

# Trabalho Prático (Etapa 3)

---

*Fazer o projeto detalhado e implementação do sistema de software como um simulador.*

## Atividade como DEVELOPER

A partir da documentação já produzida: (i) documentação dos requisitos através de histórias de usuário ou casos de uso; (ii) modelo de análise; e (iii) projeto arquitetural e tecnologias a serem utilizadas, o grupo DEVELOPER deve realizar o projeto detalhado e implementação de um simulador do sistema, onde seja possível simular as três *features* especificadas. O projeto e a implementação devem ser realizados de acordo com as tarefas planejadas na Etapa 2.

Como parte da Etapa 2, o grupo já deve ter definido as tecnologias a serem utilizadas determinando como dispositivos, interfaces com o usuário, persistência, comunicação distribuída e outros componentes e preocupações serão simulados. Conforme explicado no enunciado da Etapa 2, o simulador deve permitir visualizar o estado do sistema (estado dos dispositivos, interface com o usuário) e fazer alterações no estado para que o comportamento autônomo do sistema possível de ser executado (de forma simulada). Todos os dispositivos e interfaces definidos na especificação do sistema devem fazer parte da simulação. Porém, podem ser feitas simplificações em algoritmos sofisticados. Tais simplificações devem ser explicitadas. Ao final desta etapa, devem ser apresentados o projeto e a implementação, com uma demonstração do simulador em execução.

**ATENÇÃO!** Esta não é uma disciplina de implementação, mas sim de Engenharia de Software. Software codificado sem a definição do projeto arquitetural e detalhado que norteou esta implementação tem pouco valor. O projeto será avaliado através dos **MODELOS** fornecidos em termos dos princípios da Engenharia de Software: modularidade, separação de interesses, baixo acoplamento, alta coesão, etc. Já a implementação deve estar **consistente** com o projeto detalhado, que por sua vez deve estar **consistente** com o projeto arquitetural.

## Tarefa Inicial com Entregas Periódicas

O grupo DEVELOPER deverá considerar as tarefas definidas na Etapa 2 e criar um *Kanban board* na aplicação web KanbanFlow (<https://kanbanflow.com/>) ou alguma ferramenta com propósito similar. Devem ser criadas pelo menos três colunas no *board*: *To-Do*, *Doing* e *Done*. As tarefas devem ser inicialmente designadas aos diferentes membros dos grupos, mas pode haver mudanças ao longo da execução do trabalho (o *board* deve ser atualizado de acordo).

O grupo deverá fazer uma captura de tela do *board* e converter a imagem para PDF em três datas diferentes e submete-las via moodle. As datas estão informadas nas respectivas tarefas do moodle. É esperada a visualização do progresso na execução do trabalho a cada submissão.

## Entrega 3a como DEVELOPER (um por grupo)

*Enviar via moodle um arquivo compactado (formato ZIP) intitulado “GXX – Etapa 3a”, contendo (i) arquivo do projeto da ferramenta CASE escolhida (e.g. Astah, StarUML, etc.) contendo os diagramas desenvolvidos; e (ii) arquivo PDF intitulado “GXX – Projeto Detalhado”, apresentando o conteúdo descrito na seção “Modelo Etapa 3a”.*

## Entrega 3b como DEVELOPER (um por grupo)

*Enviar via moodle um arquivo compactado (formato ZIP) intitulado “GXX – Etapa 3b”, contendo (i) arquivo do projeto da ferramenta CASE escolhida (e.g. Astah, StarUML, etc.) contendo os diagramas*

*desenvolvidos; e (ii) arquivo PDF intitulado “GXX – Projeto Detalhado”, apresentando o conteúdo descrito na seção “Modelo Etapa 3b”.*

### **Entrega 3b INDIVIDUAL (um por aluno)**

*Entregar via moodle o documento com o relatório individual descrevendo sua experiência em um arquivo único no formato PDF.*

Elaborar um documento contendo um relatório individual descrevendo sua experiência ao longo do desenvolvimento do trabalho. Descreva sua experiência como PRODUCT OWNER e DEVELOPER. Apresente quais os principais problemas e dificuldades enfrentados, como foram contornados, qual sua avaliação sobre os resultados obtidos com relação ao seu trabalho/do seu grupo e dos grupos que trabalharam baseados no seu trabalho, as lições aprendidas (o que funcionou, ou que pode ser melhorado).

**Atenção:** sua capacidade de expressão será avaliada, assim como seu posicionamento crítico em relação ao conteúdo da disciplina, e o bom emprego da língua portuguesa.

### **Apresentação e Entrega 3c do Sistema como DEVELOPER (um por grupo)**

*Enviar via moodle um arquivo compactado (formato zip) contendo:*

- Código fonte do sistema desenvolvido
  - Observar o que foi destacado acima a respeito da avaliação do projeto!
- Executável (.exe, .jar, .war, .ear, etc.) do sistema

A apresentação deve mostrar:

- Descrição do produto desenvolvido
- Arquitetura do sistema (estruturação do sistema e tecnologias utilizadas)
- Diagrama de classes (partes principais)
- Diagrama de interação (visão geral, não é necessário descrever cada invocação de método)
- Visão geral do código
- Demonstração da implementação

Tempo de Apresentação: **15min + 5min Q&A**

#### **Observações:**

- A ordem de apresentação será definida por sorteio no início da aula de apresentação. Alunos não presentes no momento de sua apresentação serão descontados.
- Atente ao tempo limite para a apresentação. Caso o tempo se esgote, a apresentação será interrompida e os alunos serão avaliados pelo que foi apresentado.
- Sugere-se fortemente fazer uma apresentação com slides, assim a apresentação tende a ser organizada e facilita o controle do tempo. Também, aconselha-se definir previamente quais membros do grupo apresentarão quais partes do trabalho.
- Verifique com antecedência a adequabilidade dos computadores em sala de aula para a apresentação.

## Modelo Etapa 3a

INF01127 – Engenharia de Software N – Trabalho Prático: Etapa 3a

Grupo XX

**Membros:** Aluno 1 <e-mail aluno 1>, Aluno 2 <e-mail aluno 2>, Aluno 3 <e-mail aluno 3>, e Aluno 4 <e-mail aluno 4>.

**Produto de Software:** Nome do Produto

**Product Owner:** GYY

### 1. Modelo Arquitetural

Refinar a **arquitetura** do sistema já projetada, atendendo modificações sugeridas pela professora na Etapa 2 ou eventuais alterações necessárias a partir da evolução do desenvolvimento. Caso haja alterações, o grupo deve destacá-las através de uma descrição textual. Assim, nesta seção deve haver: diagrama de componentes (imagem legível) refinado e descrição textual de modificações e refinamentos realizados.

### 2. Projeto Detalhado

#### 2.1 Projeto da Interface com o Usuário

Projeto da **interface com o usuário do simulador** (podem ser desenhos das telas, *screenshots*, etc.). O objetivo é compreender o ponto de partida para que o usuário possa interagir com o simulador.

#### 2.2 Diagrama de Classes

Diagrama de **classes** de projeto relativo à semântica da aplicação para a realização da Feature Autônoma 1. O diagrama de classes deve conter classes, atributos (com tipo e demais restrições), operações (com documentação de seu propósito, argumentos e seus detalhes, retorno e seu tipo), associações com sentido de navegação definidos.

**Requisito:** Deve ser utilizado pelo menos um padrão de **projeto GoF**. O mesmo deve estar destacado no projeto detalhado.

#### 2.3 Diagramas de Interação

Diagramas de **interação** mostrando o comportamento do sistema a partir da ocorrência dos eventos que disparam o seu comportamento autônomo da Feature Autônoma 1. É esperado pelo menos um diagrama (podendo ser particionados em mais por questões de legibilidade). Ou seja, não é necessário modelar o comportamento do sistema para detalhar configurações do usuário, como informar parâmetros e *thresholds*. Acrescentar uma descrição explicando como ocorre a colaboração entre os objetos através de troca de mensagens.

**Observação.** Cuidado com a legibilidade dos diagramas entregues no documento PDF e consistência entre o projeto arquitetura, diagrama de classes e diagramas de sequência. Deve ser possível lê-los sem o auxílio de uma lente de aumento.

## Modelo Etapa 3b

Partir do documento feito para a Etapa 3a, e acrescentar as seguintes informações em cada seção:

### INF01127 – Engenharia de Software N – Trabalho Prático: Etapa 3b

Grupo XX

**Membros:** Aluno 1 <e-mail aluno 1>, Aluno 2 <e-mail aluno 2>, Aluno 3 <e-mail aluno 3>, e Aluno 4 <e-mail aluno 4>.

**Produto de Software:** Nome do Produto

**Product Owner:** GYY

#### 1. Modelo Arquitetural

Apresentar novos refinamentos necessários a partir da evolução do desenvolvimento (se houver).

#### 2. Projeto Detalhado

##### 2.1 Projeto da Interface com o Usuário

Apresentar novos refinamentos necessários a partir da evolução do desenvolvimento (se houver).

##### 2.2 Diagrama de Classes

Refinar o diagrama de **classes** de projeto, acrescentando as classes – com atributos (com tipo e demais restrições), operações (com documentação de seu propósito, argumentos e seus detalhes, retorno e seu tipo), associações com sentido de navegação definidos – necessárias para a realização da Feature Autônoma 2 e Feature Configurada pelo Usuário. Obter como resultado o diagrama de classes completo da aplicação.

##### 2.3 Diagramas de Interação

Acrescentar os diagramas de **interação** mostrando o comportamento do sistema a partir da ocorrência dos eventos que disparam o seu comportamento autônomo para a Feature Autônoma 2 e Feature Configurada pelo Usuário. Como resultado são esperados pelo menos três diagramas (podendo ser particionados em mais por questões de legibilidade):

- comportamento autônomo descrito na Feature Autônoma 1 (Etapa 3a);
- comportamento autônomo descrito na Feature Autônoma 2;
- comportamento autônomo configurado pelo usuário através da Feature Configurada pelo Usuário.

Ou seja, não é necessário modelar o comportamento do sistema para detalhar configurações do usuário, como informar parâmetros e *thresholds* das features autônomas ou a realização de casos de uso onde há interação com o usuário para configurar o sistema na Feature Configurada pelo Usuário. Acrescentar uma descrição explicando como ocorre a colaboração entre os objetos através de troca de mensagens para cada uma das features.

**Observação.** Cuidado com a legibilidade dos diagramas entregues no documento PDF e consistência entre o projeto arquitetura, diagrama de classes e diagramas de sequência. Deve ser possível lê-los sem o auxílio de uma lente de aumento.