**Trabalho Prático**

O grupo deverá aplicar os conceitos aprendidos na disciplina Projeto de Software em um sistema cujo tema é de livre escolha. Esse sistema pode ser o Projeto de Aplicação de algum membro do grupo, algum sistema legado do estágio de um dos membros do grupo, ou qualquer outro sistema que não tenha documentação de análise e projeto acessível aos membros do grupo.

A professora deverá ser consultada previamente para analisar o grau de dificuldade e a aderência no escopo da disciplina Projeto de Software.

O trabalho consiste em, durante o decorrer do curso, utilizar as técnicas aprendidas para projetar o sistema em questão e produzir os diagramas da fase de projeto para esse sistema.

**Entregas**

Serão realizadas duas entregas:

* **Entrega 1 (Peso 5,0) — Documentação parcial da arquitetura** conforme os modelos e exemplos anexados a atividade do Google Classroom. Este documento deve conter ao menos:
  + Descrição do escopo do sistema (Feito - Willian)
  + Descrição dos requisitos arquiteturais, objetivos e restrições da arquitetura (Feito - Willian)
  + Definição dos padrões arquiteturais adotados (Feito - Willian)
  + Justificativa das decisões arquiteturais adotadas com base nos atributos de qualidade do sistema. Descrição de como a arquitetura de software contribui para os atributos de qualidade importantes do sistema (Feito - Willian)
  + Documentação das visões arquiteturais
    - Ao menos diagramas de casos de usos, modelo conceitual e diagramas de sequência do sistema (Se o sistema for grande, não precisa fazer os DSS dos CRUDs. Mínimo 1 DSS por integrante da equipe) (Feito - Willian/Arthur)
    - Diagrama com a visão geral de **arquitetura** do sistema (organização), incluindo os estilos arquiteturais utilizados e justificativas de uso (Feito - Willian)
  + Diagramas de interação. Pelo menos um diagrama por caso de uso. Escolher a operação do DSS mais complexa. Obs: Descrever o contrato da operação escolhida. (Mínimo 1 diagrama por integrante da equipe) (Feito - Willian)
  + Diagramas de estado ou atividades, quando achar necessário (Feito - Willian)
  + Diagramas de classe detalhado (Feito - Willian)
* **Entrega final (Peso 5) — Documentação final da arquitetura e implementação do sistema.** A documentação da arquitetura deve ser ev oluída considerando:
  + Deve ser descrito como os princípios SOLID e padrões GRASP foram seguidos. Para tanto devem ser usados exemplos de elementos de modelo que foram refatorados devido à identificação de violações desses princípios (Feito - Willian)
  + Devem ser **adotados** pelo menos um padrão **GoF** para cada membro do grupo. (Feito - Willian)
    - Considere a evolução do seu projeto para uma LPS. Projete o modelo de características, listando as características mandatórias, opcionais e alternativas. (Feito - Willian)
    - Código do sistema. Não é necessário implementar todas as funcionalidades do [sistema.](http://sistema.no) O requisito mínimo exigido é a implementação das interfaces dos principais elementos arquiteturais, além dos seguintes casos de uso (i) um cadastro (CRUD) por membro da equipe; (ii) dois casos de uso que descrevem funcionalidades transacionais; e (iii) um caso de uso relativo a uma listagem do sistema (relatório) (iv) implementados ao menos dois padrões GoF.. (Feito - Willian/Arthur)
    - **Obs:** Casos de uso transacionais são casos de uso que representam funcionalidades do sistema que envolvam alguma operação que precise ser completada por inteiro para garantir consistência dos dados, ou seja, ou tudo acontece corretamente, ou, em caso de falha, nenhuma parte da operação deve ser concluída. Por exemplo, “Efetuar Compra”, pode envolver registar um pedido, atualizar um estoque e gerar cobrança. Se algo falhar nesse processo, o sistema deve desfazer as alterações. Já um cadastro (CRUD), não costuma ser considerado transacional nesse sentido.

Implemente o seu sistema observando os seguintes aspectos e tendo as seguintes diretivas em mente:

* + - * Obedeça aos princípios de projeto OO apresentados no curso ou justifique caso seja necessário relaxar algum princípio.
      * Identifique os domínios a que pertencem às classes do sistema e evite classes mistas.
      * Procure desenvolver seu sistema prevendo possíveis extensões e mudanças. Explique os mecanismos arquiteturais que estão sendo utilizados para este fim.
      * Desenvolva uma interface gráfica simples para o sistema, cuidando para não haver acoplamento com o domínio do negócio.
      * Implemente a persistência do sistema usando JDBC, ou um framework de persistência como Hibernate, JPA ou qualquer outro equivalente. Em caso de uso de JDBC, utilize um padrão apropriado para lidar com acesso aos dados na base de dados. (Opcional). Se usar outra tecnologia, explicite o(s) padrão(ões) de projeto que está(ão) sendo usado(s) para acesso a dados
        + **OU** implemente o backend do seu sistema usando um framework OO

Todos os artefatos deverão ser entregues no repositório: <https://classroom.github.com/a/QH9z7FhU>. A entrega no Google Classroom deverá ser realizada por apenas **um** integrante do grupo que informará apenas o link do repositório do grupo.

Documentos de texto devem ser elaborados no Google Docs, com todos os membros logados em suas contas, de forma que seja possível visualizar a colaboração individual de cada integrante. Os links para esses documentos devem estar disponíveis no arquivo *README* do repositório.

**Avaliação**

Para solução serão avaliados o uso adequado dos conceitos de Projeto de Software

1. Artefatos produzidos com base na sua completude, corretude e capacidade de argumentação em relação às decisões tomadas
2. Resultados e colaboração individual no trabalho. Serão considerados participação em sala no desenvolvimento do trabalho, colaboração no GitHub e descrição das responsabilidades do aluno.
3. Apresentação e corretude das respostas individuas durante a apresentação final do trabalho.

**Regras**

O trabalho deve ser feito em grupos **de 4** pessoas. **As responsabilidades de cada aluno deve ser documentada e registrada**

1. Todos os alunos devem apresentar o trabalho **final** no prazo definido. O não cumprimento do prazo invalida o trabalho. Alunos que não participarem da apresentação **ficarão sem nota do trabalho.**

**Cronograma**

1. Formação dos grupos e definição do tema: **05/04/2024**
2. Entrega 1: **19/05/2025**
3. Entrega e apresentação final do trabalho:**07/07/2025**

**Considerações**

1. Os membros da equipe serão avaliados pelo resultado final e pelos resultados individuais alcançados. Assim, numa mesma equipe, um membro pode ficar com nota 9.0 e outro com nota 5.0, por exemplo.
2. Será considerado o nível de dificuldade do projeto na composição da nota final. **Importante consultar a professora para adequação**