



# Projecto Final de Curso

## Engenharia Informática

**Trabalho realizado por:**  
Tiago Miguel Timóteo da Silva de Jesus  
Nº 2400847  
Engenharia Informática

## **Índice**

### **1. Introdução**

### **2. Requisitos**

#### **2.1. Requisitos Funcionais**

#### **2.2. Requisitos não Funcionais**

#### **2.3. Requisitos Técnicos**

### **3. Implementação**

#### **3.1. Ferramentas Utilizadas**

##### **3.1.1. Base de Dados**

##### **3.1.2. Adobe Dreamweaver CS4**

##### **3.1.3. Adobe Fireworks CS4**

##### **3.1.4. Microsoft Office Visio 2003**

##### **3.1.5. Linguagem Php**

##### **3.1.6. Linguagem Sql**

##### **3.1.7. Linguagem Javascript**

##### **3.1.8. WampServer**

### **4. Modelação do Sistema**

#### **4.1. Base de Dados**

#### **4.2. Diagramas Comportamentais**

##### **4.2.1. Diagrama de Casos de Uso**

##### **4.2.2. Diagrama de Actividades dos Leilões**

##### **4.2.3. Diagrama de Transição**

### **5. Funcionamento do Sistema**

#### **5.1. Interface Administrador**

#### **5.2. Interface Utilizador**

### **6. Conclusão e Trabalhos Futuros**

### **7. Bibliografia**

## **Resumo**

Este documento é o relatório final da cadeira de TFC (Trabalho Final de Curso) e nele está o desenvolvimento de um site de leilões online que permitirá aos seus utilizadores a compra e venda de produtos.

O documento é constituído por diversos pontos desde o estudo de mercado até ao seu desenvolvimento e à obtenção do produto final.

O trabalho irá começar com uma pequena introdução sobre o impacto que os leilões virtuais existentes têm no nosso dia-a-dia e a oportunidade de negócio que pode proporcionar a quem deseja começar um negócio a partir de casa.

De seguida, irá concentrar-se no desenvolvimento da aplicação desde a sua modelação até à realização do projecto em si.

## **Abstract**

This document is the final report from the discipline of TFC and in him is the development of a online auction site which will allow to all users to buy and sell all kind of products.

The document comprise various points since the market study till is development and the obtaining of the final product.

The paper will start with a small introduction about the impact tahat the virtual auctions have on our days and the business opportunity that it can provide to people who want to start a business from home.

Then, it will focus on the development of the application since is modelling to the project realization.

## **1. Introdução**

O objectivo deste trabalho é a modelação e o desenvolvimento de um sistema de leilões online que irá permitir a venda e a compra de todo o tipo de produtos colocados pelos utilizadores.

Entre as formas de comunicação por computador, o comércio electrónico tem representado uma grande percentagem na sua evolução. Este permite a compra e venda de mercadorias e serviços por meio da Internet, onde as chamadas Lojas Virtuais oferecem produtos e formas de pagamento online. O comércio electrónico é um meio facilitador dos negócios, tornando o processo de venda fácil, seguro e rápido, reduzindo os custos das empresas que actuam neste tipo de mercado estimulando a competitividade. Devido a este facto ter um negócio em casa é algo que vai ganhando adeptos em Portugal. Para além da poupança de custos, gerir em casa permite conciliar melhor a vida pessoal e a profissional.

## 2. Requisitos

### 2.1. Requisitos Funcionais

- Fornecimento de uma página de registo para utilizadores e outra que permite o login de utilizadores;
- Possibilidade dos utilizadores fazerem a compra e venda de produtos;
- O site permite a pesquisa de produtos segundo as categorias existentes na base de dados e ainda uma pesquisa avançada que permite refinar essa pesquisa;
- Página pessoal para os utilizadores onde podem editar dados pessoais, ver comentários e mensagens;
- Página de manutenção para leilões.

### 2.2. Requisitos não Funcionais

- Existência de uma base de dados;
- Aplicação Web para os administradores;
- Aplicação Web para acesso dos utilizadores.

### 2.3. Requisitos Técnicos

- **IDE** → Adobe Dreamweaver CS4
- **Base de Dados** → Sybase Power Designer
- **Diagramas Comportamentais** → Microsoft Office Visio 2003
- **Linguagem** → Php, MySql, Javascript
- **Grafismo** → Adobe Fireworks CS4
- **Servidor** → WampServer versão 2.0

### **3. Implementação**

#### **3.1. Ferramentas Utilizadas**

##### **3.1.1. Base de Dados**

O termo “base de dados” faz hoje parte do vocabulário corrente. Contudo, essa vulgarização não significa que o conceito técnico tenha sido inteiramente apreendido.

A noção corrente está associada à existência de um volume apreciável de dados relativos a um determinado assunto, supostamente organizados, e acessíveis através de computadores. Muitas vezes, identifica-se base de dados com o “programa” que permite o acesso aos dados. Por vezes identifica-se base de dados com uma parte do todo. Assim, não é de todo improvável ouvir dizer-se que se consultaram as “bases de dados da Internet”, que lidamos com “bases de dados Access” ou que “a base de dados de clientes” é muito completa.

Devemos assim reter que uma Base de Dados é um modelo estruturado de dados com operadores associados, permitindo uma visão unificada desses mesmos dados. Assim, há que distinguir entre Base de Dados (BD) e Sistema de Gestão de Base de Dados (SGBD). A BD em si, como modelo, obedece fundamentalmente a critérios de estruturação puramente conceptuais podendo ser descrita através de símbolos, textuais ou gráficos, normalizados. O SGBD, por seu lado, permite a implementação concreta em computador daquele modelo, impondo uma estrutura física particular, normalmente dele dependente.

##### **3.1.2. Adobe Dreamweaver CS4**

O Macromedia Dreamweaver é uma ferramenta de desenvolvimento para a web criada pela Macromedia (agora Adobe Systems). Versões iniciais da aplicação serviam como um simples editor HTML, porém versões mais recentes incorporaram notável suporte para muitas outras tecnologias web tais como XHTML, CSS, JavaScript e vários scripts de servidor. Desde o final dos anos 90, o Dreamweaver vem tendo um sucesso crescente e hoje domina cerca de 80% do mercado de editores HTML. Existem versões tanto para Mac OS quanto para Windows, mas também é possível executá-lo

em plataformas Unix. Como um editor, o Dreamweaver pode esconder os detalhes do código HTML do utilizador, tornando possível que não especialistas criem facilmente páginas web. Alguns desenvolvedores web criticam esta abordagem por produzirem páginas HTML muito maiores do que o necessário, o que pode levar a um desempenho pobre dos navegadores web. Isto pode ser particularmente verdade porque a aplicação torna muito fácil a criação de layouts baseados em tabelas. Além disso, alguns desenvolvedores de sites web, criticaram o Dreamweaver no passado, pela introdução de código, que frequentemente não era compatível com os padrões W3C. No entanto, a Macromedia aumentou o suporte para CSS e layout de página sem tabelas em versões posteriores. O Dreamweaver permite seleccionar a maioria dos navegadores para se ter uma previsão dos sites web. O software possui também ótimas ferramentas de manutenção de sites tais como a habilidade de encontrar e substituir, no site inteiro, linhas de texto ou código através de parâmetros especificados. O painel de comportamentos também permite a criação de JavaScript básico, sem qualquer conhecimento de codificação. Com o advento da versão MX, a Macromedia incorporou ferramentas de criação de conteúdo dinâmico ao Dreamweaver. No espírito das ferramentas HTML, permite que utilizadores se conectem a base de dados (tais como MySQL e Microsoft Access) para filtrar e mostrar conteúdo usando tecnologias de script tais como PHP, ColdFusion, ASP e ASP.NET, sem qualquer experiência prévia em programação.

### **3.1.3. Adobe Fireworks CS4**

O Adobe Fireworks é um poderoso editor de imagens e aplicativos gráficos para Web. Ele é um software do Adobe Photoshop, porém completamente independente dele, ou seja, sua instalação não depende do Photoshop. Como é notório entre os web designers esse programa é simplesmente essencial para a criação de páginas web mais interactivas e graficamente melhor talhadas. Com várias funcionalidades e o já tradicional alto desempenho dos produtos da Adobe.

Para este trabalho foi unicamente utilizado para fazer as imagens que aparecem pelo site.

### **3.1.4. Microsoft Office Visio 2003**

O Microsoft Visio é um aplicativo para criação de diagramas para o ambiente Windows. Os pontos fortes do Visio são os diagramas técnicos e profissionais, com imagens vectoriais, que podem ser ampliados e manipulados com facilidade. O Visio pode ser utilizado para gerar diagramas de diversos tipos, como organogramas, fluxogramas, modelagem de dados (usando UML ou outra notação gráfica qualquer), diagramas de redes, plantas baixas, cartazes, etc.

O Visio pode ser considerado um intermediário entre um programa de CAD (especializado em desenho vectorial de alta precisão para engenharia) e um programa de desenho artístico vectorial, como o Adobe Illustrator. Na comparação com essas ferramentas mais especializadas, as funcionalidades do Visio são bem inferiores; não é tão preciso quando o AutoCAD, e não oferece recursos artísticos como o Illustrator. Porém, a flexibilidade, a simplicidade de uso e qualidade das ferramentas o tornam uma ferramenta excelente para aplicações rápidas, rascunhos, e diagramas de uso pessoal ou profissional.

Para este trabalho foi utilizado para fazer os diagramas de casos de uso, diagrama de actividades e os diagramas de transição de estado.

### **3.1.5. Linguagem Php**

A linguagem php é já a mais utilizada a nível mundial no que respeita a programação de soluções web dinâmicas. No panorama do mercado global do software livre, a comunidade PHP Portugal tem vindo a registar um crescimento gradual e consistente. O reconhecimento cada vez maior que as empresas têm conferido à tecnologia open source, devido principalmente às suas inúmeras vantagens em relação à concorrência, tem permitido à esfera PHP Portugal atingir novos horizontes no que diz respeito aos seus utilizadores e à sua notoriedade. Ao crescimento da comunidade PHP Portugal é indissociável a grande capacidade dos consultores profissionais das empresas associadas a esta plataforma.



### 3.1.6. Linguagem Sql

Um SGDB requer uma linguagem de consulta que permita aos utilizadores o acesso aos dados. A linguagem SQL (Structured Query Language) foi desenvolvida pela IBM em meados dos anos 70. Em 1979 a Oracle apresentou uma implementação do SQL comercial. As principais características desta linguagem são:

- Sintaxe dos comandos o mais próximo possível da língua natural inglesa;
- Não procedimental: indica-se a informação que se pretende obter sem qualquer preocupação em "como se vai obter". O utilizador não se preocupa com o método de acesso aos dados que fica a cargo do SGBD;
- Trabalha com conjuntos de registos e não com um registo de cada vez. Não existem comandos como "Next record" ou "Previous record";
- É utilizada tanto pelos utilizadores normais como pelo DBA (Database Administrator).

Os seus comandos permitem:

- Consulta aos dados;
- Inserir, actualizar e apagar linhas numa tabela;
- Criar, modificar e apagar objectos de uma base de dados (tabelas, índices, vistas, restrições, sequências, etc);
- Controlar e auditar o acesso aos objectos armazenados na base de dados;
- Garantir a consistência dos dados.

A linguagem SQL tornou-se o standard da indústria para bases de dados relacionais. Em 1986 o American National Standards Institut (ANSI) definiu a sua sintaxe e adoptou-a como linguagem standard para SGBDR's. A "International Standard Organization" (ISO) também a adoptou.

### 3.1.7. Linguagem Javascript

Javascript é uma linguagem de script incorporada a um documento HTML. Historicamente, trata-se da primeira linguagem de scripts para a web. Esta linguagem é uma linguagem de programação que traz melhorias para a linguagem HTML, permitindo a execução de comandos do cliente, ou seja, em termos do navegador e não do servidor web.

Assim, a linguagem Javascript é altamente dependente do navegador que chama a página web onde o script está incorporado, mas, por outro lado, não requer nenhum compilador, ao contrário da linguagem Java, com a qual ela tem sido confundida, há muito tempo.

O JavaScript foi desenvolvido pela Netscape em 1995. Originalmente, era chamado LiveScript e fornecia uma linguagem de script simples para o navegador Netscape Navigator 2. Este foi criticado durante muito tempo pela sua falta de segurança, o seu desenvolvimento pouco desenvolvido e pela ausência de alertas de erro explícitos tornando o seu uso difícil. Em 4 de Dezembro de 1995, após uma parceria com o fabricante Sun, o Netscape mudou o nome para Javascript.

### 3.1.8. WampServer

WampServer (conhecido anteriormente como WAMP5) é um software publicado sob a *GNU General Public License* desenvolvido pela PHP Team. É usado para se instalar rapidamente no computador os softwares PHP 5, MySQL e Apache, disponibilizando suporte ao uso de scripts PHP localmente no Windows. Existem vários add-ons para download no site oficial, como por exemplo, o add-on para activar o uso de PHP 4 ao invés do PHP 5.

## 4. Modelação do Sistema

### 4.1. Base de Dados

adminusers	
id	int
username	varchar(32)
password	varchar(32)
hash	varchar(5)
created	varchar(8)
lastlogin	varchar(14)
status	int

bids		
<u>id</u>	<u>int</u>	<u>&lt;pk&gt;</u>
auction	int	
bidder	int	
bid	double	
bidwhen	varchar(14)	
quantity	int	

counters	
users	int
inactiveusers	int
auctions	int
closedauctions	int
bids	int
suspendedauctions	int

auccounter	
<u>auction_id</u>	<u>int &lt;pk&gt;</u>
counter	int

categories		
<u>cat_id</u>	<u>int</u>	<u>&lt;pk&gt;</u>
parent_id	int	
left_id	INT	
right_id	INT	
level	INT	
cat_name	tinytext	
sub_counter	int	
counter	int	
cat_colour	varchar(15)	
cat_image	varchar(150)	

countries	
<u>country</u>	<u>varchar(40) &lt;pk&gt;</u>

auctions		
<u>id</u>	<u>int</u>	<u>&lt;pk&gt;</u>
user	int	
title	varchar(70)	
starts	varchar(14)	
description	text	
pict_url	tinytext	
category	int	
minimum_bid	double	
shipping_cost	double	
reserve_price	double	
buy_now	double	
auction_type	char(1)	
duration	varchar(7)	
increment	double	
shipping	char(1)	
payment	tinytext	
international	char(1)	
ends	varchar(14)	
current_bid	double	
closed	char(2)	
photo_uploaded	tinyint	
quantity	int	
suspended	int	
relist	int	
relisted	int	
num_bids	int	
sold	enum('y','n','s')	
shipping_terms	tinytext	
bn_only	enum('y','n')	
bold	enum('y','n')	
highlighted	enum('y','n')	
featured	enum('y','n')	
current_fee	double	

closedrelisted	
auction	int
relistdate	varchar(8)
newauction	int

currencies	
id	int
currency	varchar(100)

durations	
days	int
description	varchar(30)

comm_messages	
id	int
boardid	int
msgdate	varchar(14)
user	int
username	varchar(255)
message	text

feedbacks		
<u>id</u>	<u>int</u>	<u>&lt;pk&gt;</u>
rated_user_id	int	
rater_user_nick	varchar(20)	
feedback	mediumtext	
rate	int	
feedbackdate	INT	
auction_id	int	

community	
id	int
name	varchar(255)
messages	int
lastmessage	varchar(14)
msgstoshow	int
active	int

fees		
<u>id</u>	INT	<pk>
fee_from	double	
fee_to	double	
fee_type	enum('flat', 'perc')	
value	double	
type	varchar(15)	

filterwords	
word	varchar(255)

fontsandcolors	
err_font	int
err_font_size	int
err_font_color	varchar(7)
err_font_bold	enum('y','n')
err_font_italic	enum('y','n')
std_font	int
std_font_size	int
std_font_color	varchar(7)
std_font_bold	enum('y','n')
std_font_italic	enum('y','n')
sml_font	int
sml_font_size	int
sml_font_color	varchar(7)
sml_font_bold	enum('y','n')
sml_font_italic	enum('y','n')
tlf_font	int
tlf_font_size	int
tlf_font_color	varchar(7)
tlf_font_bold	enum('y','n')
tlf_font_italic	enum('y','n')
nav_font	int
nav_font_size	int
nav_font_color	varchar(7)
nav_font_bold	enum('y','n')
nav_font_italic	enum('y','n')
footer_font	int
footer_font_size	int
footer_font_color	varchar(7)
footer_font_bold	enum('y','n')
footer_font_italic	enum('y','n')
bordercolor	varchar(7)
headercolor	varchar(7)
tableheadercolor	varchar(7)
linkcolor	varchar(7)
vlinkcolor	varchar(7)
highlighteditems	varchar(7)

gateways	
gateways	text
paypal_address	varchar(50)
paypal_required	int
paypal_active	int
alertpay_address	varchar(50)
alertpay_password	varchar(50)
alertpay_required	int
alertpay_active	int

groups		
<u>id</u>	int	<pk>
group_name	varchar(50)	
can_sell	int	
can_buy	int	
count	int	
auto_join	int	

increments	
id	char(3)
low	double
high	double
increment	double

maintainance	
id	int
active	enum('y','n')
superuser	varchar(32)
maintainancetext	text

membertypes		
<u>id</u>	int	<pk>
feedbacks	int	
membertype	varchar(30)	
discount	tinyint	
icon	varchar(30)	

messages		
<u>id</u>	int	<pk>
sentto	int	
sentfrom	int	
fromemail	varchar(50)	
sentat	varchar(20)	
message	text	
isread	int	
subject	varchar(50)	
replied	int	
reply_of	INT	
question	int	
public	INT	

online		
<u>ID</u>	bigint	<pk>
SESSION	varchar(32)	
time	bigint	

payments	
id	int
description	varchar(30)

pendingnotif		
<u>id</u>	int	<pk>
auction_id	int	
seller_id	int	
winners	text	
auction	text	
seller	text	
thisdate	varchar(8)	

proxybid	
itemid	int
userid	int
bid	double

rates	
id	int
ime	tinytext
valuta	tinytext
symbol	char(3)

rememberme	
userid	int
hashkey	char(32)

settings	
sitename	varchar(255)
siteurl	varchar(255)
copyright	varchar(255)
moneyformat	int
moneydecimals	int
moneysymbol	int
currency	varchar(10)
showacceptancetext	int
acceptancetext	longtext
adminmail	varchar(100)
newsletter	int
logo	varchar(255)
timecorrection	int
archiveafter	int
datesformat	enum('USA','EUR')
errortext	text
errormail	varchar(255)
picturegallery	int
maxpictures	int
maxpicturesize	int
buy_now	int
alignment	varchar(15)
thumb_show	smallint
lastitemsnumber	int
higherbidnumber	int
endingsoonnumber	int
boards	enum('y','n')
boardsink	enum('y','n')
wordfilter	enum('y','n')
aboutus	enum('y','n')
aboutustext	text
terms	enum('y','n')
termstext	text
privacypolicy	enum('y','n')
privacypolicytext	text
defaultcountry	varchar(30)
relisting	int
defaultlanguage	char(2)
pagewidth	int
pagewidthtype	enum('perc','fix')
catsorting	enum('alpha','counter')
usersauth	enum('y','n')
background	tinytext
brepeat	enum('repeat','repeat-x','repeat-y','no-repeat','no')
descriptiontag	text
keywordstag	text
maxuploadsize	int
contactseller	enum('always','logged','never')
theme	tinytext
catstoshow	int
uniqueseller	int
bn_only	enum('y','n')
winner_address	enum('y','n')
boardsmsgs	int
activationtype	INT
https	enum('y','n')
bn_only_disable	enum('y','n')
bn_only_percent	int
buyerprivacy	ENUM('y','n')
cust_increment	INT
fees	ENUM('y','n')
fee_type	int
fee_max_debt	double
fee_sgnup_bonus	double
fee_disable_acc	enum('y','n')
ae_status	enum('enabled','disabled')
ae_timebefore	int
ae_extend	int
cache_theme	ENUM('y','n')
hours_countdown	int
edit_starttime	int
counter_auctions	enum('y','n')
counter_users	enum('y','n')
counter_online	enum('y','n')
banemail	text
mandatory_fields	varchar(255)
displayed_fields	VARCHAR(255)
ao_hpf_enabled	enum('y','n')
ao_hi_enabled	enum('y','n')
ao_bi_enabled	enum('y','n')
proxy_bidding	enum('y','n')
recaptcha_public	varchar(40)
recaptcha_private	varchar(40)
spam_sendtofriend	int
spam_register	int

users	
id	int <pk>
nick	varchar(20)
password	varchar(32)
hash	varchar(5)
name	tinytext
address	tinytext
city	varchar(25)
prov	varchar(10)
country	varchar(30)
zip	varchar(10)
phone	varchar(40)
email	varchar(50)
reg_date	int
rate_sum	int
rate_num	int
birthdate	int
suspended	int
nletter	int
balance	double
auc_watch	varchar(20)
item_watch	text
endemailmode	enum('one','cum','none')
startemailmode	enum('yes','no')
emailtype	enum('html','text')
lastlogin	datetime
payment_details	text
groups	text
bn_only	enum('y','n')
timecorrection	int
paypal_email	varchar(50)
alertpay_id	varchar(50)
alertpay_pass	varchar(50)
language	char(2)

userfees	
id	INT <pk>
auc_id	int
user_id	int
amt	double
paid	int

usersps	
id	int <pk>
user	int
ip	varchar(15)
type	enum('first','after')
action	enum('accept','deny')

winners	
id	int
auction	int
seller	int
winner	int
bid	double
closingdate	int
feedback_win	tinyint
feedback_sel	tinyint
qty	int
paid	int
bf_paid	INT

## 4.2. Diagramas Comportamentais

### 4.2.1. Diagrama de Casos de Uso

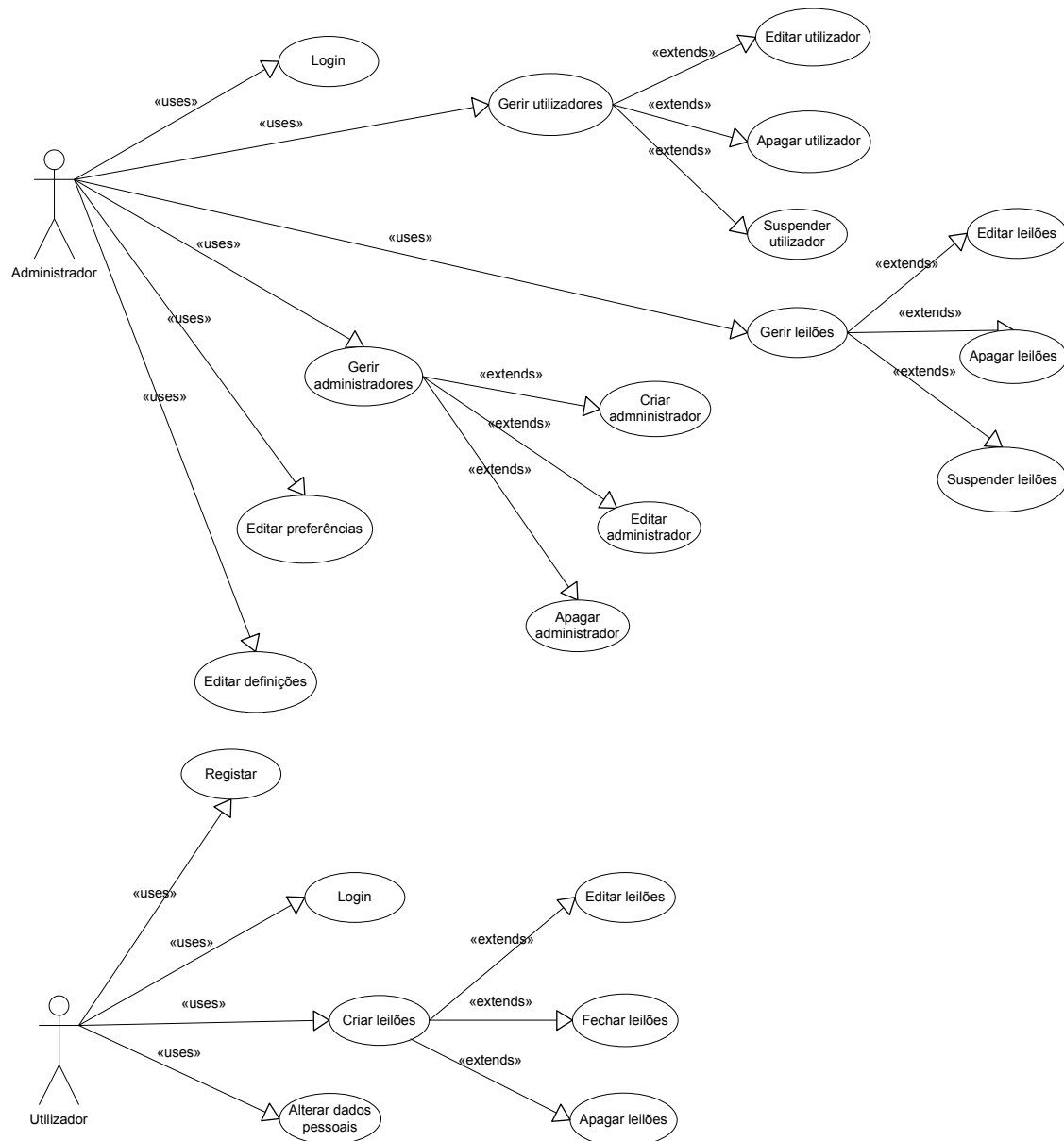
A modelação de um diagrama de casos de uso é uma técnica usada para descrever e definir os requisitos funcionais de um sistema. São escritos em termos de actores externos, casos de uso e o sistema modelado. Os actores representam o papel de uma entidade externa ao sistema como um utilizador, um hardware, ou outro sistema que interage com o sistema modelado. Os actores iniciam a comunicação com o sistema através dos casos de uso, onde o caso de uso representa uma sequência de acções executadas pelo sistema e recebe do actor que lhe utiliza dados de um tipo ou formato já conhecido, e o valor de resposta da execução de um caso de uso também já é de um tipo conhecido, tudo isso é definido juntamente com o caso de uso através de texto de documentação.

Actores e casos de uso são classes. Um actor é ligado a um ou mais casos de uso através de associações, e tanto actores como casos de uso podem possuir relacionamentos de generalização que definem um comportamento comum de herança em super classes especializadas em subclasses.

O uso de casos de uso em colaborações é muito importante, onde estas são a descrição de um contexto mostrando classes e objectos, os seus relacionamentos e sua interacção exemplificando como as classes e objectos interagem para executar uma actividade específica no sistema. Uma colaboração é descrita por diagramas de actividades e um diagrama de colaboração.

Quando um caso de uso é implementado, a responsabilidade de cada passo da execução deve ser associada às classes que participam da colaboração, tipicamente especificando as operações necessárias dentro destas classes juntamente com a definição de como elas irão interagir. Um cenário é uma instância de um caso de uso, ou de uma colaboração, mostrando o caminho específico de cada acção. Por isso, o cenário é um importante exemplo de um caso de uso ou de uma colaboração. Quando visto a nível de um caso de uso, apenas a interacção entre o actor externo e o caso de uso é vista, mas observando a nível de uma colaboração, toda as interacções e passos da execução que implementam o sistema serão descritos e especificados.

## Diagrama de casos de uso



### 4.2.2. Diagrama de Actividades dos Leilões

Diagramas de actividade capturam acções e os seus resultados. Eles incidem no trabalho executado na implementação de uma operação (método), e nas actividades numa instância de um objecto. O diagrama de actividade é uma variação do diagrama de estado e possui um propósito um pouco diferente do diagrama de estado, que é o de

capturar acções (trabalho e actividades que serão executados) e os resultados em termos das mudanças de estados dos objectos.

Os estados no diagrama de actividade mudam para um próximo estágio quando uma acção é executada (sem ser necessário especificar nenhum evento como no diagrama de estado).

Um diagrama de actividade é uma maneira alternativa de se mostrar interacções, com a possibilidade de expressar como as acções são executadas, o que elas fazem (mudanças dos estados dos objectos), quando elas são executadas (sequência das acções), e onde elas acontecem.

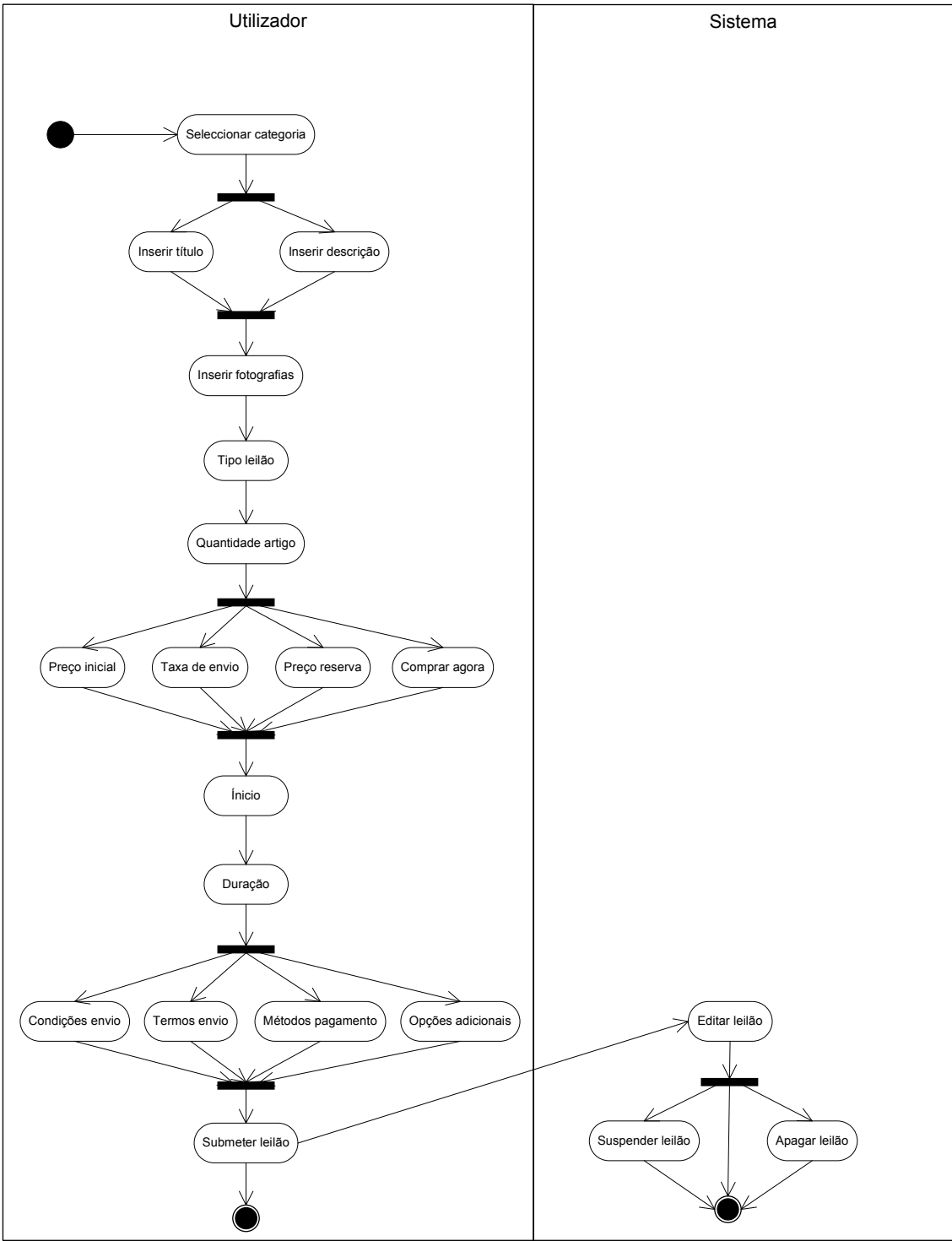
Um diagrama de actividade pode ser usado com diferentes propósitos:

- Para capturar os trabalhos que serão executados quando uma operação é disparada (acções). Este é o uso mais comum para o diagrama de actividade;
- Para capturar o trabalho interno num objecto;
- Para mostrar como um grupo de acções relacionadas podem ser executadas, e como vão afectar os objectos em torno delas;
- Para mostrar como uma instância pode ser executada em termos de acções e objectos;
- Para mostrar como um negócio funciona em termos de trabalhadores (actores), fluxos de trabalho, organização, e objectos (factores físicos e intelectuais usados no negócio).

O diagrama de actividade mostra o fluxo sequencial das actividades, é normalmente utilizado para demonstrar as actividades executadas por uma operação específica do sistema. Consistem em estados de acção, que contém a especificação de uma actividade a ser desempenhada por uma operação do sistema. Decisões e condições, como execução paralela, também podem ser mostrados no diagrama de actividade. O diagrama também pode conter especificações de mensagens enviadas e recebidas como partes de acções executadas.



Diagrama de Actividades dos Leilões



### 4.2.3. Diagrama de Transição

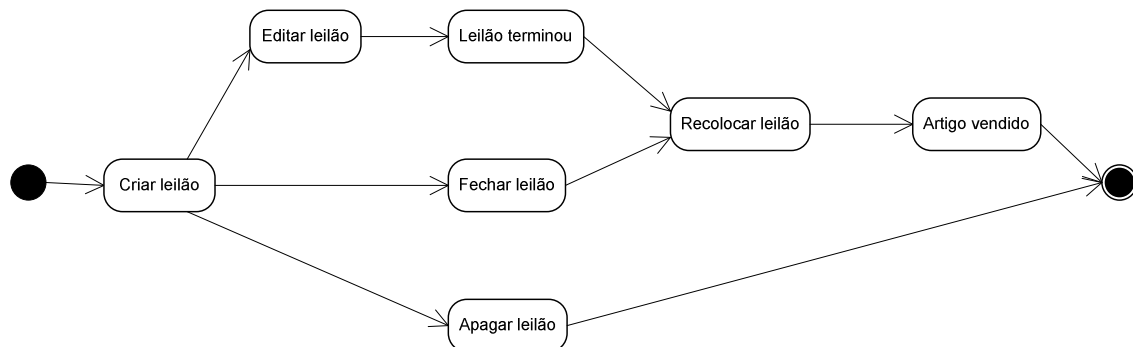
O diagrama de transição de estado é tipicamente um complemento para a descrição das classes. Este diagrama mostra todos os estados possíveis que os objectos de uma certa classe podem encontrar e mostra também quais são os eventos do sistemas que provocam tais mudanças. Os diagramas de transição de estado não são escritos para todas as classes de um sistema, mas apenas para aquelas que possuem um número definido de estados conhecidos e onde o comportamento das classes é afectado e modificado pelos diferentes estados.

Diagramas de transição de estado capturam o ciclo de vida dos objectos, subsistemas e sistemas. Estes mostram os estados que um objecto pode possuir e como os eventos (mensagens recebidas, erros, e condições a ser satisfeitas) afectam estes estados com o passar do tempo.

Diagramas de transição de estado possuem um ponto de início e vários pontos de finalização. Um ponto de início (estado inicial) é mostrado como um círculo todo preenchido, e um ponto de finalização (estado final) é mostrado como um círculo em volta de um outro círculo menor preenchido. Um estado é mostrado como um rectângulo com cantos arredondados. Entre os estados estão as transições, mostrados como uma linha com uma seta no final de um dos estados. A transição pode ser nomeada com o seu evento causador. Quando o evento acontece, a transição de um estado para outro é executada ou disparada.

Uma transição de estado normalmente possui um evento ligado a ela. Se um evento é anexado a uma transição, esta será executada quando o evento ocorrer. Se uma transição não possuir um evento ligado a ela, a mesma ocorrerá quando a acção interna do código do estado for executada (se existirem acções internas como entrar, sair, fazer ou outras acções definidas pelo programador). Então quando todas as acções forem executadas pelo estado, a transição será disparada e serão iniciadas as actividades do próximo estado no diagrama de estados.

## Diagrama de transição de estados



## 5. Funcionamento do Sistema

### 5.1. Interface Administrador

Nesta secção irei abordar o que cada administrador do sistema poderá realizar, explicando as suas funcionalidades e características.

Para o administrador aceder à interface terá que primeiro realizar o registo na base de dados quando se fizer a instalação da aplicação e só assim poderá depois fazer o login para aceder à aplicação.

Na página principal do administrador irão ser apresentadas algumas definições do site e estatísticas de leilões e utilizadores registados.

- Definições: Nesta secção o administrador pode alterar o nome do site, url, email do administrador e mensagem de copyright;
- Erros: Aqui o administrador irá colocar o texto de erros a exibir aos utilizadores caso ocorram erros quando estiverem a navegar pelo site. Também pode colocar o email para qual os erros irão ser enviados;
- Categorias: Nesta secção o administrador pode adicionar categorias na base de dados que vão ser apresentadas no site;

- Preferências: Nesta secção o administrador pode fazer uma série de alterações no que diz respeito a definições de moedas, fuso horário a apresentar, formato da data, contadores a exibir na homepage do site sobre leilões abertos, utilizadores registados e utilizadores online;
- Utilizadores e Administradores: Nesta secção o administrador pode fazer o controle sobre os utilizadores registados no site. Esta secção permite ao administrador editar dados pessoais dos utilizadores e suspender ou apagar contas de utilizadores por violação de regras ou utilizadores que já não estejam activos.  
Aqui também será colocado o texto de confirmação de registo dos utilizadores a apresentar na página de registo do site. Esta secção permite ainda criação de novos administradores;
- Leilões: Nesta secção o administrador poderá ver o número de leilões abertos, suspensos e fechados. Esta permite editar todos os aspectos dos leilões, ver quem colocou o leilão, o início e o fim da data limite do leilão assim como a categoria do leilão;
- Página em manutenção: Caso ocorram alguns erros inesperados ou o site for para actualização o administrador pode activar esta opção de modo a exibir uma mensagem no site a avisar que o site se encontra em manutenção.

## 5.2. Interface Utilizador

- Home: Página principal do site onde o utilizador pode ver os leilões abertos recentemente, licitações mais altas e leilões finalizados.  
O utilizador pode também ver todas as categorias existentes na base de dados e ao carregar sobre uma categoria irão ser apresentadas noutra página as subcategorias existentes dentro dessa categoria. Aí irão aparecer os leilões activos correspondentes à categoria que carregou e permite também fazer pesquisas dentro dessa categoria.

Na homepage o utilizador também pode proceder ao registo no site e ao login o que permitirá vender produtos.

Ainda existe a página de pesquisa avançada que permite ao utilizador refinar a sua pesquisa sobre produtos postos à venda.

- Painel de Controle: Dentro do painel de controle existem 4 páginas:
  - Geral: Nesta página o utilizador irá receber avisos sobre mensagens enviadas por outros utilizadores de perguntas sobre leilões e avisos para o utilizador deixar comentários aos artigos comprados;
  - A Minha Conta: Nesta página o utilizador pode editar todos os seus dados pessoais que fez quando se registou, ver comentários sobre os seus produtos, ver mensagens que outros utilizadores tenham enviados sobre os leilões, deixar comentários a outros utilizadores e ver os pagamentos que necessitam de ser feitos sobre produtos comprados;
  - A Vender: Nesta página o utilizador pode aceder à secção vender onde pode colocar os seus produtos para venda no site. Pode verificar os leilões activos, pendentes, suspensos e fechados. Isto permite ao utilizador editar os seus leilões, fechar leilões, recolocar leilões à venda caso não tenham sido vendidos os produtos, ver número de licitações sobre um leilão e quem o licitou e mais algumas definições. O utilizador pode ainda ver os artigos vendidos e caso pretenda colocar artigos semelhantes à venda e ainda ver os vencedores dos seus leilões;
  - A Comprar: Nesta página o utilizador pode pesquisar palavras nos leilões sendo depois notificado caso sejam postos à venda leilões que contenham essa palavra e ver os seus artigos vigiados que o utilizador escolhe para poder segui-los mais de perto. O utilizador pode ainda ver as suas licitações feitas nos leilões e ver os leilões que ganhou.

## 6. Conclusão e Trabalhos Futuros

Este trabalho teve como objectivos realizar a modelação e desenvolvimento de um site de leilões online.

Uma vez terminada a implementação do sistema, notou-se a importância da modelação no processo de concepção de um produto de *software*, pois além de servir de base para a construção do sistema, a modelação tem o intuito de servir também como documentação do *software* caso seja necessário realizar alguma alteração no código.

Os leilões online tornaram-se para muitas pessoas uma forma de ganhar dinheiro. O modelo de negócio dos leilões online inclui um utilizador que vende produtos e serviços, enquanto outro utilizador licita para os comprar. Há uma série de vantagens neste modelo de negócio de leilões online, entre elas a capacidade de não ter barreiras geográficas. A internet permite a vendedores e licitadores participar em leilões online a partir de qualquer parte no mundo. O custo de participação num leilão é praticamente nulo. Esta vantagem também aumenta o número de produtos e serviços que podem ser listados, levando a um número acrescido de vendedores e licitadores.

Outra vantagem é que os produtos podem ser postos em sites de leilões a qualquer hora do dia e as licitações dos produtos também podem ser feitas a qualquer hora do dia.

Uma vez que os leilões online permitem a possibilidade de encontrar uma grande variedade de produtos a preços inferiores aos de retalho, leva muitos consumidores a tentar a sua sorte em conseguir um bom negócio através de um leilão online. A facilidade do processo de licitação também atrai os consumidores que podem licitar, através dos seus computadores, no conforto das suas casas.

Por outro lado, a possibilidade de conseguir vender produtos a preços mais elevados do que os de retalho também atrai muitos vendedores. Vender os seus produtos, através de leilões online, também reduz os seus custos de venda e também é muito simples de o fazer. Daí vemos que hoje em dia a expansão deste tipo de negócio cresce cada vez mais.

Para trabalho futuro, visto que o Wampserver não permitiu o teste de envio de emails para os utilizadores irá ser testado num servidor online para confirmação de

registo de utilizadores e de leilões como também para envio de mensagens entre utilizadores que permite colocar perguntas sobre os seus produtos à venda,

Outro método a ser implementado será a inclusão de outros modos de pagamento visto que para isso outras entidades terão que estar envolvidas no processo, como por exemplo, pagamentos através de visa, multibanco online, etc.

Concluindo posso dizer que este projecto realiza as principais utilidades para que foi proposto permitindo a venda e compra de produtos online com rapidez e facilidade.

## 7. Bibliografia

- <http://pt.wikipedia.org/wiki/>
- <http://www.php.net/>
- <http://pear.php.net/>
- <http://ckeditor.com/>
- <http://freshmeat.net/>
- <http://plugins.jquery.com/project/dimensions>