FACULDADE DE TECNOLOGIA DO IPIRANGA

CURSO DE ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

André Yuki nakamura

Pedro de castro souza

RENAN CAMAROSANO PIRES

NuggetX

São Paulo

2016



FACULDADE DE TECNOLOGIA DO IPIRANGA

CURSO DE ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

André Yuki nakamura

Pedro de castro souza

RENAN CAMAROSANO PIRES

Nuggetx

Trabalho de Graduação apresentado à Faculdade de Tecnologia do Ipiranga, como requisito parcial para obtenção do grau de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientadora: Profª. Me. Andreia Machion

São paulo

2016

André Yuki nakamura

Pedro de castro souza

RENAN CAMAROSANO PIRES

nuggetx

Trabalho de Graduação apresentado à Faculdade de Tecnologia do Ipiranga, como requisito parcial para obtenção do grau de Tecnólogo em Análise e Desenvolvimento de Sistemas.

São Paulo

2016

Sumário

[1. INTRODUÇÃO 9](#_Toc448531173)

[1.1Objetivo 9](#_Toc448531174)

[1.2Justificativa 9](#_Toc448531175)

[2 REVISÃO BIBLIOGRÁFICA 10](#_Toc448531176)

[2.1 Seção Nível Dois 10](#_Toc448531178)

[*2.1.1 Seção nível três* 10](#_Toc448531179)

2.1.1.1 Seção nível quatro ......................................................................10

[3 REQUISITOS DO SISTEMA DE SOFTWARE 11](#_Toc448531183)

[3.1 Identificação dos requisitos 11](#_Toc448531184)

3.1.1 Prioridades dos Requisitos .............................................................11

[3.1.2 Requisitos Funcionais 12](#_Toc448531188)

[3.1.3 Regras de Negócio 14](#_Toc448531189)

[3.2 Modelagem dos requisitos funcionais 16](#_Toc448531190)

[3.2.1 Atores 1](#_Toc448531191)7

[3.2.2 Diagrama de Caso de Uso 17](#_Toc448531193)

3.2.3 Especificações do Caso de Uso .........................................................18

[4 DESENVOLVIMENTO DO PROJETO 23](#_Toc448531194)

[4.1 Análise 2](#_Toc448531195)4

[4.1.1 Diagrama de Classes de Análise (Visão de Negócio) 24](#_Toc448531196)

[4.2 Projeto 24](#_Toc448531197)

[4.2.1Arquitetura do Sistema 25](#_Toc448531198)

[4.2.2Diagrama de Classes de Projeto por Caso de Uso 25](#_Toc448531199)

[4.2.4 Diagrama de estados 34](#_Toc448531200)

# INTRODUÇÃO

Na maioria dos bares e restaurantes do Brasil os garçons fazem todo o trabalho de receber os pedidos dos clientes e servi-los. Contudo, dependendo do estabelecimento há uma gama de clientes e apenas alguns funcionários, gerando uma lentidão no atendimento. E também no termino de suas refeições alguns grupos de pessoas gostam de dividir uma única conta, causando discussões, lentidão na fila do caixa, sobrecarregando o funcionário sem necessidade.

Com o excesso de trabalho que é exigido conforme a demanda de clientes, os responsáveis devem contratar mais mãos-de-obra, ocasionando em gastos, que as vezes eles não dispõem. Já existem algumas soluções para realizar pedidos digitalmente pelo garçom ou pelo cliente, porém todos são semelhantes, deixando a inovação para trás, sem implementações que os usuários precisam, como saber se o estabelecimento está lotado ou não antes de sair de casa.

## 1.1Objetivo

Com este problema e com ideias de inovações, iremos planejar e implementar uma aplicação que visa simplificar o trabalho do garçom e melhorar a experiência do cliente, buscando a satisfação de ambos os lados e surpreendendo os usuários. Ao longo do projeto analisaremos o funcionamento de alguns estabelecimentos para conseguir integrar nossa aplicação com o serviço deles. E para alcançar nossos objetivos iremos realizar testes com um único estabelecimento aos poucos para analisarmos como funcionará e o que devemos ajustar.

## 1.2Justificativa

A cada dia a rotina da população fica mais corrida, sobrando poucas horas para realizar algumas refeições, e claro, sem tempo para cozinhar algo. Com isso os estabelecimentos enchem nas horas de picos, com poucos funcionários e muitos clientes estressados com a rotina. Aumentando o numero de reclamações e os problemas relacionados ao estabelecimento. Com a nossa aplicação, iremos tentar contribuir com a experiência de atendimento, de uma forma que agilize o processo e impressione os usuários.

**2 REVISÃO BIBLIOGRÁFICA**

Neste capítulo, o objetivo é levantar o estado da arte referente ao tema de pesquisa que está sendo realizada. Pode ser realizada uma pesquisa sobre as tecnologias empregadas no desenvolvimento, metodologia de desenvolvimento e plataforma.

O professor orientador que irá elaborar a estrutura desse capítulo, a fim de delinear a pesquisa.

Na Figura 1é apresentado....

Figura 1- Exemplo de figura

**Fonte: Autora (2014)**

**2.1 Seção Nível Dois**

Nesta seção o título deve ter a primeira letra de cada palavra deve estar em maiúsculo e negrito

***2.1.1 Seção nível três***

Nesta seção o título deve ter somente a primeira palavra iniciada com maiúsculo. É permitido utilizar negrito e/ou itálico. Na FATEC Ipiranga iremos utilizar o itálico.

**2.1.1.1 Seção nível quatro**

Nesta seção o título deve ter somente a primeira palavra iniciada com maiúsculo. É permitido utilizar itálico ou sublinhado. Na FATEC Ipiranga iremos utilizar o sublinhado, bem como definimos também que serão utilizados no máximo 4 níveis.

2. REQUISITOS DO SISTEMA DE SOFTWARE

Neste capítulo são apresentados os requisitos funcionais, os não funcionais, as regras de negócio, os modelos UML e os protótipos. Os requisitos são parte importante do trabalho pois definem o que o aplicativo deverá fazer com base nas necessidades do cliente.

**3.1 Identificação dos requisitos**

Por convenção, a referência a requisitos é feita através do nome da subseção onde eles estão descritos, seguidos do identificador do requisito, de acordo com a especificação a seguir:

* **Requisitos funcionais** devem ser identificados por: [RFxxx] – nome e descrição e serem descritos na seção requisitos funcionais.
* **Regras de negócio** devem ser identificadas por: [RNxxx] – nome e descrição e serem descritas na seção requisitos funcionais, subitem regras de negócio.
* **Requisitos não funcionais** devem ser identificados por: [RNFxxx] – nome e descrição e serem descritos na seção requisitos não funcionais.

Os requisitos devem ser identificados com um identificador único. A numeração inicia com o identificador [RF001], [RN001] ou [RNF001] e prossegue sendo incrementada à medida que forem surgindo novos requisitos ou regras.

Por convenção, a referência aos casos de uso é feita através do nome da subseção onde eles estão descritos, seguidos do identificador do caso de uso, de acordo com a especificação a seguir:

Casos de Uso devem ser identificados por: CSUXXX – nome a ser descrito na função modelagem funcional.

Os casos de uso devem ser identificador com um identificador único. A numeração inicia com o identificador CSU001 e prossegue sendo incrementada à medida que forem surgindo novos casos de uso.

**3.1.1 Prioridades dos requisitos**

Para estabelecer a prioridade dos requisitos, foram adotadas as denominações “essencial”, “importante” e “desejável”.

* **Essencial** é o requisito sem o qual o sistema não entra em funcionamento. Requisitos essenciais são requisitos imprescindíveis, que têm que ser implementados impreterivelmente.
* **Importante** é o requisito sem o qual o sistema entra em funcionamento, mas de forma não satisfatória. Requisitos importantes devem ser implementados, mas, se não forem, o sistema poderá ser implantado e usado mesmo assim.
* **Desejável** é o requisito que não compromete as funcionalidades básicas do sistema, isto é, o sistema pode funcionar de forma satisfatória sem ele. Requisitos desejáveis podem ser deixados para versões posteriores do sistema, caso não haja tempo hábil para implementá-los na versão que está sendo especificada.



3.1.2 Requisitos Funcionais

São requisitos que o aplicativo deverá atender relacionados basicamente às entradas, tratamentos, processamento e saídas de dados.

**[RF001] – Realizar login**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: | ◼ | Essencial | 🞎 | Importante | 🞎 | Desejável |

**Descrição**: Este requisito permite que todos os usuários do sistema entrem no sistema com um login do Facebook ou com um login criado no próprio sistema.

**[RF002] – Manter estabelecimento**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: | ◼ | Essencial | 🞎 | Importante | 🞎 | Desejável |

**Descrição**: Este requisito permite que o Administrador cadastre um estabelecimento, modifique e visualize dados dos estabelecimentos cadastrados, e delete estabelecimentos do sistema.

**[RF003] – Cadastrar funcionário**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: | ◼ | Essencial | 🞎 | Importante | 🞎 | Desejável |

**Descrição**: Este requisito permite que os funcionários do tipo gerente cadastrem funcionários, modifiquem e visualizem dados dos funcionários cadastrados, e deletem funcionários do sistema.

**[RF004] – Manter cardápio**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: | ◼ | Essencial | 🞎 | Importante | 🞎 | Desejável |

**Descrição**: Este requisito permite que os funcionários do tipo gerente cadastrem cardápios, modifiquem e visualizem dados dos cardápios cadastrados, e deletem cardápios do sistema.

**[RF005] – Manter item**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: | ◼ | Essencial | 🞎 | Importante | 🞎 | Desejável |

**Descrição**: Este requisito permite que os funcionários cadastrem itens, modifiquem e visualizem dados dos itens cadastrados, e deletem itens do sistema.

**[RF006] – Modificar status**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: | 🞎 | Essencial | ◼ | Importante | 🞎 | Desejável |

**Descrição**: Este requisito permite que o funcionário modifique o status do estabelecimento, tais como: “Vazio”, “Moderado” e “Cheio”.

**[RF007] – Cadastrar comanda**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: | ◼ | Essencial | 🞎 | Importante | 🞎 | Desejável |

**Descrição**: Este requisito permite que os funcionários cadastrem comandas, modifiquem e visualizem dados das comandas cadastradas, e deletem comandas do sistema.

**[RF009] – Manter pedido**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: |  | ◼ | Essencial | 🞎 | Importante | 🞎 | Desejável |

**Descrição**: Este requisito permite que os funcionários cadastrem pedidos, modifiquem e visualizem dados dos pedidos cadastrados, e deletem pedidos do sistema.

**[RF010] – Gerar valor**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: | ◼ | Essencial | 🞎 | Importante | 🞎 | Desejável |

**Descrição**: Este requisito permite que o funcionário e o cliente escolha o valor a ser pago por cada cliente.

## Regras de Negócio

São regras que devem ser obedecidas a fim de cumprir a finalidade do negócio.

**[RN001] – Segurança**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: | ◼ | Essencial | 🞎 | Importante | 🞎 | Desejável |

**Descrição**: O login será realizado pelo Facebook e Firebase, que já possuem seus próprios sistemas de segurança.

**[RN002] – Usabilidade**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: | ◼ | Essencial | 🞎 | Importante | 🞎 | Desejável |

**Descrição**: O sistema deve prover interface gráfica simples e intuitiva, de fácil navegação para facilitar o uso do mesmo por parte dos usuários.

**[RN003] – Pagamento via PagSeguro**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: | 🞎 | Essencial | 🞎 | Importante | ◼ | Desejável |

**Descrição**: O sistema disponibiliza um sistema de pagamento via PagSeguro para fechar a comanda, o mesmo gerará uma NFE( Nota Fiscal Eletrônica) que será enviada ao email do cliente.

**[RN004] – Assinatura do Termo de Responsabilidade**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: | 🞎 | Essencial | ◼ | Importante | 🞎 | Desejável |

**Descrição**: Ao instalar o aplicativo e usá-lo pela primeira vez, o usuário deverá concordar com o termo de responsabilidade de que os desenvolvedores do aplicativo não são responsáveis por quaisquer gastos ou custos relacionados ao uso do veículo decorrentes de tomadas de decisão auxiliadas pelo aplicativo ou não.

**[RN005] – Divisão dos valores**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: | 🞎 | Essencial | ◼ | Importante | 🞎 | Desejável |

**Descrição**: Ao gerar o valor a pagar, o sistema oferecerá três opções: 1º pagar valor maior ao consumido para subtrair do valor total da mesa; 2º pagar valor dividido igualmente entre os integrantes da mesa; 3º pagar o valor dos itens consumidos.

## Requisitos não Funcionais

**[RNF001] – Arquitetura de software**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: | ◼ | Essencial | 🞎 | Importante | 🞎 | Desejável |

**Descrição**: O sistema deve empregar arquitetura de (três) camadas: apresentação, negócios e dados.

**Requisitos de Tecnologia:** Relativos à tecnologia adota no desenvolvimento do sistema.

**[RNF002] – Recursos utilizados no desenvolvimento**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: | ◼ | Essencial | 🞎 | Importante | 🞎 | Desejável |

**Descrição**: O sistema é desenvolvida para a plataforma web, baseado nos frameworks Ionic, Bootstrap e AngulasJs. Para disponibilizar na plataforma mobile, será utilizado o framework Ionic que interpreta a parte web para mobile.

**[RNF003] – Banco de Dados**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: | ◼ | Essencial | 🞎 | Importante | 🞎 | Desejável |

**Descrição**: O sistema deve utilizar o sistema gerenciador de banco de dados Firebase, utilizando a arquitetura JSON tree. Sincronizando os dados em nuvem e criando persistência local.

* 1. Modelagem dos requisitos funcionais

Apresenta os requisitos a serem atendidos funcionalmente pelo sistema de uma forma simples através de diagramas e detalhamento de cada situação possibilitando a compreensão do comportamento do sistema pela perspectiva do usuário.

### 3.2.1 Atores

A seguir são apresentados exemplos da especificação de atores.

**CLIENTE:** Representa pessoas físicas que desejam consultar dados de estabelecimentos e/ou consumir algo pelo aplicativo.

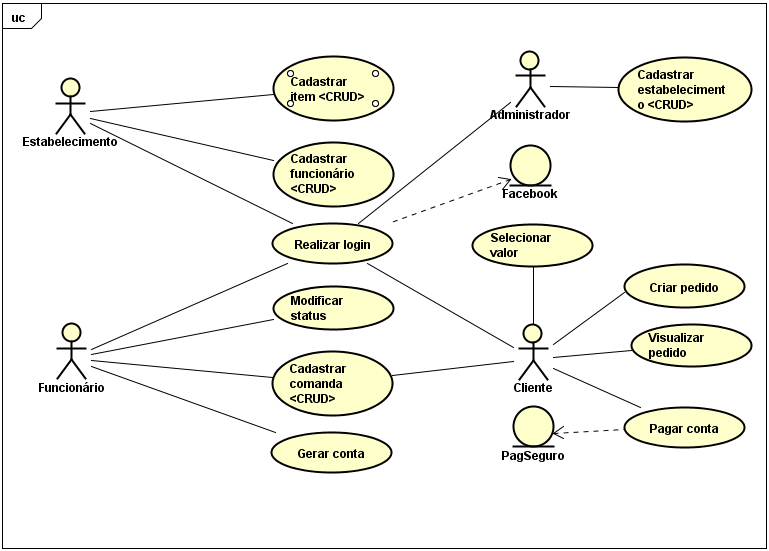
**FUNCIONÁRIO:** Representa funcionários contratados para atender os clientes, responsáveis por realizar as vendas. Porem existem dois tipos de funcionários, garçom e estabelecimento.

**ADMINISTRADOR:** Representa a pessoa física que realizará a manutenção da aplicação inteira.



### 3.2.2 Diagrama de Caso de Uso

Este diagrama ilustra todas as situações de uso que o usuário poderá ter com o aplicativo. A partir desta abstração é possível aumentar o nível de detalhe e verificar as interações que o usuário poderá ter em cada situação de uso. Nos próximos itens temos o detalhamento para cada caso de uso que mostram qual é o comportamento do aplicativo quando um desses casos de uso é realizado.



**3.2.3 Especificação do Caso de Uso**

Especificar um caso de uso significa detalhar o comportamento passo a passo do caso em questão e é muito útil para validar o entendimento do caso antes de começar a implementação. Abaixo temos todas as especificações dos casos de uso do modelo principal do sub-capítulo anterior.

|  |  |
| --- | --- |
| **CSU001 – Cadastrar estabelecimento** | |
| Sumário: | Cadastrar um novo estabelecimento no sistema |
| Ator Primário: | Administrador |
| **Pré-condição:**  **a.** O administrador deve estar logado no sistema.  **b.** O estabelecimento deve ter fornecido dados necessários para efetuar o cadastro, tais como: nome, CNPJ, Endereço, telefones e email.  **c.** O estabelecimento deve ter fechado um acordo de comprometimento com o Administrador. | |
| **Fluxo Principal**  **1-** O Administrador insere os dados cadastrais do estabelecimento no sistema. | |
| **Pós-condições:**   1. **O estabelecimento estará cadastrado no sistema.** | |
| **Requisitos : RF001** | |
| **Regras de Negócio:** RN1, RN2, RN5, RN7, RN9. | |

|  |  |
| --- | --- |
| **CSU002 – Cadastrar item** | |
| Sumário: | Cadastrar o cardápio do estabelecimento no sistema |
| Ator Primário: | Estabelecimento |
| **Pré-condição:**  **a.** O Estabelecimento deve estar cadastrado no sistema.  **b.** O Estabelecimento deve estar logado no sistema. | |
| **Fluxo Principal**  **1-** O Estabelecimento cadastra itens do cardápio.  **2-** O Estabelecimento insere uma breve descrição do item e seu valor. | |
| Fluxo Alternativo **(1): Modificar itens do cardápio**   1. O Estabelecimento modifica informações(descrição e/ou valor) dos itens. | |
| Fluxo Alternativo **(2): Consultar itens do cardápio**   1. O Estabelecimento consulta dados(descrição e/ou valor) dos itens. | |
| Fluxo Alternativo **(2): Apagar itens do cardápio**   1. O Estabelecimento remove os itens. | |
| **Pós-condições:**  **a.** Os itens serão salvos no sistema e ficarão disponíveis para os Funcionários e Clientes consultarem. | |
| **Requisitos : RF002** | |
| **Regras de Negócio:** RN1, RN2, RN5, RN7, RN9. | |

|  |  |
| --- | --- |
| **CSU003 – Cadastrar funcionário** | |
| Sumário: | Cadastrar o funcionário |
| Ator Primário: | Estabelecimento |
| **Pré-condição:**  **a.** O Estabelecimento deve estar cadastrado no sistema.  **b.** O Estabelecimento deve estar logado no sistema. | |
| **Fluxo Principal**  **1-** O Estabelecimento cadastra funcionários no sistema.  **2-** O Estabelecimento insere os dados pessoais( CPF, RG, nome completo, telefone e email) e insere o tipo de funcionário( garçom ou estabelecimento). | |
| Fluxo Alternativo **(1): Modificar dados do funcionário**   1. O Estabelecimento modifica dados pessoais( CPF, RG, nome completo, telefone e email) do funcionário. | |
| Fluxo Alternativo **(2): Consultar dados do funcionário**   1. O Estabelecimento consulta dados pessoais( CPF, RG, nome completo, telefone e email) do funcionário. | |
| Fluxo Alternativo **(2): Apagar funcionário**   1. O Estabelecimento remove funcionário. | |
| **Pós-condições:**  **a.** Os dados pessoais( CPF, RG, nome completo, telefone e email) do funcionário ficarão salvos no sistema. | |
| **Requisitos : RF003** | |
| **Regras de Negócio:** RN1, RN2, RN5, RN7, RN9. | |

|  |  |
| --- | --- |
| **CSU004 – Realizar login** | |
| Sumário: | Realizar login |
| Ator Primário: | Estabelecimento, Administrador, Cliente, Funcionário |
| **Pré-condição:**  **a.** O usuário deve ter uma conta no Facebook. | |
| **Fluxo Principal**  **1-** O usuário irá inserir o nome de usuário ou email, e sua senha.  **2-** O login será validado pelo sistema externo do Facebook. | |
| Fluxo de Exceção **(1):**  **a.** Se o nome do usuário/email ou a senha estiver errada a entrada no sistema será impedida. | |
| **Pós-condições:**  **a.** Dependendo do login aparecerá uma tela correspondente, tais como: tela de administrador, estabelecimento, funcionário e cliente. | |
| **Requisitos : RF004** | |
| **Regras de Negócio:** RN1, RN2, RN5, RN7, RN9. | |

|  |  |
| --- | --- |
| **CSU005 – Modificar status** | |
| Sumário: | Modificar status do estabelecimento |
| Ator Primário: | Funcionário |
| **Pré-condição:**  **a.** O Funcionário deve estar logado no sistema.  **b.** O Estabelecimento deve ter cadastrado seu Funcionário. | |
| **Fluxo Principal**  **1-** O funcionário modifica o status no sistema, o status pode estar como lotado, moderado e vazio. | |
| **Pós-condições:**  **a.**O status correspondente ficará disponível para os Clientes verem. | |
| **Requisitos : RF005** | |
| **Regras de Negócio:** RN1, RN2, RN5, RN7, RN9. | |

|  |  |
| --- | --- |
| **CSU006 – Cadastrar comanda** | |
| Sumário: | Cadastrar comanda |
| Ator Primário: | Funcionário |
| **Pré-condição:**  **a.** O Funcionário deve estar logado no sistema. | |
| **Fluxo Principal**  **1-** O Funcionário cria a mesa no sistema.  **2-** O Sistema irá gerar um QRcode referente a mesa.  **3-** O Funcionário registrará Clientes na mesa. | |
| Fluxo Alternativo **(1): Alterar dados da mesa**   1. O Funcionário poderá remover e incluir Clientes da mesa. 2. O Funcionário poderá inserir e remover Itens do cardápio na mesa. | |
| Fluxo Alternativo **(2): Consultar dados da mesa**   1. O Funcionário poderá consultar os dados da mesa, tais como: código, número, Clientes participantes, itens consumidos e valor total. | |
| Fluxo Alternativo **(2): Remover mesa**   1. O Funcionário poderá excluir a mesa. | |
| **Pós-condições:**   1. **O sistema dará continuidade ao pagamento indo para o RF010.** | |
| **Requisitos : RF006** | |

|  |  |
| --- | --- |
| **CSU007 – Gerar conta** | |
| Sumário: | Gerar conta |
| Ator Primário: | Funcionário |
| **Pré-condição:**   1. O Funcionário deve estar logado no sistema. | |
| **Fluxo Principal**  **1-** O funcionário seleciona o cliente.  **2-** O funcionário seleciona o valor a ser pago. | |
| **Pós-condições:**   1. **O sistema emitirá uma conta para o cliente pagar.** | |
| **Requisitos : RF007** | |
| **Regras de Negócio:** RN1, RN2, RN5, RN7, RN9. | |

|  |  |
| --- | --- |
| **CSU008 – Selecionar valor** | |
| Sumário: | Selecionar o valor a ser pago |
| Ator Primário: | Cliente |
| **Pré-condição:**  **a.** O cliente deve estar logado no sistema.  **b.** O cliente deve fazer parte de uma mesa.  **c.** A mesa deve ter realizado ao menos um pedido. | |
| **Fluxo Principal**  **1-** O cliente solicita a verificação do valor a ser pago no sistema.  **2-** O Sistema disponibiliza disponibiliza o valor dividido igualmente da mesa por cada integrante.  **3-** O sistema oferece a opção de pagar a porcentagem da casa.  4- O Sistema oferece a opção de selecionar o valor que o usuário deseja pagar e a opção de pagar apenas os itens consumidos por ele e uma opção de pagar um determinado valor que o usuário irá selecionar. | |
| Fluxo Alternativo **(1): Escolher por pagar somente os itens consumidos**   1. O sistema mostra uma lista com os itens consumidos e o valor total a ser pago. | |
| Fluxo Alternativo **(2): Escolher um determinado valor a pagar**   1. O usuário escolhe a quantia que irá pagar. | |
| **Pós-condições:**   1. **O sistema dará continuidade ao pagamento indo para o RF010.** | |
| **Regras de Negócio:** RN1, RN2, RN5, RN7, RN9. | |

|  |  |
| --- | --- |
| **CSU009 –Criar pedido** | |
| Sumário: | Criar um pedido |
| Ator Primário: | Funcionário |
| **Pré-condição:**   1. O Cliente deve estar logado no sistema. | |
| **Fluxo Principal**  **1-** O cliente visualiza os itens do cardápio.  **2-** O cliente seleciona o item desejado. | |
| **Requisitos : RF009** | |
| **Regras de Negócio:** RN1, RN2, RN5, RN7, RN9. | |

|  |  |
| --- | --- |
| **CSU0010 – Visualizar pedido** | |
| Sumário: | Visualizar um pedido |
| Ator Primário: | Funcionário |
|  | |
| Casos de Uso Associados: CSU001 | |
| **Pré-condição:**   1. O Funcionário deve estar logado no sistema. | |
| **Fluxo Principal**  **1-** O funcionário seleciona o item escolhido.  **2-** O Funcionário visualiza o pedido selecionado. | |
| **Requisitos : RF010** | |
| **Regras de Negócio:** RN1, RN2, RN5, RN7, RN9. | |

|  |  |
| --- | --- |
| **CSU011 – Pagar conta** | |
| Sumário: | Pagar conta pelo PagSeguro |
| Ator Primário: | Cliente |
|  | |
| Casos de Uso Associados: CSU001 | |
| **Pré-condição:**   1. O Cliente deve estar logado no sistema. 2. O Cliente deve ter conexão com a internet. | |
| **Fluxo Principal**  **1-** O Cliente seleciona a forma de pagamento como o PagSeguro. | |
| **Pós-condições:**   1. **O sistema redirecionará o cliente para o sistema externo do PagSeguro.** | |
| **Requisitos : RF011** | |
| **Regras de Negócio:** RN1, RN2, RN5, RN7, RN9. | |

1. DESENVOLVIMENTO DO PROJETO

Este capítulo tem como objetivo analisar, detalhar e propor uma solução geral do sistema, sob o ponto de vista de negócio, de acordo com os requisitos levantados e validados no Capítulo 3. Além disso, é apresentado o refinamento da proposta de solução geral do sistema, apresentando a solução técnica, incluindo a visão de projeto e implementação, a arquitetura e a tecnologia utilizada.

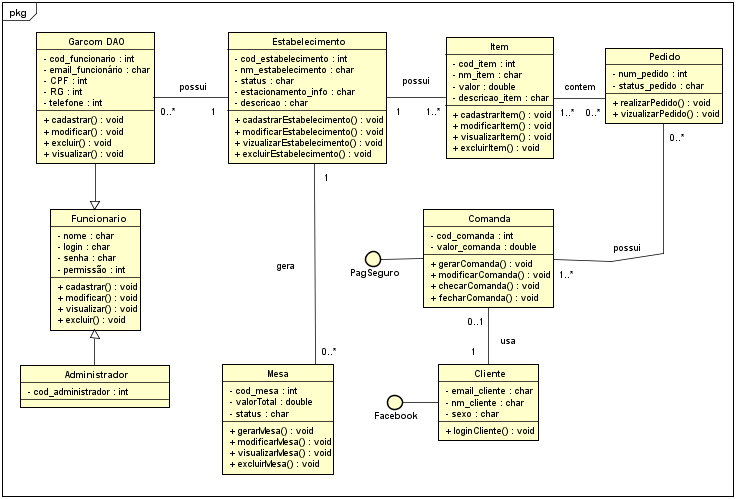
* 1. Análise

Neste item deve ser apresentado o modelo do domínio, visão de negócio, que representa um primeiro modelo conceitual do diagrama de classes. Posteriormente, esse diagrama deve ser validado, refinado e complementado para compor o diagrama de classes de projeto.

* + 1. Diagrama de Classes de Análise (Visão de Negócio)

O diagrama de classes deve possuir todas as classes identificadas do sistema, deve conter os atributos e métodos de cada classe, e os relacionamentos entre elas. Na Figura 4 é apresentada a notação básica de um diagrama de classes.

Figura 9 - Diagrama de Classes

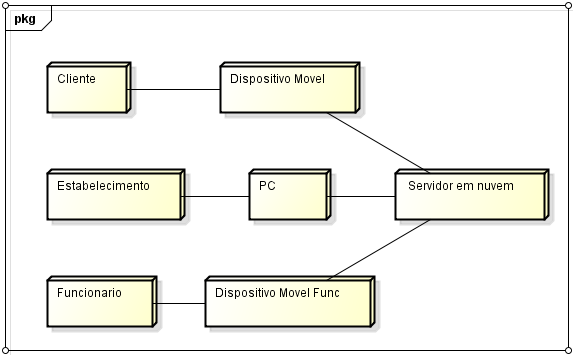


* 1. Projeto

Neste item deve ser apresentada a arquitetura de infraestrutura do sistema, demonstrando o tipo de arquitetura que será utilizada (por exemplo, cliente/servidor de n-camadas, MVC, ...), a configuração de hardware, de rede, de software, padrões de projeto, componentes específicos (dll, jar, ...) e componentes externos a serem utilizados, bem como o dimensionamento mínimo de conexões.

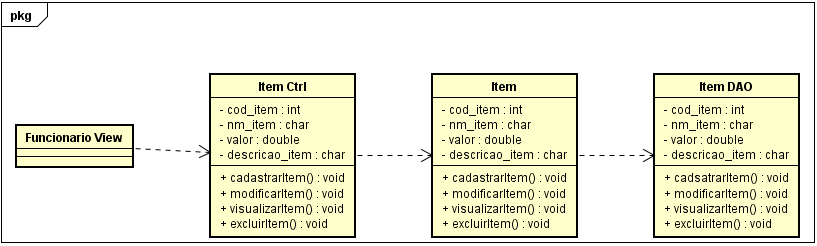
4.2.1Arquitetura do Sistema

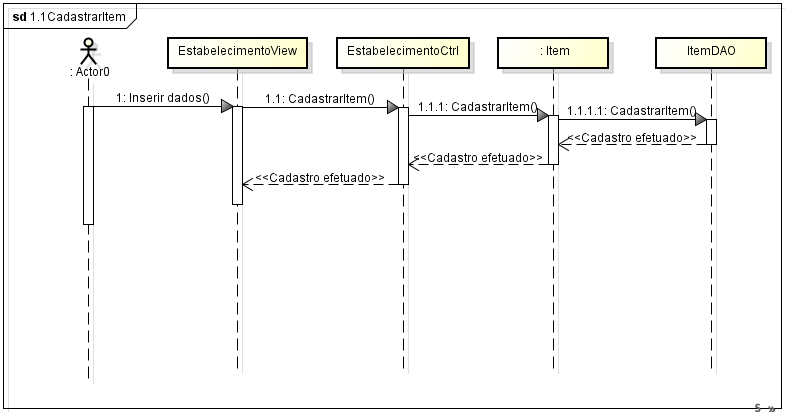
Para a representação da arquitetura de infraestrutura pode-se utilizar uma figura ilustrativa ou o diagrama de implantação. Na Figura 5 é apresentada a notação básica de um diagrama de implantação.

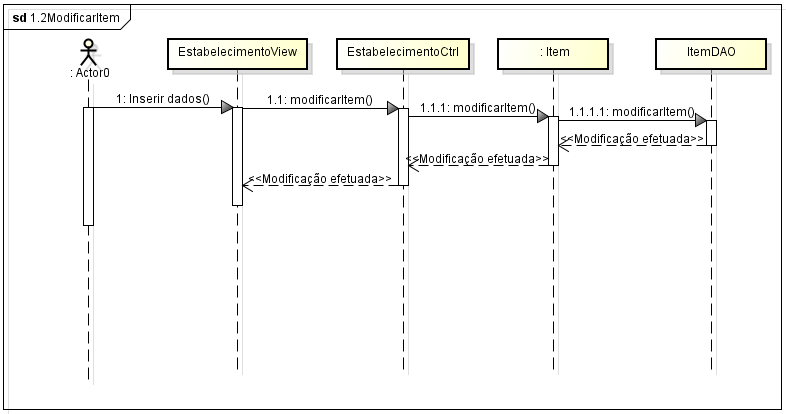


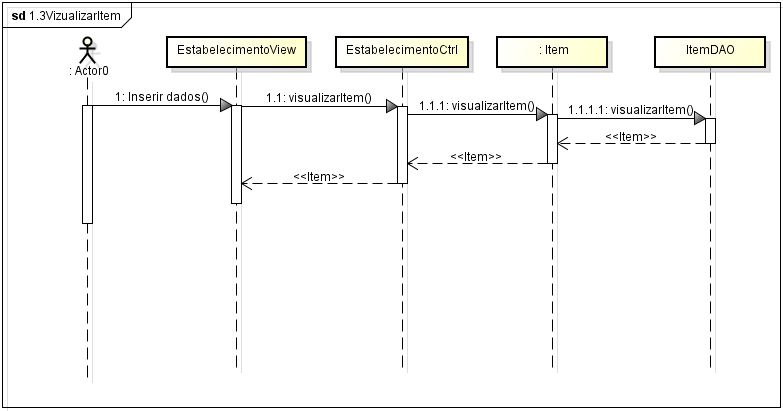
4.2.2Diagrama de Classes de Projeto por Caso de Uso

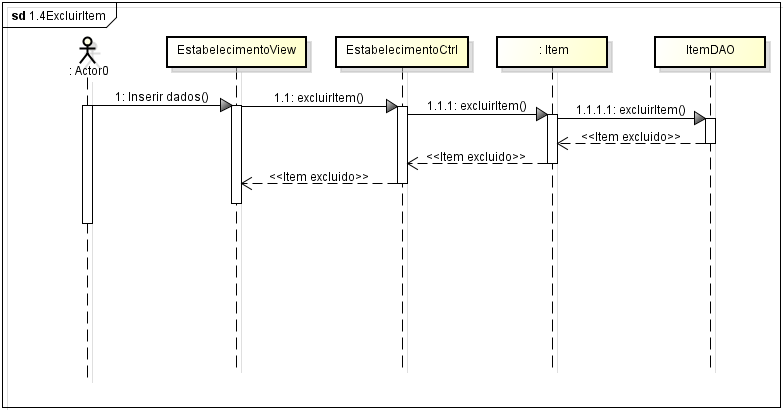
Um diagrama de sequência representa interações de objetos organizadas em uma sequência temporal, apresentando os objetos que participam da interação e a sequência das mensagens trocadas. O diagrama de sequência deve validar o diagrama de classes e vice-versa. Nas figuras a seguir ilustrarão cada função de cada classe do projeto:

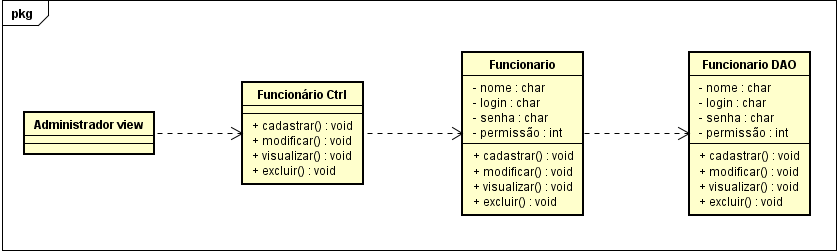


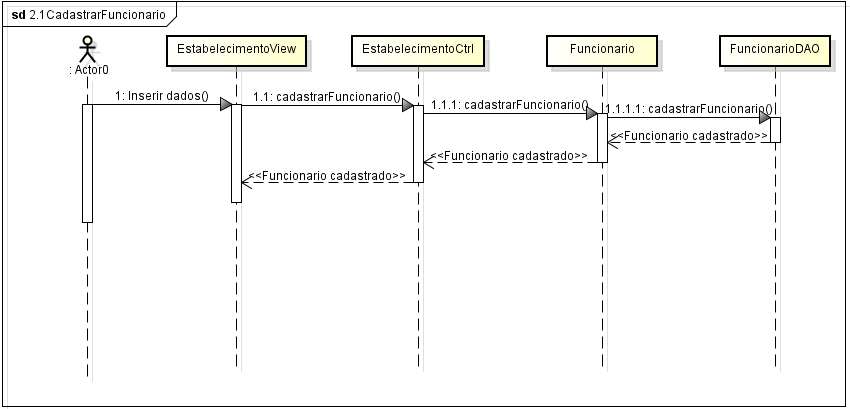


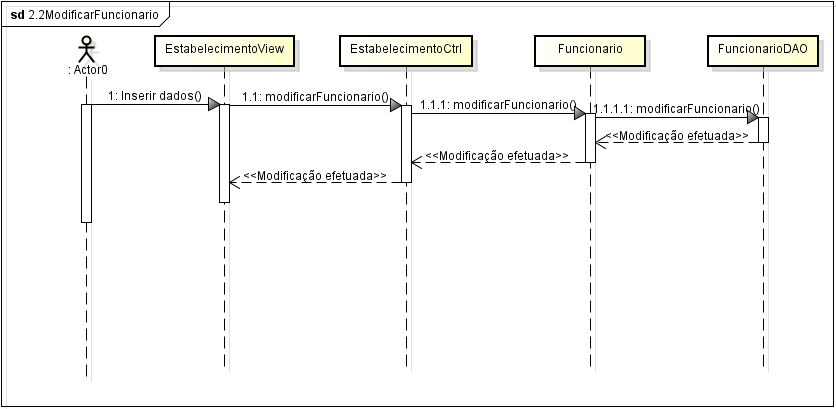


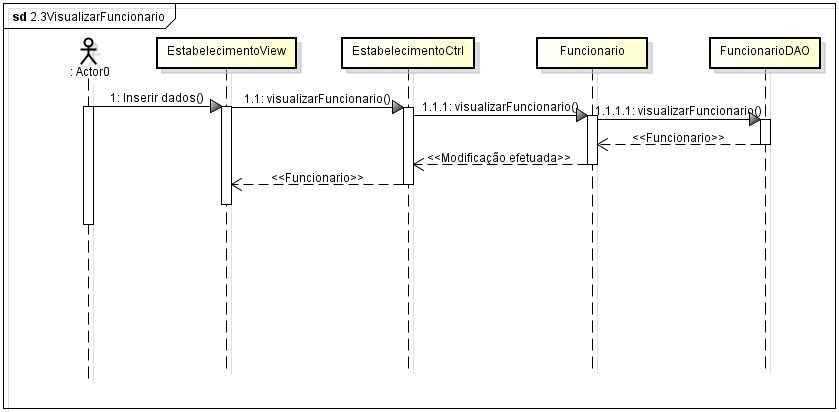


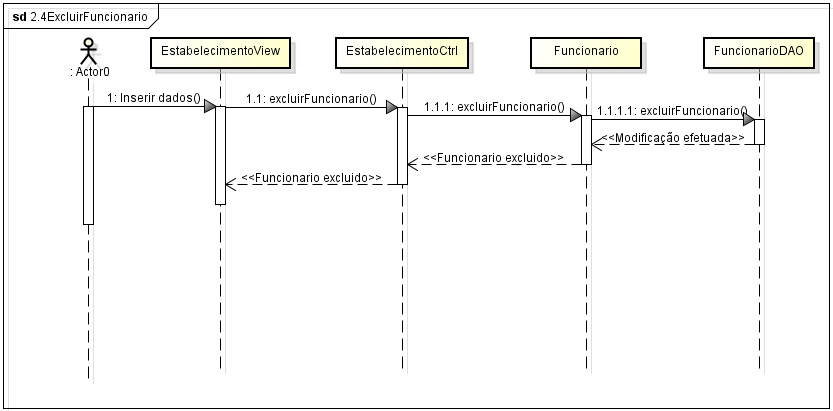


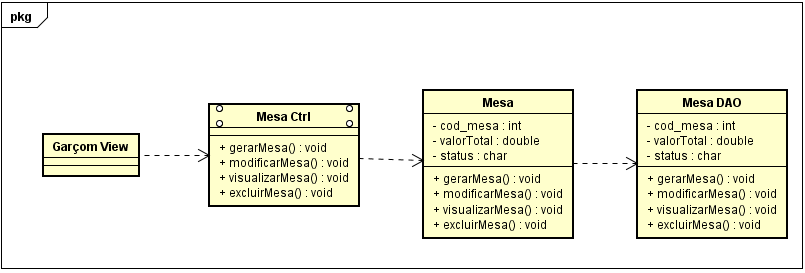


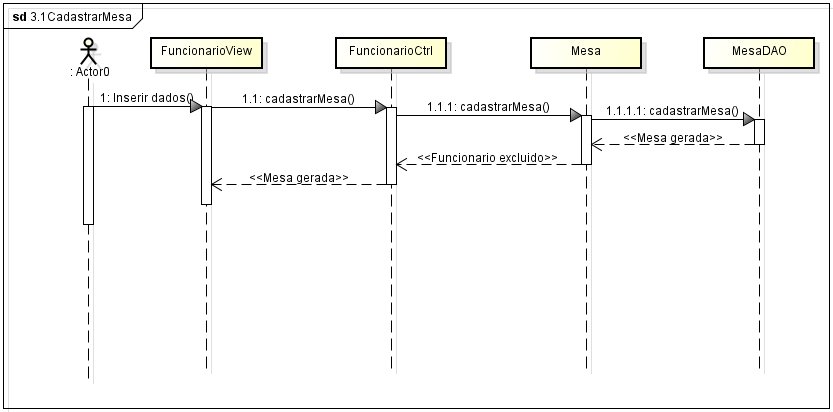


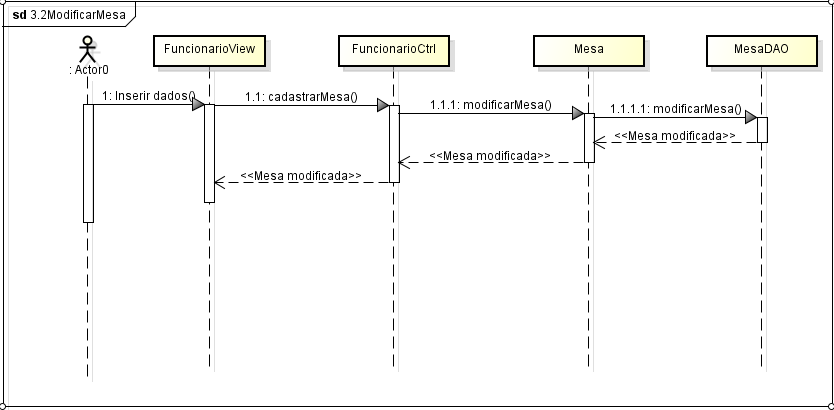


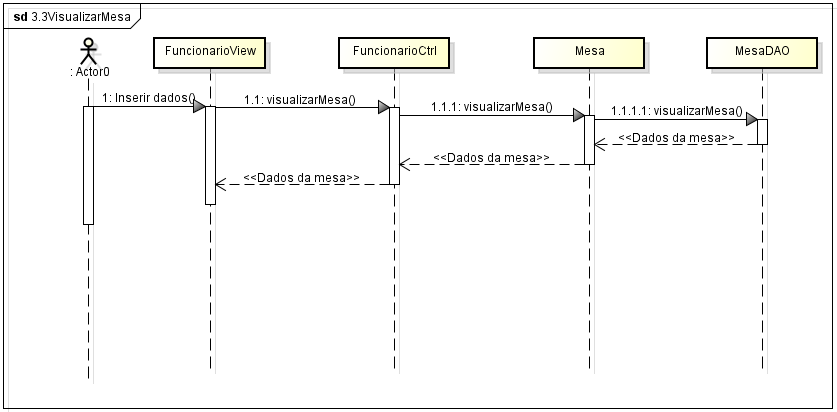


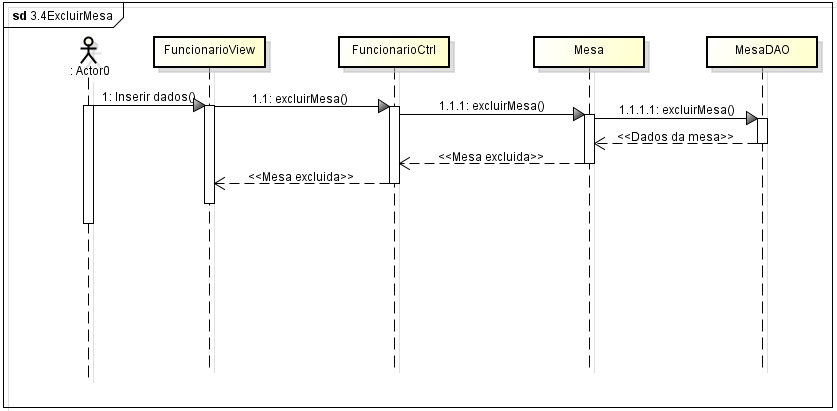


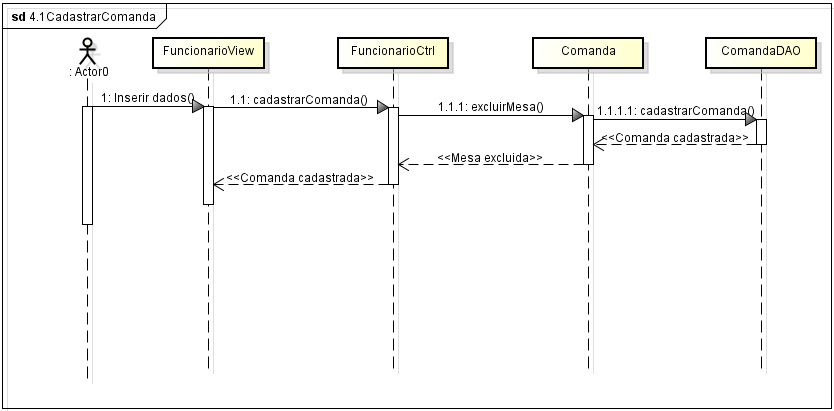


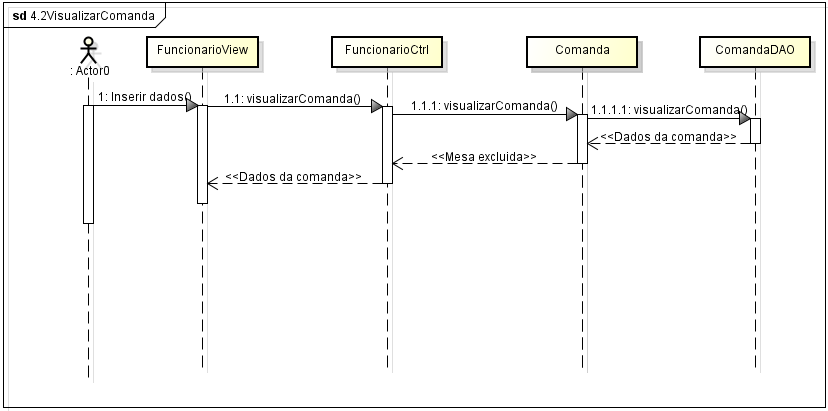


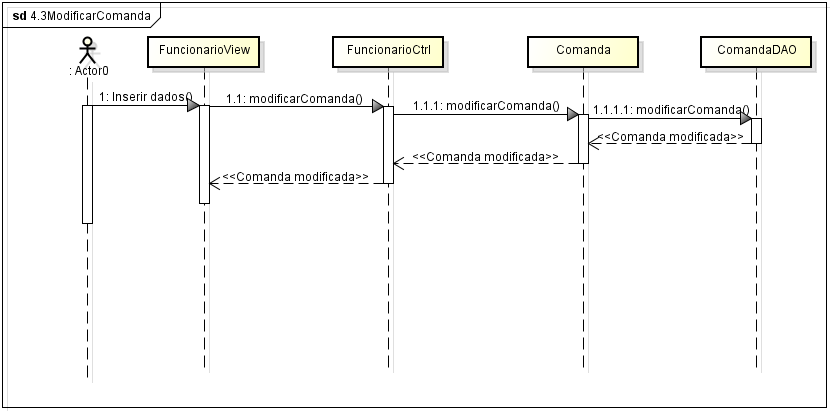


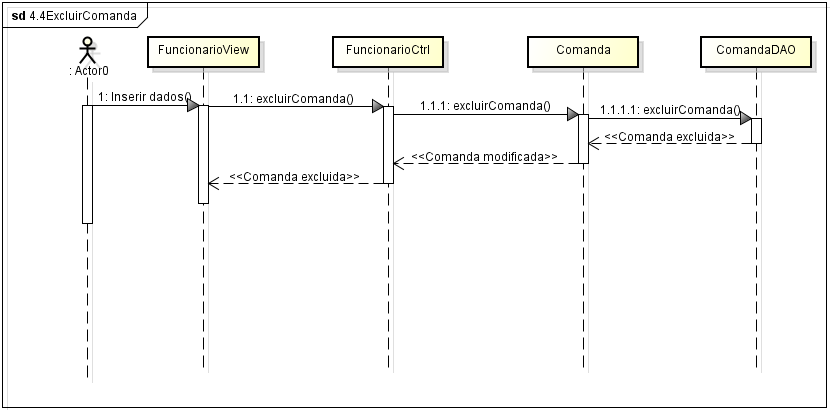


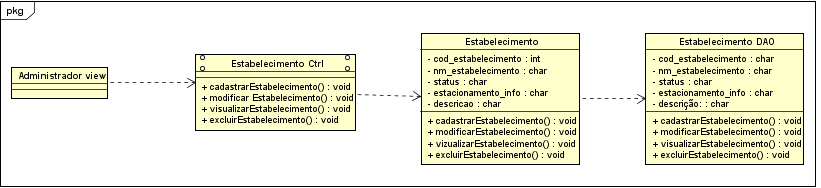


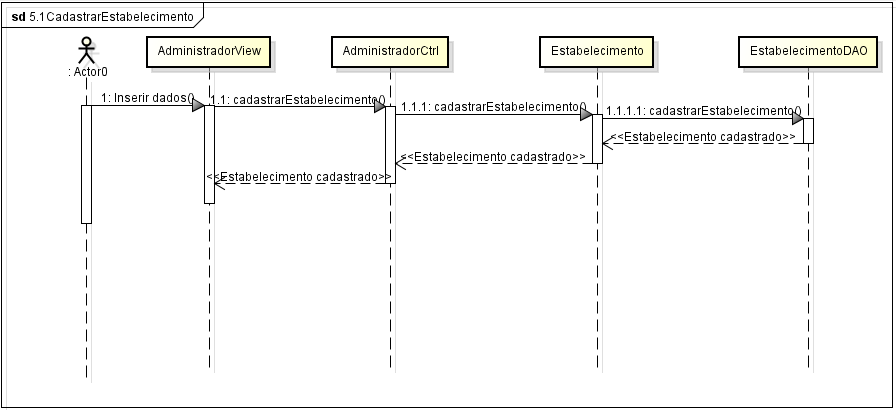


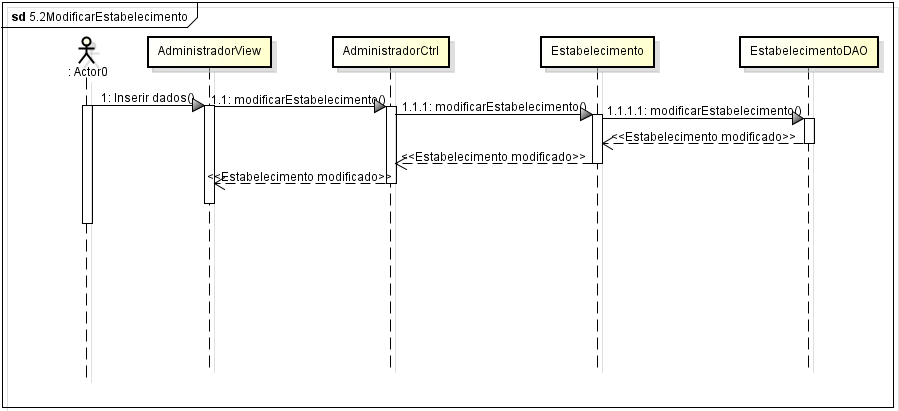


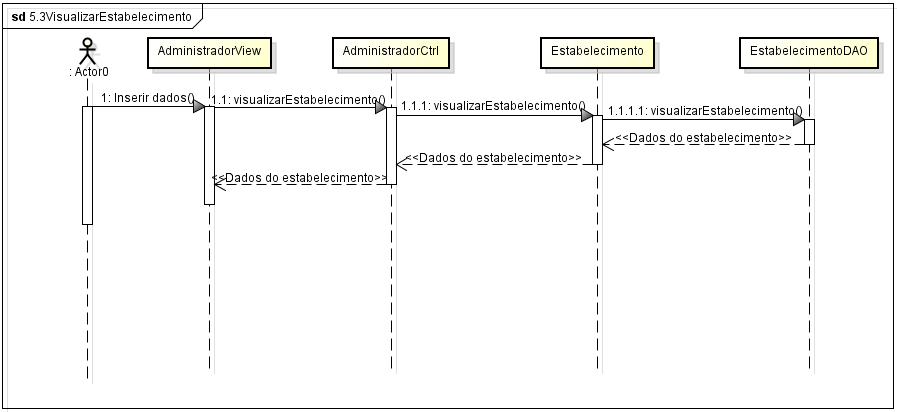


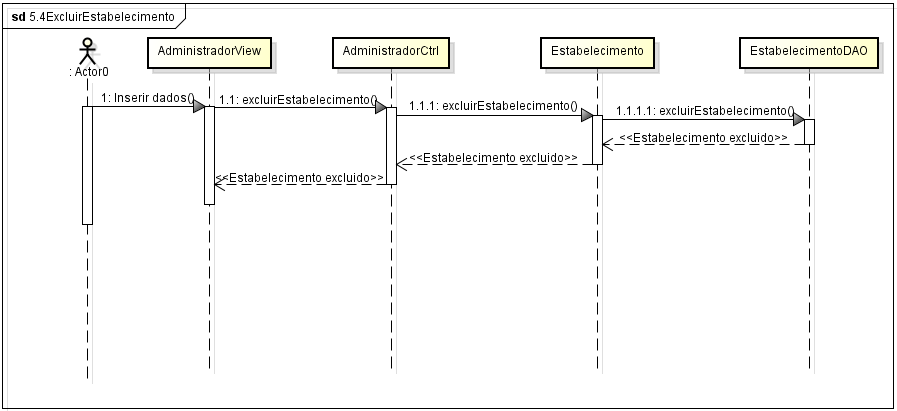


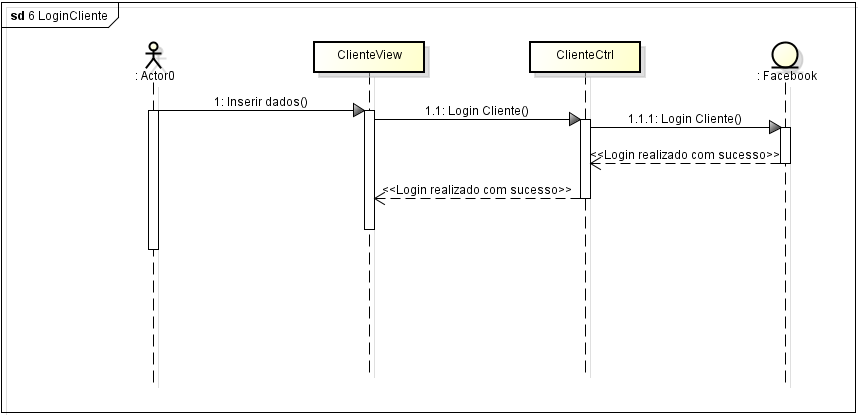


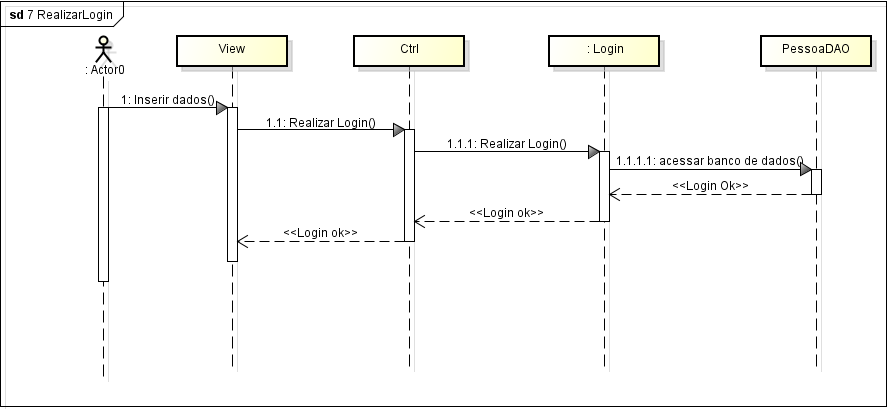


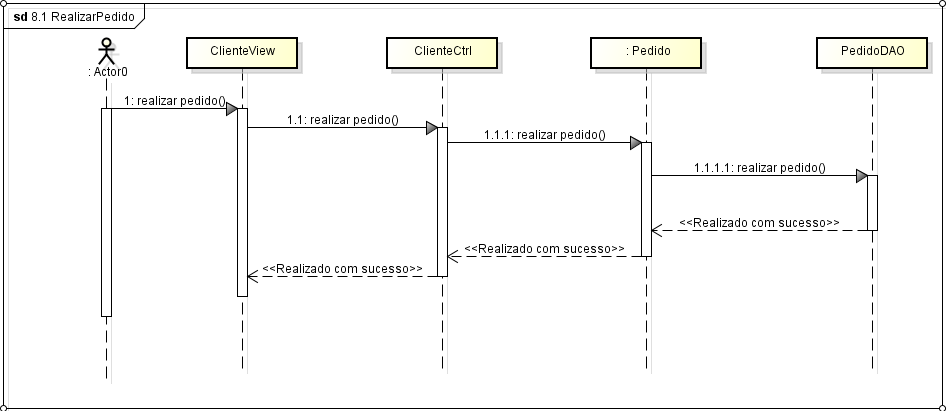


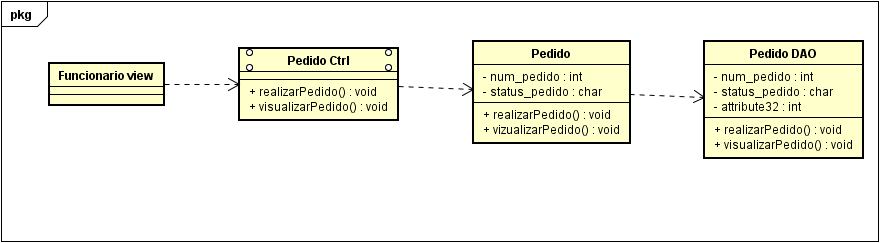


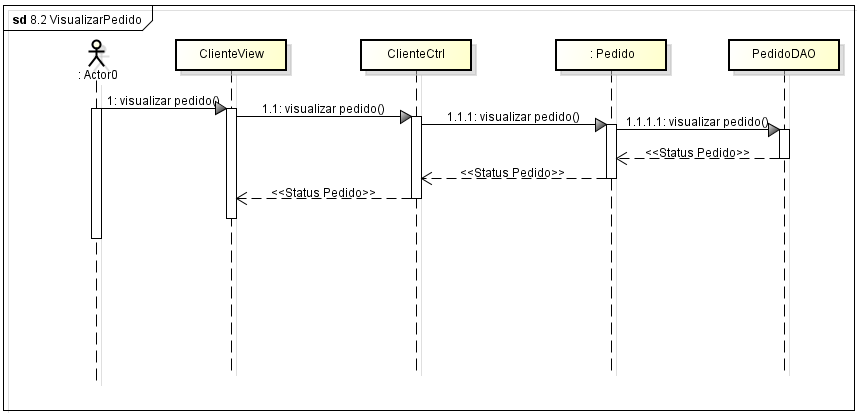


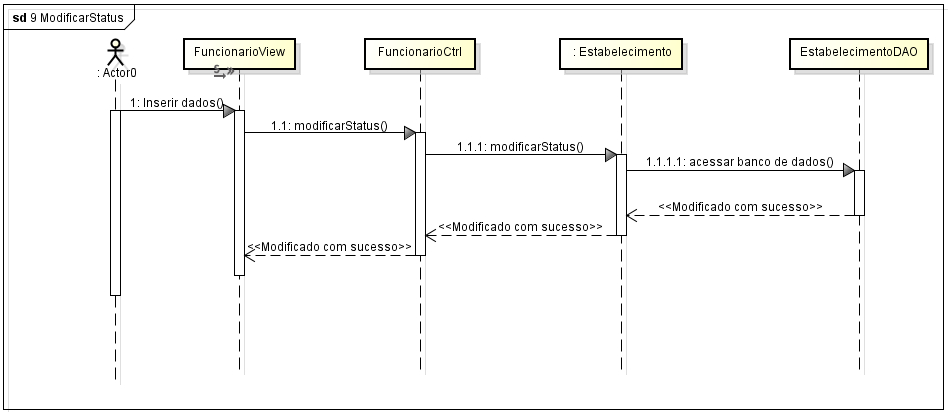






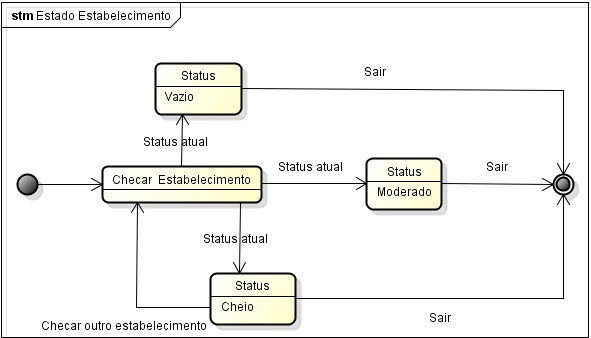


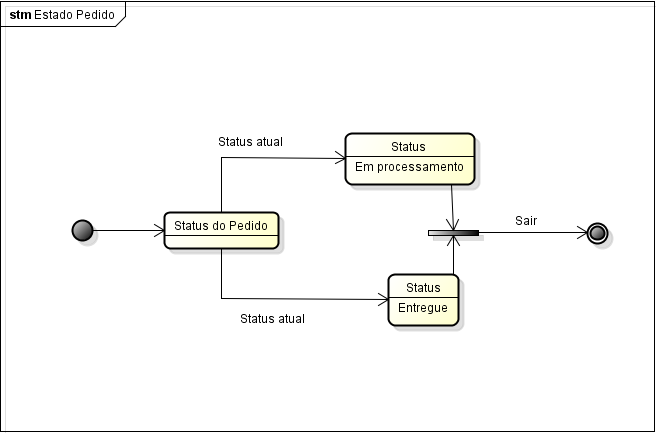




* + 1. Diagrama de estados

O diagrama de estados especifica as sequências de estados pelas quais o objeto pode passar durante seu ciclo de vida em resposta a eventos. Na Figura 8 apresentada a notação básica de um diagrama de estados. **Esse diagrama somente deverá ser elaborado se houver necessidade e agregar valor ao projeto.**





APÊNDICE A – PLANO E EXECUÇÃO DE TESTES

# PLANO DE TESTES

# 5.2 Introdução

### 5.2.1 Finalidade

Este Plano de Teste referente ao Sistema Presentão atende aos seguintes objetivos:

1. Identifica os itens que devem ser inspecionados pelos testes.
2. Identifica a motivação e as idéias subjacentes às áreas de teste a serem abrangidas.
3. Descreve a abordagem de teste que será usada.
4. Identifica os recursos necessários e fornece uma estimativa dos esforços de teste.
5. Lista os elementos liberados do projeto de teste.

### 5.2.2 Escopo

Este plano de teste abordará testes de unidade e sistema do Sistema NugetX para a iteração C1 da Fase de Construção do projeto.

Os testes visam comprovar a qualidade funcional e usabilidade dos seguintes casos de uso:

* Login de Usuário;
* Selecionar Valor;
* Pagar conta (Sistema externo).

## Missão de avaliação e motivação dos testes

Esta atividade tem como missão avaliar na iteração atual o maior número de erros possível e verificar a conformidade com a especificação (requisitos e design), avaliando também os riscos da qualidade perceptível.

### Motivadores dos testes

Os testes a serem realizados têm como motivadores os requisitos funcionais do projeto, descritos pelos casos de uso implementados até esta iteração.

## Itens-alvo dos testes

A tabela abaixo lista os itens que serão sujeitos a testes funcionais e a priorização de cada um deles:

|  |  |  |
| --- | --- | --- |
| Item-alvo | Fator de Risco (Impacto) | Ordem de Prioridade |
| Login de usuário | 3 | 3º. |
| Selecionar Valor | 4 | 2º. |
| Pagar conta | 5 | 1º. |

O Fator de Risco (Impacto) de um item-alvo refere-se, numa escala crescente de 1 a 5, ao impacto que será causado no negócio caso o item não funcione adequadamente.

## Resumo dos testes planejados

### Resumo das Inclusões dos Testes

Os principais testes planejados para a iteração atual são:

* Teste funcional do caso de uso “Login de Usuário”;
* Teste funcional do caso de uso “Selecionar Valor”;
* Teste funcional do caso de uso “Pagar conta”.

### Resumo dos Outros Candidatos a Possível Inclusão

A seguir temos um resumo de áreas de teste cuja avaliação e investigação poderão ser úteis, mas que ainda não foram suficientemente pesquisadas:

* Teste de performance;
* Teste de estrutura;
* Teste de instalação;
* Teste de configuração;
* Teste de Interface com usuário.

### Resumo de Exclusão de Testes

### A seguir temos um resumo de nivel superior dos possiveis testes que poderiam ter sido conduzidos, as que foram excluidos deste plano de testes:

| Testes Excluídos | Razão |
| --- | --- |
| Criptografia de Senha | Não há recursos suficientes no momento para realizar este teste especifico. |

## Necessidades ambientais

### Hardware básico do sistema

Os conjuntos de tabelas a seguir apresentam os recursos do sistema necessários ao esforço de teste descrito neste Plano de Teste.

| Recursos do Sistema | | |
| --- | --- | --- |
| Recurso | Quantidade | Nome e Tipo |
| Rede | 1 | Rede para conexão a banco em nuvem |
| PCs de Teste | 1 | Estação de trabalho com navegador e acesso à Rede |

### Elementos de software básicos do ambiente de teste

| Nome do Elemento de Software | Versão | Tipo e Outras Observações |
| --- | --- | --- |
| Windows 7 ou superior | Service Pack 1 | Sistema Operacional |

## Responsabilidades, perfil da equipe e necessidades de treinamento

### Pessoas e papéis

| Recursos Humanos | | |
| --- | --- | --- |
| Papel | Recursos Mínimos Recomendáveis  (número de papéis alocados em tempo integral) | Responsabilidades ou Comentários Específicos |
| Gerente de Testes | 1 | Supervisiona o gerenciamento.  Estas são as responsabilidades:   * Planejamento e logística * Combinar missão * Identificar motivadores * Adquirir recursos apropriados * Apresentar relatórios de gerenciamento * Defender os interesses do teste * Avaliar a eficiência do esforço de teste |
| Analista de Teste | 1 | Identifica e define os testes específicos a serem conduzidos.  Estas são as responsabilidades:   * Identificar ideias de teste * Definir detalhes dos testes * Determinar os resultados dos testes * Documentar solicitações de mudança * Avaliar a qualidade do produto |
| Designer de Teste | 1 | Define a abordagem técnica referente à implementação do esforço de teste.  Estas são as responsabilidades:   * Definir a abordagem dos testes * Definir a arquitetura de automação de teste * Verificar as técnicas de teste * Definir os elementos de testabilidade * Estruturar a implementação dos testes |
| Testador | 1 | Implementa e executa os testes.  Estas são as responsabilidades:   * Implementar os testes e os conjuntos de testes * Executar os conjuntos de testes * Registrar os resultados * Analisar as falhas dos testes e possibilitar a recuperação posterior * Documentar incidentes |

## Riscos, dependências, suposições e restrições

O risco mais evidente na execução deste Plano de Teste é a falta de Rede, uma vez que para a utilização do sistema e realizar todos os testes propostos é necessário ter conexão à internet.