

Visão Geral do Projeto

Disciplina de Programação Orientada a Objetos

1 Identificação do Grupo

Matrícula	Nome

2 Descrição do Sistema

Incluir um resumo contendo:

- *os objetivos do projeto,*
- *a justificativa pela escolha do projeto*
- *estado atual (caso seja um projeto já em desenvolvimento)*

exemplo:

O objetivo desse projeto é implementar um software para caixas eletrônicos. Corresponde a uma simplificação das aplicações que executam em caixas eletrônicos, onde tal simplificação não considera aspectos como distribuição e controle de concorrência como críticos (em ambientes bancários reais, esses aspectos são extremamente críticos). Por outro lado, tal projeto serve como um bom exercício para o design de um software orientado a objetos com qualidade. O software será desenvolvido do início, uma vez que não existe um projeto em andamento para sua implementação. Por outro lado, apenas um subconjunto das funcionalidades providas por caixas eletrônicos estarão disponíveis.

...

2.1 Requisitos Funcionais

*Apresentar uma lista com os **requisitos funcionais** do sistema, com breve descrição de cada requisito. Nesta breve descrição, caso existam regras negociais aplicáveis, essas regras podem estar descritas aqui.*

exemplo:

Os seguintes requisitos funcionais serão implementados nesse projeto:

- *Criação de contas (cada cliente do banco pode ter apenas uma conta-corrente, mas o sistema suporta diferentes tipos de contas)*
- *Autenticação: clientes do banco acessam o serviço com o identificador da agência, o identificador da conta e uma senha*
- *Realização de transações: solicitar saldo, saques, depósito ou transferência.*
- ...

2.2 Requisitos não Funcionais

*Apresentar uma lista com os **requisitos não funcionais do sistema**, com breve descrição de cada requisito. Nesta breve descrição. Requisitos não funcionais típicos incluem interface com o usuário, interface com sistemas externos, performance na execução das transações, disponibilidade do sistema, etc. Para sermos mais precisos, as definições dos requisitos não funcionais devem ser precisas o suficiente para serem testadas.*

exemplo:

- *O sistema será desenvolvido utilizando interface gráfica baseada em Qt, com o intuito de favorecer a portabilidade da aplicação.*
- *O sistema possui interface com o gerenciador de transações do banco, mas, por questões de flexibilidade, o sistema deve oferecer facilidades para interagir diretamente com o SGBD, no caso deste projeto, será usado o banco de dados SQLite.*
- *O sistema deve suportar um volume de transações da ordem de 1000 TPS (transações por segundo)*
- *O sistema deve apresentar uma disponibilidade mínima de 95% do tempo de uso, ou seja, deve estar disponível pelo menos 22.8 horas por dia (não é tão crítico, podendo estar fora do ar por alguns instantes ao longo do dia, particularmente para a reposição e coleta de valores).*
- ...

3 Plano de entregas

Apresentar um plano para a entrega da implementação dos requisitos do sistema. Cada entrega deve incluir código fonte, casos de teste (escritos com CPPUNIT ou Unit++), documentação usando diagramas UML, Doxygen (<http://www.doxygen.org>) e/ou Cartões CRC), documentação com a visão geral da arquitetura (particularmente na primeira entrega) e análise de cobertura dos testes.

3.1 Primeira entrega (24/01/2012)

exemplo:

A primeira entrega, a ser realizada no dia (24/01/2012), contemplará:

- a) Implementação das classes com a representação do domínio (cliente, agência, conta, transação, etc.)*
- b) Implementação dos requisitos funcionais: autenticação, saldo e saque*
- c) Implementação da integração com o banco de dados, necessária para atender o item (b)*
- d) Implementação da interface gráfica necessária para os requisitos funcionais listados no item (b)*
- e) Implementação de uma interface para simular o mecanismo de hardware que seleciona e disponibiliza cédulas em uma operação de saque.*

3.2 Segunda entrega (14/02/2012)

exemplo:

A segunda entrega, a ser realizada no dia (14/02/2012), contemplará:

- a) Revisão das classes com a representação do domínio (cliente, agência, conta, transação, etc.)*
- b) Implementação dos requisitos funcionais: depósito e transferência*
- c) Implementação da integração com o banco de dados, necessária para atender o item (b)*
- d) Implementação da interface gráfica necessária para os requisitos funcionais listados no item (b)*