RentaHoliX

CS 16L2 Mini Project

12140816 ATHUL RAJ T S
12140820 DEON SUNNY
12140825 HARIKISHEN H
12140840 NARENDRAN T

B.Tech Computer Science & Engineering

Department of Computer Engineering
Model Engineering College, Thrikkakara
Kochi 682021
Phone: +91.484.2575370
http://www.mec.ac.in
hodcs@mec.ac.in

Model Engineering College, Thrikkakara
Dept. of Computer Engineering

C E R T I F I C A T E

This is to certify that, this report titled ***RentaHoliX*** is a bonafide record of the work done by

12140816 ATHUL RAJ T S
12140820 DEON SUNNY
12140825 HARIKISHEN H
12140840 NARENDRAN T

Sixth Semester B. Tech Computer Science & Engineering

students, for the course work in **CS 16L2 Mini Project**, under our guidance and supervision, in partial fulfilment of the requirements for the award of the degree, B. Tech Computer Science and Engineering of **Cochin University of Science & Technology**.

Guide

Vineetha K.V
Asst. Professor
Computer Engineering

Coordinator

Jaimon Jacob
Asst. Professor
Computer Engineering

Head of the Department

Manilal D L
Associate Professor
Computer Engineering

Acknowledgments

At this moment of accomplishment, as we are presenting our work with great pride and pleasure, we would like to express our sincere gratitude to all those who helped us in the successful completion of this venture. First of all, we would like to thank our Principal **Prof. Dr. V.P. Devassia** for providing us with a conductive environment and requisite lab facilities. We wish to express our sincere thanks to **Mr. Manilal D L** , Head of the Department, Computer Science and Engineering, for his inspiration and constant encouragement which made us take up the project and bring it to the completion. We are indebted to our project coordinator **Mr. Jaimon Jacob**, Assistant Professor, Department of Computer Science and Engineering, for his constant help and support. We extend our sincere and heartfelt thanks to our guide **Mrs. Vineetha K V** for helping us throughout the course of this project by providing us with valuable advice and suggestions. We also take this opportunity to specially thank our Parents and Friends for their motivation, encouragement and for being a constant support throughout. Above all we thank God Almighty for giving us the courage to move forward with confidence and for all his blessings throughout.

Athul Raj T S
Deon Sunny
Harikishen H
Narendran T

Abstract

This mini project aims at creating an e-Commerce portal for renting out products. Both people who intend to rent out their less used products for a few days, months, or years and those who intend to rent them can register in this website. The website provides facility to post details of products such as cameras, bikes, cycles, lawn mowers etc that are to be rented out. The buyers and renters are matched mainly based on location, period of rental, and expected rent amount. This website can be extremely useful in scenarios such as the following :- Suppose you are planning on a long business trip and have a bike which could be rented out , thereby being an extra source of income and the bike won't be damaged due to not using it for a long time.

Contents

Chapter 1

Introduction

RentaHoliX - This Online Product Rental Portal is a web based application for people to rent-out and rent-in products at the leisure of their homes. It provides a User-Friendly interface which can be accesed using a web browser. People can rent-out their stuff by providing details of the product such as sample images, expected rent, name, description etc. Users can rent-in products by logging in to the web app providing necessary details and by paying a initial deposit amount. Users need not register to browse through the various products available for renting, but for rent-out and rent-in sign-up/registration is necessary. Products can be searched for category-wise and also, results can be filtered based on location, rent, bonds etc.

Renting is a sensible way to enjoy life, while reducing our impact on the planet, our home. Every time you rent something its a product which doesnt get made, packaged, shipped and sold.

Chapter 2

Literature Survey

2.1 Existing Systems

In the existing systems, people have to buy products they want and sell them if they don't use them anymore. The issues regarding this is that even if someone wants a particular product for small periods of time they have to buy them. This may not be a problem for small things but can be expensive for larger products. In the case of selling, there is a large possibility of loss.

2.1.1 Limitations of Existing Systems

- Buying products is not a viable solution for temporary use.
- Selling products may lead to unprecedented loss.
- Buying/Selling is a one off process.
- A product which may be useful later but not now cannot be sold.

2.2 Proposed System

The proposed system is an online product rental portal with which a user can create an account, using which he/she can rent-out and rent-in products on per day/per week/per month basis. If he/she wants to rent out a product, he/she must provide the basic details of the product alongwith images, description, cost and the intended rent. If he/she wants to rent-in a product, he/she must place a rent request upon which an E-mail will sent to both renter/rentee. He/She can verify the product physically, place the deposit amount and approve it in the web app. He/She can only rent an available product. When the product is returned, the renter has to approve inorder refund the deposit amount. Also the renter can request a return for a rented product. Login is not necessary to browse/search products. Also, automated mails are sent for various actions such as on signup, rent-in, rent-out, posting complaints, adding products etc.

2.2.1 Advantages of Proposed System

- Renting is a good solution for short as well as long periods as the product can be returned and/or can be requested back if a need for it arises later on.

- Renting a product makes it a form of passive revenue.
- Renting products can be helpful for people relocating for a short span.
- Product will generally be in good shape as it doesn't stay unused for a prolonged interval.
- Unnecessary wastage of space with unusable products will be reduced.

Chapter 3

Proposed System

3.1 Problem Statement

To create an E-Commerce portal to facilitate renting out less used or spare products to willing people. It matches based on location, expected rent etc.

3.2 Proposed Solution

The proposed online portal consists of the following options :-

3.2.1 Admin and User Authentication

A privileged user can access the admin page with his/her username and password. This user has access to all user, product, and transaction details.
Other users must register/sign-up and login inorder to rent-out/rent-in products. But users must be able to browse products without log-in.

3.2.2 Posting Details

A user who wants to rent out a product must first register/login to the web app. The registered users must have an option to edit their personal details. The facility to post details of a product including short description, expected rent, images etc. Moreover, the renters must have an option to edit/remove the details of products they have rented out. Also, a post may be automatically removed on expiry of rental period specified by the renter.

3.2.3 Categorized Navigation

Grouping of products based on category and an interface to navigate between them.

3.2.4 Renting Products

When a user wants to rent a product, he/she can search for the product in the relevant sections. On finding the required product, a list of all available offers will be shown which can be further filtered based on location, rating, expected rent etc. The user can then select a suitable offer. He/she must

then proceed to put a request alongwith paying a deposit amount. On acceptance by the renter, the rentee will be contacted by the agency.

3.2.5 Automated Mail

An e-mail will be automatically generated and sent to the renter when a request for his/her product is received. A mail will also be sent to the rentee when the request has been accepted. A mail will also be generated to either the renter or the rentee when a complaint regarding a transaction has been submitted.

3.2.6 Report Issues

A rentee can report issues regarding a renter or the web app. The renter/rentee can report bugs or issues with the web app.

3.2.7 Input:

- User credentials for registration/login to the web app.
- Product details with images.
- Payment Details for deposit payment.
- Reviews/Complaints of products.
- Product Rating.
- Search string.

3.2.8 Output:

- User account details.
- Product details.
- Search results.
- Reviews of products.
- Product Rating.
- Transaction response.
- About Us.
- Help and Rules of Usage.

Chapter 4

Software Requirement Specification

4.1 Introduction

4.1.1 Purpose

The purpose of this document is to provide a detailed description of the Online Product Rental Portal. It explains the purpose and features of the application, it's interfaces, functions, the constraints under which it must operate and how the system will respond to external stimuli. This document is intended for both the users and the developers of the system.

4.1.2 Intended audience

This document is intended equally for both the developers, the users of the application, the staff and students.

4.1.3 Project Scope

This software provides an online interface for users to rentin and rentout products. Users can rentin products for shortterm usage rather than buying them. It can also be a means of passive income for the renters. Furthermore, a location based search module is provided in order to facilitate product rental in a specific location.

4.2 Overall Description

4.2.1 Product Perspective

The software is a new selfcontained product. It consists of three parts a web interface, a backend script and a database.

4.2.2 Product Functions

This software's main functions include providing an easy to use interface for users, transaction management, product rental, product addition, automated mail, user and admin authentication and a search function.

4.2.3 Operating Environment

The application is designed to run on all existing mainstream web browsers such as (Google Chrome, Mozilla Firefox etc).

4.2.4 Design and Implementation Constraints

- Expansion of the software onto a larger user base may lead to server failure.
- Uploading large size images may cause memory shortage.
- As the application involves payment and billing, certain security issues can arise.

4.2.5 Assumptions and Dependencies

It is assumed that the number of active users at any time will be less than the maximum number the server can handle. Also, it is assumed that one of the browsers specified in the operating environment section of this document will be used.

4.3 External Interface Requirements

4.3.1 Software Interfaces

A back end script is required to interface the web front end and the database. Also a script to interface the application and the payment gateway is required.

4.3.2 Communication Interfaces

The response of the computations done with the user input is an HTML page. Asynchronous data transfer from the frontend and backend is done using JSON format. The following protocols are used: HTTP, HTTPS. Also, the SMTP is used for automated emails.

4.4 Functional Requirements

4.4.1 User and Admin Authentication

The Privileged User has to enter the username and password to access the Admin page. Other users have to signup/login to use certain application features such as renting in and renting out.

4.4.2 Product Addition

The renter can post product details such as name, images, expected rent and a small description.

4.4.3 Product Rental

The rentee can rentin products according to his/her requirements. Products can be filtered on the basis of different parameters like rent, location etc.

4.4.4 Search

Users can search for specific products. Location based searches are also provided.

4.4.5 Transaction Management

Each transaction has 3 stages: Rented, Pending and Returned. Pending: The transaction moves to the pending stage whenever the product is rented or returned. Rented: The transaction moves from the pending state to the rented state when the rentee approves the product. Returned: The transaction moves from the pending state to the returned state once the renter approves the returned product.

4.4.6 Automated Mail

An automated mail is sent to the renter once a request for a product is made.

4.5 Nonfunctional Requirements

4.5.1 Performance Requirements

The minimum RAM requirement for proper and optimal working of the web server is 512MB. The maximum number of concurrent users is 256.

4.5.2 Safety Requirements

Not Applicable.

4.5.3 Security Requirements

All users are required to login in order to perform transactions. User passwords are encrypted before storing in the database.

4.5.4 Software Quality Attributes

Flexibility

The software is web based and is not environment dependent and hence can be used in windows or Linux environment with ease.

Maintainability

This software is easily maintainable as it is well documented. Future developers can refer this Software Requirements Specification Document to understand it's functionalities.

Usability

This application has a neat and clear web layout which can be used by all types of users. Also, it is mobile friendly, hence easily accessible.

4.6 Hardware & Software Requirements

4.6.1 Hardware Requirements

- PC with 2GB hard disk and 512MB RAM.

4.6.2 Software Requirements

- Frontend (Interface) : Web interface using HTML and JavaScript.
- Backend : PHP
- Database Management System : MySQL Version 14.14
- Web Server : Apache HTTP Server v2

Chapter 5

System Design

5.1 Block Diagrams

include figure if any. Else remove this subsection

5.1.1 Architectural Design

5.3 Algorithms

details about algorithm if you have any. else remove this section

5.3.1 Administrator

Purpose and algorithm description in correct format.

5.3.2 Renter

Purpose and algorithm description in correct format.

5.3.3 Rentee

Purpose and algorithm description in correct format.

5.3.4 Miscellaneous

Purpose and algorithm description in correct format.

5.4 Database Design

Provide the database design here like DB table, its description etc.

5.4.1 Database Tables

DB table details

5.4.2 ER Diagram

include diagrams with proper titles

5.5 Human Interface Design

Provide the database design here like DB table, its description etc.

5.5.1 Overview of User Interface

DB table details

5.5.2 Screen Images

include diagrams with proper titles

5.5.3 Screen Objects and Actions

include diagrams with proper titles

5.2 Dataflow Diagrams

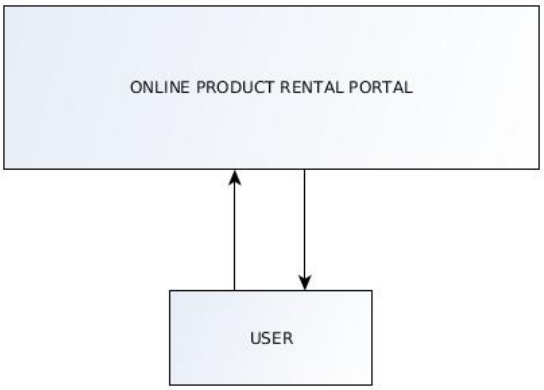

Figure 5.1: Level 0 DFD

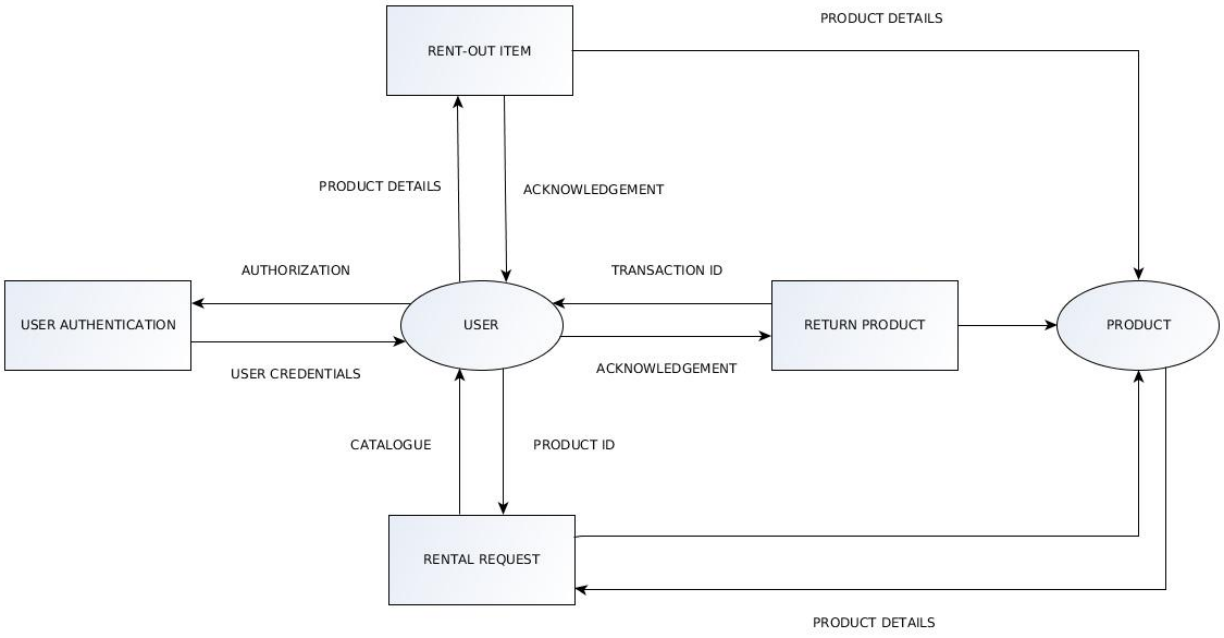

Figure 5.2: Level 1 DFD

Chapter 6

Implementation

6.1 Overview of Technologies Used

6.1.1 XAMP

XAMP is a free and open source cross-platform web server solution stack package, consisting mainly of the Apache HTTP Server, MySQL database, and interpreters for scripts written in the PHP and Perl programming languages. X stands for Windows/Linux/Mac etc.

6.1.2 MSMTP

msmtp is an SMTP(Simple Mail Transfer Protocol) client.In the default mode, it transmits a mail to an SMTP server (for example at a free mail provider) which takes care of further delivery. Features include:

- Sendmail compatible interface (command line options and exit codes)
- Support for multiple accounts
- TLS/SSL support including client certificates
- Many authentication methods
- Support for Internationalized Domain Names (IDN)
- Fast SMTP implementation using command pipelining
- DSN (Delivery Status Notification) support
- SOCKS proxy support

msmtp is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 3 of the License, or (at your option) any later version.

6.1.3 PHP

PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language.PHP code can be simply mixed with HTML code, or it can be used in combination with various templating engines and web frameworks. After the PHP code is interpreted and executed, the web server sends resulting output to its client, usually in form of a part of the generated web page.

6.1.4 MySQL

MySQL is the worlds second most widely used relational database management system (RDBMS) and most widely used open-source RDBMS.MySQL is a popular choice of database for use in web applications, and is a central component of the widely used XAMMP open source web application software stack.

6.1.5 Git

Git is a widely used source code management system for software development. It is a distributed revision control system with an emphasis on speed,[6] data integrity, and support for distributed, non-linear workflows. Git was initially designed and developed in 2005 by Linux kernel developers for Linux kernel development. As Git is a distributed version control system, it can be used as a server out of the box. Dedicated Git server software helps, amongst other features, to add access control, display the contents of a Git repository via the web, and help managing multiple repositories. Bitbucket was the git client used for this project.

6.1.6 Web Frontend Technologies

6.1.6.1 HTML

HyperText Markup Language, commonly referred to as HTML, is the standard markup language used to create web pages. Along with CSS, and JavaScript, HTML is a cornerstone technology used to create web pages, as well as to create user interfaces for mobile and web applications. Web browsers can read HTML files and render them into visible or audible web pages. HTML describes the structure of a website semantically along with cues for presentation, making it a markup language, rather than a programming language.

6.1.6.2 CSS

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language. CSS is designed primarily to enable the separation of document content from document presentation, including aspects such as the layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple HTML pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

6.1.6.3 JavaScript

JavaScript is a high-level, dynamic, untyped, and interpreted programming language. It has been standardized in the ECMAScript language specification. Alongside HTML and CSS, it is one of the three essential technologies of World Wide Web content production; the majority of websites employ it and it is supported by all modern Web browsers without plug-ins. JavaScript is prototype-based with first-class functions, making it a multi-paradigm language, supporting object-oriented, imperative, and functional programming styles. It has an API for working with text, arrays, dates and regular expressions, but does not include any I/O, such as networking, storage, or graphics facilities, relying for these upon the host environment in which it is embedded.

6.1.6.4 jQuery

jQuery is a cross-platform JavaScript library designed to simplify the client-side scripting of HTML.[2] jQuery is the most popular JavaScript library in use today. jQuery is free, open-source software licensed under the MIT License.

6.1.6.5 Materializecss

Created and designed by Google, Material Design is a design language that combines the classic principles of successful design along with innovation and technology. Google's goal is to develop a system of design that allows for a unified user experience across all their products on any platform. Materializecss is a modern responsive front-end framework based on Material Design.

6.2 Testing

6.2.1 Types of Testing

6.2.1.1 Alpha Testing

We populated the database with different users and products (along with other resources such as images etc.) and then tried carried out the different operations such as renting items out and renting them in. The working of automatic mail was checked and the entire flow of transaction was also verified.

6.2.1.2 Beta Testing

The product was distributed among a selected group of individuals and tested for any possible bugs. Administration using the admin dashboard was also done in this phase.

6.3 Results

Chapter 7

Conclusion

7.1 Conclusion

With this product rental portal implemented, a lot of unused products can be turned into a form of passive revenue. Also the costs of buying, shipping and packaging a new product can be reduced.

7.2 Future Scope

This product can be upgraded in the future to include a feature to provide various other services like plumbers, electricians, gardeners etc. for daily wages managed by the web app.

References

[1] "XAMPP Installers and Downloads for Apache Friends "[Online].
Available: $https://www.apachefriends.org/$

[2] "PHP: Hypertext Preprocessor "[Online].
Available: $http://www.php.net/$

[3] "Materialize design in css "[Online].
Available: $http://materializecss.com/$

[4] "Msmtp: An SMTP Client "[Online].
Available: $http://msmtp.sourceforge.net/$