

Engenharia de Software (DCC 603)

Explicação do TP1

Prof. Marco Tulio Valente

mtov@dcc.ufmg.br

Trabalho Prático #1 (20 pontos)

Visão Geral

Implementar uma biblioteca digital de artigos

- Disponibilizar fácil acesso aos artigos publicados em determinados eventos científicos
- Inspirações principais:
 - ACM Digital Library: <https://dl.acm.org>
 - SBC Open-Lib: <https://sol.sbc.org.br>
- Outras inspirações: arxiv.org e dblp.org

Importante: professor será o PO

Histórias de Usuário

1. Como administrador, eu quero cadastrar (editar, deletar) um evento. (Exemplo: Simpósio Brasileiro de Engenharia de Software)
2. Como administrador, eu quero cadastrar (editar, deletar) uma nova edição de um evento (exemplo: edição de 2025 do SBES)
3. Como administrador, eu quero cadastrar (editar, deletar) um artigo manualmente, incluindo seu pdf
4. Como administrador, eu quero cadastrar artigos em massa, a partir de um arquivo bibtex, com dados de vários artigos

Histórias de Usuário

5. Como usuário, eu quero pesquisar por artigos: por título, por autor e por nome de evento
6. Como administrador, eu quero que todo evento tenha uma home page, com suas edições; cada edição, por sua vez, também deve ter uma home page, com seus artigos. (Exemplos: `simple-lib/sbes` e `simple-lib/sbes/2025`)
7. Como usuário, eu quero ter uma home page com meus artigos, organizados por ano (Exemplos: `simple-lib/nome-autor`)
8. Como usuário, eu quero me cadastrar para receber um mail sempre que eu tiver um novo artigo disponibilizado

Dúvidas sobre histórias: preferencialmente, postar
uma msg no fórum da disciplina (suponha que
somos um time remoto)

Tecnologias

- Linguagens, bancos de dados, etc definidas pelos times
- Tecnologias obrigatórias:
 - GitHub
 - Ferramenta de IA

Ferramenta de IA

- Objetivo: ter uma experiência com uma ferramenta de IA de 2a geração; time vai escolher a ferramenta
- 1a geração: Copilot (auto-complete)
- 2a geração: Agentes
 - Principais opções: Cursor, Claude Code
 - Outras opções válidas: Copilot (agentic), VSCode (agentic), Jules, etc.

Observação: possuem versões gratuitas e/ou para estudantes, mas atentar para limitações de tokens (pode ser um critério de escolha)

Quanto da implementação pode ser gerada por IA?

- Entre 0% e 100%
- 0% \Rightarrow justificar porque decidiram não usar, com exemplos concretos. Quais testes fizeram e por que abandonaram?
- Acima de 0%: descrever experiência (pequeno relatório e apresentação)

Importante: todo código gerado por IA deve ser entendido, revisado e aprovado por pelo menos um membro do grupo. Por isso, evitar gerar muito código de uma só vez.

Arquitetura

- Backend: com uma API bem definida + banco de dados
- Frontend: Web

Primeira Tarefa

- Criar um repositório no GitHub e, no README, informar:
 - Membros da equipe e papel (full, backend, frontend)
 - Tecnologias (linguagem, frameworks e BD) + ferramenta IA
- Preencher o seguinte form com a URL do repositório:
 - <https://forms.gle/gdjRtqEzXptnQjFw8>
- Prazo: 4a feira, 27/08, 23:59

Entrega do Trabalho (mais detalhes em breve)

- Apresentação em sala de aula (~15 min)
 - Demo de funcionamento
 - Perguntas do professor sobre código gerado por IA
 - Resumo da visão do grupo sobre uso de IA
- Retrospectiva assíncrona:
 - Via submissão de um Google Forms
 - Incluindo breve relatório sobre uso de IA

Critérios de avaliação (20 pontos)

- Backlog da sprint (1 ponto)
- Implementação das histórias (10 pontos)
- Qualidade da UI (3 pontos)
- Diagramas UML (2 pontos)
- Relatório sobre o uso de IA (3 pontos)
- Retrospectiva (1 ponto)

Backlog da Sprint

- Documentar após o término do sprint, no próprio README do repositório
- Para pelo menos 4 histórias
- Veja um exemplo no próximo slide (para uma história apenas, de um sistema hipotético, que não é aquele do trabalho).

Exemplo de Backlog do Sprint

- **História #1:** Como usuário, eu gostaria de criar uma pergunta
- **Tarefas e responsáveis:**
 - Instalar banco de dados e criar primeiras tabelas [José]
 - Instalar node.js e Express [José]
 - Criar e testar uma primeira rota usando o Express [José]
 - Implementar versão inicial da tela principal [Maria]
 - Implementar no backend a lógica de criar e listar perguntas [José]
 - Adaptar tela principal para incorporar a lista e a criação de perguntas [Maria]