**Universidade Estadual de Maringá**

**Departamento de Informática**

**Curso: Informática**

**Disciplina: Projeto de Sistemas de Software Ano: 2015**

**ESPECIFICAÇÃO DO SISTEMA DE GERENCIAMENTO DA PADARIA GEGÊ**

**Grupo:**

**83154 Igor Santana**

**83196 Leonardo Catharin**

**84261 Rafael Mangolin**

Sumário

1. Introdução ................................................................................................... 3
2. Concepção .................................................................................................. 3
   1. Divisão dos requisitos ........................................................................... 3
   2. Glossário de termos ............................................................................ 15
   3. Estudo de viabilidade .......................................................................... 16
      1. Contribuições para a empresa ...................................................... 16
      2. Custo X Benefício ......................................................................... 16
      3. Viabilidade técnica ........................................................................ 16
      4. Viabilidade legal ............................................................................ 17
      5. Alternativas: (avaliação de alternativas) ....................................... 17
   4. Avaliação de risco ............................................................................... 18
      1. Riscos ........................................................................................... 18
      2. Processo no risco “incerteza do cliente em questão aos requisitos”

18

* 1. Modelo de Domínio ou Diagrama de objetos de negócios .................. 19
  2. Pacotes ............................................................................................... 20
     1. Diagrama de pacotes da Área de negócio .................................... 20
     2. Diagrama de pacotes de análise e serviço ................................... 20
  3. Arquitetura de Software ....................................................................... 20

1. Elaboração ................................................................................................ 21
   1. Modelo de Casos de Uso .................................................................... 21
      1. Identificação dos Casos de Uso ................................................... 21
      2. Priorização dos Casos de Uso ..................................................... 23
      3. Diagrama de Casos de Uso .......................................................... 24
      4. Detalhamento dos Casos de Uso ................................................. 25
   2. Modelo de Análise ............................................................................... 28
   3. Planejamento das iterações ................................................................ 29
      1. Iteração Nº1 – CRUD’S ................................................................ 29
      2. Iteração Nº2 .................................................................................. 30
      3. Iteração Nº3 .................................................................................. 30
      4. Iteração Nº4 .................................................................................. 30
   4. Diagrama de Classes do projeto ......................................................... 31
2. Frameworks .............................................................................................. 31

2. CONCEPÇÃO

## 2.1. Divisão dos requisitos

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Responsável:** Leonardo Catharin | | | | | | | | |
| **Identificador:** RFCE001- Cadastrar produto.  **Descrição:** O sistema deve receber os dados de um novo produto, verificar se já existe esse produto cadastrado. Caso esse produto já esteja cadastrado deverá exibir mensagem informando que o produto já cadastrado, caso não o produto será incluído no banco de dados do estoque e será exibido uma mensagem de confirmação de cadastros.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | *Gerência* | | | | | | | **Requisitos associados:** | RFCE004 | | | | | | | | | | | | | | |
| **Identificador:** RFCE002- Alterar dados do cadastro de produto.  **Descrição:** O sistema deverá receber o campo a ser alterado, o dado alterado e o cadastro que deve ocorrer a alteração. Caso o produto não esteja cadastrado deverá ser exibido uma mensagem informando que o produto não está cadastrado. Caso esteja deverá ser feito a alteração indicada no banco de dados e exibida uma tela alteração bem sucedida. | | | | | | | | |
|  | **Prioridade:** |  | Essencial | **X** | Importante |  | Desejável |  |
| **Ator(es):** | *Gerência* | | | | | |
| **Requisitos associados:** | RFCE004 | | | | | |

|  |
| --- |
| **Responsável:** Leonardo Catharin |
| **Identificador:** RFCE009- Cadastrar fornecedor.  **Descrição:** O sistema deve receber os dados de um novo fornecedor, verificar se já existe esse fornecedor cadastrado. Caso esse fornecedor já esteja cadastrado deverá exibir mensagem informando que o fornecedor já cadastrado, caso não o fornecedor será incluído no banco de dados e será exibido uma mensagem de confirmação de cadastros.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | *Gerência* | | | | | | | **Requisitos associados:** | RFCE004 | | | | | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Identificador:** RFCE003- Excluir cadastro de produto.  **Descrição:** O sistema deverá receber a identificação do cadastro que será excluir e verificar se esse cadastro existe no banco de dados. Caso o cadastro não exista deverá ser exibido uma mensagem de cadastro inexistente. Caso o cadastro exista ele e excluído do banco de dados e exibido uma mensagem de confirmação de exclusão.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** |  | Essencial | **X** | Importante |  | Desejável | | **Ator(es):** | *Gerência* | | | | | | | **Requisitos associados:** | RFCE004 | | | | | | |
| **Identificador:** RFCE004- Verificar existência de cadastro.  **Descrição:** Recebe a identificação do cadastro, verifica no banco de dados e retorna se ele existe ou não.   |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | *Gerência* | | | | | | | **Requisitos associados:** | *RFCE001, RFCE002, RFCE003.* | | | | | | |
| **Identificador:** RFCE005- Aumentar a quantidade dos produtos no estoque.  **Descrição:** Recebe uma lista de produtos e suas respectivas quantidades que devem ser aumentadas. Atualiza a quantidade do produto que está armazenada acrescentando ao estoque a quantidade que chegou na encomenda. Se não for encontrado o produto no banco de dados, deve-se se enviada uma mensagem para o setor de gerência informando que deve ser cadastrado esse produto e que esse ele deve, após ser cadastrado, ter sua quantidade atualizada.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | *Almoxarifado* | | | | | | | **Requisitos associados:** |  | | | | | | |
| **Identificador:** RFCE006- Diminuir a quantidade dos produtos no estoque. **Descrição:** Recebe uma lista de identificação de produtos e suas respectivas quantidades para ser dada a baixa. Acessa o estoque e vai executando a diminuição da quantidade dos produtos no estoque.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | *Caixa e produção.* | | | | | | | **Requisitos associados:** | RFCE007 | | | | | | |
| **Identificador:** RFCE007- Verificar quantidade dos produtos no estoque. |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Descrição:** Recebe a identificação de um produto e a quantidade requisitada, e verifica a quantidade do produto, comparando a quantidade que tem armazenada com a quantidade mínima de estoque do produto. Se a quantidade armazenada, após ser feita baixa que foi solicitada, for menor que a quantidade mínima do produto é enviado para quem controla o estoque uma mensagem informando que o produto está acabando. Se a quantidade em estoque for menor a solicitada e retornada uma mensagem para o setor que solicitou que o produto está em falta e só possui a quantidade armazenada. Se não for encontrado o produto no estoque, deve ser informado por meio de mensagem para o setor que originou a baixa que esse produto não existe no estoque e é enviado para outro processo que vai diminuir a quantidade no estoque o nome do produto e a quantidade restante do estoque. Se quantidade armazenada for maior que a quantidade requisitada e enviado para o processo eu vai dar baixa no estoque a identificação do produto e a quantidade que foi requisitada.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | *Caixa e produção.* | | | | | | | **Requisitos associados:** | RFVEN012*, RFCE006 e* RFVEN013 | | | | | | |
| **Identificador:** RFCE008- Armazenar pedido de compra.  **Descrição:** Recebe as informações do pedido, entre ela os produtos e sua quantidades, seus preços de compra e as informações para efetuar o pagamento do pedido. Cria um documento de pedido e o armazena no banco de dados e envia para o gerador de conta de pedido a informação que o pedido foi feito.   |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | *Almoxarifado* | | | | | | | **Requisitos associados:** | RFCF003 | | | | | | |
| **Identificador:** RFPRO001- Criar ordem de produção  **Descrição:** O sistema deve receber informações sobre o produto a ser manufaturado e seu destinatário, armazená-las no banco de dados como ordens de produção e informá-las à produção.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | *Produção* | | | | | | | **Requisitos associados:** | RFPRO002 | | | | | | |
| **Identificador:** RFPRO002- Processar pedido  **Descrição:** O sistema deve receber dados de encomenda ou de produto em falta no setor de vendas a ser manufaturado e encaminhar estas informações para que a ordem de produção seja criada.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | *Produção* | | | | | | | **Requisitos associados:** | RFPRO001, RFVEN011, RFENC004, | | | | | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Identificador:** RFPRO003- Verificar destino dos produtos  **Descrição:** O sistema deve receber a sinalização de que os produtos solicitados forma produzidos e encaminhar a sinalização para o setor de vendas ou de encomendas, de acordo com qual deles solicitou a manufatura do produto.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | Produção | | | | | | | **Requisitos associados:** | RFVEN013, RFENC002. | | | | | | |
|  |
| **Responsável:** Igor Santana |
| **Identificador:** RFCF001 - Gerar Conta de Encomenda  **Descrição:** O sistema deverá receber do funcionário dados da transação das encomendas que foram realizadas, tais como o Cliente que efetuou o pedido e qual o valor que foi pago. Caso a transação já tenha sido incluída, informar ao usuário que ela já se encontra no banco de dados.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | *Funcionário responsável pela encomenda.* | | | | | | | **Requisitos associados:** | RFCF011. | | | | | | |
| **Identificador:** RFCF002- Gerar Balanço do Caixa  **Descrição:** O sistema deverá receber do caixa informações sobre os produtos que foram vendidos, tais como Nome, Quantidade e Preço, para que se tenha um controle de todo o dinheiro que foi recebido no caixa, que será mostrado no Relatório de Despesas.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | *CAIXA* | | | | | | | **Requisitos associados:** | *RFCF009* | | | | | | |
| **Identificador:** RFCF003 - Gerar Pedido de Compra  **Descrição:** Os pedidos de compra vêm do Controle de Estoque, informando quais os produtos que deverão ser adquiridos. Com o nome do produto que deverá ser adquirido, será feita uma busca no banco de dados para que a quantidade que deverá ser feito o pedido conste no relatório.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | *Gerencia* | | | | | | | **Requisitos associados:** | *RFCF011,* RFCE008 | | | | | | |
| **Identificador:** RFCF004 - Gerar Folha Salarial da Empresa  **Descrição:** O sistema recebe do Pagamento de Funcionários as informações necessárias para que uma nova conta dos salários seja aberta, pedindo para que o Ator informe para qual funcionário será efetuado o pagamento e o valor do pagamento do funcionário. |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável |  |
| **Ator(es):** | *Gerência* | | | | | |
| **Requisitos associados:** *RFCF008, RFCF010.* | | | | | | |
| **Identificador:** RFCF005 - Selecionar tipo de relatório  **Descrição:** O Ator deverá escolher qual tipo de relatório que será gerado para que as informações sejam interpretadas na tela. Caso seja selecionado qualquer uma das opções, um relatório detalhado seria mostrado na tela com opções para imprimir ou salvar em algum formato de texto desejado pelo Ator.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** |  | Essencial | **X** | Importante |  | Desejável | | **Ator(es):** | *Gerência* | | | | | | | **Requisitos associados:** | *RFCF004, RFCF003, RFCF002, RFCF001.* | | | | | | | | | | | | | | |
| **Identificador:** RFCF006- Gerar relatório dos pagamentos dos funcionários **Descrição:** Função que realiza a organização e processamento das atividades que o funcionário realiza na empresa, que possa gerar um relatório para futuramente gerar o salário.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** |  | Essencial | **X** | Importante |  | Desejável | | **Ator(es):** | *Gerência* | | | | | | | **Requisitos associados:** | *RFCF005, RFCF004.* | | | | | | | | | | | | | | |
| **Identificador:** RFCF007-Gerar relatório geral  **Descrição:** Tem como função realizar um relatório superficial de todas as atividades geradas na empresa para fornecer a gerência uma visão geral da situação da empresa.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** |  | Essencial |  | Importante | **X** | Desejável | | **Ator(es):** | *Gerencia, funcionários responsáveis por cada setor.* | | | | | | | **Requisitos associados:** | *RFCF005* | | | | | | | | | | | | | | |
| **Identificador:** RFCF008-Gerar relatório de despesas  **Descrição:** Com as informações de todas as despesas que acontecem na empresa, essa funcionalidade faz um relatório que reuni todas essas despesas para que a gerência tenha o controle das mesmas.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** |  | Essencial | **X** | Importante |  | Desejável | | **Ator(es):** | *Gerência* | | | | | | | **Requisitos associados:** | *RFCF005* | | | | | | | | | | | | | | |
| **Identificador:** RFCF009- Criar conta de balanço de fluxo de caixa.  **Descrição:** Após receber os dados de fluxo de caixa essa funcionalidade deve criar uma variável para ser armazenada como se fosse uma conta. | | | | | | | | |
|  | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável |  |
|  | **Ator(es):** | *Caixa* | | | | | |  |
| **Requisitos associados:** *RFCF011, RFCF002,* | | | | | | |
| **Identificador:** RFCF010- Criar conta dos salários  **Descrição:** Para dar baixa no salário como despesa é necessário usar essa função para criar uma variável que representa o salário pago aos funcionários que servira de informação para o sistema como gasto da empresa.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | *Gerência* | | | | | | | **Requisitos associados:** | *RFCF011, RFCF006* | | | | | | | | | | | | | | |
| **Identificador:** RFCF011- Adicionar conta  **Descrição:** Para o setor financeiro é importante ter essa funcionalidade, pois essa funcionalidade irá inserir as contas que a empresa tem no banco de dados de contas para que a gerência possa ter um controle sobre o fluxo de dinheiro na instituição.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | *Gerencia, caixa, funcionário responsável por cada setor.* | | | | | | | **Requisitos associados:** | *RFCF010, RFCF009, RFCF001* | | | | | | | | | | | | | | |
| **Identificador:** RFCFUN001- Dar sinal de início/fim de expediente  **Descrição:** O sistema deve receber a identificação do funcionário e hora, e verifica se ele já deu início ao expediente, se sim ele finaliza o expediente do funcionário inserindo esse horário no banco que armazena esses dados, se não ele dá início ao expediente do funcionário armazenando esse horário no banco. O sistema deve verificar a data, se bater com a data do pagamento ele deve fornecer a identificação do funcionário para ser feito o cálculo do salário.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | *Funcionário* | | | | | | | **Requisitos associados:** | RFCFUN002 | | | | | | | | | | | | | | |
| **Identificador:** RFCFUN002- Calcular e armazenar salario.  **Descrição:** O sistema recebe a identificação do funcionário pega sua função, suas horas trabalhadas os descontos tributário e calcula o salário desse mês desse funcionário e armazena no banco de dados dos salários.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | *Funcionário* | | | | | | | **Requisitos associados:** | RFCFUN001 | | | | | | | | | | | | | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Identificador:** RFCFUN003- Cadastrar novo funcionário.  **Descrição:** O sistema recebe os dados do novo funcionário, e os insere no bd que armazena os dados dos funcionários. Se os dados que forem ser inseridos derem conflitos com outro cadastro deverá ser exibido para o usuário uma tela de erro informado o cadastro duplo.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** |  | Essencial | **X** | Importante |  | Desejável | | **Ator(es):** | *Gerência* | | | | | | | **Requisitos associados:** |  | | | | | | |
| **Identificador:** RFCFUN004- Alterar dados do cadastro de um funcionário. **Descrição:** O sistema recebe a identificação do funcionário e o campo que será alterado, e faz alteração no bd. Se o a identificação do funcionário não existir deverá ser exibido uma tela de erro informando cadastro inexistente.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** |  | Essencial |  | Importante | **X** | Desejável | | **Ator(es):** | *Gerência* | | | | | | | **Requisitos associados:** |  | | | | | | |
| **Identificador:** RFCFUN005- Excluir cadastro.  **Descrição:** O sistema recebe a identificação do funcionário que terá o cadastro excluído e o elimina do bd. Se o a identificação do funcionário não existir deverá ser exibido uma tela de erro informando cadastro inexistente.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** |  | Essencial |  | Importante | **X** | Desejável | | **Ator(es):** | *Gerência* | | | | | | | **Requisitos associados:** |  | | | | | | |
| **Identificador:** RFUSR001- Cadastrar usuário para *login* no sistema.  **Descrição:** Recebe os dados requisitados para ser efetuado o cadastro do usuário, e salva, caso haja conflito com usuário já existente deve ser informado a tentativa invalida, ele só pode ser efetuado por usuários de nível de acesso gerencia.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | *Gerência* | | | | | | | **Requisitos associados:** |  | | | | | | |
| **Identificador:** RFUSR002- Alterar cadastro de usuário.  **Descrição:** Recebe os dados requisitados para ser efetuado a alteração do cadastro do usuário, e salva, caso, ele só pode ser efetuado por usuários de nível de acesso gerencia.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | *Gerência e usuário.* | | | | | | | **Requisitos associados:** |  | | | | | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Identificador:** RFUSR003- Remover cadastro de usuário.  **Descrição:** Recebe identificador para ser efetuado a remoção do cadastro do usuário, e confirma-la, ele só pode ser efetuado por usuários de nível de acesso gerencia.       |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | *Gerência e usuário.* | | | | | | | **Requisitos associados:** |  | | | | | | | | | | | | | | |
| **Identificador:** RFUSR004- Validar *login* de usuário.  **Descrição:** Recebe identificador e senha para ser efetuado a validação do usuário, e caso exista e tenha acesso aquela parte do sistema o acesso é confirmado, caso não ele é informado acesso invalido, e na terceira tentativa invalida em 5 min o usuário é bloqueado.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | *Gerência e usuário.* | | | | | | | **Requisitos associados:** |  | | | | | | | | | | | | | | |
|  | | | | | | | | |
| **Responsável:** Rafael Mangolin | | | | | | | | |
| **Identificador:** RFVEN001- Adicionar produto vendido.  **Descrição:** Recebe a identificação do produto a ser vendido, pega as informações do mesmo no estoque de vendas o adiciona na lista de produtos vendidos nessa compra.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | *Caixa* | | | | | | | **Requisitos associados:** |  | | | | | | | | | | | | | | |
| **Identificador:** RFVEN002- Retirar produto da lista de compra.  **Descrição:** Recebe a identificação do produto, e remove o produto do banco de dados. Se o produto não estiver no banco de dados e informado para o caixa que produto não está na lista.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** |  | Essencial |  | Importante | **X** | Desejável | | **Ator(es):** | *Caixa* | | | | | | | **Requisitos associados:** |  | | | | | | | | | | | | | | |
| **Identificador:** RFVEN003- Confirmar venda.  **Descrição:** Recebe a confirmação da venda, envia que a venda foi finalizada para o processo que irá dar baixa no estoque e para o processo que irá salvar a venda realizada. | | | | | | | | |
|  | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Ator(es):** | *Caixa* | | | | | |  |
| **Requisitos associados:** RFVEN004 e RFVEN006 | | | | | | |
| **Identificador:** RFVEN004- Montar relatório de venda.  **Descrição:** Recebe a autorização para montar o relatório, monta o relatório com as informações que estão armazenadas na lista de produtos da venda, faz o cálculo do valor total da venda, uma simples relação dos produtos da venda e data que ela foi realizada. Armazena o relatório da venda no banco de dados reservado para as vendas e envia um sinal de venda finaliza para o processo que vai atualizar o estoque de venda e então envia um mesmo sinal para processo que limpa o banco de dados que armazena a lista de produtos da venda. Se não existir produtos armazenados na lista de produtos vendidos é informado o caixa que a venda não possui produtos.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | *Caixa* | | | | | | | **Requisitos associados:** | RFVEN003 e RFVEN005 | | | | | | | | | | | | | | |
| **Identificador:** RFVEN005- Adequar histórico de produtos.  **Descrição:** Recebe a confirmação de venda finalizada, e exclui todos os produtos que estão no banco de dados da lista da venda.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | *Caixa* | | | | | | | **Requisitos associados:** | RFVEN004 | | | | | | | | | | | | | | |
| **Identificador:** RFVEN006- Verificar quantidade do produto.  **Descrição:** Recebe a confirmação da venda, pega a lista de produtos vendidos no banco que armazena essa lista, vai verificando a quantidade dos produtos comparando a quantidade que tem em estoque com a quantidade mínima que tem que ter no estoque e vai enviando o produto para o processo que vai dar baixa no estoque. Se nessa verificação a quantidade mínima for maior que a quantidade em estoque é enviado o produto que vai ser feito o pedido de reposição para o processo responsável pelo mesmo.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | *Caixa* | | | | | | | **Requisitos associados:** | RFVEN003, RFVEN007 e RFVEN010 | | | | | | | | | | | | | | |
| **Identificador:** RFVEN007 - Dar baixa do produto.  **Descrição:** Recebe a identificação produto e diminui a quantidade dele no estoque. | | | | | | | | |
|  | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável |  |
| **Ator(es):** | *Caixa* | | | | | |
| **Requisitos associados:** | RFVEN006 | | | | | |

|  |
| --- |
|  |
| **Identificador:** RFVEN008- Alterar preço do produto.  **Descrição:** Recebe a identificação do produto que vai ser alterado o preço e o novo preço, faz a alteração de preço. Se o produto não for encontrado deverá ser informada para a gerência que o produto não existe cadastrado no estoque.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** |  | Essencial | **X** | Importante |  | Desejável | | **Ator(es):** | *Gerência* | | | | | | | **Requisitos associados:** |  | | | | | | |
| **Identificador:** RFVEN009- Cadastrar novo produto no estoque de venda.  **Descrição:** Recebe os dados do produto como o nome, marca, tipo de produto (manufaturado ou revenda) e preço de revenda, monta o cadastro e insere no estoque de vendas. Se o produto já existir no estoque é informado para a gerência que o produto já existe no estoque.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator (es):** | *Gerência* | | | | | | | **Requisitos associados:** |  | | | | | | |
| **Identificador:** RFVEN010 - Verificar tipo de produto.  **Descrição:** Recebe os dados do produto em falta e verifica seu tipo. Se ele for manufaturado é enviado os dados do produto para o processo que irá montar o pedido de manufaturado. Se ele for produto de revenda será enviado o produto para o processo que fará o pedido para repor o estoque de vendas.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator (es):** | *Caixa* | | | | | | | **Requisitos associados:** | RFVEN006, RFVEN011 e RFVEN012 | | | | | | |
| **Identificador:** RFVEN011- Fazer pedido do produto manufaturado.  **Descrição:** Recebe os dados do produto manufaturado que está em falta, e é envia um pedido de produção para o produto, contendo o produto e a quantidade que deverá ser produzido para o processo que irá processar o pedido.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | *Caixa* | | | | | | | **Requisitos associados:** | RFVEN010 e RFPRO002 | | | | | | |
| **Identificador:** RFVEN012-Fazer pedido de produto de revenda.  **Descrição:** Recebe os dados do produto de revenda que está em falta, e é enviado um pedido reabastecimento do estoque para o processo que fará a verificação dessa quantidade no estoque. Esse pedido contém o produto e a |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| quantidade que deverá ser transferida do estoque geral para o estoque de vendas.         |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | *Caixa* | | | | | | | **Requisitos associados:** | RFVEN010 e RFCE007 | | | | | | | | | | | | | | |
| **Identificador:** RFVEN013- Aumentar a quantidade do produto no estoque de vendas.  **Descrição:** Recebe a identificação do produto e a quantidade que vai ser transferida do estoque geral para o estoque de venda, é feito o acréscimo do produto no estoque de vendas.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | *Caixa* | | | | | | | **Requisitos associados:** | RFCE007 | | | | | | | | | | | | | | |
| **Identificador:** RFENC001- Alterar status  **Descrição:** O sistema deve receber sinalização de que certa etapa do processo de uma encomenda foi cumprida (pedido, produção ou entrega) e alterar o status da entrega correspondente.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** |  | | | | | | | **Requisitos associados:** | RFENC002, RFENC005. | | | | | | | | | | | | | | |
| **Identificador:** RFENC002- Validar encomenda  **Descrição:** O sistema deve receber sinalização do controle de produção de que a encomenda já está pronta para entrega e sinalizar que a produção foi feita para que o status seja alterado no registro da encomenda correspondente e informar dados da encomenda para que sejam mostrados ao entregador.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** |  | |  | |  | | | **Requisitos associados:** | RFENC001, RFPRO003. | | RFENC003, | | RFENC004, | | | | | | | | | | |
| **Identificador:** RFENC003- Montar entrega  **Descrição:** O sistema deve receber dados da encomenda e do cliente correspondente e mostrá-los ao entregador. | | | | | | | | |
|  | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável |  |
| **Ator(es):** | Entregador | | | | | |
| **Requisitos associados:** | RFENC002 | | | | | |
|  | | | | | | | | |
| **Identificador:** RFENC004- Processar encomenda  **Descrição:** O sistema deve receber dados de encomenda, armazená-los no banco de dados e encaminhar essas informações para o setor de produção.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | Atendente de encomendas | | | | | | | **Requisitos associados:** | RFENC002, RFENC005, RFPRO002. | | | | | | | | | | | | | | |
| **Identificador:** RFENC005- Cadastrar encomenda  **Descrição:** O sistema deve receber informações sobre encomendas, sinalizar que ela está em processo de produção e a inserir no banco de dados.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** |  | | | | | | | **Requisitos associados:** | RFENC004. | | | | | | | | | | | | | | |
| **Identificador:** RFENC006- Cadastrar Cliente  **Descrição:** O sistema deve receber informações de um novo cliente e inserir no banco de dados.       |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | Atendente de encomendas | | | | | | | **Requisitos associados:** |  | | | | | | | | | | | | | | |
| **Identificador:** RFENC007- Alterar Cadastro  **Descrição:** O sistema deve receber informações de campos a serem alterados no cadastro de clientes no banco de dados a realizar estas alterações.     |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Prioridade:** | **X** | Essencial |  | Importante |  | Desejável | | **Ator(es):** | Atendente de encomendas | | | | | | | **Requisitos associados:** |  | | | | | | | | | | | | | | |

## 2.2. Glossário de termos

**Produto** - É o item que é comercializado na instituição, ele pode ser do tipo manufatura, matéria-prima e revenda.

**Manufatura** - É o tipo de produto que e produzido na instituição.

**Matéria** **prima** - É o tipo de produto utilizado na produção de produtos manufaturados, ele não é comercializado na instituição.

**Pedido de compra (Fatura)** - É um pedido que contém as informações do fornecedor e os itens de compra associado a ele.

**Item de compra** - É formado por um produto, a quantidade que deve ser comprada dele e seu valor de compra, não existe se não estiver associado com um Pedido de compra.

**Ordem de Produção** - É um conjunto de itens de produção gerados por uma encomenda, possui também prazo de entrega.

**Item de produção** - É formado por um produto manufaturado e a quantidade que ele deve ser produzido, não existe se não estiver associado com uma Ordem de produção.

**Encomenda** - É um pedido de produtos manufaturados que é associado a um cliente, que deve ser produzido e entregue na data e endereço por ele solicitada. **Cliente** – É a pessoa ou instituição que fará uso do serviço de encomendas.

**Item de venda** - É o formado por um produto manufaturado ou de revenda, a quantidade que será vendida e o seu preço de venda, não existe se não estiver associado com uma Venda.

**Venda** – É um conjunto de itens de venda que são acumulados durante uma venda, possui também o valor total da venda e o horário e data da qual foi feita.

**Caixa** – É o acumulo de vendas que aconteceu enquanto o caixa estava operante, tem associado a ele o identificador do operador de caixa, o período que ficou operante e o valor arrecadado.

**E/S Funcionário** – É o que controla a frequência do funcionário, conhecido popularmente como “Bater cartão”.

**Funcionário** – É a pessoa que realiza as atividades necessárias para a manutenção da instituição, podendo ser atendente de caixa ou gerente, seu registro possui os dados pessoais do funcionário e seu cargo na instituição.

**Pagamento de Funcionário** – É a relação mensal feita pela quantidade de horas trabalhadas e pelo cargo exercido pelo funcionário, é a remuneração monetária dado em troca do serviço prestado.

**Usuário** – É formado por *login*, senha e nível de autorização, é por meio dele que poderá ser feito o acesso ao sistema e suas funcionalidades.

**Evento financeiro**- É todo tipo de transação que tenha valor monetário atrelado a ela, onde poderá, com isso, gerar um relatório.

## 2.3. Estudo de viabilidade

### 2.3.1. Contribuições para a empresa

Com a implantação do sistema, o estabelecimento padaria *Gegê* terá como benefício um melhor gerenciamento do estoque de seus produtos proporcionando assim um melhor atendimento aos seus clientes. Um outro benefício está relacionado a visão gerencial do negócio. O sistema fornece uma gama de relatórios que possibilita ao administrador do negócio, ter uma visão mais ampla da empresa e assim tomar melhores decisões.

Com também um mais fácil controle dos funcionários sobra mais tempo para realização de tarefas administrativas e planejamentos de crescimento da empresa.

### 2.3.2. Custo X Benefício

O custo para o desenvolvimento deste sistema será cobrado pelas quantidades de horas estimada para o desenvolvimento do projeto. Existe também o custo para fornecer treinamento aos funcionários, que já está incluso no custo do projeto, que irão utilizar o sistema, que além monetário também de tempo na linha de produção da empresa, que vai ser parcialmente interrompida devido ao treinamento.

Com uma melhor organização do setor de produção e do setor de encomenda, a empresa pode controlar melhor a produção de suas encomendas e suas entregas, aumentando assim o número de encomendas simultâneas, gerando mais receita.

Um outro benefício será a fácil comunicação entre os setores, facilitando os processos que são realizados em conjuntos ou possuem dependências de processos que são realizados por outros setores, pois com o sistema será possível o acompanhamento tornando mais fácil a identificação de onde ocorre o entrave no processo.

### 2.3.3. Viabilidade técnica

O projeto apresentado é viável tecnicamente pois será utilizado o navegador do computador pessoal de cada funcionário, não necessitando assim de uma máquina que possui um custo alto, e com as máquinas atuais, não são necessárias atualizações do hardware. Nestas máquinas, será necessário instalar os navegadores mais atuais, pois os que estão presente não oferecem total suporte as tecnologias que serão utilizadas. Recomenda-se utilizar os navegadores Google Chrome ou Mozilla Firefox mais atuais, pois eles suportam todas as tecnologias atuais e não tem custo para o projeto.

Inicialmente foi cogitado a aquisição de uma máquina para manter ligada e que iria conter o servidor, pois o sistema utilizará uma API (*Interface de Programação de Aplicativos***)** WEB para fazer a comunicação entre o cliente/servidor, porém ao fazer o estudo de caso, foi percebido que utilizar um servidor na nuvem seria mais vantajoso, visto que o tempo *offline* destes servidores é baixo, além de não ter que arcar com os custos de adquirir uma máquina, um *no-break* (utilizado para manter o computador ligado mesmo com queda de energia, para prevenir perda de dados).

Os frameworks utilizados no projeto diminuirão o tempo destinado para a programação, pois abstraem as linguagens de programação utilizadas, tanto no cliente, quanto no servidor, requerendo assim que o programador não tenha um conhecimento tão profundo da linguagem, e mesmo assim consiga produzir bem.

O projeto será construído utilizando a tecnologia Cliente/Servidor, e para isto, a tecnologia *Rails* é uma das melhores opções, pois permite uma prototipação rápida de como será construído e como é uma plataforma que está em ascensão, a comunidade é muito forte, e vários dos empecilhos que podem ser encontrados durante o desenvolvimento, são rapidamente resolvidos com pesquisas simples.

### 2.3.4. Viabilidade legal

O sistema não entra em conflito com nenhuma lei, decreto ou norma legal, e os requisitos legais para emissão de documentos e armazenamento de informações pessoais de clientes (físicos e jurídicos) e funcionários será seguido como é informado por lei.

A equipe enfatiza o comprometimento legal no desenvolvimento do sistema, não permitindo alterações popularmente conhecidas como “caixa-dois”, controle alterado de estoque e de entrada de vendas.

### 2.3.5. Alternativas: (avaliação de alternativas)

Uma alternativa de projeto poderia ser a funcionalidade do cliente se cadastra e fazer ele mesmo suas encomendas, como um *e-commerce*, assim a visibilidade da empresa, sua abrangência de mercado e suas vendas aumentam. Essas melhorias geram um aumento de custo do desenvolvimento do projeto e da infraestrutura para implementa-la, tendo de dar suporte para a conexão *web*.

As alternativas de projeto poderiam ser mudança na linguagem usada para Java, alterando também os frameworks de suporte onde seriam usados para o *hibernate* para comunicação com o banco, e o Spring para suporte de criação de *api-rest.* Essa mudanças implicariam no aumento de tempo para o desenvolvimento do projeto, mas deixa o projeto mais escalável e manutenível.

## 2.4. Avaliação de risco

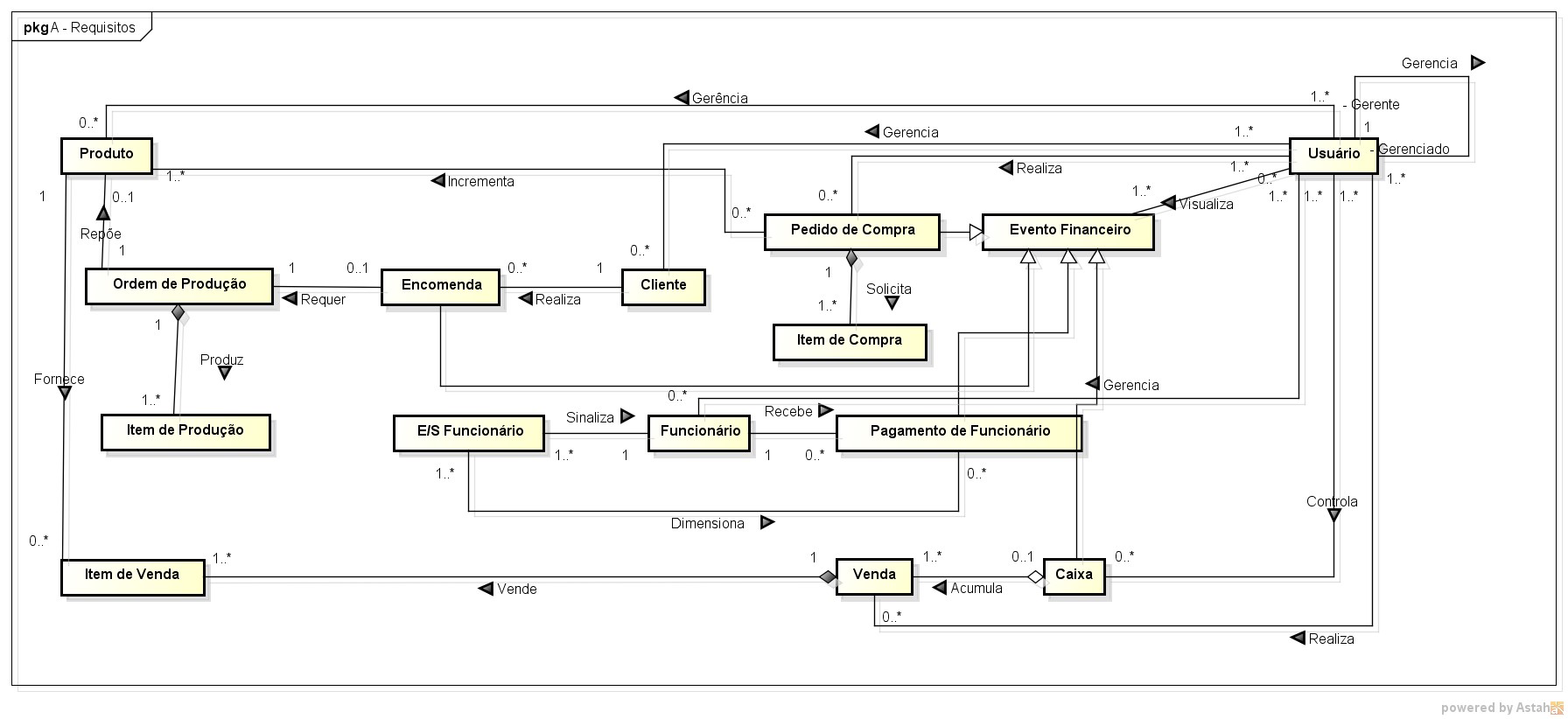
### 2.4.1. Riscos

|  |
| --- |
| **Risco:** Incerteza do cliente em questão aos requisitos  **Descrição:** O cliente pode achar durante o desenvolvimento do projeto que alguns requisitos devem ser adicionados no sistema, que não estavam previstos pelo documento de requisitos que foi acordado pelas partes. **Ocorrência:** Não muito provável de ocorrer depois do processo de análise do sistema  **Grau de impacto:** Dependendo do requisito pedido pode ser grande |
| **Risco:** Perda de pessoas da equipe  **Descrição:** A equipe pode ser desfalcada durante o desenvolvimento do projeto. Neste projeto em especifico as saídas podem ser ocasionadas por desistência da matéria ou do curso, ou casos de saúde que impossibilitem o integrante de realizar suas tarefas  **Ocorrência:** Dado o histórico do curso esse risco é provável, dado que em outras matérias que servem de requisitos para essa, esse já aconteceu  **Grau de Impacto:** Pode impactar severamente no desenvolvimento do projeto, tomando por base que o trabalho do integrante que saiu teria que ser repartido entre os outros integrantes do grupo |
| **Risco:** Documentação de requisitos mal especificada  **Descrição:** A documentação dos requisitos pode estar especificação muito fraca, dado a relevância do requisito, o deixando ambíguo, podendo assim levar a uma errada implementação do que seria necessário  **Ocorrência:** Pode ocorrer dado alta quantidade de requisitos, mas com poucas chances dado as várias revisões feitas sobre o documento  **Grau de Impacto:** Pode impactar no desenvolvimento do projeto, tomando por base que haverá um retrabalho sobre aquele requisito |
| **Risco:** Descontinuidade no suporte a componentes utilizado no sistema **Descrição:** Como o sistema está usando ferramentas e utilitários que são grátistem a possibilidade das empresas que mantém esses projetos, pararem de oferecer suporte a eles  **Ocorrência:** Possível, mas com pouca chance de acontecer, sendo a maioria legados e altamente utilizados no mercado  **Grau de Impacto:** Caso ocorra pode vir a interromper totalmente o desenvolvimento do projeto para a adaptação de outras opções. |
| **Risco:** Infraestrutura do Cliente pequena  **Descrição:** A infraestrutura do cliente para receber o projeto não equivale as funcionalidades que ele requisitou para o projeto, forçando a equipe a ‘escovar bits’ para otimizar ao máximo a performance  **Ocorrência:** Possível, mas com baixíssima chance de acontecer, ao baixo custo da infraestrutura hoje em dia  **Grau de Impacto:** Pode retardar o desenvolvimento do projeto dado o foco do desenvolvimento estar voltado para o alto rendimento e sendo uma aplicação de média a pequena |

### 2.4.2. Processo no risco “incerteza do cliente em questão aos requisitos”

|  |
| --- |
| **Identificação:** Incerteza do cliente em questão aos requisitos |
| **Avaliação:** Esse risco é avaliado como impeditivo, pois dependendo da funcionalidade pedida pelo cliente o projeto tem que ser parado para ser feito uma reformulação, podendo gerar retrabalho em partes já prontas.  E a ocorrência dele caso a captura de requisitos não seja completa, pode ser frequente. |
| **Priorização:** 1 |
| **Resolução:** Caso ocorra, deve ser conversado com o cliente os impactos que as mudanças solicitadas irão gerar, e que o preço e o tempo de entrega irão aumentar, se mesmo após informado o cliente decidir prosseguir com as mudanças, deve se verificar quais componentes do sistema não serão afetados, não interrompendo assim o desenvolvimento deles, e deve ser modificado o documento de requisitos, e os diagramas que o projeto possui. Depois das mudanças na documentação e de definir a nova organização do projeto deve dar continuidade no desenvolvimento, adaptando as partes já prontas para o novo modelo |
| **Monitoração:** Para monitora a ocorrência desse risco e minimizar seu impacto, deve ser feito uma captura de requisitos de qualidade, e o cliente deve receber feedbacks do projeto. |

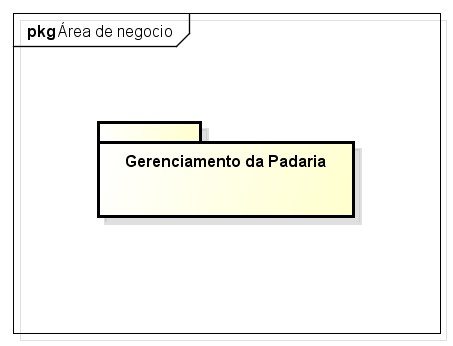
## 2.5. Modelo de Domínio ou Diagrama de objetos de negócios



*Figura 1.Modelo de Domínio ou Diagrama de objetos de negocio*

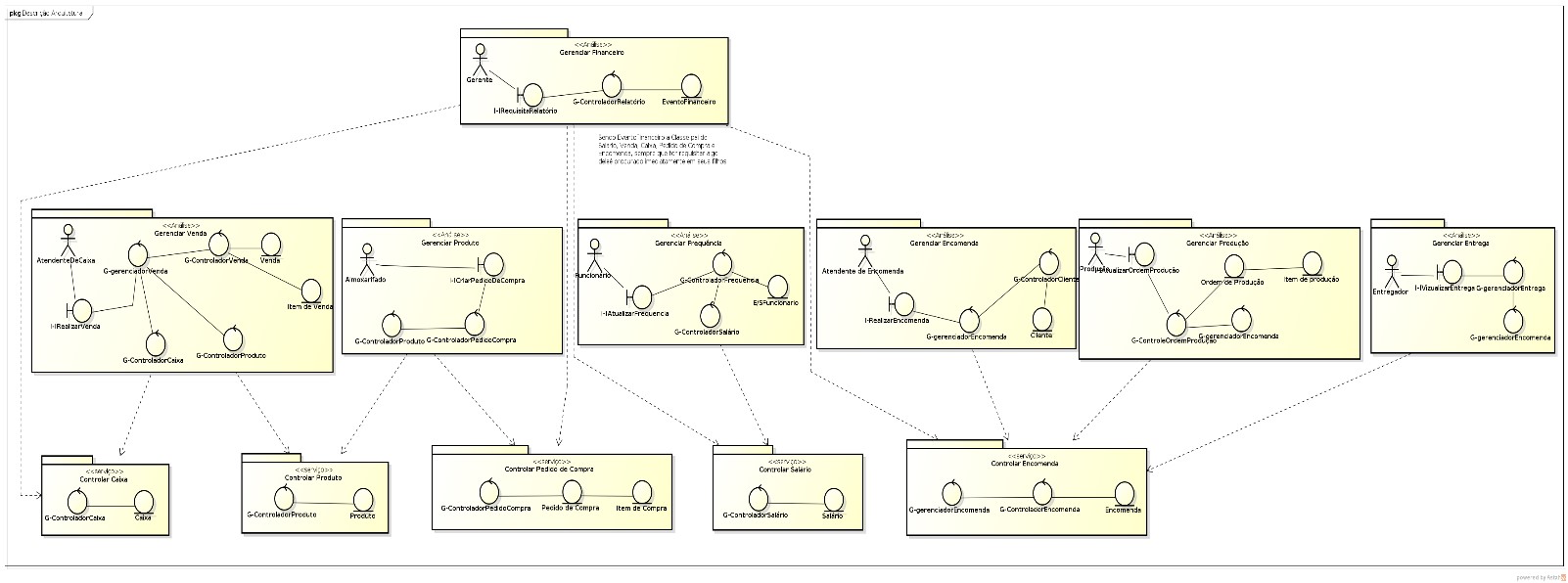
## 2.6. Pacotes

### 2.6.1. Diagrama de pacotes da Área de negócio



*Figura 2. Diagrama de pacotes da Área de negócio, informando que o sistema produzido trabalha sem comunicação externa*

### 2.6.2. Diagrama de pacotes de análise e serviço



## 2.7. Arquitetura de Software

Após avaliar o projeto a ser desenvolvido foram escolhidos os estilos de arquitetura:

* Estilo de arquitetura baseado em objetos: foi escolhido devido a sua estrutura que mantém dados e procedimentos em uma só entidade, facilitando assim o desenvolvimento e a manutenção e dando suporte para o desenvolvimento de sistemas orientados a objetos. Com esse estilo facilita aplicar as características de modelos orientados a objetos (encapsulamento, polimorfismo e herança), dando a possibilidade do crescimento do sistema acrescentando as novas partes e fazendo pequenas mudanças no código.
* Estilo de arquitetura baseado em eventos: foi escolhido para dar suporte na interação do usuário com o sistema (*front)*, baseando os componentes de interação (*inputs,* *buttons, links, etc*) em “disparadores e receptores” de eventos quando utilizados pelo usuário, e os tornando independentes e reutilizáveis em diferentes partes do sistema. Também fazendo a manutenção no *front* fácil, sendo que quando deseja alterar o comportamento de um componente, ele é alterado para todo sistema.
* Estilo de arquitetura em camadas (MVC): foi escolhido por facilitar alterações no código, onde quando alterado o código em uma camada ele irá gerar efeito em de reorganização no projeto. A divisão das camadas será baseada no modelo MVC (*model – view – control*), facilitando assim a portabilidade do projeto, onde se ele sofrer migração de web para desktop, só deverá ser alterado a parte da *view*, o mesmo acontece caso tenha mudança na parte de persistência (*model*).
* Estilo de arquitetura cliente/servidor: Tendo em vista que o sistema cliente servidor diminui o custo de infraestrutura, onde a processamento dos dados ocorre em um computador (o servidor) que precisa ter um poder maior de processamento, e os que farão acesso a ele deve apenas suportar uma conexão com o computador principal por meio da intranet ou internet, para poder emitir suas requisições. Esse estilo também facilita a escalabilidade do projeto caso ele possa crescer para outras alocações.

**Estilos que foram descartados:**

-Estilo de arquitetura de *pipes* e filtros: Tomando por base a estrutura do estilo (*bash*), que para a complexidade de interação que esse projeto necessita, o retrabalho na reorganização dos filtros seria grande, tornando também inviável devido a manutenção que ele iria gerar.

-Estilo de arquitetura de quadro negro: Dado a amplitude de processos e a execuções desnecessárias em alguns processos devido à procura da solução o quadro, esse estilo foi descartado.

3. ELABORAÇÃO

## 3.1. Modelo de Casos de Uso

### 3.1.1. Identificação dos Casos de Uso

**Responsável: Igor Santana**

|  |
| --- |
| **Identificador:** CSU10 – Atualizar frequência  **Ator(es):** Funcionário  **Descrição:** Realiza a funcionalidade conhecida como “bater ponto”, que armazena a identificação do funcionário e o seu horário de entrada e saída do expediente. |
| **Identificador:** CSU11 – Gerar relatório de fluxo  **Ator(es):** Gerente  **Descrição:** Exibe o relatório de fluxo de despesas e receitas organizados por data |
| **Identificador:** CSU12 – Gerar relatório de folha de pagamento  **Ator(es):** Gerente  **Descrição:** Exibe o relatório da folha de pagamento da empresa, pode se por funcionário individual, e por período |

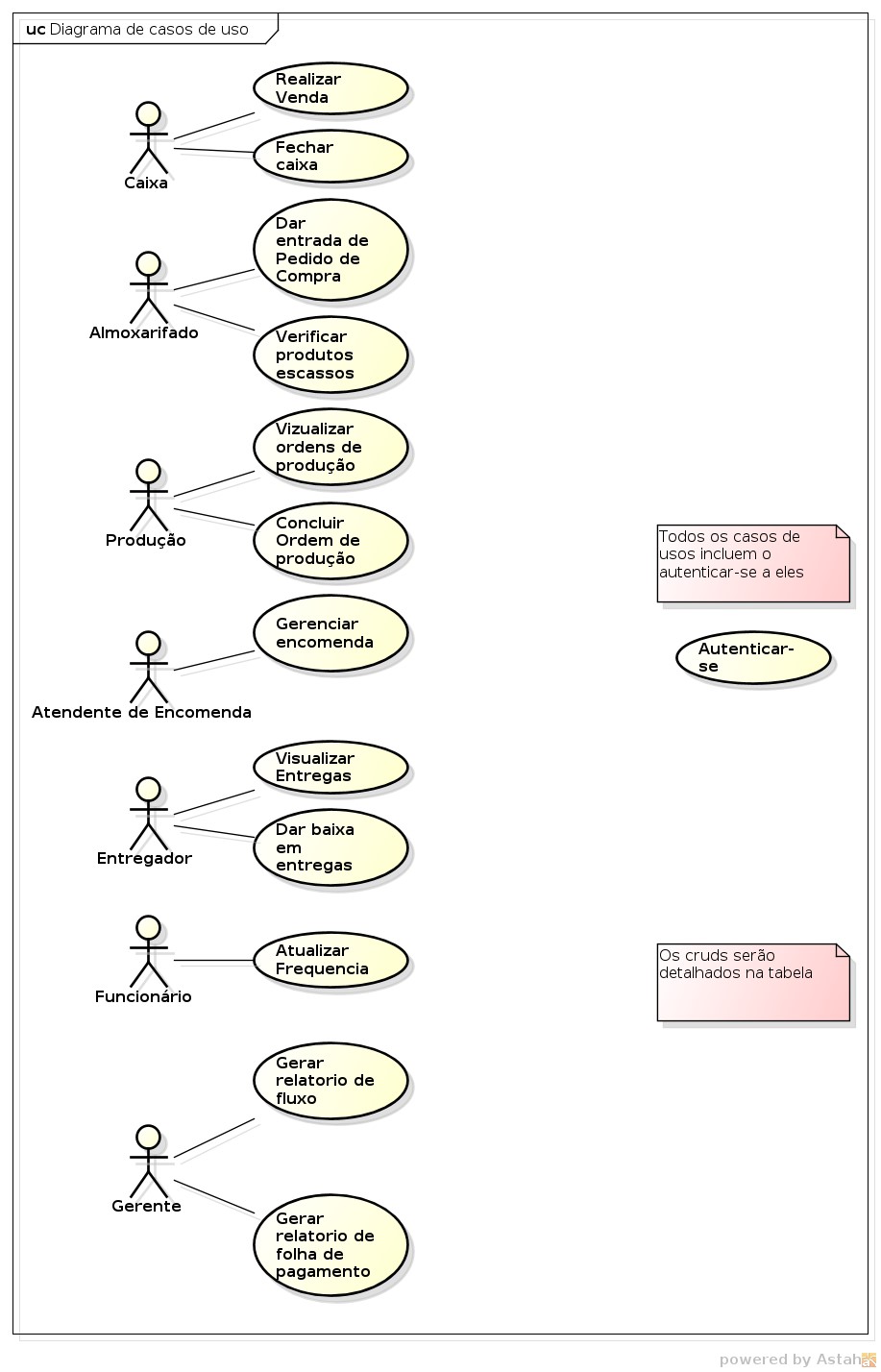
|  |
| --- |
| **Responsável:** Leonardo Catharin |
| **Identificador:** CSU03 – Dar entrada de pedido de compra  **Ator (es):** Almoxarifado  **Descrição:** Faz entrada de pedidos de compra para cadastrar o pedido de compra e modificar as quantidades dos produtos. |
| **Identificador:** CSU04 – Verificar produtos escassos  **Ator(es):** Almoxarifado  **Descrição:** Exibe os produtos que estão acabando ou estão em falta |
| **Identificador:** CSU05 – Visualizar ordens de produção  **Ator(es):** Produção  **Descrição:** Exibe as ordens de produção e suas datas de produção |
| **Identificador:** CSU06 – Concluir ordem de produção  **Ator(es):** Produção  **Descrição:** Modifica o *status* da ordem de produção para pronto |

|  |
| --- |
| **Responsável:** Rafael Mangolin |
| **Identificador:** CSU01 – Realizar venda  **Ator(es):** Caixa  **Descrição:** Contém as operações necessárias para se efetuar uma venda no caixa. |
| **Identificador:** CSU02 – Fechar caixa  **Ator(es):** Caixa  **Descrição:** Executa o fechamento do caixa armazenando o operador de caixa e a hora do fechamento |
| **Identificador:** CSU07 – Gerenciar encomenda  **Ator(es):** Atendente de Encomendas  **Descrição:** Contém operações para a manutenção das encomendas |
| **Identificador:** CSU08 – Visualizar Entregas  **Ator(es):** Entregador  **Descrição:** Exibe as entregas pendentes |
| **Identificador:** CSU09 – Dar baixa em entregas  **Ator(es):** Entregador  **Descrição:** Altera o *status* da encomenda para entregue |

### 3.1.2. Priorização dos Casos de Uso

|  |
| --- |
| 1 - CSU01 – Realizar venda |
| 2-CSU04 – Verificar produtos escassos |
| 3 - CSU07 – Gerenciar encomenda |
| 4 - CSU02 – Fechar caixa |
| 5 – CSU03 - Dar entrada de pedido de compra |
| 6 – CSU12 – Gerar relatório de folha de pagamento |
| 7 – CSU11 – Gerar relatório de fluxo |
| 8 - CSU08 – Visualizar Entregas |
| 9 - CSU09 – Dar baixa em entregas |
| 10 -CSU05 – Visualizar ordens de produção |
| 11 - CSU06 – Concluir ordem de produção |
| 12 – CSU10 – Atualiza frequência |

### 3.1.3. Diagrama de Casos de Uso



### 3.1.4. Detalhamento dos Casos de Uso

|  |
| --- |
| **Responsável**: Rafael Mangolin |
| **Caso de Uso**: Realizar Venda |
| **Identificador**: CSU01 |
| **Descrição**: Inicia uma nova venda, podendo incluir e remover produtos dela, informando a quantidade de produto que ela contém e a soma de seus valores, fazendo no final o armazenamento da venda que contém o tipo de pagamento selecionado |
| **Ator (es)**: Caixa |
| **Pré-Condições**: O usuário estar logado no sistema e ser um usuário autorizado. |
| **Pós-Condições**: Armazenar a venda e realizar o decréscimo na quantidade dos produtos que estavam na lista |
| **Fluxo Principal**:   1. - O usuário seleciona a opção de início de uma nova venda 2. - O sistema abre a tela de venda 3. - O usuário adiciona produtos na venda informando o produto e a quantidade vendida dele 4. - O usuário seleciona finalizar venda 5. - O sistema requisita como será realizado o pagamento (cartão, dinheiro, etc) 6. - O usuário seleciona a forma de pagamento 7. - O sistema requisita a quantia fornecida para o pagamento   O usuário fornece a quantia  O sistema informa o troco e finaliza a venda |
| **Fluxo Alternativo:**   * Se entre os passos 3 e 4 o usuário decidir remover um item ou alterar a quantidade de um produto, o sistema irá requisitar o código de acesso para validar a operação. * Se no meio do processo o usuário quiser cancelar a venda ou fechar a janela em que o processo estiver aberto, o Sistema deverá gerar uma mensagem perguntando se ele quer mesmo sair do processo, se sim fechar a janela ou finalizar a venda não armazenando os dados já digitados, se não continuar com o processo normalmente. |
| **Fluxo de Exceção:**   * Caso o usuário tente finalizar a compra sem nenhum produto adicionado informar o que não há nenhum produto na lista * Caso o usuário informar um valor de pagamento que seja menor que que o valor da compra, informar valor invalido. |

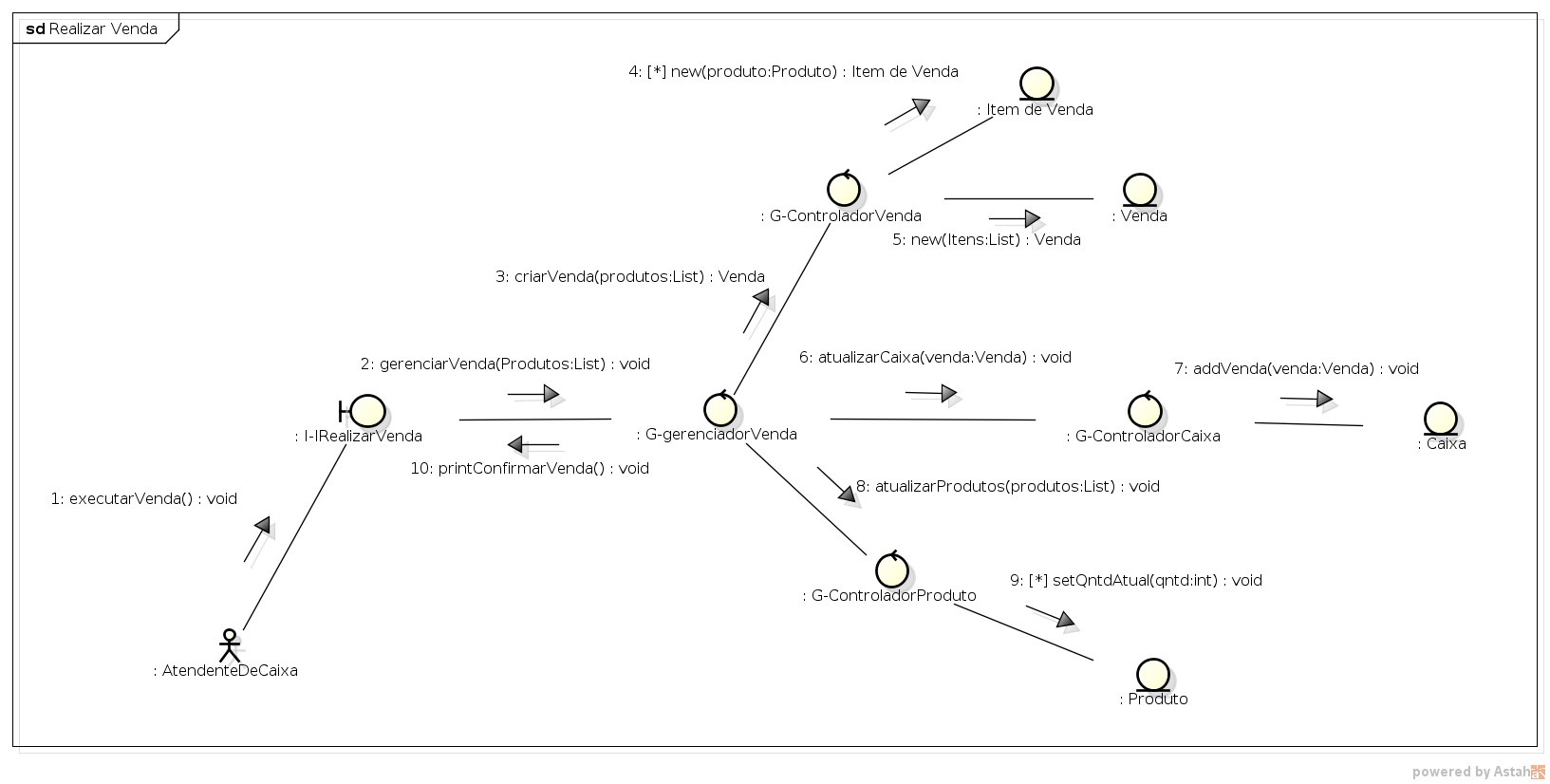
|  |
| --- |
| **Responsável:** Leonardo Catharin |
| **Caso de Uso:** Dar entrada no pedido de compra |
| **Identificador:** CSU03 |
| **Descrição:** Inclui, exclui ou modifica um tipo de produto presente no estoque, além de verificar a quantidade de produto em estoque. Alimentação por intermédio do controlador de pedido de compras. |
| **Ator (es):** Almoxarifado, Gerente |
| **Pré-Condições:** O usuário estar logado no sistema e ser um usuário autorizado. |
| **Pós-Condições:** Inclusão, exclusão ou modificação de um determinado produto. |
| **Fluxo principal:**   1. - O Usuário faz login no sistema; 2. - O Sistema exibe na tela a lista de produtos do estoque; 3. - O Sistema mostra opções de incluir, excluir ou atualizar os registros; 4. - O usuário escolhe a opção desejada; 5. – O Sistema aplica as operações solicitadas e retorna ao item 3. |
| **Tratamento de Exceção:**   1. - Usuário entra com login ou senha errado   1.1-O sistema emite mensagem de erro, retorna e solicita novamente *login* e *senha;*  1.2-O sistema emite mensagem de login impossibilitado, e bloqueia a tentativa de login  por um período;     1. – Algum dado informado não atende as restrições requeridas   2.1-O sistema emite mensagem de erro e solicita o dado novamente;     1. – Há erro de conexão com a base de dados   3.1-O sistema emite mensagem de erro e apresenta as alternativas “cancelar” e “tentar  novamente”       1. – Na consulta ao banco de dados, não existe o produto solicitado   4.1. Retorna uma mensagem avisando que não existe aquele produto |

|  |
| --- |
| **Responsável**: Igor Santana |
| **Caso de Uso**: Gerar relatório de folha de pagamento |
| **Identificador**: CSU12 |
| **Descrição**: Gera um relatório documentando todos os gastos com pagamentos dos funcionários, com o usuário podendo escolher os períodos (Mensal, Trimestral, Semestral, Anual), as categorias em que os usuários estão divididos (Produção, Venda, todas, entre outras), contendo todas as informações necessárias dos usuários. |
| **Ator (es)**: Gerente |
| **Pré-Condições**: O usuário deve estar logado no sistema como Gerente e ter funcionários cadastrados. |
| **Pós-Condições**: Adicionar o relatório na lista dos relatórios realizados. |
| **Fluxo Principal**:   1. - O usuário seleciona a opção de gerar relatório 2. - O sistema abre a tela de gerar relatório 3. - O usuário seleciona como será o período utilizado no relatório 4. - O sistema armazena qual período foi escolhido 5. - O usuário selecionará qual categoria será utilizada no relatório 6. - O sistema faz a busca no banco de dados e retorna os dados relativos a categoria e o período 7. - O usuário escolhe entre adicionar o relatório aos favoritos ou imprimir. |
| **Fluxo Alternativo:**  Se entre os passos 6 e 7 o usuário trocar o período, os dados serão *resetados*. Se entre os passos 6 e 7 o usuário trocar a categoria, os dados serão *resetados*.  Se entre algum dos passos o usuário desejar fechar a tela ou cancelar o relatório, o sistema deverá alertar o usuário de que os dados serão perdidos |
| **Fluxo de Exceção:**  Caso o usuário tente gerar o relatório sem escolher uma categoria ou período. |

## 3.2. Modelo de Análise

Responsável Rafael Mangolin

Diagrama referente ao caso de uso CSU01 - Realizar venda



Responsável Leonardo Catharin

Diagrama referente ao caso de uso CSU03 – Dar entrada no pedido de compra

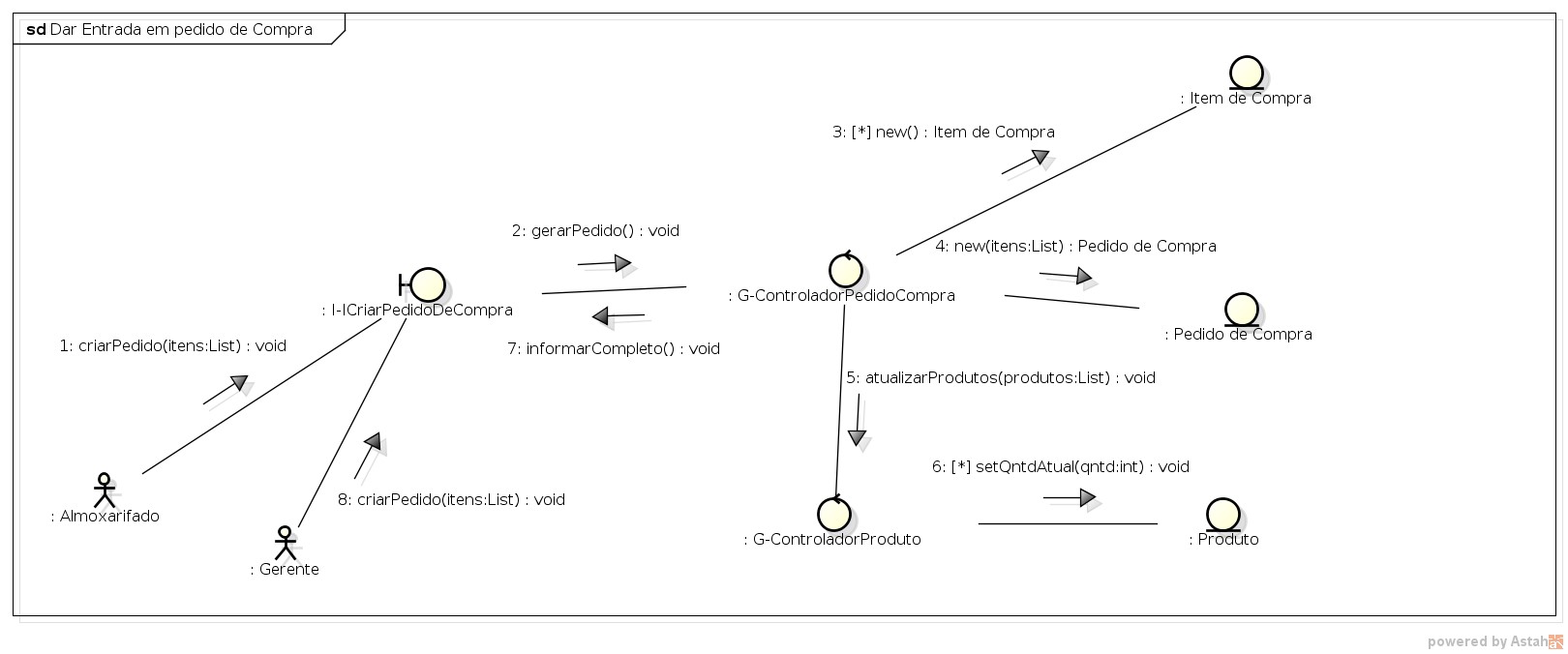
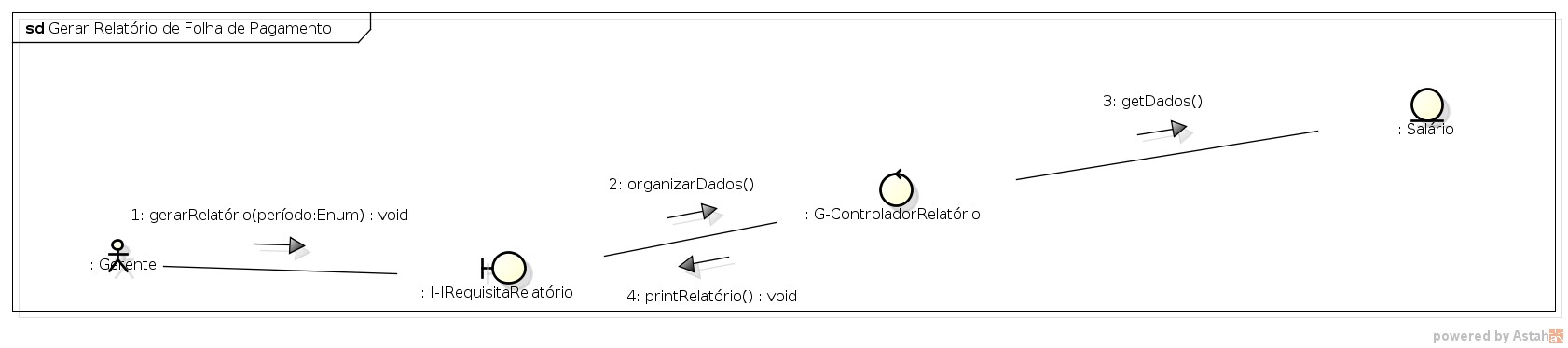


Diagrama referente ao caso de uso CSU12 – Gerar relatório de folha de pagamento



## 3.3. Planejamento das iterações

### 3.3.1. Iteração Nº1 – CRUD’S

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Conceito** | **C** | **R** | **U** | **D** | **Observações** |
| **Produto** | X | X | X | X | -Não é possível excluir um produto que ainda tenha alguma quantidade no estoque; |
| **Pedido de compra**  **(Fatura)** | X | X |  | X | -Não é possível excluir uma fatura, apenas inserir e visualizá-la; |
| **Item de compra** | X |  |  | X | -Não é possível criar um item de compra sem ele estar associado a uma Pedido de Compra; -Um item só pode ser excluído durante a criação de um Pedido de Compra, após isso ele não pode ser excluído; |
| **Ordem de Produção** | X | X | X | X | -Não é possível excluir uma ordem de produção que esteja vinculada com uma encomenda ou que já tenha o estado de “*Em produção*”;  -Uma Ordem de Produção só pode sofrer  alteração na sua data de entrega; |
| **Item de Produção** | X | X |  | X | -Não é possível criar um item de produção sem ele estar associado a uma Ordem de Produção; -Um item só pode ser excluído durante a criação da Ordem de Produção, após isso ele não pode ser excluído; |
| **Encomenda** | X | X | X | X | -Uma encomenda não pode ser criada sem estar vinculada com um cliente cadastrado;  -Uma encomenda só pode ser alterada a data e/ou o endereço da entrega, após a sua inserção; -Uma encomenda só pode ser excluída se o *status* da Ordem de Produção não estiver em *“em produção”* ou *“Pronto”;* |
| **Cliente** | X | X | X | X | -Não pode ter clientes cadastrados com o mesmo identificador (CPF para pessoa física, CNPJ para pessoa jurídica);  -Não pode excluir um cliente que possui uma encomenda em pendente; |
| **Item de venda** | X | X |  | X | -Não é possível criar e/ou excluir um item de  Venda sem ele estar associado a uma Venda; |
| **Venda** | X | X |  | X | -Uma Venda não pode ser inserida sem nenhum  Item de Venda associada a ela; |
| **Caixa** | X | X |  |  | -Um Caixa não pode ser inserido se não estiver associado a ele ao menos uma Venda e o funcionário responsável pelas vendas; |
| **E/S**  **Funcionári o** | X |  | X | X | -Para inserir E/S o funcionário deve estar cadastrado;  -Para alterar ou excluir E/S Funcionário deve possuir nível de acesso *“Gerência”*; |
| **Funcionári o** | X | X | X | X | -Não pode ter funcionários cadastrados com o mesmo identificador (CPF); |
| **Pagamento de Funcionári o** |  | X |  |  | -Pagamento de Funcionário é gerado  automaticamente, e só tem a opção de consulta; |
| **Usuário** | X |  | X | X | -Para cadastrar ou excluir um usuário é preciso ter nível de acesso “*Gerência*”; |

### 3.3.2. Iteração Nº2

CSU01 - Realizar venda

CSU07 – Gerenciar encomenda

CSU02 – Fechar caixa

CSU08 – Visualizar Entregas

CSU09 – Dar baixa em entregas

### 3.3.3. Iteração Nº3

CSU10 – Atualizar frequência

CSU11 – Gerar relatório de fluxo

CSU12 – Gerar relatório de folha de pagamento

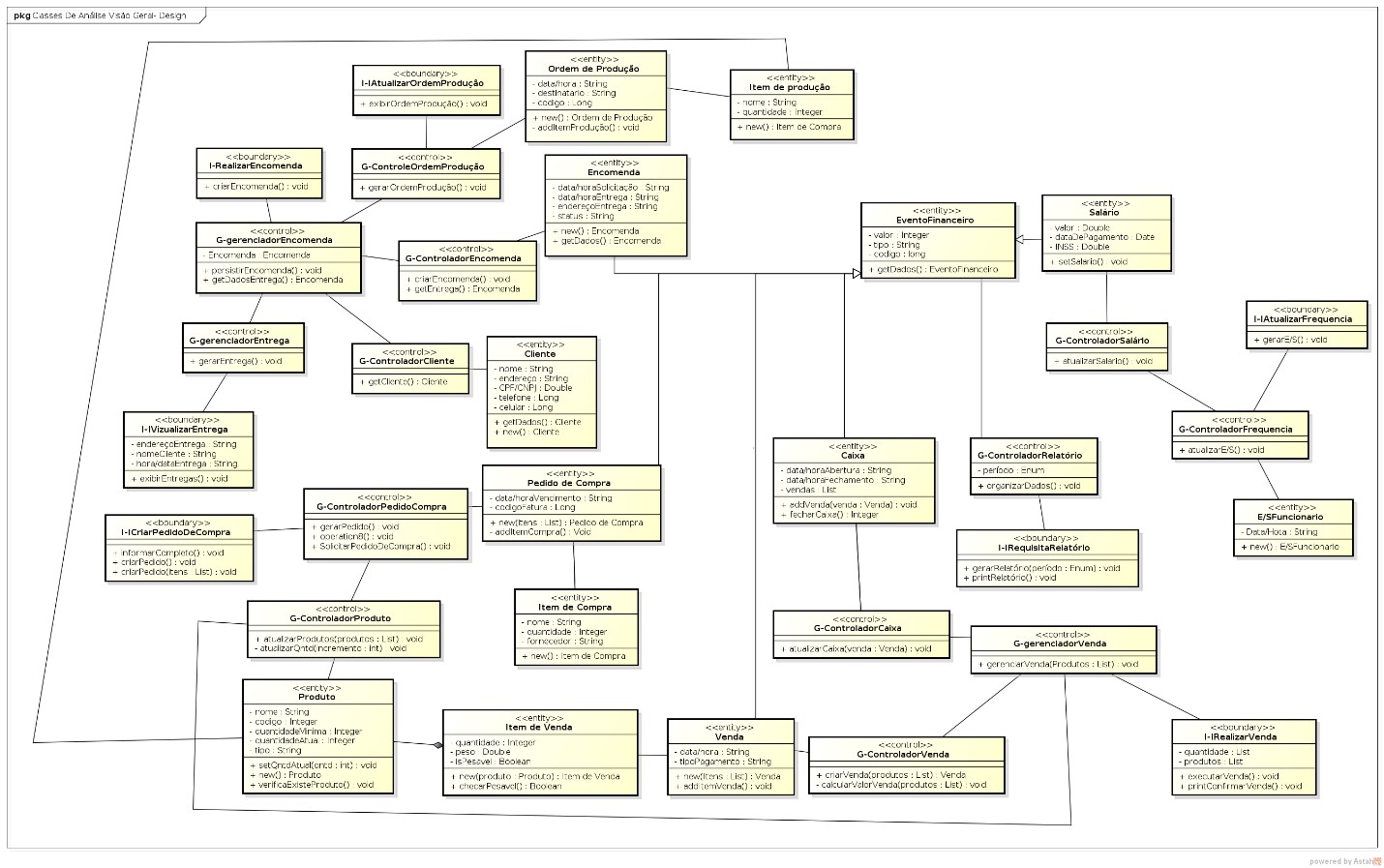
### 3.3.4. Iteração Nº4

CSU04 – Verificar produtos escassos

CSU05 – Visualizar ordens de produção

CSU06 – Concluir ordem de produção

## 3.4. Diagrama de Classes do projeto



# FRAMEWORKS

*Framework* é um conjunto de classes abstratas e genéricas que são reutilizáveis entre vários projetos, seja em uma empresa ou com o código aberto. A grande característica de um *framework* é que ao invés de bibliotecas e *API’s* (Interface de Programação de Aplicativos), ele é quem dita o modo com o qual o programador irá trabalhar em sua aplicação.

Um *framework*, quando bem utilizado pelo programador, torna a programação de uma aplicação muito mais rápida e dinâmica, pois a abstração que ele provém faz com que o programador não tenha que se preocupar com problemas de mais baixo nível, como por exemplo, conexões com banco de dados, transferência de dados cliente/servidor, entre outros. O programador assim fica mais livre para lidar regras de negócio específicas da aplicação.

No sistema desenvolvido neste trabalho, três *frameworks* serão utilizados: *Rails*, Angular.js e *Twitter Bootstrap*. Serão utilizados frameworks de códigolivre, pois assim o custo é reduzido e a manutenção é menos custosa, visto que a comunidade que utiliza estes *três* frameworks é grande.

Além destes três, existem outras opções no mercado para o desenvolvimento tanto de aplicações comerciais, quanto de sites, sistemas, etc. Dois bastante utilizados são o Spring e o *Hibernate*, ambos escritos em Java. O Spring ajuda na criação de aplicações MVC (assim como o *Rails*), e o *Hibernate* simplifica a conexão de aplicações Java com diversos Banco de Dados.

*Rails* é um framework web escrito em *Ruby*, e foi desenvolvido por David *Heinemeier Hansson* em 2003, e seu código é aberto desde então. Seu código já conta com mais de 2700 contribuidores. Ele inclui tudo que é necessário para o desenvolvimento de aplicações de padrão MVC (*Model-ViewController*), com suporte a conexão com banco de dados e solicitações HTTP.

Angular.js é um framework mantido pela Google Inc., e foi escrito em

Javascript. Ele é utilizado para construir aplicações web no formato *Single Page Application* (Aplicação de uma página), e simula o padrão MVC. Ele possui atualmente mais de 1200 contribuidores, e é um dos frameworks *Javascript* mais utilizados no mundo.

*Twitter Bootstrap* é um framework criado dentro do *Twitter* por Mark Otto em 2010, e é pensado para que o desenvolvimento de aplicações e *websites* responsivos se tornem mais simples. Atualmente tem mais de 650 contribuidores e está na terceira versão.