito para construção dos testes
- Mesclagem de aulas teóricas e práticas
- Integração com a disciplina de Projeto Back-end Monolítico com ORM.
- Contemplar, de acordo com os estudantes com necessidades educacionais específicas (NEEs), adaptação e/ou flexibilização da unidade curricular, considerando a diversidade, o significado prático e instrumental dos conteúdos, metodologias de ensino e recursos didáticos diferenciados, como também processos de avaliação adequados.

## 6. Recursos Didáticos

- Google Drive e google forms;
- Correio eletrônico;
- Projetor Multimídia;
- Lousa;
- Apostilas, resumos;
- Listas de exercícios;
- Linguagem Java e frameworks : JUnit, Mockito, Selenium;
- Google classroom, Código da turma: hlimmrr

## 7. Avaliação (critérios, valores, procedimentos, recuperação)

Os seguintes instrumentos serão utilizados para avaliação dos alunos:

- 1º Atividade avaliativa: 5 pontos (1 pontos de avaliação atitudinal/participação)
- 2º Atividade avaliativa: 20 pontos (3 pontos de avaliação atitudinal/participação)
- 3º Atividade avaliativa: 20 pontos (3 pontos de avaliação atitudinal/participação)
- 4º Atividade avaliativa: 15 pontos (2 pontos de avaliação atitudinal/participação)
- 5º Atividade avaliativa: 15 pontos (2 pontos de avaliação atitudinal/participação)
- 6º Atividade avaliativa: 10 pontos (2 pontos de avaliação atitudinal/participação)
- 7º Atividade avaliativa: 15 pontos (3 pontos de avaliação atitudinal/participação)

Plano de recuperação paralela: Para cada avaliação de competências técnicas em que o estudante obtiver menos que 60% de aproveitamento, será oferecido ao estudante um programa de estudos de recuperação, seguido de uma nova avaliação. O resultado desta tarefa avaliativa não pode reduzir o valor da nota já obtida pelo aluno neste tipo de atividade. O aluno não poderá recuperar a pontuação de avaliação atitudinal/participação.

## 8. Referências

Bibliografia Básica:

DELAMARO, M. E.; JINO, M.; MALDONADO, J. C. Introdução ao teste de software. São Paulo: Editora Campus, 2007.  
FILHO, W. P. P. Engenharia de software: fundamentos, métodos e padrões. Rio de Janeiro: LTC, 2009.  
KOSCIANSKI, A. Qualidade de software. São Paulo: Novatec, 2007.

Bibliografia Complementar:

PILONE, D., MILES, R. Use a cabeça! Desenvolvimento de software. Rio de Janeiro: Alta Books, 2008.  
SOMMERVILLE, I. Engenharia de software. São Paulo: Addison Wesley, 2007.

## Cronograma das aulas

**PLANO DE ENSINO**

SEMANA	Nº AULAS PREVISITAS	DESCRIÇÃO	C. H. TEÓRICA	C.H. PRÁTICA
1	4	(02/03) Apresentação Disciplina e Pesquisa sobre Métricas de testes	01:40	01:40
2	4	(09/03) Fundamentos de Teste : tipos de testes: Testes de Unidade, Testes de Integração, Testes de Aceitação, Testes Funcionais (end-to-end), Benefícios, boas práticas. Apresentação do Framework JUNIT.	01:40	01:40
3	4	(16/03) TDD - Desenvolvimento orientado a testes	01:40	01:40
4	7	(23/03) Testes de Unidade - JUNIT Básico(3 aulas EAD - Atividade A1)	01:40	04:10
5	6	(30/03) Testes de Unidade - JUNIT Básico e segunda atividade avaliativa(2 aulas EAD - Atividade A2)	01:40	03:20
6	4	(13/04) Testes de Unidade - JUNIT Básico	01:40	01:40
7	6	(20/04) Testes de Unidade - JUNIT Spring Boot(2 aulas EAD - Atividade A3)	01:40	03:20
8	4	(27/04) Testes de Unidade - JUNIT Spring Boot	01:40	01:40
9	6	(04/05) Testes de Unidade - JUNIT Spring Boot(2 aulas EAD - Atividade A4)	01:40	03:20
10	4	(11/05) Mockito	01:40	01:40
11	8	(18 e 20/05) Mockito	03:20	03:20
12	8	(25/05) Mockito e quinta atividade avaliativa(4 aulas EAD - Atividade A5)	01:40	05:00
13	4	(01/06) Teste de integração	01:40	01:40
14	4	(15/06) Teste de integração	01:40	01:40
15	12	(22 e 24/06) Teste de integração e sexta atividade avaliativa(4 aulas EAD - Atividade A6)	03:20	06:40
16	4	(29/06) Teste de aceitação	01:40	01:40
17	4	06/07 - Testes de Aceitação	01:40	01:40
18	7	(13/07) Teste de aceitação(3 aulas EAD - Atividade A7)	01:40	04:10
<b>18</b>	<b>100</b>	<b>Total</b>	<b>33h20</b>	<b>50h</b>

BRUNO QUEIROZ PINTO

PRINCIPAL

EDSON ANGOTI JUNIOR

COORDENADOR(ES) DO CURSO