

UNIVERSIDADE DO MINHO

DEPARTAMENTO DE INFORMÁTICA

Desenvolvimento de Sistemas de Software

Fase 3
Grupo Nº 17

Ariana Lousada (A87998) Carlos Gomes (A77185)
Márcia Teixeira (A80943) Tiago Sousa (A67674)

22 de maio de 2021



Ariana Lousada
(A87998)



Carlos Gomes
(A77815)



Márcia Teixeira
(A80943)



Tiago Sousa
(A67674)

Conteúdo

1	Introdução	3
2	Base de Dados	4
3	Estrutura do projeto	5
3.1	Controller	5
3.2	Business Layer	6
3.2.1	Localizacao	6
3.2.2	Palete	6
3.2.3	CodigoQR	6
3.2.4	OrdemTransporte	7
3.2.5	Robot	7
3.2.6	Utilizador	7
3.2.7	IArmazemLN	8
3.3	Data Access	9
4	Conclusão	10
A	Diagramas de Atividade	11
A.1	Use Case - Comunicar código QR	11
A.2	Use Case - Sistema comunica ordem de transporte	12
A.3	Use Case - Notificar recolha de paletes	13
A.4	Use Case - Notificar entrega de paletes	14
A.5	Use Case - Consultar listagem de localizações	15
B	Diagrama de Máquina de Estado	16
B.1	Menu Principal	17
B.2	Menu de Login (Robot/Encarregado/Gestor)	17
B.3	Menu Gestor	18
B.3.1	Registar Ordens de Transporte	18
B.3.2	Registar Encarregados	19
B.3.3	Registar Robots	20
B.3.4	Consultar Paletes	20
B.3.5	Consultar Ordens de Transporte	21
B.3.6	Consultar Robots	21
B.4	Menu Encarregado	22
B.5	Menu Robot	23

Capítulo 1

Introdução

No âmbito da Unidade Curricular de Desenvolvimento de Sistemas de Software, foi-nos proposta a realização de um trabalho prático que visa a criação de um sistema de gestão de *stocks* de um armazém de uma fábrica.

Com este relatório temos o objectivo de apresentar o desenvolvimento do projeto que nos foi proposto. Nesta terceira e última fase, vamos aplicar o que se modelou nas fases anteriores e aplicar as ideias anteriormente desenvolvidas num programa em Java com auxílio a uma base de dados. Para conseguirmos visualizar melhor a estrutura da aplicação, decidimos também fazer um diagrama de Máquina de Estado, assim como vários diagramas de Atividade para cada Use Case estudado.¹

Uma vez que foram propostos vários Use Cases pela equipa docente, é com estes que vamos trabalhar ao longo desta fase do projeto.

¹Para analisar os diagramas, respetivos menus e funcionalidades da aplicação por favor consultar Apêndices A e B

Capítulo 2

Base de Dados

De modo a criar um cenário mais próximo do real, decidimos desenvolver uma base de dados para o nosso sistema, na qual se vai armazenar toda a informação. Esta base de dados mostrou-se extremamente útil, uma vez que nos permitiu guardar e consultar a nossa informação de uma maneira muito mais organizada e sistemática. Como já foi visto nas fases anteriores deste projeto, pretende-se gerir o inventário de um armazém. Este armazém vai ser constituído por secções, prateleiras e robots que vão organizando o armazém de acordo com as ordens de transporte criadas. Para o início do projeto, decidimos colocar na nossa base de dados o suficiente para testar se o programa conseguia executar todas as funcionalidades necessárias.

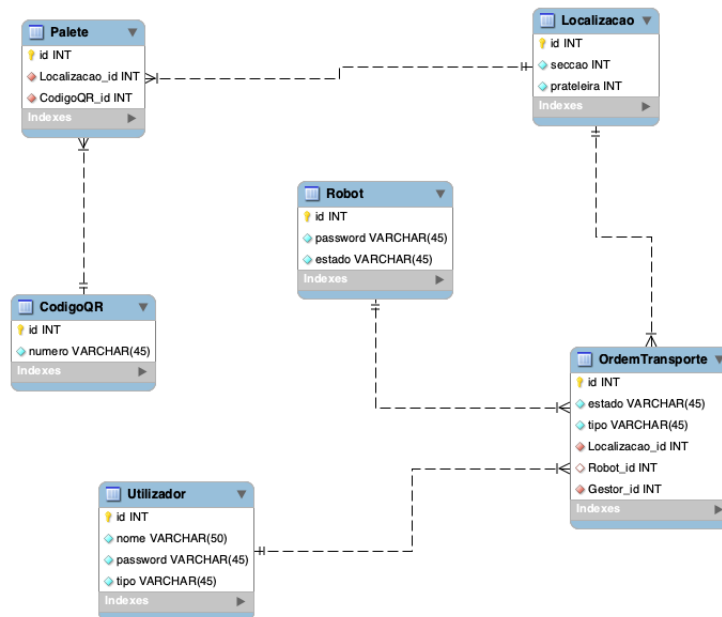


Figura 2.1: Esquema lógico da base de dados do projeto

Capítulo 3

Estrutura do projeto

Nesta última fase do projeto temos como objetivo desenvolver um programa devidamente organizado em *Controller*, *Business Layer* e *Data Access*.

Nesta secção deste documento vamos passar a explicar a fundo cada um destes módulos.

3.1 Controller

O *Controller* vai ser responsável pela interação entre a *Business Layer* e a *Data Access*, funcionando como uma "ponte de ligação" entre os dois módulos.

A sua principal função é redirecionar o utilizador à medida que este vai escolhendo as opções que pretende. Vejamos um exemplo: se um utilizador Gestor pretender consultar a lista de Ordens de Transporte do sistema, tem de existir alguma entidade que seja capaz de redirecionar do menu da opção para a lista de Ordens de Transporte já anteriormente registadas. O *Controller* desempenha essa função, para qualquer utilizador do sistema como para qualquer menu ou tipo de dados que sejam necessários de apresentar.

3.2 Business Layer

Este módulo vai ser o responsável pela manipulação e "criação" de informação. É responsável por criar listas para serem posteriormente apresentadas ao utilizador se necessário, como por exemplo listas de Robots no sistema, Ordens de Transporte, Localizações do armazém, entre outras.

Foi também neste módulo que criámos todas as classes de objeto necessárias. Vamos passar a explicar cada uma delas, nomeadamente a sua estrutura, função e utilidade no sistema.

3.2.1 Localizacao

Cada classe representa uma localização real no armazém. Cada localização vai ter um identificador único, assim como uma secção e prateleira. Em cada localização apenas poderá ser armazenada uma paleta.

3.2.2 Palete

Cada classe deste tipo representa uma paleta. Cada paleta vai ter uma Localização, um identificador próprio e um Código QR. Ao entrar no armazém, o sistema cria um código QR e atribui esse código à paleta em causa, de modo a evitar códigos QR iguais no sistema.

3.2.3 CodigoQR

Cada código QR vai ter um número (identificador). É utilizado para descrever as paletes, uma vez que cada uma vai ter um código QR único.

3.2.4 OrdemTransporte

Esta classe representa uma Ordem de Transporte. Cada ordem de transporte vai ter uma localização, um identificador, estado, tipo e robot (número identificador do robot que a executou), assim como o identificador do Gestor que a criou.

Neste sistema, é o Gestor o responsável por criar as ordens de transporte, identificando logo no início de que tipo se trata (se é uma recolha ou um armazenamento). De seguida, o sistema escolhe um robot que esteja disponível no sistema para este guardar ou retirar a paleta pretendida na Localização escolhida. O estado indica se a Ordem de Transporte foi aceite ("Aceite"), rejeitada ("Rejeitado") ou se está pendente ("Pendente"), isto se ainda não foi enviada a nenhum Robot. Todas as ordens de Transporte criadas podem ser consultadas numa lista pelo Gestor e pelo Encarregado.

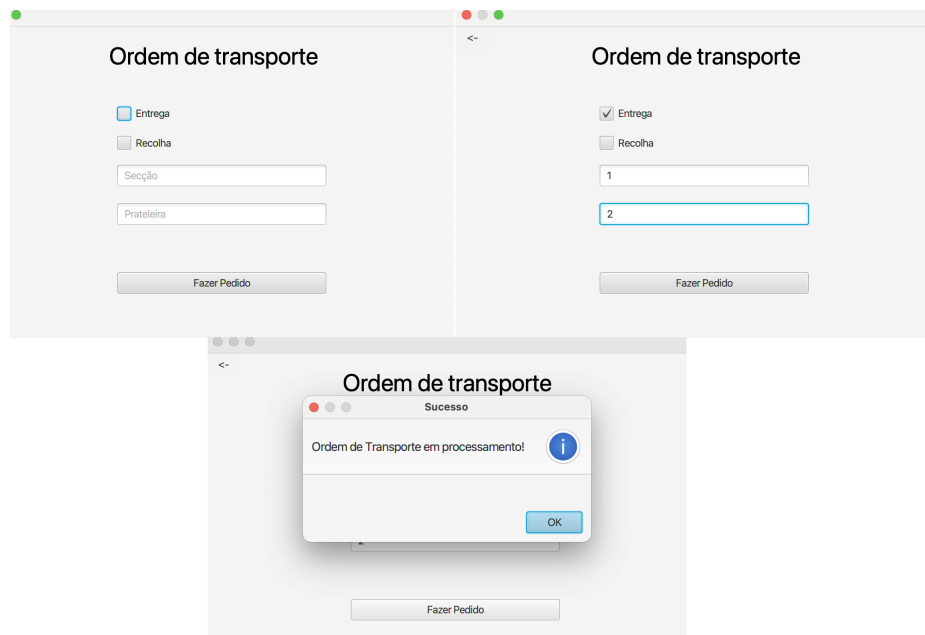


Figura 3.1: Processamento de Ordens de Transporte

3.2.5 Robot

Esta classe tem como intuito representar um robot. Cada robot vai possuir credenciais (que consistem no seu identificador e numa password, de modo a ter acesso ao sistema), um estado (livre ou ocupado) e um código identificador, para facilitar a distribuição das Ordens de Transporte. Cada robot é capaz de aceitar ou recusar uma Ordem de Transporte que lhe foi enviada.

3.2.6 Utilizador

O utilizador pode ser um Gestor ou um Encarregado. O Gestor pode criar Ordens de Transporte, Encarregados, Robots e consultar informações tais como o inventário, robots presentes no sistema e ordens de transporte do armazém. Enquanto isto, o Encarregado só tem permissão

para consultar informações, sendo incapaz de criar ordens de transporte ou qualquer outro tipo de objeto.

3.2.7 IArmazenLN

Esta classe é uma das mais importantes do sistema, uma vez que é a que interage com as restantes classes anteriormente mencionadas. Tudo que envolva atualizar o inventário do armazém vai ser feito por esta classe, gerindo os envios das ordens de transporte assim como a validação das credenciais de todos os utilizadores. Também é responsável pelo acesso ao inventário, isto é, à lista de todas as localizações do armazém.

3.3 Data Access

Este módulo vai ser responsável por toda a informação assim como o armazenamento desta, interagindo com a base de dados sempre que necessário.

Começou-se por inserir a classe *ClassInterface*, que é responsável pela interação com o utilizador; é a partir desta que se vai disponibilizar todas as opções possíveis para que o utilizador possa interagir com o sistema e seleccionar a opção que pretende, obtendo de seguida os resultados pedidos no ecran.

Para além disto, também vimos necessário criar uma classe que fosse capaz de criar uma ligação para a nossa base de dados. A classe *DBConnection* tem precisamente essa função.

As restantes classes presentes neste módulo são necessárias para "converter" os nossos dados em Java para SQL, para depois serem colocados devidamente na base de dados (a partir de uma ligação criada através da classe anteriormente referida - *DBConnection*).

Capítulo 4

Conclusão

No decorrer do desenvolvimento das três fases deste projeto, tornou-se evidente a necessidade de criação de um plano e consequente elaboração de um modelo antes de iniciar a implementação de um sistema que responda aos requisitos do projeto. Em prol de uma melhor organização da componente de programação adjacente a este projeto, foi crucial o planejamento da mesma através de esquemas (representados por todos os diagramas, exploração dos use cases, e identificação das principais classes e packages). O desenvolvimento de diagramas permitiu um desenvolvimento mais acelerado do processo de criação da aplicação. No entanto, e dada a necessidade de adaptação dos vários diagramas desenvolvidos para melhor descrever e preparar o caminho para uma aplicação mais coerente, este exercício apresentou-se como sendo o mais trabalhoso desta última fase, exigindo portanto, um maior espírito crítico e capacidade de gestão de tempo e recursos.

Consideramos, no entanto, que os principais pontos do projeto como um todo foram atingidos, tendo culminado no desenvolvimento de uma aplicação útil, e com capacidade expansiva, tanto por parte do programador, como do utilizador.

Apêndice A

Diagramas de Atividade

A.1 Use Case - Comunicar código QR

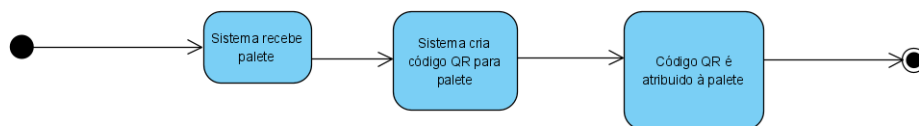


Figura A.1

A.2 Use Case - Sistema comunica ordem de transporte

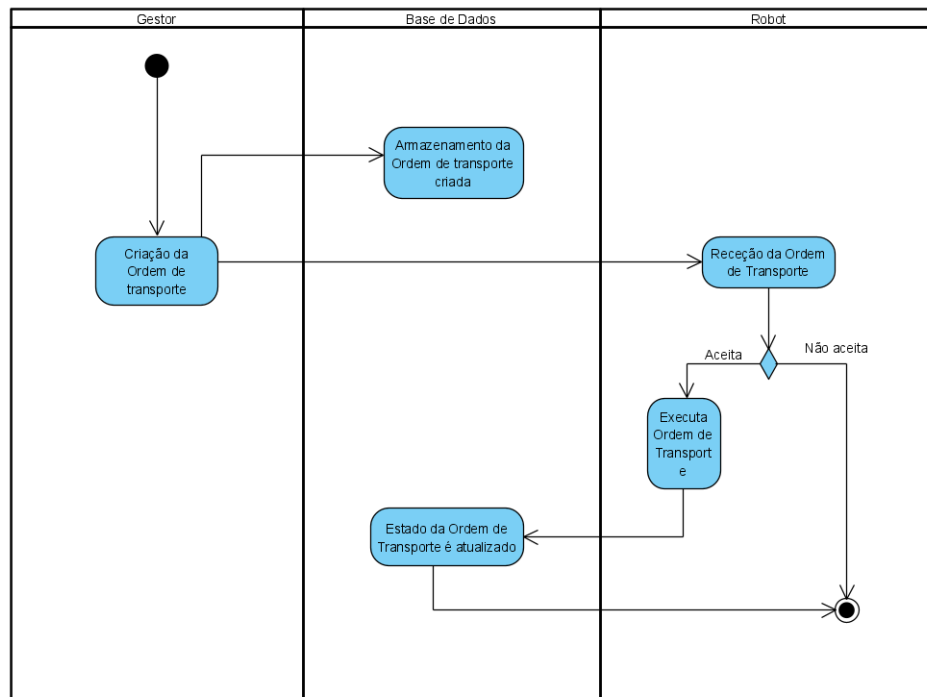


Figura A.2

A.3 Use Case - Notificar recolha de paletes

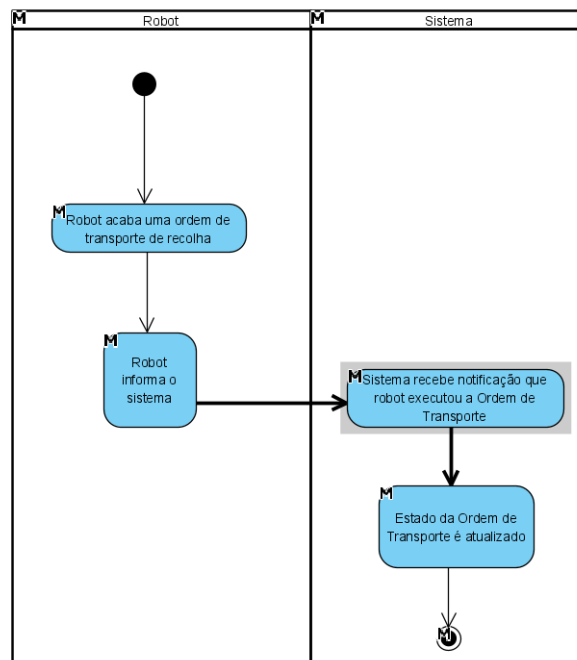


Figura A.3

A.4 Use Case - Notificar entrega de paletes

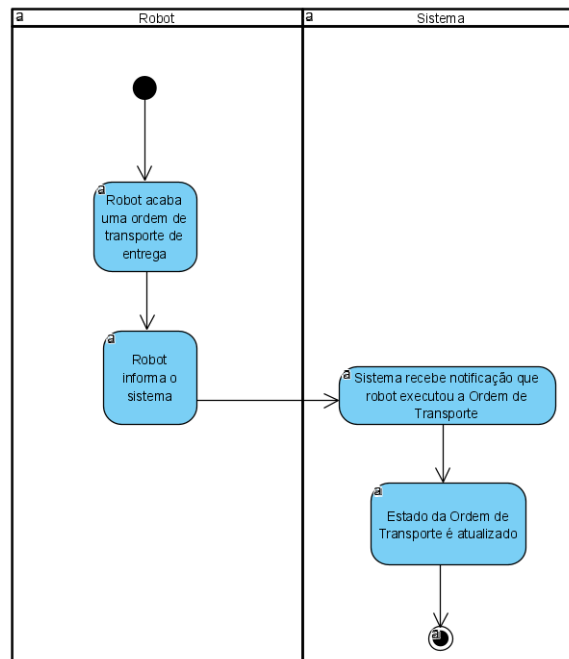


Figura A.4

A.5 Use Case - Consultar listagem de localizações

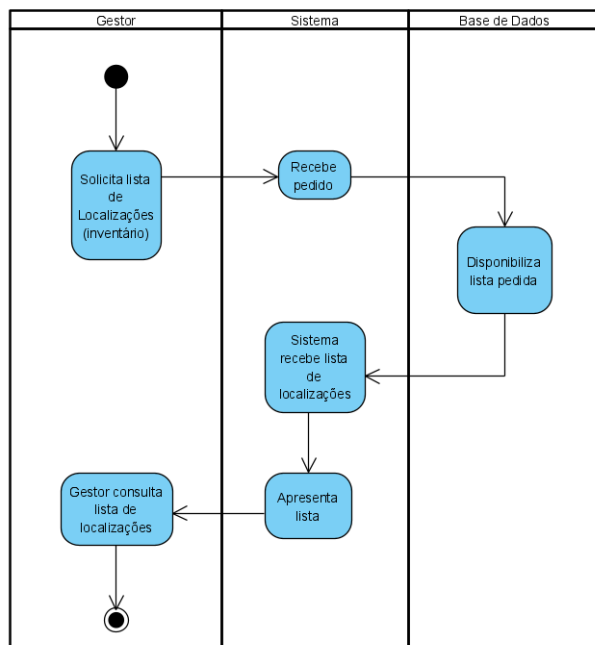


Figura A.5

Apêndice B

Diagrama de Máquina de Estado

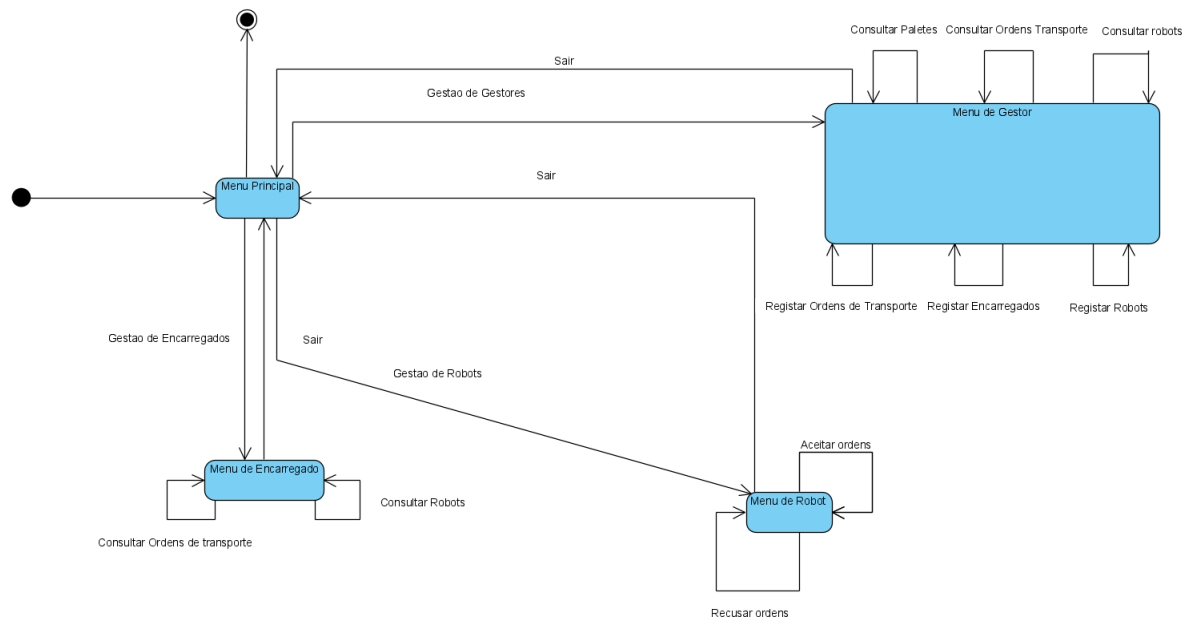


Figura B.1: Diagrama de máquina de estado

B.1 Menu Principal



Figura B.2

B.2 Menu de Login (Robot/Encarregado/Gestor)

Os menus de Login do Robot, Enarregado e Gestor vão ser semelhantes. Se as credenciais inseridas forem incorretas, a aplicação informa o utilizador e solicita as credenciais novamente.

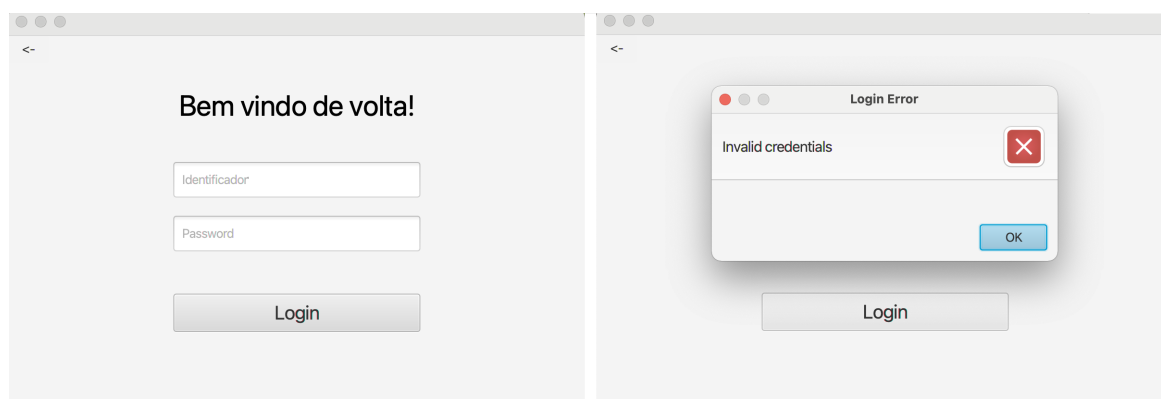


Figura B.3

B.3 Menu Gestor

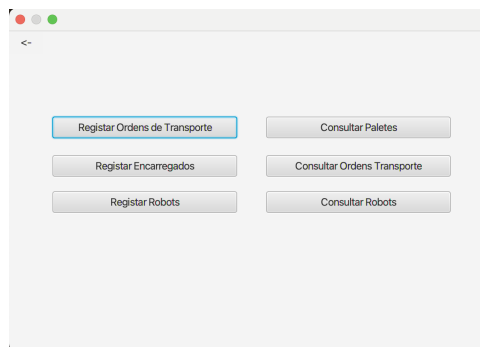


Figura B.4

B.3.1 Registrar Ordens de Transporte

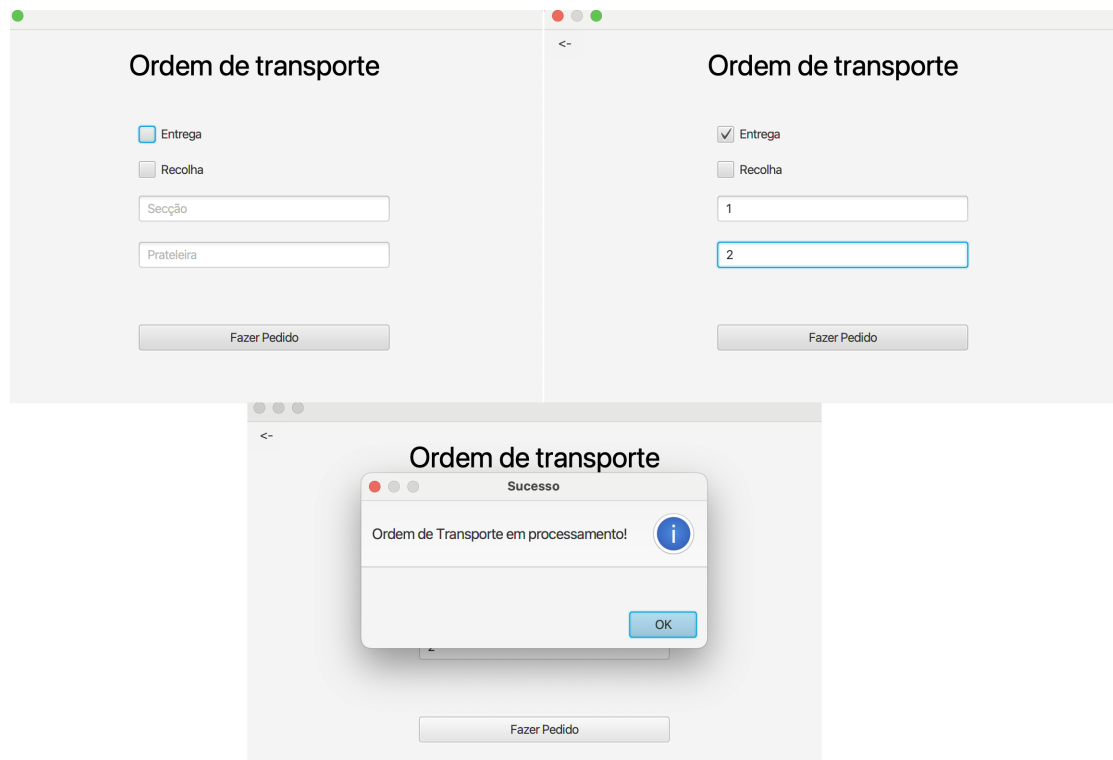


Figura B.5

B.3.2 Registrar Encarregados

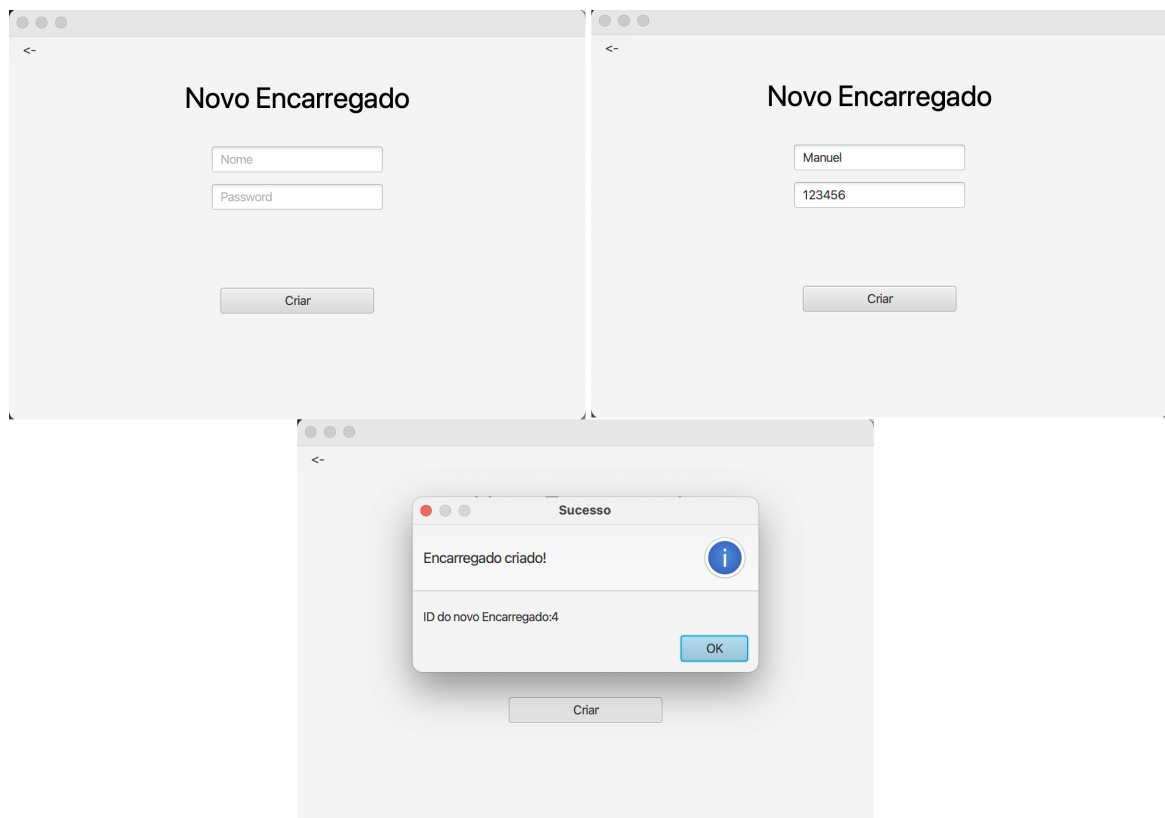


Figura B.6

B.3.3 Registrar Robots

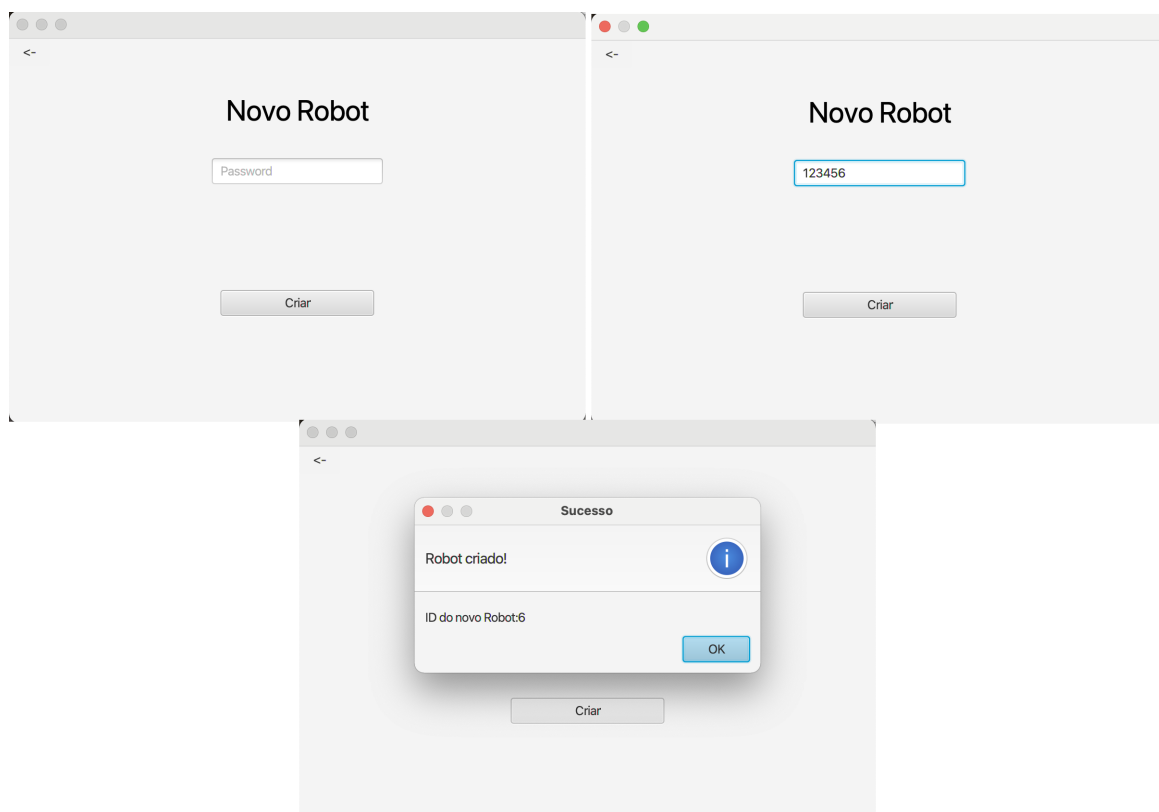


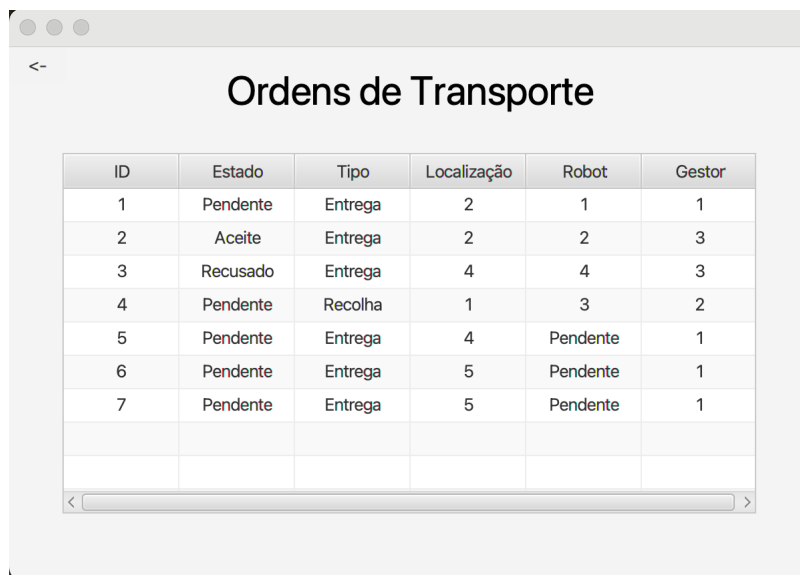
Figura B.7

B.3.4 Consultar Paletes

ID	Secção	Prateleira	
0	0	0	
1	0	1	
2	0	2	
3	0	3	
4	1	1	
5	1	2	
6	1	3	

Figura B.8

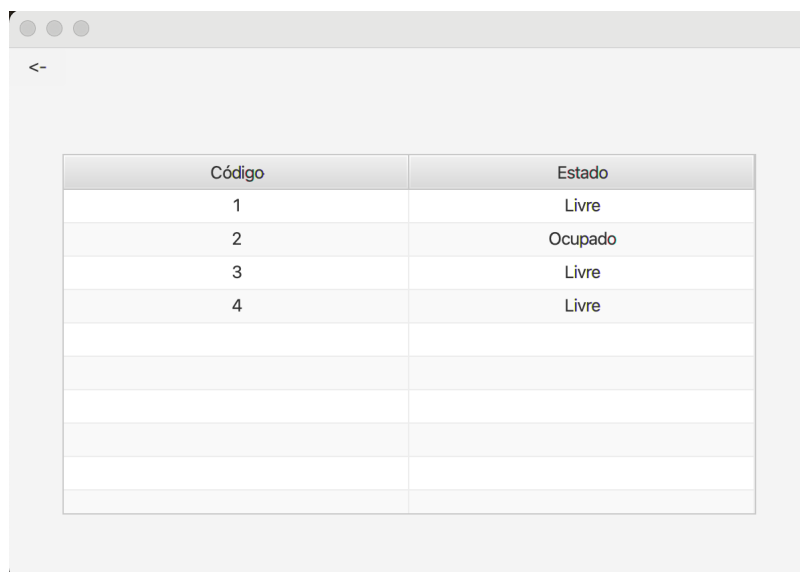
B.3.5 Consultar Ordens de Transporte



ID	Estado	Tipo	Localização	Robot	Gestor
1	Pendente	Entrega	2	1	1
2	Aceite	Entrega	2	2	3
3	Recusado	Entrega	4	4	3
4	Pendente	Recolha	1	3	2
5	Pendente	Entrega	4	Pendente	1
6	Pendente	Entrega	5	Pendente	1
7	Pendente	Entrega	5	Pendente	1

Figura B.9

B.3.6 Consultar Robots



Código	Estado
1	Livre
2	Ocupado
3	Livre
4	Livre

Figura B.10

B.4 Menu Encarregado

Estas duas opções disponíveis vão corresponder às opções do Gestor B.3.5 e B.3.6 respetivamente.

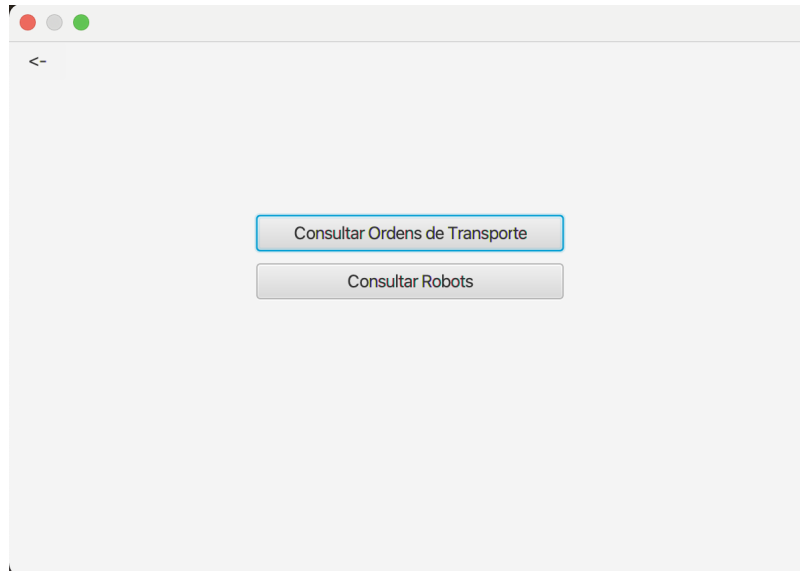


Figura B.11

B.5 Menu Robot

Este menu apenas vai ter a opção de rejeitar ou recusar uma Ordem de Transporte

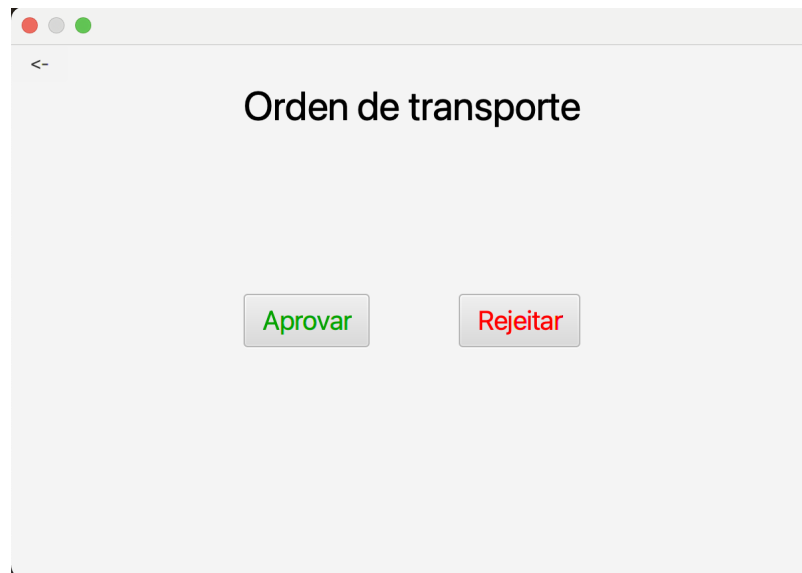


Figura B.12