

CHALMERS



GARAGESALE



Garage Sale

An E-Commerce Solution

Master of Science Thesis [in the Programme Software Engineering & Technology]

MAHBOOB HUSSAIN KHOKHAR

Chalmers University of Technology
University of Gothenburg
Department of Computer Science and Engineering
Göteborg, Sweden, November 2010

The Author grants to Chalmers University of Technology and University of Gothenburg the non-exclusive right to publish the Work electronically and in a non-commercial purpose make it accessible on the Internet.

The Author warrants that he/she is the author to the Work, and warrants that the Work does not contain text, pictures or other material that violates copyright law.

The Author shall, when transferring the rights of the Work to a third party (for example a publisher or a company), acknowledge the third party about this agreement. If the Author has signed a copyright agreement with a third party regarding the Work, the Author warrants hereby that he/she has obtained any necessary permission from this third party to let Chalmers University of Technology and University of Gothenburg store the Work electronically and make it accessible on the Internet.

Online Garage Sale System

MAHBOOB HUSSAIN KHOKHAR

© MAHBOOB HUSSAIN KHOKHAR, November 2010.

Examiner: PER ZARING

Chalmers University of Technology
University of Gothenburg
Department of Computer Science and Engineering
SE-412 96 Göteborg
Sweden
Telephone + 46 (0)31-772 1000

Garage Sale is a web application which is developed to sale and purchase the garage for cars.

Department of Computer Science and Engineering
Göteborg, Sweden November 2010

Garage Sale

An E-Commerce Solution

Thesis Report

Table of Contents

Acknowledgements

Abstract

1. INTRODUCTION:	1
1.1. BACKGROUND:	1
1.2. PROBLEM STATEMENT:	2
1.3. PURPOSE:	2
1.4. DEMARCATIONS:	2
2. PREVIOUS WORK:	4
2.1. E-COMMERCE SOLUTIONS:	4
2.2. GARAGESALE-A C2C SOLUTION:	6
2.3. USABILITY:	8
2.4. TESTING:	8
2.4.1. <i>Black box testing:</i>	8
2.4.2. <i>White box testing:</i>	8
2.4.3. <i>Security testing:</i>	9
3. BUILDING A GARAGE SALE:	10
3.1. TEST CASES OF BLACK-BOX TESTING:	10
3.1.1. <i>Home Page:</i>	10
3.1.2. <i>Search Engine:</i>	11
3.1.2.1. Search Engine scenario 1:	11
3.1.2.2. Search Engine scenario 2:	12
3.1.2.3. Search Engine scenario 3:	12
3.1.2.4. Search Engine garage detail page:	13
3.1.3. <i>Lägg till annons:</i>	14
3.1.3.1. Lägg till annons Steg 1:	14
3.1.3.2. Lägg till annons Steg 2 image uploading:	15
3.1.3.3. Lägg till annons Steg 3 Ta bort:	17
3.1.3.4. Lägg till annons stages navigation problem:	19
3.1.3.5. Lägg till annons email format:	19
3.1.3.6. Lägg till annons display of items pictures:	20
3.1.3.7. Lägg till annons verification email:	21
3.2. TEST CASES OF WHITE-BOX TESTING:	22
3.2.1. <i>Search Engine:</i>	22
3.2.2. <i>Database table status field:</i>	23
3.2.3. <i>Database table flag field:</i>	23
3.2.4. <i>Verification Email Problem:</i>	24
3.3. TEST CASES OF SECURITY-TESTING:	25
3.3.1. <i>SQL Injection attack:</i>	25
3.3.2. <i>XSS (Cross site scripting) attack:</i>	26
3.4. NEW FUNCTIONALITY:	26
3.4.1. <i>Edit Garage:</i>	27
3.4.1.1 Edit Garage – Login:	27

3.4.1.2 Edit Garage – Steg 1:	28
3.4.1.3 Edit Garage – Steg 2:	28
3.4.1.4 Edit Garage – Steg 3:	30
3.4.2. Delete Garage Advertisement:	31
3.4.3. Suggestion list of search engine:.....	32
3.4.3.1. Suggestion list of search engine-Vad:.....	32
3.4.3.2. Suggestion list of search engine-Var:	33
3.4.4. Cron Jobs:.....	34
3.4.4.1. Cron Jobs Setting:.....	35
3.4.5. Refinement of search engine:	37
3.4.6. User friendly URLs:	37
3.5. POSSIBLE FUTURE WORK SUGGESTIONS:	38
3.5.1. Super Administration:	38
3.5.2. Human verification code:.....	38
3.5.3. Keep tracking for visitors:	38
3.5.4. Keep tracking for advertisers:	39
3.5.5. Advance Search:	39
3.5.6. Pagination:	39
3.5.7. Send Email to seller:	39
3.5.8. Send SMS to seller:	40
3.5.9. Tell to friends:.....	40
3.5.10. Share garage on facebook:	40
3.5.11. Report as abuse:	40
3.5.12. Forget password:	41
3.5.13. Auto email for expired garage:	41
4. RESULT:	42
5. CONCLUSION:.....	44
REFERENCE LIST:	46

Acknowledgement

This document is a master thesis report about Online Garage Sale System written for Chalmers University of Technology, Sweden. This master thesis was conducted by Mahboob Hussain Khokhar at Department of Computer Science and Engineering of Chalmers University of Technology. The examiner and supervisor of this thesis was Per Zaring. This thesis was the idea of a company Phoenix located in Gothenburg, Sweden. Work on this project has been done with close coordination with them.

I am very thankful to my supervisor Per Zaring who helped me a lot throughout to complete this work and give me direction to writing report. I am also thankful to Iradj Pakkhoo and Daniel Albinsson from Phoenix to give me this opportunity.

Abstract

Garage Sale is a web application which is a product of Phoenix1¹. The main idea of this thesis is to develop a web application that provides different features e.g. sale and purchase garages for cars. Sellers can post their advertisement on website with all specifications about garage and detail information of items of garage for example vacuum cleaner, stool, pulley, wheel etc. Moreover, buyers can look advertisement about garage, look items of garage and find specific garages. Search engine is used to browse specific garages on the base of search criteria such as item name and zip code. Target market of this idea is very specific because users of this website are those who have vehicles and/or garage.

In the initial stages of thesis work some testing has been done on previously developed application and refinement of existing features to make it userfriendly. After that some new modules have been developed to extend the usability and scope of the application. Moreover this document covers the conceptual model, test cases, use cases, features and screen analysis. The Implementation work has been done in PHP², Codeigniter³ to follow MVC⁴ design pattern.

¹ Name of company

² Server-side HTML embedded scripting language

³ PHP Framework's name

⁴ Model View Controller

1. Introduction:

Introduction of thesis is described below in detail which contains background, purpose, problem statement and demarcations.

1.1. Background:

In recent years there has been huge advancement in the field of information technology. E-commerce and e-solutions have been grown rapidly. Web technology is the most powerful and quickest way to find any kind of information. Concept of online shopping was a revolution in the world of e-commerce. Online shopping is now a common trend, almost every internet user knows about it and wants to use this timesaving way for online shopping. There are lots of e-commerce solutions such as EBAY, BLOCKET etc, where sellers publishing their announcements with detail i.e. price, model, pictures etc. Buyer can have look on specifications of announcements at description pages of announcement. Search engines have been introduced to find quickest information of desired entity, where buyer can find things by using different search criteria such as title, category, price and location etc.

Target market performs important role in any business especially for e-commerce business. Different e-commerce solutions have been designed for different purposes. It is important the business idea for developing a new website must be targeting to need of customers.

This report has been written to enlightened the business idea which is a web application named as garage sale. Garage sale system is a combination of traditional shopping and e-commerce. This system is bound yet within the Sweden. By using this system user may easily find the physical location of garage. User may search garages those are nearest to their residence, by giving zip code in search criteria of search engine. User may visit physical location of garage after browsing contact and geographical information on the website.

1.2. Problem Statement:

This business idea is targeting only people those have cars and/or garage of cars. These people either could buy or sell their garages for cars. Initial work has already been done on this idea. To proceed further, testing of website was required. Aim of testing was maximum facilitating targeted customers and they can purchase/sale garages by getting/posting information about garages on website without any trouble. It was necessary that all information about a specific garage must have to proceed under some administrations where seller can modify its own data, and only useful information which will show on website to visitors.

1.3. Purpose:

The objective of online garage sale is to discuss the technical potential of creating a centralized system where electronic communication and digital information processing could be possible among the targeted customers in all over the Sweden. In the future there will be some registration fee for sellers and sellers will also pay for uploading pictures as per picture of items. Purpose of the report is as follows:

- Analysis of idea on the base of e-commerce solutions
- Studying the already developed system
- Testing of existence system
- Designing and implementation of the prototype

1.4. Demarcations:

- Online payment system is a typical functionality of e-commerce systems. But payment system is not integrated yet with garage sale system, because according to business plan this application is free of cost for all in the beginning.
- Website URL name is not yet finalized such that `www.example.com` this could be consider as extension of future work.

- Load testing has not been performed, because it depends upon the performance of online server which is not provided yet from company.
- SEO (Search Engine Optimization) has not been applied yet. SEO is totally dependent on domain name. Website contents have to be online to perform SEO.

2. Previous Work:

Previously done work on this project is mentioned in this chapter.

2.1. E-commerce Solutions:

E-commerce is the trade which has been using the electronic communication and digital information processing technology to create business relationships. E-commerce is also using to make communication easy between organizations and between organization and individuals. E-commerce has lot of advantages for example consumer can get access or can buy the goods from all over the world, even by sitting at home. It is also easy for business as well, business can order to their supplier by using their websites. Much cost of business has been reduced due to e-commerce. Before e-commerce business was restricted within the specific range or area. It was much costly to set up offices in different areas [2]. E-Commerce solves these issues by providing online market. Some major types of e-commerce are explained in **Fig.1** below.

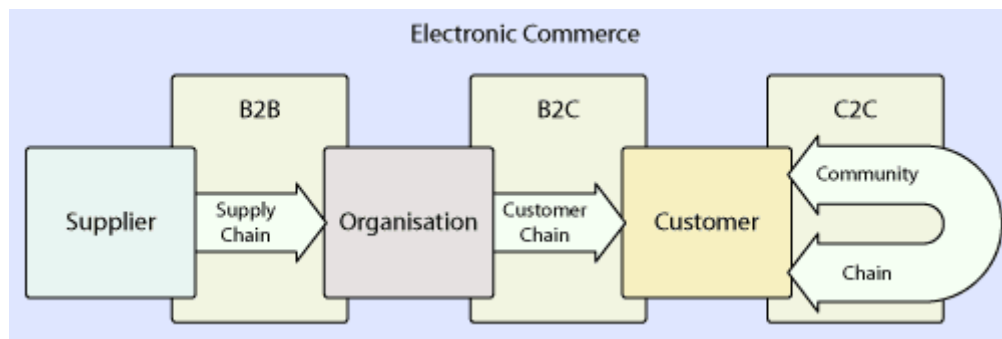


Fig.1

A clear explanation and distinction between e-commerce types is given in **Fig.1**. This diagram has been taken from a forum of e-commerce [3].

Some major types of e-commerce are mentioned below with brief explanation.

- Business-to-Business (B2B)
- Business-to-Consumer (B2C)
- Consumer-to- Business (C2B)
- Consumer-to-Consumer (C2C)

B2B is the e-commerce between companies. Companies deals with relationships among business. Some examples of B2B are logistic-exportation, application service providers and Cisco etc. Another example of B2B e-commerce is Ford and General Motors (GM), they buy and sell supply chain-related auto products.

B2C is the e-commerce between business and consumers. They selling and purchasing things on internet. Examples of B2C e-commerce are Amazon, Dell etc. These companies are selling goods like books, laptops to their consumer via internet.

C2B e-commerce is the reverse auction, which is between consumers to business. In C2B consumers offering services to companies and getting pay from company. Trading model of some companies considering C2B for example Google-AdSense. Individuals can display Google-AdSense i.e. banners, promotional items and advertisements on their personal websites. To provide advertise or selling service, individual are directly commissioned by companies. Price-line is also a kind of C2B, where airline tickets booking and hotel reservation system could be done online.

C2C is the e-commerce between consumers to consumer. Individuals are doing business in online environment in C2C e-commerce. There are several types of C2C e-commerce. Consumers can communicate face-to-face with each other's i.e. Blocket in Sweden. Consumers also can made online payment transactions in case of long distance. It is a common case for example in EBay, seller living in Hong Kong and buyer wants to get

product at home from anywhere in the world. **Fig.2** below describes the major types of e-commerce with examples.

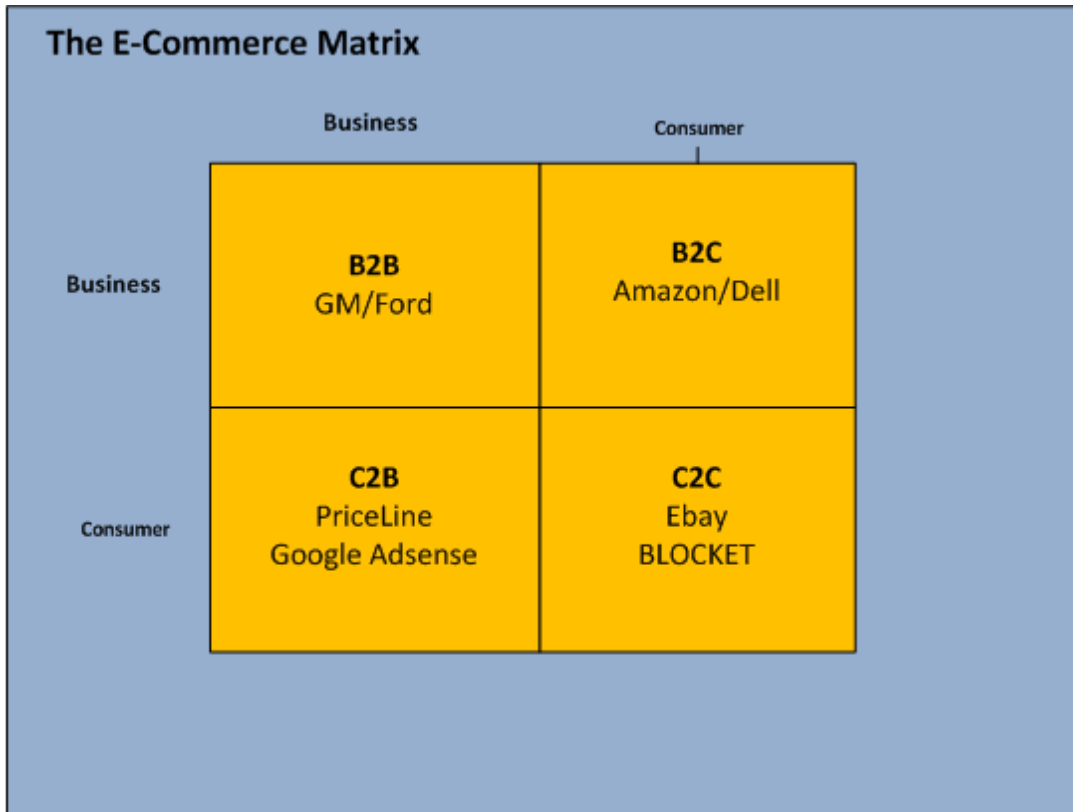


Fig.2

2.2. GarageSale-A C2C Solution:

Garage-sale system has been implemented to follow the C2C e-commerce. In **Fig.3** whole structure and working flow of the system has been mentioned. Seller may create new garage announcement on website. Seller may also modify contents of existing garage announcement. Buyer can use website to get information about garages those are going to sell. Buyer can search garage by using search engine. Search engine will takes two kinds of inputs into search criteria fields, zip code and garage item name. Buyer may contact directly to seller by reading contact information on announcement detail page. So, this website is working as a communication channel for both buyer and seller.

Bottom part of **Fig.3** is describing how code is working behind this application. If seller will give input then input request will be sent to the Add/Edit controllers. Business logic is written in classes of controller. If request is invalid then acknowledgement will return to back in the form of error message otherwise request will forward to model. Database queries have been written in classes of model. In case of invalid request an acknowledgement will return to controller otherwise data will store or modify into database, depends upon nature of queries. Same as in the case of buyer but 'search' classes of controller and model will be used there, and data will be fetched from database instead of insert or modify.

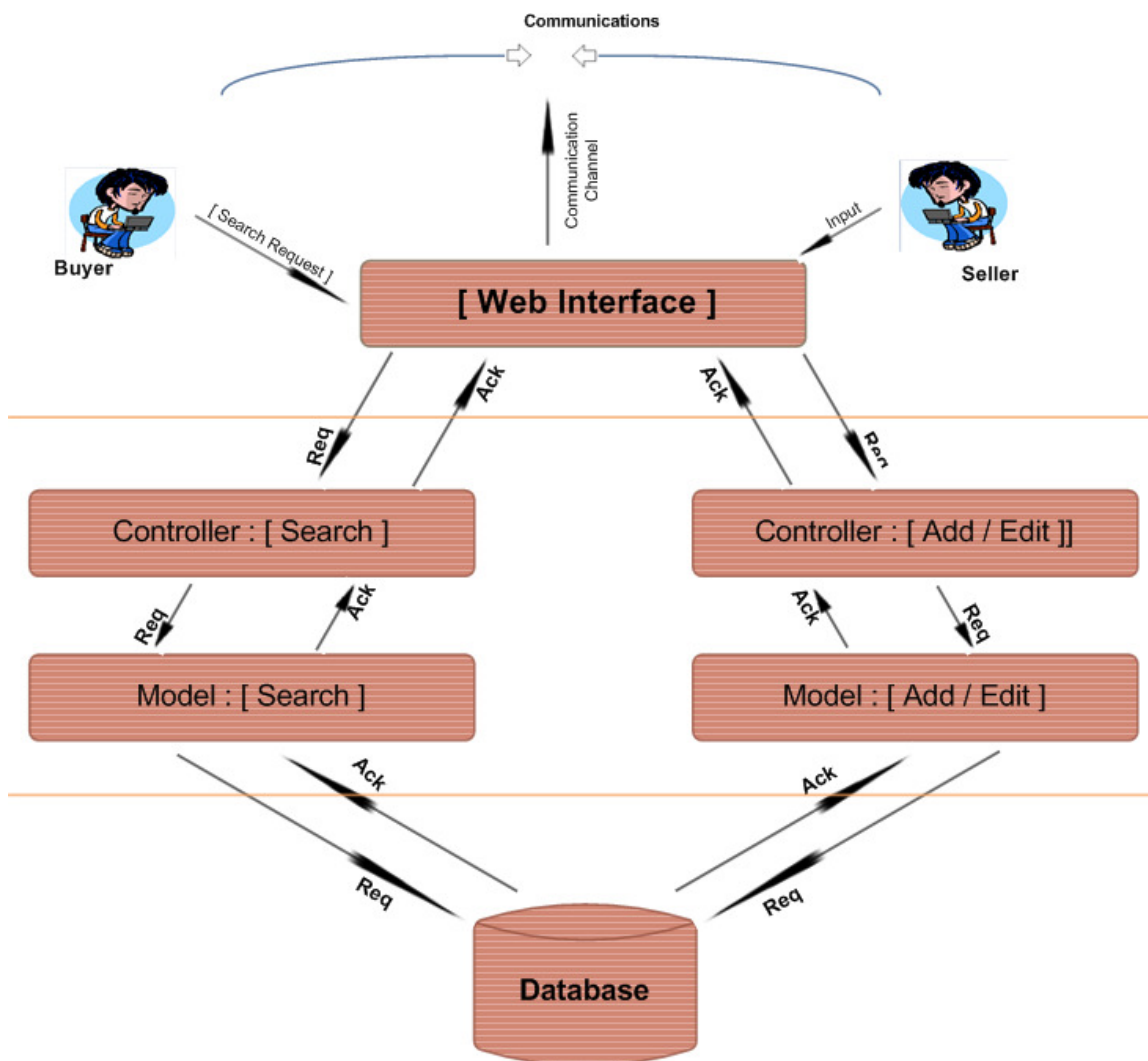


Fig.3

2.3. Usability:

Interface of website is very simple, user friendly, light weight and having light color combinations. Website is very simple to use for both sellers and buyers and having attractive look. Website having correct flow of setup, data element and controls are nicely grouped. Feedback for errors or exceptions to user is showing in very attractive ways. JavaScript library J-Query has been used to make pages user friendlier.

2.4. Testing:

Software testing is a process which makes sure, does the software is working according to requirement specification. There is lots of testing approaches or techniques but some common techniques those are used in this project are mentioned below.

2.4.1. Black box testing:

Black box testing is the testing which tests the functionality of the system. It doesn't concern about the code or internal structure of the system. Test cases are based upon the specification and requirement of the system that what the application suppose to do. Test cases can be implementing by giving correct or incorrect input to expecting right output. Test cases of black-box testing are described in chapter 3.1 of this report.

2.4.2. White box testing:

White box testing is also known as structural testing or transparent testing. By using white box testing, internal structure and working of application is being test. To perform white box testing, it is necessary to have programming skills and internal knowledge of system. Test cases of white-box testing are described in chapter 3.2 of this report.

2.4.3. Security testing:

Use of internet is a common activity of daily life. This field continuously growing day by day and currently this domain is in a big extension. There is some sensitive web applications those contain secrete data, for example online payment systems, online banking systems etc. Every web application those contains private data needs to be secure by attacks. Web application security is very difficult to apply but it is important as well. Most common attacks on web are XSS (Cross Site Scripting) and SQL injection. There are two areas those are considered secure and being experimented by some attacks. One experiment is to get login access without having login information. Second experiment is to status value of garage database table which is pending or confirm. Test cases of security testing are described in chapter 3.3 of this report.

3. Building a garage sale:

All new work which has been done on this project is mentioned in this chapter.

3.1. Test Cases of black-box testing:

Environment:

Operating System: Windows 7

Browsers: Mozilla Firefox, Google Chrome

3.1.1. Home Page:


Description:

To verify that following links are working properly after user clicks on one of these links.

- Information
- Vanliga frågor
- Kontakt

Steps to reproduce:

1. Open "Garage Sale" Application
2. Click on "Information" link in footer
3. Click on "Vanliga frågor" link in footer
4. Click on "Kontakt " link in footer



Information | Vanliga frågor | Kontakt

Expected Result:

System should lead to browser to the clicked link page and also show proper information

on this page.

Actual Result:

System had shown no response. When user has click on link, application don't show any activity.

3.1.2. Search Engine:

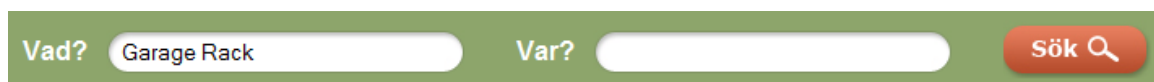
3.1.2.1. Search Engine scenario 1:

Description:

To verify that user can perform search from "Vad" field.

Steps to reproduce:

1. Open "Garage Sale" Application
2. Enter some Search string in "Vad" field
3. Click on "Sök" button



Expected Result:

System should show exact or similar search result according to this search criterion. Those garages should be appeared in list of search result, which has this item "Garage Rack".

Actual Result:

System shown some results but algorithm of search engine needs to more refine in order to produce better results.

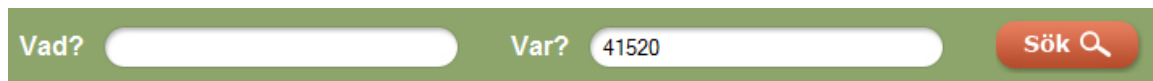
3.1.2.2. Search Engine scenario 2:

Description:

To verify that user can perform search from “Var” field.

Steps to reproduce:

1. Open "Garage Sale" Application
2. Enter zip code of some area in "Var" field
3. Click on "Sok" button



Expected Result:

System should show exact and/or similar search result according to this search criterion and expired garage should not shown. That garage should be appearing in list of search result, which is located in this area “41520”

Actual Result:

System has shown some results expired garage as well.

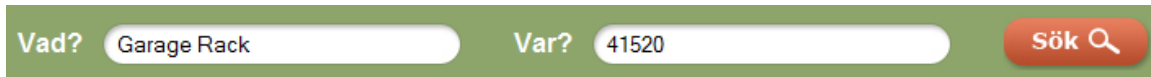
3.1.2.3. Search Engine scenario 3:

Description:

To verify that user can perform search from “Vad” field and “Var” field.

Steps to reproduce:

1. Open "Garage Sale" Application
2. Enter some Search string in "Vad" field
3. Enter zip code of some area in "Var" field
4. Click on "Sok" button



The image shows a search bar interface with a green background. On the left, there is a label "Vad?" followed by a white input field containing the text "Garage Rack". To the right of this is a label "Var?" followed by another white input field containing the text "41520". On the far right, there is a red button with the text "Sök" and a magnifying glass icon.

Expected Result:

System should show exact or similar search result according to this search criterion. That garage should be appearing in list of search result, which has this item “Garage Rack” and which is located in this area “41520”.

Actual Result:

System has shown some results but algorithm of search engine needs to more refine in order to produce better results.

3.1.2.4. Search Engine garage detail page:

Description:

To verify that the results which are produced by search engine are displayed properly on garage advertisement detail page.

Steps to reproduce:

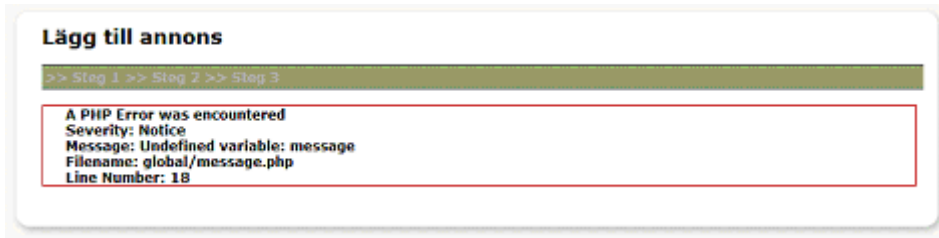
1. Open "Garage Sale" Application
2. Enter some Search string in "Vad" or zip code in “Var” field
3. Click on "Sok" button
4. Garage listing will be shown
5. Click on one of them in the list

Expected Result:

By clicking on link of garage advertisement in the list system should show detail page of garage advertisement.

Actual Result:

System don't shown detail page. An error has been appeared on detail page.

**3.1.3. Lägg till annons:****3.1.3.1. Lägg till annons Steg 1:**Description:

To verify that all fields working well, in order to save accurate data.

Steps to reproduce:

1. Open "Garage Sale" Application
2. Click on "Lägg till annons" tab
3. Fill the form by writing any information in the fields
4. Click on "Nasta Steg" button

Vem? Fyll i dina kontaktuppgifter.

Namn:
Mahboob Hussain

E-post:
khokhar@student.chalmers.se

Telefon:
abcdefg

Välj typ: Vanlig garagesale Flytt / Dödsbo

Vad vill du kalla din garagesale?
My Garage

Beskrivning - fritext
a

Allt ska bort!
Markera om du vill sälja precis allt på din loppmarknad.

Expected Result:

System should show proper error message that "Please enter correct phone number", "Text should not be less than 15 characters".

Actual Result:

System has accepted all kind of information and do not restrict or prompt any kind of error or user message.

3.1.3.2. Lägg till annons Steg 2 image uploading:

Description:

To verify that image uploading module working well when user want to add item's image.

Steps to reproduce:

1. Open "Garage Sale" Application
2. Click on "Lägg till annons" tab
3. Enter all information on "Stage 1" and go to "Stage 2"

4. Select an image file through "browse" button in first field
5. Fill some items title fields
6. Leave all other browse image fields empty
7. Press "Lägg till prylar" button

Vad? Vad vill du sälja?

Lägg till prylar (Minst 10 prylar till)

Beskrivande titel:

item 1
item 2
item 3
item 4
item 5

C:\Users\Public\Pictures	Browse...
	Browse...
	Browse...
	Browse...
	Browse...
	Browse...
	Browse...
	Browse...
	Browse...
	Browse...

+ Lägg till rad

Återställ alla fält

Lägg till prylar

Expected Result:

System should upload only image for the browse field and show blank or default "no image" with all other items.

Actual Result:

Image uploading module creates problem with uploading image. System is showing image with all fields even those don't had images at uploading time.

Vad? Vad vill du sälja?

Redan tillagda prylar: (5 prylar tillagda)

	Beskrivning: item 1 <input type="button" value="Ta bort"/>
	Beskrivning: item 2 <input type="button" value="Ta bort"/>
	Beskrivning: item 3 <input type="button" value="Ta bort"/>
	Beskrivning: item 4 <input type="button" value="Ta bort"/>
	Beskrivning: item 5 <input type="button" value="Ta bort"/>

3.1.3.3. Lägg till annons Steg 3 Ta bort:Description:

To verify that "Ta bort" button working well when user wants to delete a specific item.

Steps to reproduce:

1. Open "Garage Sale" Application
2. Click on "Lägg till annons" tab
3. Enter all information on "Stage 1" and go to "Stage 2" and "Stage 3"
4. Click on "Ta bort" button

Expected Result:

System should remove the specific item of 'Ta bort' and remove image of this item as well.

Actual Result:

If one item deleted then all other images are disappeared and design of page has been affected. Some times system showed error message exception on the page which is wrong.

Vad? Vad vill du sälja?

Redan tillagda prylar: (4 prylar tillagda)

Beskrivning: item 1 <input type="button" value="Ta bort"/>
Beskrivning: item 2 <input type="button" value="Ta bort"/>
Beskrivning: item 3 <input type="button" value="Ta bort"/>
Beskrivning: item 4 <input type="button" value="Ta bort"/>

Some times page has been crashed after clicking "Ta bort" button and system has shown these kinds of messages.

```
A PHP Error was encountered  
  
Severity: Warning  
Message: unlink(tmp_upload/7fd93432f722d43aa20d014d7c03d4ba) [function.unlink]: No such file or directory  
Filename: controllers/add.php  
Line Number: 356
```

By pressing "Ta bort" button some time deleted wrong data when user wants to delete some other items.

3.1.3.4. Lägg till annons stages navigation problem:

Description:

To verify that all three stages working well.

Steps to reproduce:

1. Open "Garage Sale" Application
2. Click on "Lägg till annons" tab
3. Enter data on "Stage 1" and click on "Natsa Steg" button
4. Enter all in formation on "Stage 2"
5. Now click on link "Steg 1" to again come to Stage 1

Lägg till annons

Efter att du klarat ett steg kan du gå tillbaka till steget genom att trycka på länkarna i navigationsmenyn nedan.



Vad? Vad vill du sälja?

Expected Result:

System should populate all information into the form on "Stage 1" which user had filled before switching to "Stage 2".

Actual Result:

System has shown all fields of form at "Stage 1" empty when user again comes on "stage 1" from "stage2", it seems all data has been lost.

At stage 2 users will redirect on stage 1 by using navigator link but on stage 1 there is no link to move again on stage 2.

3.1.3.5. Lägg till annons email format:

Description:

To verify that all auto generated emails by system are working well.

Steps to reproduce:

1. Open "Garage Sale" Application
2. Click on "Lägg till annons" tab
3. Enter all information on "Stage 1" and go to "Stage 2"
4. Select an image file through "browse" button
5. Fill items title fields
6. Press "Lägg till prylar" button
7. Press "Gå till nästa steg" button to switch on stage 3
8. Press "Publicera annonsering" button to publish advertisement

Expected Result:

System should submit page and must have to send a verification email in readable format.

Actual Result:

System has generated email but whole email text is unreadable and in scattered form.

User is unable to understand.

3.1.3.6. Lägg till annons display of items pictures:Description:

To verify that all item's pictures are uploaded well and shown perfectly in animated slide show.

Steps to reproduce:

1. Open "Garage Sale" Application
2. Click on "Lägg till annons" tab
3. Enter all information on "Stage 1" and go to "Stage 2"
4. Select an image file through "browse" button
5. Fill items title fields
6. Press "Lägg till prylar" button
7. Press "Gå till nästa steg" button to switch on stage 3

8. Click on any picture

Expected Result:

System should have proper small resized thumbnail images and also proper size of bigger image which will displaying in animated slide show.

Actual Result:

System did not generate the thumbnail of pictures. Width and height of picture is fixed in HTML by forcing.

Bigger picture is not in proper size. In slide show actual picture is displaying which can be too big size.

3.1.3.7. Lägg till annons verification email:

Description:

To verify that verification email working well, when user publish garage advertisement successfully.

Steps to reproduce:

1. Open “Garage Sale” Application
2. Click on ”Lägg till annons” tab
3. Enter all information on “Stage 1” and go to “Stage 2”
4. Select an image file through “browse” button
5. Fill items title fields
6. Press “Lägg till annonser” button
7. Press “Gå till nästa steg” button to switch on stage 3
8. Press ”Publicera annonsering” button to publish advertisement

Expected Result:

An automatic email with garage id and password should be sent on publisher’s email

address. System should show the proper message of email sent.

Actual Result:

When user clicking on link which is given in verification email then page has been crashed and following error message has been occurred 'Uninitialized string offset: 36'.

3.2. Test Cases of white-box testing:

Environment:

Windows 7,

Apache/2.2.11 (Win32)

PHP/5.3.0

phpMyAdmin 3.2.0.1

SVN

3.2.1. Search Engine:

Description:

System also has shown advertisements of garage which has been expired. Search engine also showing only exact results but does not producing the similar search results according to search criteria.

Steps to reproduce:

1. Open controller
2. Open search.php file
3. Go to result function
4. Find 'What' request for search

Expected Result:

Code should be according to find similar results of description field of garage items. And expired garage must not be shown in search list.

Actual Result:

There has no condition in code which could prevent to display expired garages in search result. Search code is written according to items of garagesale but producing only exact result, similar result are not shown.

3.2.2. Database table status field:Description:

Database 'garagesale' table status field type is varchar but it should be Enum('pending', 'confirm').

Steps to reproduce:

1. Open "phpMyAdmin" Application
2. Chose 'garagesale' database
3. Click on "garagesale" table
4. View status field

Expected Result:

System should show drop-down list when developer want to insert data manually into table by using phpMyAdmin interface.

Actual Result:

System shown text field and developer can insert wrong data which is harmful at the time when retrieving data by code.

3.2.3. Database table flag field:Description:

Added new ENUM type flag field in database 'garagesale' table enum('complete', 'incomplete'), which will keep track the uncompleted garages.

Steps to reproduce:

1. Open "phpMyAdmin" Application
2. Chose 'garagesale' database
3. Click on "garagesale" table
4. View flag field

Expected Result:

System should show drop-down list when developer want to insert data manually into table by using phpMyAdmin interface. It should be marked 'incomplete' if user will leave advertisement without doing finish.

Actual Result:

There was no such a field which could keep track of uncompleted garages, that's why navigation steps were not working properly which was totally based upon sessions.

3.2.4. Verification Email Problem:Description:

At publish time on stage 3 after clicking 'Publicera annonsering' button an email will be sent to user. After clicking on link which is given in email opened page has been crashed and following error message has been exposed 'Uninitialized string offset: 36'. This error was appeared by random string generated function genRandomString() due to empty array index.

Steps to reproduce:

1. Open the add controller class
2. Visit validation_mail() function
3. Visit genRandomString() function which is called by validation_mail()

Expected Result:

Random string should be generated every time according to length of string.

Actual Result:

Random string is not generating every time according to given length of string but some times string length is greater/less than array list which is the reason of the error.

3.3. Test Cases of security-testing:

Environment:

Operating System: Windows 7

Browsers: Mozilla Firefox, Google Chrome

3.3.1. SQL Injection attack:

SQL Injection is a technique which is used to operate queries of database on server side. This trick is used by sending some special characters through input field or through address bar to manipulate queries on the server side.

The main idea is to bypass the user authentication at login time. The ideal query at that time is the following.

```
"SELECT * FROM garagesale WHERE id = '"+garageID+"' AND password = '
'+userPassword+'";
```

Attacker can enter the following string into the input field of email to force query for execute. Such as "17 OR 1=1;--", then the query which will be executing will look like this:

```
"SELECT * FROM garagesale
WHERE garageID = 17 '
OR 1=1; --
AND password = '";
```

To perform this basic SQL injection, hacker needs to know what database structure is,

what database table name is, what table fields name is etc. There are two kinds of SQL Injections which has been experimented. First of all user can modify URL string to attack, and secondly some characters strings can be sent by input fields.

Active Record library of Code Igniter framework has been used on every action which is used to communicate with database, and this library take cares of this problem. So, system does not affected by using these SQL Injection techniques. Although, implementation of security more dependent on developer than framework.

3.3.2. XSS (Cross site scripting) attack:

Cross site scripting is the technique to exploit computer security vulnerability by injecting malicious script into web from client side. This trick is mostly used to attack on web applications by hackers. They use weak pages of web to bypassing the client side security mechanisms and inserting their malicious script into the web.

There are several ways to protect the web application by XSS attacks and CodeIgniter have some built-in methods to prevent these attacks. CodeIgniter has its own XSS hack prevention filter that filters post and cookies data that is encountered. This filter is use to take action at submission time. Following function is use to filter the data through XSS filter `xss_clean()`.

For example if we have data into `$data` variable then it can be filter by using this function, such that.

```
$data = $this->input->xss_clean($data);
```

XSS filtering is also use in form validation class. Active record library of CodeIgniter is filtering data automatically before sending data to database.

3.4. New functionality:

Environment:

Windows 7,

Apache/2.2.11 (Win32)

PHP/5.3.0

phpMyAdmin 3.2.0.1

SVN

3.4.1. Edit Garage:Description:

Edit Garage module working on the base of add garage. User will get modification access of specific garage advertisement by an email after publishing garage announcement successfully. For edit his/her own garage announcement user will click the link 'Redigera annons' at top header. A login page will appear which demanding garage-ID and password which already sent to user by auto generated email.

3.4.1.1 Edit Garage – Login:

This login module will use to authenticate the user who is going to modify garage advertisement. Login page will demanding garage-ID and password which has been sent to user via auto generated system's email after publishing garage successfully.

Inloggning

Garage ID:

Lösenord:

3.4.1.2 Edit Garage – Steg 1:

After logged in successfully stage one of garage announcement page will be shown to user. User can modify data according to his/her wish and submit for by clicking 'Nästa Steg' button. User can also use navigation bar to visit all stages without modify any data. User can modify following data on stage one.

Name, Email, Telephone number, Type, Garage Title, Description, Date, Time, Address, Zip code,

Namn:
Mahboob

E-post:
mehboobkhokher@gmail.com

Telefon:
123456

Välj typ: Vanlig garagesale Flytt / Dödsbo

Vad vill du kalla din garagesale?
Garage1 for Cars

Beskrivning - fritext
This is Garage1 for Cars, visit it and hope you will enjoy. This is Garage1 for Cars, visit it and hope you will enjoy.

Allt ska bort!
Markera om du vill sälja precis allt på din loppmarknad.

Jag har minst 50 föremål att sälja
Du måste ha minst 50 föremål att sälja för att få annonsera!

När? Välj tillfälle för din garagesale.

Välj datum: (AAAA-MM-DD) 2010-10-22 

Välj tid: (HH:MM)
Från: 10:00 Till: 12:00

Var? Välj plats för din garagesale.

Gatuadress: CHALMERS TEKNISKA H

Postnummer: (XXXXX) 41290

3.4.1.3 Edit Garage – Steg 2:

On second stage user can watch all the item descriptions with small pictures. Item's

pictures can be shown in bigger in animation slide show by click on any small picture.

User can delete any/every item. User can add more items and the description of items.

If user wants to add more items then after fill the form user must press the button 'lägg till prylar'. If user want to reset the input fields then he/she hast to press the button 'Återställ all fält'. If user wants to go next stage then he/she has to press the button 'Gå till näst steg'. User can also use navigation bar as well to visit all stages without modify any data at any stage.

User can perform following actions on second stage:

- Can watch big pictures of items
- Can delete items
- Can add more items without pictures
- Can add more items with pictures
- Can go to previous stage
- Can go to next stage

Vad? Vad vill du sälja?

Redan tillagda prylar: (3 prylar tillagda)



Beskrivning: garage item 1

Ta bort



Beskrivning: garage item 2

Ta bort



Beskrivning: garage item 3

Ta bort

Lägg till prylar (Minst 7 prylar till)

Beskrivande titel:

 Browse...
 Browse...
 Browse...
 Browse...
 Browse...
 Browse...
 Browse...
 Browse...
 Browse...
 Browse...
 Browse...

+ Lägg till rad

Återställ alla fält

Lägg till prylar

Tryck på knappen "Lägg till prylar" innan du går vidare till nästa steg.

Gå till nästa steg

3.4.1.4 Edit Garage – Steg 3:

On third stage user can watch the summery of fist and second stage and can watch all the item descriptions with small pictures. Item's pictures can be shown bigger in animation slide show by click on any small picture. If summery is not according to wishes of user then can be changed by using navigation bar above or buttons at bottom.

User can perform following actions on second stage:

- Can watch all data which was entered at first and second stage
- Can watch pictures of items in animated slide show

- Can go to first stage
- Can go to second stage

Redigera annons

Efter att du klarat ett steg kan du gå tillbaka till steget genom att trycka på länkarna i navigationsmenyn nedan.

>> [Steg 1](#) >> [Steg 2](#) >> [Steg 3](#)

Din garagesale - steg ett

Namn: Mahboob	E-mail: mehboobkhokher@gmail.com	Telefon: 123456
Titel: Garage1 for Cars	Typ: Dödsbo	
Beskrivning: This is Garage1 for Cars, visit it and hope you will enjoy. This is Garage1 for Cars, visit it and hope you will enjoy.		
Allt ska bort: Ja		
Datum: 2010-10-22 00:00:00	Från: 10:00	Till: 12:00
Gatuadress: CHALMERS TEKNISKA H	Postkod: 41290	

Dina föremål - steg två



Beskrivning: garage item 1



Beskrivning: garage item 2



Beskrivning: garage item 3

Jag vill ändra steg ett!

Jag vill ändra steg två!

3.4.2. Delete Garage Advertisement:

Description:

User can delete his/her garage advertisement by clicking delete icon. After logged in successfully delete icon will be shown on every stage of garage announcement pages. Every thing will be deleted which has concern on this garage. All items description and pictures also will delete.



3.4.3. Suggestion list of search engine:

Description:

When user wants to search some items of garage or garage location by zip code then he/she needs to write in vad and var text fields. Item name should be typed Vad field and zip code should be typed in var field. When user will start typing text then suggestion list of relevant search criteria will be displayed right bottom of text fields. User needs to click on any item name on the suggestion list to populate text into field.

3.4.3.1. Suggestion list of search engine-Vad:

Description:

In Vad field user needs to start writing item name and suggestion list according to item name will be display right bottom of that text field.



GARAGESALE 

Sök annons Lägga till annons Redigera annons Hjälp

Vad? Var?

Lägg

>> Steg 1

Alla fält måste

Vem?

- garage item 1
- garage item 2
- garage item 3
- garage item 4
- garage item 5
- garage item 6
- garage item 7
- garage item 8

3.4.3.2. Suggestion list of search engine-Var:

Description:

In Var field user needs to start writing zip code of his/her required area and suggestion list according to zip code will be display right bottom of that text field.



GARAGESALE 

Sök annons Lägga till annons Redigera annons Hjälp

Vad? Var?

Lägg till annons

>> Steg 1 >> Steg 2 >> Steg 3

Alla fält måste fyllas i.

- 41520
- 41290
- 41296

3.4.4. Cron Jobs:

Description:

Cron Jobs module managing the website's database features. It will run automatically on server and will perform actions on database according to nature of functions in given time slots.

Those garages belongs to expired dates and their flag field is marked 'incomplete' will be delete by the following script. Flag field 'incomplete' referring those garages which remained incomplete while creating new garage.

```
function deleteIncompleteGarages()
{
    $incomplete_garages = $this->cronjobs_model->get_incomplete_garages();

    foreach ($incomplete_garages as $incomplete_garage)
    {
        $this->_deleteGarageItems($incomplete_garage['id']);
        $this->cronjobs_model->delete_garage($incomplete_garage['id']);
    }
}
```

Those garages belongs to expired dates and their flag field is marked 'pending' will be delete by the following script. Flag field 'pending' referring those garages which are not verified yet.

```
function deleteExpiredPendingGarages()
{
    $expired_pending_garages = $this->cronjobs_model->get_expired_pending_garages();

    foreach ($expired_pending_garages as $expired_pending_garage)
    {
        $this->_deleteGarageItems($expired_pending_garage['id']);
        $this->cronjobs_model->delete_garage($expired_pending_garage['id']);
    }
}
```

A private function `_deleteGarageItems` is calling by both functions. This function is using to delete items of a particular garage. Code of this function is given below.

```

function _deleteGarageItems($garageId)
{
    $items = $this->cronjobs_model->get_items_by_garage_id($garageId);

    if(!empty($items))
    {
        foreach ($items as $item)
        {
            $file_root_path = $_SERVER['DOCUMENT_ROOT']."/garagesale/items/thumbnails/".$item['thumbnail'];
            if(file_exists($file_root_path) && $item['thumbnail'] != 'no_photo.jpg')
                unlink($file_root_path);

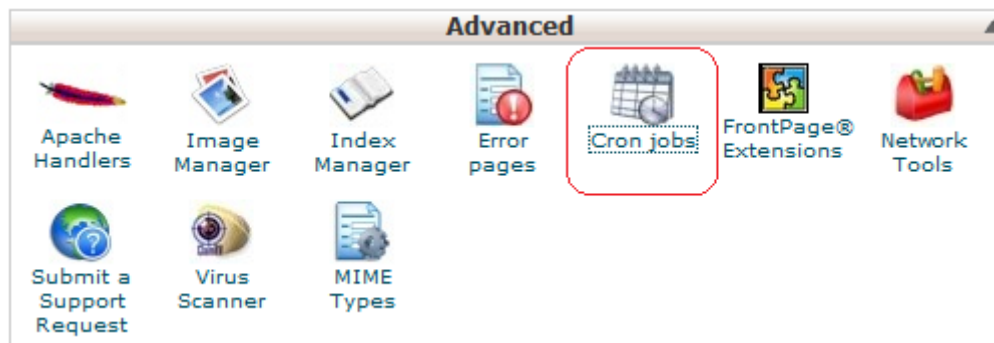
            $file_root_path = $_SERVER['DOCUMENT_ROOT']."/garagesale/items/medium/".$item['medium_photo'];
            if(file_exists($file_root_path) && $item['medium_photo'] != 'no_photo.jpg')
                unlink($file_root_path);

            $file_root_path = $_SERVER['DOCUMENT_ROOT']."/garagesale/items/pictures/".$item['filename'];
            if(file_exists($file_root_path) && $item['filename'] != 'no_photo.jpg')
                unlink($file_root_path);
        }

        $this->cronjobs_model->delete_items_by_garage_id($garageId);
    }
}

```

Cron jobs setting will be manage at control panel of domain.



These functions will be given to cron jobs settings in control panel of domain.

3.4.4.1. Cron Jobs Setting:

Description:

In common settings time of script can be decided that when script will be run automatically on server. There are several options i.e. every minute, every five minute, twice an hour, twice a day, once a day, once a week, once a month, once a year etc.

URL of `deleteIncompleteGarages()` function will be

<http://www.mahboobhussain.com/garagesale/deleteIncompleteGarages>

By running this URL all those advertisements and their items will be removed from database those are not completed successfully at posting time. Setting shown in screen shot is the setting of URL running cycle is once a day.

Add New Cron Job

Common Settings:	Once a day (0 0 ***)			
Minute:	0	:00 top of the hour (0)		✓
Hour:	0	12:00 a.m. midnight (0)		✓
Day:	*	Every day (*)		✓
Month:	*	Every month (*)		✓
Weekday:	*	Every weekday (*)		✓
Command:	<input type="text" value="sain.com/garagesale/deleteIncompleteGarages/ "/>			✓
<input type="button" value="Add New Cron Job"/>				

Current Cron Jobs

MINUTE	HOURL	DAY	MONTH	WEEKDAY	COMMAND	ACTIONS
No Cron Jobs						

URL of **deleteExpiredPendingGarages()** function will be

<http://www.mahboobhussain.com/garagesale/deleteExpiredPendingGarages>

By running this URL all those advertisements and their items will be removed from database those are not confirmed by verification email. This URL will run on server once a day according to setting shown in screen shot.

Add New Cron Job

Common Settings:	Once a day (0 0 * * *)		
Minute:	0	:00 top of the hour (0)	✓
Hour:	0	12:00 a.m. midnight (0)	✓
Day:	*	Every day (*)	✓
Month:	*	Every month (*)	✓
Weekday:	*	Every weekday (*)	✓
Command:	:om/garagesale/deleteExpiredPendingGarages/		✓
<input type="button" value="Add New Cron Job"/>			

Current Cron Jobs

MINUTE	HOURL	DAY	MONTH	WEEKDAY	COMMAND	ACTIONS
0	0	*	*	*	http://www.mahboobhussain.com/garagesale/deleteIncompleteGarages/	Edit Delete

3.4.5. Refinement of search engine:

Old search engine was requiring refinement to show exact, similar and quick results. To do achieve this task some changing has been performed in algorithm of search engine. For example to get exact matched and similar results 'LIKE' SQL operation has been used. Which is displaying now exact results, and those results that's first or last string are matched. To get quick results algorithm has been modified, to find items SQL query was used within the nested loop instead having joins in database tables. That's why results were a bit slower.

3.4.6. User friendly URLs:

Complex and hard typed URLs are often considered bad, because these are difficult to remember and also too long. Dirty URLs also containing security risk, because the question mark (?) used in is often modified by hackers to make front door attack. Also dirty URLs giving file extension such that .jsp, .asp, .php etc which are also helpful for hackers to utilized. So, URL should have some meaning and easy to remember for both user and search engine as well, that's why user friendly URLs often called search engine

friendly URLs.

In order to get user friendly URLs .htaccess file has been used. By using this file some extra characters such that (?), (&) or (.php) file extensions etc has been eliminated.

3.5. Possible future work suggestions:

Construction cannot stop, continuous changing is necessary to keep growing in dynamic market. Variation in requirement is very sudden and it can be manage by doing continuous development. Web is the part of life in daily routine and this is the major source to get information quickly by saving time. Although the basic functionality is enough to just launch this site but there are some suggestions those are future need of this website

3.5.1. Super Administration:

Almost every dynamic website has its super administration part, where super admin can manage whole system. Although advertiser have administration rights in garage sale but a super administration part is require to keep track of advertisers and block or delete irrelevant or fake ads.

3.5.2. Human verification code:

To prevent website by attacks of hackers, there should be human verification code at the first page of post advertisement, so that no automatic posting script or system could work there. Only a human will be able to post form nothing else.

3.5.3. Keep tracking for visitors:

There should be a tracking module which will be use to keep track that how many visitors this web site have in one day, one week, one month and one year etc. This module is help full to get idea about visitors traffic of this website. This module can work for both admin

as well as visitors.

3.5.4. Keep tracking for advertisers:

There should be a tracking module which will be use to keep track that how many users posting ads in one day, one week, one month and one year etc. This module is help full to get idea that how much is the use of this website.

3.5.5. Advance Search:

Advance search option is very common now days. If search engine will produce hundreds of results then it would be definitely hard for a visitor to getting exact choice without seeing detail of particular result. So, in advance search visitor could choose search criteria according to location, category, price range, size/area of garage etc.

3.5.6. Pagination:

After getting search results by using search engine, there might be hundreds of results, those will be quite heavy to load a webpage and system will be slow down and page will be unnecessary bigger due to lot of data. So, to make better performance and well looking search result page there should be pagination system.

3.5.7. Send Email to seller:

To make deal final there is necessary to have communication between visitor and seller. Although an email address has been mentioned on garage detail page, but to make communication quicker there should be a quick email send system for buyer on detail page of garage.

3.5.8. Send SMS to seller:

Most fast electronic way for communication is mobile. To make deal done there is necessary to have communication between visitor and seller. Although mobile number has been mentioned on garage detail page, but to make communication quicker there should be a quick SMS sending system for visitor on detail page of garage. By using SMS option buyer can send a text message to seller's mobile number.

3.5.9. Tell to friends:

Information sharing is a daily routine of life. Everyone wants to share useful information with their friends or family. So, on the detail page of garage there should be an option from where visitors could share this detail to their friends. This could be simply an email address text field.

3.5.10. Share garage on facebook:

Facebook is the world's largest online social network. If someone is sharing information on the facebook wall then every friend of him can have look and they can also share as well. This could be a good way to increase popularity of website. Good thing is that facebook providing free APIs to share information on user's wall, so that everyone can add this functionality to their websites.

3.5.11. Report as abuse:

To manage the website it is required that there should not any kind of fake or abusing material on it. To keep handle this situation there should be a 'Report' option on the detail page of garage where visitor can report the ad by clicking on it. To take action on reported ad admin must have rights to block or remove.

3.5.12. Forget password:

To manage the advertisement there is a login system, but it's very easy to forget password or auto generated email could also be deleted sometimes. To cover up this loss there should be a forget password system, from where user could get new password by email.

3.5.13. Auto email for expired garage:

In case a garage is expired then this will no longer display to the visitors by using search engine. If a garage is not useful then this is just occupying space and memory, which should not be. To resolve this issue there should be auto email generating system for those users whom garage ads have been expired. Email should have information that if they want their advertisement any more then they must have to upgrade their last dates by using garage id and password given in email contents. Otherwise garage will automatically delete from system after 15 days.

4. Result:

During the different test scenarios of black box testing number of issues have been reported. Some bugs captured by applying common scenarios of test cases, some critical issues founded through applying complex test cases. All test cases of black-box are written in chapter 3.1.

During the review of code it has been noticed that code was written in excellent style, very clean, properly commented and fulfilling the MVC design pattern. During the white box testing when code has been reviewed some logical faults have been reported. These faults were the reason of producing bugs which were mentioned in black box testing. Test cases of white-box are written in chapter 3.2.

On the security point of view Codeigniter framework is quite secure. Some common SQL Injections and cross site scripting attack has been applied but no vulnerability could found in the system. There are built-in mechanisms to prevent these kinds of attacks. By using these experiments no negative results has been produced. Active Record Library of CodeIgniter framework prevents the SQL Injection attacks and XSS attacks. Active Record Library methods have been used in past development as well as in latest code. Common attacks for security testing are written in chapter 3.3.

Different kinds of picture's thumbnails have been generated. There are two kinds of thumbnail, small and medium. Actual bigger pictures were too heavy for webpage in listing of items.

Administration module for sellers has been developed to manage the contents of announcements by seller. Following is the description of results. Seller can modify the data of his own posted garage by using administration rights. Login information of administration would be sent automatically to seller via auto generated email. Sellers may upload or delete more photos of their own garage items. If garage has been sold out and/or sellers don't want to see this announcement any more on website then s/he can

delete advertisement. More detail is mentioned in chapter 3.4.1 and 3.4.2.

To facilitate the users there is suggestion list for search criteria. Suggestion list providing the list of similar records of user input in search engine, which is helpful and user friendly to choose search criteria. More detail is mentioned in 3.4.3.

CRONJOBS module has been developed to remove the pending-expired or uncompleted garages announcement. Script of CRONJOB will run automatically on the server and remove all expired or uncompleted advertisements of garage. About CRONJOB detail follow the chapter 3.4.4.

Search engine can find both exact and similar search result of search request. Main achievement is to hide expired garages while displaying in the listing of search results. Chapter 3.4.5 will describe the more detail about it.

During the development of new module and code review, I observe the coding style that was used in existing system is truly good. All code was properly commented, naming conventions are used in very good manners and I follow the same pattern throughout.

5. Conclusion:

Garage sale system is based upon C2C E-Commerce. This is easy to use and a quickest way to finding online information about garage. On the business point of view garage sale is a quite unique idea that has a specific target market. Garage sale is recently implemented. In the beginning it will difficult to grow up in market. It is tricky to getting more visitors/user in the beginning. To promote website and to get increase the traffic, it is necessary to apply some SEO and general strategies for instance offering free service in the beginning and for the time being free registration for sellers etc.

On the technical point of view system has been developed in CodeIgniter framework to follow standard design pattern MVC. Therefore, maintenance and enhanced future work will be managed easily. Testing is an important part of work. It is impossible to have a totally bug free website, even most popular and stable web applications sometimes producing functionality errors. It is also important to test web application against different kind of injections or attacks to check security vulnerability. So, common test cases have been applied but testing is continuous processes. CodeIgniter framework is a quite light weight and secure as well.

Some issues might appear in future research. For example new versions of CodeIgniter, MYSQL and apache server might not supportable with current technology. System might not compatible with latest versions of browsers. To find right resource for further development will be an issue, because developer must have knowledge of PHP and MVC design pattern. S/he should have full understanding about new requirements as well as previously developed system.

For the future research on this project, following is the set of project proposal with prerequisites and defined output. Researcher should have background in computer science and/or software engineering background. Knowledge of e-commerce is required. As thesis work web development experience will be considering a plus but not essential. Research content will be includes the analysis of full requirements of project, analysis of

previous work, analysis of technology for example framework, MVC design pattern, code review etc. On the development point of view implementation of some new module will be included such as super administration, keep track of advertisers, advance search. Super administration module will use to keep check and balance of all the announcements of sellers. There should be a weekly, monthly, yearly advertisers list, which will be show to administrator. Pagination is requiring for listing of garage and listing of items. So that data could be load easily on webpage. Possible future work suggestions are descriptively mentioned in chapter 3.5.

Reference List:

- [1] Aarhus School of Business Aarhus University, "WRITING A MASTER THESIS", "<http://www.asb.dk/article.aspx?pid=5295&lang=en-GB>" (Access: November, 2010)
- [2] Jarubula PraveenKumar (2008), E-Commerce, University Of Texas-Pan American
- [3] Forms of e-commerce, "Prifysgol cymru abertawe university of wales swansea", http://www.designersplus.co.uk/unit3/forms_of_e_commerce/level2.html (Access: November, 2010)
- [4] <http://fox.wikis.com/wc.dll?Wiki~E-commerceMatrix> (Access: November, 2010)
- [5] Wikipedia, the free encyclopedia, Black-box testing from: < http://en.wikipedia.org/wiki/Black-box_testing > (Access: November, 2010)
- [6] Wikipedia, the free encyclopedia, White-box testing from: < http://en.wikipedia.org/wiki/White-box_testing> (Access: November, 2010)
- [7] Cross-site Scripting (XSS), from: <http://www.owasp.org/index.php/Cross_site_scripting> (Access: November, 2010)
- [8] Advanced SQL Injection In SQL Server Applications, Chris Anley, An NGSSoftware Insight Security Research (NISR) Publication ©2002 Next Generation Security Software Ltd
- [9] Web Application Security, SQL-injection & Cross-Site Scripting attacks, Jonas Magazinius, Computer Science and Engineering, Chalmers University of Technology, 2009-09-28
- [10] Practical guide to subversion on Windows with TortoiseSVN, from: <<http://www.shokhirev.com/nikolai/programs/SVN/svn.html>> (Access: November, 2010)
- [11] The coolest Interface to (Sub)Version Control, from: <<http://tortoisesvn.net/>> (Access: November, 2010)
- [12] PHP Manual, from: <<http://php.net/manual/en/index.php>> (Access: November, 2010)
- [13] CodeIgniter User Guide, <http://codeigniter.com/user_guide/> (Access: November, 2010)
- [14] CodeIgniter forum, <<http://codeigniter.com/forums/>> (Access: November, 2010)
- [15] JQuery, JavaScript Library, < <http://jquery.com/> > (Access: November, 2010)
- [16] The world's most popular open source database, < <http://www.mysql.com/> > (Access: November, 2010)
- [17] Thesis reports of "Danish Ali Khan", "Muhammad Asif", "Muhammad Ayaz" and "Wende Linje".
- [18] Sami I Mäkeläinen(2006), From B2C to C2C e-commerce, UNIVERSITY OF HELSINKI, Department of Computer Science.