

DISCIPLINA: FGA0084 - Desenvolvimento de Software

TURMA: TA

SEMESTRE: 2021/1

CRÉDITOS: 4

PROFESSOR: Dr. Sergio Antônio Andrade de Freitas
sergiofreitas@unb.br

PLANO DE ENSINO

1. OBJETIVOS DA DISCIPLINA

O objetivo da disciplina é proporcionar uma visão ampla sobre o processo de desenvolvimento de software, incluindo aspectos de programação, metodologias de desenvolvimento, arquiteturas de software, entre outros aspectos da Engenharia de Software. Dessa forma, espera-se que ao final da disciplina o estudante tenha uma visão geral sobre a Engenharia de Software e tenha aplicado alguns dos seus conceitos no desenvolvimento de um produto de software.

2. EMENTA

- | | |
|---|--|
| 1. Práticas de programação em ambientes gráficos e lúdicos; | 3. Introdução à arquitetura de software; |
| 2. Organização e processamento de dados; | 4. Introdução aos conceitos de Engenharia de Software. |

3. HORÁRIO DAS ATIVIDADES SÍNCRONAS E PLATAFORMAS VIRTUAIS

Atividades virtuais síncronas: segundas e sextas, das 16h às 17h50.

Plataforma de aula e de apoio: Microsoft Teams

Link da equipe Teams:

https://teams.microsoft.com/l/team/19%3aZ4Nurkgw3E6_tiXQ07d8KftFTS8DAc56omVLFA8pt5l1%40thread.tacv2/conversations?groupId=d84ded37-572e-4935-ad09-8fa43b4b4f9d&tenantId=ec359ba1-630b-4d2b-b833-c8e6d48f8059

4. METODOLOGIA

A metodologia a ser utilizada é a *Project Based Learning* (PBL). Os estudantes formarão **equipes de 5 a 7 membros** para o desenvolvimento de um produto de *software* ao longo do semestre. A escolha do tema do produto será discutida entre os membros da equipe e o professor. As tecnologias adotadas (tais como linguagem de programação, armazenamento de dados, plataformas de execução etc.) devem ser discutidas e acordadas entre os membros da equipe e o professor. O tamanho do software a ser desenvolvido deve ser compatível com a quantidade de integrantes da equipe.

As equipes devem utilizar ferramentas virtuais para a organização e o acompanhamento das atividades de desenvolvimento do produto. As ferramentas sugeridas são Trello (organização e acompanhamento das atividades do projeto), GitHub (disponibilização do código do produto) e Overleaf (relatório final e apresentação). Outras ferramentas virtuais podem também ser utilizadas, ficando a escolha a critério de cada equipe.

O aprendizado das tecnologias necessárias ao desenvolvimento do projeto é atribuição da equipe e precisa ser incorporado no planejamento das atividades. O desenvolvimento do produto deve seguir as orientações do **método Scrum**.

A atribuição das responsabilidades e atividades dentro do projeto, bem como o acompanhamento do cronograma de entregas do trabalho a ser desenvolvido, deve ser feita pela equipe. É importante que fique claro o papel e a responsabilidade de cada membro dentro do projeto.

Nas atividades síncronas de *Acompanhamento dos Projetos* será reservado um horário para cada equipe (10 minutos). Nos horários agendados, pelo menos um membro da equipe deve comparecer e relatar o andamento do projeto (e.g. avanços, dúvidas, problemas da equipe, entre outros).

Nas atividades síncronas de *Ponto de Controle*, as equipes devem apresentar a evolução do trabalho até aquele momento. A presença de todos os membros da equipe no ponto de controle e na apresentação final é obrigatória, pois constituem momentos avaliativos.

5. CRITÉRIOS DE AVALIAÇÃO

A avaliação da disciplina se dará por meio de:

- **Avaliação do Projeto (NP)** (valor: 0 a 10)

É nota aplicada à equipe de acordo com os entregáveis para cada ponto de controle e da entrega final do produto. É obrigatório que o código da implementação do projeto esteja no GitHub e que o professor seja incluído na equipe no GitHub (sergiofreitas@unb.br). A nota NP é composta pela avaliação de:

- Apresentações realizadas em cada ponto de controle, 20% da nota de NP;
- Relatório técnico final do produto, composto de 5 a 10 páginas (formato IEEE). 30% da nota de NP;
- Apresentação final do produto, 20% da nota de NP;
- Produto final *demonstrado* na apresentação final, 30% da nota de NP.

- **Avaliação Individual (NI)** (valor: 0 a 10)

É a nota relativa ao envolvimento individual de cada membro da equipe (NI) nas diversas etapas de desenvolvimento do projeto. A percepção do professor é aferida:

- na participação do membro nas reuniões da equipe durante os encontros síncronos;
- nas atividades realizadas no GitHub;
- na participação nas apresentações da equipe; e
- na percepção da atuação do membro pelo restante da equipe.

O cálculo da **Nota Final (NF)** é dado por:

$$NF = (NP * 0,8 * NI/10) + (NI*0,2)$$

Para ser aprovado na disciplina o estudante precisa:

- Ter NF maior ou igual a 5,0.

6. CRONOGRAMA DE ATIVIDADES SÍNCRONAS

Semana	Data	Tópicos
1ª	19/07	Apresentação da Disciplina
	23/07	Introdução à Engenharia de Software
2ª	26/07	Atividade assíncrona (Reunião do Colegiado FGA)
	30/07	Introdução ao Método Scrum
3ª	02/08	Introdução ao GitHub
	06/08	Introdução ao Overleaf/LaTeX
4ª	09/08	Definição das Equipes e Temática dos Projetos
	13/08	Planejamento das Atividades do Projeto
5ª	16/08	Ponto de Controle 1
	20/08	Ponto de Controle 1
6ª	23/08	Atividade assíncrona (Reunião do Colegiado FGA)
	27/08	Acompanhamento dos Projetos
7ª	30/08	Acompanhamento dos Projetos
	03/09	Acompanhamento dos Projetos
8ª	06/09	Acompanhamento dos Projetos
	10/09	Acompanhamento dos Projetos
9ª	13/09	Acompanhamento dos Projetos
	17/09	Acompanhamento dos Projetos Ponto de Controle 2
10ª	20/09	Atividade assíncrona (Reunião do Colegiado FGA)
	24/09	Ponto de Controle 2
11ª	27/09	Ponto de Controle 2 <u>Semana Universitária</u>
	01/10	Acompanhamento dos Projetos <u>Semana Universitária</u>
12ª	04/10	Acompanhamento dos Projetos
	08/10	Acompanhamento dos Projetos
13ª	11/10	Acompanhamento dos Projetos
	15/10	Acompanhamento dos Projetos
14ª	18/10	Atividade assíncrona (Reunião do Colegiado FGA)
	22/10	Acompanhamento dos Projetos Apresentação Final dos Projetos
15ª	25/10	Apresentação Final dos Projetos
	29/10	Apresentação Final dos Projetos
16ª	01/11	Apresentação Final dos Projetos Feriado
	05/11	Revisão de Menção

Monitoria

Horário: segunda das 14:00 às 15:30

terça das 15:00 às 16:30

Local: Canal Monitoria da equipe Teams.

7. BIBLIOGRAFIA

BÁSICA:

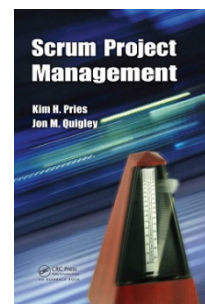
SOMMERVILLE, Ian. **Engenharia de software**. 10. ed. São Paulo: Pearson Education do Brasil, 2018.

<http://minhabcedigital.bce.unb.br/ValidaAcessoEBSCO.aspx?isbn=9788543024974&biblio=pearson>

PRESSMAN, Roger S; MAXIM, Bruce R. **Engenharia de software**. 8. ed. Bookman, 2016.

PRIES, Kim H.; QUIGLEY, Jon M. **Scrum Project Management**, Taylor & Francis Group, 2010.

<https://ebookcentral.proquest.com/lib/univbrasil-ebooks/detail.action?docID=589930>.



COMPLEMENTAR:

JINO, Mario; MALDONADO, José C.; DELAMARO, Márcio E. **Introdução ao Teste de Software**. 2ª. ed. Rio de Janeiro: Elsevier, 2016.

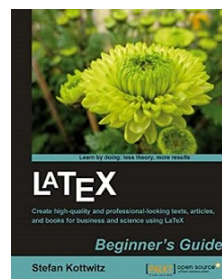
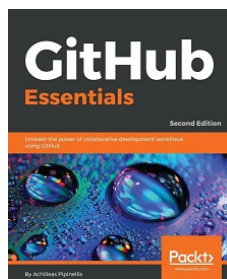
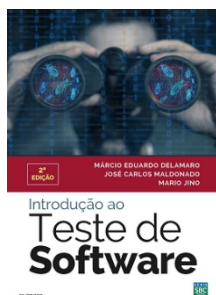
<http://minhabcedigital.bce.unb.br/ValidaAcessoEBSCO.aspx?biblio=minhabiblioteca&isbn=9788595155732>.

SHARMA, Umesh R.; PIPINELLIS, Achilleas. **GitHub Essentials: Unleash the Power of Collaborative Development Workflows Using GitHub**, 2nd Edition, Packet Publishing, Limited, 2018.

<https://ebookcentral.proquest.com/lib/univbrasil-ebooks/detail.action?docID=5446049>.

KOTTWITZ, Stefan. **LaTeX Beginner's Guide**, Packet Publishing, Limited, 2011.

<https://ebookcentral.proquest.com/lib/univbrasil-ebooks/detail.action?docID=943982>.



Observações para acesso virtual:

Livros em minhabcedigital.bce.unb.br – Login com CPF e senha usada na BCE.

Livros em ebookcentral.proquest.com – Configurar proxy na máquina local.