

**Software Requirement Specification
For
“Flea Market” System**

By

Ilya Verlinsky,
Alexander Sarkisyan,
Ambartsum Keshishyan,
Igor Gleyser,
Andrey Ishuninov

1 INTRODUCTION

1.1 Purpose

1.1.1 Purpose of SRS document

The purpose of this SRS is to supply our customer (in this case the general market) with an outline of a software product that will provide them with an affordable and reliable e-commerce solution.

1.1.2 Intended Audience

Individuals that are responsible for reviewing proposals for this system. This may include Managers, Network and DB Administrators of the company that may be interested in our software.

1.1 Scope

1.2.1 Software Product to be produced

This product currently code-named "Flea Market"

1.2.2 Purpose of "Flea Market"

Provide an environment for users of this system (See Section 2.3) to conduct a virtual "Flea market"/"Swap meet" and for the company that installs this product - a steady stream of revenue. This procedure can be summarized as follows:

1.2.2.1 The user will be able to register as a member and be a part of our virtual community.

1.2.2.1.1 This system could be accessible from anywhere over the Internet, or internally over the company's Intranet.

1.2.2.1.2 The user should be able to provide positive identification and a valid e-mail address

1.2.2.1.3 The system will verify users e-mail address and the checksum for the credit card submitted.

1.2.2.1.4 The user should be able to provide a unique user name and password for a positive identification during subsequent visits to the site.

1.2.2.1.5 There will be a user agreement and a privacy policy that the user will have to agree to upon registering.

1.2.2.2 Upon registering, member will be able to post items for sale on our system

1.2.2.2.1 The member will be able to specify a picture on his computer or URL for a picture that corresponds to the posted item.

1.2.2.2.2 User will receive an email confirmation upon posting item.

- 1.2.2.2.3 A full sales history will be recorded for each seller on the system to provide easy accounting solutions for the host company.
- 1.2.2.2.4 A user would be able to post his/her item under a system-defined category.
- 1.2.2.2.5 User will receive e-mail when the item posting expires, with an option to extend the posting.
- 1.2.2.2.6 IP from which the item for sale was posted will be logged for security purposes.
- 1.2.2.2.7 User will be able to remove items that have not been sold yet.
- 1.2.2.3 Upon registering member will be able to buy items posted by other members of our community
 - 1.2.2.3.1 The buyer will be able to rate the seller upon completion of sale.
 - 1.2.2.3.2 Buyer will receive an e-mail notification/receipt upon purchase of the item.
 - 1.2.2.3.3 IP from which the item was bought will be logged for security purposes
- 1.2.2.4 The user will be able to search for items posted by other members
 - 1.2.2.4.1 The user will be able to search item by category.
 - 1.2.2.4.2 The user will be able to search books by ISBN number.
- 1.2.2.5 The system will provide members with a shopping cart subsystem for the ease of shopping online.
 - 1.2.2.5.1 The user will be able to have a wish list of items that he/she wants to buy in the future
- 1.2.2.6 The user will be provided with an interface to view and modify his/her account information.
 - 1.2.2.6.1 The user will be able to view the outstanding balance that he/she owes to the company.
- 1.2.2.7 The system will have an administrative interface.
- 1.2.2.8 Host company will receive revenue based on the percentage of the amount of each transaction. This amount will be billed to the seller of the item.

1.3 Definitions, Acronyms and Abbreviations

SRS:	Software Requirement Specification.
URL:	Universal Resource Locator. Text mapping to IP
RDBMS:	Relational Database Management System
SQL:	Sequential Query Language (A Querying and reporting language used by all modern RDBMS)
HTML:	Hypertext Markup Language. A Formatting Language Used to encode web pages
DB:	The RDBMS used by our software.

ERD:	Entity relationship diagram.
DFD:	Data flow diagram.
ANSI:	American National Standards Institute
IEEE:	The Institute of Electrical and Electronics Engineers, Inc.
IP:	Internet Protocol (Commonly used as the unique address that identifies computers on the internet)
ISBN:	International Standard Book Number
GPL:	GNU General Public License
API:	Application Program Interface
RFC:	Internet Request For Comment. This is the documentation of the Internet Protocols.
GUI:	Graphical User Interface

1.4 REFERENCES

ERD (Attachment A)
DFD (Attachment B)
Data dictionary (Attachment C)
GPL license (Attachment D)
URL to a prototype GUI of the system

1.5 OVERVIEW

1.5.1 Contents of the SRS

This SRS will provide a description of "Flea Market" system. It will provide an outline of the requirements. It will also provide an overview of the characteristics and constraints of the "Flea Market" system.

1.5.2 SRS Organization

This SRS is organized in accordance with ANSI/IEEE Std. 830-1984

2. GENERAL DESCRIPTION

2.1 Product Perspective

The “Flea Market” system would be one of the first systems of its type and functionality. “Flea Market” is a system that will provide an affordable e-commerce solution for companies that want to receive revenue from the current Internet boom. It also provides and creates a virtual store front for end users to buy and sell items from each other. Flee Market e-commerce solution will give businesses an environment to generate revenue without offering of in-house products. “Flea Market” is an independent system. After installation, it will be self-contained. It does not require external information, although it does require a system administrator to maintain the system. The principle external interfaces that will be used include a Register Screen, Login Screen, Shopping Cart Screen, Wish List Screen, Administrator Screen, Add Item Screen, Account Status Screen and Search Screen. “Flea Market” is a very flexible system that could be integrated into any existing web site. The company running this system will have the freedom to use custom created web site, because of “Flea Market’s” flexibility and platform independence. For the same reason “Flea Market” will be easy to integrate into an existing site design.

2.2 Product Function

The main functions of "Flea Market" will be divided into two categories: user (buyer or seller) and administrative. The user will be able to post/buy items, add/delete items from shopping cart, add/delete items from wish list, check their account status and rating, rate other users. Administrator will have a separate web-based interface. Administrator will be able to change template, check history of sales, access users

information and account status of each user, have a total control of database, and have a total control of system configuration.

2.3 User Characteristics

The users of the “Flea Market” include the public and an administrator. The users are required to have basic knowledge and experience using a web browser. However, the GUI and online help should provide enough guidance to an average user. Administrator of this system should have knowledge of the specifics of hardware, operating system and software environment of the host system. This includes but is not limited to the knowledge of web server and database administration. Preferably the administrator will have basic knowledge of ANSI SQL, however this is not required.

2.4 General Constraints

- 2.4.1** This software system should be engineered to be as platform independent as possible, thus excluding the proprietary technologies. Preferably the development technology would be open-source, making it easy to understand particular features and shortcomings.
- 2.4.2** This software system should be easily installable by a System Administrator who possesses only knowledge about the system that he/she is working on.
- 2.4.3** This software system should run in conjunction with all the Web Server Software that is popular during the time of this writing. For the initial revision this set can be restricted to such web servers as – Apache (version 1.3 and above) and Microsoft IIS (version 4.0 and above). In the future revisions this software should be able to support such web servers as Netscape Enterprise, and Zeus.
- 2.4.4** This software system should be designed to easily interface with most of the modern SQL databases that are ANSI SQL compliant. For the initial revision this set can be restricted to such RDBMSs as MySQL (version 3.23 and above), and Oracle8i. For future revisions of the product it should have support for IBM’s DB2 database, Microsoft SQL server and other databases.
- 2.4.5** This system should be designed to run on a variety of hardware/OS platforms that are POSIX compliant, however it does NOT have to run on such platforms as IBM AS/400, any mainframe, or VAX systems. There are numerous other exotic operating systems that are not mentioned here that this product does not have to function under.

2.5 Assumptions and Dependences

- 2.5.2** Software developers assume that an ANSI compliant SQL database is available, or can be installed on the system under which it should function (See 2.4.4).
- 2.5.3** Software developers assume that a modern web-server with server-side scripting capabilities is available on the host system (See 2.4.3).
- 2.5.4** Software developers assume that hardware/software combination of the host system will be able to support the load associated with an e-commerce application.
- 2.5.5** Software developers assume that there will be a System Administrator that will maintain this system.

3. SPECIFIC REQUIREMENTS

3.1 Functional Requirements

3.1.1 LOGIN functional requirements

3.1.1.1 Introduction

The login function provides the means for the user to access “Flea Market’s” functions, which require user’s identification.

3.1.1.2 Inputs

Username
Password

3.1.1.3 Processing

The login function takes in a username and a password. It runs a check against the database to verify this information. If the check is successful then it will write a cookie on the users computer for subsequent access to the system during the current session.

3.1.1.4 Outputs

user.u_key in a cookie format

3.1.2 REGISTER functional requirements

3.1.2.1 Introduction

The register function takes a user’s information and creates a user account for that person on the system.

3.1.2.2 Inputs

User Information

3.1.2.3 Processing

This function takes a users information and creates an account on the system for that user.

3.1.2.4 Output

An e-mail confirmation is sent to the user.
User Record.

3.1.3 SHOPPING CART functional requirements

3.1.3.1 Introduction

The shopping cart function allows a member to purchase items from the “Flea Market” system. It also provides easy item tracking for the customer.

3.1.3.2 Inputs

user.u_key
item.i_key(s)
Item Quantity

3.1.3.3 Processing

The shopping cart takes the user.u_key and the item.i_key(s) of the items that the user wants to purchase and creates a transaction record. It may inactivate the item and remove it from the search table. It also decreases the item.qty_current.

3.1.3.4 Outputs

Transaction Record

3.1.4 WISH LIST functional requirements

3.1.4.1 Introduction

The wish list function allows a member to save list of items for easy retrieval and simplified purchase process at a later time.

3.1.4.2 Inputs

user.u_key
item.i_key(s)

3.1.4.3 Processing

This function takes a user.u_key and accesses the wish list table in the database. It gets the user's items in the wish list. After that it uses the item.i_key to get information about the specific item from the items table. It can also store an user.u_key and item.i_key combinations in wish list table for later retrieval.

3.1.4.4 Outputs

Wish list record
A buy request

3.1.5 RATING functional requirements

3.1.5.1 Introduction

This function allows a buyer to rate a seller of an item.

3.1.5.2 Inputs

transaction.t_key
user.u_key (For Buyer)
user.u_key (For Seller)
User comments about transaction
Rating Value

3.1.5.3 Processing

This function takes the transaction.t_key, and user.u_key for both buyer and seller and creates a record in the rating table for that transaction. The seller ID and the buyer ID also gets saved under this rating record.

3.1.5.4 Outputs

Rating Record
user.u_rating

3.1.6 SEARCH functional requirements

3.1.6.1 Introduction

This function searches the index for the keywords identifying the items

3.1.6.2 Inputs

Search String

3.1.6.3 Processing

This function takes search parameters from the user, parses them. It then searches the index for item.i_key that match the search strings and displays it to the user

3.1.6.4 Outputs

item.i_key

3.1.7 ACCOUNT STATUS functional requirements

3.1.7.1 Introduction

This function allows a user to view activity on their account and modify his/her membership information.

3.1.7.2 Inputs

user.u_key

3.1.7.3 Processing

This function takes the user.u_key identifies the user. With this information the function can fetch account information. The user can view their sales history, view ratings given by his/her customers and the user may view their outstanding balance. Users can also modify their membership information (i.e address, email, etc.)

3.1.7.4 Outputs

Outputs parts of user record that users has modified.

3.1.8 DISPLAY functional requirements

3.1.8.1 Introduction

This function is called from other functions to display listings of items in a formatted output.

3.1.8.2 Inputs

item.i_key(s)
category.c_key(s)

3.1.8.3 Processing

This function takes the item.i_keys fetches item records and prints out the listing in a formatted way

3.1.9 PRODUCT ADD functional requirements

3.1.9.1 Introduction

This function allows the user of the system to add products for sale.

3.1.9.2 Inputs

user.u_key
category.c_key
Item Info

3.1.9.3 Processing

This function takes the user.u_key and retrieves the user record from the user table. It also gets item information from the user. It then saves this new item in the items table, it also saves image of the item if any, and sends an email to the user notifying him of his actions.

3.1.9.4 Outputs

Item Record
Image file
Email

3.1.10 ADMINISTRATOR functional requirements

3.1.10.1 Introduction

This function is a whole interface for the administrator. This is from where the “Flea Market” administrator will maintain the system.

3.1.10.2 Inputs

user.u_key
user.passwd

3.1.10.3 Processing

This function provides the System Administrator with an interface to manage/maintain system. The system admin has access to all the tables in the database and may change/update/delete/add users, items and properties of these entities. He/she will also have system administration options.

3.1.10.4 Outputs

Modifications to any/all tables in the DB

3.1 External Interface Requirements

3.2.1 User Interfaces

Any modern web browser including, but not limited to Microsoft Internet Explorer Version 4.0 and above and Netscape Navigator Version 4.0 and above, using display resolution of 800x600 pixels or greater. The prototypes of user interfaces could be viewed at:

<http://www.bungisoft.com/>

NOTE: At this time the prototype ONLY works properly with Internet Explorer 4.0 and above.

3.2.2 Hardware Interfaces

“Flea Market” system will not have a direct hardware access, all hardware interfacing will be done through the documented operating system routines.

3.2.3 Software Interfaces

“Flea Market” system will interface with the web server and RDBMS software using APIs provided by the implementation language. (See 2.4.3 – 2.4.5)

3.2.4 Communication Interfaces

All of the communication between client and server computers will be handled by the Web Server software, that may include, but is not limited to the protocols documented by the following:

RFC1939

RFC2920

RFC1867

3.3 Performance Requirements

3.3.1 Hardware Requirements

The host computer hardware and operating system should be able to handle minimum number of simultaneous users utilizing our system, which at this time is limited to 50, without significant performance degradation. This requires that with the minimum number of users accessing the system, load average of the system over 1 minute period should not be more than 1.00, where load average is defined as the average number of processes waiting in the job queue.

3.4 Design Constraints

3.4.1 Standards Compliance

The “Flea Market” system should only contain source code that complies with internationally accepted standards, if such standards are available.

3.4.1.1 It should be POSIX compliant

3.4.1.2 The system should comply with ANSI SQL standard
(See 2.4.4)

3.4.2 Hardware Limitations

If the usage of the “Flea Market” system should go beyond the acceptable performance level. There should exist an option to move the system to a more powerful computer, however this system is not designed to be used in a distributed computing environment.

3.4.3 Software Limitations

The “Flea Market” system only supports ANSI SQL compliant databases. Thus limiting the RDBMS software choices for a host company.

3.4.4 Portability Constraints

The “Flea Market” system should be designed in such a way, that porting this system to a foreign operating system/software environment should incur only minimal effort.

3.5 Attributes

3.5.1 Security

All transactions at the “Flea Market” require a valid user ID and password.

3.5.1.1 Secure data transfer over the Internet could be provided using SSL encryption that will be configured by the host system administrator.

3.5.1.2 The IP addresses where the transaction was made from will be logged.

3.5.1.3 User passwords will be stored in the database in an encrypted format.

3.5.1.4 During the registration process, users’ e-mail account will be verified for validity by contacting their e-mail server.

3.5.2 Maintainability

3.5.2.1 Detailed documentation of the code and SRS will enable easy maintenance of the “Flea Market”.

3.5.2.2 Most of the system functions will be controlled using the administrative interface.

3.6 Other Requirements

3.6.1 Database

See section 2.4.4

3.6.2 Operating System

See section 2.4.5

3.6.3 Interfaces with other 3rd party software

The “Flea Market” system should not need any external 3rd party software to operate, excluding web server (See section 2.4.3) and any tools for the language of implementation (i.e. if the system was designed to utilize interpreted language, interpreter for that language should be available on the host system.) However, if the system were designed to utilize native binaries for the host system, those pre-compiled binaries would be provided to the customer. Source code for the system may also be provided for the customer, depending on the licensing mode that we may chose.

3.6.4 Licensing

At the time of this writing it is our intension as developers of this software to provide it under the extended GPL licensing, for non-commercial use. (See attachment D). However, for commercial use of this application a commercial license will be required.

Attachment A
(Entity Relationship Diagram)

Attachment B
(Data Flow Diagrams)

Attachment C
(Data Dictionary)

Attachment D (License)