

Instruções para o Trabalho Prático (TP)

1. O trabalho pode ser feito em grupo de até 5 pessoas. A formação dos grupos deve ocorrer até a data da **Aula 08**; quando deve-se informar ao professor os nomes dos integrantes do grupo. Os alunos que não entrarem em nenhum grupo até a data estipulada, farão individualmente o trabalho ou, a critério do professor, serão alocados randomicamente a um novo grupo.
2. O grupo pode escolher um dos temas de trabalho listados no anexo “Lista de Temas” (disponível no site da disciplina). O grupo deve notificar o professor qual o tema escolhido até a data da **Aula 08**. Se o professor não receber notificação até esta data, o grupo será associado a um tema a escolha do professor.
3. Cada tema poderá ser alocado à apenas um grupo. A alocação segue a estratégia FIFO (primeiro email recebido, primeiro grupo alocado). Se o tema escolhido já tiver sido alocado, o grupo deverá escolher outro tema.
4. Se um grupo desejar desenvolver o trabalho sobre um tema que não esteja na lista, o tema deve ser aprovado pelo professor. Neste caso, o grupo deve encaminhar por email ao professor até a data da **Aula 07**: (i) o tema proposto para o trabalho prático, (ii) uma breve descrição do sistema com no máximo 200 palavras e (iii) uma breve justificativa para a escolha do tema com até 200 palavras.
5. Cada grupo deve desenvolver um projeto de software sobre o tema alocado. O desenvolvimento será feito em 3 interações (*Sprints*), como detalhado a seguir.
 - a. *Sprint 1*: Definir um modelo de processo, incluindo as atividades para desenvolvimento do projeto, os responsáveis pelas atividades e um cronograma para execução das atividades. Deve-se ainda criar um documento de especificação de requisitos, definindo os principais requisitos funcionais e não-funcionais.
 - b. *Sprint 2*: Detalhar os requisitos funcionais usando Diagramas de Casos de Uso e descrição dos cenários de casos de uso. Definir um projeto arquitetural, incluindo um Diagrama de Componentes. Fazer o projeto detalhado do sistema, incluindo Diagramas de Classes e outros dois tipos de diagramas definidos na UML.
 - c. *Sprint 3*: Fazer uma implementação parcial do sistema em Java e testes automatizados em JUnit para testar a implementação parcial.
6. Os grupos devem desenvolver o sistema usando não somente o conhecimento adquirido nas aulas, mas também buscar novas fontes, como os livros adotados na disciplina.
7. Os grupos podem utilizar as ferramentas que preferirem (não apenas as demonstradas na disciplina). Entretanto, independente da ferramenta adotada, os documentos e diagramas devem ser entregues em formato PDF. A implementação e os testes (*.JAVA) além um manual de instalação (Readme.md) devem estar disponíveis no GitHub.
8. O uso de inteligência artificial (ex., LLM) é autorizado, mas deve ser explicado o seu uso nos documentos do trabalho. Os integrantes do grupo são os únicos responsáveis pelos artefatos entregues e podem sofrer penalidades por problemas nestes artefatos, incluindo plágio.
9. Cada grupo deverá fazer três apresentações sobre o trabalho, conforme agenda da disciplina. As apresentações parciais (*Sprint 1* e *Sprint 2*) terão duração de **4 a 6 minutos** enquanto a apresentação final (*Sprint 3*) terá duração de **6 a 10 minutos**. Uma demonstração da execução do sistema é desejável na apresentação final.
10. O trabalho final deve ser entregue (exemplo, email ou em mãos) incluindo a URL do projeto no GitHub até a data da **Aula 27**.

Se julgar conveniente, o professor reserva-se no direito de alterar as regras definidas neste documento e notificar os alunos das alterações pelo website da disciplina.