



Internet Databases

By 1068664

This document shows a website and database creation from the initial design to the database in production. It details the website design and why it was designed that way. It details a list of web pages used and a hierarchy to show how the pages fit together. It also shows sample code of the most complex parts of the site and some simple code examples too. It contains screenshots as a user guide.

INDEX

Introduction	3
Analysis & Design	3
Final Design	8
Full Website List	9
Hierarchy for /	11
Hierarch for /admin	12
Data Dictionary	13
Process Specifications	16
Complex Code Samples	36
Simple Code Samples	40
User Guide	41
Bibliography	53
Appendix (Visited Websites)	54

INTRODUCTION

The aim of this assignment was to build and implement a PHP website with a MySQL backend database. The aim of the website was to mirror that of a car sales website. The website had to be able to connect to a database so that records could be searched for a car, or a dealer. The website had to allow for the purchase of a car. It also had to allow for an admin section so that any admin duties could be performed on the records.

ANALYSIS AND DESIGN

A meeting with the client was set up so that an analysis of what was wanted and a design could be drafted to show what the website would look like. The client stated that they had looked at various other websites, namely autotrader.co.uk and motors.co.uk

The client stated that they had looked at other sites, but that the 2 mentioned were more pleasing. The client stated that they liked the 'square-ish' look of AutoTrader and how the cars were presented in a 'business card like' format. They also stated that they liked the light, neutral colours of AutoTrader.

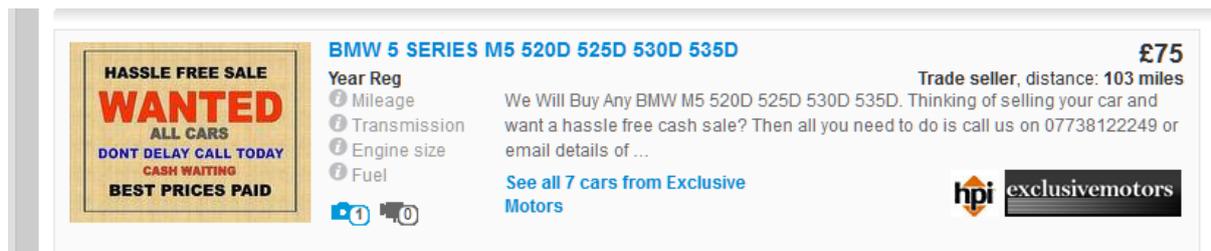


Light, neutral colours of AutoTrader



Corners of AutoTrader

The client provided the team with an image of a search result from AutoTrader and specified that they would like something similar. They said that they didn't like the fact that it didn't have a defined border or separation from other returned results and this is a must for their site.



Search result from AutoTrader

The client also stated that they liked the tabular menu format of motors.co.uk. They stated they were, however, unimpressed with the colour scheme of motors.co.uk and would like to stay away from those colours.



Tabs from motors.co.uk

MotorMarket

The super way to find YOUR next car!



Home

Search query: Any make, Any model, Any colour, Any price, Any region, Any dealer (996 cars)

Jeep Cherokee
K Reg. Red. 45,000 miles. **£7,995**
[Off Road Sales](#)



Kia Pride
K Reg. White. 36,000 miles. **£1,850**
[Rook Used Cars](#)



Kia Pride
M Reg. Blue. 14,000 miles. **£3,400**
[South West Cars](#)



Kia Pride
P Reg. Red. 28,000 miles. **£4,150**
[Rook Used Cars](#)



Kia Pride
R Reg. Pale Blue. 1,650 miles. **£5,700**
[Hansen](#)



Quick Search

Select Make

Select Model

Select Colour

Select Price

Select Region

Select Dealer

Sort By

Keyword Search

Contact

01234 567891
sales@motormarket.com

Latest Updates

We now have 1000 cars for sale

« First Last »

© MotorMarket

First draft presented to client

Having finished the meeting with the client, a design was put together and then presented to the client. The client stated that they weren't happy with the rounded corners of each section and could they be squared corners instead.

MotorMarket

The super way to find YOUR next car!



Home

Search query: Any make, Any model, Any colour, Any price, Any region, Any dealer (998 cars)



[Jeep Cherokee](#)

K Reg. Red. 45,000 miles.

£7,995

[Off Road Sales](#)



[Kia Pride](#)

K Reg. White. 36,000 miles.

£1,850

[Rook Used Cars](#)



[Kia Pride](#)

M Reg. Blue. 14,000 miles.

£3,400

[South West Cars](#)



[Kia Pride](#)

P Reg. Red. 28,000 miles.

£4,150

[Rook Used Cars](#)



[Kia Pride](#)

R Reg. Pale Blue. 1,650 miles.

£5,700

[Hansen](#)

Quick Search

Select Make ▼

Select Model ▼

Select Colour ▼

Select Price ▼

Select Region ▼

Select Dealer ▼

Sort By ▼

Keyword Search

Contact

01234 567891
sales@motormarket.com

Latest Updates

We now have 1000 cars for sale

« First « Prev 1 2 3 4 Next » Last »

© MotorMarket

Second draft presented to client

5

The client also stated that a white background was not what they wanted either and they proposed to have a grey background instead.

MotorMarket

The super way to find YOUR next car!



Home

Search query: Any make, Any model, Any colour, Any price, Any region, Any dealer (996 cars)



[Jeep Cherokee](#)
K Reg. Red. 45,000 miles. **£7,995**
[Off Road Sales](#)



[Kia Pride](#)
K Reg. White. 36,000 miles. **£1,850**
[Rook Used Cars](#)



[Kia Pride](#)
M Reg. Blue. 14,000 miles. **£3,400**
[South West Cars](#)



[Kia Pride](#)
P Reg. Red. 28,000 miles. **£4,150**
[Rook Used Cars](#)



[Kia Pride](#)
R Reg. Pale Blue. 1,650 miles. **£5,700**
[Hansen](#)

Quick Search

Select Make

Select Model

Select Colour

Select Price

Select Region

Select Dealer

Sort By

Keyword Search

Contact

01234 567891
sales@motormarket.com

Latest Updates

We now have 1000 cars for sale

« First **1** [Next](#) » Last »

© MotorMarket

Third draft presented to client

The client said that they wanted a cartoon-like, relatable and friendly logo. This was the only spec given for the logo.



Logo presented to client

The client stated they were very happy at the logo that was designed for the site.

The client was impressed with the site name and site tag-line. As with the logo, little to no spec was given for this name. The marketing team of the design company was tasked with this. The response of the client was 'It's catchy, I like it' Well done marketing.

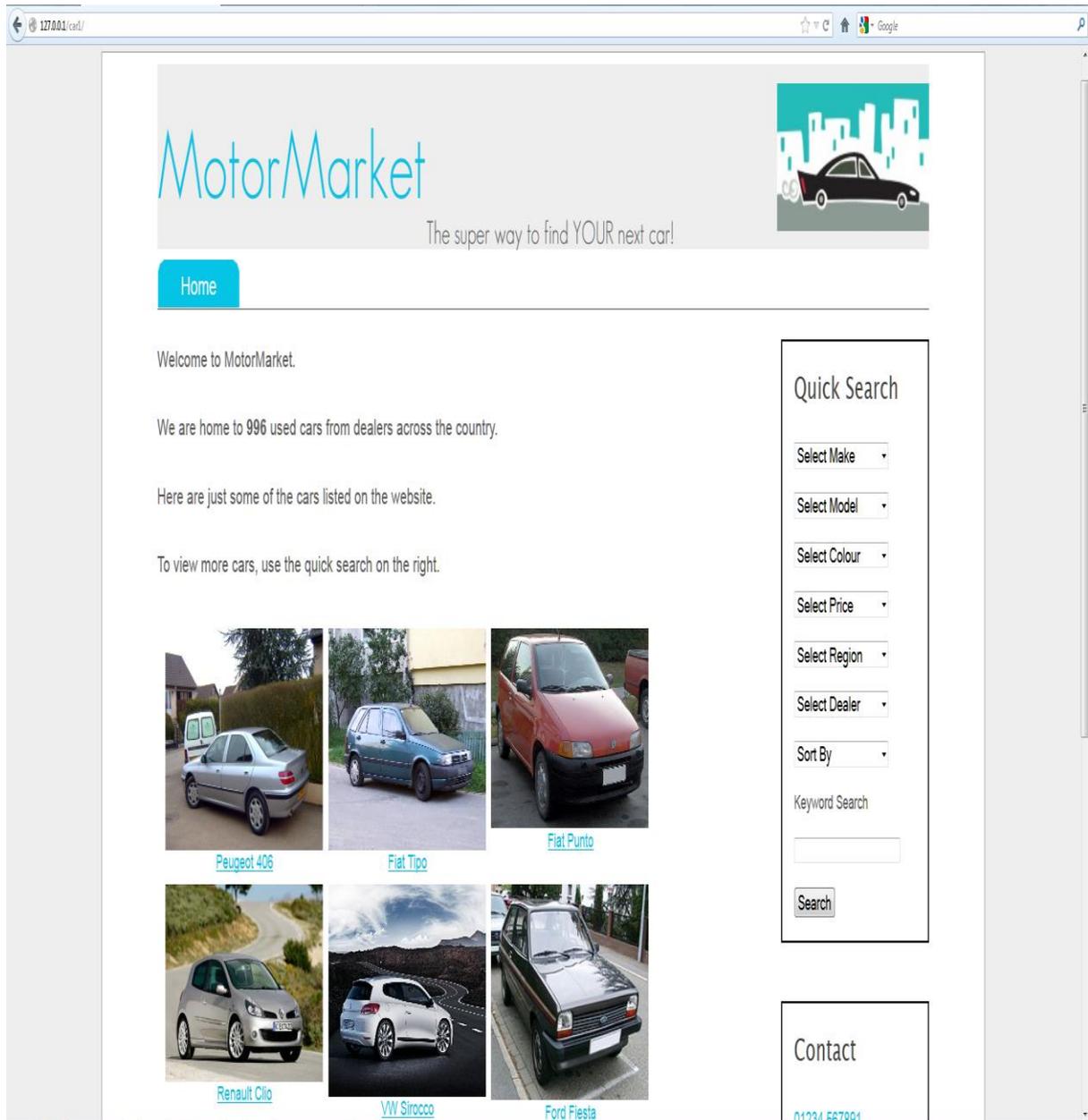
After the changes were made, the client was then presented with the revised changes and signed them off. The client stated, after they signed off on the design that they were colour blind. They stated that the design team had captured the simple, elegant and complimentary colours that the company was looking for to make the site pleasing to the eye.

After getting the documents signed off by the client, the coding team set to work on building the site. It was decided that the site should be built primarily in PHP with HTML playing a supporting role. PHP was chosen as it offers the flexibility of connecting to the MySQL database.

For the searching of the site, it was decided that a jQuery library would be utilised to provide AJAX searching.

You can view a copy of the final design on the next page.

FINAL DESIGN



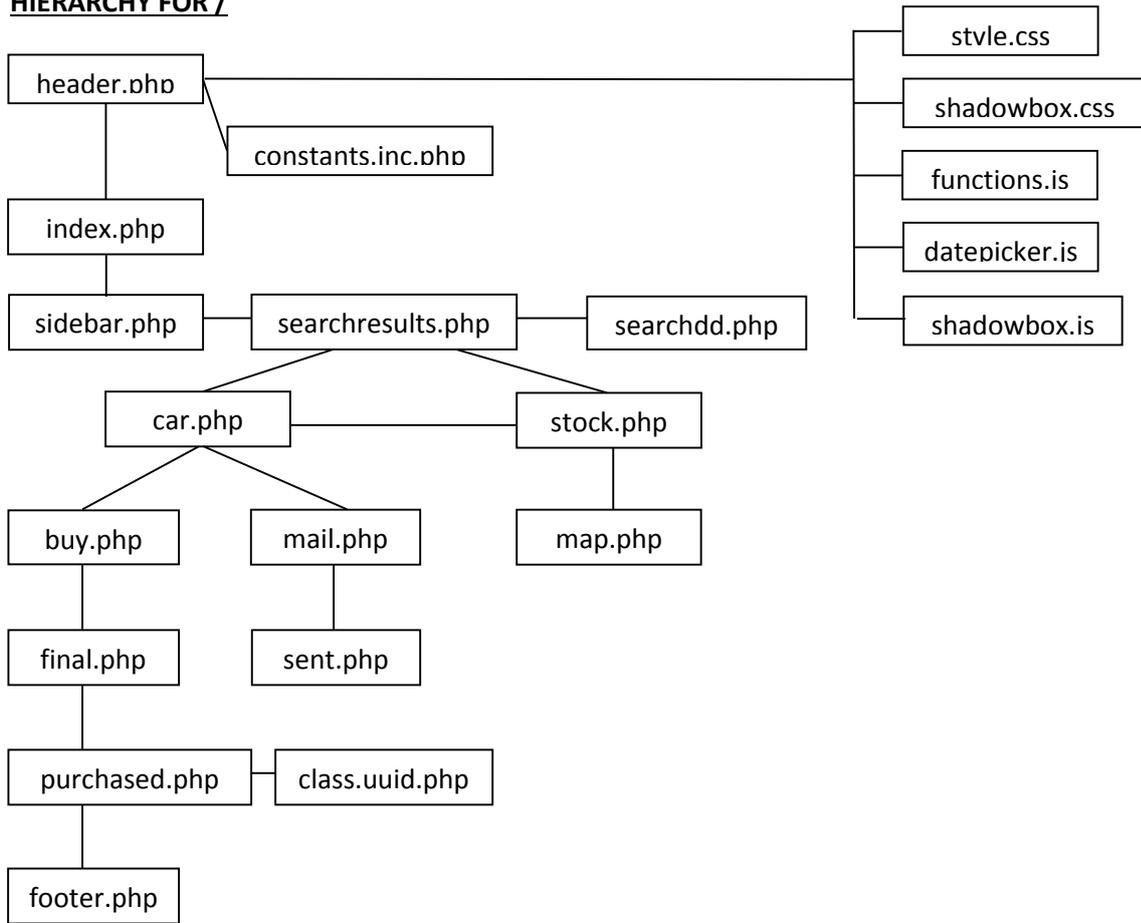
The final design that was signed off by the client

FULL WEBSITE LIST

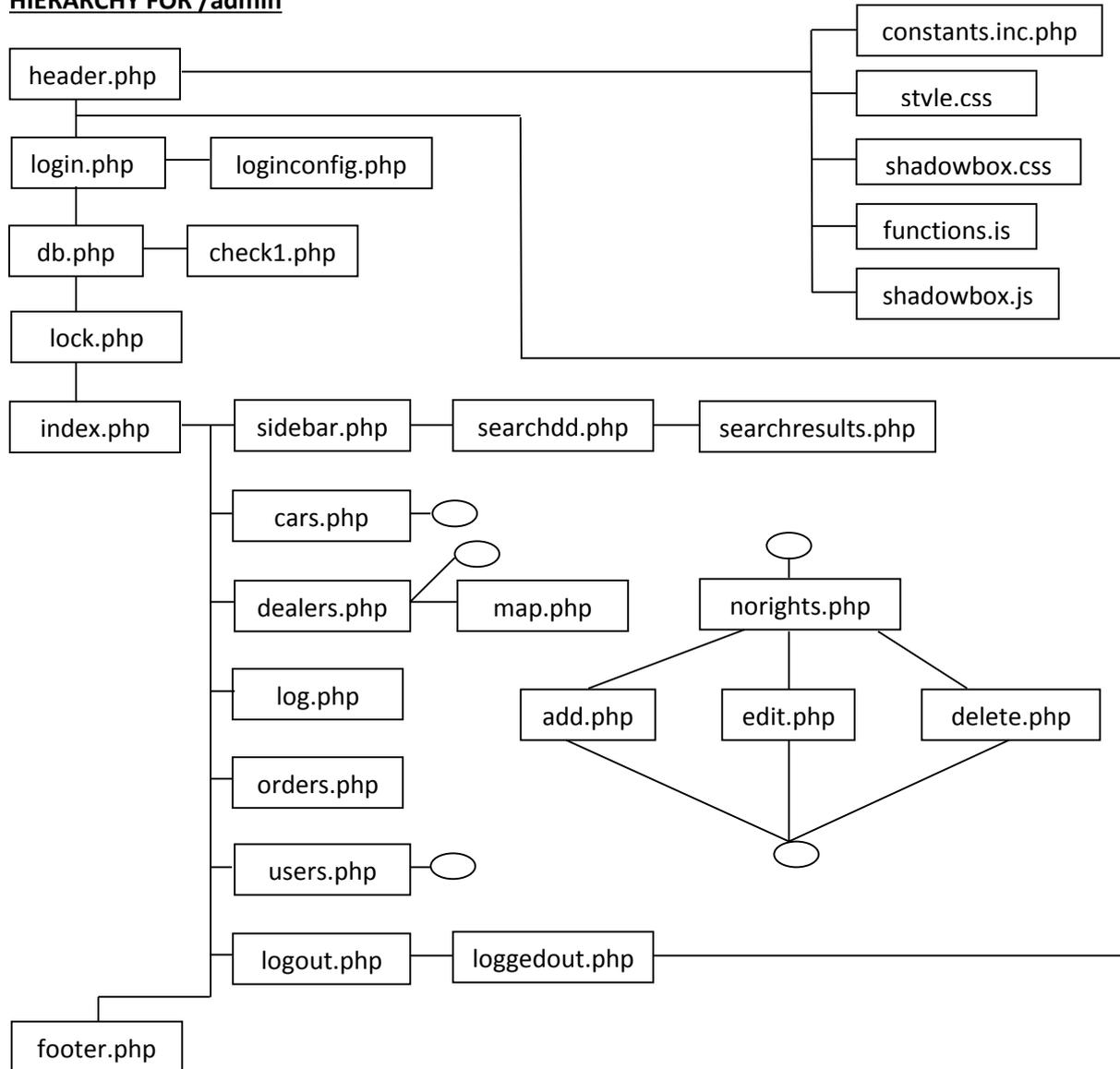
PAGE NAME	LOCATION	DESCRIPTION
buy.php	/	Starts the process of buying the car
car.php	/	Shows full details of the car
class.uuid.php	/	A class to generate a UUID to use as an order reference
constants.inc.php	/	Holds variables for use in the site. Was chosen so changes only need to be made on one page
final.php	/	Allows the customer to review order before purchasing
footer.php	/	Shows the footer of the website
header.php	/	Shows the site logo, name and tagline. Loads CSS and JS files
index.php	/	Main page of the website
mail.php	/	Allows the customer to email details of the car
map.php	/	Shows the location of a particular dealer
purchased.php	/	Shows the customer the purchase, needed documentation, details of dealer and insurance companies
searchdd.php	/	Holds the SQL for search box manipulation. Uses Ajax.
searchresults.php	/	Shows the results of the search a customer makes
sent.php	/	Shows the mail that the customer sent. Also used as mail doesn't send on Windows
sidebar.php	/	Holds the search, contact and latest update sections
stock.php	/	Shows what cars a dealer has in stock
shadowbox.css	/css	Stylesheet for the shadowbox
style.css	/css	Stylesheet for the website
datepicker.js	/scripts	Script for choosing collection date
functions.js	/scripts	Script for disabling/enabling delete button
jquery-1.7.1.js	/scripts	jQuery library
shadowbox.js	/scripts	Shadowbox library
add.php	/admin	Allows for the addition of cars, dealers and users
cars.php	/admin	Shows all cars in the database and admin options
check1.php	/admin	Used to get the password of a user
db.php	/admin	Creates a database string to get the password of a user
dealers.php	/admin	Shows all dealers in the database and admin options
delete.php	/admin	Allows for the deletion of cars, dealers and users
edit.php	/admin	Allows for the editing of cars, dealers and users
header.php	/admin	Shows the admin header, site logo, name and tagline. Loads CSS and JS files
index.php	/admin	Main admin page
lock.php	/admin	Used to check whether a user is logged in
log.php	/admin	Shows all database actions that have been logged
loggedout.php	/admin	Shows successful logout
login.php	/admin	Login page for a user to gain access to site
loginconfig.php	/admin	All the configuration for logging in to the site
logout.php	/admin	Logs a user out and destroys the session
norights.php	/admin	This page shows when a user has no access to a certain part of the site
orders.php	/admin	Shows all orders in the database
searchdd.php	/admin	Holds the SQL for search box manipulation. Uses Ajax.

searchresults.php	/admin	Shows the results of the search a user makes
sidebar.php	/admin	Holds the search, contact and latest update sections
users.php	/admin	Shows all users in the database and admin options

HIERARCHY FOR /



HIERARCHY FOR /admin



DATA DICTIONARY

Entity Name: admin						
Entity Description: Holds the users information						
Attribute	Description	Data Type	Length	Validation	Required	PK/FK
Id	Record ID	int	11		Y	PK
username	Username	varchar	30		Y	
password	Password	varchar	32		Y	
name	First Name	varchar	100		Y	
surname	Surname	varchar	100		Y	
emailaddress	Email address	varchar	255		Y	
usrlvl	User level	int	1	Dropdown	Y	
active	Current or past user	int	1	Dropdown	Y	
Linked Phrases						
Entity Name	Link Phrase	Relationship	Linked Entity Name			
NA						

Entity Name: carinfo						
Entity Description: Holds information about cars						
Attribute	Description	Data Type	Length	Validation	Required	PK/FK
umake	For makes with spaces	varchar	50		Y	
make	Car make	varchar	50		Y	
model	Car model	varchar	50		Y	
Reg	Reg letter	varchar	1		Y	
colour	Car colour	varchar	10		Y	
miles	Car mileage				Y	
price	Car price	Int	11		Y	
dealer	Dealership	varchar	7		Y	
description	Description	varchar	30		N	
keywords		varchar	255		N	
image	Car image	varchar	255		N	
sis	Show in search – for deleted cars	varchar	1		Y	
cbd	Can be deleted – whether car is allowed deletion	varchar	1		Y	
carIndex	Record ID	int	11		Y	PK
Linked Phrases						
Entity Name	Link Phrase	Relationship	Linked Entity Name			
NA						

Entity Name: customer						
Entity Description: Holds customer details						
Attribute	Description	Data Type	Length	Validation	Required	PK/FK
id	Record ID	Int	11		Y	PK
custnum	Customer Number	varchar	255		Y	
title	Title	varchar	5		Y	
surname	Surname	varchar	255		Y	
forename	First name	varchar	255		Y	
addressone	Address line one	varchar	255		Y	
addresstwo	Address line two	varchar	255		N	
town	Town	varchar	255		Y	
county	County	varchar	255		Y	
postcode	Postcode	varchar	8		Y	
landline	Home phone number	varchar	11		N	
mobile	Mobile number	varchar	11		N	
email	Email address	varchar	255		Y	
Linked Phrases						
Entity Name		Link Phrase		Relationship		Linked Entity Name
NA						

Entity Name: dealer						
Entity Description: Holds dealer details						
Attribute	Description	Data Type	Length	Validation	Required	PK/FK
id	Record ID	Int	11		Y	PK
code	Dealer code	varchar	7		Y	
name	Name	varchar	255		Y	
street	Street	varchar	100		Y	
town	Town	varchar	50		Y	
county	County	varchar	50		Y	
pcode	Postcode	varchar	6		Y	
lat	Latitude of address	float	11,7		Y	
lng	Longitude of address	float	11,7		Y	
region	Region	varchar	3		Y	
telephone	Phone number	varchar	20		Y	
Sis	Show in search – for deleted cars	varchar	1		Y	
cbd	Can be deleted – whether car is allowed deletion	varchar	1		Y	
Linked Phrases						
Entity Name		Link Phrase		Relationship		Linked Entity Name
NA						

Entity Name: log						
Entity Description: Keeps a log of all changes to database						
Attribute	Description	Data Type	Length	Validation	Required	PK/FK
id	Record ID	int	11		Y	PK
username	Username	varchar	30		Y	
date	Record date	date			Y	
time	Record time	time			Y	
task	What was performed	varchar	30		Y	
whattable	What table was it done on	varchar	50		Y	
notes	Any other information	text			Y	
Linked Phrases						
Entity Name	Link Phrase	Relationship	Linked Entity Name			
NA						

Entity Name: orders						
Entity Description: Holds order details						
Attribute	Description	Data Type	Length	Validation	Required	PK/FK
id	Record ID	int	11		Y	PK
custID	Customer ID	int	11		Y	
car	Car	int	11		Y	
dealer	Dealer	varchar	7		Y	
coldate	Collection date	date			Y	
orderid	Order ID	varchar	255		Y	
orderdate	Order date	date			Y	
ordertime	Order time	time			Y	
Linked Phrases						
Entity Name	Link Phrase	Relationship	Linked Entity Name			
NA						

PROCESS SPECIFICATIONS

Data that has (\$) before it, denotes that it is a variable but also used in a \$_GET or \$_POST

Program Name	/buy.php
Control Object	/buy.php
Calls	/final.php, /stock.php
Description	<p>Store data</p> <p>Stores the value of submit that is POSTED from buy.php for use in setting session data</p> <p>Store data</p> <p>Stores the session data for 'title, surname, forename, addressone, addresstwo, town, county, postcode, landline, mobile, email, confemail, cardtype, cardholdername, cardnumber, frommonth, fromyear, tomonth, toyear, issuenum, cv2, coldate'</p> <p>Get data</p> <p>Get data that is passed in the URL through 'q'</p> <p>Do process</p> <pre>SELECT ".CARS.".*, (price / 100 * 10) AS pc, ".DEALERS.".* FROM ".CARS." JOIN ".DEALERS." ON ".DEALERS.".code = ".CARS.".dealer WHERE carIndex = \$q</pre> <p>Do process</p> <pre>\$num=mysql_numrows(\$result);</pre> <p>Do process</p> <pre>SELECT ".CARS.".*, ".DEALERS.".* FROM ".CARS." JOIN ".DEALERS." ON ".DEALERS.".code = ".CARS.".dealer WHERE carIndex = \$q</pre> <p>Do process</p> <pre>\$row = mysql_fetch_array(\$result)</pre> <p>Do process</p> <pre>\$fields=mysql_fetch_object(\$res1)</pre>
Data	<p>title, surname, forename, addressone, addresstwo, town, county, postcode, landline, mobile, email, confemail, cardtype, cardholdername, cardnumber, frommonth, fromyear, tomonth, toyear, issuenum, cv2, coldate, price, pc, code, name, telephone, make, model, Reg, colour, miles, description, price, image, carIndex, (\$)q, \$result, \$num, \$res1, \$row, \$fields</p>

Program Name	/car.php
Control Object	/car.php
Calls	/buy.php, /mail.php, /stock.php
Description	Get data
	Get data that is passed in the URL through 'q'
	Do process
	Link to database and run SELECT ".CARS".*, ".DEALERS".* FROM ".CARS." JOIN ".DEALERS." ON ".DEALERS.".code = ".CARS.".dealer WHERE carIndex = \$q
	Do process
	\$num=mysql_numrows(\$result)
	Do process
	\$row = mysql_fetch_array(\$result)
Data	(\$)q, \$result, \$num, \$row, price, code, name, town, telephone, carIndex, make, model, Reg, colour, miles, image

Program Name	/constants.inc.php
Control Object	/constants.inc.php
Calls	/header.php
Description	Define data
	Defines constants for use in the site
	Do process
	Links to database through \$conn
Data	FILE_UPLOAD, SCRIPTS, IMAGES, STYLES, PATH, ADMIN_PATH, VAL_REF_ONE, VAL_REF_TWO, VAL_REF_THREE, ADMIN, CARS, DEALERS, ORDER, EMAILS, LOG, CUSTS. IMG, CARIMG, INSIMG, NOPIC, CTM, AVIVA, DL, DB_HOST, DB_USER, DB_PASSWORD, DB_NAME, SITE_NAME, TAGLINE, COPYRIGHT, EMAIL_DISCLAIMER, SALES_EMAIL, \$conn

Program Name	/final.php
Control Object	/final.php
Calls	/buy.php, /purchased.php
Description	Store data
	Stores the session data for 'title, surname, forename, addressone, addresstwo, town, county, postcode, landline, mobile, email, confemail, cardtype, cardholdername, cardnumber, frommonth, fromyear, tomonth, toyear, issuenum, cv2, coldate' POSTED from buy.php
	Get data
	Get data that is passed in the URL through 'q'
	Do process
	SELECT ".CARS.".*, (price / 100 * 10) AS pc, ".DEALERS.".* FROM ".CARS." JOIN ".DEALERS." ON ".DEALERS.".code = ".CARS.".dealer WHERE carIndex = \$q
Do process	
\$num=mysql_numrows(\$result)	
Do process	
\$row = mysql_fetch_array(\$result)	
Data	title, surname, forename, addressone, addresstwo, town, county, postcode, landline, mobile, email, confemail, cardtype, cardholdername, cardnumber, frommonth, fromyear, tomonth, toyear, issuenum, cv2, coldate, (\$)q, \$result, \$num, \$res1, \$row, price, pc, code, name, telephone, make, model, Reg, colour, miles, image, \$card

Program Name	/index.php
Control Object	/index.php
Calls	/car.php
	Do process
	Link to database and run SELECT carIndex, make, model, image FROM carinfo ORDER BY RAND() LIMIT 6
	Do process
	Link to database and run SELECT * FROM ".CARS." WHERE sis = 'Y'
	Do process
	\$num=mysql_numrows(\$q)
	Do process
\$row = mysql_fetch_array(\$sql)	
Data	\$sql, \$q, \$num, \$row, image, carIndex, make, model

Program Name	/mail.php
Control Object	/mail.php
Calls	/sent.php
Description	Get data
	Get data that is passed in the URL through 'q'
	Do process
	Link to database and run SELECT ".CARS.".*, ".DEALERS."* FROM ".CARS." JOIN ".DEALERS." ON ".DEALERS.".code = ".CARS.".dealer WHERE carIndex = \$q
Data	Do process
	\$row = mysql_fetch_array(\$sql)
	(\$)q, \$sql, \$row, price, code, name, town, telephone, make, model, Reg, colour, miles, image, carIndex

Program Name	/map.php
Control Object	/map.php
Calls	nil
Description	Get data
	Get data that is passed in the URL through 'id'
	Do process
	Run query SELECT * FROM ".DEALERS." WHERE id = \$id
	Do process
	\$num1=mysql_numrows(\$res1)
Data	Do process
	\$row = mysql_fetch_array(\$res1)
	(\$)id, \$res1, \$num1, \$row, lat, lng, name, street, town, county, pcode

Program Name	/purchased.php
Control Object	/purchased.php
Calls	nil
Description	Store data
	Stores the value of submit that is POSTED from purchased.php for use in setting session data
	Store data
	Stores the session data for 'title, surname, forename, addressone, addresstwo, town, county, postcode, landline, mobile, email, confemail, cardtype, cardholdername, cardnumber, frommonth, fromyear, tomonth, toyear, issuenum, cv2, coldate'
	Get data
	Get data that is passed in the URL through 'q'
	Do process
	Link to database and run SELECT ".CARS.".*, ".DEALERS."* FROM ".CARS." JOIN ".DEALERS."

	ON ".DEALERS.".code = ".CARS.".dealer WHERE carIndex = \$q
	Do process
	\$num=mysql_numrows(\$result)
	Do process
	Link to database and run SELECT ".CARS.".*, (price / 100 * 10) AS pc, (price / 100 * 90) AS rem, ".DEALERS.".* FROM ".CARS." JOIN ".DEALERS." ON ".DEALERS.".code = ".CARS.".dealer WHERE carIndex = \$q
	Do process
	SELECT ".CARS.".*, ".DEALERS.".* FROM ".CARS." JOIN ".DEALERS." ON ".DEALERS.".code = ".CARS.".dealer WHERE carIndex = \$q
	Do process
	\$record1=mysql_fetch_object(\$res2);
	Do process
	Link to database and run INSERT INTO ".CUSTS." (custnum, title, surname, forename, addressone, addresstwo, town, county, postcode, landline, mobile, email) VALUES ('\$custnum', "".\$_SESSION['title'].", "".\$_SESSION['surname'].", "".\$_SESSION['forename'].", "".\$_SESSION['addressone'].", "".\$_SESSION['addresstwo'].", "".\$_SESSION['town'].", "".\$_SESSION['county'].", "".\$_SESSION['postcode'].", "".\$_SESSION['landline'].", "".\$_SESSION['mobile'].", "".\$_SESSION['email'].")
	Do process
	\$lastid=mysql_insert_id();
	Do process
	Link to database and run INSERT INTO ".ORDER." (custID, car, dealer, coldate, orderid, orderdate, ordertime) VALUES ('\$lastid', '\$q', '\$dealer', "".\$_SESSION['coldate'].", '\$ordid', '\$today', '\$now')
	Do process
	Link to database and run UPDATE ".CARS." SET sis = 'N' WHERE carIndex = \$q
	Do process
	\$row = mysql_fetch_array(\$result)
	Do process
	\$record=mysql_fetch_object(\$res1)
Data	submit, title, surname, forename, addressone, addresstwo, town, county, postcode, landline, mobile, email, confemail, cardtype, cardholdername, cardnumber, frommonth,

	fromyear, tomonth, toyear, issuenum, cv2, coldate, (\$)q, \$sn, \$fn, \$hn, \$pc, \$ucsn, \$ucfn, \$uchn, \$ucpc, \$ssn, \$sfn, \$shn, \$custnum, \$today, \$now, \$ordday, \$guid, \$cb, \$ordid, \$result, \$num, \$res1, \$res2, \$record1, \$dealer, \$cust, \$lastid, \$orderid, \$car, \$row, \$record, \$lastfour, \$card, price, code, name, telephone, make, model, Reg, colour, miles, image, carIndex, pc, rem, street, pcode
--	--

Program Name	/searchdd.php
Control Object	/searchdd.php
Calls	/searchresults.php
Description	<p>Get data</p> <p>Get data that is passed in the URL through 'umake'</p> <p>Do process</p> <p>Link to database and run SELECT DISTINCT model FROM ".CARS." WHERE umake='\$umake' ORDER BY model ASC</p> <p>Do process</p> <p>\$nt=mysql_fetch_array(\$q)</p>
Data	(\$)umake, \$q, \$myarray, \$str, \$nt

Program Name	/searchresults.php
Control Object	/searchresults.php
Calls	/car.php, /stock.php
Description	<p>Get data</p> <p>Get data that is passed in the URL through 'mak, mod, col, keywords, price, area, deal, sort, page'</p> <p>Do process</p> <p>Link to database and run SELECT * FROM ".CARS." WHERE MATCH (make, model, colour, description) AGAINST('\$kw') AND ".CARS.".sis = 'Y' ORDER BY carIndex ASC LIMIT \$cur, \$max</p> <p>Do process</p> <p>\$num=mysql_numrows(\$result)</p> <p>Do process</p> <p>Link to database and run SELECT * FROM ".CARS." WHERE MATCH (make, model, colour, description) AGAINST('\$kw') AND ".CARS.".sis = 'Y' ORDER BY carIndex ASC</p> <p>Do process</p> <p>\$num1=mysql_numrows(\$res1)</p> <p>Do process</p> <p>Link to database and run SELECT name FROM dealer WHERE code = '\$deal'</p> <p>Do process</p> <p>Link to database and run SELECT name FROM ".DEALERS." WHERE code = '\$deal'</p>

	Do process \$draw = mysql_fetch_array(\$res2)
	Do process Link to database and run SELECT ".CARS".*, ".DEALERS".* FROM ".CARS." JOIN ".DEALERS." ON ".DEALERS.".code = ".CARS.".dealer WHERE umake LIKE '\$scarmake' AND model LIKE '\$scarmodel' AND colour LIKE '\$scarcolour' AND price \$pq AND region LIKE '\$region' AND dealer LIKE '\$deal' AND ".CARS.".sis = 'Y' \$so LIMIT \$cur, \$max
	Do process SELECT ".CARS".*, ".DEALERS".* FROM ".CARS." JOIN ".DEALERS." ON ".DEALERS.".code = ".CARS.".dealer WHERE umake LIKE '\$scarmake' AND model LIKE '\$scarmodel' AND colour LIKE '\$scarcolour' AND price \$pq AND region LIKE '\$region' AND dealer LIKE '\$deal' AND ".CARS.".sis = 'Y' \$so
	Do process \$row = mysql_fetch_array(\$result)
Data	\$scarmake, mak, \$scarmodel, mod, \$scarcolour, col, \$kw, keywords, (\$)price, \$region, area, (\$)deal, (\$)sort, (\$)page, \$sr, \$max, \$cur, \$result, \$num, \$res1, \$num1, \$total_pages, \$pq, \$reg, \$res2, \$draw, \$fullname, name, \$so, \$c, \$row, price, code, name, carIndex, make, model, Reg, colour, miles, image, \$range, \$prevpge, \$x, \$nextpage

Program Name	/sent.php
Control Object	/mail.php
Calls	nil
Description	Get data Get data that is passed in the URL through 'q'
	Do process Link to database and run SELECT carinfo.*, dealer.* FROM ".CARS." JOIN dealer ON dealer.code = carinfo.dealer WHERE carIndex = \$q
	Store data Store data from 'yname, yemail, fname, femail, make, model, message, carid' that is POSTED from /mail.php
	Do process \$row=mysql_numrows(\$sql)
Data	(\$)q, \$sql, (\$)yname, (\$)yemail, (\$)fname, (\$)femail, (\$)make, (\$)model, (\$)carid, \$subject, \$message, \$message1, \$header, \$row, name

Program Name	/sidebar.php
Control Object	/sidebar.php
Calls	/searchdd.php, /searchresults.php
Description	<p>Do process</p> <p>Link to database and run SELECT DISTINCT umake, make FROM ".CARS." ORDER BY make ASC</p> <p>Do process</p> <p>\$n=mysql_fetch_array(\$q)</p> <p>Do process</p> <p>Link to database and run SELECT DISTINCT colour FROM ".CARS." ORDER BY colour ASC</p> <p>Do process</p> <p>\$cr=mysql_fetch_array(\$c)</p> <p>Do process</p> <p>Link to database and run SELECT DISTINCT region FROM ".DEALERS." ORDER BY region ASC</p> <p>Do process</p> <p>\$rr=mysql_fetch_array(\$r)</p> <p>Do process</p> <p>Link to database and run SELECT DISTINCT name, code FROM ".DEALERS." WHERE sis = 'Y' ORDER BY name ASC</p> <p>Do process</p> <p>\$dn=mysql_fetch_array(\$d)</p>
Data	\$q, \$n, umake, make, \$c, \$cr, colour, \$r, \$rr, (\$)region, \$d, \$dn, code, name

Program Name	/stock.php
Control Object	/stock.php
Calls	/car.php, /map.php
Description	<p>Get data</p> <p>Get data that is passed in the URL through 'q, page'</p> <p>Do process</p> <p>Link to database and run SELECT ".CARS".*, ".DEALERS".* FROM ".CARS." JOIN ".DEALERS." ON ".DEALERS.".code = ".CARS.".dealer WHERE code = '\$q' AND ".CARS.".sis = 'Y' LIMIT \$cur, \$max</p> <p>Do process</p> <p>\$num=mysql_numrows(\$result)</p> <p>Do process</p> <p>Link to database and run SELECT ".CARS".*, ".DEALERS".* FROM ".CARS." JOIN ".DEALERS." ON ".DEALERS.".code = ".CARS.".dealer WHERE code = '\$q' AND ".CARS.".sis = 'Y'</p> <p>Do process</p> <p>\$num1=mysql_numrows(\$res1)</p>

	Do process
	\$record=mysql_fetch_object(\$res1)
	Do process
	\$row = mysql_fetch_array(\$result)
Data	(\$)q, (\$)page, \$max, \$cur, \$result, \$num, \$res1, \$num1, \$total_pages, \$record, telephone, name, id, street, town, pcode, \$row, price, telephone, carIndex, make, model, Reg, colour, miles, image, \$range, \$prevpag, \$x, \$nextpage

Program Name	/admin/add.php
Control Object	/admin/add.php, /admin/sidebar.php
Calls	/admin/add.php
Description	<p>Store data</p> <p>Store data from 'rectype, submit, username, password, forename, surname, emailaddress, usrlvl, make, model, Reg, colour, miles, price, dealer, description, cid, image, name, street, town, county, pcode, lat, lng, region, telephone' that is POSTED from /admin/add.php and /admin/sidebar.php</p> <p>Get data</p> <p>Get session data for 'rights'</p> <p>Get data</p> <p>Get data that is passed in the URL through 't'</p> <p>Do process</p> <p>Link to database and run INSERT INTO ".ADMIN." (username, password, name, surname, emailaddress, usrlvl, active) VALUES ('\$username', '\$password', '\$forename', '\$surname', '\$emailaddress', \$usrlvl, 1)</p> <p>Do process</p> <p>Link to database and run INSERT INTO ".LOG." (username, date, time, task, whattable, notes) VALUES (".\$_SESSION['login_user'].", '\$today', '\$now', 'Update', ".ADMIN.", '\$uid')</p> <p>Do process</p> <p>Link to database and run INSERT INTO ".CARS." (umake, make, model, Reg, colour, miles, price, dealer, description, image, sis, cbd) VALUES ('\$umake', '\$make', '\$model', '\$Reg', '\$colour', '\$miles', '\$price', '\$dealer', '\$description', '\$image', 'Y', 'N')</p> <p>Do process</p> <p>Link to database and run INSERT INTO ".LOG." (username, date, time, task, whattable, notes) VALUES (".\$_SESSION['login_user'].", '\$today', '\$now', 'Insert', 'car', '\$notes')</p> <p>Do process</p>

	<p>Link to database and run INSERT INTO ".DEALERS." (code, name, street, town, county, pcode, lat, lng, region, telephone, sis, cbd) VALUES ('\$code', '\$dealer', '\$street', '\$town', '\$county', '\$pcode', '\$lat', '\$lng', '\$region', '\$telephone', 'Y', 'N')</p>
	<p>Do process</p>
	<p>Link to database and run INSERT INTO ".LOG." (username, date, time, task, whattable, notes) VALUES (".\$_SESSION['login_user'].", '\$today', '\$now', 'Insert', ".DEALERS.", '\$code')</p>
	<p>Do process</p>
	<p>Link to database and run SELECT DISTINCT name, code FROM ".DEALERS." WHERE sis = 'Y' ORDER BY name ASC</p>
	<p>Do process</p>
	<p>\$dn=mysql_fetch_array(\$d)</p>
<p>Data</p>	<p>\$rec, rectype, \$type, t, (\$)username, (\$)password, (\$)forename, (\$)surname, (\$)emailaddress, (\$)usrlvl, \$sql, \$today, \$now, \$ulog, \$conn, (\$)make, (\$)model, (\$)Reg, (\$)colour, (\$)miles, (\$)price, (\$)dealer, (\$)description, \$carIndex, cid, \$image, \$umake, \$name_of_file, \$file_name, \$temp_name, \$file_type, \$file_size, \$base, \$target, \$ful, \$notes, name, (\$)street, (\$)town, (\$)county, (\$)pcode, (\$)lat, (\$)lng, (\$)region, (\$)telephone, \$deal, \$pcod, \$ucdn, \$ucdp, \$dn, \$dp, \$code, \$d, \$dn</p>

Program Name	/admin/cars.php
Control Object	/admin/cars.php
Calls	/admin/delete.php, /admin/edit.php
Description	<p>Get data Get value of URL passed through 'page'</p> <p>Get data Get session data for 'rights'</p> <p>Do process Link to data base and run SELECT ".CARS.".*, ".DEALERS.".* FROM ".CARS." JOIN ".DEALERS." ON ".DEALERS".code = ".CARS".dealer WHERE ".CARS".sis = 'Y' ORDER BY carIndex ASC LIMIT \$cur, \$max</p> <p>Do process \$num=mysql_numrows(\$result)</p> <p>Do process Link to database and run SELECT * FROM ".CARS." JOIN dealer ON ".DEALERS".code = ".CARS".dealer WHERE ".CARS".sis = 'Y' ORDER BY carIndex ASC</p> <p>Do process \$num1=mysql_numrows(\$res1)</p> <p>Do process \$row = mysql_fetch_array(\$result)</p>
Data	(\$)page, \$max, \$cur, \$result, \$num, \$res1, \$num1, \$total_pages, \$row, price, name, rights, carIndex, make, model, Reg, colour, miles, description, image, \$range, \$prevpage, \$x, \$nextpage

Program Name	/admin/check1.php
Control Object	/admin/login.php
Calls	/admin/db.php
Description	<p>Stores data Store data from 'search' POSTED from /admin/login.php</p> <p>Do process Link to database and run SELECT * FROM admin WHERE username = "" . \$word . ""</p> <p>Do process \$row = \$db->select_list(\$sql)</p>
Data	search, \$db, \$pword, \$sql, \$row, \$end_result, \$r, \$result, password

Program Name	/admin/db.php
Control Object	/admin/check1.php
Calls	nil
Description	Do process
	Link to database and run query from /admin/check1.php
	Do process
	\$row = mysql_fetch_array(\$q)
Data	\$dbh, \$q, \$ret, \$row

Program Name	/admin/dealers.php
Control Object	/admin/dealers.php
Calls	/admin/delete.php, /admin/edit.php
Description	Get data
	Get value of URL passed through 'page'
	Do process
	Link to database and run SELECT * FROM ".DEALERS." WHERE sis = 'Y' ORDER BY name ASC LIMIT \$cur, \$max
	Do process
	\$num=mysql_numrows(\$result)
	Do process
	Link to database and run SELECT * FROM ".DEALERS." WHERE sis = 'Y' ORDER BY name ASC
	Do process
\$num1=mysql_numrows(\$res1)	
Do process	
\$row = mysql_fetch_array(\$result	
Data	(\$)page, \$max, \$cur, \$result, \$num, \$res1, \$num1, \$total_pages, rights, \$row, id, name, street, town, county, pcode, telephone, region, \$region, \$reg, \$rs, \$range, \$prevpag, \$x, \$nextpage

Program Name	/admin/delete.php
Control Object	/admin/cars.php, /admin/dealers.php, /admin/delete.php, /admin/users.php
Calls	/admin/delete.php
Description	Get data
	Get session data for 'rights'
	Get data
	Get data passed in the URL through 'q, t, c, HTTP_REFERER'
	Store data
	Store data from 'getrid' that is POSTED from /admin/delete.php
	Do process
Link to database and run UPDATE ".CARS." SET sis = 'N' WHERE id = '\$id'	
Do process	

	<p>Link to database and run INSERT INTO ".LOG." (username, date, time, task, whattable, notes) VALUES (".\$_SESSION['login_user'].", '\$today', '\$now', 'Delete', 'Car', '\$notes')</p> <p>Do process</p> <p>Link to database and run UPDATE ".CARS." SET sis = 'N' WHERE dealer = '\$code'</p> <p>Do process</p> <p>Link to database and run UPDATE ".DEALERS." SET ".DEALERS.".sis = 'N' WHERE id = \$id</p> <p>Do process</p> <p>Link to database and run INSERT INTO ".LOG." (username, date, time, task, whattable, notes) VALUES (".\$_SESSION['login_user'].", '\$today', '\$now', 'Delete', 'Dealers & cars', '\$notes')</p> <p>Do process</p> <p>Link to database and run UPDATE ".ADMIN." SET active = 0 WHERE id = '\$id'</p> <p>Do process</p> <p>Link to database and run INSERT INTO ".LOG." (username, date, time, task, whattable, notes) VALUES (".\$_SESSION['login_user'].", '\$today', '\$now', 'Delete', 'User', '\$notes')</p> <p>Do process</p> <p>Link to database and run SELECT * FROM ".DEALERS." WHERE id = \$id</p> <p>Do process</p> <p>\$row = mysql_fetch_array(\$res2)</p> <p>Do process</p> <p>Link to database and run SELECT ".CARS.".*, ".DEALERS.".* FROM ".CARS." JOIN ".DEALERS." ON ".DEALERS.".code = ".CARS.".dealer WHERE carIndex = \$id</p> <p>Do process</p> <p>\$num1=mysql_numrows(\$res1)</p> <p>Do process</p> <p>\$row = mysql_fetch_array(\$res1)</p> <p>Do process</p> <p>Link to database and run SELECT * FROM ".ADMIN." WHERE id = \$id</p>
Data	<p>\$ref_page, HTTP_REFERER, (\$)id, q, \$type, t, \$code, c, \$today, \$now, rights, \$sqlc, \$notes, \$ulog, \$sqld, \$sqlu, \$referer, \$res2, \$row, code, name, street, town, county, pcode, lat, lng, region, telephone, \$res1, \$num1, carIndex, make, model, Reg, colour, miles, price, description, image, surname, username, password, emailaddress, usrlvl,</p>

Program Name	/admin/edit.php
Control Object	/admin/cars.php, /admin/dealers.php, /admin/edit.php, /admin/users.php
Calls	/admin/edit.php
Description	<p>Get data Get session data for 'rights'</p> <p>Get data Get data passed in the URL through 'q, t, p, HTTP_REFERER'</p> <p>Store data Store data from 'submit, make, model, Reg, colour, miles, price, dealer, description, cid, origimage, image, code, dealer, street, town, county, pcode, lat, lng, region, telephone, uid, username, password, forename, surname, emailaddress, usrlvl' that is POSTED from /admin/add.php and /admin/sidebar.php</p> <p>Do process Link to database and run UPDATE ".CARS." SET make='\$make', model='\$model', Reg='\$Reg', colour='\$colour', miles='\$miles', price='\$price', dealer='\$dealer', description='\$description', image='\$pic', carIndex = \$id WHERE carIndex = \$id</p> <p>Do process Link to database and run INSERT INTO ".LOG." (username, date, time, task, whattable, notes) VALUES ("\$_SESSION['login_user'].", '\$today', '\$now', 'Update', ".CAR.", '\$notes')</p> <p>Do process Link to database and run UPDATE ".DEALERS." SET code='\$code', name='\$dealer', street='\$street', town='\$town', county='\$county', pcode='\$pcode', lat='\$lat', lng='\$lng', region='\$region', telephone='\$telephone' WHERE id = \$id</p> <p>Do process Link to database and run INSERT INTO ".LOG." (username, date, time, task, whattable, notes) VALUES ("\$_SESSION['login_user'].", '\$today', '\$now', 'Update', ".DEALERS.", '\$id')</p> <p>Do process Link to database and run UPDATE ".ADMIN." SET id='\$uid', username='\$username', password='\$password', name='\$forename', surname='\$surname', emailaddress='\$emailaddress', usrlvl='\$usrlvl' WHERE id = \$uid</p> <p>Do process Link to database and run INSERT INTO ".LOG." (username, date, time, task, whattable, notes)</p>

	VALUES ("\$_SESSION['login_user'].", '\$today', '\$now', 'Update', ".ADMIN.", '\$uid')
	Do process
	Link to database and run SELECT ".CARS.*", ".DEALERS.*" FROM ".CARS." JOIN ".DEALERS." ON ".DEALERS."code = ".CARS."dealer WHERE carIndex = \$id
	Do process
	\$row = mysql_fetch_array(\$res1)
	Do process
	Link to database and run SELECT DISTINCT name, code FROM ".DEALERS." WHERE sis = 'Y' ORDER BY name ASC
	Do process
	\$dn=mysql_fetch_array(\$d)
	Do process
	Link to database and run SELECT * FROM ".DEALERS." WHERE id = \$id
	Do process
	\$row = mysql_fetch_array(\$res2)
	Do process
	Link to database and run SELECT * FROM ".ADMIN." WHERE id = \$id
	Do process
	\$row = mysql_fetch_array(\$res3)
Data	\$ref_page, HTTP_REFERER, \$id, q, \$type, t, (\$p, \$today, \$now, (\$make, (\$model, (\$Reg, (\$colour, (\$miles, (\$price, (\$dealer, (\$description, \$carIndex, cid, \$oi, origimage, (\$image, \$phmsg, \$name_of_file, \$pic, \$base, \$target, \$ful, \$notes, \$sql, \$ulog, rights, (\$code, (\$street, (\$town, (\$county, (\$pcode, (\$lat, (\$lng, (\$region, (\$telephone, (\$uid, (\$username, (\$password, (\$forename, (\$surname, (\$emailaddress, (\$usrlvl, \$referer, \$res1, \$row, code, carIndex, make, model, Reg, colour, miles, price, \$d, \$dn, description, image, \$res2, name, street, town, county, pcode, lat, lng, region, telephone, \$res3, usrlvl, id, surname, password, emailaddress

Program Name	/admin/index.php
Control Object	/admin/index.php
Calls	nil
Description	Get data Get session data for 'rights'
Data	rights

Program Name	/admin/lock.php
Control Object	/admin/loginconfig.php
Calls	/admin/add.php, /admin/cars.php, /admin/dealers.php, /admin/delete.php, /admin/edit.php, /admin/log.php, /admin/orders.php, /admin/searchresults.php, /admin/users.php,
Description	Store data Stores the session data for 'login_user, login_retval, timeout' Do process Link to database and run SELECT username FROM admin WHERE username='\$user_check' Do process \$row=mysql_fetch_array(\$ses_sql)
Data	login_user, \$ses_sql, \$row, \$login_session, username, \$ref_page, \$inactive, \$session_life

Program Name	/admin/log.php
Control Object	/admin/log.php
Calls	nil
Description	Get data Get session data for 'rights' Do process Link to database and run SELECT * FROM ".LOG." ORDER BY date DESC, time DESC LIMIT \$cur, \$max Do process \$num=mysql_numrows(\$result) Do process Link to database and run Do process SELECT * FROM ".LOG." ORDER BY date DESC, time DESC Do process \$num1=mysql_numrows(\$res1) Do process \$row = mysql_fetch_array(\$result)
Data	rights, (\$)page, \$max, \$cur, \$result, \$num, \$res1, \$num1, \$total_pages, row, username, date, time, task, whattable, notes, \$range, \$prevpage, \$x, \$nextpage

Program Name	/admin/login.php
Control Object	/admin/login.php
Calls	
Description	Store data
	Store data from 'username_box, password' that is POSTED from /admin/login.php
	Do process
	Link to database and run SELECT id, usrlvl, emailaddress FROM ".ADMIN." WHERE username='\$username' and password='\$mypassword'
	Store data
	Stores the session data for 'login_user, rights, mailfrom'
	Do process
	Link to database and run SELECT DISTINCT username FROM ".ADMIN."
	Do process
	\$n=mysql_fetch_array(\$q)
Do process	
Link to database and run SELECT id, usrlvl, emailaddress FROM admin WHERE username='\$username' and password='\$mypassword'	
Get data	
Get data passed in the URL through 'rp'	
Data	\$username, username_box, \$mypassword, password, \$sql, \$row, \$active, \$rights, \$mailfrom, \$count, (\$)rp, \$error, \$q, \$n

Program Name	/admin/loginconfig.php
Control Object	/admin/loginconfig.php
Calls	nil
Description	Store data
	Stores data for connection to database
	Do process
	\$bd = mysql_connect(\$mysql_hostname, \$mysql_user, \$mysql_password)
Data	\$mysql_hostname, \$mysql_user, \$mysql_password, \$mysql_database, \$bd

Program Name	/admin/norights.php
Control Object	/admin//add.php, /admin/delete.php, /admin/edit.php, /admin/log.php, /admin/orders.php, /admin/users.php
Calls	nil
Description	Store data
	Store the session data for rights
Data	rights

Program Name	/admin/orders.php
Control Object	/admin/orders.php
Calls	nil
Description	Store data
	Store the session data for rights
	Get data
	Get data that is passed in the URL through 'page'
	Do process
	Link to database and run SELECT ".CUSTS".*, ".ORDER".*, ".DEALERS".code, ".DEALERS".name, ".CARS".* FROM ".CUSTS." JOIN ".ORDER." ON ".ORDER".custID = ".CUSTS".id JOIN ".CARS." ON ".ORDER".car = ".CARS".carIndex JOIN ".DEALER." ON ".ORDER".dealer = ".DEALER".code ORDER BY orderid ASC, surname ASC, forename ASC LIMIT \$cur, \$max
	Do process
	\$num=mysql_numrows(\$result)
	Do process
	Link to database and run SELECT ".CUSTS".*, ".ORDER".*, ".DEALERS".code, ".DEALERS".name, ".CARS".* FROM ".CUSTS." JOIN ".ORDER." ON ".ORDER".custID = ".CUSTS".id JOIN ".CARS." ON ".ORDER".car = ".CARS".carIndex JOIN ".DEALER." ON ".ORDER".dealer = ".DEALER".code ORDER BY orderid ASC, surname ASC, forename ASC
Do process	
\$num1=mysql_numrows(\$res1)	
Do process	
\$row = mysql_fetch_array(\$result)	
Data	Rights, (\$)page, \$max, \$cur, \$result, \$num, \$res1, \$num1, \$total_pages, \$row, image, orderid, surname, forename, custnum, adresssone, addresstwo, town, county, postcode, landline, mobile, email, make, model, name, orderdate, ordertime, coldate, \$range, \$prevpge, \$x, \$nextpage

Program Name	/admin/searchdd.php
Control Object	/admin/searchdd.php
Calls	/admin/searchresults.php
Description	Get data
	Get data that is passed in the URL through 'umake'
	Do process
	Link to database and run SELECT DISTINCT model FROM ".CARS." WHERE umake='\$umake' ORDER BY model ASC
	Do process
	\$nt=mysql_fetch_array(\$q)
Data	(\$)umake, \$q, \$myarray, \$str, \$nt

Program Name	/admin/searchresults.php
Control Object	/admin/searchresults.php
Calls	/admin/edit.php, /admin/delete.php
Description	Get data
	Get data that is passed in the URL through 'mak, mod, deal, sort, page'
	Do process
	Link to database and run SELECT name FROM ".DEALERS." WHERE code = '\$deal'
	Do process
	\$drow = mysql_fetch_array(\$res2)
	Do process
	SELECT ".CARS.".*, ".DEALERS.".* FROM ".CARS." JOIN ".DEALERS." ON ".DEALERS.".code = ".CARS.".dealer WHERE umake LIKE '\$scarmake' AND model LIKE '\$scarmodel' AND dealer LIKE '\$deal' AND ".CARS.".sis = 'Y' \$so LIMIT \$cur, \$max
	Do process
	\$num=mysql_numrows(\$result)
	Do process
	Link to database and run SELECT * FROM ".CARS." JOIN ".DEALERS." ON ".DEALERS.".code = ".CARS.".dealer WHERE umake LIKE '\$scarmake' AND model LIKE '\$scarmodel' AND dealer LIKE '\$deal' AND ".CARS.".sis = 'Y' \$so
	Do process
	\$num1=mysql_numrows(\$res1)
Get data	
Get session data for 'rights'	
Do process	
\$row = mysql_fetch_array(\$result)	
Data	\$scarmake, mak, \$scarmodel, mod, (\$)deal, (\$)sort, (\$)page, \$sr, \$scarmake1, \$res2, \$drow, \$fullname, \$so, \$max, \$cur, \$result, \$num,

	\$res1, \$num1, \$total_pages, rights, \$c, \$row, price, name, carIndex, make, model, Reg, colour, miles, description, image, \$range, \$prevpage, \$x, \$nextpage
--	---

Program Name	/admin/sidebar.php
Control Object	/admin/sidebar.php
Calls	/admin/searchdd.php, /admin/searchresults.php
Description	Do process
	Link to database and run SELECT DISTINCT umake, make FROM ".CARS." ORDER BY make ASC
	Do process
	\$n=mysql_fetch_array(\$q)
	Do process
	Link to database and run SELECT DISTINCT name, code FROM ".DEALERS." WHERE sis = 'Y' ORDER BY name ASC
Data	Do process
	\$dn=mysql_fetch_array(\$d)
Data	\$q, \$n, \$d, \$dn

Program Name	/admin/users.php
Control Object	/admin/users.php
Calls	/admin/delete.php, /admin/edit.php
Description	Get data
	Get session data for 'rights'
	Do process
	Link to database and run SELECT * FROM ".ADMIN." WHERE active = 1 ORDER BY name ASC LIMIT \$cur, \$max
	Do process
	\$num=mysql_numrows(\$result)
	Do process
	SELECT * FROM ".ADMIN." WHERE active = 1 ORDER BY name ASC
	Do process
	\$num1=mysql_numrows(\$res1)
Data	Do process
	\$row = mysql_fetch_array(\$result)
Data	rights, (\$)page, \$max, \$cur, \$result, \$num, \$res1, \$num1, \$total_pages, \$row, id, name, surname, username, password, emailaddress, \$range, \$prevpage, \$x, \$nextpage

COMPLEX CODE SAMPLES

The following coding examples show where the assignment scope has been extended.

The assignment doesn't ask that emails be sent to the customer. It is important with any purchase that the customer receives an email once this is complete. The code below is for emailing a link to the car to a friend. The method for emailing the transaction to the customer follows the same format.

```
$subject = "You've been sent a car to view on ".SITE_NAME."";

/* This is to encode and decode the message box if it contains HTML characters. If it does contain
them, sanitize them for database entry. Then decode them so that they can be shown if HTML type is
selected */
$message = htmlspecialchars($_POST['message'], ENT_NOQUOTES);
$message1 = htmlspecialchars_decode($_POST['message'], ENT_QUOTES);
$message1 .= "Dear ".$fname;
$message1 .= "\n\n";
$message1 .= $yname." [".$yemail."] thought you might be interested
in the following car";
$message1 .= "\n\n";
$message1 .= $make." ".$model;
$message1 .= "\n\n";
$message1 .= "Click <a href='".PATH."?q=".$scarid."'>here</a> to view
the car";

/* Headers are set so that when the email is received it shows normally as would any other email that
is sent. We need to include these as the very least requirement. We need to set the correct mime type
or it will show up as a garbled message. */
$header = "From: ".$yemail."\r\n";
$header .= "Reply-To: ".$yemail."\r\n";
$header .= "MIME-Version: 1.0\r\n";

/* This is the actual function for sending the email. We need to put it all together in mail(). We are
telling it the address we want it sent to, the subject, leave the message blank as we will send it with
the headers. Then we set the headers. Should sending fail, we need a way to inform the user and this
is done with an if else. */
if (mail($emailto, $subject, "", $header))
{
    echo "Mail would usually be sent. Mail function is not enabled
on Windows";
}
else
{
    include ("header.php");
    echo "<title>Error sending email</title>";
    echo "There seems to be an error sending your email.";
    include ("footer.php");
}
}
```

A method for uploading a file – used for adding a car image. This has been used so that file names don't have to manually be entered into the database. It also helps to eliminate the need to remember to upload the file and edit the record in the database. This function takes care of all of that.

```

/* First we check to see if the image file box is empty. If it isn't then we can proceed with the function
*/
if (!empty($_FILES['image']['name']))
{
    /* We need to get the file name from the upload */
    $name_of_file = $_FILES['image']['name'];
    /* Next we assign the file name to a variable so we can use it elsewhere in the code */
    $file_name = $name_of_file;
    /* We use a temporary name so we can move the file */
    $temp_name = $_FILES['image']['tmp_name'];
    /* We get the image type and the image size in case any file manipulation on these variables
needs to occur */
    $file_type = $_FILES['image']['type'];
    $file_size = $_FILES['image']['size'];

    /* We need to get the base filename so we can move the file to a location that we specify in
$target */
    $base = basename($file_name);

    /* $target is the target of where we want the file to go. We then set the target to be the file
location and the file basename */
    $target = "../.IMAGES./.CARIMG./";
    $target = $target.$base;
}

/* This is a shorthand if else statement. We are saying here that if move_uploaded_file() was
successful, we give the user the message that it was moved and where it was moved to. If it isn't
successful, we tell the user that the upload didn't happen */
$ful = (move_uploaded_file($temp_name, $target)) ? ".$file_name."
was uploaded to ".$target." : ".$file_name.", was not uploaded.
Please try a manual upload.";

```

In a system that requires a user to login, a method for timing out the user is imperative to protect the user and the company. The code below sets a timeout in motion.

```
/* We need to set an inactive period. We need to set this in seconds. 900 seconds = 15 minutes
(900/60) */
$inactive = 900;
/* We check to see if the timeout has been added to the session. If it has, we can then proceed to run
the timeout function. */
if(isset($_SESSION['timeout']))
{
    /* We then get the current time and subtract it from the timeout */
    $session_life = time() - $_SESSION['timeout'];
    /* If the life of the inactive session is older than the timeout, we then regenerate a new
    session and destroy the old session */
    if ($session_life > $inactive)
    {
        session_regenerate_id();
        session_destroy();
    }
}
/* We set the current time to the session so we can check it against the timeout function */
$_SESSION['timeout'] = time();
```

A method for storing the page a user is trying to access before they were redirected to login

```

/* If the login session is not set, we need to get the user redirected so they can log in */
if(!isset($login_session))
{
    /* If the user has been redirected to log in, we set the referring page to be the page
    they are currently on */
    $refpage      =      $_SESSION['login_retval']      =
$_SERVER['PHP_SELF'];
    /* We then redirect the user to login.php with the referring page to be the page
    they were trying to access */
    header("Location: login.php?rp=$refpage");
}

```

A method for logging in the user and redirecting them to the page they were trying to access before they came to login.php

```

/* If there is a username and password match in the database, we can then proceed to register the
user and log them in. If not, we inform them that either the username or password is wrong */
if($count==1)
{
    /* Here we register the username and add it to the session */
    session_register("username");
    $_SESSION['login_user']=$username;
    /* We add the rights to the session, so we can determine if a user has access to that
    particular page */
    $_SESSION['rights'] = $rights;
    /* We add mailfrom to the session so we can record the emailaddress of the user should we
    need to send email inter-departmentally. This eliminates the need for the user to share
    personal email addresses. */
    $_SESSION['mailfrom'] = $mailfrom;
    /* We get the referring page so we can send the user back to the page they were trying to
    access before being asked to log in. */
    $rp = $_GET['rp'];
    header("location: $rp");
}
else
{
    $error="Your Login Name or Password is invalid";
}

```

SIMPLE CODE SAMPLES

The following coding examples are based on the criteria of the assignment.

The first is for a server connection and database selection.

```
We need to set up a connection to the server. We do this through mysql_connect()
$conn = mysql_connect(DB_HOST,DB_USER,DB_PASSWORD);
if (!$conn)
{
    We give an error message if the connection to the server was unsuccessful
    die('Could not connect: ' . mysql_error());
}
```

Successful connections proceed to database selection

```
mysql_select_db(DB_NAME, $conn);
```

Once a successful database connection is established, we can run queries on our tables

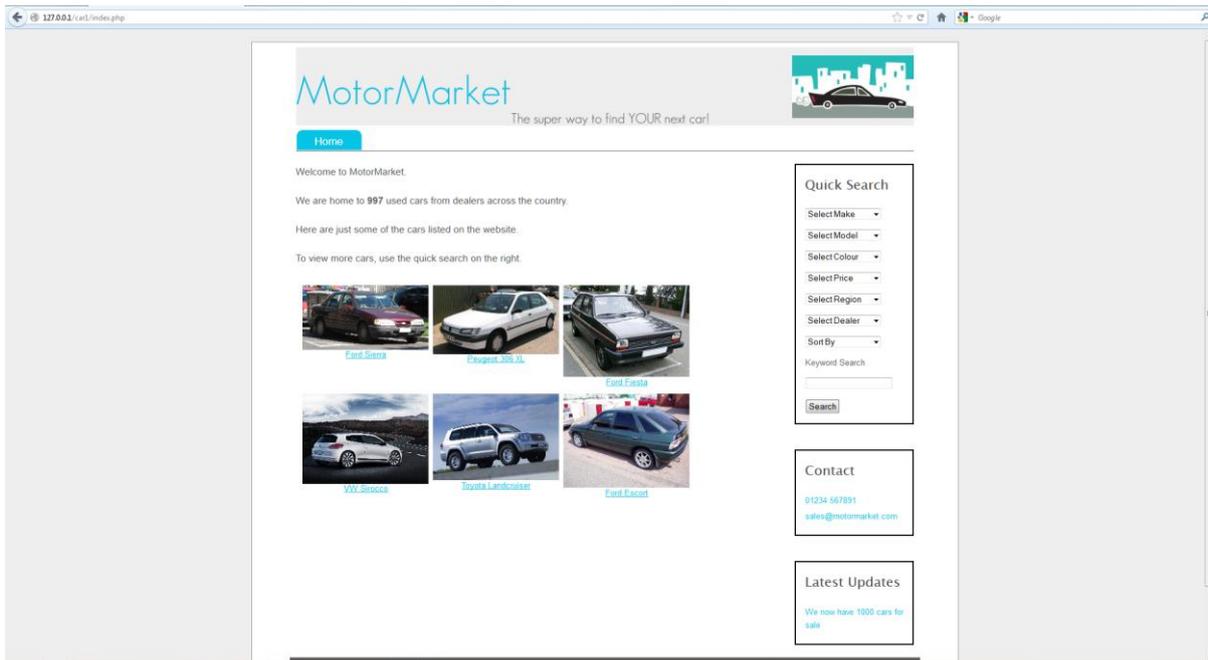
This next example shows a query on the 'cars' and 'dealers' tables.

```
We assign a variable and connect then run the query, this is done through mysql_query()
$result=mysql_query("SELECT ".CARS.".*, ".DEALERS.".* FROM ".CARS."
JOIN ".DEALERS." ON ".DEALERS.".code = ".CARS.".dealer WHERE
carIndex = $q");
```

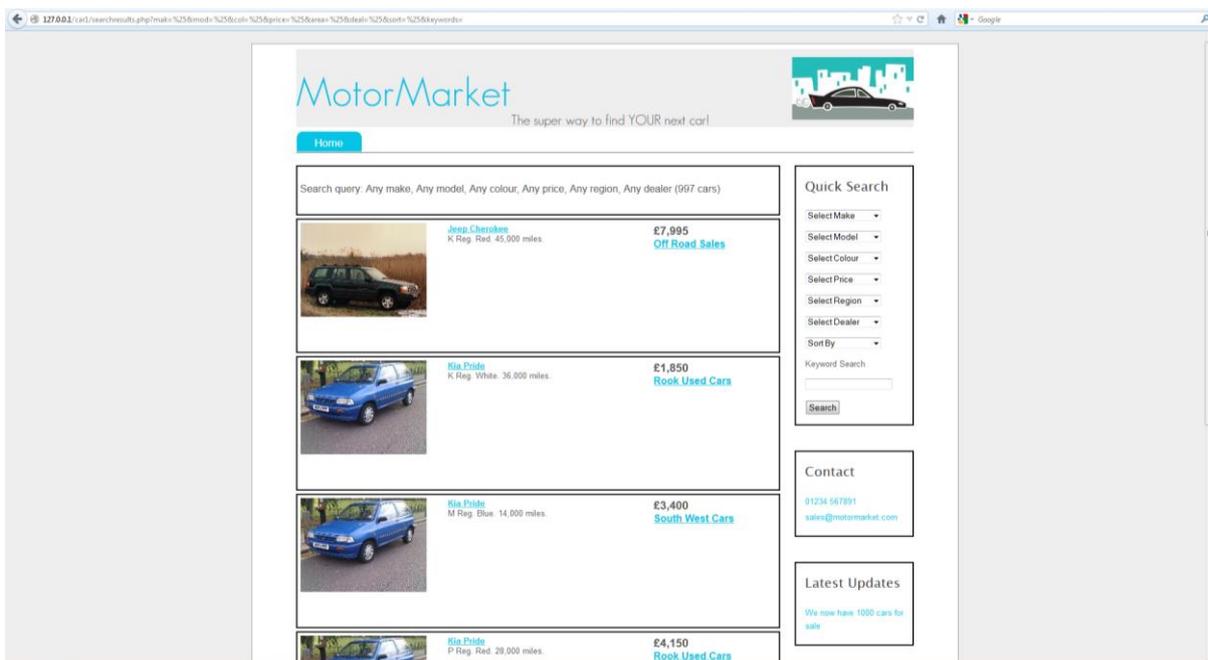
We can then get the number of records in the table. This is done with mysql_numrows(), it enables a record count to be shown anywhere in the program

```
$num=mysql_numrows($result);
```

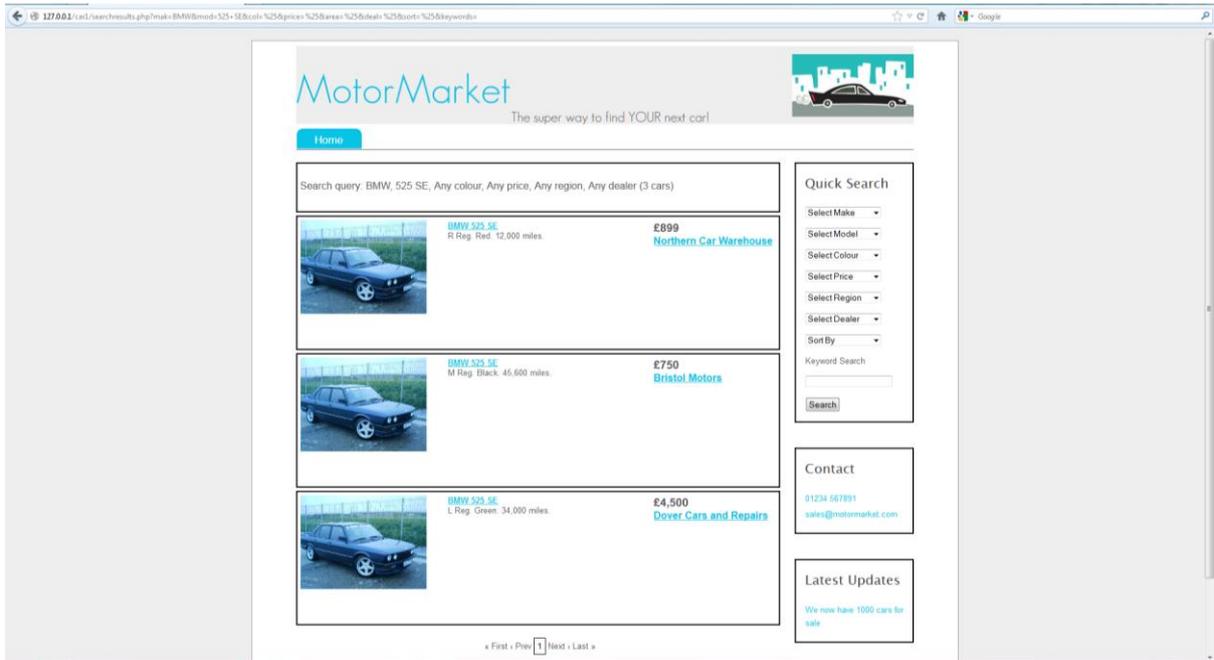
USER GUIDE



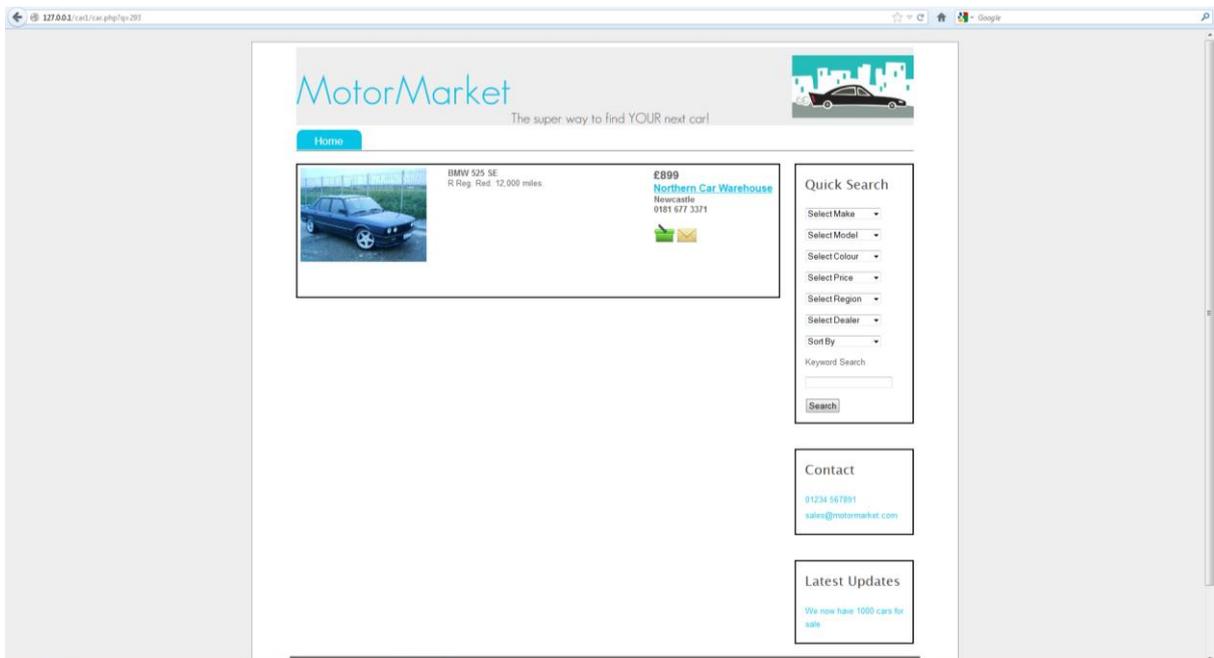
Home page of MotorMarket



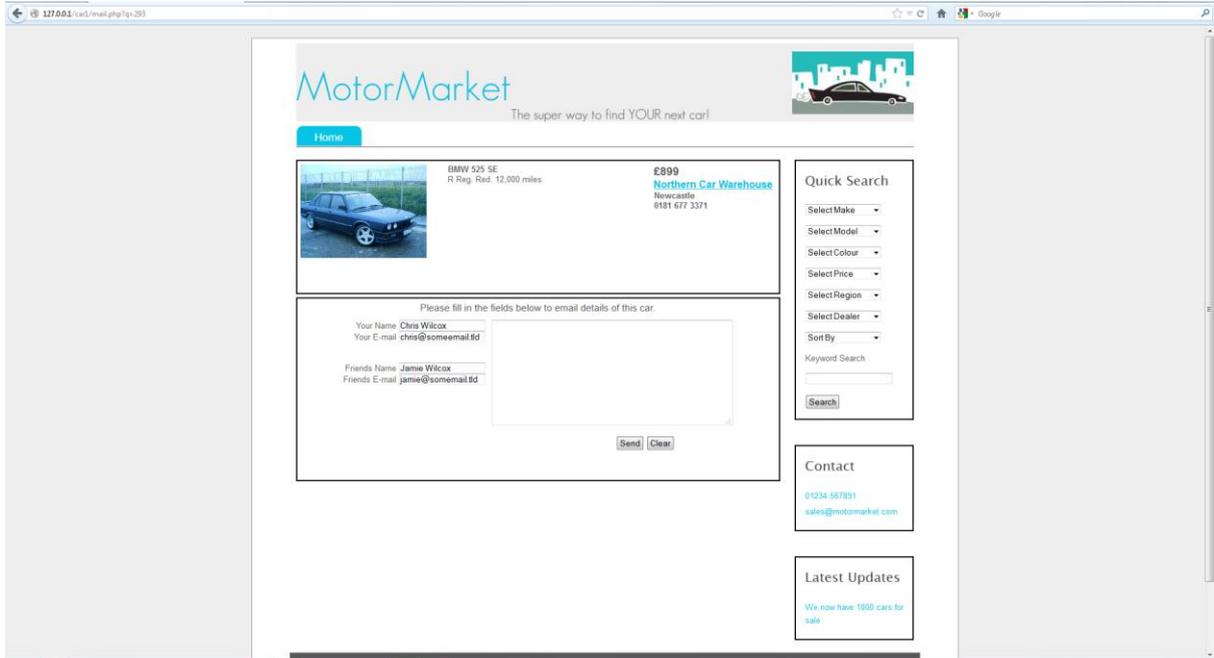
Results of generalised search



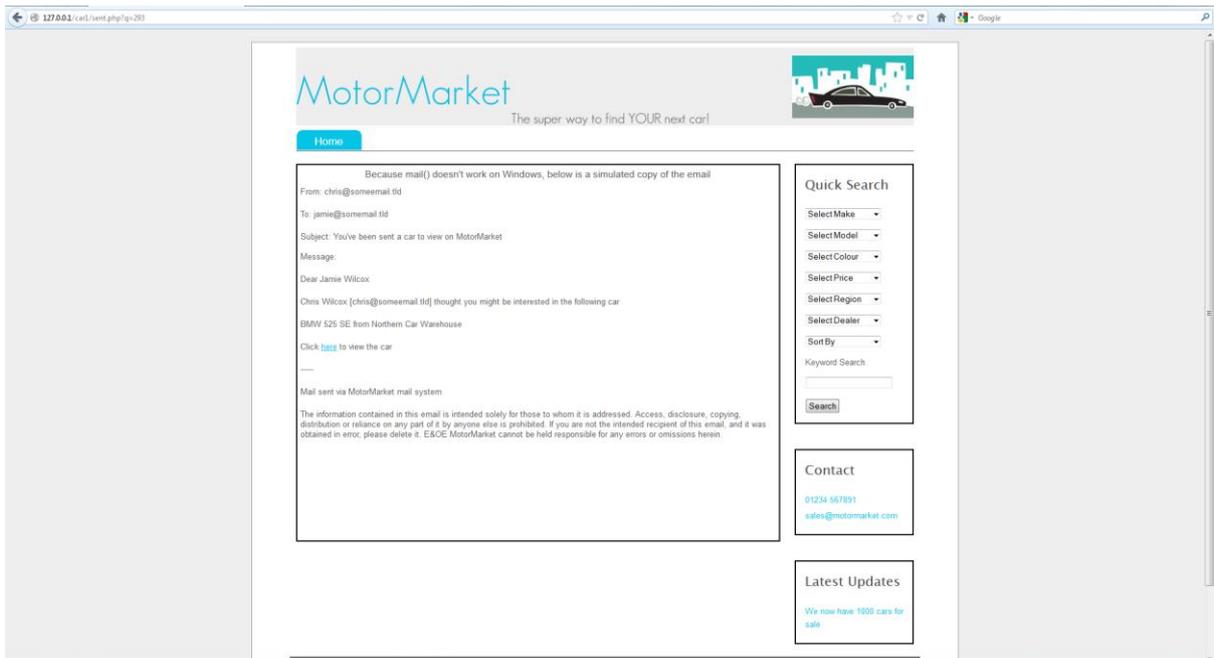
Result of make and model search



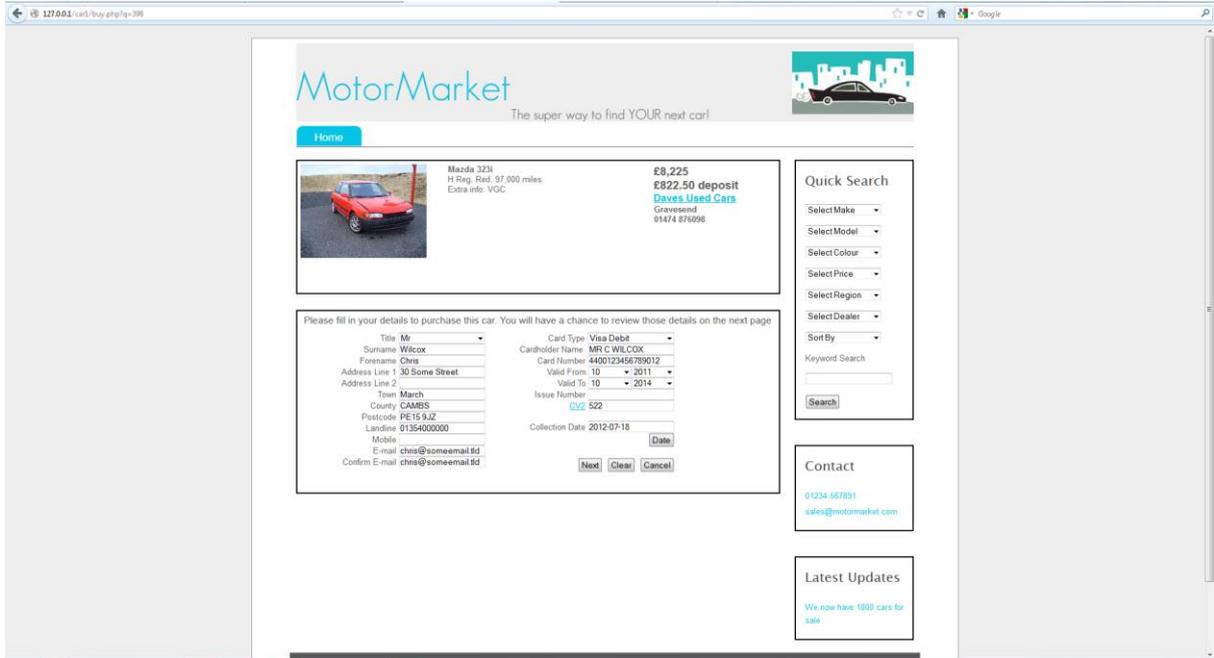
Car selected from search



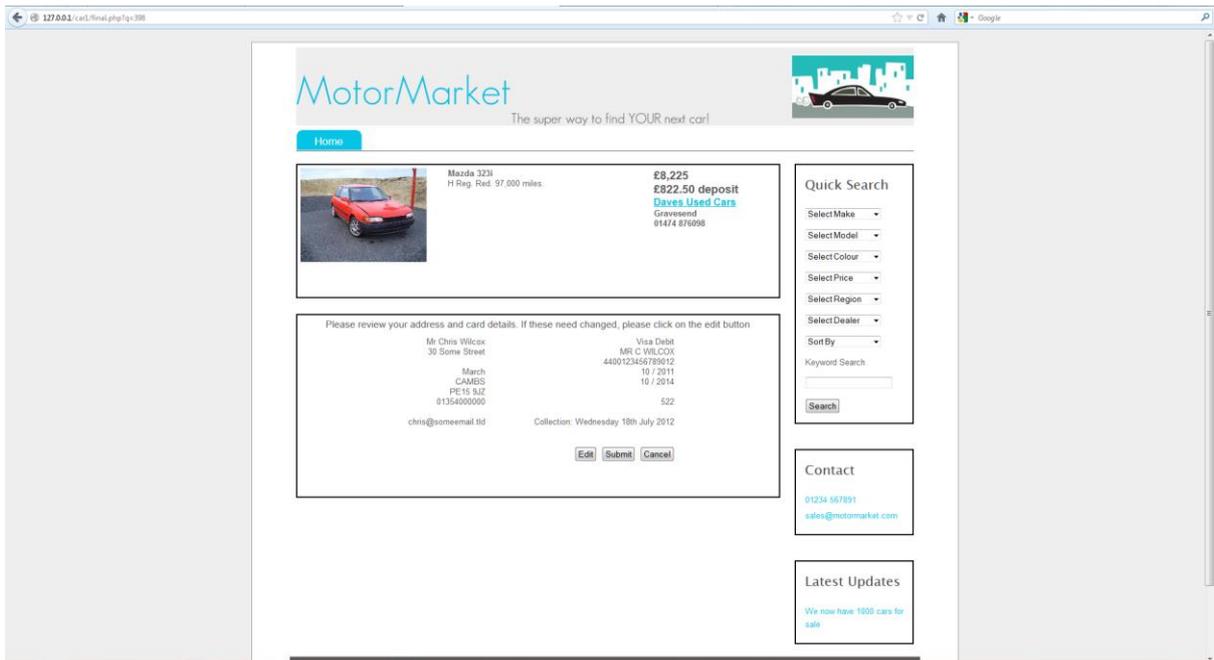
Send to friend selected



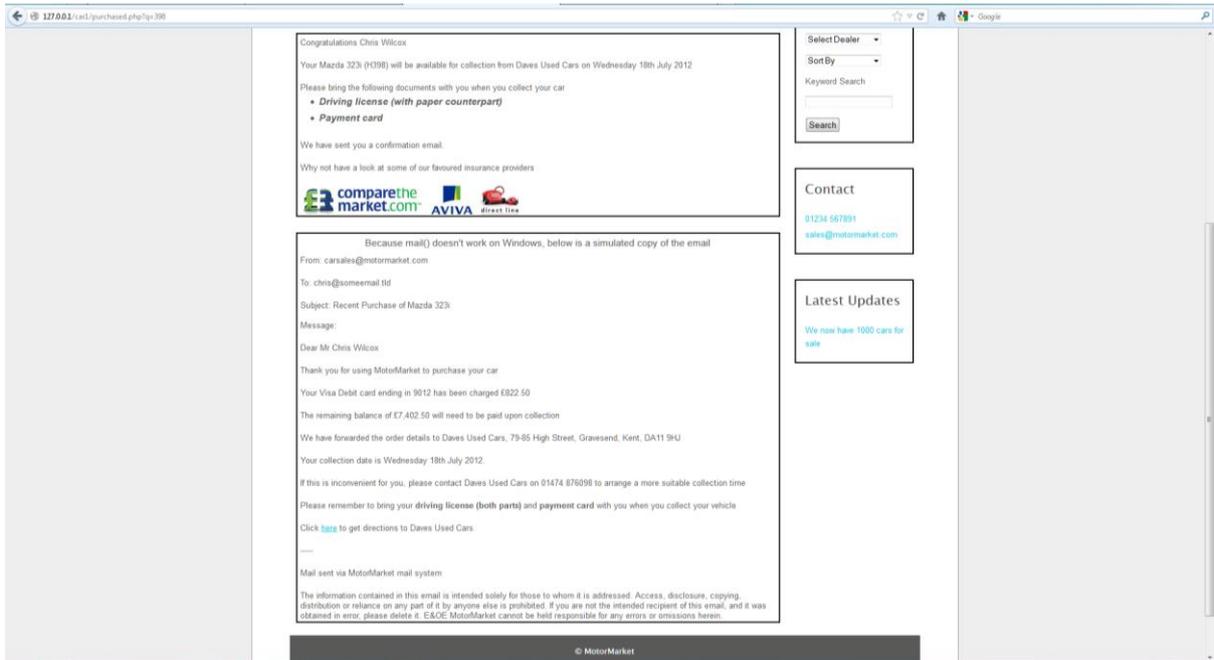
Results of sent mail to friend (mail() function doesn't work on Windows – this is simulated)



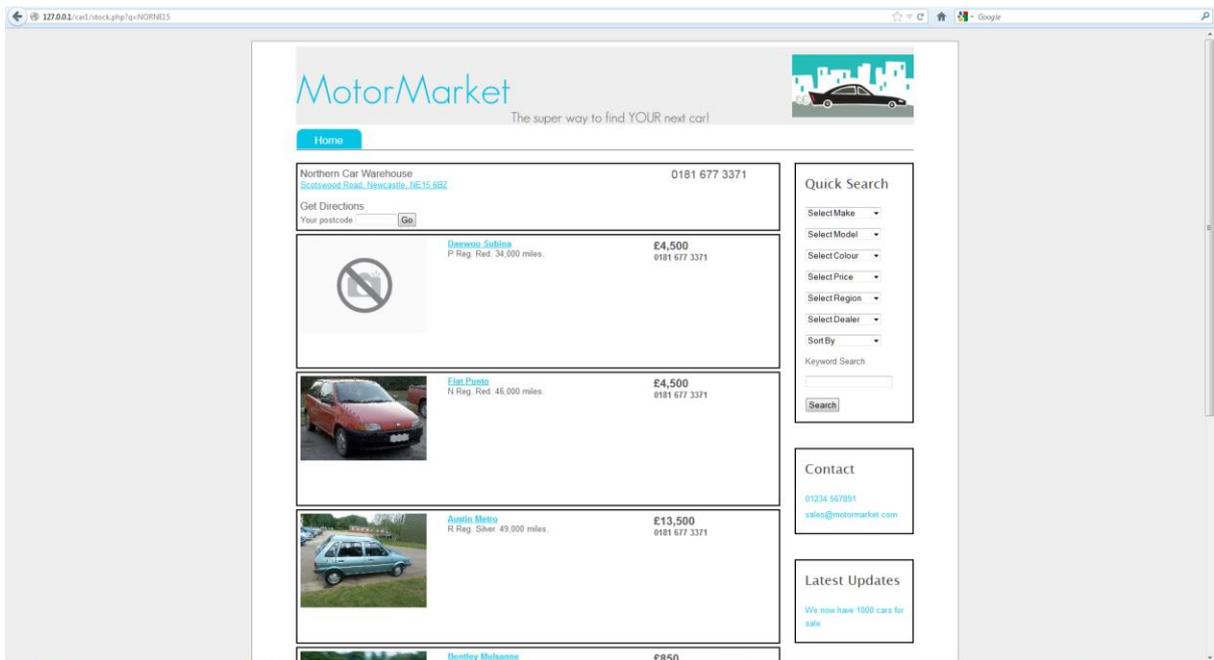
Purchase option selected



Reviewing of order before purchase

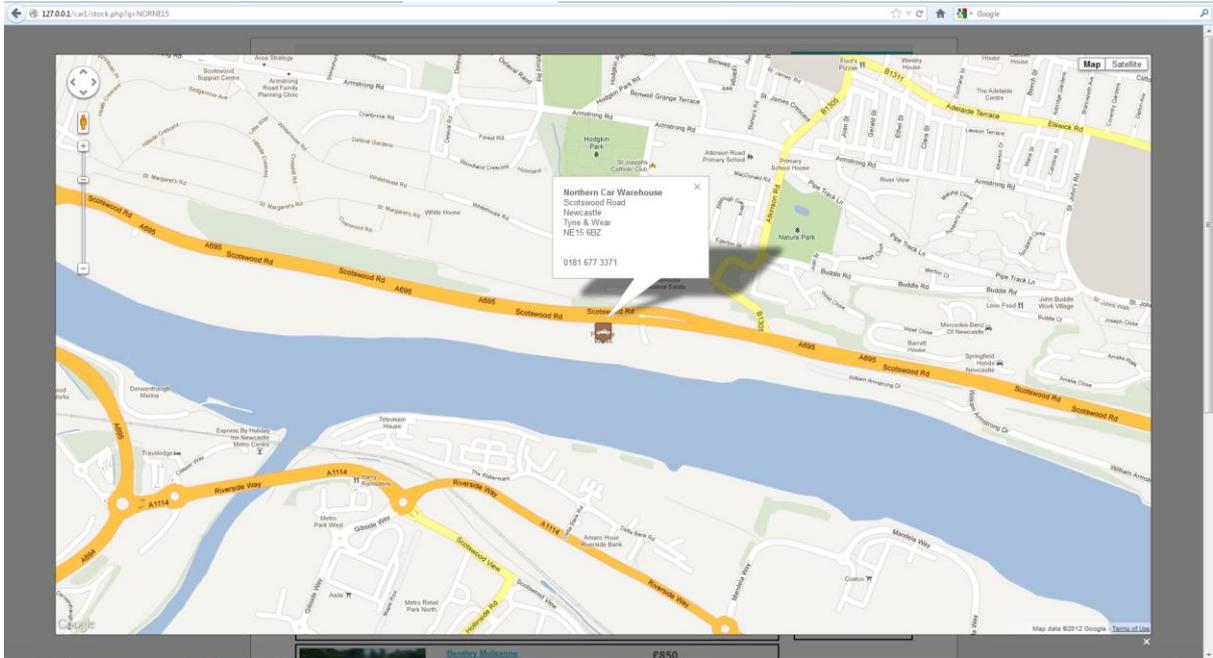


Purchase made, a message and simulated email to tell customer what they need to bring and who to contact if anything needs to change

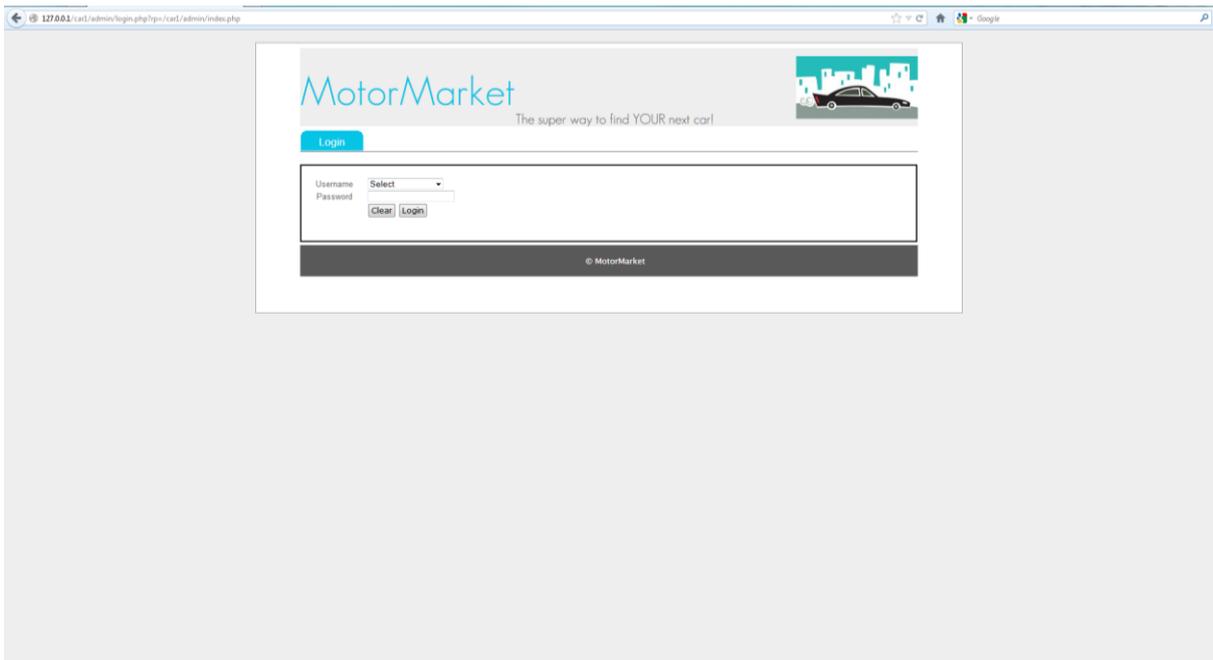


Showing the cars from a particular dealer

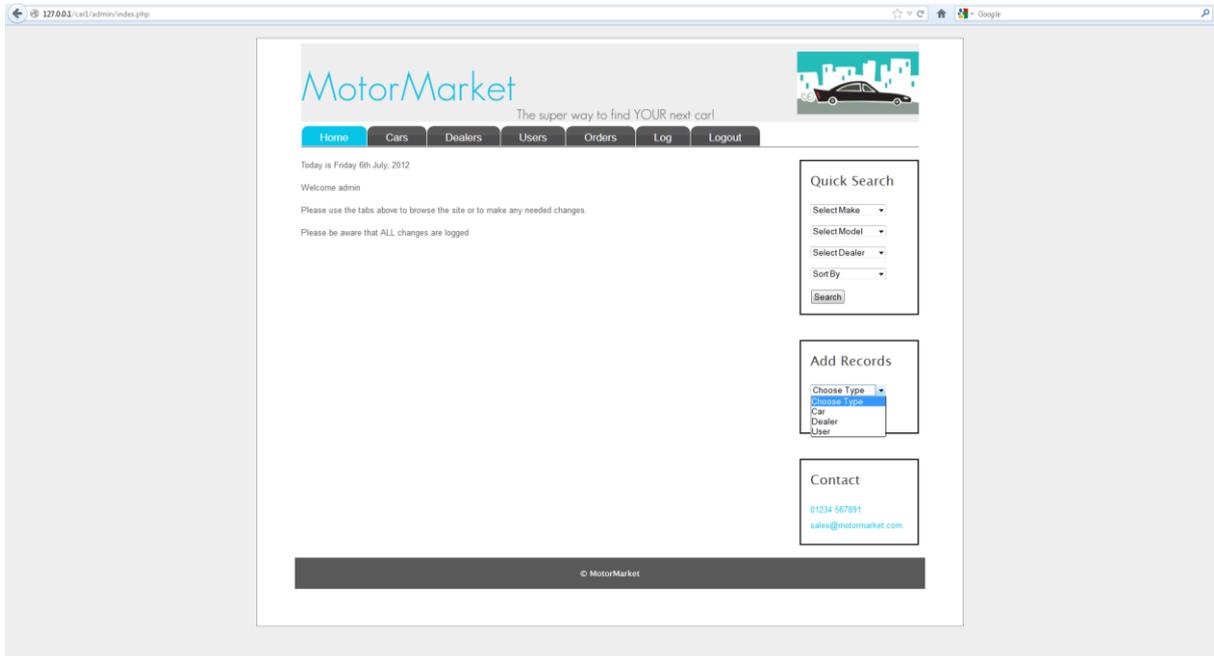
1068664



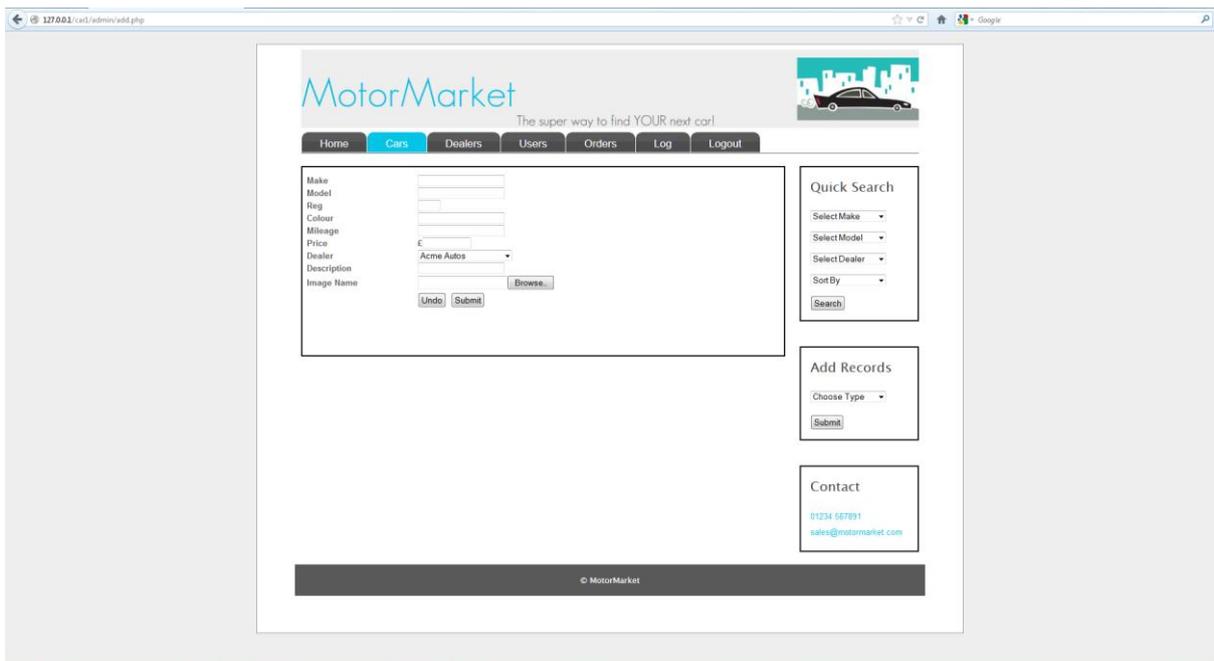
A map showing the location of the dealer



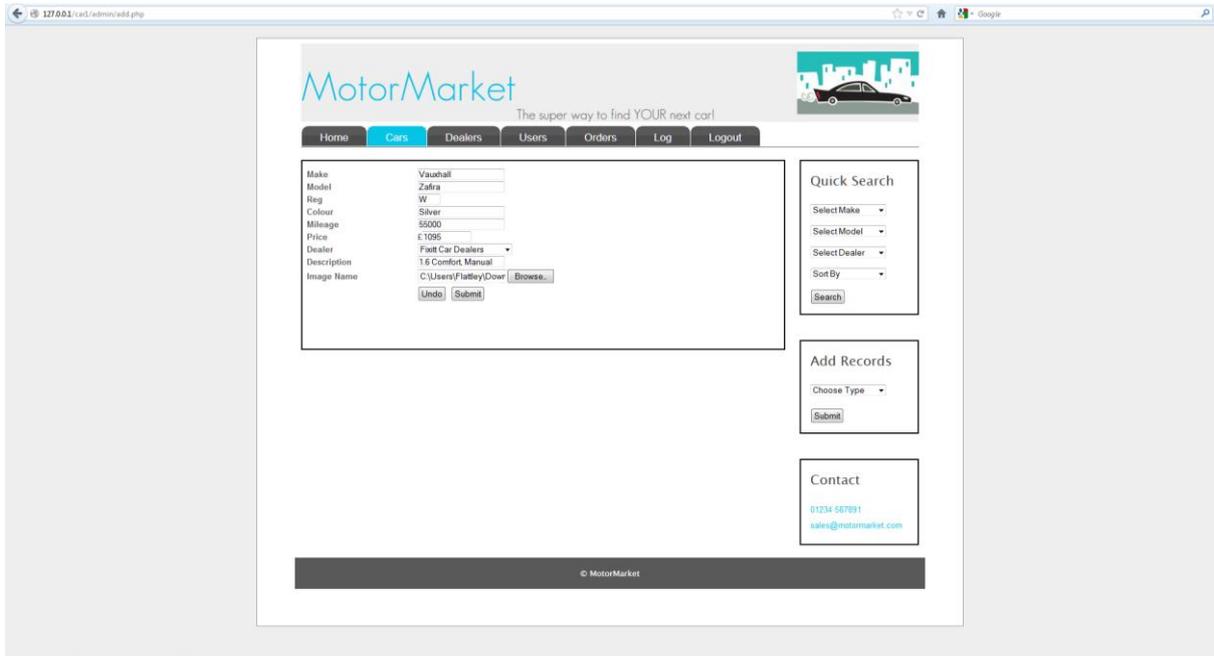
Admin login screen



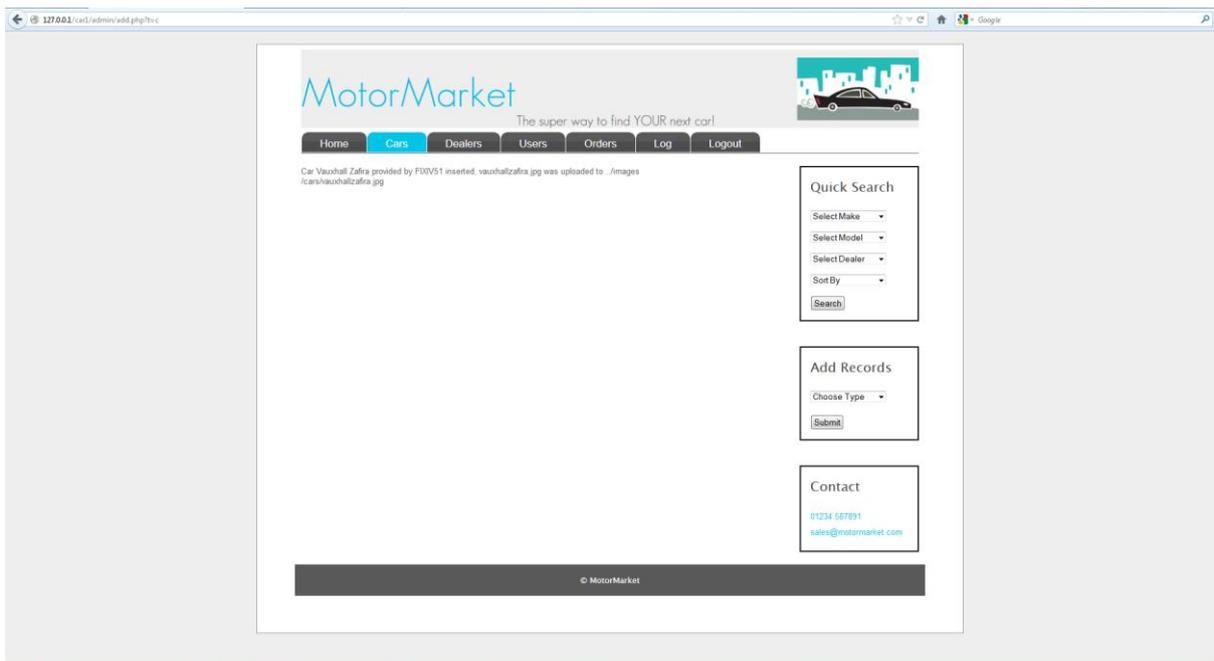
Admin homepage after logging in



Adding a record. In this case a car is being added

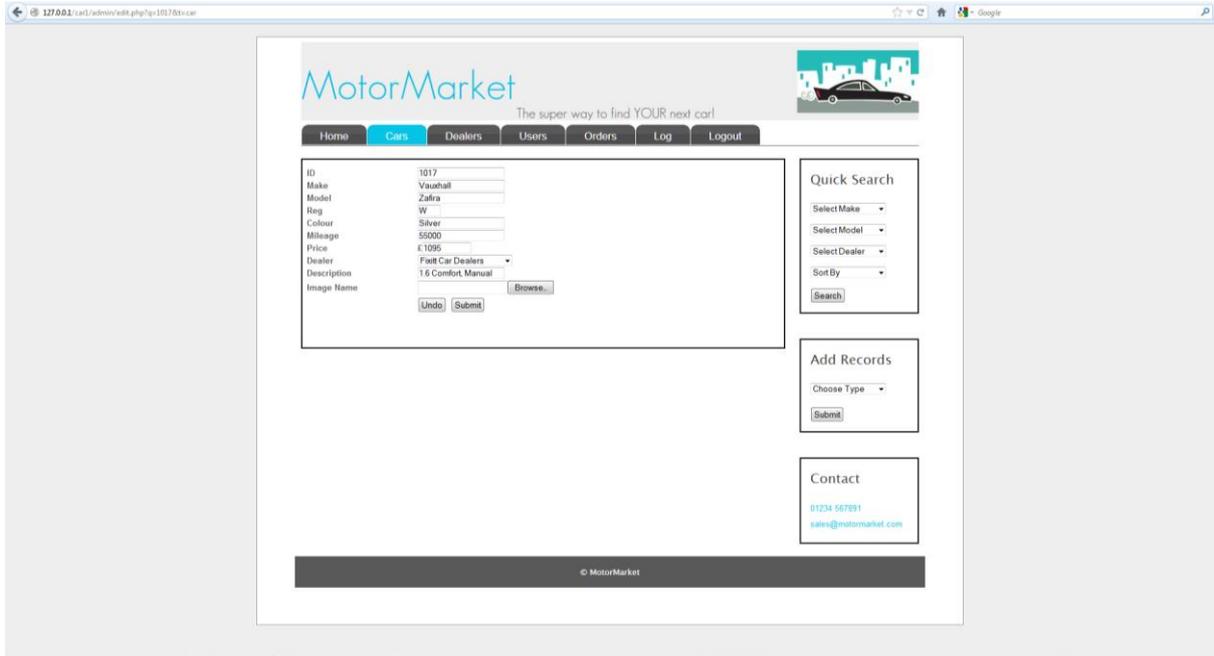


Fields filled in so the record can be added

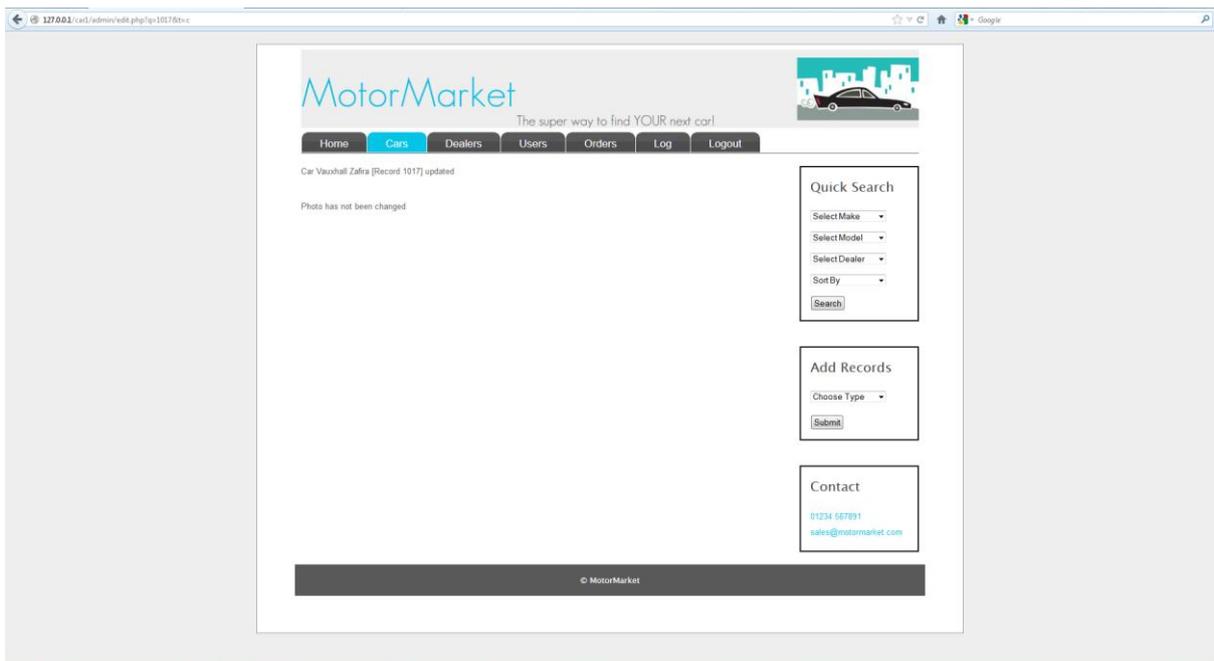


Confirmation that the record was added, and that the image uploaded

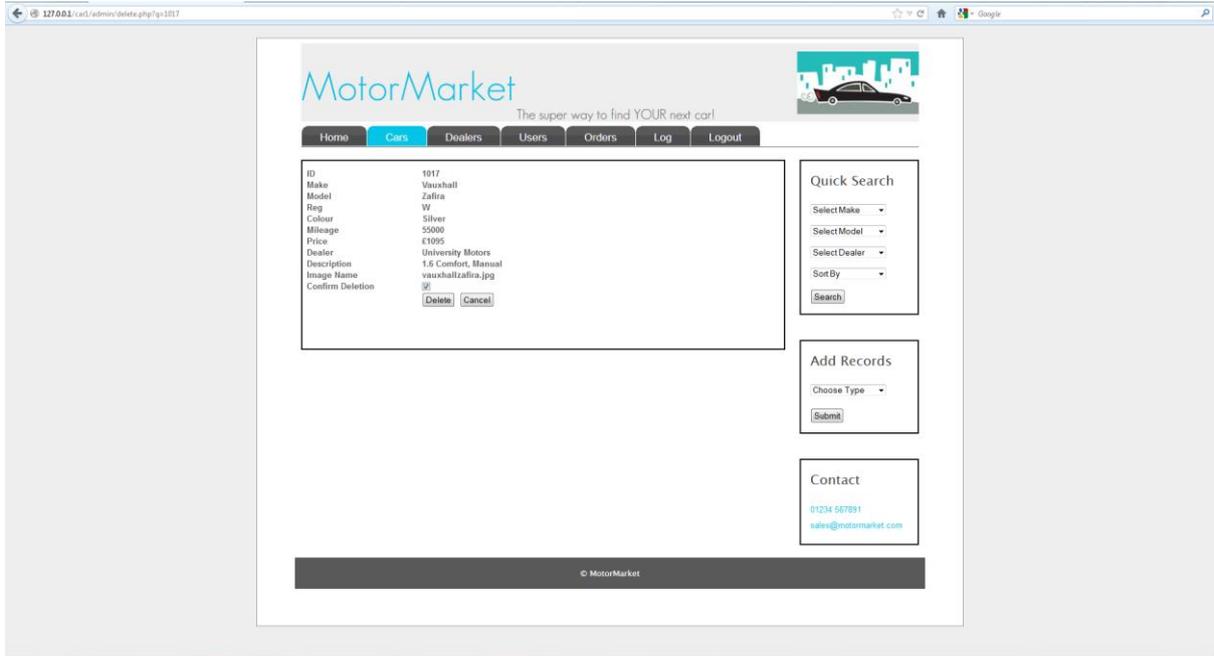
1068664



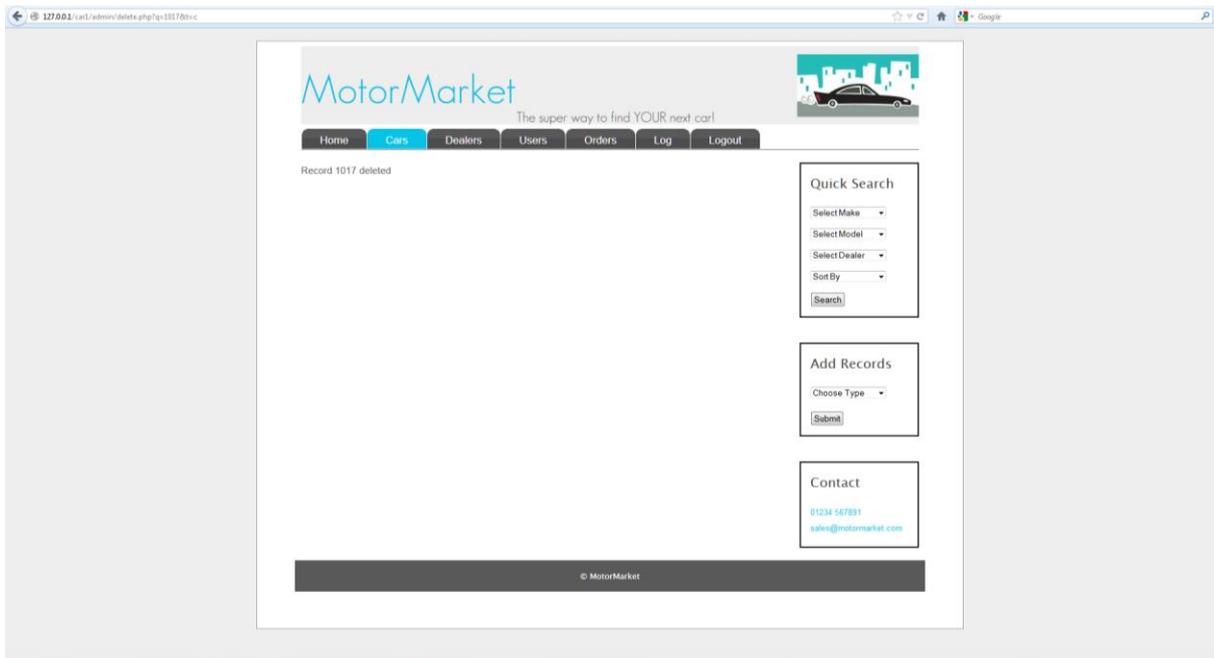
A record ready to be edited



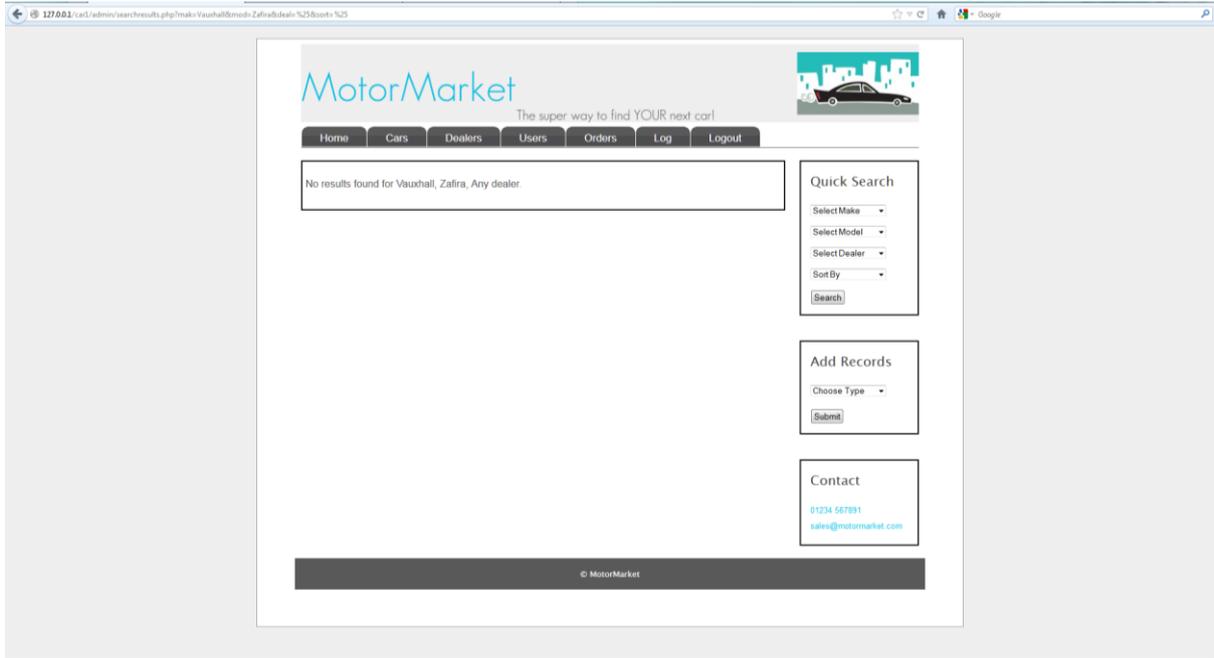
Confirmation that the record has been updated, without a photo change



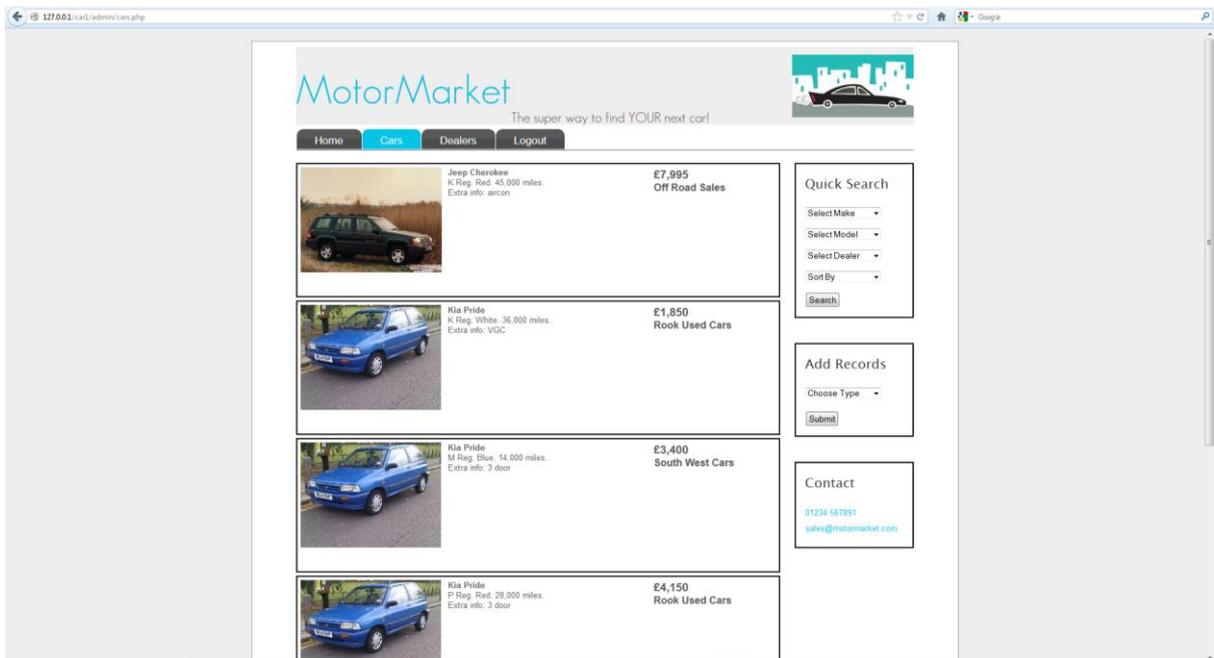
Record selected for deletion



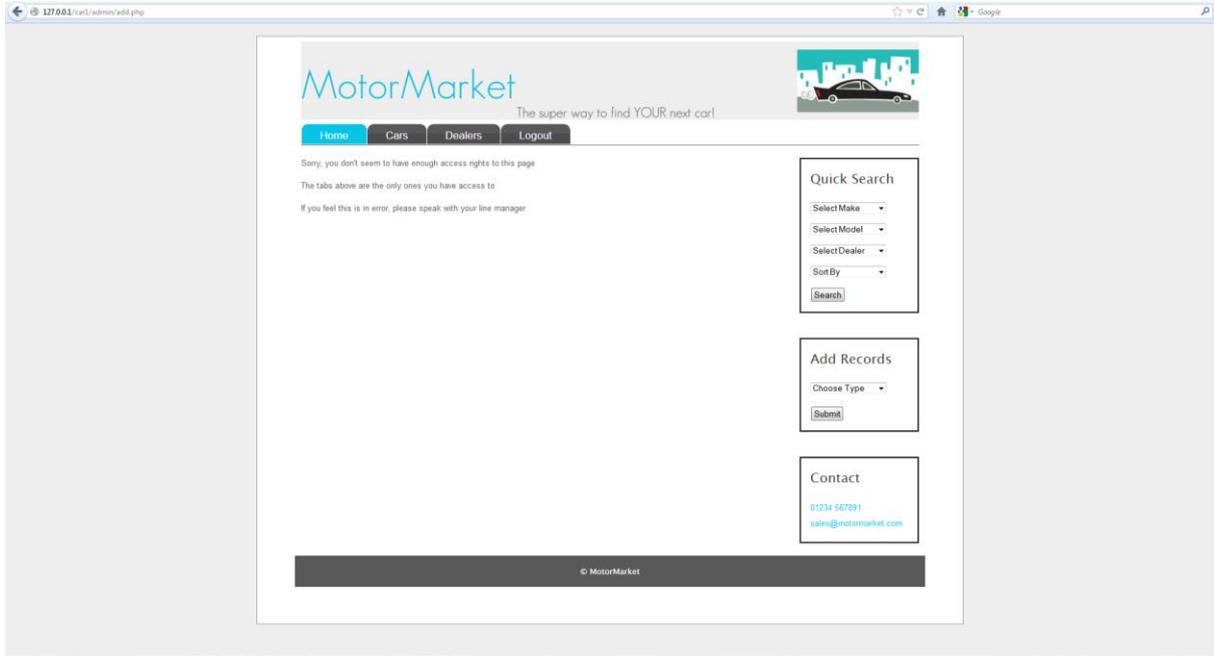
Confirmation of deletion



Car is no longer listed



Search results without admin options



Message displayed when a user doesn't have access to a certain page

BIBLIOGRAPHY

Anthony, 2011. *How to Use Colour to Successfully Brand Your Website*. [Online]
Available at: <http://uxmovement.com/content/how-to-use-color-to-brand-your-website/>
[Accessed 01 April 2012].

AutoTrader, n.d. *Used Car for Sale - AutoTrader UK*. [Online]
Available at: <http://www.autotrader.co.uk>
[Accessed 01 April 2012].

Bear, J. H., 2012. *Neutral Background Colors and Their Symbolism*. [Online]
Available at: http://desktoppub.about.com/cs/color/a/symbolism_5.htm
[Accessed 01 April 2012].

Certain Web Design, 2008. *Web Colour Design In Your Web Site*. [Online]
Available at: http://certainwebdesign.com/choosing_web_site_colors.htm
[Accessed 01 April 2012].

motors.co.uk, n.d. *Motors.co.uk - Used Cars For Sale - The UK's No1 Used Car Search*. [Online]
Available at: <http://www.motors.co.uk>
[Accessed 01 April 2012].

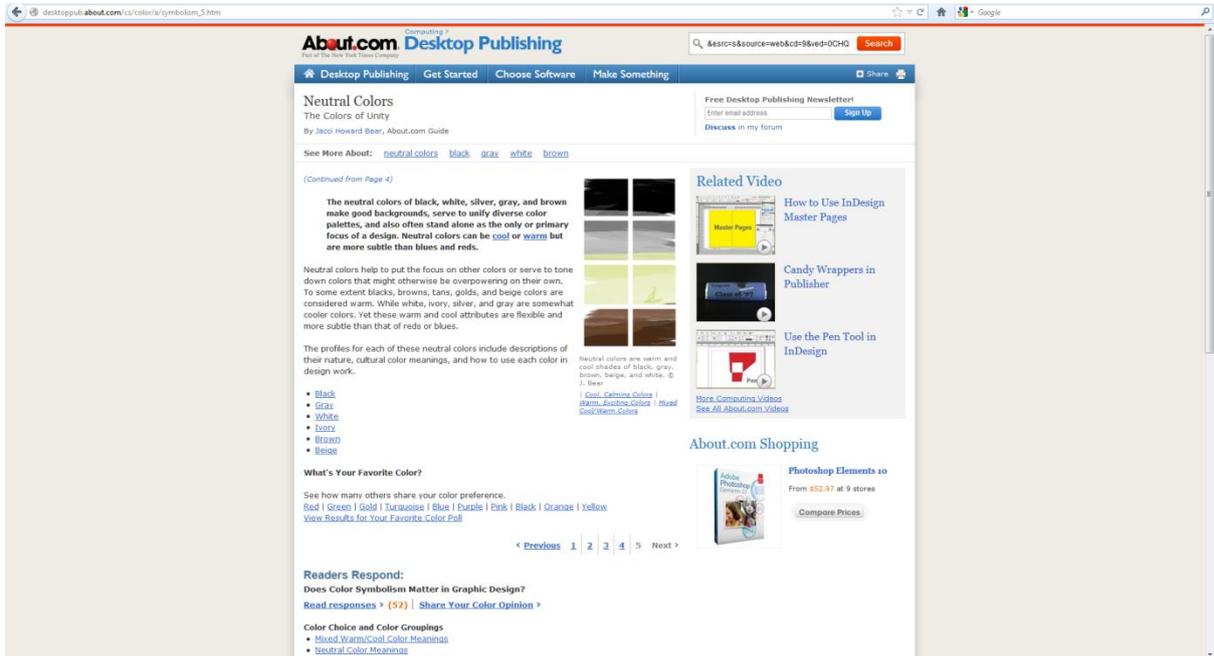
Noack, S., 2012. *A Look into Color Theory in Web Design*. [Online]
Available at: http://sixrevisions.com/web_design/a-look-into-color-theory-in-web-design/
[Accessed 01 April 2012].

Peterson, D., n.d. *Choosing Web Design Colours: Web Design Portfolio & Guide*. [Online]
Available at: <http://newark1.com/color/000055.html>
[Accessed 01 April 2012].

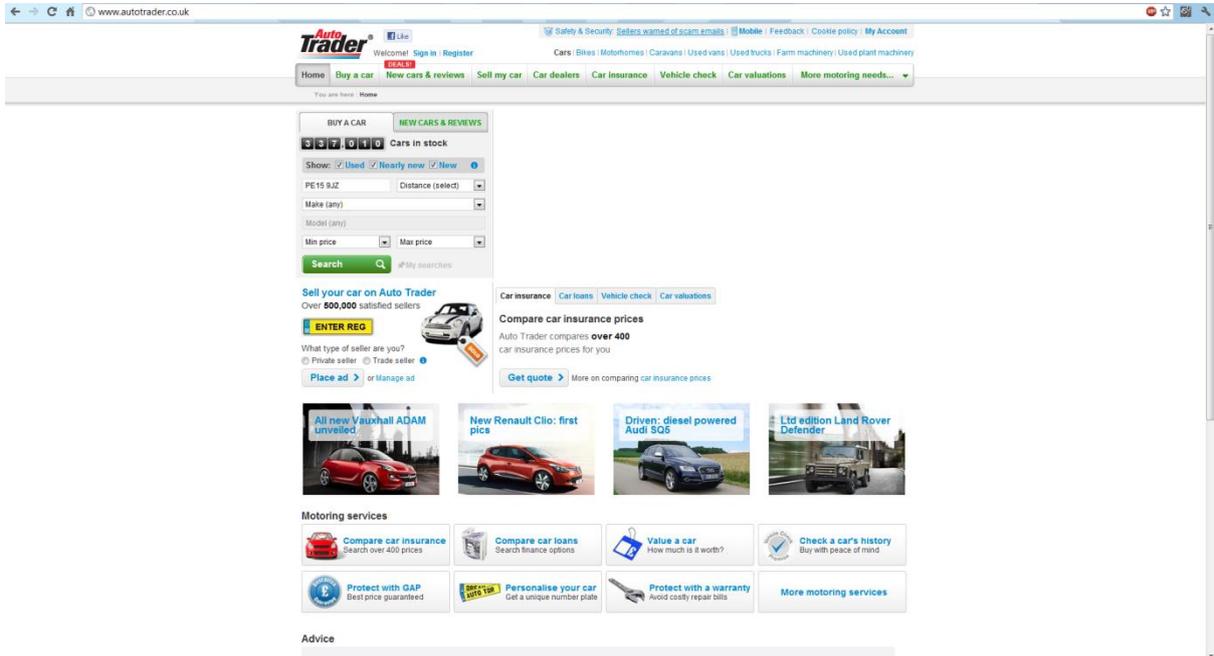
Smith, K., n.d. *Color: Meaning, Symbolism & Psychology*. [Online]
Available at: <http://www.squidoo.com/colorexper>
[Accessed 01 April 2012].

APPENDIX

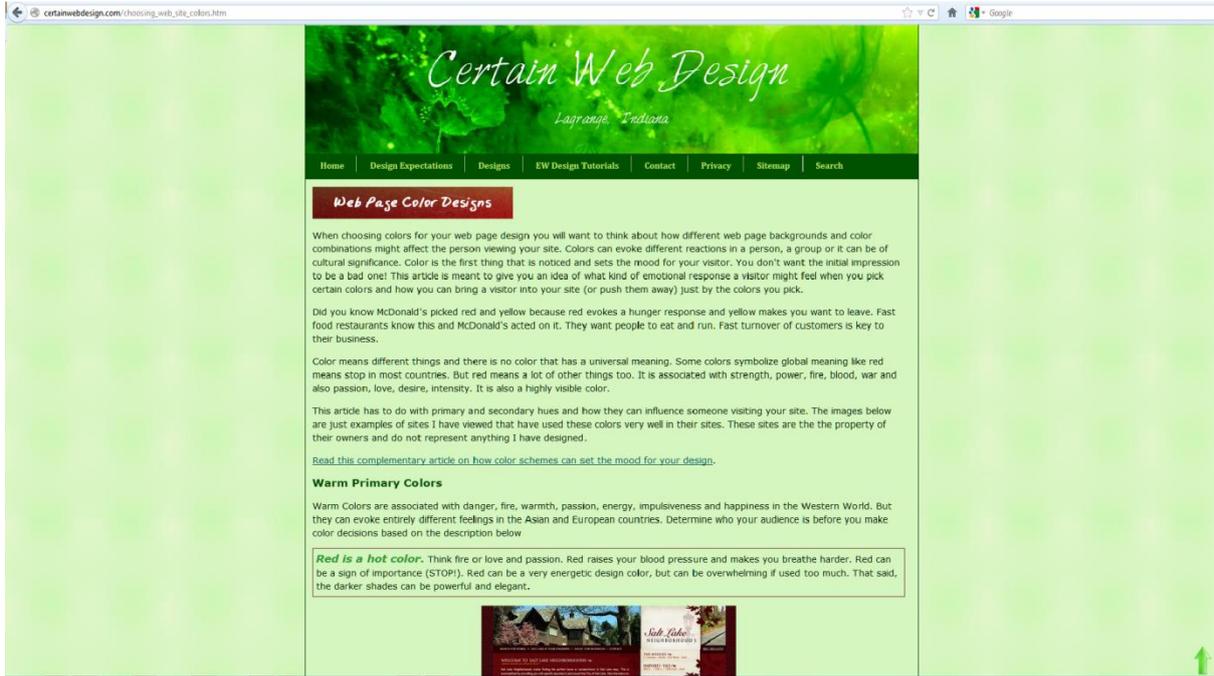
Visited Websites



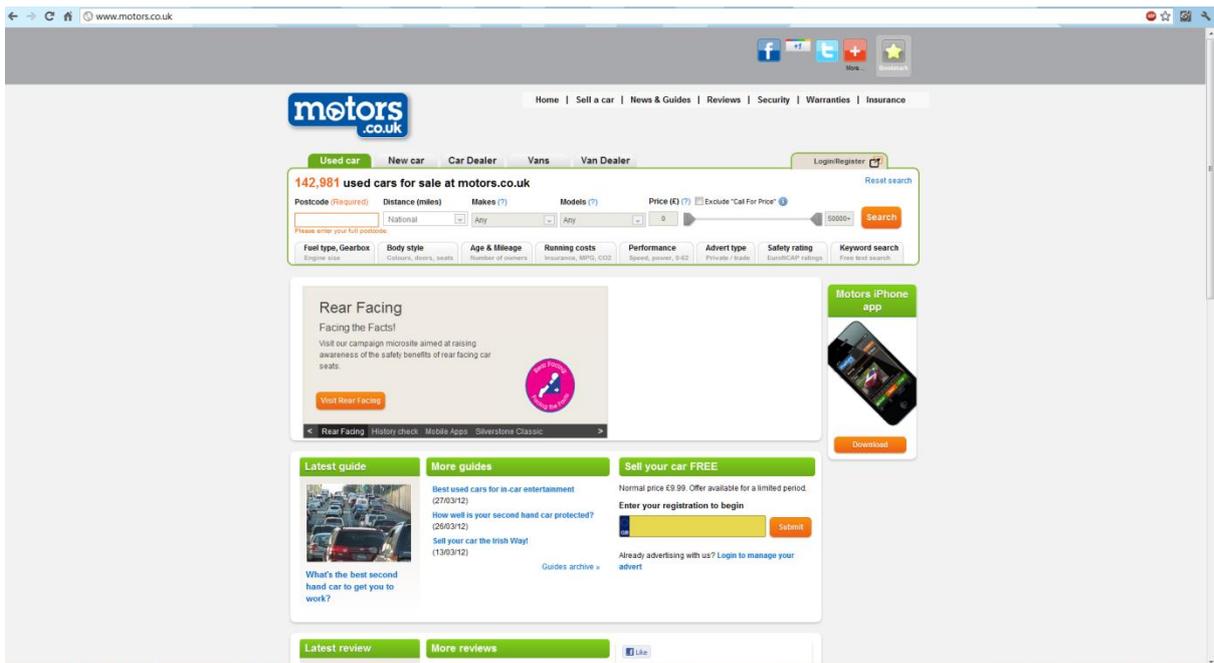
About.com



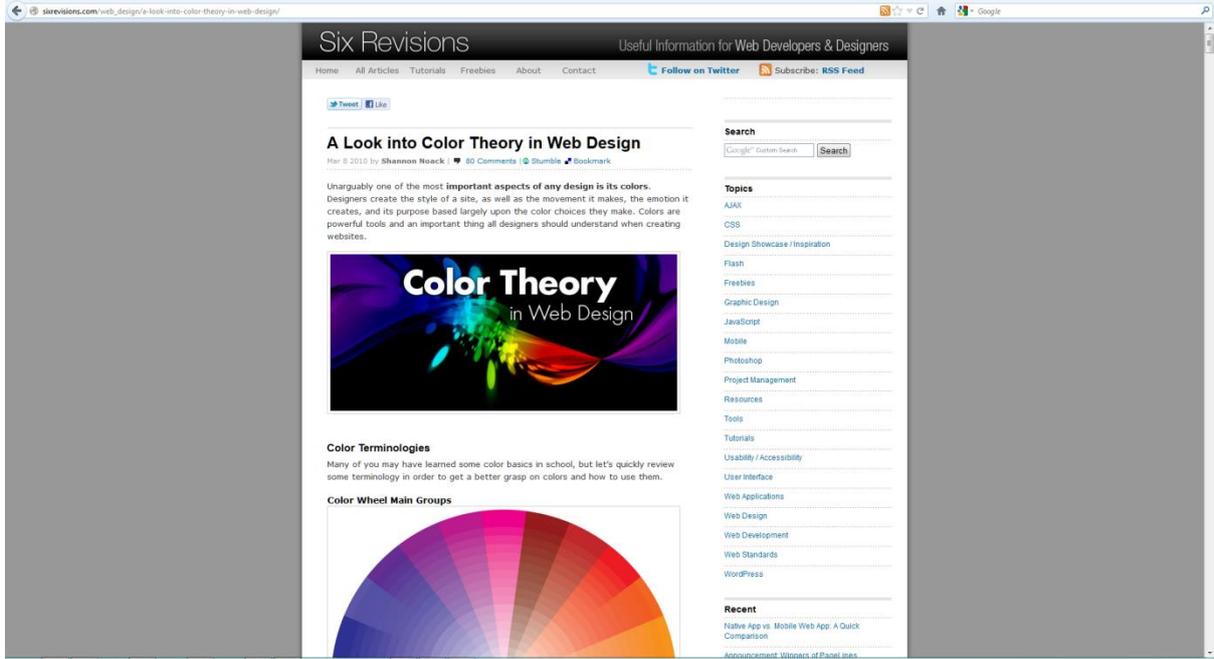
AutoTrader



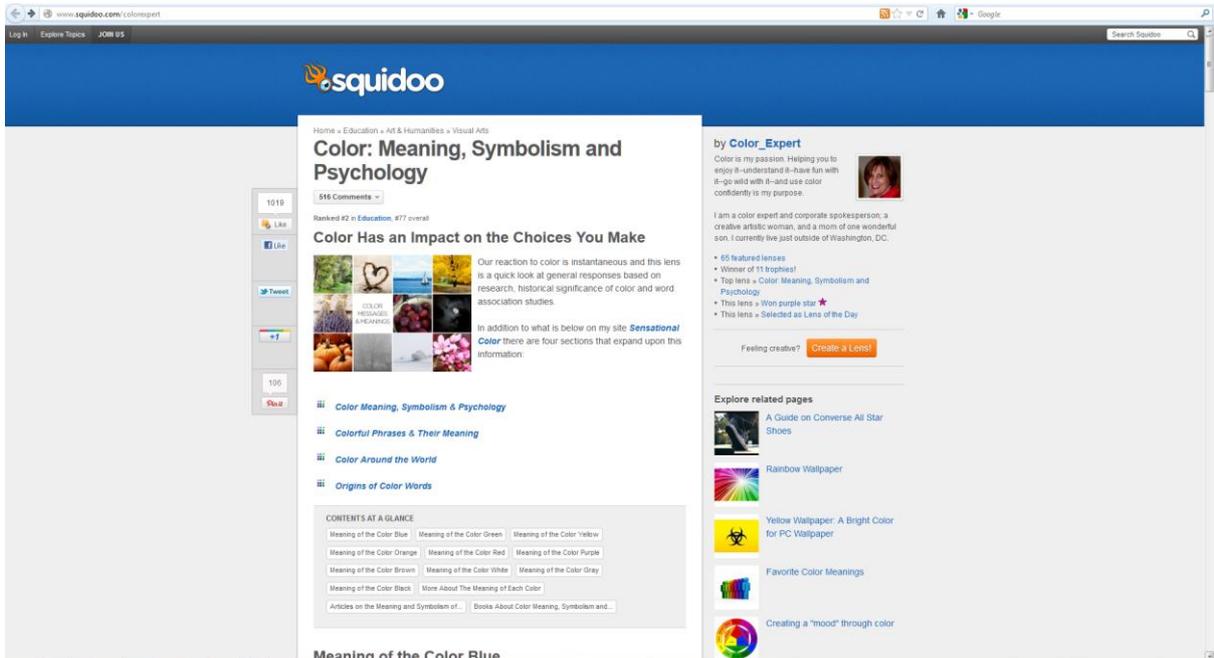
Certain Web Design



motors.co.uk



Six Revisions



Squidoo

Home About Twitter RSS Feed Contribute Contact

UXMovement

Search

Forms Navigation Buttons Content Wireframes Thinking Resources Giveaways Products

How to Use Color to Successfully Brand Your Website

by arthony on 01/10/11 at 3:57 pm

Have you ever met someone who had such a strong personality that when they entered a room they would grab your attention, and when they left they would leave an imprint in your memory? That's exactly how your users should react when they enter and leave your website. Your website should have a strong brand identity that stands out from everyone else's. When users leave your website, they should remember your brand. Using color properly on your website can help make this happen.

There are a few common mistakes most designers make with colors.

- o They pick inconsistent colors.
- o They use too many colors.
- o They apply colors to the wrong places.

This article will address each of these issues and offer guidance to help brand your website.

What colors to use

Logos are often underestimated. But when it comes to branding, logos are the most important thing. They set the tone for what colors you should use on your website. You should use the color of your logo on your website, so that when users visit your site, they'll feel like they're on the right site, and they'll feel secure about using your site. If the colors you use are inconsistent with your logo, it's likely users will not only feel like they're on the wrong site, but they'll also feel that your site is not professional. This affects their trust in your site. If users don't trust your site, there's no chance they're going to use it.

Color consistency makes users feel comfortable about using your website. It looks more professional when the color you use for your site is the color of your logo. However, that doesn't mean your site should only have one color tone. You should also vary the color saturation and brightness of your logo color to make your website look more dynamic. Combine that with the neutral colors of black and white and you'll be able to give your website a strong brand identity.

For example, look at the websites of H&R Block and Virgin America, two very successful brands. They use the color of their logo as the primary color on their website. However, to make it more dynamic they play with the saturation and brightness of that color. The color hue is still the same, but the color saturation and brightness vary throughout the site. They also use black and white for their background and text color. This makes their site easy to read because black text on a white background gives the text high color contrast. Black and white won't clash with your brand color because they're neutral colors. Your website should always have an equal balance of black, white and your brand color.

Comments Featured Tags

Tracey: Evan, are you really saying that your considered f...

NK: True, though you could just as easily make it big...

Ian: Fluid UI is now launched and live: https://www.flu...

Blessed Geek: We should call a spade a spade, and a click a click.

ReaderX: I hope folks understand what's being proposed here...

UXMovement

newark1.com/color/000055.html

WEB DESIGN: WEB DESIGN PORTFOLIO MORE WEB DESIGNS WEB DESIGNS FOR SALE

WEB DESIGNERS GUIDE: WEB DESIGN COLOR WEB DESIGN ORDER WEB DESIGN FORM

WEB DESIGN TIPS: WEB DESIGN BLOG MY PHOTOGRAPHY CONTACT ME HOME PAGE

FOLLOW ME

CALL US AT 1 917-267-9281 for a free quote

WEB DESIGN COLOR

Recommend This

How To: Color In Web Design Color Fundamentals & Design Green Web Design Find Great Color Palettes! Does Color Really Matter? Web Design Color Resources Color Restraint in Web Design Add Texture to Your Color Clean, Simple & Fresh Set the Mood with Color Unify Your Web Design Colors! Add Light To Your Web Design Great Tree Color Web Design Great Web Design Color Sells! Choosing Web Design Colors

CHOOSING WEB DESIGN COLORS

When designing a Web site, most web designers give a lot of attention to two major factors: the design (choice of graphics) and the content. But what about the colors? This is one of the most important things to consider when designing your site. The colors of your Web site are important because they will define the mood and emotions of your visitors as well as reflect your identity and image branding.

Color is symbolism. We say someone who is jealous is "Green with envy". Someone who is sad is said to be "Blue." When we are angry, we see red." Colors affect us psychologically. They can stimulate our appetite or suppress it. They can make us happy, excited, angry or sad. Let's take a look at colors and what they mean to your visitors. Then we'll discover the different color combinations and how they apply to your Web site.

Warm Colors

Red: Red is one of the most powerful and attention-getting colors. Red symbolizes passion and fire, love and lust. It has been proven to raise blood pressure and cause perspiration. Red reflects energy and can motivate an individual to take action. It can also symbolize war, violence, blood, and aggression.

Pink, however, is the softer side of red. Pink is romantic, calming and feminine.

Orange: Vibrant and warm, orange is associated with autumn, pumpkins and Halloween. It inherits many of the traits of red, but is less passionate. Orange has been known to stimulate the appetite. As a citrus color, it can also symbolize health.

Yellow: Yellow symbolizes sunshine and warmth. It can mean hope, light and energy. On the other hand, it can mean weakness or illness, or symbolize cowardice.

Cool Colors

Green: Green signifies health and growth (vegetation) and wealth (money). Green symbolizes spring, renewal, and fertility. It can also mean jealousy or inexperience.

Blue: Blue is one of the most calming colors and is associated with the

Web Design Color