**Notify Me**

A Dissertation submitted

for the partial fulfillment of the degree of

**Bachelor of Engineering in**

**Computer Engineering**

**(Session 2019 - 2020)**

**Guided By: Submitted By:**

**Miss Harshita Sharma Mayank Singh Parmar (16C7129)**

**Prajjwal Gavander (16C7139)**

**Ravi Patidar (16C7146)**

**Department of Department name**

**Institute of Engineering & Technology**

**Devi Ahilya Vishwavidyalaya, Indore (M.P.)**

**(**[**www.iet.dauniv.ac.in**](http://www.iet.dauniv.ac.in/)**)**

**December 2019**

**Dissertation Approval Sheet**

The dissertation entitled **“Notify Me”** submitted by **MAYANK SINGH PARMAR, PRAJJWAL GAVANDER, RAVI PATIDAR** is approved as partial fulfillment for the award of **Bachelor of Engineering in Computer Engineering** degree by **Devi Ahilya Vishwavidyalaya, Indore**.

**Internal Examiner External Examiner**

**Director**

**Institute of Engineering & Technology**

**Devi Ahilya Vishwavidyalaya,**

**Indore (M.P.)**

**Recommendation**

The dissertation entitled “**Notify Me”** submitted by **“Mayank Singh Parmar, Prajjwal Gavander, Ravi Patidar”** isa satisfactory account of the bonafide work done under my supervision is recommended towards the partial fulfillment for the award of **Bachelor of Engineering in Computer Engineering** degree by **Devi Ahilya Vishwavidyalaya, Indore.**

**Date: Name of Project Guide**

Miss Harshita Sharma

**.**

**Endorsed By:**

**Dr. G.L. Prajapati**

**Head, Department of Computer Engineering**

**Candidate Declaration**

We hereby declare that the work which is being presented in this project entitled Project Name in partial fulfillment of degree of Bachelor of Engineering in Name of the Branch is an authentic record of our own work carried out under the supervision and guidance of **Miss Harshita Sharma** in Department of **Computer Engineering**, Institute of Engineering and Technology, Devi Ahilya Vishwavidyalaya, Indore

We are fully responsible for the matter embodied in this project in case of any discrepancy found in the project and the project has not been submitted for the award of any other degree.

**Date:**

**Place: Mayank Singh Parmar**

**Prajjwal Gavander**

**Ravi Patidar**

**ACKNOWLEDGEMENTS**

We wish to express our sincere thanks to **Miss Harshita Sharma**, our Guide in this project, for providing us with all the facilities for implementation of the respective project.

We place on record, our sincere thanks to the classmates, for continuous encouragement.

We are also grateful to **Dr. G.L. Prajapati**, HOD in the Department of Computer Engineering. We are extremely thankful and indebted to him for sharing expertise, and sincere and valuable guidance and encouragement extended to us.

We thank the Director of Academics for providing us with the compatible environment to develop the project.

We also place on record, our sense of gratitude to one and all, who directly or indirectly, have lent their hand in this venture.

**ABSTRACT**

E-commerce (Electronic Commerce) is process of doing business through computer networks. The primary goal of an e-commerce site is to sell goods and services online. Online shopping is a form of electronic shopping store where the buyer is directly online to the seller’s computer usually via the internet. A person sitting on his chair in front of a computer can access all the facilities of the Internet to buy or sell the products. The products available on this ecommerce website have dynamic pricing. Generally, it is seen that in festive season, price are usually lower. If a person want to buy a particular product at his desired price, then he need to keep on checking price regularly.

The proposed system will allow its users to get rid of this above mentioned problem. It gives email notification alert to its registered users when product’s price goes below budget price. It is just like click and forget system where user once add product url and their required deal price, rest work is done by our system. When product’s price is less than this required deal price user will get notified along with product page url.

## TABLE OF CONTENTS Page No

|  |  |  |
| --- | --- | --- |
| **Dissertation Approval Sheet** | | **I** |
| **Recommendation** | | **II** |
| **Candidate Declaration** | | **III** |
| **Acknowledgements** | | **IV** |
| **Abstract** |  | **V** |
| **List of Figures** | | **VII** |
| **Chapter 1** | **Introduction** | **1** |
| 1.1 | Overview and issues involved | 1 |
| 1.2 | Problem Definition | 1 |
| 1.3 | Proposed Solution | 2 |
| **Chapter 2** | **Literature Survey** | **4** |
| 2.1 | Methodology | 4 |
| 2.2 | Technologies and Tools | 4 |
| 2.3 | Existing Solutions | 4 |
| **Chapter 3** | **Analysis** | **6** |
| 3.1 | Software Requirements | 6 |
| 3.2 | Hardware Requirements | 6 |
| 3.3 | Use Case Model | 7 |
| 3.4 | Use Case Description | 7 |
| **Chapter 4** | **Design** | **12** |
| 4.1 | Technology Section | 12 |
| 4.2 | Sequence Diagrams | 12 |
| 4.3 | Activity Diagrams | 18 |
| **Chapter 5** | **Implementation and Testing** | **21** |
| 5.1 | Dependencies | 21 |
| **Chapter 6** | **Conclusion** | **22** |
|  | **References** | **23** |
|  | **Glossary** | **23** |

**List of Figures**

|  |  |  |
| --- | --- | --- |
| **Figure No.** | **Figure Name** | **Page No.** |
| Fig 3.3.1 | Usecase diagram | 14 |
| Fig 4.2.1 | Register and Login Sequence Diagram | 19 |
| Fig 4.2.2 | Search Sequence Diagram | 20 |
| Fig 4.2.3 | Show Categories Sequence Diagram | 20 |
| Fig 4.2.4 | Profile Sequence Diagram | 21 |
| Fig 4.2.5 | Sell Sequence Diagram | 21 |
| Fig 4.2.6 | Cart Sequence Diagram | 22 |
| Fig 4.2.7 | Current Orders Sequence Diagram | 22 |
| Fig 4.2.8 | Logout Sequence Diagram | 23 |
| Fig 4.2.9 | FAQs Sequence Diagram | 23 |
| Fig 4.2.10 | Add, Remove and Update Product Sequence Diagram | 23 |
| Fig 4.2.11 | Purchase Sequence Diagram | 24 |
| Fig 4.2.12 | Register Outlet Sequence Diagram | 24 |
| Fig 4.3.1 | Register and Login Activity Diagram | 25 |
| Fig 4.3.2 | Search Activity Diagram | 26 |
| Fig 4.3.3 | Cart Categories Activity Diagram | 26 |
| Fig 4.3.4 | Add, Remove and Update Product Activity Diagram | 26 |
| Fig 4.3.5 | Current Orders Activity Diagram | 26 |
| Fig 4.3.6 | Profile, FAQ, Logout Activity Diagram | 27 |
| Fig 4.3.7 | Sell Activity Diagram | 27 |
| Fig 4.3.8 | Register Outlet Activity Diagram | 27 |

**Chapter – 1 Introduction**

* 1. **Overview and Issues involved**

Ecommerce is one of the fastest growing industry in India since past five years. We have a huge userbase that uses internet on daily basis and this keeps on increasing. Out of these, about 6 million new users keep on adding to India’s ecommerce giants which includes Amazon, Flipkart, Myntra, Paytm on monthly basis. But why these ecommerce industries increasing rapidly? This question can have many answer but major factor affecting this answer is their deals and price. In festive season, this platform gives huge discounted offers and good budgeted price. The price of the products available on this ecommerce platform is dynamic. Sometimes it’s higher than our budget and sometimes lower. If we want to buy some products of our need from this online platform at our budget, so who will help us? Here, Notify Me comes into picture. It gives you a email notification of your desired product when it’s deal go handy at your budget price.

## Problem Definition

Popular ecommerce websites & applications have dynamic pricing for their products. This means that once in a month or year price of a product is lowest. If an user wishes to buy a product at his/her budget price then he/she needs to keep checking price regularly. There is no available notification system which notifies us that your product is available at your budget price. This ecommerce platform notification system keeps you reminding other offers, deals etc. Even when you add product in cart they don’t notify us about product’s current price which you need to check again and again. Our project focuses on creating a Django based website that helps you to get rid of this problem.

Objectives: -

1. Our project help user so that user does not need to check product price regularly
2. Provide email notification on daily basis if price drops by budget of the user
3. Can check price drop for multiple products
4. Multiple ecommerce website available for notification like Amazon, Flipkart, Snapdeal.
5. User can check number of products added for price drop alert.
6. Make an easy and interactive web application

## Proposed Solution

Our proposed solution plan consists of making a web application that is made using Django framework. It has following functionalities: -

* Register: New user can register themselves on web application.
* Login: Registered user can login themselves.
* Home: Showcase use of website.
* Add new product: Authenticated user can add new product for price check alert.
* List of products: List out all products details that is on price check alert.
* Contact us: Here user can give feedback.

Technology stack used in our web application:-

* HTML: HTML is used in making web pages for our web application.
* CSS: CSS is used in aligning textual data and keeping GUI clean.
* Django: It is a python based full stack web development framework which is used in developing our web application.
* Posgre: Postgre is an open source object-relational database that contains product’s whose price drop alert need to be checked in tabular form.

Modules used in our web application:-

* Django: Django module is imported in our web application. It is used in interacting with database and it also creates an overall structure of web application. Django creates various python files which are views.py, urls.py, models.py. These files is used in providing response to request by client, maintaining urls of various html pages involved and migrating data from web application to database respectively.
* Selenium: Selenium module is imported in our web application. It is used in web scrapping of product details from various ecommerce websites. It also used in browser integration.
* Date and time: Date and time module is imported in our web application which is used in maintaining timestamp and notification alert delay.

**Chapter - 2**

**Literature survey**

## Methodology

After analysis of the problem domain we gathered all the requirements and system specification needed for our project. It includes study of all the available solutions and the gap which were removed through our project.

Once the objectives and the features were known, break each element into deliverable and lay down a strategy to go ahead with. Work on designing part (UI/UX) and prepare a design that delivers the best user experience. Test it on different devices.

While working on designing, Start implementing back-end Processes. Deploy each functionality one by one and keep testing it. Firstly, the basic visual layout of the application will be developed on Visual Studio Code. Later the application will be linked to server.

Test the functionalities and make necessary changes. Ask dummy users to use the application and make necessary changes according to their inputs.

## Technologies and Tools

* HTML: HTML is used in making web pages for our web application.
* CSS: CSS is used in aligning textual data and keeping GUI clean.
* Django: It is a python based full stack web development framework which is used in developing our web application.
* Posgre: Postgre is an open source object-relational database that contains product’s whose price drop alert need to be checked in tabular form.
* Editor/IDE used: Visual Studio Code.

## Existing Solutions

To develop our web application, we researched various websites and ecommerce email notifier, very few are working in this field right now. The resources which we used as a starting point for our project with their advantages and drawbacks are as follows:

* **Camelcamelcamel.com** provides price drop alert for products available on amazon. It is also showcases wide range of popular products with dropped price.

Advantages: - Clean GUI based website with extension for browser integration.

Drawbacks: - Works only for amazon product where price of product is in dollar. Hence, not available for Indian users.

* **Keepa.com** provides price drop alert for products available on amazon.

Advantages: - Provides chart representation of history of price drop of product.

Drawbacks: - To unlock its full feature like chart representation you need to buy premium membership. Just like camelcamelcamel.com it also works in US region, hence not applicable in India.

Through python official documentation[1] and book “Adventures in Python” by Craig Richardson[5], we have gained information to use various module needed to make our project. Using this documentation, we get to know deeply about various python modules like Date & Time, OS modules, System module, Django, Selenium which we are used in our project.

Through Django official documentation[2] we learnt how to use Django framework and create web based application.

Through selenium official documentation[3] we learnt how to use selenium module for web scrapping. It allowed us to fetch price of product from different ecommerce websites and compare it with the user’s listed budget price.

Through Postgresql official documentation[4] we learnt how to use postgre as our database management tool. We are able to manage tabular data which is submitted by user’s product listing

Stackoverflow[7]: It was a go through website for debugging and error handling.

**Chapter – 3**

**Analysis**

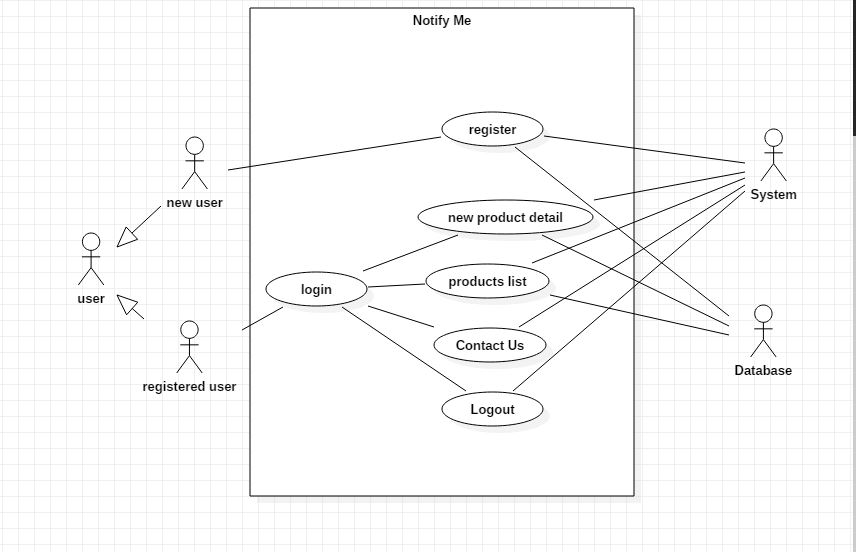
## Software Requirements

* + - Operating system:- Windows 7 or later, open source linux distribution like ubuntu
    - Python 3.x installed on system.
    - Python Interpreter
    - PyCharm, Visual Studio Code or any other python IDE.
    - Chrome version – latest (72.0.3626.121)
    - Firefox version – latest (68.0.2)
    - Internet Explorer version – 9 or later
    - Internet Explorer version (for mobile) – 11

**3.2 Minimum Hardware Requirements:**

* Processor – Intel Atom® processor or Intel® Core™ i3 processordual or i3 processor.
* Disk space – 3 GB
* Memory – 1 GB RAM

## Use Case Model



### **Fig. 3.3.1. Usecase diagram**

* 1. **Use Case Description**

1. **Register**
   * **Actors-** User
   * **Basic flow-** opening the registration page, filling the required details, registered
   * **Special requirements-** System, internet connection

### **Pre-conditions-** **none**

* + **Post-conditions-** The registration may or may not be done. If not done the user is redirected to the registration page.

### **Login**

* + **Actors-** User, Seller
  + **Basic flow-** opening the login window, filling user id password, logged in
  + **Special requirements-** System, internet connection
  + **Pre-conditions-** should be already registered
  + **Post-conditions-** you may or may not logged in. If not then the user will be redirected to login window.

### Search

* + **Actors-** user, seller
  + **Basic flow-** open the web application, go to the Search bar, find the products/ services
  + **Special requirements-** System, internet connection

### Pre-conditions- **none**

* + **Post-conditions-** you may or may not find the required product or service.

### ShowCategories

* + **Actors-** user, seller
  + **Basic flow-** open the web application, show categories
  + **Special requirements-** System, internet connection

### Pre-conditions- **none**