**Lista de Figuras**

[Figura 1- Diagrama de Fluxo de Dados (exemplo) 7](#_Toc523483735)

[Figura 2 - Diagrama de Contexto (Exemplo) 7](#_Toc523483736)

[Figura 3 - Diagrama/Tabela de Decisão (Exemplo) 8](#_Toc523483737)

[Figura 4 - Digrama de Caso de Uso (Exemplo) 9](#_Toc523483738)

[Figura 5 - Diagrama de Sequencia (Exemplo) 10](#_Toc523483739)

[Figura 6 - Diagrama de estados (Exemplo) 10](#_Toc523483740)

[Figura 7 - Diagrama de Classes (Exemplo) 11](#_Toc523483741)

[Figura 8 - Estrutura Orgânica (Portos e Caminhos de Ferros de Moçambique) 14](#_Toc523483742)

[Figura 9- Diagrama Entidade Relacionamento 17](#_Toc523483743)

[Figura 10 - Diagrama de caso de uso funcionário 48](#_Toc523483744)

[Figura 11 - Diagrama de caso de uso camião 49](#_Toc523483745)

[Figura 12 - Diagrama de caso de uso comboio 49](#_Toc523483746)

[Figura 13 - Diagrama de caso de uso navio 50](#_Toc523483747)

[Figura 14 - Diagrama de caso de uso mercadoria 50](#_Toc523483748)

[Figura 15 - Diagrama de caso de uso noticia 51](#_Toc523483749)

[Figura 16 - Diagrama de caso de uso departamento 51](#_Toc523483750)

[Figura 17 - Diagrama de caso de uso dependente 52](#_Toc523483751)

[Figura 18 - Diagrama de Sequencia 52](#_Toc523483752)

[Figura 19 - Diagrama de Estados 53](#_Toc523483753)

[Figura 20 - Diagrama de Classes 53](#_Toc523483754)

[Figura 21 - Diagrama de atividades 54](#_Toc523483755)

**Sumario**

1. Introdução 1

1.1. Contextualização 1

1.2. Objetivos 2

1.2.1. Objetivo Geral 2

1.2.2. Objetivo especifico 2

1.3. Metodologia 2

1.4. Problematização 2

1.5. Justificativa 2

2. Fundamentação Teórica 3

2.1. Porto 3

2.2. Terminal Portuário (Importação) 3

2.3. Dados 3

2.4. Informação 3

2.5. Sistema de informação 3

2.6. Software 3

2.6.1. Software de Sistema 3

2.6.2. Software Aplicativo 3

2.6.3. Software de Programação 4

2.7. Aplicativos para gerar Relatórios 4

2.7.1. mPDF 4

2.7.2. iText 4

2.8. Linguagem de Programação 4

2.8.1. Linguagem de Alto nível 4

2.8.2. Linguagem de Baixo nível 4

2.9. Linguagem de Maquina 4

2.10. Ferramentas CASE (Computer Aided Software Engineering) 5

2.10.1. Modelação 5

2.10.2. IDE (Integral Development Envorement) 5

2.11. Base de Dados 5

2.11.1. Base de Dados para Sistemas de Processamento de Transações Online (OLTP) 5

2.11.2. Base de Dados para Sistemas de Apoio e Decisão (OLAP) 5

2.11.3. Base de Dados de Referencia ou Fonte 6

2.12. Sistema de Gestão de Base de Dados 6

2.13. Analise de Sistema 6

2.13.1. Analise estruturada 6

2.13.2. Diagrama de Fluxo de Dados (DFD) 6

2.13.3. Diagrama de Contexto 7

2.13.4. Diagrama / Tabela de Decisão 8

2.13.5. Analise Orientada a Objetos 8

2.13.6. Diagrama de Caso de Uso 9

2.13.7. Diagrama de Sequencia 9

2.13.8. Diagrama de Estados 10

2.13.9. Diagrama de Classes 11

2.14. Metodologias Ágeis de Desenvolvimento de Sistemas 11

2.14.1. SCRUM 12

2.14.2. XP (eXtreme Programming) 12

2.14.3. RUP (Rational Unified Process) 12

2.14.4. MSF (Microsoft Solutions FrameWork) 12

3. Caso de Estudo 13

3.1. Situação Atual 13

3.2. Descrição do Sistema Atual 13

3.3. Estrutura Orgânica 14

3.4. Definição do Problema 14

4. Desenvolvimento do Sistema Proposto 15

4.1. Descrição do Sistema Proposto 15

4.2. Ferramentas Usadas 15

4.2.1. Linguagem de Programação JAVA 15

4.2.2. Linguagem de Programação PHP 15

4.2.3. Linguagem de Programação JavaScript 15

4.2.4. HTML 15

4.2.5. mPDF 16

4.2.6. iText 16

4.2.7. NetBeans 16

4.2.8. StarUML 16

4.2.9. MySQL 16

4.2.10. Linguagem de Modelação UML 16

4.3. Modelação de Dados 16

4.3.1. Diagrama Entidade Relacionamento 17

4.3.2. Diagrama e descrição de Caso de Uso 17

4.3.2.1. Descrição de caso de uso 17

4.3.2.2. Diagrama de caso de uso 48

4.3.3. Diagrama de Sequencia 52

4.3.4. Diagrama de Estados 52

4.3.5. Diagrama de Classes 53

4.3.6. Diagrama de atividades 54

5. Conclusão e Recomendações 55

5.1. Conclusão 55

5.2. Recomendações 55

6. Referências Bibliográficas 56

1. Introdução
   1. Contextualização

A muitos anos, o Homem sentiu necessidade de criar um meio de transporte que permitisse viajar sobe o componente mais abundante na terra, à agua. Dai surgem as embarcações, que com o passar do tempo vem aumentando de tamanhos, formas, tipos e ate das mais distintas denominações.

Com o passar do tempo, são descobertos novos mundos, lugares e continentes, com condições precárias, oque criou necessidade de transporte de produtos, pessoas, e mais para diversas localidades.

Atualmente o transporte marítimo, tem sido oque mais tem influencia no comercio, transporte de mercadorias e turismo a nível mundial. Como todos os outros meios de transporte, tinha de haver um terminal para albergar estas embarcações, cuja denominação é Porto.

Sendo assim, no nosso país não é diferente. Atualmente, Moçambique é o terceiro país com maiores portos da Africa Austral, que são responsáveis pelas entradas e saídas de mercadorias em mais de 50%.

Porem, nota-se que organizações responsáveis pela concessão dos portos moçambicanos, apresentam dificuldades graves na gestão dos mesmos, devido a falta de pessoal capacitado, meios tecnológicos, que abre espaço para roubos, fraudes, entre outos.

Nos dias de hoje, as tecnologias de informações têm trazido inovações que ajudam no melhoramento da gestão dos mesmos, porem os nossos não dispõem dos mesmos

.

* 1. Objetivos
     1. Objetivo Geral
* Desenvolver um Sistema de Gestão de Terminal Portuário (importação)
  + 1. Objetivo especifico
* Analisar o funcionamento dos portos (através de entrevistas);
* Identificar os problemas do sistema atual;
* Solucionar os problemas (através da apresentação de um sistema de gestão recente);
* Desenvolver a solução proposta.
  1. Metodologia

Metodologia é um conjunto de métodos para se chegar a um fim. O seu objetivo é de analisar o quê deverá ser feito.

As fases de elaboração são:

* Definir objetivos do trabalho;
* Fazer entrevistas, de modo a conhecer o funcionamento e definir oque será implementado na organização.
  1. Problematização

Todas as operações, requerimentos, e outras, são armazenadas no papel. A probabilidade de perda, e de conflito é maior, podendo causar congestionamento no terminal. Devido a esse facto, há necessidade de automatizar as atividades.

* 1. Justificativa

Com vista no melhoramento, surge a necessidade de desenvolver-se um software, que será responsável pela automatização das actividades que são realizadas.

As informações referentes a navios, mercadorias, funcionários e demais serão armazenadas numa base de dados, que serão manipulados através do sistema.

1. Definicoes
   1. Porto

Considera-se Porto, todo trecho de mar, rio ou lago, próximo à costa, com profundidade suficiente para as embarcações atracarem e ter acesso fácil à margem.

* 1. Terminal Portuário (Importação)

Como o nome diz, é uma infraestrutura situada num porto, que se destina ao transporte de passageiros e as mercadorias importadas para comercialização.

* 1. Dados

Dados é conjunto de valores pela qual através deles são obtidas informações para obter beneficio. Eles são divididos em estruturados (organizados em tabela) e não estruturados (mais difíceis de processar, como imagens, mensagens, documentos).

* 1. Informação

Em informática, informação é o resultado do processamento de dados, de modo a modificar o conhecimento do sistema.

* 1. Sistema de informação

Actualmente, este termo é usado para descrever qualquer sistema, seja ele automatizado (computorizado) ou manual (que abrange pessoas). Assim sendo, tudo que manipula dados e gera informação é sistema de informação.

* 1. Software

é uma sequência de instruções a serem seguidas e/ou executadas, na manipulação, redireccionamento ou modificação de um dado/informação ou acontecimento.

* + 1. Software de Sistema

É todo software que engloba diversos programas que permite ao usuário interagir com a maquina e seus periféricos. Muitas vezes denominado Sistema Operativo.

**Ex.** Microsoft Windows, Apple Mac OS, Android, Ubuntu.

* + 1. Software Aplicativo

É todo software que permite ao usuário executar determinadas tarefas, dentro dum Software de Sistema.

**Ex.** Microsoft Office, Jogos, VLC, Google Chrome.

* + 1. Software de Programação

É o conjunto de ferramentas que permitem ao programador desenvolver Softwares.

* 1. Aplicativos para gerar Relatórios
     1. mPDF

Essa ferramenta é responsável por gerar e manipular documentação do tipo PDF na linguagem de programação PHP.

* + 1. iText

É uma biblioteca e/ou ferramenta que não se difere da referida acima, porem essa é para uso na linguagem de programação JAVA.

* 1. Linguagem de Programação

Linguagem de programação, são métodos padronizados para informar instruções a uma maquina. Permite que um programador especifique quais dados o computador ira atuar, armazenar e transmitir os mesmos em determinadas funções. Ela pode ser dividida em duas partes: Alto e Baixo nível.

* + 1. Linguagem de Alto nível

É toda linguagem de programação, cuja sintaxe se aproxima mais da nossa linguagem e se distancia da linguagem de maquina. Ela permite que o ser humano, facilmente entenda o código.

**Ex.** JavaScript, Phython, PHP e Ruby.

* + 1. Linguagem de Baixo nível

É toda linguagem de programação que mais se aproxima da linguagem de maquina. Esse tipo de linguagem, requer mais conhecimento da arquitetura do computador para fazer qualquer coisa.

**Ex.** Assembly, Linguagem de Maquina.

* 1. Linguagem de Maquina

Linguagem de Maquina, são códigos em sequencia de Bytes que tratam de instruções a serem executadas pelo processador. Essas instruções são representadas por (0 e 1) e chamam-se Opcodes.

* 1. Ferramentas CASE (Computer Aided Software Engineering)

Ferramentas CASE, pode ser definido como ferramentas computacionais automatizadas que auxiliam no desenvolvimento de sistemas em uma ou varias etapas.

* + 1. Modelação

Modelação de Software, é contruir gráficos que expliquem características e/ou funcionalidades de um sistema de software. Para o mesmo usa-se o UML (Linguagem de Modelação Unificada) e BPM (Modelação de Processos de Negocio).

**Ex.** StarUML, Microsoft Visual Modeler, Umbrello UML, BPMN, JUDE

* + 1. IDE (Integral Development Envorement)

Ambiente Integral de Desenvolvimento, é um software criado para ajudar no desenvolvimento de programas, com o objetivo de simplificar o processo.

As principais ferramentas comuns são:

* Editor;
* Compilador;
* Linker;
* Depurador;
* Modelagem;
* Gerador de Código;
* Teste Automáticos;
* Refactoração.

**Ex.** NetBeans, Android Studio, Brackets, Eclipse, Visual Studio, Swift.

* 1. Base de Dados

É qualquer conjunto de dados organizados de modo a ter sentido. Elas são divididas em três (OLAP, OLTP e Base de Referencia).

**Ex.** Agenda Telefónica, livro, apontamentos de aulas, lista de discos.

* + 1. Base de Dados para Sistemas de Processamento de Transações Online (OLTP)

Esse tipo de babe de dados, é usada na automação de sistemas informáticos.

* + 1. Base de Dados para Sistemas de Apoio e Decisão (OLAP)

Esse tipo é usado no apoio a decisão.

* + 1. Base de Dados de Referencia ou Fonte

Essa base de dados, é usado para armazenar meta dados de recurso informacionais, permitindo fácil recuperação dos mesmos.

* 1. Sistema de Gestão de Base de Dados

é conjunto de programas informáticos, responsáveis por gerir as Bases de Dados.

**Ex.** Firebird, MySQL (Maria DB), Oracle, SQL-Server, Microsoft Access.

* 1. Analise de Sistema

Todo sistema na fase de criação, precisa duma analise detalhada sobre a área de trabalho, de modo a definir as necessidades para sustentar o mesmo.

Existem dois tipos de analise:

* **Estruturada-** define o conteúdo e o fluxo de informação que são usados pelo sistema;
* **Orientada a objeto-** define a compreensão e o processamento da realidade obtendo o conceito que as pessoas tenham adquirido.
  + 1. Analise estruturada

A análise estruturada, tem como objetivos centrais:

* Descrever os principais processos que são realizados na instituição;
* Identificar e definir quais as aplicações necessárias para melhor suporte dos processos da instituição e para a gestão dos dados de alguma importância para estes processos.

Na análise estruturada, as técnicas utilizadas com maior frequência são: o diagrama de fluxo, o dicionário de dados, o diagrama de contexto e o modelo entidade-relacionamento/associação.

* + 1. Diagrama de Fluxo de Dados (DFD)

É uma representação gráfica do fluxo de dados através de um sistema de informação modelando seus aspetos de processo. Fornece apenas uma visão do sistema, a visão estruturada das funções. Um DFD mostra que tipo de informação entrará e sairá do sistema, para onde os dados virão e irão e onde os dados serão armazenados.

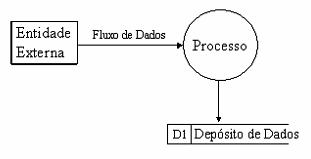


Figura 1- Diagrama de Fluxo de Dados (exemplo)

* + 1. Diagrama de Contexto

É composto por fluxos de dados que mostram as interfaces entre o sistema e as entidades externas. É uma forma de representar o objeto do estudo, o projeto, e sua relação ao ambiente. Permite identificar os limites dos processos, as áreas envolvidas com o processo e os relacionamentos entre os processos e elementos externos. Não é objetivo de o diagrama representar os processos internos do sistema.

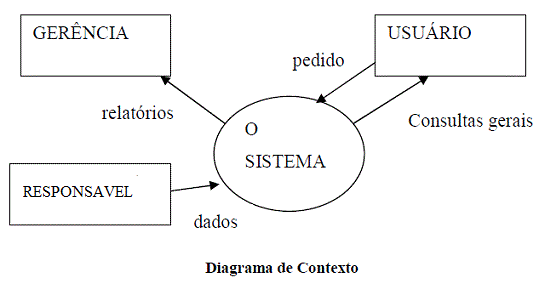


Figura 2 - Diagrama de Contexto (Exemplo)

* + 1. Diagrama / Tabela de Decisão

É uma maneira de expressar, em forma de tabela, qual o conjunto de condições que é necessário ocorrer para que um determinado conjunto de ações deva ser executado.

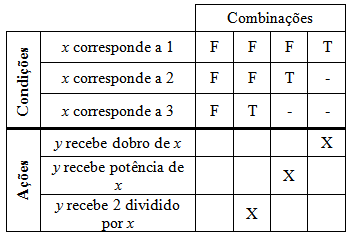


Figura 3 - Diagrama/Tabela de Decisão (Exemplo)

* + 1. Analise Orientada a Objetos

Em AOO, o sistema é um conjunto de objetos que interagem entre si, com características próprias, representadas por atributos (dados) e operações (processos).

|  |  |
| --- | --- |
| **Análise Estruturada**  Módulos  Dados  Processos | **Análise Orientada à Objeto**  Classes  Atributos  Métodos/Operações |

Os obcjetos são abstrações de entidades do mundo real ou de algum sistema, que se Auto gerenciam. Possuem atributos que são os elementos que o caracterizam e os métodos ou operações que ditam o seu comportamento.

A OOA utiliza os diagramas da UML. A seguir, serão ilustrados alguns desses diagramas que foram usados neste trabalho:

* + 1. Diagrama de Caso de Uso

Este diagrama tem foco no comportamento do sistema. Sua construção deve ser feita de forma clara sem ambiguidade. A sua construção é baseada nos requisitos que foram levantados durante a análise do sistema.

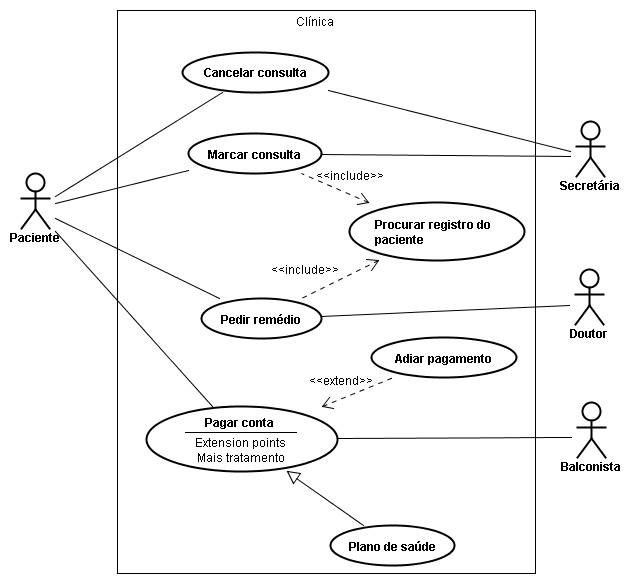


Figura 4 - Digrama de Caso de Uso (Exemplo)

* + 1. Diagrama de Sequencia

Este diagrama serve para demonstrar a interação entre os objetos. Essa interação acontece através de mensagens ordenadas entre os objetos do sistema. Possui dois eixos: vertical, que mostra o tempo e a horizontal que nos mostra os objetos do sistema.

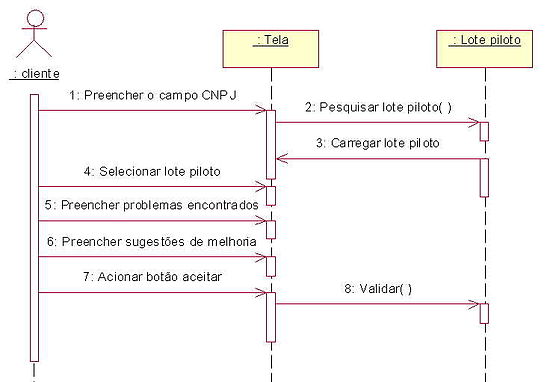


Figura 5 - Diagrama de Sequencia (Exemplo)

* + 1. Diagrama de Estados

Um diagrama de estado descreve como a funcionalidade de um obcjeto depende do seu estado e como este estado é alterado em face do resultado dos eventos que ele recebe.

Existem dois conceitos a conhecer neste diagrama:

- Estado, que representa um período entre eventos durante o qual o objeto é estável;

- Evento, que representa uma transição de um estado inicial para um estado final.

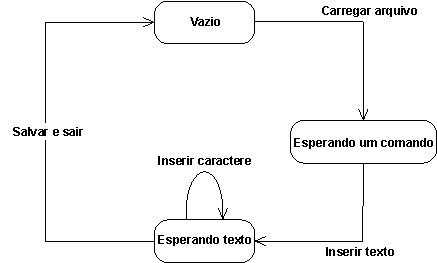


Figura 6 - Diagrama de estados (Exemplo)

* + 1. Diagrama de Classes

O diagrama de classes é uma representação da estrutura e relações das classes que servem de modelo para obcjetos.

É uma modelação muito útil para o desenvolvimento de sistema, pois define todas as classes que o sistema necessita.

Existem quarto conceitos básicos:

- Classe, elemento abstrato que representa um conjunto de objetos;

- Atributo, define as características da classe;

- Operação, função requerida a um objeto abstrato; e

- Associação, que define os relacionamentos entre as classes.

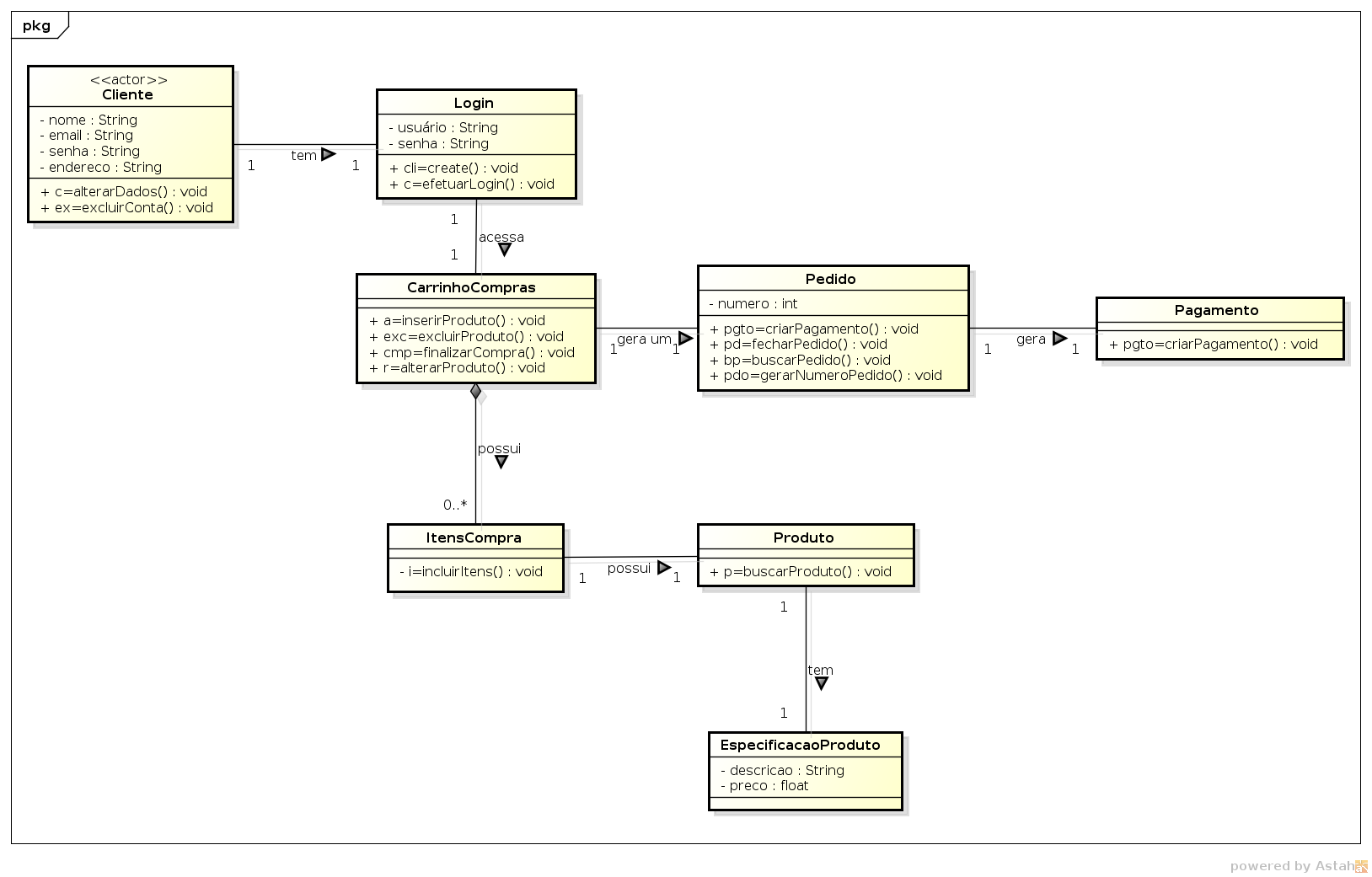


Figura 7 - Diagrama de Classes (Exemplo)

* 1. Metodologias Ágeis de Desenvolvimento de Sistemas

Os métodos ágeis podem ser vistos como uma resposta natural do mercado de TI. a cada ano que passa, estamos nos tornando melhores e mais eficientes quanto ao tempo que levamos para concluir um projeto de software. Estes métodos permitem que um sistema que antigamente era desenvolvido em 5 anos, leve 3 meses no máximo.

Eles tornar-se populares 2001, quando um grupo de 20 especialistas se reuniu nos Estados Unidos para avaliar a forma como os seus projetos de software estavam sendo conduzidos.

* + 1. SCRUM

Nesta metodologia, os projetos são divididos em ciclos chamados **Sprints** (conjunto de atividades que tem de ser executadas num período de tempo).

Em cada inicio de Sprint, é realizada uma reunião de planeamento. Assim como no final, é feita uma reunião para apresentar as funcionalidades implementadas.

* + 1. XP (eXtreme Programming)

Essa metodologia é recomendada, para equipas pequenas e medias que pretendam desenvolver Softwares com vários requisitos e mudanças constantes, onde os principais valores são:

* Comunicação;
* Simplicidade;
* Feedback;
* Coragem;
* Respeito.
  + 1. RUP (Rational Unified Process)

É uma metodologia criada pela Rational Software, ela é recomendada para equipes grandes por ser um processo pesado, porem é customizada.

O RUP é um produto de software, modular e automatizado, e é apoiada por diversas ferramentas da IBM.

* + 1. MSF (Microsoft Solutions FrameWork)

Essa metodologia é proprietária da Microsoft, e tem sido usado por ela mesma e diversos clientes e parceiros em todo o mundo.

O MSF é dividido em ciclos, onde cada membro da equipe executa sua função e atualiza o mesmo em função da necessidade. Estes ciclos repetem-se ate que o projeto seja concluído.

Porem a Microsoft não classifica o MSF como uma metodologia, mas sim como uma disciplina de boas praticas.

1. Caso de Estudo
   1. Situação Atual

Porto de Maputo, inaugurado em 1972, situado na baixa da cidade, praça dos trabalhadores, CP 2841, gerida pela MPDC (Maputo Port Development Company), concessionaria com direitos de operar ate 2033.

A MPDC é uma empresa nacional, que resulta da parceria entre os CFM (Caminhos de Ferros de Moçambique) e a Portus Indico (constituída pela Grindrod, DP World e a empresa moçambicana Mozambique Gestores).

A 15 de abril de 2003 foi atribuída a ela a concessão do Porto de Maputo por um período de 15 anos com opção de extensão por mais 15 anos. Em 2010 o período de conceção foi estendido por mais 15 anos (como mencionado acima), com opção de extensão de mais 10 anos apos o termino do atual período.

Ela detém os direitos de financiamento, reabilitação, construção, operação, gestão, manutenção, desenvolvimento e otimização de toda área de concessão. A empresa tem também poder de Autoridade Portuária, sendo responsável pelas operações marítimas, reboque, estiva, operações nos terminais e armazéns, bem como planeamento e desenvolvimento portuário.

* 1. Descrição do Sistema Atual

As operações efetuadas diariamente no porto, são armazenadas em livros, o quê torna impossível obter informações em tempo real.

As requisições de permissão de atracagem, partida, chegada, entrada e saída de meios de transporte, pessoas e mercadorias, é feita por emails, cartas, chamadas telefónicas e/ou entre os funcionários, responsáveis (de maneira presencial), oque leva muito tempo.

* 1. Estrutura Orgânica

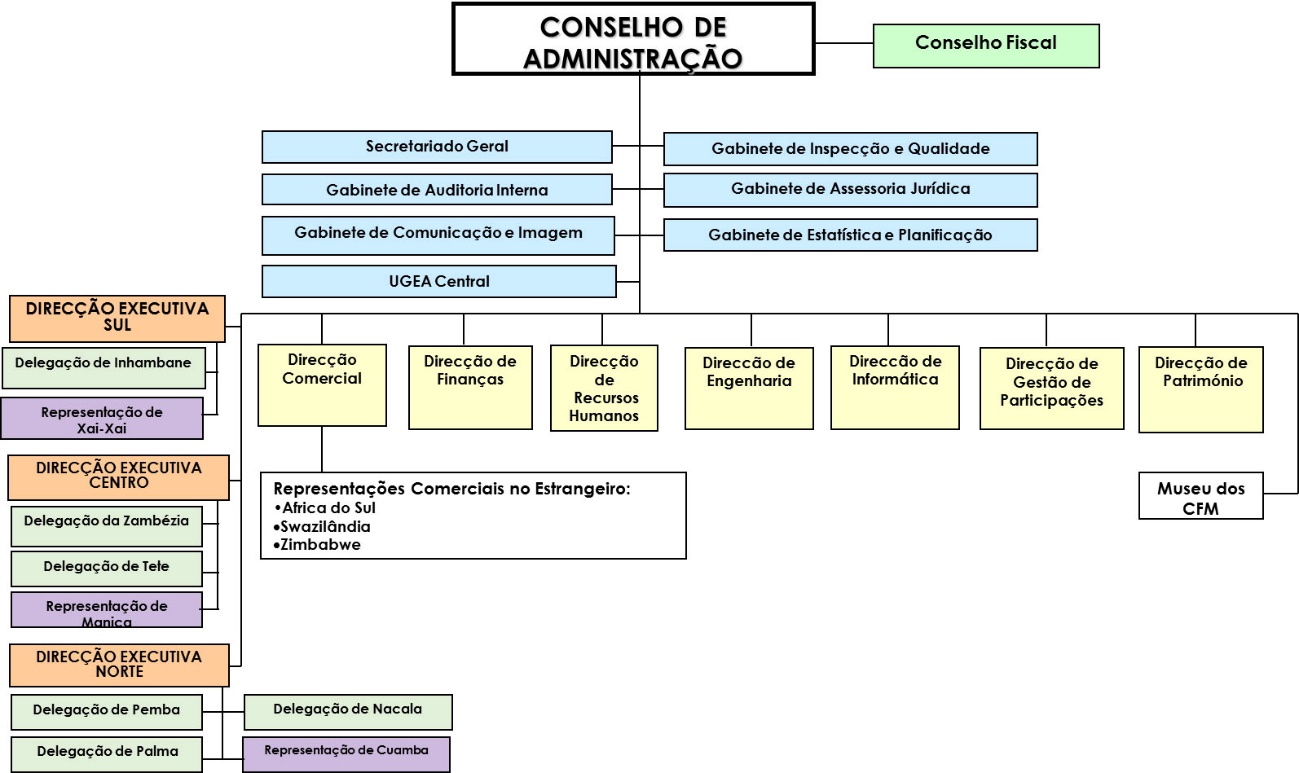


Figura 8 - Estrutura Orgânica (Portos e Caminhos de Ferros de Moçambique)

* 1. Definição do Problema

O sistema usado actualmente pela MPDC, é bom, porem não o suficiente no quesito de segurança, portabilidade e busca de informações, atrasando assim as operações portuárias, o que não devia acontecer devido ao alto fluxo de informações num dos principais portos da SADC.

Com a presente plataforma, as mesmas operações que agora são feitas em 2 horas numa única maquinam, serão feitas em 1 minuto no máximo e acedidas/ controladas em N maquinas em qualquer parte do mundo.

1. Desenvolvimento do Sistema Proposto
   1. Descrição do Sistema Proposto

O presente sistema, oferece planeamento eficaz, redução de custos, controle real das operações portuárias, de maneira pratica com integração de funcionários, clientes, agencias, órgãos governamentais. Alem de ajudar na tomada de decisões com informações atualizadas e confiáveis, melhorando a qualidade e eficiência dos serviços e instalações.

**Principais Funções**

* Fornecer previsão de chegada de navios e cargas;
* Programar e controlar operações (terra/navio);
* Registar todas as movimentações de cargas;
* Controlar a entrada e saída de carga, equipamentos e pessoas;
* Informar sobre operações em andamento;
* Calculo automático dos preços de permanecia, bem como multas e prazos;
* Programação previa de recebimentos e entregas de carga
  1. Ferramentas Usadas
     1. Linguagem de Programação JAVA

Essa linguagem, tendo em conta que o sistema é para ser usado numa organização, ela é segura, portátil (o mesmo código, pode rodar em qualquer maquina) e tem recursos de rede.

* + 1. Linguagem de Programação PHP

Essa linguagem permite que qualquer conteúdo do sistema, seja acedido remotamente em qualquer dispositivo conectado a internet, facilitando os funcionários, clientes e todos aqueles que não trabalhem no escritório (operacionais).

* + 1. Linguagem de Programação JavaScript

É uma linguagem que cria e/ou auxilia nas animações e funções, bem como no design da plataforma em HTML e na linguagem PHP.

* + 1. HTML

É um formato padrão para criação de paginas online e aplicativos webs, em conjunto com JavaScript.

* + 1. NetBeans

Para o desenvolvimento do presente sistema, escolhi essa IDE, pois ela é fácil de usar, suporta todas as linguagens e ferramentas referidas acima, é pratica e eficiente.

* + 1. StarUML

Usei essa ferramenta de modelação, por ser pratica e fácil de usar.

* + 1. MySQL

Usei esse sistema de gestão de base de dados, pelo seu excelente desempenho, exige pouco da máquina e compatibilidade com as linguagens que usei no desenvolvimento.

* + 1. Linguagem de Modelação UML

Uso da mesma, para a criação de todos os diagramas que compõem o sistema. Que é feito pelo StarUML, como referi acima.

# Apache

Usei esse servidor, por ser o mais usado a nivel mundial e por permitir o acesso remote e criacao de varias aplicacoes Web em módulos.

* 1. Modelação de Dados

O ER, é um mecanismo de modelação de dados, baseado no conceito de que o mundo real é constituído por conjunto de objetos chamados de entidades e pelo conjunto de relacionamentos entre estes objetos. Existem três conceitos relevantes no ER:

- A entidade, que define um objeto do mundo real. Normalmente são representados por retângulos;

- Os atributos, são propriedades que caracterizam cada membro de um conjunto de propriedades. Estes são representados por elipses;

- Relacionamentos, associação entre uma ou várias entidades. São representados por losangos.

* + 1. Diagrama Entidade Relacionamento

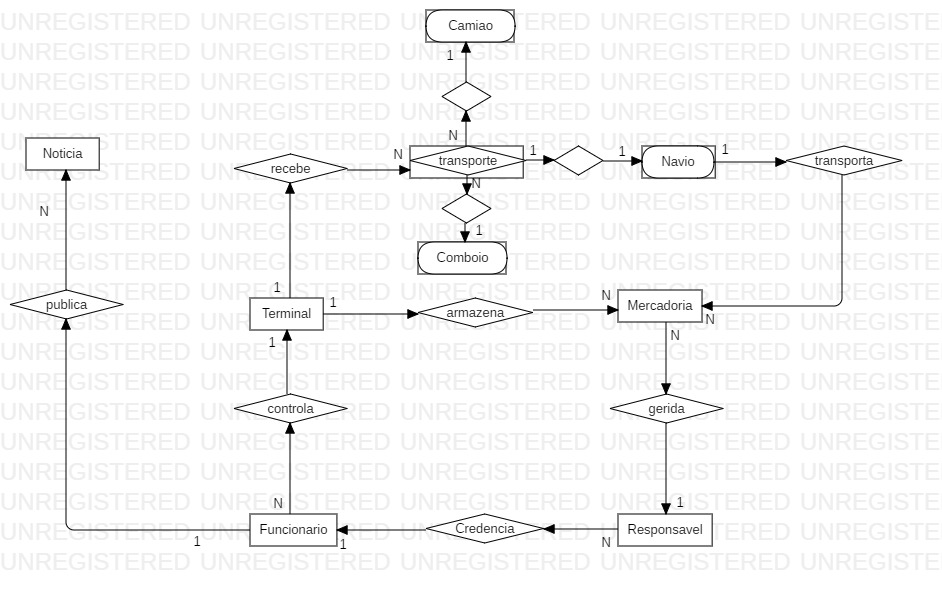


Figura 9- Diagrama Entidade Relacionamento

* + 1. Diagrama e descrição de Caso de Uso
       1. Descrição de caso de uso

|  |  |
| --- | --- |
| **Nome:** Cadastrar funcionário | |
| **Actor:** Administrador, utilizador | |
| **Campos:** nome, email, endereço, telefone, senha (caixa de texto), cargo (caixa de seleção), cadastrar (botão) | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** Funcionário cadastrado na base de dados | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu do funcionário | 4.valida dados preenchidos |
| 2.preenche os dados do funcionário | 5.cadastra funcionário |
| 3.clica no botão cadastrar |  |
| **Sequência alternativa** | |
| 4a.Dados incorretos | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Procurar funcionário | |
| **Actor:** Administrador | |
| **Campos:** nup (caixa de texto), procurar (botão), tabela | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** Procurar funcionário na base de dados | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu do funcionário | 4.valida nup preenchido |
| 2.preenche o nup do funcionário | 5.procura funcionário |
| 3.clica no botão procurar |  |
| 6.visualiza funcionário |  |
| **Sequência alternativa** | |
| 4a.nup invalido | |
| 6a.funcionario não encontrado | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Editar funcionário | |
| **Actor:** Administrador | |
| **Campos:** nome, email, endereço, telefone(caixa de texto), cargo (caixa de seleção), editar (botão), tabela | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** Funcionário editado na base de dados | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu do funcionário | 5.valida dados preenchidos |
| 2.seleciona o funcionário na tabela | 6.edita dados do funcionário |
| 3.edita os dados do funcionário |  |
| 4.clica no botão editar |  |
| **Sequência alternativa** | |
| 5a.Dados incorretos | |
| 2a.funcionario não selecionado | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Editar funcionário | |
| **Actor:** Usuário | |
| **Campos:** nome, email, endereço, telefone, senha(caixa de texto), gravar (botão) | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** Funcionário editado na base de dados | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu mais | 5.valida dados preenchidos |
| 2.seleciona editar | 6.edita dados do funcionário |
| 3.edita os dados |  |
| 4.clica no botão gravar |  |
| **Sequência alternativa** | |
| 5a.Dados incorretos | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Editar senha funcionário | |
| **Actor:** Administrador | |
| **Campos:** senha(caixa de texto), editar senha (botão), tabela | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** Senha do funcionário editado na base de dados | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu do funcionário | 5.valida dado preenchido |
| 2.seleciona o funcionário na tabela | 6.edita dado do funcionário |
| 3.edita a senha do funcionário |  |
| 4.clica no botão editar senha |  |
| **Sequência alternativa** | |
| 5a.Dado incorreto | |
| 2a.funcionario não selecionado | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Visualizar funcionário | |
| **Actor:** Administrador | |
| **Campos:** tabela | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** Visualizar funcionário | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu do funcionário | 2.procura todos funcionários |
| 3.visualizar funcionário |  |
| **Sequência alternativa** | |
| 3a.sem funcionários | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Recontratar funcionário | |
| **Actor:** Administrador | |
| **Campos:** recontratar (botão), tabela | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** funcionário recontratado | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu do funcionário | 4.procura funcionário |
| 2.seleciona o funcionário na tabela | 5.recontrata funcionário |
| 3.clica no botão recontratar |  |
| **Sequência alternativa** | |
| 5a.funcionario já foi recontratado | |
| 2a.funcionario não selecionado | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Despedir funcionário | |
| **Actor:** Administrador | |
| **Campos:** despedir (botão), tabela | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** funcionário despedido | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu do funcionário | 4.procura funcionário |
| 2.seleciona o funcionário na tabela | 5.despede funcionário |
| 3.clica no botão despedir |  |
| **Sequência alternativa** | |
| 5a.funcionario já foi despedido | |
| 2a.funcionario não selecionado | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Cadastrar navio | |
| **Actor:** Usuário | |
| **Campos:** matricula, nome, email, telefone, senha(caixa de texto), tipo, pais (caixa de seleção) e registar (botão) | |
| **Pré-condição:** aceder o sistema web | |
| **Pôs-condição:** Navio cadastrado na base de dados | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu navio | 4.valida dados preenchidos |
| 2.preenche os dados do navio | 5.cadastra dados do navio |
| 3.clica no botão registar |  |
| **Sequência alternativa** | |
| 4a.Dados incorretos | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Editar navio | |
| **Actor:** Usuário | |
| **Campos:** matricula, nome, email, telefone, senha(caixa de texto), gravar (botão) | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** navio editado na base de dados | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu mais | 5.valida dados preenchidos |
| 2.seleciona editar info | 6.edita dados do navio |
| 3.edita os dados |  |
| 4.clica no botão gravar |  |
| **Sequência alternativa** | |
| 5a.Dados incorretos | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Procurar navio | |
| **Actor:** Administrador e Usuário | |
| **Campos:** nup (caixa de texto), procurar (botão), tabela | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** Procurar navio na base de dados | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu do navio | 4.valida nup preenchido |
| 2.preenche o nup do navio | 5.procura navio |
| 3.clica no botão procurar |  |
| 6.visualiza navio |  |
| **Sequência alternativa** | |
| 4a.nup invalido | |
| 6a. navio não encontrado | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Visualizar navio | |
| **Actor:** Usuário e Administrador | |
| **Campos:** tabela | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** Visualizar navio | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu navio | 3.procurar todos os navios |
| 2.visualizar navio |  |
| **Sequência alternativa** | |
| 3a.sem navios | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Editar senha navio | |
| **Actor:** Administrador | |
| **Campos:** senha(caixa de texto), editar senha (botão), tabela | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** Senha do navio editado na base de dados | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu do navio | 5.valida dado preenchido |
| 2.seleciona o navio na tabela | 6.edita dado do navio |
| 3.edita a senha do navio |  |
| 4.clica no botão editar senha |  |
| **Sequência alternativa** | |
| 5a.Dado incorreto | |
| 2a. navio não selecionado | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Marcar chegada navio | |
| **Actor:** Usuário | |
| **Campos:** data de chegada, passageiros, senha(caixa de texto), gravar (botão) | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** Chegada do navio marcada | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu marcar chegada | 4.valida dados preenchidos |
| 2.preecher os dados da chegada | 5.marca chegada do navio |
| 3.clica no botão gravar |  |
| **Sequência alternativa** | |
| 4a.Dados incorretos | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Marcar partida navio | |
| **Actor:** Usuário | |
| **Campos:** data de partida, passageiros, senha(caixa de texto), gravar (botão) | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** partida do navio marcada | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu marcar partida | 4.valida dados preenchidos |
| 2.preecher os dados da partida | 5.marca partida do navio |
| 3.clica no botão gravar |  |
| **Sequência alternativa** | |
| 4a.Dados incorretos | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Apagar navio | |
| **Actor:** Administrador | |
| **Campos:** apagar (botão), tabela | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** navio apagado | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu do navio | 4.procura navio |
| 2.seleciona o navio na tabela | 5.apaga navio |
| 3.clica no botão apagar |  |
| **Sequência alternativa** | |
| 2a. navio não selecionado | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Cancelar chegada navio | |
| **Actor:** Administrador | |
| **Campos:** cancelar chegada (botão), tabela | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** Chegada do navio cancelada | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu do navio | 4.procura navio |
| 2.seleciona o navio na tabela | 5.cancela chegada do navio |
| 3.clica no botão cancelar chegada |  |
| **Sequência alternativa** | |
| 5a.Dado incorreto | |
| 2a. navio não selecionado | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Cancelar partida navio | |
| **Actor:** Administrador | |
| **Campos:** cancelar partida (botão), tabela | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** partida do navio cancelada | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu do navio | 4.procura navio |
| 2.seleciona o navio na tabela | 5.cancela partida do navio |
| 3.clica no botão cancelar partida |  |
| **Sequência alternativa** | |
| 5a.Dado incorreto | |
| 2a. navio não selecionado | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** atracar navio | |
| **Actor:** Administrador | |
| **Campos:** atracar (botão), tabela | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** navio atracado | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu do navio | 4.procura navio |
| 2.seleciona o navio na tabela | 5.atraca navio |
| 3.clica no botão atracar |  |
| **Sequência alternativa** | |
| 5a.Dado incorreto | |
| 2a. navio não selecionado | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** descarregar navio | |
| **Actor:** Usuário | |
| **Campos:** descarregar (botão), tabela | |
| **Pré-condição:** Aceder ao sistema web | |
| **Pôs-condição:** mercadoria do navio descarregada | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu do navio | 4.procura navio |
| 2.seleciona o navio na tabela | 5.autoriza descarga de mercadoria |
| 3.clica no botão descarregar |  |
| **Sequência alternativa** | |
| 5a.Dado incorreto | |
| 2a. navio não selecionado | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Cadastrar responsável | |
| **Actor:** Usuário | |
| **Campos:** nome, email, telefone, endereço , senha(caixa de texto), e registar (botão) | |
| **Pré-condição:** aceder o sistema web | |
| **Pôs-condição:** responsável cadastrado na base de dados | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu responsável | 4.valida dados preenchidos |
| 2.preenche os dados do responsável | 5.cadastra dados do responsável |
| 3.clica no botão registar |  |
| **Sequência alternativa** | |
| 4a.Dados incorretos | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Editar responsável | |
| **Actor:** Usuário | |
| **Campos:** nome, email, telefone, senha(caixa de texto), gravar (botão) | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** responsável editado na base de dados | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu mais | 5.valida dados preenchidos |
| 2.seleciona editar info | 6.edita dados do responsável |
| 3.edita os dados |  |
| 4.clica no botão gravar |  |
| **Sequência alternativa** | |
| 5a.Dados incorretos | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Procurar responsável | |
| **Actor:** Administrador e Usuário | |
| **Campos:** nup (caixa de texto), procurar (botão), tabela | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** Procurar responsável na base de dados | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu do responsável | 4.valida nup preenchido |
| 2.preenche o nup do responsável | 5.procura responsável |
| 3.clica no botão procurar |  |
| 6.visualiza responsável |  |
| **Sequência alternativa** | |
| 4a.nup invalido | |
| 6a. responsável não encontrado | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Visualizar responsável | |
| **Actor:** Usuário e Administrador | |
| **Campos:** tabela | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** Visualizar responsável | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu responsável | 3.procurar todos os responsáveis |
| 2.visualizar responsável |  |
| **Sequência alternativa** | |
| 3a.sem responsável | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Editar senha responsável | |
| **Actor:** Administrador | |
| **Campos:** senha(caixa de texto), editar senha (botão), tabela | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** Senha do responsável editado na base de dados | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu do responsável | 5.valida dado preenchido |
| 2.seleciona o responsável na tabela | 6.edita dado do responsável |
| 3.edita a senha do responsável |  |
| 4.clica no botão editar senha |  |
| **Sequência alternativa** | |
| 5a.Dado incorreto | |
| 2a. responsável não selecionado | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Cadastrar mercadoria | |
| **Actor:** Usuário | |
| **Campos:** responsável, quantidade, senha (caixa de texto), situação, tipo de mercadoria (botão de seleção único) e registar (botão) | |
| **Pré-condição:** aceder o sistema web | |
| **Pôs-condição:** mercadoria cadastrado na base de dados | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu mercadoria | 4.valida dados preenchidos |
| 2.preenche os dados do mercadoria | 5.cadastra dados do mercadoria |
| 3.clica no botão registar |  |
| **Sequência alternativa** | |
| 4a.Dados incorretos | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Editar responsável | |
| **Actor:** Administrador | |
| **Campos:** nup mercadoria, responsável (caixa de texto), gravar (botão) | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** mercadoria editado na base de dados | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu mercadoria | 5.valida dados preenchidos |
| 2.seleciona editar responsável | 6.edita dados do mercadoria |
| 3.edita os dados |  |
| 4.clica no botão gravar |  |
| **Sequência alternativa** | |
| 5a.Dados incorretos | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Procurar mercadoria | |
| **Actor:** Administrador e Usuário | |
| **Campos:** nup (caixa de texto), procurar (botão), tabela | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** Procurar mercadoria na base de dados | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu do mercadoria | 4.valida nup preenchido |
| 2.preenche o nup do mercadoria | 5.procura mercadoria |
| 3.clica no botão procurar |  |
| 6.visualiza mercadoria |  |
| **Sequência alternativa** | |
| 4a.nup invalido | |
| 6a. mercadoria não encontrado | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Visualizar mercadoria | |
| **Actor:** Usuário e Administrador | |
| **Campos:** tabela | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** Visualizar mercadoria | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu mercadoria | 3.procurar todos as mercadoria |
| 2.visualizar mercadoria |  |
| **Sequência alternativa** | |
| 3a.sem mercadoria | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** apagar mercadoria | |
| **Actor:** Administrador | |
| **Campos:** nup(caixa de texto), apagar (botão), tabela | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** mercadoria apagada | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu do mercadoria | 4.valida dado preenchido |
| 2.seleciona a mercadoria na tabela | 6.apaga mercadoria |
| 3.clica no botão apagar |  |
| **Sequência alternativa** | |
| 4a.Dado incorreto | |
| 2a. mercadoria não selecionada | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** apreender mercadoria | |
| **Actor:** Administrador e Usuário | |
| **Campos:** apreender (botão), tabela | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** mercadoria apreendida | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu do mercadoria | 4.procura mercadoria |
| 2.seleciona o mercadoria na tabela | 5.apreende mercadoria |
| 3.clica no botão apreender |  |
| **Sequência alternativa** | |
| 5a.Dado incorreto | |
| 2a. mercadoria não selecionado | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Cancelar chegada mercadoria | |
| **Actor:** Administrador | |
| **Campos:** cancelar chegada (botão), tabela | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** chegada da mercadoria cancelada | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu do mercadoria | 4.procura mercadoria |
| 2.seleciona a mercadoria na tabela | 5.cancela chegada da mercadoria |
| 3.clica no botão cancelar chegada |  |
| **Sequência alternativa** | |
| 5a.Dado incorreto | |
| 2a. mercadoria não selecionado | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** despachar mercadoria | |
| **Actor:** Usuário | |
| **Campos:** despachar (botão), tabela | |
| **Pré-condição:** Aceder ao sistema web | |
| **Pôs-condição:** mercadoria despachada | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu do mercadoria | 4.procura mercadoria |
| 2.seleciona o mercadoria na tabela | 5.autoriza despacho de mercadoria |
| 3.clica no botão despachar |  |
| **Sequência alternativa** | |
| 5a.Dado incorreto | |
| 2a. mercadoria não selecionado | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Cadastrar noticia | |
| **Actor:** Usuário e Administrador | |
| **Campos:** titulo, descrição, senha(caixa de texto), e registar (botão) | |
| **Pré-condição:** aceder o sistema web | |
| **Pôs-condição:** noticia cadastrado na base de dados | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu noticia | 4.valida dados preenchidos |
| 2.preenche os dados do noticia | 5.cadastra dados do noticia |
| 3.clica no botão registar |  |
| **Sequência alternativa** | |
| 4a.Dados incorretos | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Procurar noticia | |
| **Actor:** Administrador e Usuário | |
| **Campos:** nup (caixa de texto), procurar (botão), tabela | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** Procurar noticia na base de dados | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu do noticia | 4.valida nup preenchido |
| 2.preenche o nup do noticia | 5.procura noticia |
| 3.clica no botão procurar |  |
| 6.visualiza noticia |  |
| **Sequência alternativa** | |
| 4a.nup invalido | |
| 6a. noticia não encontrado | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Visualizar noticia | |
| **Actor:** Usuário e Administrador | |
| **Campos:** tabela | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** Visualizar noticia | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu noticia | 3.procurar todos os noticia |
| 2.visualizar noticia |  |
| **Sequência alternativa** | |
| 3a.sem noticia | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** Editar noticia | |
| **Actor:** Administrador | |
| **Campos:** nup(caixa de texto), editar (botão), tabela | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** noticia editado na base de dados | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu do noticia | 5.valida dado preenchido |
| 2.seleciona o noticia na tabela | 6.edita dado do noticia |
| 3.edita a noticia |  |
| 4.clica no botão editar |  |
| **Sequência alternativa** | |
| 5a.Dado incorreto | |
| 2a. noticia não selecionado | |
| 1.mostra mensagem de erro | |

|  |  |
| --- | --- |
| **Nome:** apagar noticia | |
| **Actor:** Administrador e Usuário | |
| **Campos:** nup(caixa de texto), apagar (botão), tabela | |
| **Pré-condição:** Iniciar sessão | |
| **Pôs-condição:** noticia apagada na base de dados | |
| **Sequência de eventos** | |
| **Actor** | **Sistema** |
| 1.Abre menu do noticia | 5.valida dado preenchido |
| 2.seleciona o noticia na tabela | 6. apagar dado do noticia |
| 3. apagar a noticia |  |
| 4.clica no botão apagar |  |
| **Sequência alternativa** | |
| 5a.Dado incorreto | |
| 2a. noticia não selecionado | |
| 1.mostra mensagem de erro | |

* + - 1. Diagrama de caso de uso



Figura 10 - Diagrama de caso de uso funcionário

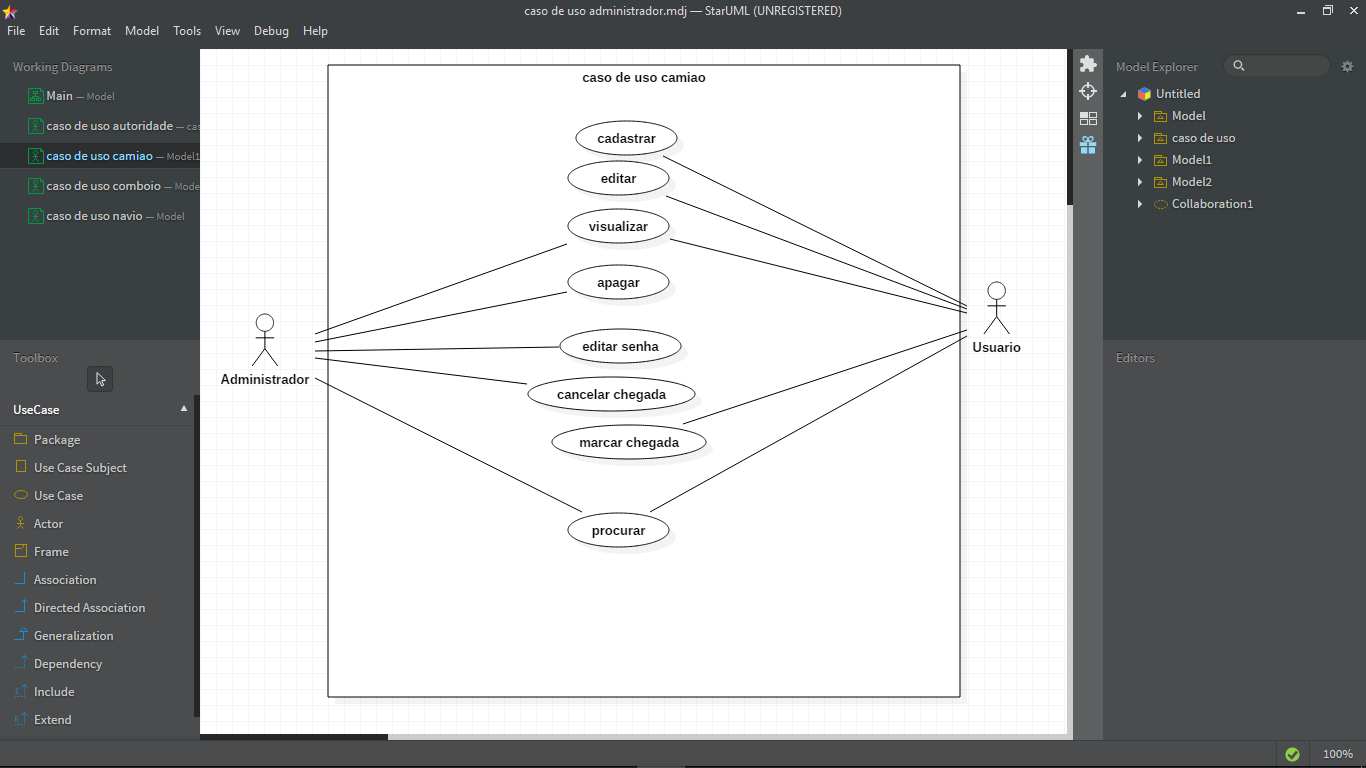


Figura 11 - Diagrama de caso de uso camião

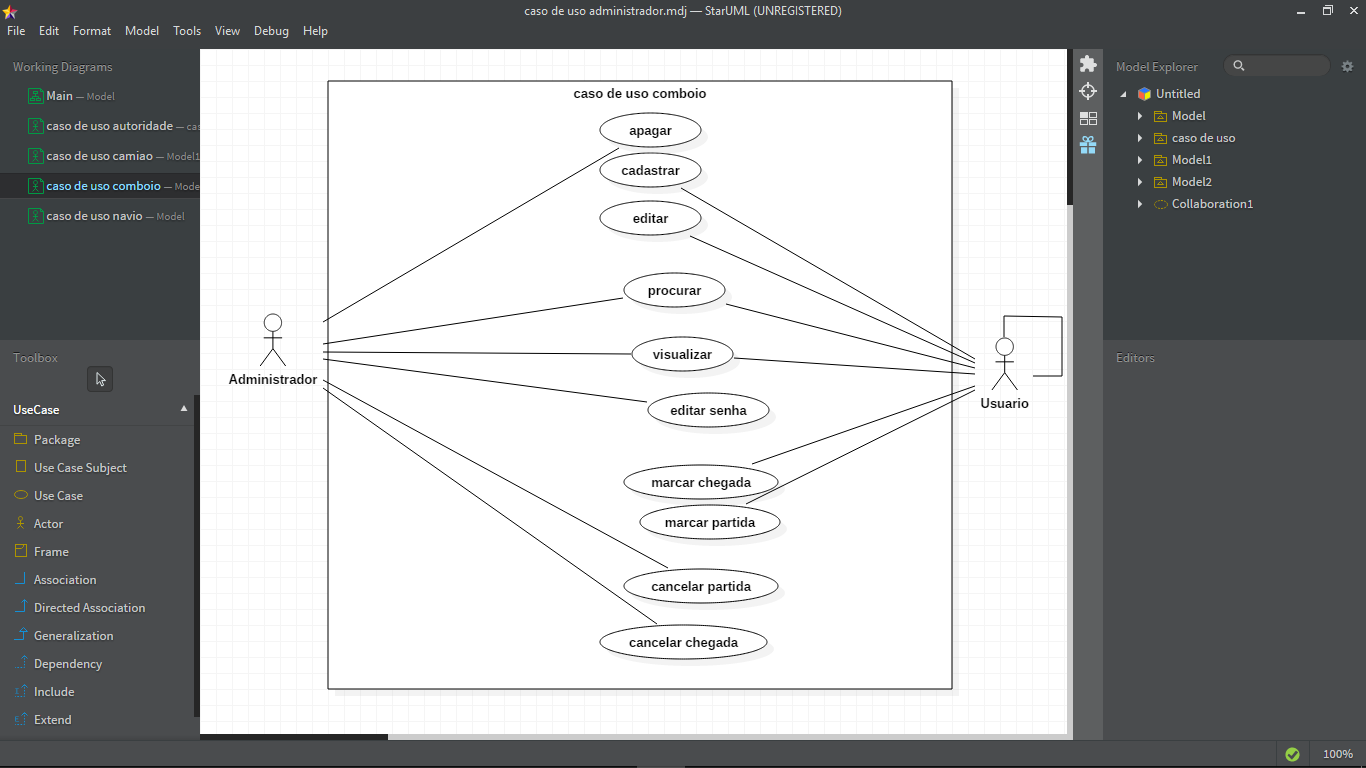


Figura 12 - Diagrama de caso de uso comboio

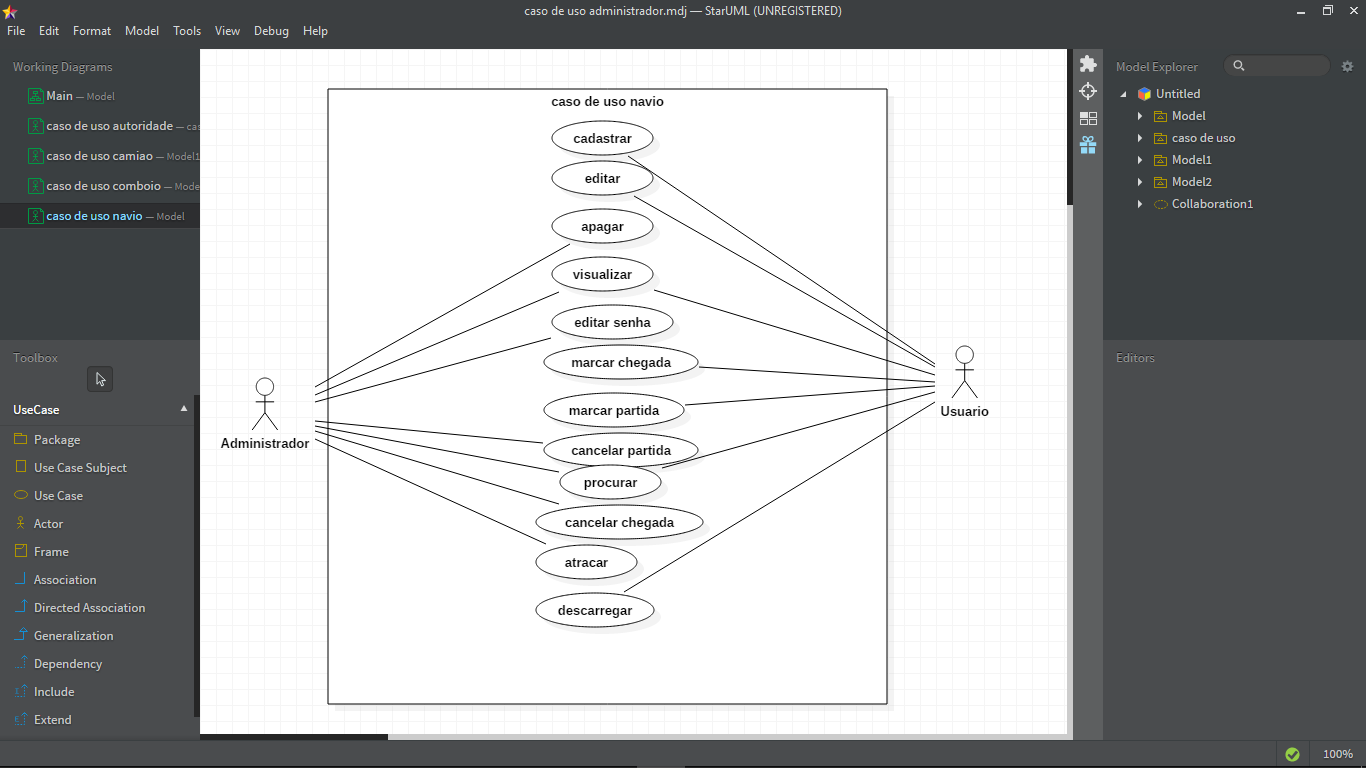


Figura 13 - Diagrama de caso de uso navio

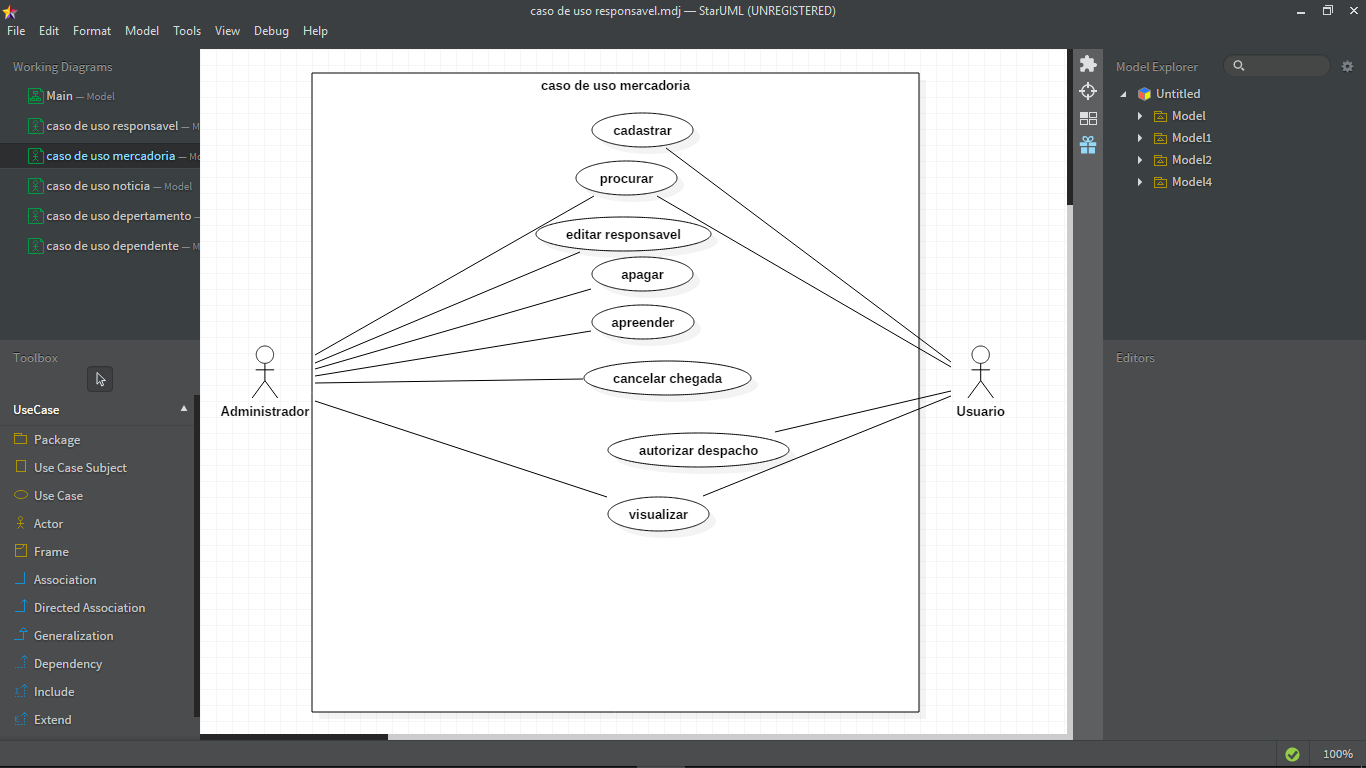


Figura 14 - Diagrama de caso de uso mercadoria

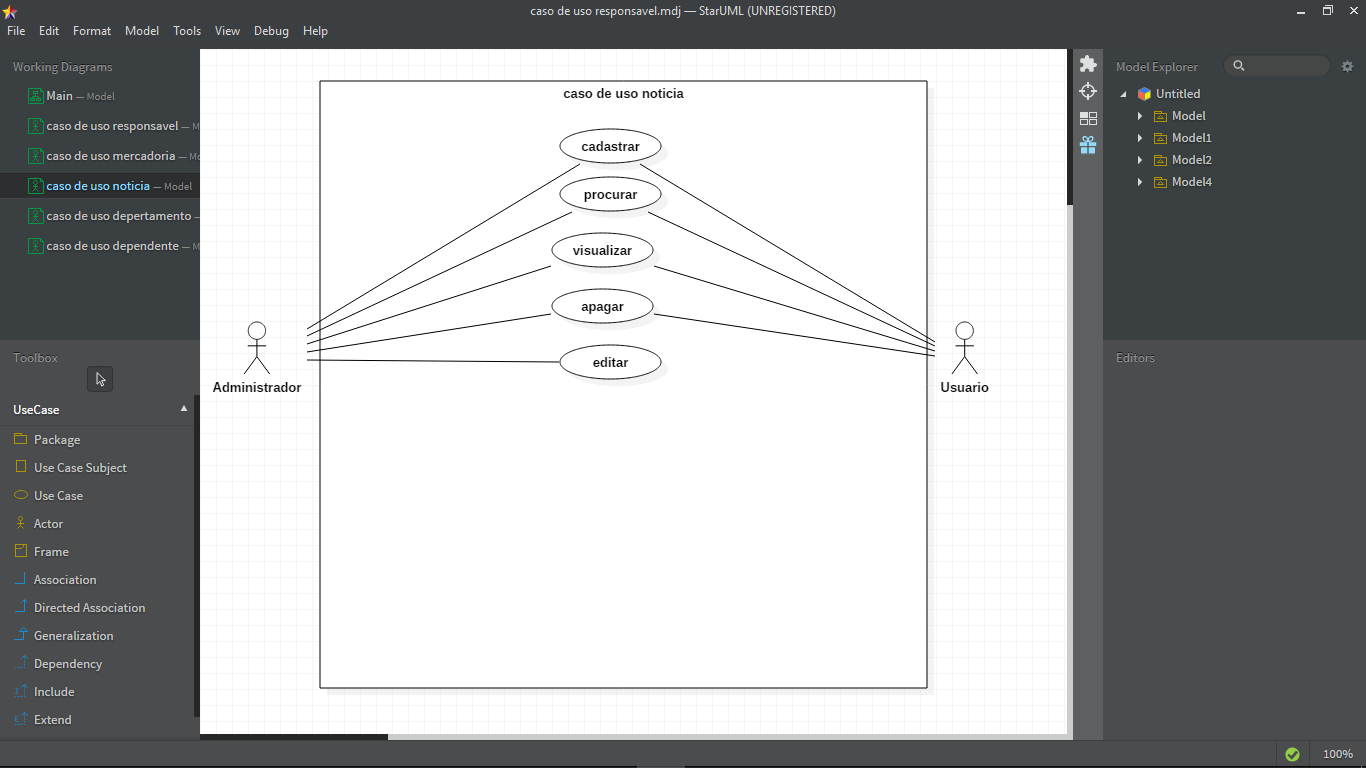


Figura 15 - Diagrama de caso de uso noticia

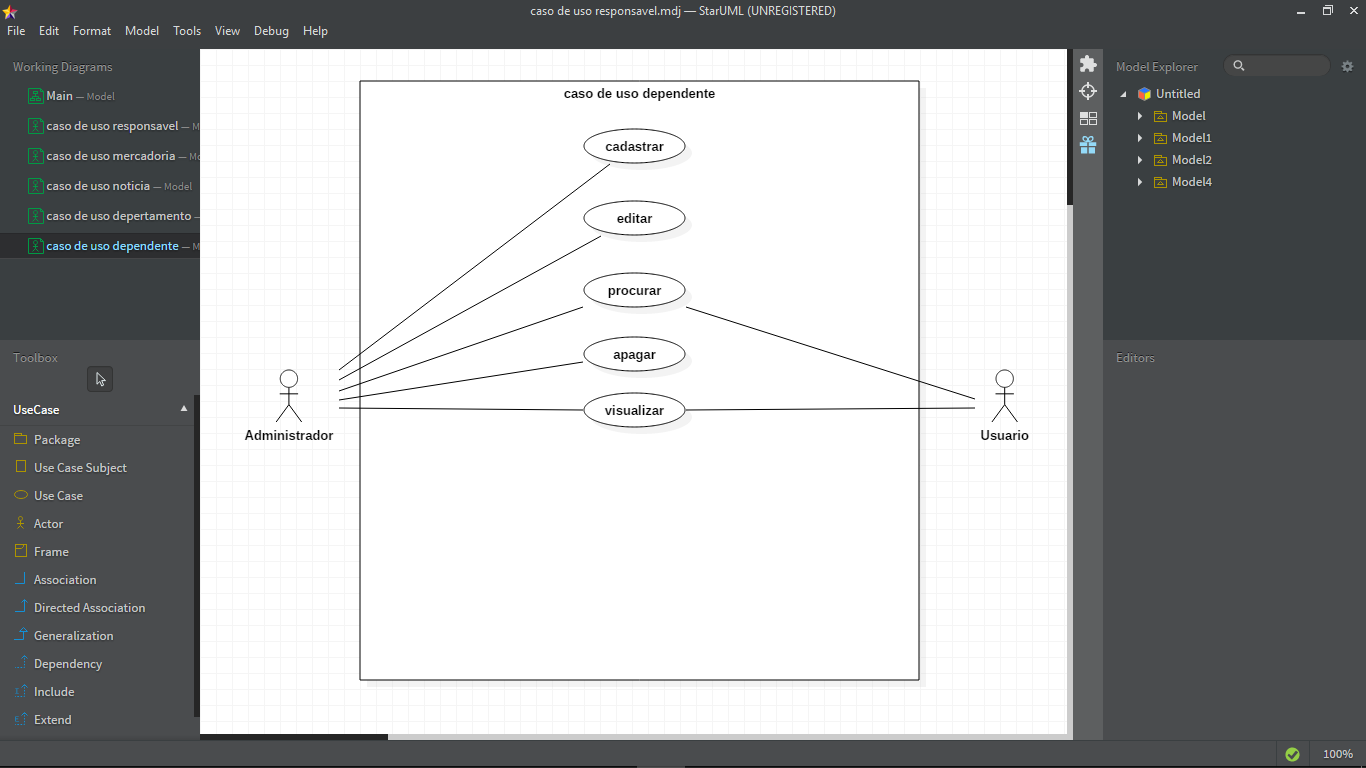


Figura 16 - Diagrama de caso de uso dependente

* + 1. Diagrama de Sequencia

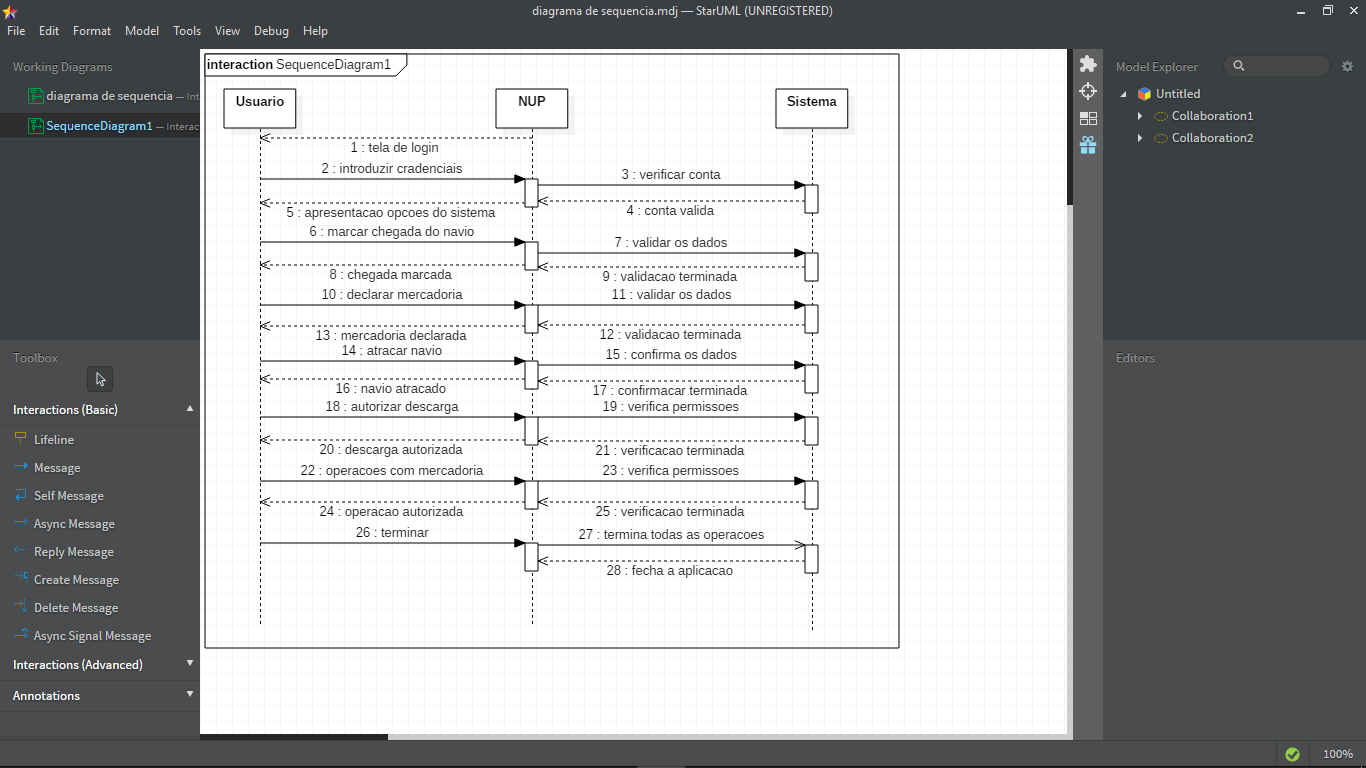


Figura 17 - Diagrama de Sequencia

* + 1. Diagrama de Estados

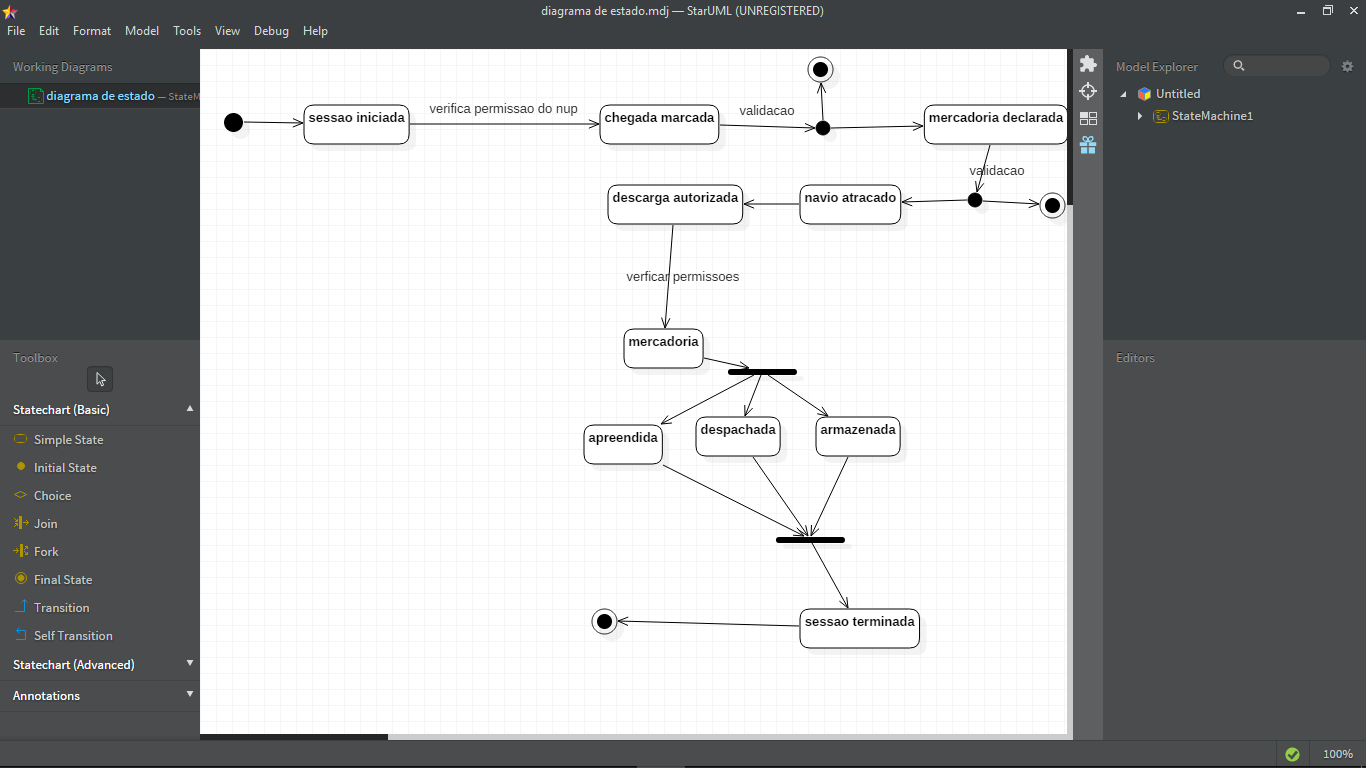


Figura 18 - Diagrama de Estados

* + 1. Diagrama de Classes

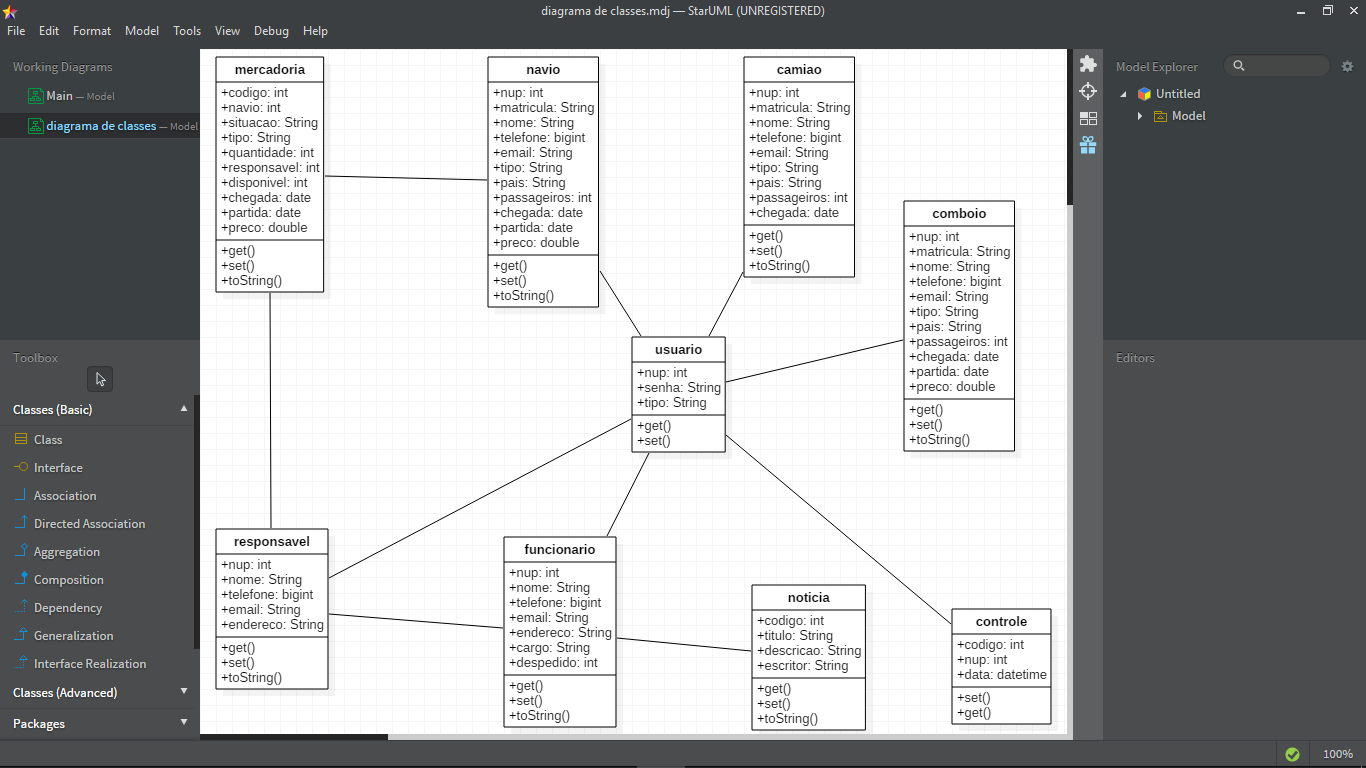


Figura 19 - Diagrama de Classes

* + 1. Diagrama de atividades

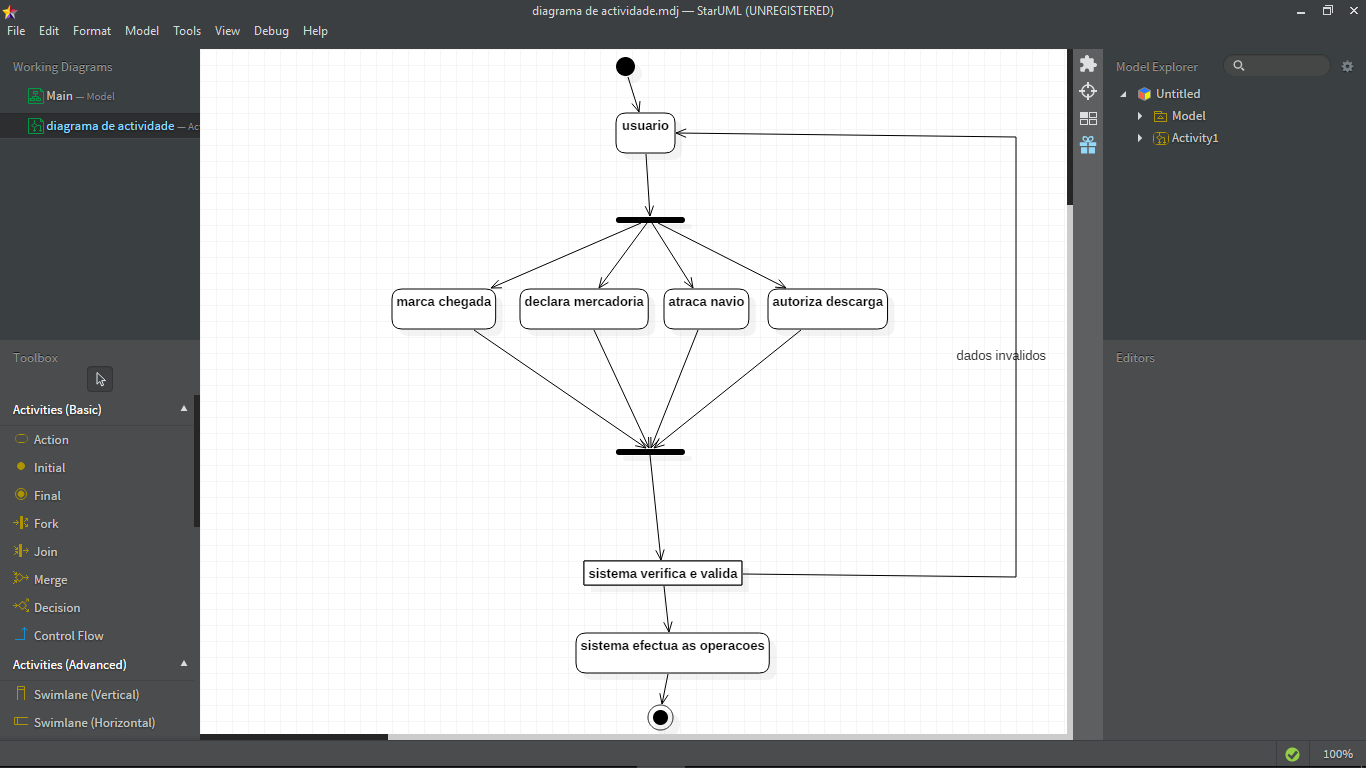


Figura 20 - Diagrama de atividades

1. Conclusão e Recomendações
   1. Conclusão

Para a criação de um sistema, inicialmente deve-se fazer uma análise detalhada no que concerne no funcionamento da instituição, afim de se identificar o problema para que seja possível propor uma solução e definir quais as atividades que o sistema deverá realizar. Para analisar e desenhar o sistema, foram feitas entrevistas aos trabalhadores a fim de se conhecer o funcionamento da instituição e consequentemente, definir as actividades que seriam implementadas no sistema.

Acredita-se que com a implementação deste sistema, a gestão do porto de Maputo melhore significativamente em relação ao anterior cenário vivido.

* 1. Recomendações

Espera-se que com a implementação do Sistema de Gestão Portuário no, mude para o melhor e aumente a gestão das instituições que o usarem.

Para a utilização do sistema, recomenda-se que o mesmo seja instalado em computador que tenha os seguintes requisitos:

* + Sistema operativo - Microsoft Windows 7 em diante;
  + Processador Pentium 4 em diante;
  + 2GB de Memória Ram no mínimo;
  + Java versão 7 em diante;
  + Sistema de Gestão de Base de Dados SQL ou MySQL;
  + Conexão a internet, com taxa de transferência no mínimo de 1 Mbit/s.

1. Referências Bibliográficas

ELMASRI, Rames; NAVATHE, Shamkant B. **Sistema de Banco de Dados**; Tradução Daniel Vieira; 6ªed. – SP: Pearson Addison Wesley, 2011.

MENDES, António J.; MARCELINO Maria J.  **Fundamentos de Programação em Java**, Lisboa: FCA Editora, 2002.

LOPES, Maria J. M., **Sistemas de Informação para Gestão**, Lisboa: Universidade Aberta, 1997.

FOWLER, Martin; SCOTT Kendall, **UML Essencial**; Tradução de Vera Pezerico e Christian Thomas Price, 2ªed, Porto Alegre: Bookman, 2000.

RODRIGUES, Luís A. D. S., **Arquitetura dos Sistemas de Informação**, Lisboa: FCA Editora, 2002.

LEE, Richard C.; TEPFENHART, William M. **UML e C++:** Guia Prático de Desenvolvimento Orientado a Objeto. Tradução de Celso Roberto Paschoa: Makron Books, 2002.