

Trabalho Prático - Etapa 1: Requisitos, projeto e interface

1. Especificação

Esta seção apresenta as características fundamentais do Trabalho Prático - Etapa 1 relativas à sua especificação.

1.1 Informações gerais

A **Etapa 1** do trabalho consiste em definir funcionalidades requeridas relativas aos **requisitos, projeto e interface** em relação ao modelo de programa definido na Etapa 0 a ser desenvolvido na disciplina. O trabalho deve estar organizado **obrigatoriamente** em, pelo menos, **quatro partes principais**, as quais são **descritas nas próximas seções** (1.2, 1.3, 1.4 e 1.5). Caso o grupo ache importante anexar seções adicionais (por exemplo, introdução, conclusão, possíveis bibliotecas para implementação, etc.), estas devem estar em seções separadas.

1.2 Mudanças em relação à Etapa anterior

Nesta seção deve ser indicado **quais mudanças ocorreram** em relação ao que foi especificado no documento da **etapa anterior**. É importante documentar todas as mudanças para que fiquem registradas, mesmo se já tiverem sido conversadas com o professor. Caso não tenha ocorrido mudanças, basta registrar por escrito que não houve mudanças em relação ao definido na etapa anterior.

Atenção: mesmo que não tenha havido mudanças, esta seção é **obrigatória**.

1.3 Requisitos

Requisitos são **descrições** do que o sistema deve fazer, os **serviços** que oferece e as **restrições** a seu funcionamento. Eles são extremamente importantes de serem adquiridos e refinados pelos analistas de software juntamente com seus clientes, principalmente durante os estágios iniciais do projeto, devendo refletir suas necessidades. Cada **requisito** pode ser **classificado** como **funcional** (relativo a funcionalidade do sistema) ou **não funcional** (relativo a características de performance, usabilidade, cronograma, orçamento, limitações tecnológicas, entre outras). Esta parte do projeto apresenta apenas uma introdução informal sobre este conceito. Mais detalhes sobre a área de engenharia de requisitos serão vistos na disciplina de Engenharia de Software.

Com base nas informações acima, esta parte do trabalho consiste em elaborar um **conjunto de requisitos** que o sistema deve possuir do ponto de vista dos autores do projeto. Cada **requisito** deve ser apresentado em uma **única frase**, devendo ser organizados em **duas listas** conforme seu tipo (**funcional e não funcional**):

- Requisitos **funcionais** devem seguir o código **RF-x**, onde $x \in \{1,2,3,\dots\}$;
- Requisitos **não funcionais** devem ter o código **RNF-x**, onde $x \in \{1,2,3,\dots\}$.

Cada grupo deve definir **pelo menos cinco requisitos funcionais e cinco requisitos não funcionais**, os quais devem ser **ordenados** de forma **crescente** por prioridade a serem seguidas durante o desenvolvimento. Por exemplo, se o grupo julgar que o requisito funcional “Um usuário deve ser capaz de pesquisar as listas de agendamentos para todas as clínicas” deve ter maior prioridade em relação aos demais funcionais, nomeie-o como RF-1. Uma sugestão de formato a ser seguido para esta parte é criar uma tabela conforme exemplo abaixo:

Requisitos Funcionais	Requisitos Não Funcionais
RF-1: Um usuário deve ser capaz de pesquisar as listas de agendamentos para todas as clínicas. RF-2: O sistema deve gerar a cada dia, para cada clínica, a lista dos pacientes para as consultas daquele dia.	RNF-1: O sistema deve verificar diariamente que os executáveis não foram modificados pela adição de código não autorizado. RNF-2: O cadastro de novos usuários no sistema deve ocorrer dentro de um segundo.

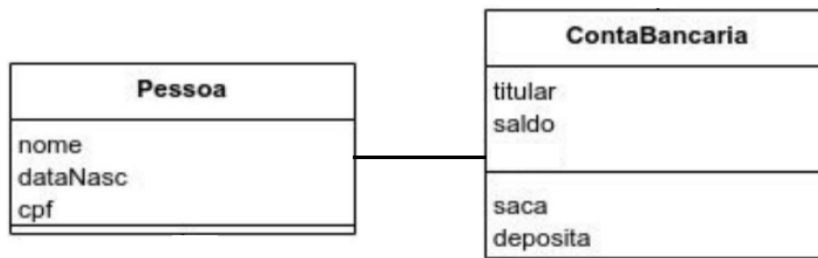
Além disso, é importante escrever um texto sucinto indicando o motivo do grupo ter escolhido esta ordem para os requisitos. Não é necessário justificar a ordem de cada requisito, mas sim comentar de forma geral o que foi priorizado nos aspectos funcionais e não funcionais.

Atenção: para as próximas etapas, não há necessidade de seguir estritamente a prioridade dos requisitos estipulados, mas é importante que haja uma primeira discussão sobre suas prioridades para guiá-los nos próximos passos.

1.4 Projeto

Esta seção deve conter o **projeto da parte funcional** a partir da definição de um **diagrama de classes** seguindo a **perspectiva de implementação**, ou seja, voltada para o time de desenvolvimento. Esta perspectiva **deve conter** informações sobre os **modificadores de acesso, tipos dos atributos, nomes e tipos dos parâmetros dos métodos e tipos de retorno dos métodos** de cada classe. Nesta fase é importante definir **pelo menos oito classes** (possivelmente será interessante definir mais), as quais devem possuir **atributos e métodos**, além de seus **construtores, getters e setters**. Garanta que o projeto contenha **pelo menos** uma **constante** e um **atributo/método** de classe.

Caso as classes do diagrama fiquem pouco legíveis devido à quantidade de informação a ser mostrada dentro do diagrama, é recomendado seguir uma perspectiva de conceito no diagrama (apenas os nomes das classes, dos atributos e dos métodos são mostrados nas classes). Neste caso, as informações completas devem ser detalhadas após o diagrama, organizadas por classes, atributos e métodos. Observe o exemplo abaixo:



Classe Pessoa:

Atributos:

- nome : String
- dataNasc : Date
- cpf : String

Métodos:

Não há*

*Não é visto como boa prática, evite classes apenas com atributos

Classe ContaBancaria:

Atributos:

- titular : String
- saldo : double

Métodos:

- + saca() : double
- + deposita(valor : double) : void

Além disso, é importante indicar todos os **relacionamentos**, contendo **pelo menos** o seu **tipo** (dependência, generalização/especialização, associação, associação-agregação e associação-composição) conforme notação gráfica vista em aula. É obrigatório que seja utilizado **pelo menos um relacionamento de cada tipo** e, em **pelo menos um caso de associação**, sejam detalhados **nome da associação, direção da leitura e suas cardinalidades**.

Após a definição completa do diagrama de classes, é necessário apresentar uma **discussão/explicação** sobre os **motivos** que levaram o grupo a criar o projeto da forma concebida, além de uma **explicação geral** sobre o diagrama de classes. A explicação **não deve** contemplar **detalhes específicos de cada classe** (isto idealmente deve ficar “naturalmente” claro pelos nomes escolhidos das classes, atributos e métodos juntamente com o contexto do programa), a menos que seja algo importante para o entendimento geral.

Atenção: é obrigatória a utilização das definições vistas em aula para o diagrama de classes, para que os trabalhos fiquem padronizados. É importante observar em particular a forma gráfica especificada de cada um dos relacionamentos vistos.

1.5 Interface com o Usuário

Nesta seção, o grupo deve projetar a **interface com o usuário** esperada para o sistema. Para isto, deverá estabelecer **protótipos de imagens** de cada uma das telas, contendo todas as tarefas previstas no sistema. A documentação deverá incluir uma figura de cada tela/unidade de apresentação por página, seguida das **explicações** e **argumentações** que justifiquem seu *layout* e seu comportamento.

Não se preocupe com a “qualidade” dos desenhos/rascunhos; o mais importante é pensar no relacionamento entre as funcionalidades e suas interfaces, buscando que sejam simples/intuitivas para o usuário.

Atenção: não é necessário “implementar” as interfaces (mesmo que sem funcionalidades) nesta etapa do trabalho. O importante é conseguir ilustrar as divisões das telas do sistema com suas funcionalidades previstas, além da organização dos elementos na tela. Para isso podem ser utilizados softwares de design, slides como o Figma, Miro (templates de wireframe), Google Slides, LucidChart (templates de wireframes de UX), etc.

2. Entrega

Este trabalho (Etapa 1) deve ser entregue exclusivamente via commit no GitHub do grupo conforme data combinada pelo professor, como um único arquivo no formato **pdf** com nome **"INF01120_grupoXX_Etapa1.pdf"**, onde **XX** é o **número do grupo** correspondente descrito no moodle da disciplina. **As orientações de como criar o repositório podem ser encontradas no arquivo INF01120 - Trabalho - Etapa 0 - Orientacoes GitHub Classroom.pdf** também disponível no Moodle da disciplina. **Siga as orientações pois o repositório será um fork de um template no classroom.**

Atenção: a entrega demanda um **único arquivo** contendo o relatório no formato pdf com todas as funcionalidades pedidas sobre a Etapa 1. Todo o material deve estar contido apenas no relatório, incluindo todas as figuras geradas. Garanta que as figuras estão em boa resolução e incorporadas adequadamente ao texto, com as respectivas explicações. Adicionalmente, caso queiram, podem adicionar links para os diagramas e interfaces geradas nas ferramentas que vocês usaram como suporte. Mas isso é adicional, o relatório precisa ser autocontido e esses links são complementares.

Bom trabalho!