

A photograph of a modern architectural complex featuring several curved glass buildings with horizontal bands. In the foreground, there is a body of water with a small fountain. The sky is clear and blue.

APLICAÇÃO DE GERÊNCIA DE OBJETOS COMPARTILHÁVEIS DE UM SGBD ORACLE

Acadêmico: Antonio de Oliveira Junior

Orientador: Prof. Alexander R. Valdameri

Roteiro

- Introdução
- Objetivos
- Motivação
- Fundamentação
- Desenvolvimento
- Conclusões
- Extensões
- Relevância pessoal

Introdução

- Aplicações corporativas (grande porte);
- Fábrica de software;
- Objetos compartilháveis Oracle;
- Complexidade da gerência de objetos.



Objetivos

- permitir liberação de objetos;
- promover agendamento de liberações;
- garantir liberações de objetos do banco de dados sem erros;
- viabilizar um retorno imediato caso ocorrer insucesso na liberação de objetos.

Motivação

- impacto de liberações em ambientes corporativos;
- dependência de pessoal capacitado para identificar problemas ou resolvê-los durante o processo de liberação de objetos

Especificação

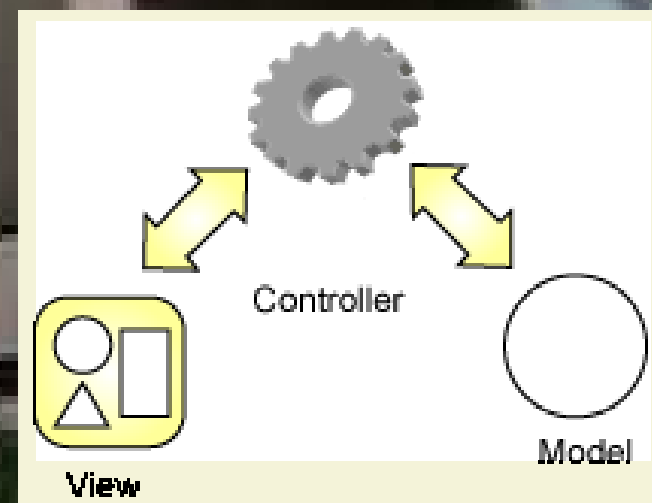
- UML
- Diagramas

Objetos compartilháveis

- Foco no SGBD Oracle;
- Contidos internamente ao SGBD;
- São exemplos:
 - Trigger;
 - Procedure;
 - Function;

Model-View-Controller

- modelo usado para desenvolvimento de software
- camada de dados
- camada de visualização
- camada de controle



Casos de Uso

pd Diagrama de pacotes

PCT02 - Módulo Desenvolvedor

- + Desenvolvedor
- + UC02.01 - Efetua login
- + UC02.02 - Consulta informações de liberação
- + UC02.03 - Manutenção e criação de código com controle de compartilhamento
- + UC02.04 - Valida objetos em banco de dados
- + UC02.05 - Efetua retomo de liberação
- + UC02.06 - Agendamento de liberações

PCT03 - Módulo Administrador

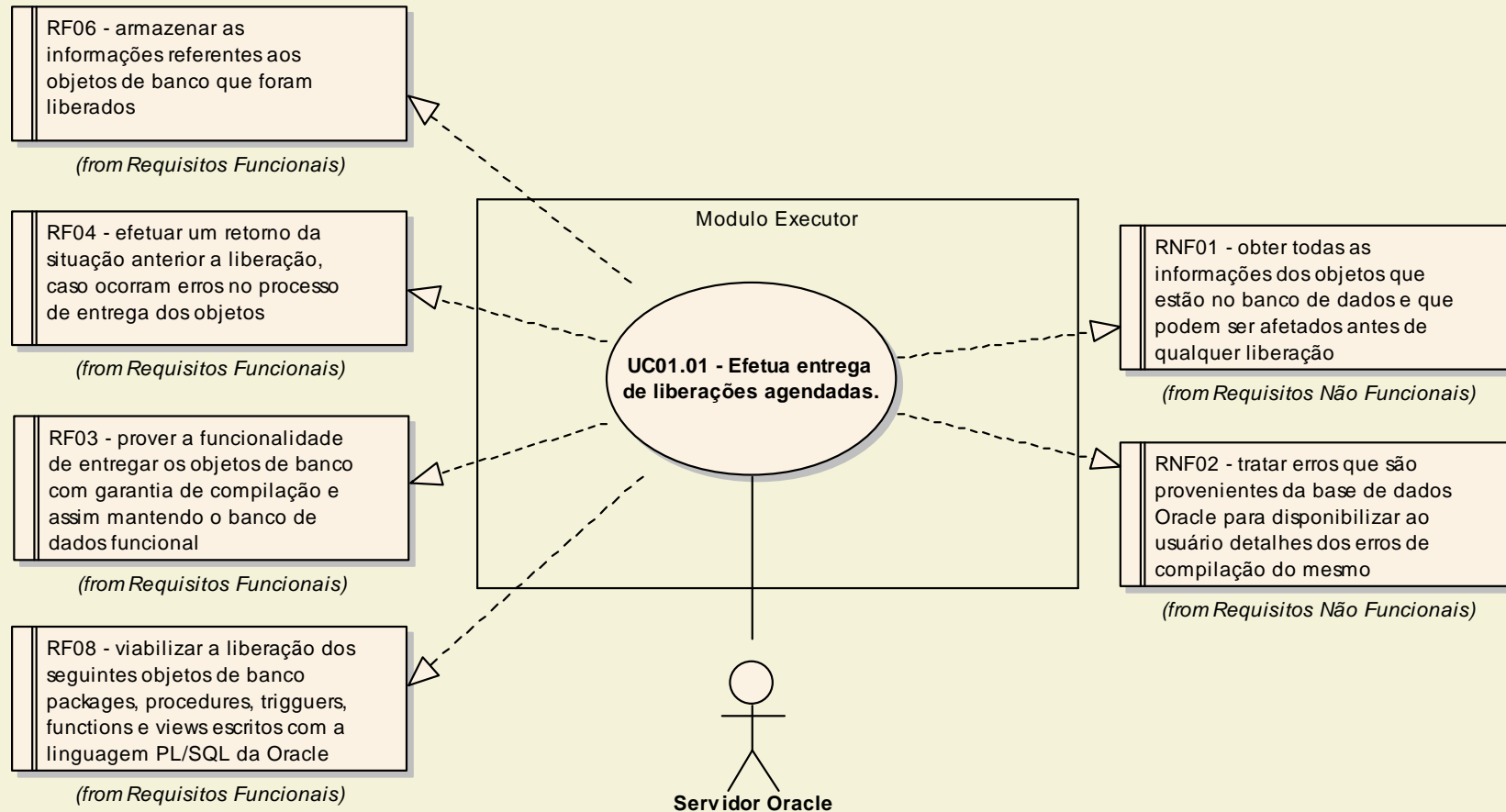
- + Administrador
- + UC03.01 - Manutenção de usuários
- + UC03.02 - Criação do repositório
- + UC03.03 - Desaloca fonte alocado pelo usuário
- + UC03.04 - Manutenção de dados para conexão

PCT01 - Módulo Executor

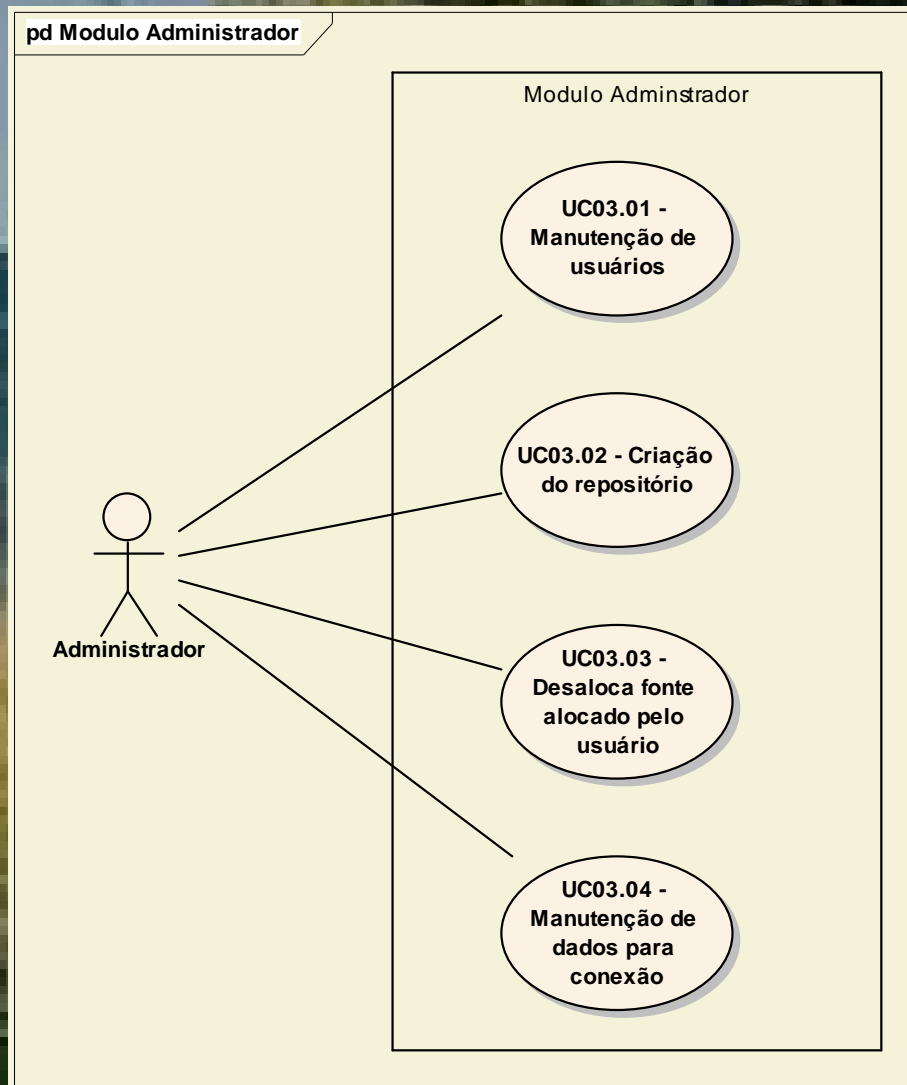
- + Servidor Oracle
- + UC01.01 - Efetua entrega de liberações agendadas.

Casos de Uso

pd Módulo Executor



Casos de Uso



Casos de Uso

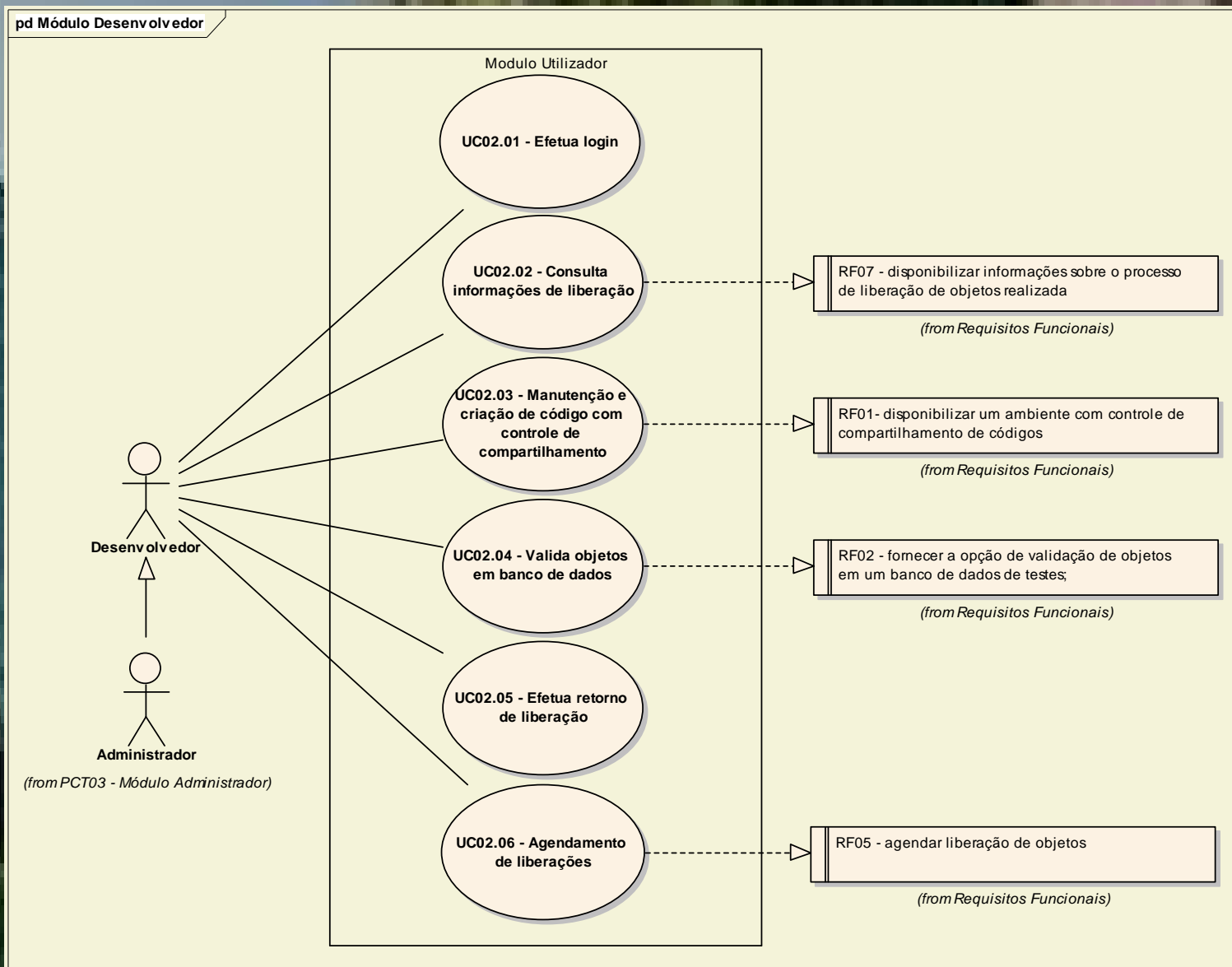


Diagrama de Classes

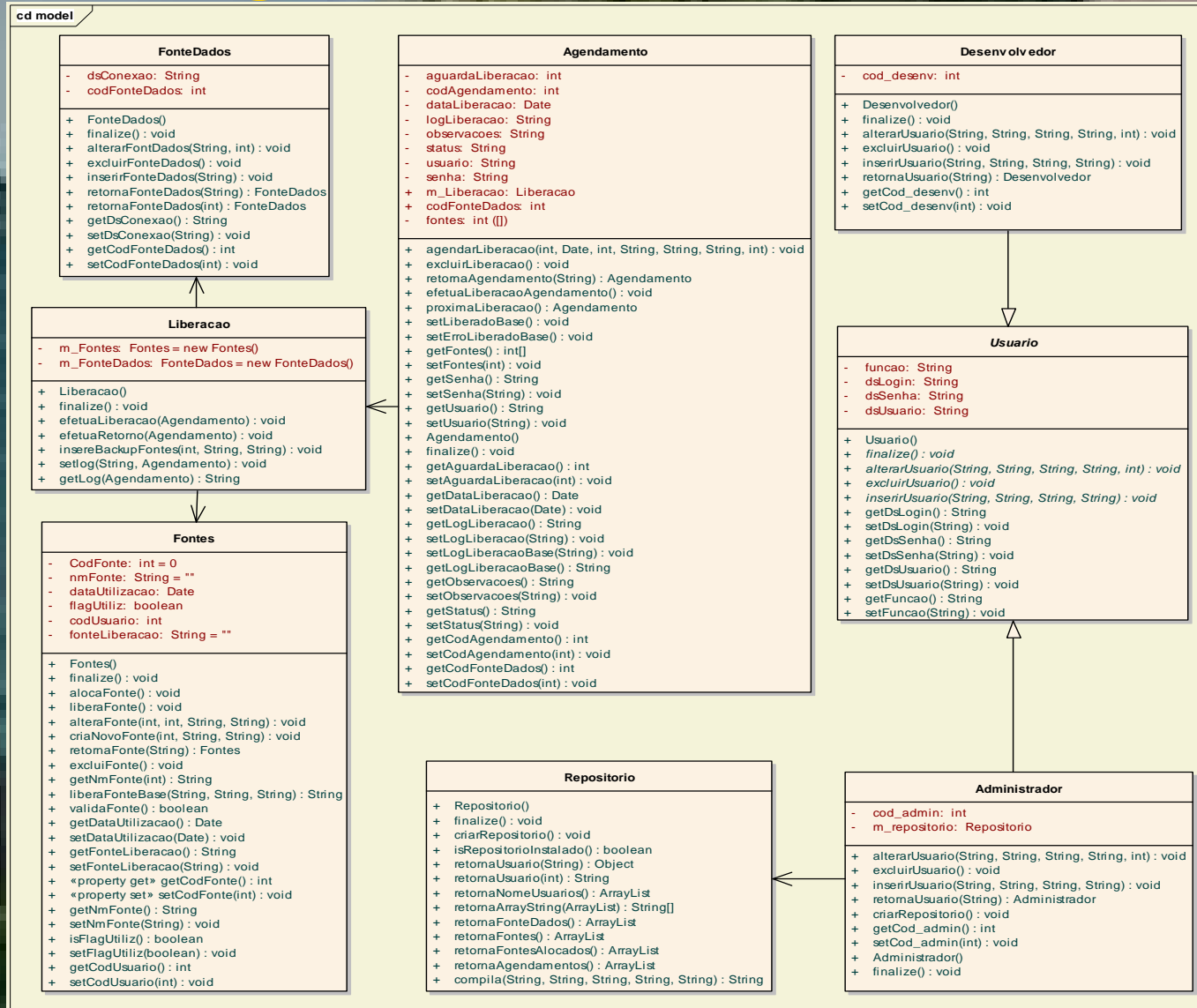


Diagrama de Classes

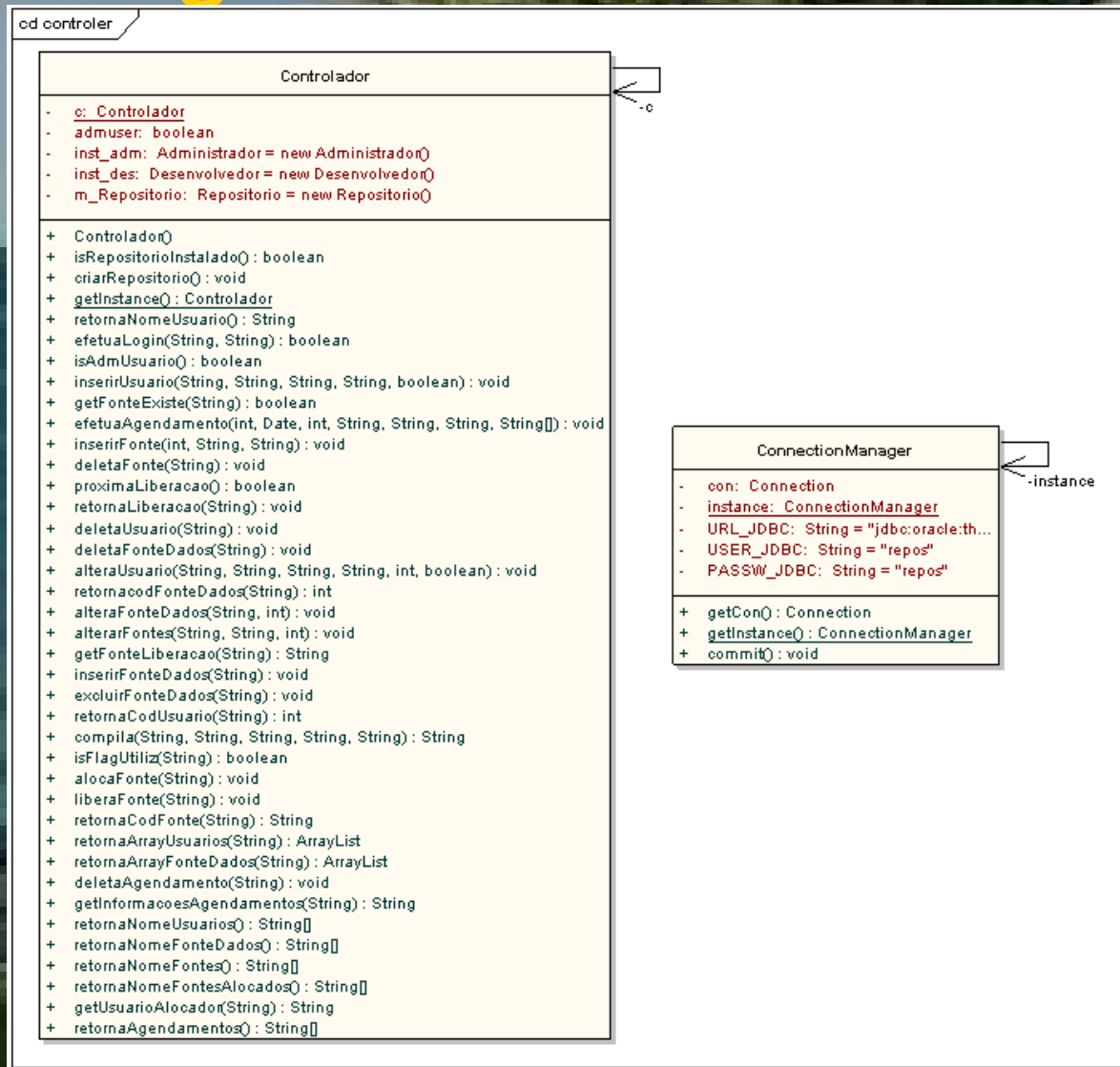
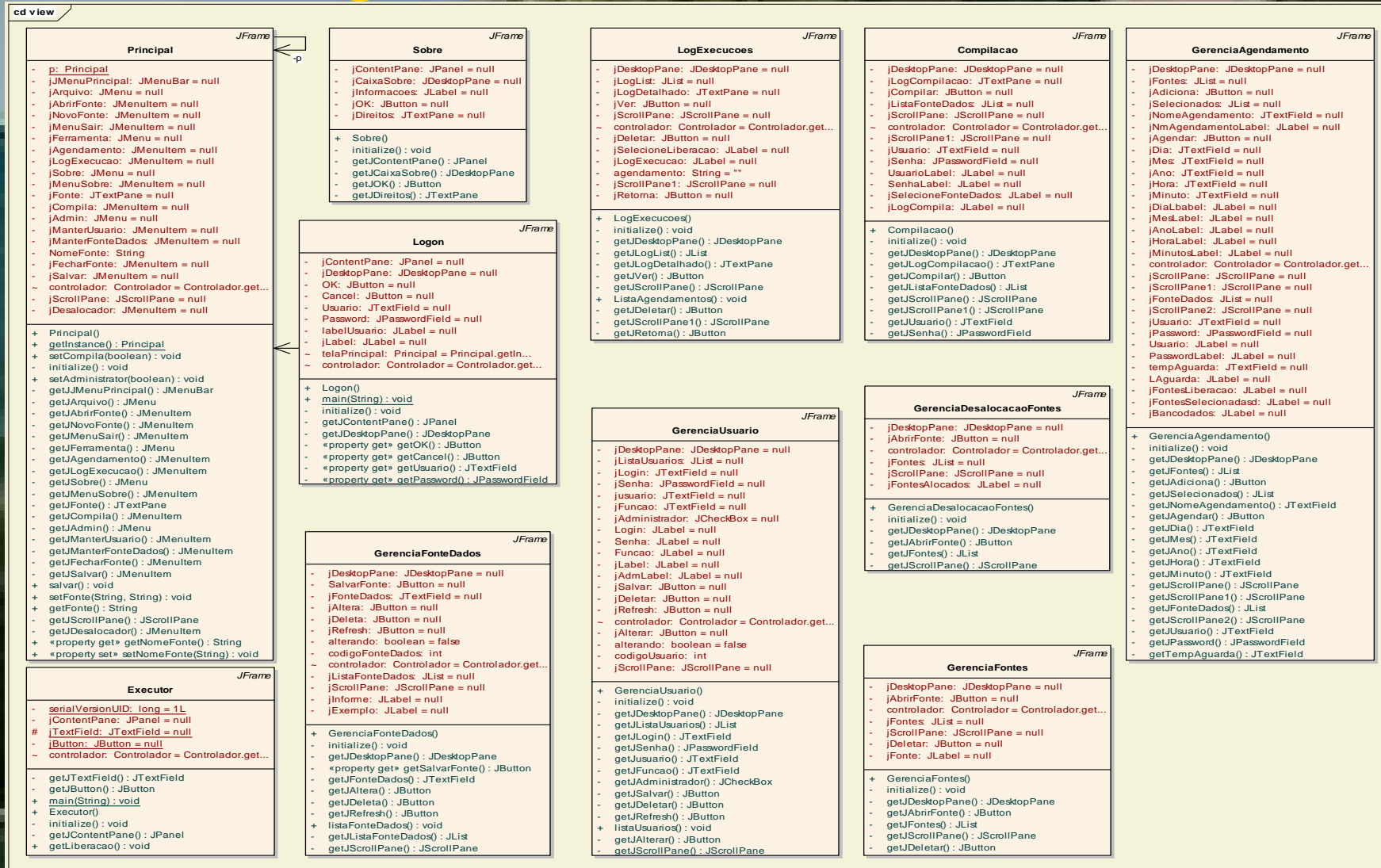


Diagrama de Classes



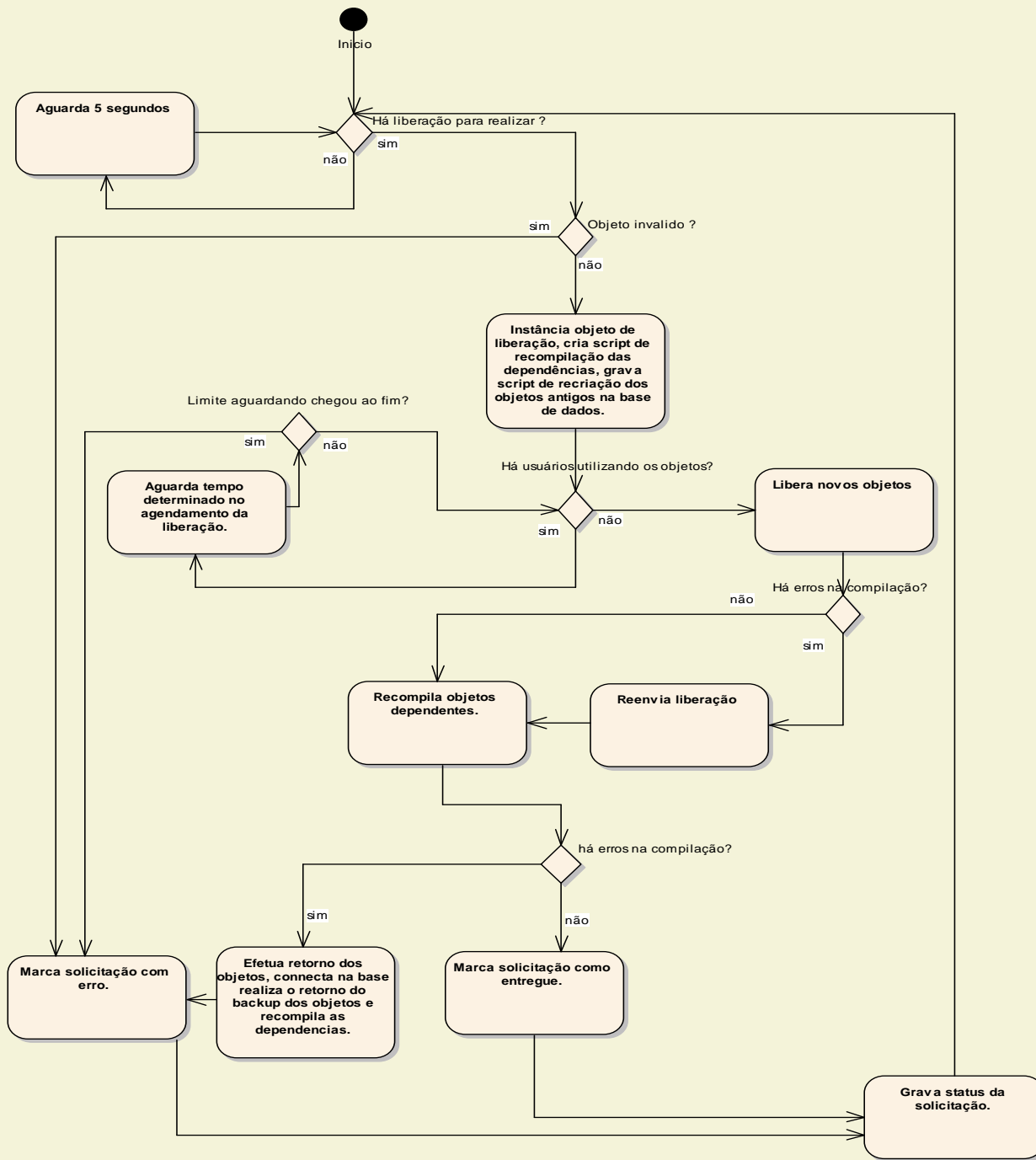
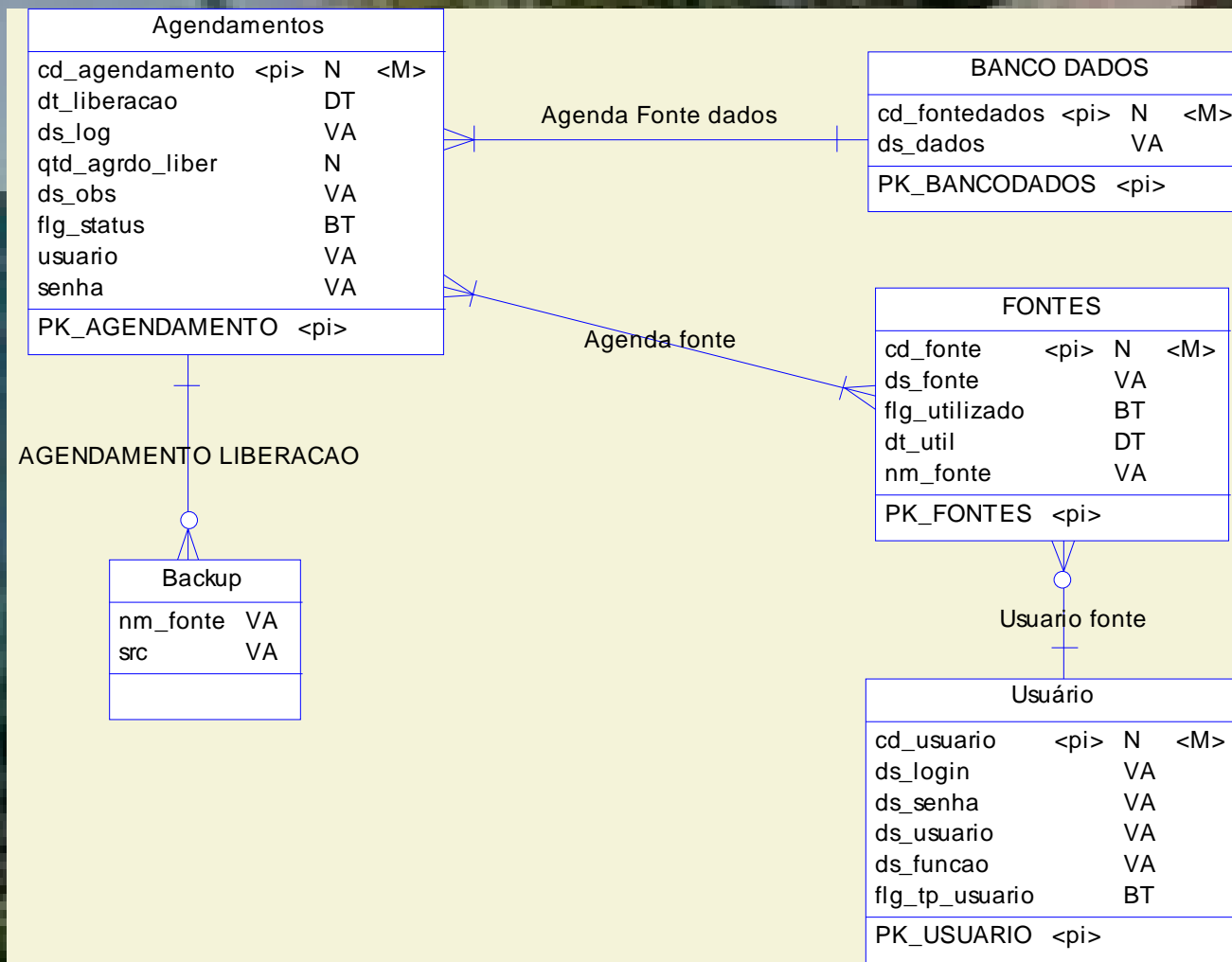


Diagrama de Atividades



Modelo de Entidade Relacionamento



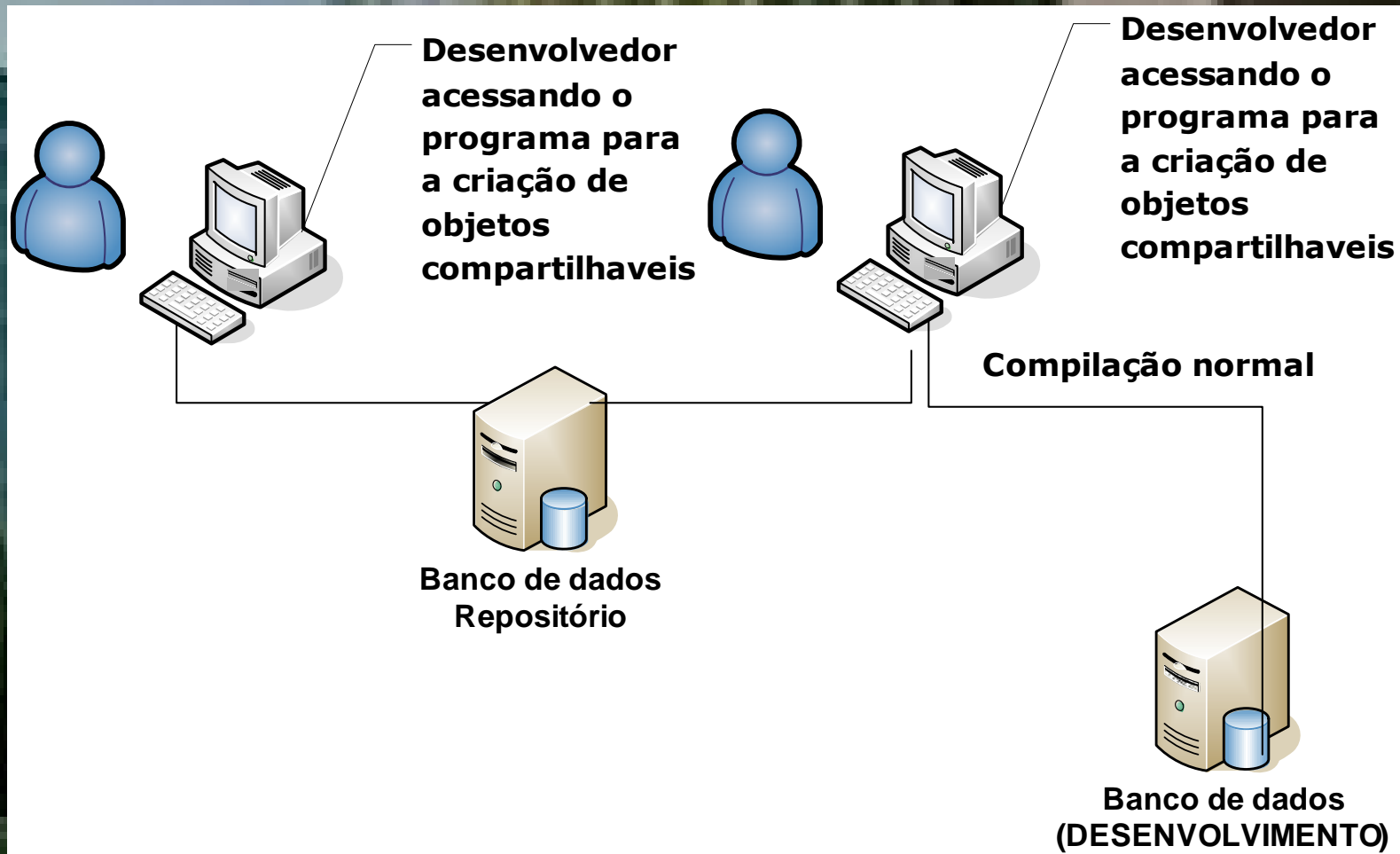
Método do controlador

```
/**
 * Efetua o Logon da aplicação verificando de acordo com os objetos.
 * @throws SQLException
 * @throws ClassNotFoundException
 * @throws IOException
 */
public boolean efetuaLogin(String p_login, String p_senha) throws SQLException, ClassNotFoundException, IOException {

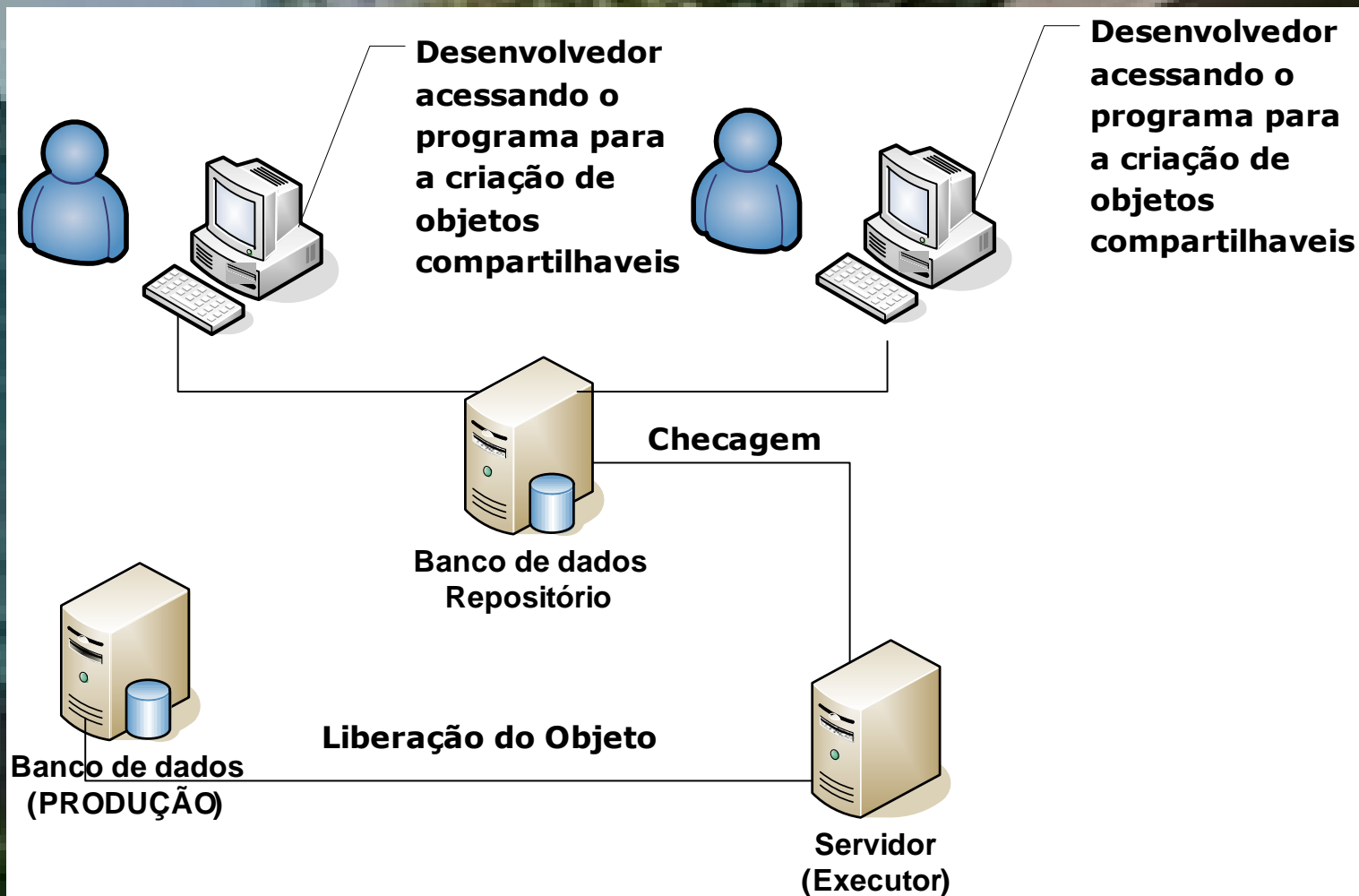
    boolean wrk_valida = false;

    if (m_Repositorio.retornaUsuario(p_login) instanceof Desenvolvedor) {
        Desenvolvedor instancia = (Desenvolvedor) m_Repositorio.retornaUsuario(p_login);
        // Não considera usuario como administrador.
        this.admuser = false;
        // Variavel local recebe instancia
        this.inst_des = instancia;
        if (p_senha.equals(instancia.getDsSenha())){
            wrk_valida = true;
        }
    }else{
        Administrador instancia = (Administrador) m_Repositorio.retornaUsuario( p_login);
        // Considera usuario como administrador.
        this.admuser = true;
        // Variavel local recebe instancia
        this.inst_adm = instancia;
        if (p_senha.equals(instancia.getDsSenha())){
            wrk_valida = true;
        }
    }
    return wrk_valida;
}
```

Esquema de Funcionamento



Esquema de Funcionamento



Compilação de código fonte

Principal

Arquivo Admin Ferramentas Sobre

```
create or replace procedure teste is  
  
  wrk_aux_caracter_ant number(3);  
  wrk_aux_caracter_dep number(3);  
  wrk_aux_igual        number(3);  
  wrk_nro_chando       number(8);  
  wrk_length_caracter  number(8);  
  wrk_linha            varchar2(1024);  
  wrk_valor_quest      varchar2(20);  
  wrk_valor_resp       varchar2(20);  
  new_body_email       varchar2(20);  
  
begin  
  wrk_nro_chando := 0;  
  wrk_aux_caracter_ant := 1;  
  wrk_aux_caracter_dep := 1;  
  wrk_aux_igual := 1;  
  new_body_email := 'nrocham';  
  
  while wrk_aux_caracter_dep <= wrk_aux_caracter_ant :=  
  
    if ((INSTR(NEW_BODY_EMAIL, wrk_aux_caracter_ant) > 0)  
        and wrk_aux_caracter_ant <= wrk_aux_caracter_dep)  
      wrk_aux_caracter_ant := wrk_aux_caracter_ant + 1;  
    else  
      if wrk_aux_caracter_ant <= wrk_aux_caracter_dep  
        wrk_aux_caracter_ant := wrk_aux_caracter_ant + 1;  
      else  
        wrk_aux_caracter_dep := INSTR(NEW_BODY_EMAIL,  
                                       CHR(13) || CHR(10),  
                                       wrk_aux_caracter_ant,  
                                       1);  
      end if;  
    end if;  
  
end if;
```

Compilação

Selecione o banco de dados para compilar:

- jdbc:oracle:thin:@10.1.1.3:1521:dbtcc
- jdbc:oracle:thin:@localhost:1521:dbtcc

Detalhes da compilação:

Objeto compilado !!!

Usuario: teste Senha: *****

Compilar

Agendamento

Agendamento

Selecione um fonte

TESTE
TESTE2

Fontes selecionados

TESTE

Selecione o banco de dados para a liberação

jdbc:oracle:thin:@10.1.1.3:1521:dbtcc
jdbc:oracle:thin:@localhost:1521:dbtcc

Adiciona >>

Usuario: tcc

Senha: ***

Tempo aguardo liberacao 1

Referencia do Agendamento: Dia: 25 Mes: 06 Ano: 2006 Hora: 10 Minuto: 14

Liberação 25062006 10-00 - TESTE

Agendar

Informações de liberação

Principal

Arquivo Admin Ferramentas Sobre

Log Execuções

Selecione um agendamento:

Liberação 18062006 10-00 -TESTE

Refresh Deletar Retorno

Informações da liberacao:

Usuario do banco: tcc
Data da liberacao: Sun Jun 18 10:00:00 PDT 2006
Status liberacao: L
Fontes :
TESTE
Log:
Conectado na base jdbc:oracle:thin:@127.0.0.1:1521:dbtcc
Nome dos fontes a liberar:
TESTE
Objeto dependente:[TESTE2]
Realizando backup:[TESTE]
Objeto não esta sendo utilizado efetuando liberação.
Fonte TESTE Liberado !

Conclusões

- O objetivo de criar um software para liberação foi realizado com sucesso.
- Não é mais necessário requisitar auxílio a um AD para realizar a liberação dos objetos.
- Com o controle de alocação de código ficou fácil gerenciar o desenvolvimento.

Extensões

- adquirir códigos fontes do banco de dados, importando os códigos no repositório já existente;
- um identador para os códigos fontes que estão sendo criados, com base na BNF do *PI/Sql*, tornando assim os códigos identados.

Relevância pessoal

- Consolidação dos conhecimentos em JAVA;
- Aprendizado do padrão MVC utilizado no desenvolvimento do projeto;
- Desenvolvimento de um programa utilizando linguagem orientada a objetos;