

INSTITUTO INFNET

Henrique Zwicker Galvão de Moura

ASSESSMENT

Rio de Janeiro
2016

Henrique Zwicker Galvão de Moura

ASSESSMENT

Professor(a): Pier Taranti .

Disciplina: Orientação a Objetos com UML e Modelagem de Dados.

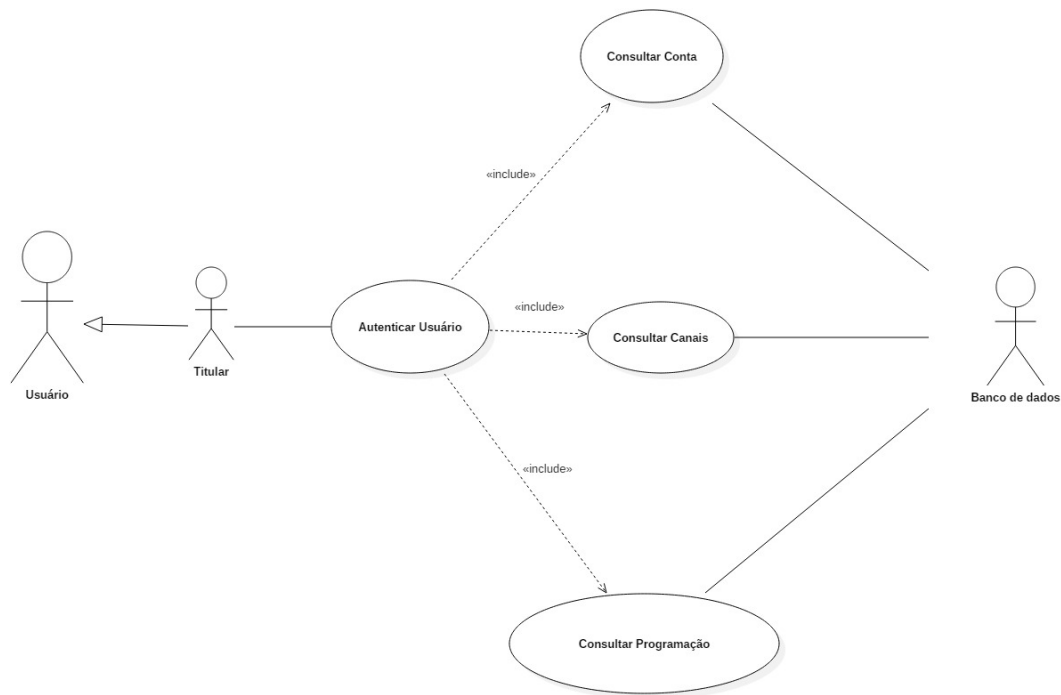
Turma: Análise e desenvolvimento de Sistemas - Noite

Rio de Janeiro
2016

Sumário

1.Diagrama de caso de uso	4
Visão do Diagrama de Caso de Uso	4
UC01 – Autenticar Usuário	4
UC02 – Consultar Conta	5
UC03 – Consultar Canais	5
UC04 – Consultar Programação	6
2.Diagrama de Classes	7
Visão do Diagrama de Classes	7
Classes do sistema	7
3.Diagrama de Sequência.....	8
Visão do Diagrama de Sequência	9
4.Diagrama de Estados	9
Visão do Diagrama de Estados.....	9
5.Diagrama de Atividades	10
Fluxo de controle sequencial	10
Fluxo de controle paralelo.....	11
Visão do Diagrama de Atividades.....	11
6.Diagrama de Pacotes	12
Visão do Diagrama de Pacotes.....	12
8. Diagramas de entidade x relacionamento.....	12
Modelo Conceitual.....	12
Visão do Diagrama de entidade x relacionamento.	13
Modelo Lógico.....	13
Modelo Físico	13
Mapeamento objeto relacional	15
9. REFERÊNCIAS.....	16

1. Diagrama de caso de uso



Visão do Diagrama de Caso de Uso

A visão usada neste diagrama foi do tipo lógica e de caso de uso.

UC01 – Autenticar Usuário

- **Ator Principal:** Titular
- **Objetivo:** O titular autenticar no sistema.
- **Pré-Condições:** Ser usuário do banco de dados do sistema.

Cenário de fluxo básico (CFB):

1. Usuário invoca programa.
2. Sistema solicita dados de autenticação de usuário.
3. Usuário insere dados de login.
4. Sistema valida dados inseridos.
5. Sistema autoriza acesso do usuário.

Cenário de fluxo alternativo (CFA):

1CFA. Sistema não valida dados do usuário: (CFB -4)

1. Sistema mostra mensagem de erro.
2. volta para CFB-3.

UC02 – Consultar Conta

- **Ator Principal:** Titular

- **Objetivo:** O titular consultar sua conta.

- **Pré-Condições:** Ser titular do banco de dados do sistema.

Cenário de fluxo básico (CFB):

1. Usuário seleciona a opção do menu “Consultar dados de cliente”.
2. Sistema informa dados da conta.
3. Sistema informa opções de editar conta ou visualizar conta.
4. Usuário seleciona opção desejada.
5. Sistema fornece opção selecionada.
6. Sistema volta ao menu.

Cenário de fluxo alternativo (CFA):

1CFA. Sistema não consegue informar dados da conta: (CFB – 3)

1. Sistema mostra mensagem de erro.

UC03 – Consultar Canais

- **Ator Principal:** Titular

- **Objetivo:** O titular consultar canais do seu plano.

- **Pré-Condições:** Ser titular do banco de dados do sistema.

Cenário de fluxo básico (CFB):

1. Usuário seleciona opção do menu “Consultar canais no plano”.

2. Sistema pesquisa e valida se a conta está em adimplência.
3. Sistema questiona qual plano.
 - 3.1 Plano Combo Ilimitado.
 - 3.2 Plano econômico.
3. Usuário seleciona opção desejada.
4. Sistema informa o plano selecionado e seus respectivos canais.
5. Sistema volta ao menu.

Cenário de fluxo alternativo (CFA):

- 1CFA. Sistema não consegue validar conta: (CFB – 2)
1. Sistema mostra mensagem sugerindo planos para contrato.
 2. Sistema mostra mensagem de contato ao suporte.
 3. Sistema encerra.

UC04 – Consultar Programação

- **Ator Principal:** Titular
- **Objetivo:** O titular consultar programação de um canal.
- **Pré-Condições:** Ser titular do banco de dados do sistema.

Cenário de fluxo básico (CFB):

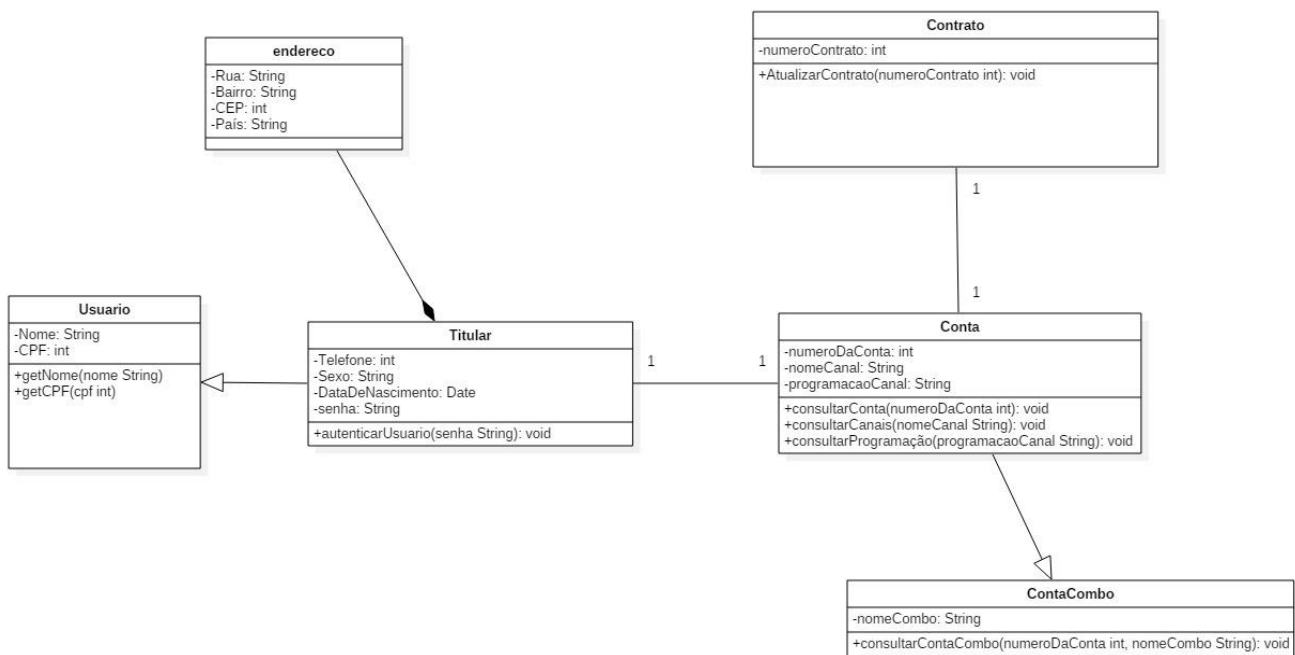
1. Usuário seleciona opção do menu “Consultar programas de um canal”.
2. Sistema informa os canais.
3. Usuário seleciona canal desejado.
4. Sistema informa programação do canal desejado.
5. Sistema informa opções de ver outros canais ou voltar ao menu.
6. Usuário seleciona opção desejada.
7. Sistema fornece opção selecionada.
8. Sistema volta ao menu.

Cenário de fluxo alternativo (CFA):

1CFA.Sistema não consegue encontrar uma programação no canal escolhido: (CFB-4)

1. Sistema mostra mensagem de contato ao suporte.
2. Volta para CFB – 2.

2.Diagrama de Classes



Visão do Diagrama de Classes

A visão usada neste diagrama foi do tipo visão lógica.

Classes do sistema

As classes do sistema de sistema de gerenciamento e venda de canais TV a cabo são:

Classe Usuário: É a classe pai da Classe Titular. Possui atributos privados como Nome, CPF. Os métodos são getNome, getCPF, com a finalidade de conseguir informações do usuário.

Classe Titular: É uma classe filha da Classe Usuário, ou seja, herda todos os atributos e métodos da classe usuário. Possui atributos privados como telefone, sexo e data de nascimento .

Classe Endereco: é uma Classe que representa a composição de atributos com a Classe Titular.

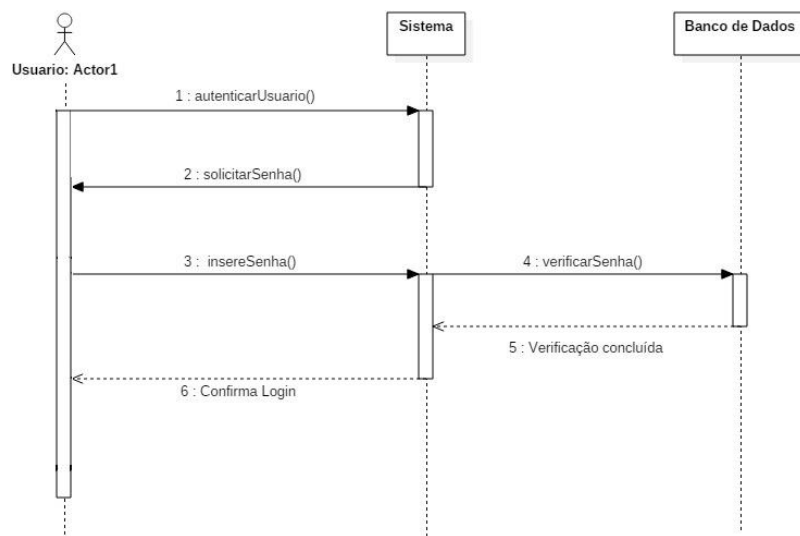
Classe Conta: É classe pai da Classe. Tem como atributos privados o Número da conta, nomeCanal programacaoCanal. Possui ainda métodos públicos como, consultarConta, consultarCanais, consultarProgramação.

Classe ContaCombo: É a classe filha de Classe Conta. Possui todos os atributos e métodos da sua classe pai e tem como atributo privados o NomeCombo Apresenta método público próprio de consultarCanaisCombo.

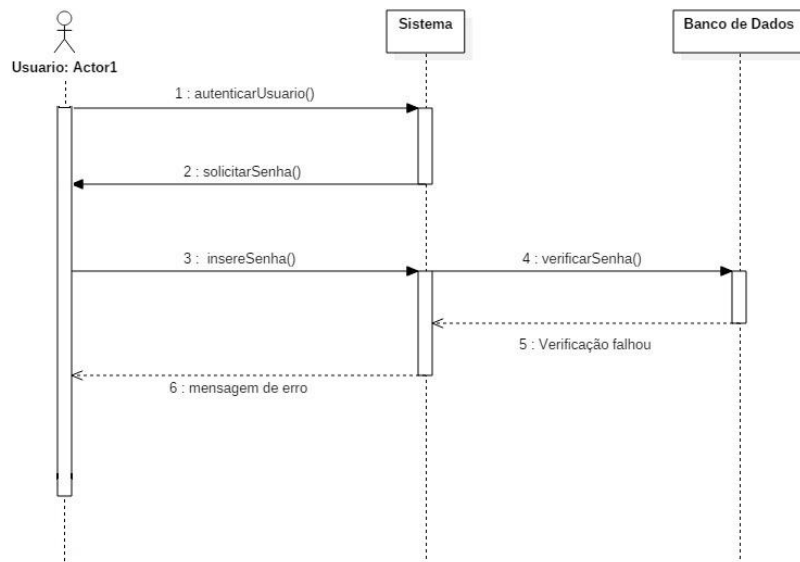
3.Diagrama de Sequência

Diagramas de sequências (Principal e Alternativo) referente ao caso de uso autenticar Usuário:

interaction SequenceDiagram1



interaction SequenceDiagram1

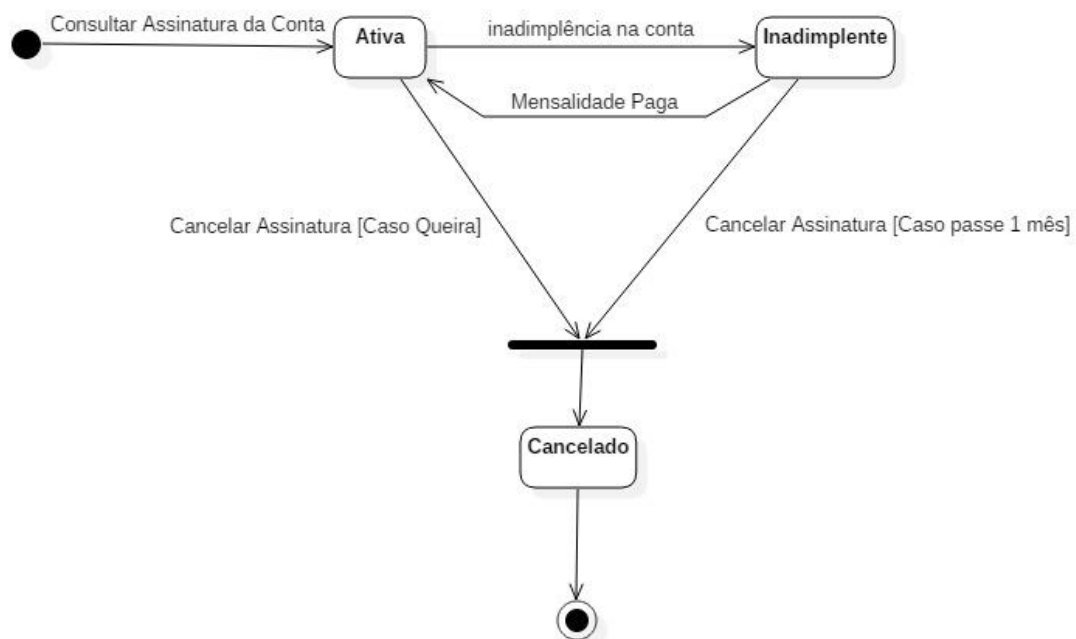


Visão do Diagrama de Sequência

A visão usada neste diagrama foi do tipo lógica .

4.Diagrama de Estados

Diagrama de Estados referente ao estado de assinatura da conta do Usuário:



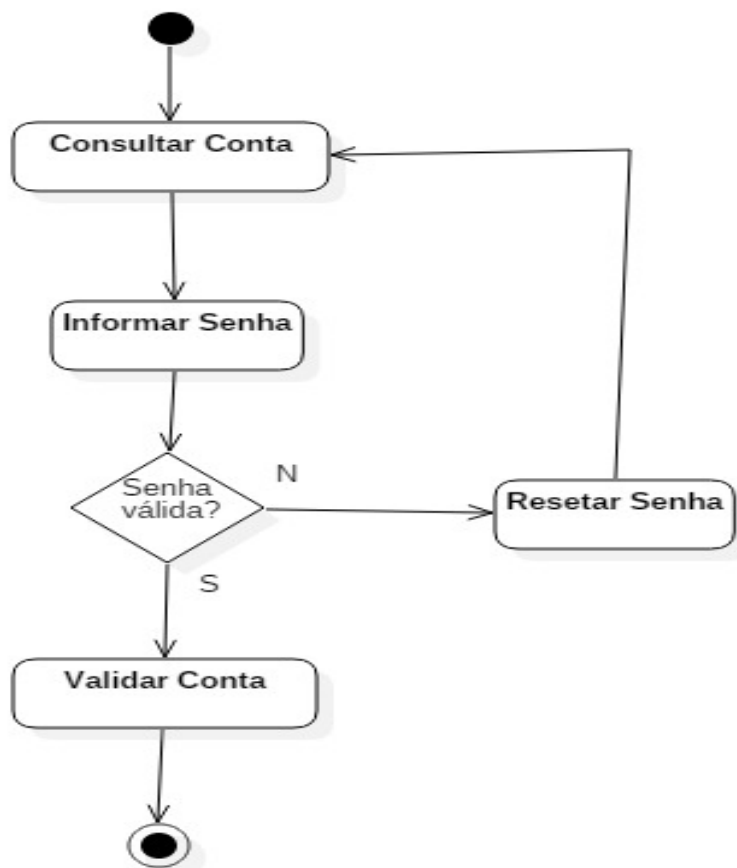
Visão do Diagrama de Estados

A visão usada neste diagrama foi do tipo lógica.

5. Diagrama de Atividades

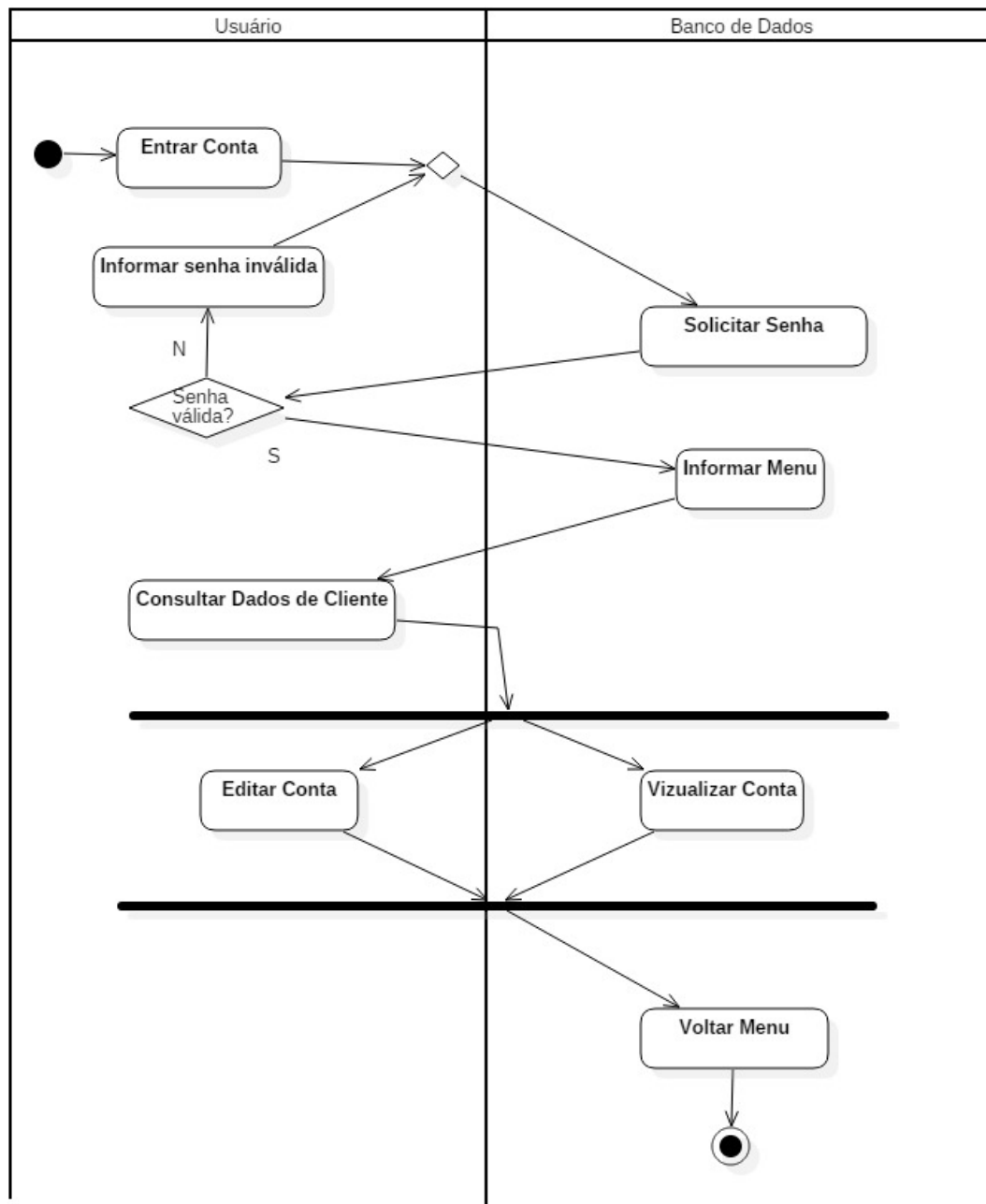
Fluxo de controle sequencial

Diagrama é referente a autenticação da conta do usuário:



Fluxo de controle paralelo

Diagrama referente a consulta de dados do cliente:

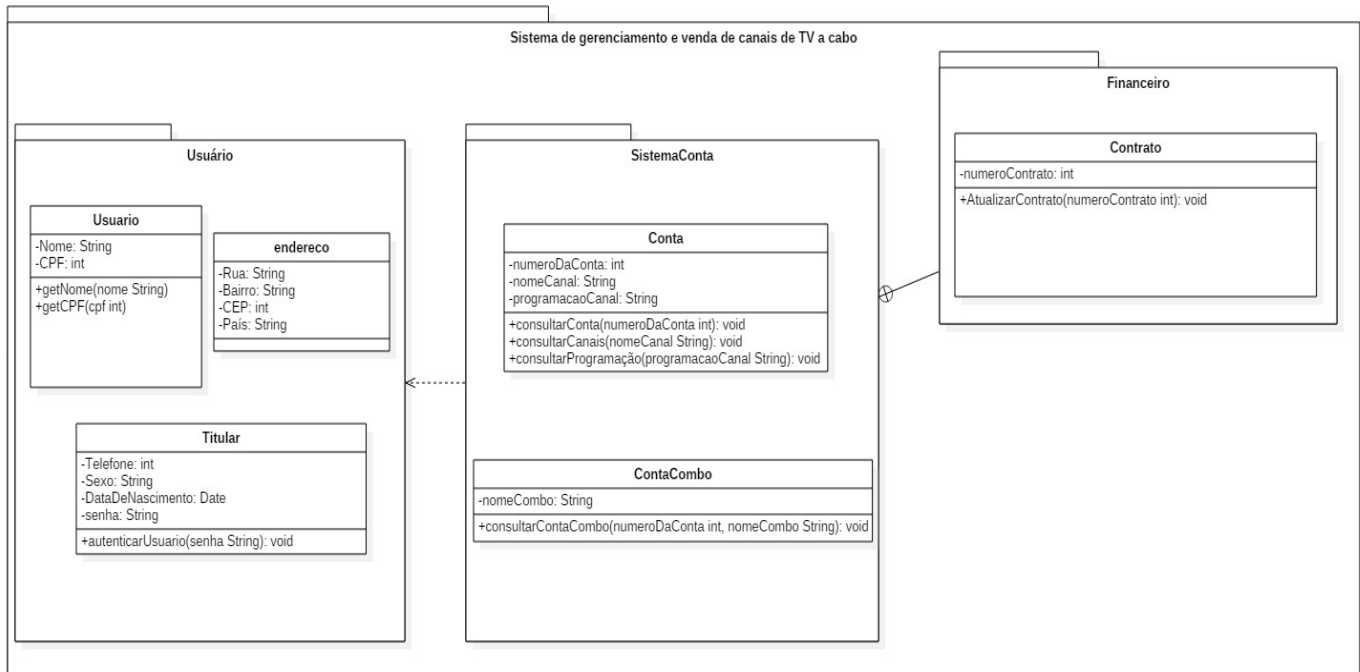


Visão do Diagrama de Atividades

A visão usada neste diagrama foi do tipo lógica.

6. Diagrama de Pacotes

Diagrama referente ao Sistema de gerenciamento e venda de canais de TV a cabo:



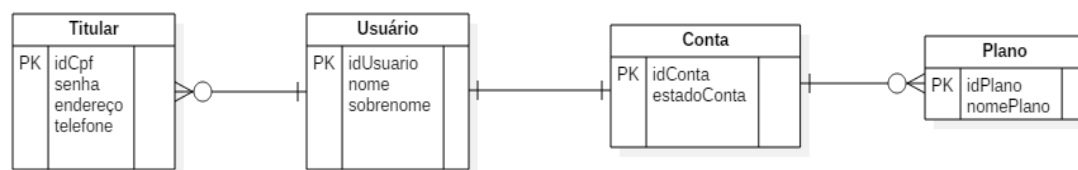
Visão do Diagrama de Pacotes.

A visão usada neste diagrama foi do tipo lógica.

8. Diagramas de entidade x relacionamento

Abaixo estão os modelos de diagramas com suas entidades, relacionamentos e chaves referente ao usuário entrando no sistema:

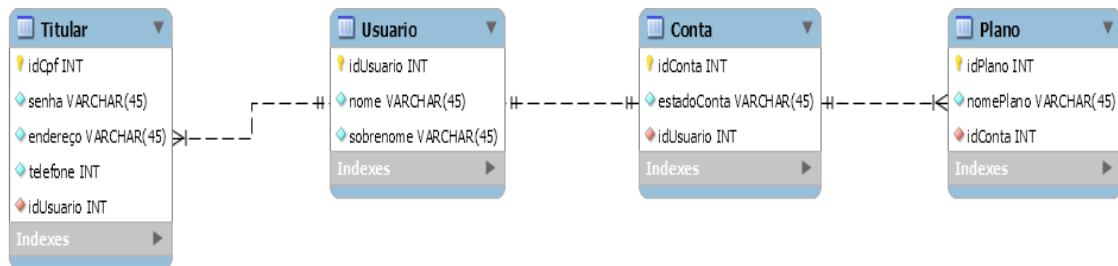
Modelo Conceitual



Visão do Diagrama de entidade x relacionamento.

A visão usada neste diagrama foi do tipo lógica.

Modelo Lógico



Modelo Físico

```
-- MySQL Script generated by MySQL Workbench
-- 12/19/16 19:31:57
-- Model: New Model   Version: 1.0
-- MySQL Workbench Forward Engineering
```

```
SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='TRADITIONAL,ALLOW_INVALID_DATES';
```

```
-- Schema mydb
```

```
-- Schema mydb
```

```
CREATE SCHEMA IF NOT EXISTS `mydb` DEFAULT CHARACTER SET utf8 ;
USE `mydb` ;
```

```
-- Table `mydb`.`Usuario`
```

```
CREATE TABLE IF NOT EXISTS `mydb`.`Usuario` (
  `idUsuario` INT NOT NULL,
  `nome` VARCHAR(45) NOT NULL,
  `sobrenome` VARCHAR(45) NOT NULL,
  PRIMARY KEY (`idUsuario`))
```

ENGINE = InnoDB;

-- Table `mydb`.`Titular`

```
CREATE TABLE IF NOT EXISTS `mydb`.`Titular` (  
  `idCpf` INT NOT NULL,  
  `senha` VARCHAR(45) NOT NULL,  
  `endereço` VARCHAR(45) NOT NULL,  
  `telefone` INT NOT NULL,  
  `idUsuario` INT NOT NULL,  
  PRIMARY KEY (`idCpf`),  
  INDEX `fk_Titular_Usuario1_idx` (`idUsuario` ASC),  
  CONSTRAINT `fk_Titular_Usuario1`  
    FOREIGN KEY (`idUsuario`)  
    REFERENCES `mydb`.`Usuario` (`idUsuario`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

-- Table `mydb`.`Conta`

```
CREATE TABLE IF NOT EXISTS `mydb`.`Conta` (  
  `idConta` INT NOT NULL,  
  `estadoConta` VARCHAR(45) NOT NULL,  
  `idUsuario` INT NOT NULL,  
  PRIMARY KEY (`idConta`),  
  INDEX `fk_Conta_Usuario1_idx` (`idUsuario` ASC),  
  CONSTRAINT `fk_Conta_Usuario1`  
    FOREIGN KEY (`idUsuario`)  
    REFERENCES `mydb`.`Usuario` (`idUsuario`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

-- Table `mydb`.`Plano`

```
CREATE TABLE IF NOT EXISTS `mydb`.`Plano` (  
  `idPlano` INT NOT NULL,  
  `nomePlano` VARCHAR(45) NOT NULL,  
  `idConta` INT NOT NULL,  
  PRIMARY KEY (`idPlano`),  
  INDEX `fk_Plano_Conta1_idx` (`idConta` ASC),  
  CONSTRAINT `fk_Plano_Conta1`  
    FOREIGN KEY (`idConta`)  
    REFERENCES `mydb`.`Conta` (`idConta`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

```
SET SQL_MODE=@OLD_SQL_MODE;  
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;  
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
```

Mapeamento objeto relacional

A diferença da modelagem de dados relacional com a modelagem UML é que o modelo relacional é um modelo de dados, e o modelo de classes é um modelo de objetos, compostos por atributos e comportamentos.

Para melhorar a redundância dos dados e o custo de armazenamento de informações em discos rígidos e outros dispositivos de armazenamento, os dados devem ser concisos e as informações aparecerem somente uma vez no SGBD.

As classes foram mapeadas a tabelas (esquemas) e as instâncias (objetos) mapeados a registros (linhas).

Um exemplo é a Tabela Conta, onde String nome, String Sobrenome e Int idUsuario são as instâncias da classe Conta e as linhas seus registros.

The screenshot displays a database management interface. At the top, a tab labeled 'usuario' is active. Below it, a SQL query is entered: `1 • SELECT * FROM mydb.usuario;`. The query is executed, and the results are shown in a 'Result Grid' below. The grid has three columns: 'idUsuario', 'nome', and 'sobrenome'. It contains four rows of data, all with 'Henrique' as the name. The last row has 'Moura' as the surname. Below the grid, there is an 'Output' section with a table showing the execution details.

	idUsuario	nome	sobrenome
1	1	Henrique	de Moura
2	2	Henrique	Galvão
3	3	Henrique	Zwicker
4	4	Henrique	Moura
*	NULL	NULL	NULL

#	Time	Action	Message
1	19:50:54	SELECT * FROM mydb.usuario LIMIT 0, 1000	4 row(s) returned

9. REFERÊNCIAS

Utilizando Uml e Padrões - Craig Larman . Editora: ARTMED – BOOKMAN.

<http://lms.infnet.edu.br/moodle/mod/page/view.php?id=39455>

<http://lms.infnet.edu.br/moodle/mod/page/view.php?id=39461>

<http://lms.infnet.edu.br/moodle/mod/page/view.php?id=39468>

<http://lms.infnet.edu.br/moodle/mod/page/view.php?id=39474>

<http://lms.infnet.edu.br/moodle/mod/page/view.php?id=39481>

<http://lms.infnet.edu.br/moodle/mod/page/view.php?id=39487>

<http://lms.infnet.edu.br/moodle/mod/page/view.php?id=39494>

<http://www.devmedia.com.br/uml-unified-modeling-language-parte-01/9427>

<https://docs.google.com/presentation/d/17AMi5amjUR3nIZK5DCGc7nVhAr9bRjnLtAV0frcNSDg/htmlpresent>

<http://ehgomes.com.br/disciplinas/bdd/sqbd.php>