**DES - Documento de Engenharia de Software**

Versão: 0.1

09 de agosto de 2017

**Projeto Pouso Alegrill**

**Luiz Felipe Pereira Lima**

Pedro Henrique Carvalho

Karla Pereira do Carmo

EC206 - AulaLab 2 - Documento Engenharia de Software.docx

Tabela de Revisões

| Versão | Principais Autores | Descrição da Versão | Data de Término | Aprovação e data | |
| --- | --- | --- | --- | --- | --- |
| V0.1 | **Luiz Felipe**  Pedro Henrique  Karla Pereira | Introdução, Escopo e Descrição de Funcionamento. | 09/08/2017 | **Marina Vilela** | xx/xx/xxxx |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

Índice

Tabela de Revisões 2

Índice 3

1. Lista de Figuras 4

2. Lista de Tabelas 5

3. Introdução 6

3.1 Definições, Acrônimos e Abreviaturas 6

4. Visão geral 7

4.1 Introdução 7

4.2 Escopo 7

4.3 Descrição de funcionamento 7

5. Especificação de Requisitos 9

5.1 Requisitos Funcionais 9

5.1.1 Req.1 - Efetuar o cadastro dos clientes de acesso 9

5.1.2 Req.2 - Exibir o relatório de backups 9

5.2 Diagrama de Casos de Uso 10

5.2.1 Descrição dos Atores 10

5.2.2 Descrição dos Casos de Uso 10

5.3 Fluxos de Eventos de Casos de Uso 11

5.3.1 Login do Administrador 11

5.4 Requisitos Não-Funcionais 12

5.4.1 Req.91 - Utilizar Windows como sistema operacional 12

5.4.2 Requisitos de Desempenho 12

5.4.2.1 Req.92 - O tempo da geração de relatório não deve exceder 1 segundo. 12

6. Projeto Arquitetural 13

6.1 Diagrama de contexto arquitetural 13

6.2 Relacionamentos UML para arquétipos 14

6.3 Diagrama de pacotes 14

6.4 Instanciação dos componentes 15

7. Projeto de Dados 16

7.1 Modelo Entidade-Relacionamento 16

8. Projeto Lógico 17

8.1 Diagrama de Classes 17

8.2 Diagrama de Sequência 17

9. Qualidade de Software 18

9.1 Rastreabilidade dos Requisitos 18

9.2 Métricas 19

9.3 Testes 19

9.4 Design Patterns 19

10. Anexos 20

10.1 Storyboarding 20

10.2 Estrutura Analítica do Projeto - EAP 20

10.3 Cronograma de Atividades 21

11. Bibliografias de Texto 22

12. Bibliografia de Imagens 23

# Lista de Figuras

**Figura 1 -** Exemplo da implementação. 7

**Figura 2 -** Diagrama do <Projeto>. 8

**Figura 3 -** Fluxograma do <Projeto>. 8

**Figura 4 -** Diagrama de casos de uso. 10

**Figura 5 -** Instanciação dos Componentes. 15

**Figura 6 -** Modelo Entidade-Relacionamento. 16

**Figura 7 -** Diagrama de Classes. 17

**Figura 8 -** Diagrama de Sequência. 17

**Figura 9 -** Telas do Software. 20

# Lista de Tabelas

**Tabela 01 -** Requisito Req.1. 9

**Tabela 02 -** Requisito Req.2. 9

**Tabela 03 -** Fluxo de evento principal < Login do Administrador >. 11

**Tabela 04 -** Rastreabilidade: Requisitos Funcionais em Casos de Uso. 18

**Tabela 05 -** Rastreabilidade: Casos de Uso em Classes. 18

**Tabela 06 -** Rastreabilidade: Classes em Pacotes. 18

**Tabela 07 -** Métricas. 19

**Tabela 08 -** Lançamento das Horas. 19

# Introdução

## Definições, Acrônimos e Abreviaturas

**À La Carte –**  é uma expressão típica do francês que significa "como está no cardápio" ou "como listado no cardápio" [1].

**Feedback** **–** Reação ou retorno de alguma ação, efeito retroativo.

# Visão geral

## Introdução

O restaurante Pouso Alegrill é uma opção tradicional de alimentação e lazer na zona rural de Pouso Alegre – MG. Seu serviço consiste de pratos feitos durante a semana e pratos à la carte aos finais de semana. Desde julho de 2017 também tem servido churrasco por peso também aos finais de semana, o que fez com que o movimento aumentasse muito. Assim o método manual de anotação de pedidos começou a ser ineficiente, com atendentes esquecendo de tomar notas e cozinha preparando pratos fora da ordem pedida.

Os donos precisam de uma solução de software rápida e confiável que aumente o controle sobre o atendimento e preparo de pratos e bebidas, gerenciando a fila de pedidos feitos tomados nas mesas.

A figura 1 a seguir apresenta o logotipo do estabelecimento:



**Figura 1 –** Logotipo do restaurante Pouso Alegrill.

## Escopo

O projeto consiste em desenvolver um programa em Java que controle o processo de atendimento do restaurante Pouso Alegrill, possibilitando a criação de novos pratos , edição dos pratos existentes e remoção pelo dono do restaurante (administrador) , seleção e listagem dos pratos para uma determinada mesa, feita pelo garçom, cadastro dos pedidos , listagem e remoção dos pedidos, além do controle de filas , feita pelo gerente.

Na função de criação de novos pratos, o gerente terá que preencher as seguintes características dos pratos: nome, descrição, unidade e preço. Os pratos serão armazenados no banco de dados, possibilitando a listagem dos pratos cadastrados, edição ou remoção dos mesmos, feitas pelo gerente (administrador). Para a seleção dos pratos, o garçom deve listar os pratos existentes e escolher o que foi pedido pelo cliente de uma determinada mesa.

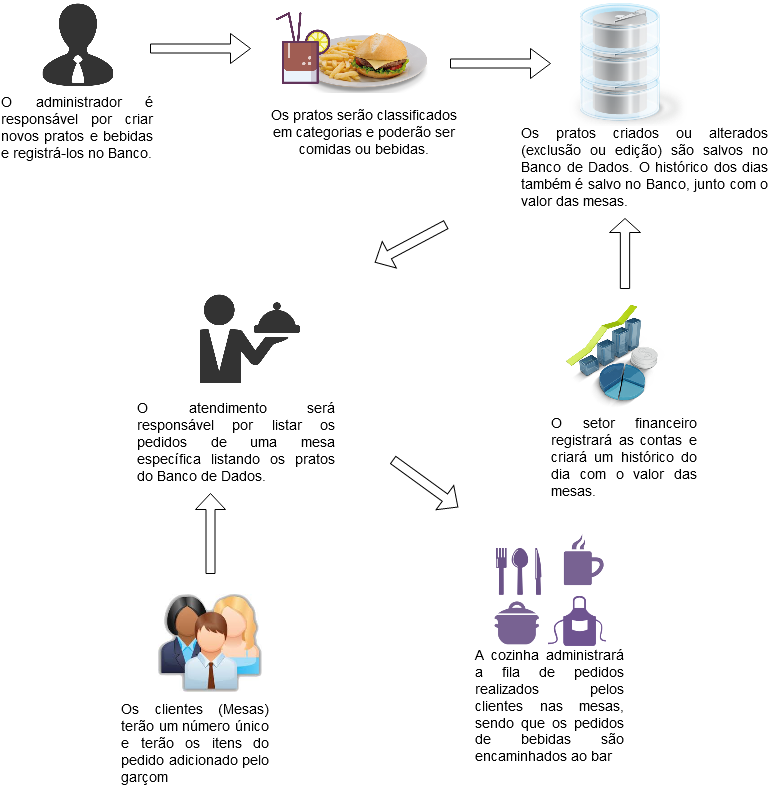
**CRUD Funcionários:** Será dividido em quatro categorias: administrativo, atendimento, cozinha e bar. O administrativo será responsável em fazer a criação dos pratos e bebidas do cardápio e a criação das categorias dos mesmos. O atendimento será responsável em adicionar os pratos e bebidas do cardápio para a mesa específica, tendo uma listagem dos produtos através das categorias. O pedido ao ser finalizado entrará na fila da cozinha e do bar, onde ganhará *feedback* de positivo para o garçom levar o pedido pronto até a mesa, apenas o administrativo poderá excluir um item do pedido. A cozinha poderá administrar a fila dos pedidos, esta fila será exibida em modo de chegada e quando o prato estiver pronto o responsável pelo prato poderá sinalizar com um *feedback*, fazendo com que o pedido saia da fila. O bar seguirá o mesmo padrão da cozinha, porém receberá apenas pedidos de bebidas.

**CRUD Mesa:** Ao adicionar a mesa ela terá um número único e nela conterá itens do pedido adicionado pelo garçom.

**CRUD Comida:** As comidas serão classificadas em categorias, contendo o nome, descrição da comida e valor.

**CRUD Bebida:** As bebidas serão classificadas em categorias, contendo o nome, descrição da bebida e valor.

**CRUD Financeiro:** Será responsável por finalizar os pedidos de todas as mesas e assim criará um histórico do dia com o valor de todas as mesas com os pedidos feitos com seus respectivos valores.



**Figura 1 –** Desenho do Escopo do Projeto.

## Descrição de funcionamento

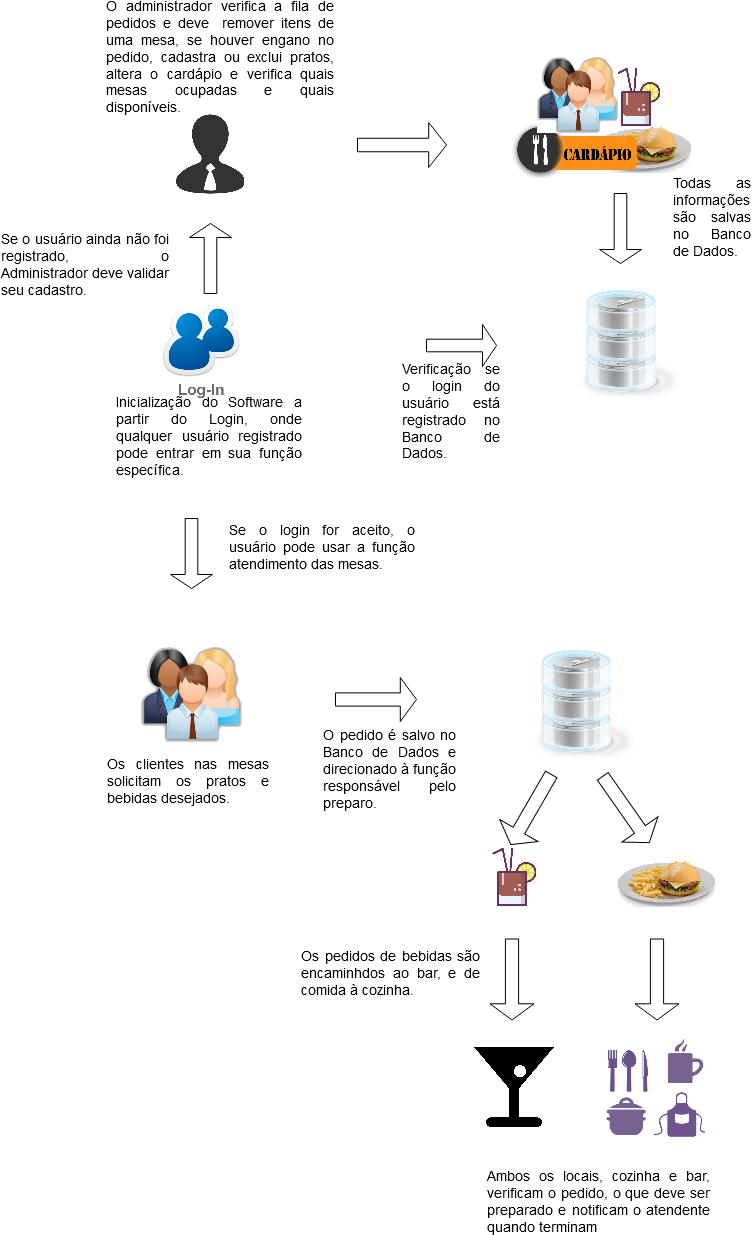
O software inicializa na tela de login, onde qualquer usuário pode entrar com suas credenciais para validação de acesso. Caso não haja registro do cadastro no Banco de Dados ocorrerá um aviso, então é possível tentar novamente ou criar uma nova conta inserindo seu nome, e-mail e senha. Novas contas só exercerão alguma função depois de serem designadas pelo administrador. Quando as informações de login corresponderem a um registro, o usuário será levado diretamente para a tela de sua função.

A tela do administrador do sistema apresenta a visualização das filas de pedidos (comida e bebida) e das mesas ocupadas no momento, assim como botões para o gerenciamento de funções dos usuários e itens do cardápio. É de sua responsabilidade remover itens de uma mesa, caso haja um engano no pedido.

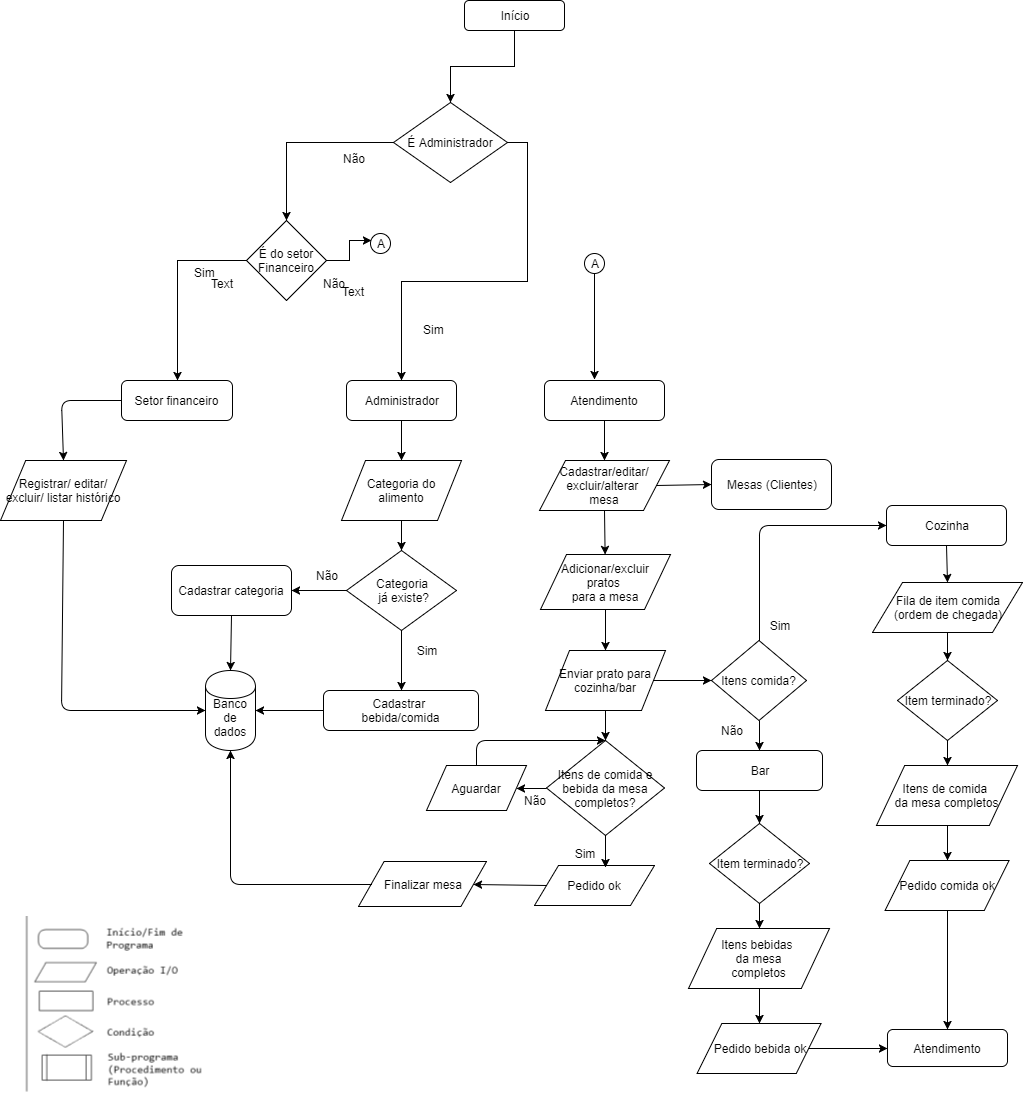
Para os usuários com função de atendimento, sua tela permite abrir uma nova conta, determinada pelo número da mesa, e adicionar itens à medida que são pedidos pelos clientes. Para tomar um pedido, o atendente deve selecionar uma mesa ocupada, navegar pelas categorias disponíveis no cardápio, determinar a quantidade dos itens e adicioná-los à uma lista temporária (pedido atual). Ao confirmar com os clientes os itens selecionados e suas quantidades, deve apertar um botão para confirmar o pedido, o que faz com que a lista criada seja adicionada à conta da mesa no Banco de Dados, e que os itens contidos sejam direcionados para as funções responsáveis pelo preparo (Cozinha e Bar), que notifica o atendente quando ficam prontos.

As telas disponíveis para usuários do Bar e da Cozinha são essencialmente a mesma. Consiste de uma tabela com os itens que devem ser preparados, ordenados em função da hora em que foram pedidos, os números das mesas a que pertencem, e botões "Pronto", que devem ser clicados quando cada item estiver pronto.

Todas informações serão mantidas em um Banco de Dados, do qual serão recuperadas e gravadas segundo as ações tomadas pelos usuários.



**Figura 2 -** Diagrama do Pouso Alegrill.



**Figura 3 -** Fluxograma do Pouso Alegrill.

# Especificação de Requisitos

## Requisitos Funcionais

### Req. - Efetuar o cadastro dos clientes de acesso

|  |  |
| --- | --- |
| **Detalhamento** | Cada cliente deve possuir os seguintes dados:  - Nome;  - IP.  ... |
| **Observação** | O CPF deve ser validado para efetuar o cadastro do cliente. Deve somente haver o cadastro, caso todas as informações citadas acima forem preenchidas, ou seja, não deve haver ausência de informação em nenhum campo do cadastro. |
| **Prioridade** | Alta. |

**Tabela 01 -** Requisito Req.1.

### Req. - Exibir o relatório de backups

|  |  |
| --- | --- |
| **Detalhamento** | O sistema deve prover meios de exibir um relatório de backups contento as seguintes informações:  - IP;  - Quantidade de backups desde o início do cadastro;  - Data do último backup. |
| **Observação** | O relatório será exibido somente para os usuários cadastrados no sistema. |
| **Prioridade** | Alta. |

**Tabela 02 -** Requisito Req.2.

## Diagrama de Casos de Uso



**Figura 4 -** Diagrama de casos de uso.

### Descrição dos Atores

**A1 - Administrador**

O Administrador tem acesso à as funcionalidades de Manter Viagens, Reservar Viagem, Manter Clientes, Manter Funcionários.

### Descrição dos Casos de Uso

**CaU1 - Manter Clientes**

Este caso de uso tem como objetivo manipular os dados dos clientes no banco de dados. Ela é composta pelas funcionalidades de cadastrar, listar, editar e excluir clientes. Somente o Administrador tem acesso a este caso de uso.

## Fluxos de Eventos de Casos de Uso

### Login do Administrador

|  |  |  |
| --- | --- | --- |
| **Nome da Use Case** | Login do **Administrador**. | |
| **Descrição** | Responsável pelo Login no software. | |
| **Requisitos associados** | Login. | |
| **Pré-condições** | Sistema tem que estar ligado. | |
| Possuir um Login. | |
| **Pós-condições** | Login certo. | |
| Login errado. | |
| **Atores** | **Administrador** e **Banco de Dados**. | |
| **Fluxo Principal** | | |
| **Ações** **Realizadas** | | **Ações Recebidas** |
| 1 - O **Administrador** deseja fazer o Login. | | 2 - O sistema solicita o Login. |
| 3 - O **Administrador** digita o Login. | | 4 - O sistema verifica o Login. |
| 5 - Se Login estiver certo, entra no software. |
| **Fluxo Alternativo** | | |
| **Ações** **Realizadas** | | **Ações Recebidas** |
| 1 - Sistema apresenta mensagem que o Login está errado. | | 3 - Usuário deseja tentar novamente. |
| 2 - Sistema pergunta se deseja efetuar Login novamente ou cancelar. | |
| 4 - Sistema solicita o Login. | | 5 - Usuário solicita cancelar operação. |
| 6 - Caso de uso encerrado. | |

**Tabela 3 -** Fluxo de evento principal < Login do Administrador >.

## Requisitos Não-Funcionais

### Req.9 - Utilizar Windows como sistema operacional

Será utilizada uma distribuição Windows ...

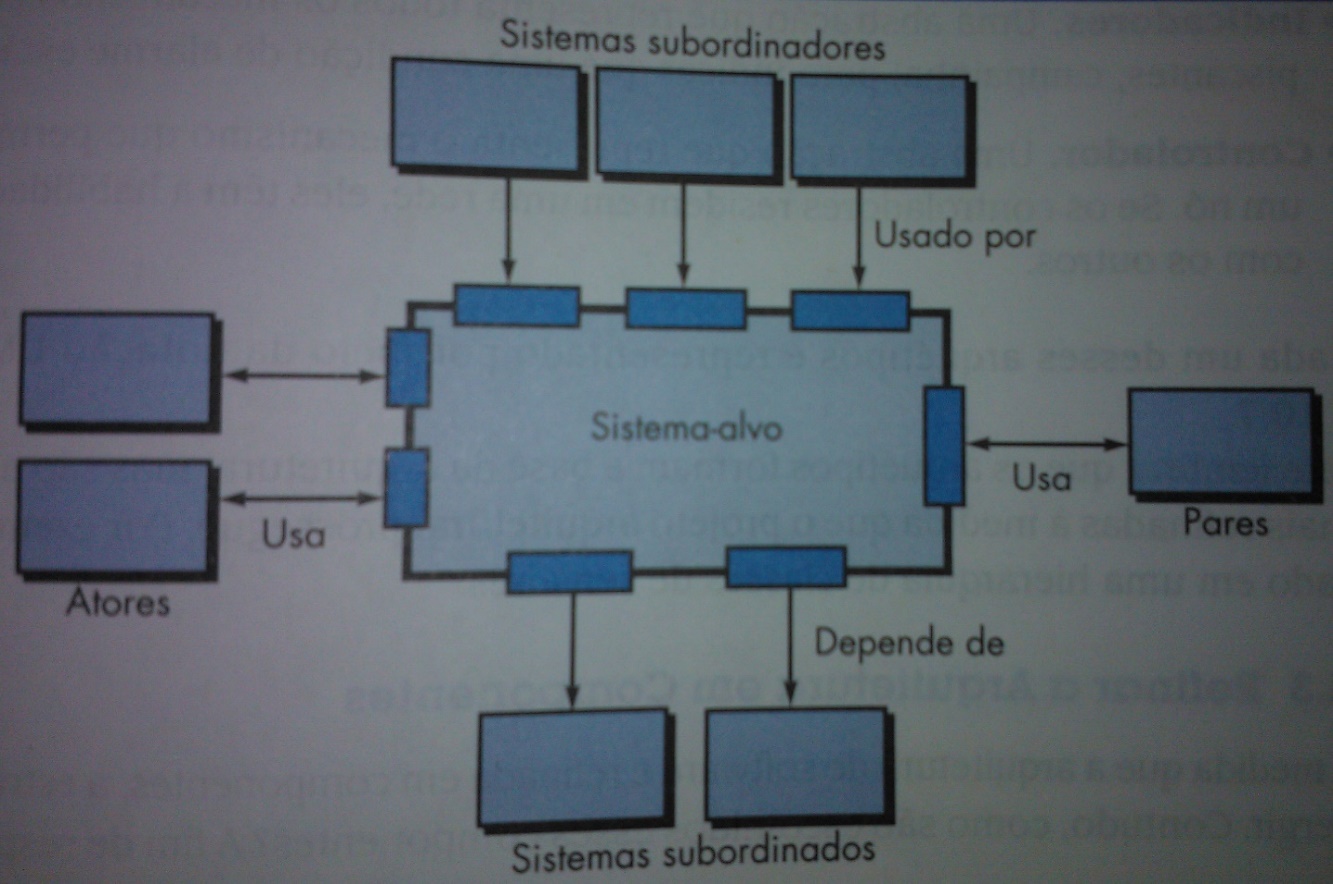
### Requisitos de Desempenho

#### Req.9 - O tempo da geração de relatório não deve exceder 1 segundo.

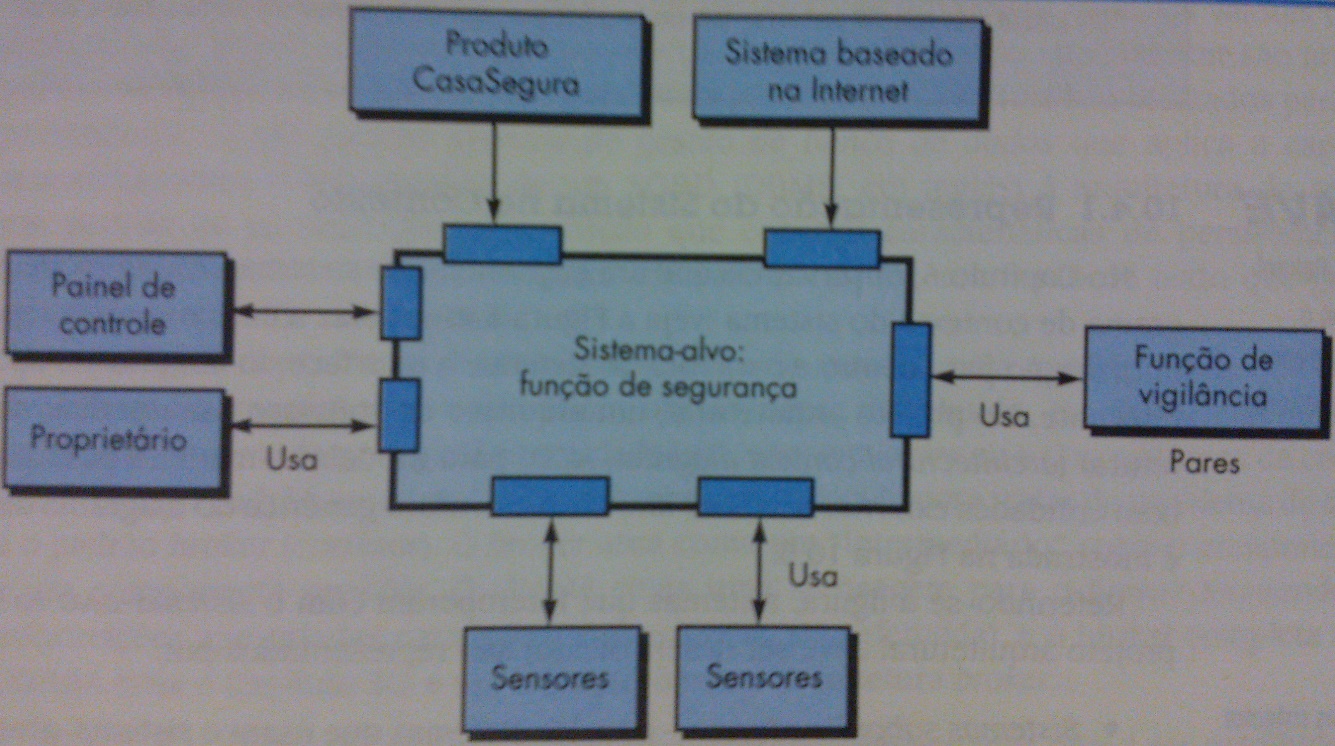
Este requisito deve ser validado somente se o sistema estiver ocioso e possua um número de registro menor que um milhão de clientes.

# Projeto Arquitetural

## Diagrama de Contexto Arquitetural

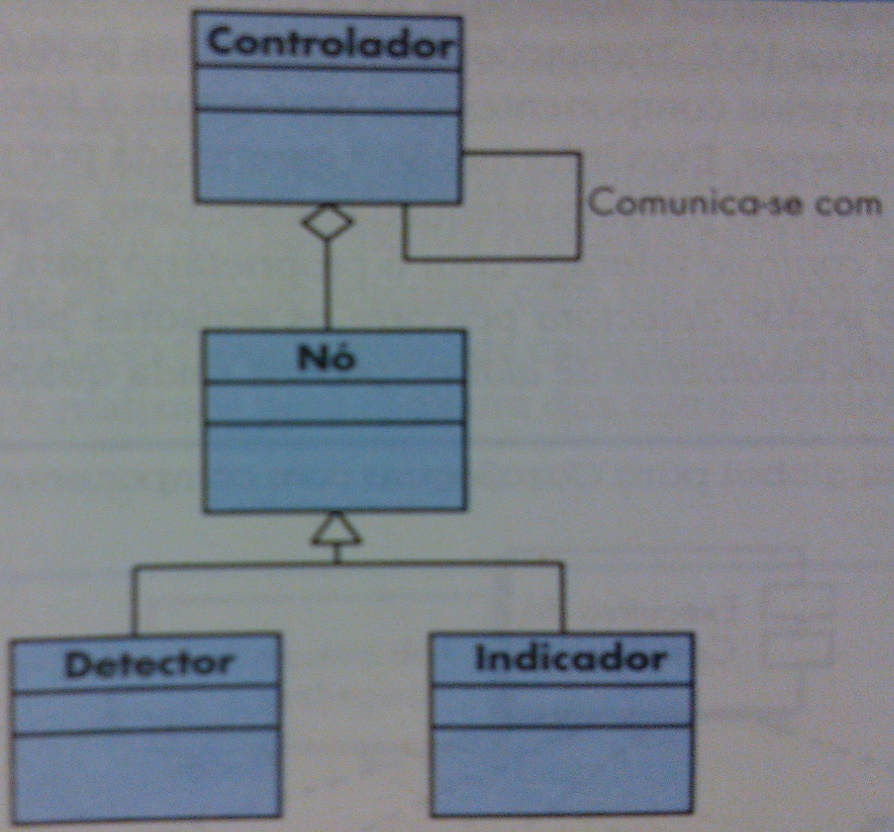


**Figura 5 -** Diagrama de Contexto Arquitetural.



**Figura 6 -** Diagrama de Contexto Arquitetural.

## Relacionamentos UML para Arquétipos



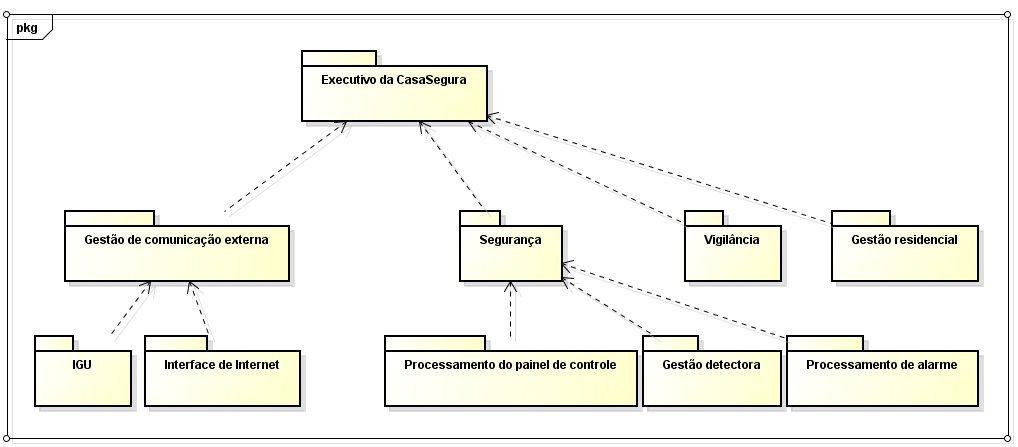
**Figura 7 -** Relacionamentos UML para Arquétipos.

## Diagrama de Pacotes



**Figura 8 -** Diagrama de Pacotes.

## Instanciação dos Componentes



**Figura 5 -** Instanciação dos Componentes.

# Projeto de Dados

## Modelo Entidade-Relacionamento



**Figura 6 -** Modelo Entidade-Relacionamento.

# Projeto Lógico

## Diagrama de Classes



**Figura 7 -** Diagrama de Classes.

## Diagrama de Sequência



**Figura 8 -** Diagrama de Sequência.

# Qualidade de Software

## Rastreabilidade dos Requisitos

**9.1.1. Requisitos Funcionais em Casos de Uso**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **<CU1>** | **<CU2>** | **<CU3>** | **...** | **<CUN>** |
| **RF1** |  |  |  |  |  |
| **RF2** |  |  |  |  |  |
| **RF3** |  |  |  |  |  |
| **...** |  |  |  |  |  |
| **...** |  |  |  |  |  |
| **...** |  |  |  |  |  |
| **RFN** |  |  |  |  |  |

**Tabela 4 -** Rastreabilidade: Requisitos Funcionais em Casos de Uso.

**9.1.2. Casos de Uso em Classes**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **<Classe1>** | **<Classe2>** | **<Classe3>** | **...** | **<ClasseN>** |
| **<CU1>** |  |  |  |  |  |
| **<CU2>** |  |  |  |  |  |
| **<CU3>** |  |  |  |  |  |
| **...** |  |  |  |  |  |
| **...** |  |  |  |  |  |
| **...** |  |  |  |  |  |
| **<CUN>** |  |  |  |  |  |

**Tabela 5 -** Rastreabilidade: Casos de Uso em Classes.

**9.1.3. Classes em Pacotes**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **<Pacote1>** | **<Pacote2>** | **<Pacote3>** | **...** | **<PacoteN>** |
| **<Classe1>** |  |  |  |  |  |
| **<Classe2>** |  |  |  |  |  |
| **<Classe3>** |  |  |  |  |  |
| **...** |  |  |  |  |  |
| **...** |  |  |  |  |  |
| **...** |  |  |  |  |  |
| **<ClasseN>** |  |  |  |  |  |

**Tabela 6 -** Rastreabilidade: Classes em Pacotes.

## Métricas

Ao longo do projeto são coletadas três medidas, sendo duas estimativas e uma real. As estimativas são de Ponto de Função (PF) e Ad Hoc. A medida Ad hoc considera a percepção da equipe de quantas horas serão consumidas para o projeto, do início ao término, incluindo as horas de todo o time. Incluir as horas dedicadas ao projeto durante as aulas de laboratório, durante as reuniões do time e nos trabalhos individuais. Os valores de fechamento devem ser os valores de PF e Ad hoc da 3ª medida e o valor real conforme o total coletado na tabela de valores reais semanais.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **1ª medida** | **2ª medida** | **3ª medida** | **Fechamento** |
| **PF** |  |  |  |  |
| **Ad hoc** |  |  |  |  |
| **Real** |  |  |  |  |

**Tabela 7 -** Métricas.

Lançamento semanal das horas reais consumidas com o projeto.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Semana** | **1ª** | **2ª** | **3ª** | **4ª** | **5ª** | **6ª** | **7ª** | **8ª** | **9ª** | **10ª** | **11ª** | **12ª** | **13ª** | **14ª** | **15ª** | **Total** |
| **Horas consumidas** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**Tabela 8 -** Lançamento das Horas.

## Testes

Ao longo da disciplina de EC206 são elaborados e aplicados vários casos de testes para o projeto em desenvolvimento. Estes casos de testes e seus respectivos resultados deverão ser adicionados nesta seção.

## Design Patterns

Na codificação do sistema, tem que ter pelo menos um padrão de projeto inserido, adicionar ao documento o padrão escolhido e explicar como foi aplicado ao projeto.

# Anexos

## Storyboarding



**Figura 9 -** Telas do Software.

## Estrutura Analítica do Projeto - EAP



**Figura 14 -** EAP.

## Cronograma de Atividades



**Figura 15 -** Cronograma.

# Bibliografias de Texto

[1] Significados. **Significado de À la carte.** Disponível em <https://www.significados.com.br/a-la-carte/>

# Bibliografia de Imagens