



## Plano de Ensino de Disciplina

<b>Código da Disciplina:</b>	56977	<b>Vigência:</b>	1 / 2024
------------------------------	-------	------------------	----------

<b>Disciplina:</b>	LABORATÓRIO DE DESENVOLVIMENTO DE SOFTWARE
--------------------	--

<b>Código do Curso:</b>	372	<b>Curso:</b>	Engenharia de Software
-------------------------	-----	---------------	------------------------

<b>Unidade:</b>	UNID. EDUC. LOURDES - PPL
-----------------	---------------------------

<b>Turno:</b>	NOITE	<b>Período:</b>	4	<b>Currículo:</b>	37202
---------------	-------	-----------------	---	-------------------	-------

<b>Carga Horária</b>
----------------------

PRÁTICA 40 (GRADE 40)

TOTAL 40

<b>Requisitos</b>
-------------------

Disciplina	Código	Classificação
PROJETO DE SOFTWARE	56976	Co-requisito

<b>Ementa</b>
---------------

Elaboração de um projeto de um software dividido em camadas. Construção da camada de apresentação (interfaces reais). Construção da camada de aplicação. Construção da camada de domínio. Construção da camada de persistência/acesso a dados (Implementação de Bancos de Dados). Desenvolvimento do sistema projetado.

<b>Objetivos</b>
------------------

Capacitar o aluno a elaborar projetos de um software dividido em camadas; construir a camada de apresentação (interfaces reais); construir a camada de aplicação; construir a camada de domínio; construir a camada de persistência/acesso a dados e desenvolver o sistema projetado.

<b>Métodos Didáticos</b>
--------------------------

O desenvolvimento da disciplina LABORATÓRIO DE DESENVOLVIMENTO DE SOFTWARE se dará com apoio do Ambiente Virtual de Aprendizagem (AVA) Canvas, da plataforma de comunicação e colaboração Microsoft Teams (em eventuais aulas on-line) e do Sistema de Gestão Acadêmica da PUC Minas (SGA), contando com as seguintes estratégias pedagógicas:

- Aulas expositivas sobre técnicas e orientações para utilização de tecnologias utilizadas na disciplina.
- Análise, projeto e implementação, em grupo, de pequenos softwares usando metodologia ágil.
- Exercícios práticos sobre conceitos que apoiam o desenvolvimento dos trabalhos.

Observação: As aulas se darão através do sistema Microsoft Teams apenas em eventuais situações de aula on-line. De modo geral, a disciplina acontecerá presencialmente, através de todos os métodos didáticos supracitados.

<b>Unidades de Ensino</b>
---------------------------

Unidade 0 - Apresentação da Disciplina (01 hora-aula)



## Plano de Ensino de Disciplina

Boas-vindas aos alunos, apresentação do plano de ensino, do contexto da disciplina no currículo do curso, da bibliografia utilizada, dos aspectos metodológicos, da distribuição de pontos no semestre, dos requisitos da disciplina que serão cobrados nos trabalhos interdisciplinares, exercícios e outras atividades.

### Unidade 1 - Revisão e nivelamento (01 hora-aula)

- 1.1. Descrição de casos de uso
- 1.2. Organização dos diagramas de classes
- 1.3. Arquitetura em camadas

### Unidade 2 - Desenvolvimento do laboratório 1 (08 horas-aula)

- 2.1. Definição da arquitetura e do ambiente de configuração
- 2.2. Planejamento do incremento
- 2.3. Documentação dos Requisitos
- 2.4. Desenvolvimento dos requisitos
- 2.5. Apresentação da primeira versão
- 2.6. Estruturação camada de persistência

### Unidade 3 - Desenvolvimento do laboratório 2 (08 horas-aula)

- 3.1. Correção do primeiro laboratório
- 3.2. Planejamento do incremento
- 3.3. Documentação dos Requisitos
- 3.4. Desenvolvimento dos requisitos
- 3.5. Apresentação da segunda versão
- 3.6. Validação dos modelos de banco de dados

### Unidade 4 - Desenvolvimento do laboratório 3 (08 horas-aula)

- 4.1. Correção do segundo laboratório
- 4.2. Planejamento do incremento
- 4.3. Documentação dos Requisitos
- 4.4. Desenvolvimento dos requisitos
- 4.5. Apresentação da versão completa
- 4.6. Verificação qualidade documentação

### Unidade 5 - Desenvolvimento do laboratório 4 (06 horas-aula)

- 5.1. Correção do terceiro laboratório
- 5.2. Planejamento do incremento
- 5.3. Documentação dos Requisitos
- 5.4. Desenvolvimento dos requisitos
- 5.5. Apresentação da versão completa
- 5.6. Verificação qualidade documentação

### Unidade 6 - Desenvolvimento do laboratório 5 (06 horas-aula)

- 6.1. Correção do quarto laboratório
- 6.2. Planejamento do incremento
- 6.3. Documentação dos Requisitos
- 6.4. Desenvolvimento dos requisitos
- 6.5. Apresentação da versão completa
- 6.6. Verificação qualidade documentação

### REAVLIAÇÃO (02 horas-aula)

#### Processo de Avaliação

1. Laboratório 01: 15 pontos
2. Laboratório 02: 20 pontos
3. Laboratório 03: 20 pontos
4. Laboratório 04: 20 pontos
5. Laboratório 05: 20 pontos



## Plano de Ensino de Disciplina

6. Avaliação de Desempenho Acadêmico (ADA): 05 pontos

TOTAL..... 100 pontos

Reavaliação (prova) ..... 20 Pontos

### Observações:

- Os trabalhos não entregues nas datas estipuladas terão sua pontuação diminuída em 10% a cada aula em atraso.
- Reavaliação: os alunos que não obtiverem o total de 60 pontos no semestre e que não tenham sido reprovados por frequência poderão realizar uma prova de reavaliação no valor de 20 pontos, que poderá substituir a menor das notas obtidas nos laboratórios 2, 3, 4 ou 5.
- Consulta a periódicos científicos para realização de atividades da disciplina, a definir.

### Observações

Caso a carga horária da disciplina não seja integralizada, serão realizadas uma ou mais atividade(s) extra-classe para complementação de carga horária.

Justificativas da Bibliografia.

### Bibliografia Básica

- ACM TRANSACTIONS ON SOFTWARE ENGINEERING AND METHODOLOGY.

Possui publicações de artigos sobre os mais variados aspectos do projeto e construção de software das mais variadas complexidades. Apresenta artigos que cobrem todo o ciclo de desenvolvimento do software: especificação, projeto, desenvolvimento e manutenção e que mostram comparações entre técnicas usadas no processo de desenvolvimento. Os artigos são úteis também para que o aluno tenha um entendimento da evolução das técnicas e das metodologias usadas no processo de desenvolvimento de software.

- LARMAN, Craig. Utilizando UML e padrões : uma introdução á análise e ao projeto orientados a objetos e desenvolvimento iterativo.

Aborda o processo de desenvolvimento de software de forma evolutiva, mostrando e ilustrando como produzir os artefatos que devem ser gerados no trabalho da disciplina. Além disso, apresenta os padrões GRASP de forma clara e simplificada. Ele mostra como definir o projeto de forma a facilitar a comunicação e a evolução do software. Trata também questões importante sobre padrões arquiteturais e frameworks de persistências. Mostra ainda como planejar e conduzir um processo de desenvolvimento de software com metodologias ágeis.

- VERNON, Vaughn. Implementando o Domain-Driven Design.

Apresenta uma abordagem completa para o entendimento de domain-driven design (DDD) a fim de conectar fluentemente padrões estratégicos às ferramentas táticas fundamentais de programação. Apresenta técnicas práticas de DDD por meio de exemplos a partir de domínios familiares. É importante para o engenheiro de software de todos os níveis de experiência que visem aprimorar seus resultados, projetar e implementar aplicações de domain-driven consistente com o melhor estado atual da prática profissional. Além disso, é relevante para o aluno, pois traz uma perspectiva diferente de tratar projetos de software.

- WAZLAWICK, Raul Sidnei. Análise e design orientados a objetos para sistemas de informação: modelagem com UML, OCL e IFML.

Apresenta de forma clara e pratica todo o processo de desenvolvimento de software orientado a objetos. Ele explica e ilustra todos artefatos solicitados no desenvolvimento do trabalho da disciplina. Além disso, apresenta e detalha padrões



## Plano de Ensino de Disciplina

organizacionais necessários para tratar a arquitetura do software proposta na disciplina.

### Bibliografia Complementar

- ELMASRI, Ramez; Navathe, Shamkant B. Sistema de Banco de Dados.

Apresenta de forma prática todos os conceitos relacionados a implementação de um banco de dados. Mostra como definir o projeto do banco de dados e como gerar as DDLs e DLLs para a geração do banco de dados que será criado no software a ser desenvolvido como trabalho da disciplina.

- BEZERRA, Eduardo. Princípios de análise e projeto de sistemas com UML.

Aborda conceitos de orientação a objetos e mostra como usar os diagramas da UML desde a análise até o projeto do software. Mostra também exemplos de códigos em Java que ilustram situações similares às encontradas no desenvolvimento do trabalho. Além disso, os principais artefatos de documentação exigidos na disciplina são exemplificados. Isso facilita o entendimento dos vários conceitos e ajuda o aluno no processo de derivar o projeto a partir de um problema.

- DENNIS, Alan. Análise e projeto de sistemas.

Apresenta todo o ciclo do processo de desenvolvimento de software. É apresentado uma aplicação exemplo e todas as etapas do processo de desenvolvimento são exemplificadas nessa aplicação. Serve de guia para o aluno. Os capítulos nove e os seguintes ajudam muito nas atividades de projeto e implementação do software.

- FOWLER, Martin. Refatoração aperfeiçoamento o projeto de código existente.

Aborda os conceitos necessários para que os alunos possam otimizar o software por meio de refatoração. Apresenta também tecnologias que podem ser utilizadas na refatoração.

- GUEDES, Gilleanes T. A. UML 2: uma abordagem prática.

Aborda de forma prática como modelar software com UML. A notação de cada diagrama é exemplificada com vários exercícios. Mostra ainda um estudo de caso que pode ser referência no desenvolvimento do trabalho da disciplina. Essa referência é usada nas atividades de análise e projeto exigidas na disciplina.

- IEEE TRANSACTIONS ON SOFTWARE ENGINEERING.

Apresenta publicações de impacto sobre todo o processo de construção de software. As publicações são úteis para que o aluno tenha um entendimento da evolução das técnicas e das metodologias usadas no processo de desenvolvimento de software.

### Bibliografia

### Básica



## Plano de Ensino de Disciplina

- ACM TRANSACTIONS ON SOFTWARE ENGINEERING AND METHODOLOGY.. New York, N.Y., USA: Association for Computing Machinery, 1992-. Trimestral. ISSN 1049-331X. Disponível em: <https://dl-acm-org.ez93.periodicos.capes.gov.br/citation.cfm?id=J790&picked=prox>. Acesso em: 2 jul. 2018. (Periódico On-line).

- LARMAN, Craig. Utilizando UML e padrões: uma introdução á análise e ao projeto orientados a objetos e desenvolvimento iterativo. 3. ed. Porto Alegre: Bookman, 2007. E-book. ISBN 9788577800476. (Livro Eletrônico).

- VERNON, Vaughn. Implementando o Domain-Driven Design. Rio de Janeiro, RJ: Alta Books, 2016. 628 p. ISBN 9788576089520. (Disponível no Acervo). Nº de Exemplares: 21.

- WAZLAWICK, Raul Sidnei. Análise e design orientados a objetos para sistemas de informação: modelagem com UML, OCL e IFML. 3. ed. Rio de Janeiro, RJ: Elsevier, Campus, c2015. 462 p. ISBN 9788535279849. (Disponível no Acervo). Nº de Exemplares: 23.

### Complementar

- BEZERRA, Eduardo. Princípios de análise e projeto de sistemas com UML. 3. ed. rev. e atual. Rio de Janeiro: Elsevier, 2015. xvii, 398 p. ISBN 9788535226263. (Disponível no Acervo). Nº de Exemplares: 13.

- DENNIS, Alan. Análise e projeto de sistemas. 5. Rio de Janeiro LTC 2014 1 recurso online ISBN 978-85-216-2634-3. (Livro Eletrônico).

- ELMASRI, Ramez; Navathe, Shamkant B. Sistemas de banco de dados, 7ª ed. Editora Pearson 1152 ISBN 9788543025001. (Livro Eletrônico).

- FOWLER, Martin. Refatoração: aperfeiçoamento o projeto de código existente. Porto Alegre: Bookman, 2004. E-book. ISBN 9788577804153. (Livro Eletrônico).

- GUEDES, Gilleanes T. A. UML 2: uma abordagem prática. 2. ed. São Paulo: Novatec, c2011. 484 p. ISBN 9788575222812. (Disponível no Acervo). Nº de Exemplares: 5.

- IEEE TRANSACTIONS ON SOFTWARE ENGINEERING. New York: IEEE Computer Society, 1975-. Mensal,. ISSN 0098-5589. Disponível em: <<https://ieeexplore-ieee-org.ez93.periodicos.capes.gov.br/xpl/RecentIssue.jsp?punumber=32>> (Periódico On-line).



## Plano de Ensino de Disciplina

**Vigência:** 1<sup>º</sup> / 2024

**Situação:** Aprovado

**Data Aprovação:** 06/03/2024

\_\_\_\_\_  
Soraia Lúcia da Silva  
**Coordenador(a) do Curso**