

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/339197695>

A Django Based Educational Resource Sharing Website: Shreic

Article in Journal of Scientific Research · January 2020

DOI: 10.37398/JSR.2020.640134

CITATIONS

22

READS

11,181

2 authors:



[Adamya Shyam](#)

University of Delhi

3 PUBLICATIONS 23 CITATIONS

SEE PROFILE

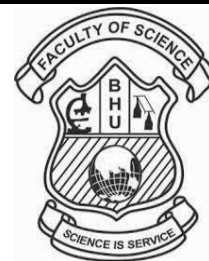


[Nitin Mukesh](#)

Here technologies

2 PUBLICATIONS 22 CITATIONS

SEE PROFILE



A Django Based Educational Resource Sharing Website: Shreic

Adamy Shyam^{*1}, Nitin Mukesh²

^{*1}Department of Computer Science, Banaras Hindu University. adamyshyam2016@gmail.com

²Department of Mathematics, Indian Institute of Technology, Mumbai. nitinmukesh681@gmail.com

Abstract: Technological Implementations in the field of Academics has helped Students as well as Professionals in very important ways. The availability of all educational resources helps the students a lot in their educational life. The paper illustrates a website model with the help of which Students can be able to access class notes, previous year question papers, syllabus, and can sell their old books from the same digital platform as well. The paper also describes the role of software engineering in project development. The project is developed on Django Framework; the backend development is in Python, Jinja2 and SQLite. The frontend consists of HTML, CSS and Java. Appropriate SDLC Model and Testing techniques have been used in the development process. Each step of the SDLC Model (Iterative Model) is described thoroughly and respective ER Diagrams and Flow Charts have been shown. The project developed is highly efficient, user-friendly and simple.

Index Terms: Software Engineering, Django, HTML, CSS, SDLC, Python, Testing.

I. INTRODUCTION

In this era of digitalization, the availability of different resources on different digital devices is making our lives easy and convenient. In the field of academics too, there are many educational websites that share educational resources like short notes, video lectures, presentations and books etc. But, none of them provide class notes or question papers related to courses taught in a particular college or university; and most of the students due to lack of communication from batch mates or seniors, often face the problem of not getting class notes, if were absent or previous year question papers from seniors. Taking these problems under consideration, the idea of an online

platform for resource sharing purposes was developed named Shreic (*Sharing Resources In Campus*). The idea is to provide all class notes and previous year question papers of different courses of a college/university at a single platform.

Also, in colleges after completion of the course, students either sell their books to junkyards or few of them donate it to their juniors. Only a few of them keep those books with themselves. So, the website also works like an e-commerce website where old books can be sold or donated as per the wish of the seller.

The project is developed in the Django framework. The backend consists of Python, Django, SQLite and Jinja2. Cloudinary, an online cloud service has been used for storing data. Finally, black box and white box testing were done in order to test the functional, structural and logical features of the website.

The development process followed the Iterative Model of Software Development. The idea was to add functionality and then to design it, test it and implement it. Although this approach takes more resources but with each iteration the next iteration takes less time to be developed and with the help of this approach, errors were easily found and rectified at the same time.

The feasibility study helped us to enlist the main objectives viz.

1. Different Educational Categories (viz. Entrance, Recruitment, Academics and Entertainment).
2. Different Subcategories within each Educational Category.
3. Each Subcategory will contain respective Old Books, Question Papers and Class Notes.
4. Each User can add Class Notes, Old Books or Question Papers which once verified by Admin Users will be added on the website.

^{*} Corresponding Author

5. Old Books can be purchased via Cash on Delivery Method.
6. A Chat-Box for communication between Customers and Sellers.
7. The User will see resources relevant to its own University only.

The testing of the project was done in two ways viz. Black Box Testing which was done by Users. The Users were asked to run the project and check all of the features. The feedback was recorded and amendments were made as required; and White Box Testing in which different test cases were made for each unit of source code and were tested. For each test case, the desired output was expected. When the desired output was not encountered it led to a bug. Each error was removed from the source code and all units were integrated at last.

The project is developed under the Django Python Web Framework. It encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of Web development, so we can focus on writing our app without needing to reinvent the wheel. The front end of the project has been made user-friendly with efficient use of HTML, CSS and JavaScript. The back end has been coded in Python Programming Language. We easily achieve more functions with fewer lines of code using Python. Web development with Python is very popular because of its readability and efficiency.

Python is used for the development process taking the security issues in concern. Python is more secure than several widely used programming languages. Django further helps enterprises to enhance the security of their websites and web applications by preventing a variety of security attacks — cross-site scripting (XSS), cross-site request forgery (CSRF), SQL injection, and clickjacking. The web application is made to exchange data with the webserver securely by deploying the web application behind HTTPS. The security issues are resolved in order to prevent any unwanted attack on the database as well as the server.

The project after development was deployed on a public server with the domain www.shreic.com. The website is very user-friendly and easy to operate. This project will help students a lot in their educational life.

II. LITERATURE SURVEY

Various websites and research papers have been developed keeping the idea of educational resource sharing and its importance in mind. Few of them are listed here:

1. Used Books Factory:
It is an online platform to sell old books of different categories.
2. Vioric-Torri, C. & Alexandrache, C. (2012):
The study reflects how educational technology influences the learning styles of students and how to form and develop the competences of learning in the new generations.

3. TutorialsPoint:
The website provides tutorials on different topics related to computer science and technology. Provides pdf notes for the same and also provides guidance for competitive exams.
4. The Physics Classroom:
For PDF notes and tutorials related to the various fields of Physics.
5. Kelly, L., & Breault, K. (2006):
The objective of the research project was to provide the Australian Museum with guidance on how to best develop a website that meets the needs of students and teachers in the primary and secondary levels across a range of curriculum areas. General objectives were to gain insights into how students and teachers are using the internet and what they are looking for when they access websites.
6. Aglasem:
Online Portal that provides previous year question papers and answer keys related to different competitive exams and some universities' semester papers.
7. BHU Student Club:
It is an online social group that provides old semester papers of a few courses that are offered at Banaras Hindu University.

After a brief study of the related works, it has been observed that all these websites possess different functions of the project proposed but none of them have all of the features collectively. Also, there is no such website where sharing class notes can be done except for social media. The following project has been developed keeping all these disadvantages in mind.

III. PROPOSED APPROACH

A. Software Development Life Cycle

SDLC is a process that defines the various stages involved in the development of software for delivering a high-quality product. SDLC stages cover the complete life cycle of software i.e. from inception to retirement of the product. The purpose of SDLC is to deliver a high-quality product which is as per the customer's requirement.

SDLC has defined its phases as Requirement gathering, Designing, Coding, Testing, and Maintenance. It is important to adhere to the phases to provide the Product in a systematic manner.

B. SDLC Model

A software life cycle model is a descriptive representation of the software development cycle. The software development model helps the developer to select a strategy to develop the software. A software development model has its own set of tools, methods and procedures, which are expressed clearly and defines the software development life cycle. This project has been developed using the Iterative model (Jalote, 2003).

In this life cycle model, a Project Control List (PCL) on the basis of current known requirements is developed. A PCL is a list containing the series of tasks/functionalities that are to be present in the given system. If at a certain phase of development, we come across any new requirement, we add it to our Project Control List.

For developing the website, a task is chosen from the given PCL and Planning, Analysis, Designing, Testing and Evaluation is performed as shown in Figure 1. When the specific functionality is added we remove it from the Project Control List. In a similar way, one task at a time from PCL is chosen, implemented and then removed from PCL. This process iterates

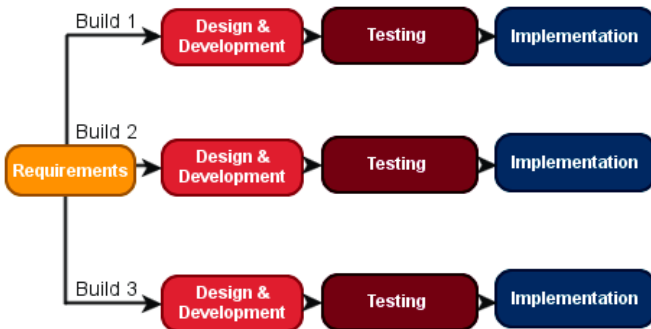


Fig. 1. Iterative Model

until the desired requirements of the product are not met.

After each iteration, the management team can do work on risk management and prepare for the next iteration. Because a cycle includes a small portion of the whole software process, it is easier to manage the development process.

In the Iterative model, the newer iterations are incrementally improved versions of previous iterations. Moreover, in the event that a new iteration fundamentally breaks a system in a catastrophic manner, a previous iteration can quickly and easily be implemented or “rolled back,” with minimal losses, which is a boon for post-release maintenance.

In the Iterative Model, the initial run-through of all stages may take some time, but each subsequent iteration will be faster and faster, allowing the life cycle of each new iteration to be trimmed down to a matter of days or even hours in some cases.

C. Feasibility Study

A feasibility analysis is used to determine the viability of an idea, like ensuring a project is legally and technically feasible as well as economically justifiable.

During the development of this project, the feasibility study was done as follows:

1) Project Requirements

The following objectives were proposed in order for the successful development of the project.

- User Registration
- User Login

- Administrator Login
- Different Educational Categories (viz. Entrance, Recruitment, Academics and Entertainment).
- Different Subcategories within each Educational Category.
- Each Subcategory will contain respective Old Books, Question Papers and Class Notes.
- Separate Account Page for each User.
- Each User can add Class Notes, Old Books or Question Papers which once verified by Admin Users will be added on the website.
- Administrators can Add or Delete Resources if desired.
- Old Books can be purchased via Cash on Delivery Method.
- A Chat-Box for communication between Customers and Seller.
- The User will see resources relevant to its own University only.
- A Dynamic Search Bar.

This requirement list was also used as the Project Control List during development.

Certain goals regarding the efficiency of the project to be developed were also proposed, which are as follows:

- **Planned Approach:** The working of the website is well planned and organized. The data will be stored properly in data stores, which will help in the retrieval of information as well as its storage.
- **Accuracy:** The level of accuracy in the proposed system will be higher. All operations would be done correctly and it ensures that whatever information is retrieved or stored is accurate.
- **Reliability:** The reliability of the proposed system will be high due to the above stated reasons. The reason for the increased reliability of the system is that now there would be proper storage of information.
- **No Redundancy:** In the proposed system utmost care would be taken so that no information is repeated anywhere, in storage or otherwise. This would assure the economic use of storage space and consistency in the data stored.
- **Immediate retrieval of information:** The main objective of the proposed system is to provide quick and efficient retrieval of information regarding users, orders, products etc.
- **Easy to Operate:** The system should be user-friendly and should be such that it can be developed within a short period of time and fit in the limited budget of the organization.

D. E-R Diagram

The project consists of many relational models. Every model consists of different attributes. The Primary Key is shown with

an underline under the name. Each Relation is characterized by Structural Constraint where the first number denotes Participation Constraint (1 for Total Participation and 0 for Partial Participation) and the second number denotes the Cardinality Ratio (1:1 or 1: N or M: N).

1) Participation Constraint:

Total Participation: If each entity of an Entity Type has a relationship instance in Relationship Set then the participation is total.

Partial Participation: If few entities of an Entity Type have a relationship instance in Relationship Set then the participation is partial.

2) Cardinality Ratio:

1:1: Only one entity of an Entity Set can be related to anyone entity of the other Entity Set.

M: N: Many entities of an Entity Set can be related to many entities of the other Entity Set.

1: N: One entity of an Entity Set can be related to any number of entities of the other Entity Set.

There are 3 ER Diagrams that are used in the development process. The ER Diagram in Figure 2 illustrates the relation of MyUser Model with different Models related to user details. The Model MyUser is connected to Models Gender, College, City, State, Country and User. The Model Country is connected to Model State and Model State is connected to Model City.

The ER Diagram shown in Figure 3 describes the relation of MyUser Model with different Models related to book and note details. The Model MyUser can add BookDet to the database. MyUser can request for book which will be recorded in BookReq Model. MyUser can add Notes to NotesDet Model. Book will be of a Category and corresponding to it there will be Subcategory I and Subcategory II. Notes also have Category, Subcategory I and Subcategory II.

And, the last one as shown in Figure 4, illustrates the relation of MyUser Model with different Models related to chats, notification and orders. The Model MyUser can chat with another MyUser in Chatroom. ChatW will create a ChatRoom for MyUser if is not created yet. All notifications that notifies MyUser is in Notification. MyUser can add products to ShoppingCart in the Session logged in. MyUser can order an Order which will be delivered by another instance of MyUser.

IV. IMPLEMENTATION

A. Technologies Used

Various front-end and back-end technologies are available in this era of digitalization. The technologies used in this project are discussed briefly in the following sections.

1) Front End Technologies

a) HTML

HTML stands for Hypertext Markup Language, and it is the most widely used language to write Web Pages. Hypertext refers to the way in which Web pages (HTML documents) are linked together. Thus, the link available on a webpage is called Hypertext. As its name

suggests, HTML is a Markup Language which means you use HTML to simply "mark-up" a text document with tags that tell a Web browser how to structure it to display (Musciano & Kennedy, 1996).

Originally, HTML was developed with the intent of defining the structure of documents like headings, paragraphs, lists, and so forth to facilitate the sharing of scientific information between researchers. Now, HTML is being widely used to format web pages with the help of different tags available in HTML language.

b) CSS

CSS (Powell, 2010) stands for Cascading Style Sheets. CSS describes how HTML elements are to be displayed on the screen, paper, or in other media. CSS saves a lot of work. It can control the layout of multiple web pages all at once.

c) JavaScript/JQuery

JavaScript (JS) is a high level, interpreted programming language. JavaScript has curly-bracket syntax, dynamic typing, prototype-based object-orientation, and first-class functions.

Alongside HTML and CSS, JavaScript is one of the core technologies of the World Wide Web (Flanagan, 2006). JavaScript enables interactive web pages and is an essential part of web applications. The vast majority of websites use it, and major web browsers have a dedicated JavaScript engine to execute it.

JavaScript supports event-driven, functional and imperative (including object-oriented and prototype-based) styles. It is important to validate the form submitted by the user because it can have inappropriate values. So, validation is must to authenticate the user. JavaScript provides the facility to validate the form on the client-side so data processing will be faster than server-side validation.

d) Bootstrap

Bootstrap (Shenoy & Sossou, 2014) is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.

To use bootstrap, we are required to either install in our system or use CDN. CDN is short for content delivery network. A CDN is a system of distributed servers that deliver pages and other web content to a user, based on the geographic locations of the user, the origin of the webpage and the content delivery server.

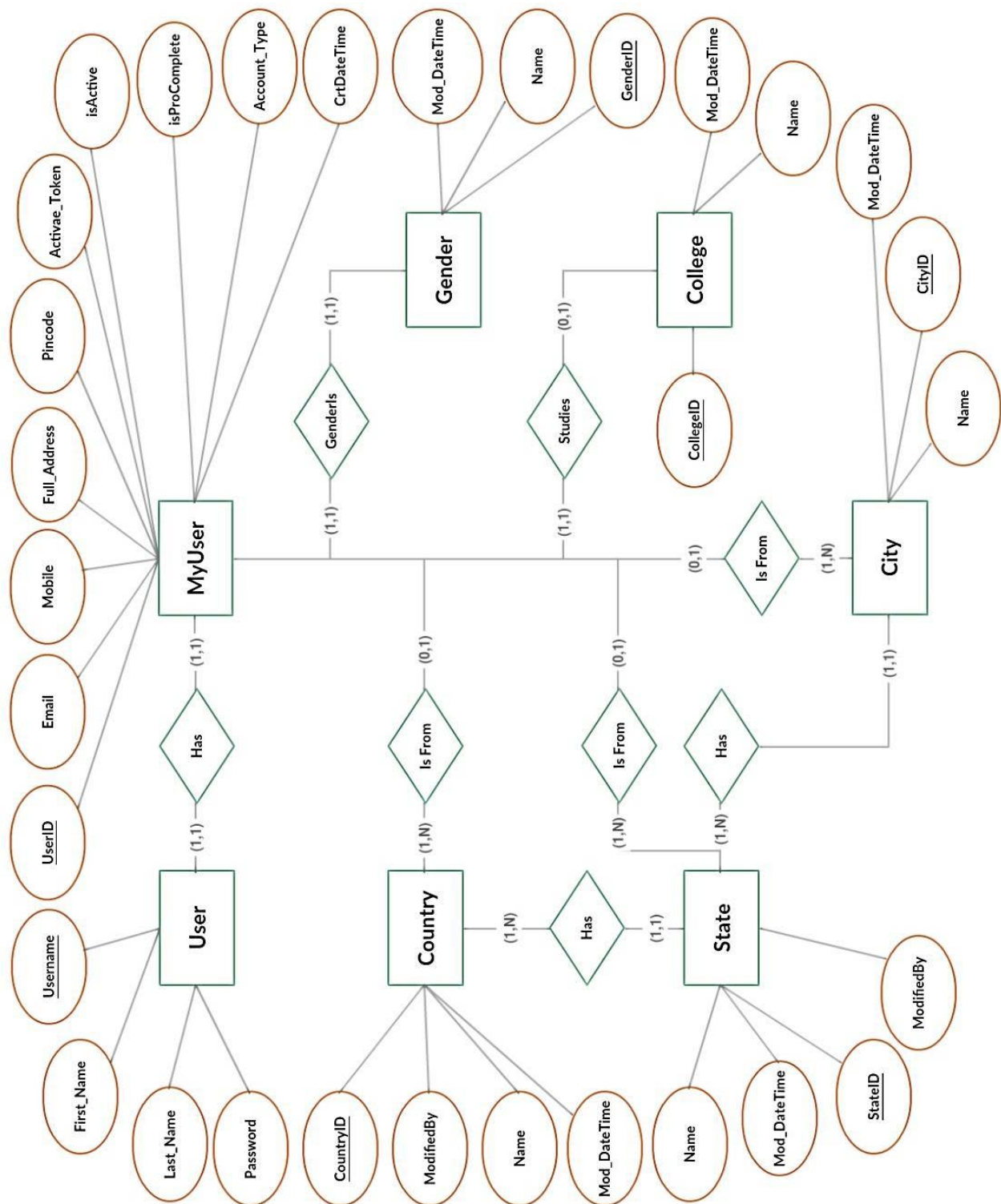


Fig. 2. ER Diagram for MyUser model and Personal Details' Models

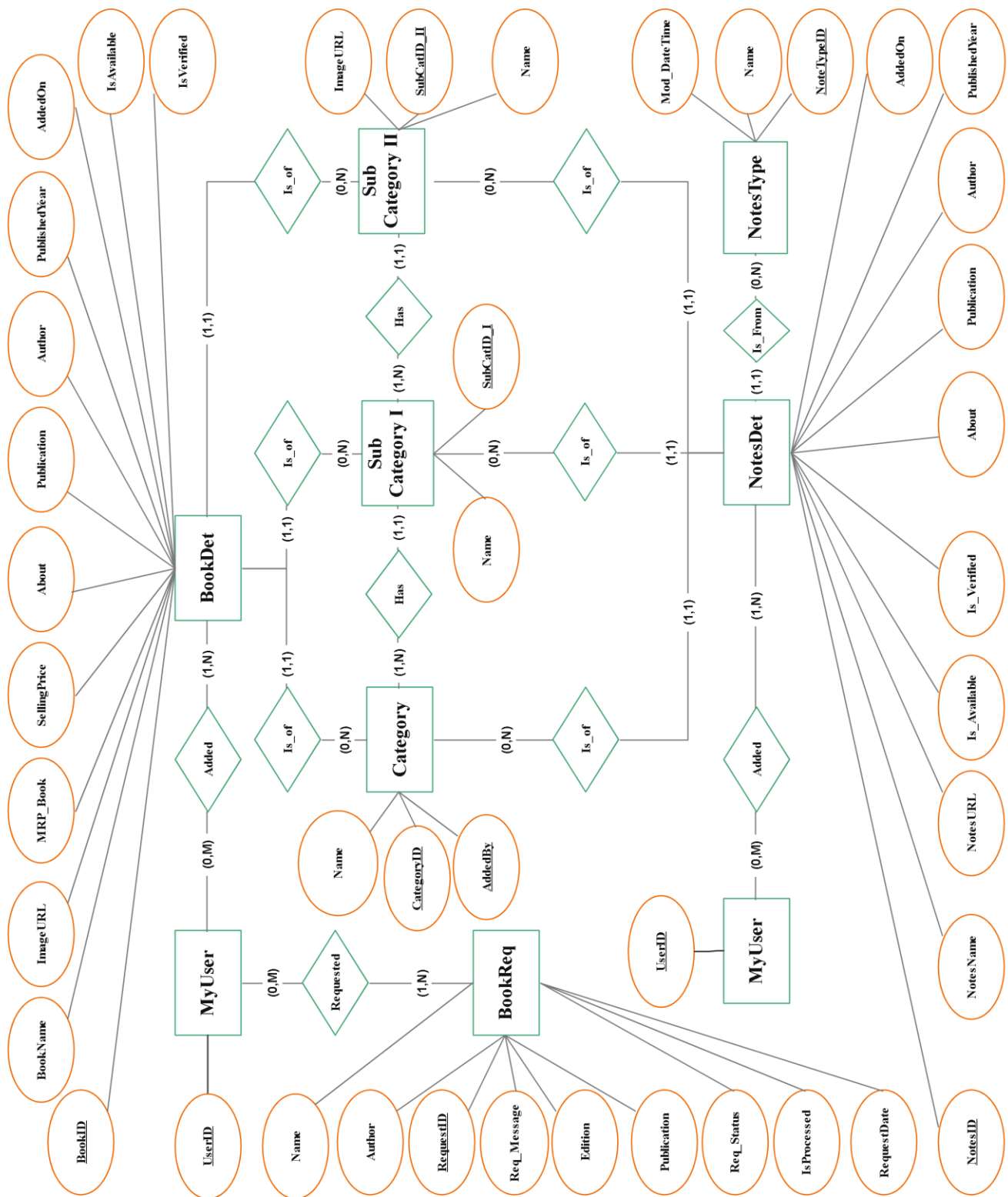


Fig. 3. ER Diagram for Book Details and Note Details Models

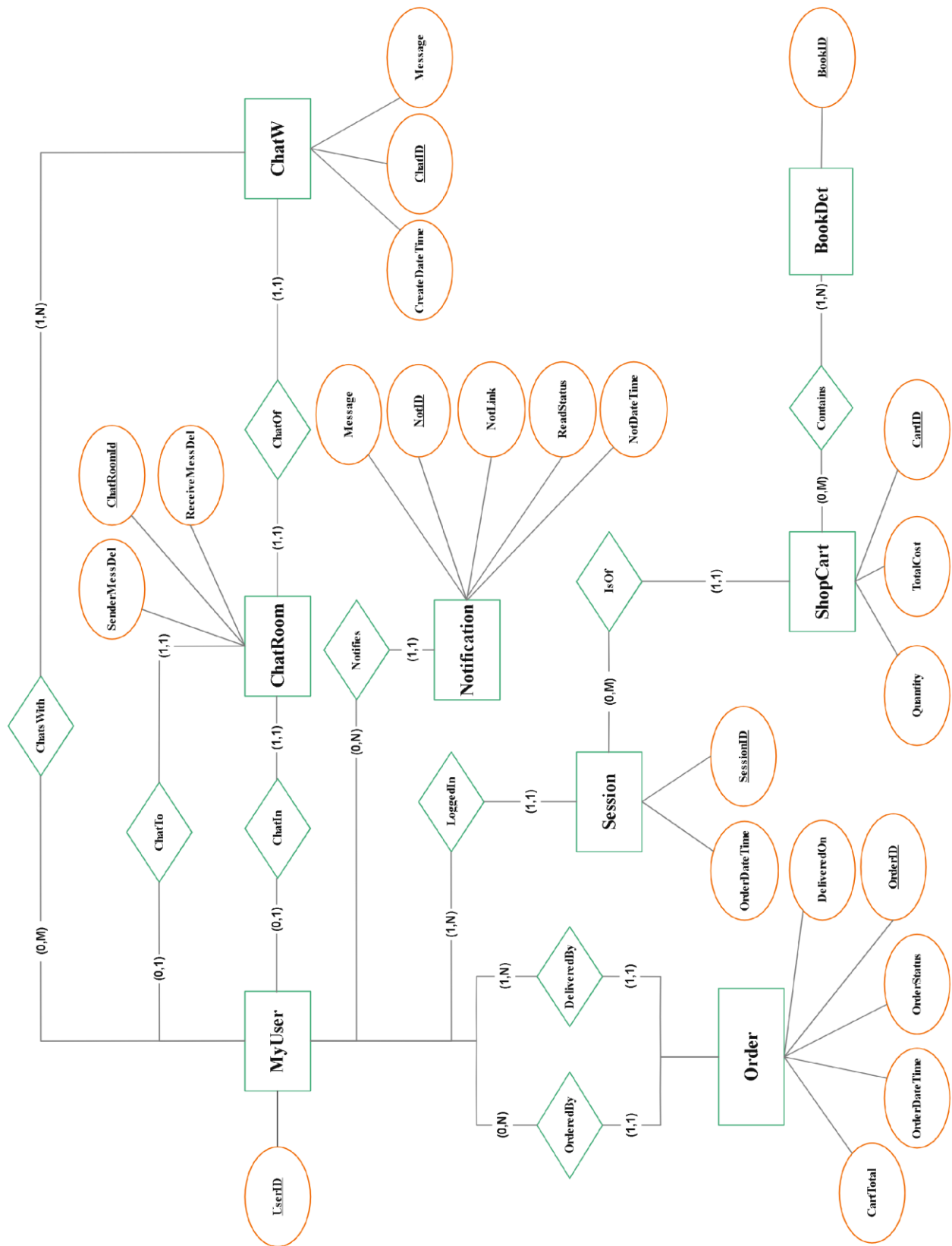


Fig. 4. ER Diagram for Order Details and Notification Models

2) Back End Technologies

a) Python

Python is an interpreted, high-level, general-purpose programming language. Created by Guido van Rossum and first released in 1991 (Kuhlman, 2011), Python's design philosophy emphasizes code readability with its notable use of significant whitespace. Its language constructs and object-oriented approach aims to help programmers write clear, logical code for small and large-scale projects.

In this website, python is used as backend language to code database part and all functionalities that the website can perform. The version of Python used in this development is Python 3.6.

b) Django

Django (Holovaty & Kaplan-Moss, 2008) is a high-level Python Web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of Web development, so we can focus on writing our app without needing to reinvent the wheel. It's free and open source. Django's primary goal is to ease the creation of complex, database-driven websites. The framework emphasizes reusability and "pluggability" of components, less code; low coupling, rapid development, and the principle of don't repeat yourself. Python is used throughout, even for settings files and data models. Django also provides an optional administrative create, read, update and delete interface that is generated dynamically through introspection and configured via admin models. The version of Django used during development is Django 2.1.5.

c) SQLite

SQLite is a C-language library that implements a small, fast, self-contained, high-reliability, full-featured, SQL database engine. SQLite is the most used database engine in the world. By default, Django used SQLite3 as its default database. Django provides a specific way to define our database using the python programming language.

d) Jinja2

Jinja2 (Lokhande et al, 2015) is one of the most used template engines for Python. It is inspired by Django's templating system but extends it with an expressive language that gives template authors a more powerful set of tools.

It adds Sandboxed execution mode i.e. every aspect of

the template execution is monitored and explicitly whitelisted or blacklisted, whatever is preferred. Some features of Jinja2 are enlisted below:

- Powerful automatic HTML escaping system for cross-site scripting prevention.
- Template inheritance makes it possible to use the same or a similar layout for all templates.
- Optional ahead-of-time compilation and Configurable syntax i.e. we can reconfigure Jinja2 to better fit output formats such as LaTeX or JavaScript.

The version of Jinja2 used as a templating language in this project is 2.10.

B. Hardware and Software Requirements

The project developed satisfies all the functional and non-functional requirements enlisted. The following specifications are required for the project to run on any device.

1) System Specifications

Processor: Intel(R) Core (TM) i5-5005U CPU @ 2.00GHz

RAM: 2 GB

System Type: 32-bit/64-bit operating system, x32 or x64 based processor

Operating System: Windows 7/8/10.

2) Software Interface

Front End: HTML, CSS, Bootstrap, JQuery

Backend: Django

Local Access Link: localhost:8000

Global Access Link: <https://www.shreic.com/home/>

C. Methodology

The Project is developed via multiple steps. The major steps are enlisted here:

1. Installing Python and adding it to the windows path.
2. Creation of Virtual Environment (Following commands are written in Command Prompt)
 - pip install virtualenvwrapper-win
 - mkvirtualenvenvironmentname (any name can be given)
 - workonenvironmentname
3. Installing Django
 - pip install Django
4. Go to Destination Place where you want the project to be kept, using cd command.
5. Create Project as follows
 - django-admin startprojectsomeprojectname
 - cd someprojectname
6. Create App of the project as
 - django-admin startappappname
 - python manage.py makemigrations
 - python manage.py migrate
7. Copy the Template Folder (if Front End Template is

downloaded) to the project folder created.

8. Run Server (localhost:8000)

- `python manage.py runserver`

The Flow Chart (Figure 5) illustrates the steps that are required in order to install the prerequisites of the project and then the steps involved in the development of project. Commands written in bracket are to be run on Command

Prompt. They are the steps that are required to install virtual environment and to run local server on the system project is to be developed on. The code of backend and frontend can be coded in any code editor (Sublime Text was used in this project).

All the changes that were made in the project can be seen on the local server. The Data was stored on online cloud service Cloudinary.

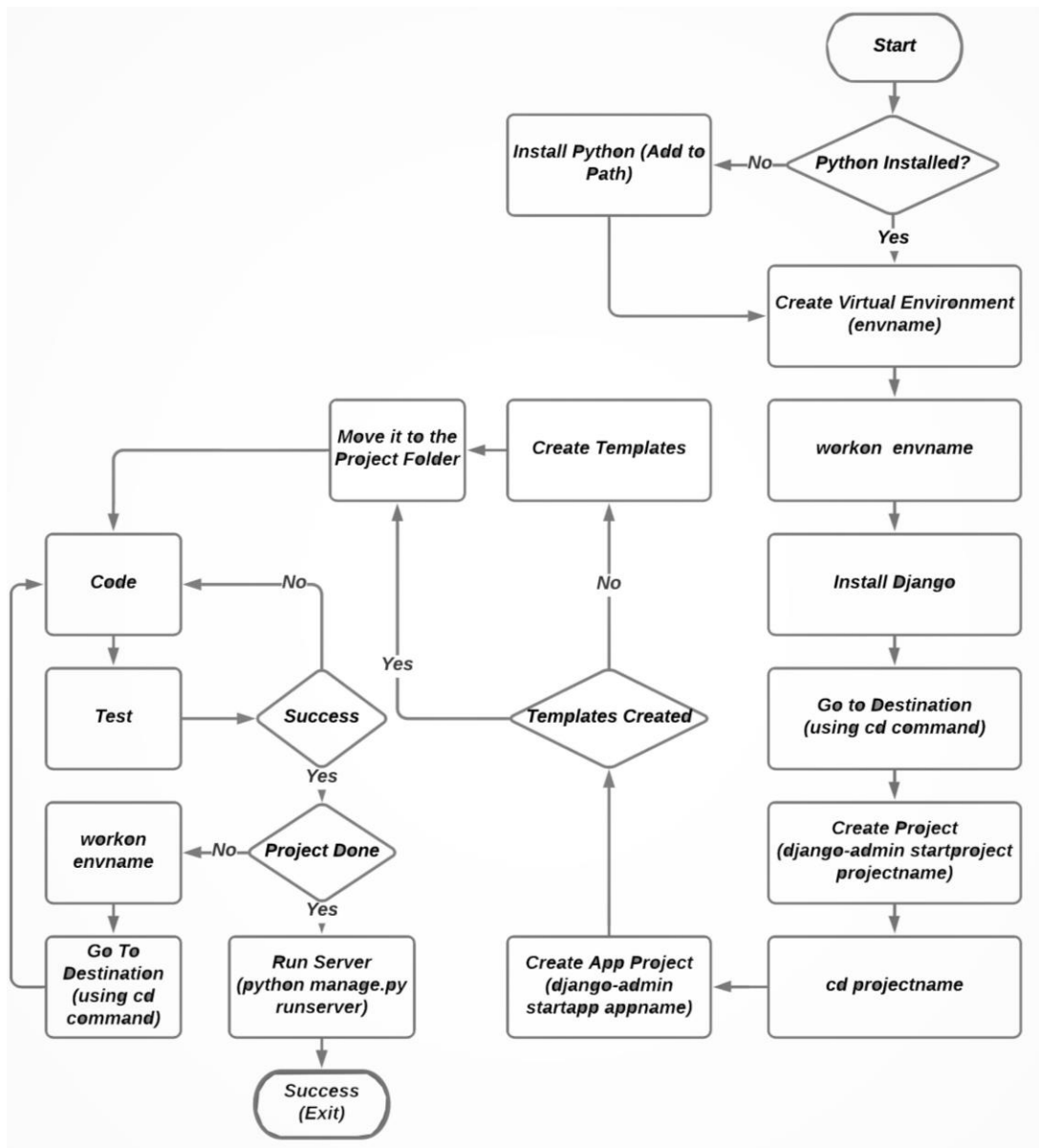


Fig. 5. Flow Chart of the Methodology

V. RESULT

Few snaps of the website are shown here showing the main functionalities of the project.

The Home Page (Figure 6) has the top bar consisting of the website logo, the drop-down menu of categories, sell books

button, add notes button, request an educational resource button, a dynamic search bar and the signup/login option. Below the top bar is the banner representing the main motive of the website. After that there is stats of the total number of books, notes and users of the website. On the bottom right corner is the recent chat button which shows the recent chats the user had done.

Figure 7 shows the dynamic search bar developed for simplifying the searching option for users of the website. A dynamic search bar possesses the feature of typing suggestions and helps the user to autocomplete the searching process. A form is created in HTML code of the home page and a function is coded in the views.py page that retrieves all books and notes uploaded, filters it and return the names on the basis of letters typed in the search box and an URL is written in urls.py to refer to the function created.

An Add a book form (Figure 8) is created for adding a book to sell or donate through the website. On clicking the Sell Button on the top bar following form will open. The form consists of Book Name, Dropdown menu for Category and Subject, Author Name, Publication Name and Year and MRP, Description and the Cover Image of the book. On Submit, the record will be added to BookDetails Model created in models.py. And Figure 9 shows the form created for uploading notes (question papers/class notes/syllabus/ebooks) on the website. On clicking the Add Button on the top bar following form will open. The form consists of Notes Name, Dropdown menu for Notes Type, Section, Category and Subject, Publication Name and Year, Description and the file (pdf format). On Submit, the record will be added to NotesDetails Model created in models.py.

The dropdown menu (Figure 10) that appears on the screen when the mouse is hovered over the Category Button shows all the Categories created and their subcategories. The dropdown menu is divided into categories and each category consists of different subcategories. These Categories and their corresponding Subcategories are retrieved with the help of functions written in views.py and called in the HTML page.

The Computer Science subcategory (Figure 11) of Science under Academics Section (Academics/Science/Computer Science) is a sample for a Particular Subcategory Page. The side bar consists of filters based on the type of notes. Function selectedproducts in views.py filters the notes and books of the selected category with the help of ID of the subcategory created using get () and filter (). Figure 12 illustrates the chat-box created for chatting between the User and the Seller of a Book. By the help of chat and mutual understanding the transaction can be done successfully between the two. The sidebar shows the persons the user recently chatted to and the right side shows chat of the particular person the user is chatting to. The Chat Box can be opened by clicking the Recent Chats button shown in the Figure 3.

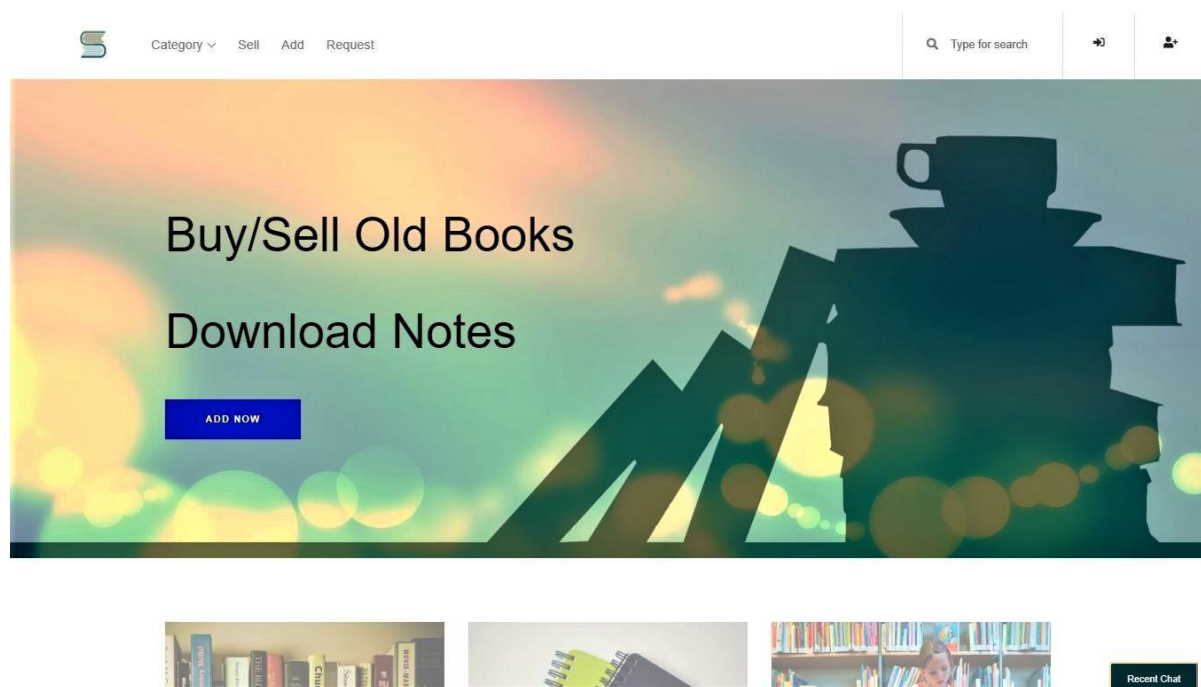


Fig. 6. Home Page

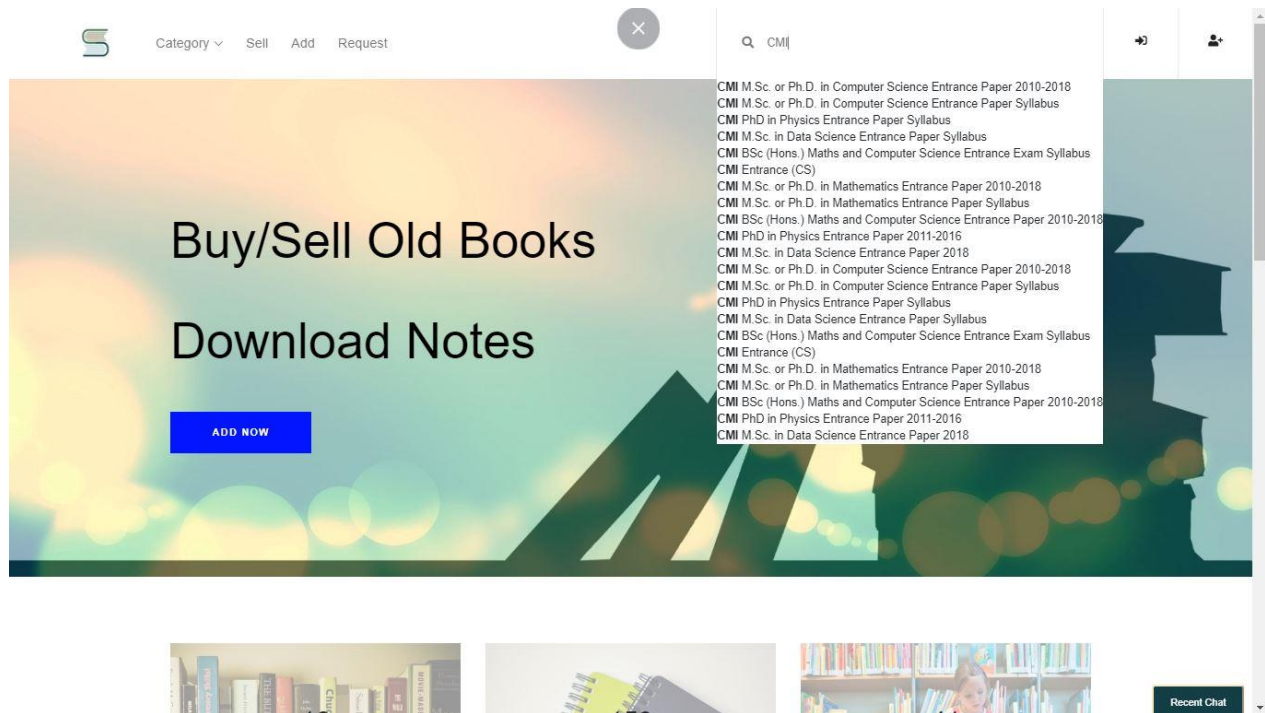
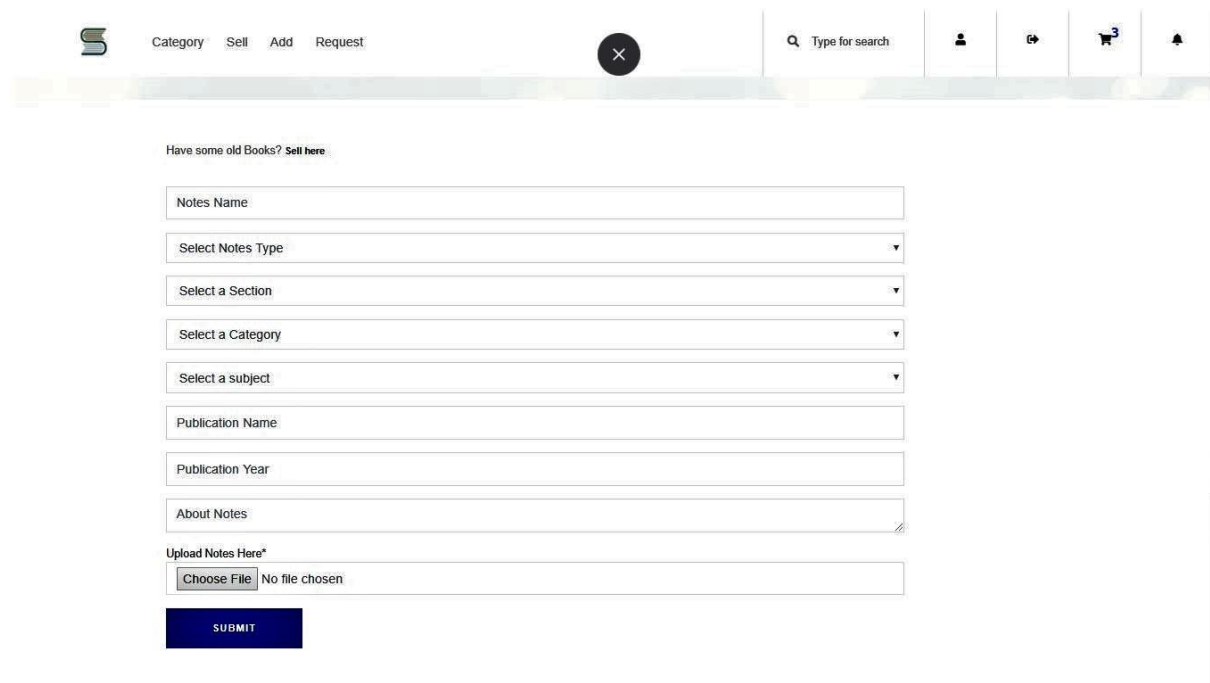


Fig. 7. Dynamic Search Box

The screenshot shows a website with a header containing a logo, navigation links (Category, Sell, Add, Request), and a search bar. The main form is titled "To Sell A Book" and contains the following fields: "Book Name", "Select a Section", "Select a Category", "Select a subject", "Author Name", "Publication Name", "Publication Year", "Your Selling Price", "About Books", and "Upload Book Cover Photo *". The "Upload Book Cover Photo *" field has a "Choose File" button and "No file chosen" text. A blue "SUBMIT" button is at the bottom of the form. The footer is a solid black bar.

Fig. 8. To Sell A Book



Have some old Books? [Sell here](#)

Notes Name

Select Notes Type

Select a Section

Select a Category

Select a subject

Publication Name

Publication Year

About Notes

Upload Notes Here*

Choose File No file chosen

SUBMIT

Fig. 9. To Upload Notes

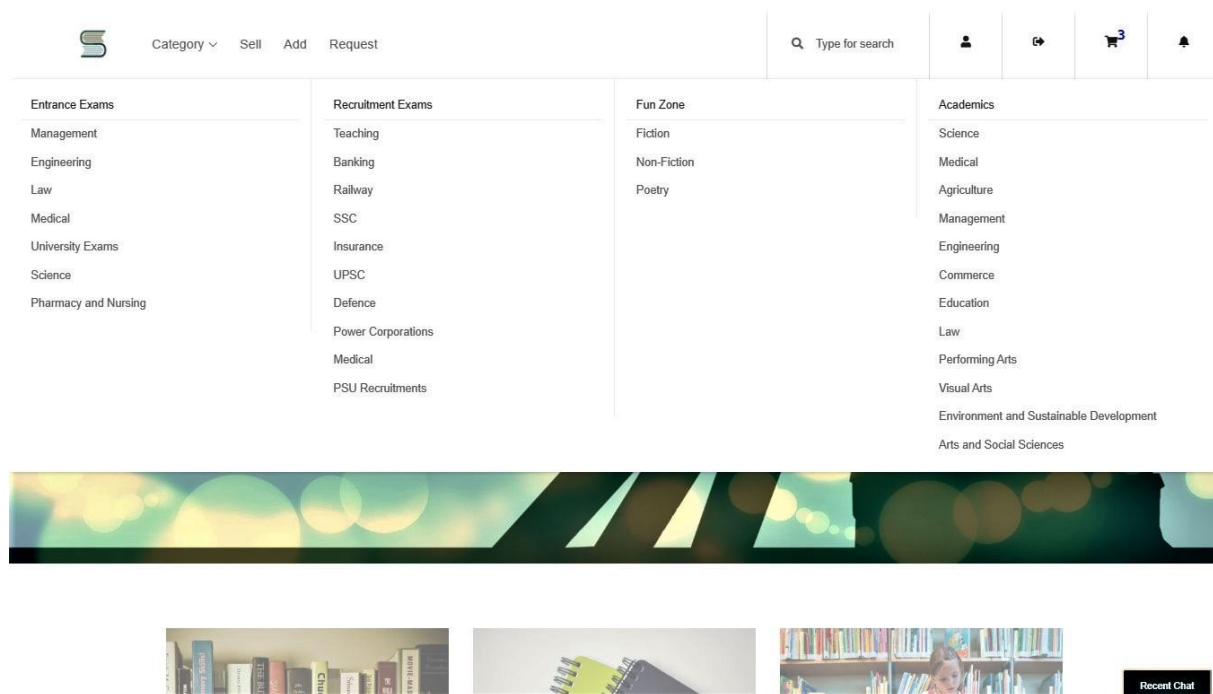


Fig. 10. Different Categories and Sub-Categories

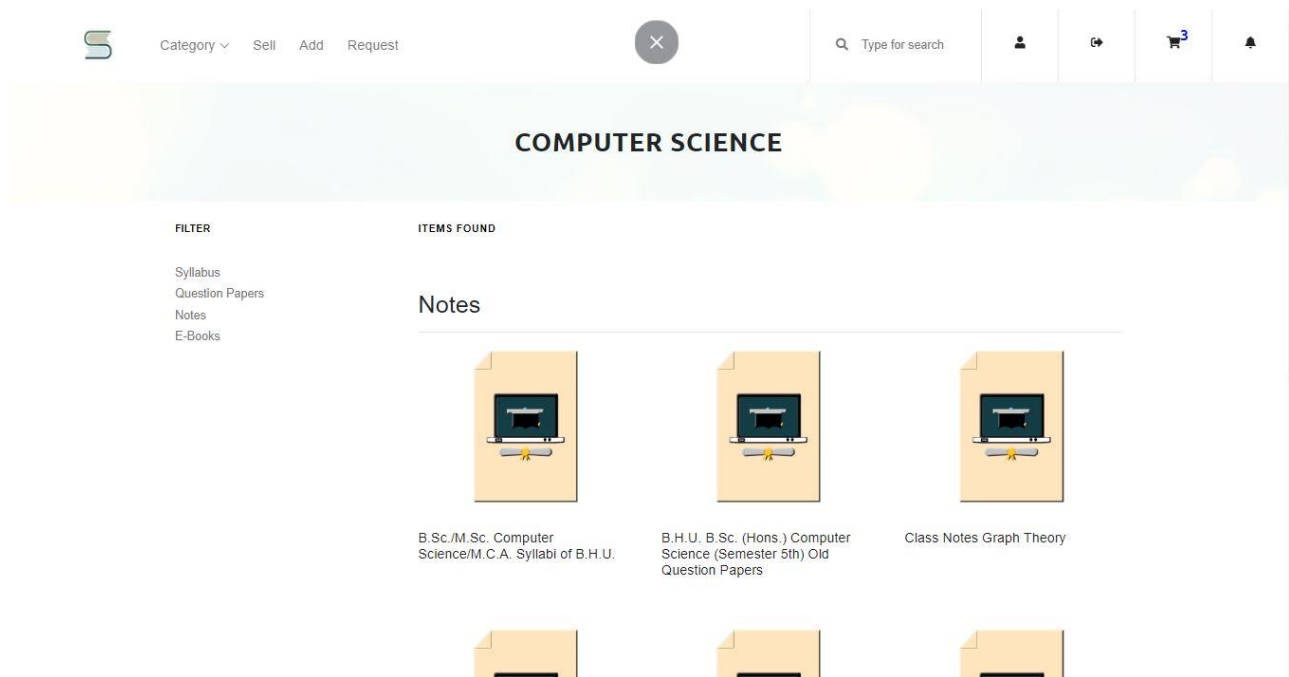


Fig. 11. Particular Page of a Category (Academics/Science/Computer Science)



Fig. 12. Chat Room

VI. TESTING

In this project the testing (Thakur, 2017) has been done as follows:

A. Black Box Testing

Black Box Testing is a software testing method in which the internal structure/ design/ implementation of the item being tested is not known to the tester.

In this testing process, ten Users were selected. Firstly, they

were said to Sign Up on the website. After successful registration of each user, they were asked to upload notes (the samples were given) in pdf format in which 8 of them succeeded. Two of the Users were unable to upload the notes. This issue was resolved and then further process took place. Following the note uploading testing, each User was provided an old Book, which they uploaded on the server to test the functionality of the book adding feature.

Further, the users were asked to fill the “request a note” form where the user can request a book or note to be made available

on the website. After that, the users were said to chat with each other to check the simplicity and functionality of the chat-box which was found satisfiable.

At last, in the process of Black Box Testing, Users were requested to review the whole website and each functionality and their feedback was recorded. According to which, corrections and better implementations were made on the website.

B. White Box Testing

White Box Testing is a software testing method in which the internal structure/ design/ implementation of the item being tested is known to the tester.

The White Box Testing is done by the developers only. There are many kinds of White Box Testing. In this project, Unit Testing was done i.e. each unit of code was separately tested and was integrated lastly. The testing of source code involved

- Internal security holes
- Broken or poorly structured paths in the coding processes
- The flow of specific inputs through the code
- Expected output
- The functionality of conditional loops
- Testing of each statement, object, and function on an individual basis.

Different test cases were made for each unit of source code and were tested. For each test case, the desired output was expected. When the desired output was not encountered it led to the bug. Each error was removed from the source code and all units were integrated at last. The benefit of unit testing was that errors/bugs were identified at the initial level of development thus avoiding any big error to occur at a further level of development.

Also, in the testing process, the security issues have been resolved. The CSRF middleware and template tag in Django provides easy-to-use protection against Cross-Site Request Forgeries. This type of attack occurs when a malicious website contains a link, a form button or some JavaScript that is intended to perform some action on your website, using the credentials of a logged-in user who visits the malicious site in their browser. A related type of attack, 'login CSRF', where an attacking site tricks a user's browser into logging into a site with someone else's credentials, is also covered. The CSRF middleware is activated by default in the middleware setting in the Django framework. Ajax Forms are used in the project and each form is tagged with the token to secure the forms from unwanted inputs that can corrupt the database. The web application is made to exchange data with the webserver securely by deploying the web application behind HTTPS.

CONCLUSION

After analyzing the results obtained, the project developed can be considered satisfiable. It can be concluded that the website will be very helpful to students in their educational life as it provides all educational resources required in a college or school life. As the project works as an Educational cum E-Commerce Website and thus students can donate or sell their old books too.

To conclude, the project is developed using the proper Software Engineering process, following the Iterative Model of SDLC. A Project Control List was created after doing the feasibility study for functionalities as well as non-functional requirements. Then proper schema and tables that were supposed to be required in the development process were made and relationships between each table were drawn. For this ER Diagram was made which has been illustrated in the paper. Also, the flow chart was created so that each process can be done sequentially. After that, each task from the project control list was coded, tested using White Box Testing and implemented separately as per the Iterative Model. At last every unit was integrated and users were selected for Black Box Testing. Each user was asked to run the project and test each functionality of the project. After the testing, feedback and suggestions were recorded and accordingly the amendments were made. Security issues were resolved with the help of CSRF tags given by the Django Framework and by deploying the Web Application behind HTTPS.

The approach used in the System Development Model can act as a roadmap for the development of similar kinds of Web Applications efficiently.

Also, for future works few more features can be added to the project. Some of them that have been enlisted are using the platform as an online assignment submission platform, creation of a chatroom consisting of teachers and students of particular university/college and adding digital payment methods for easier transactions.

REFERENCES

- Buy Second Hand Books, Old Books, Used Books Online in India. (n.d.). Retrieved from <https://www.usedbooksfactory.com/>.
- Viorica-Torii, C., & Carmen, A. (2013). The Impact of Educational Technology on the Learning Styles of Students. *Procedia-Social and Behavioral Sciences*, 851-855.
- About Company. (n.d.). Retrieved from <http://www.tutorialspoint.com/>.
- The Physics Classroom. (n.d.). Retrieved from <https://www.physicsclassroom.com/>.
- Kelly, L., & Breault, K. (2006). Developing Educational Websites: Investigating Internet Use by Students and

- Teachers. In *Proceedings of Thinking, Evaluating, Rethinking, ICOM-CECA Conference*, Rome.
- AglaSem Admission. (n.d.). Retrieved from <https://admission.aglasem.com/>.
- Sachan, N. (2019, February 20). Welcome to BHU Student Club, BHU Student Club. Retrieved from <http://bhustudentclub.in/>.
- Jalote, P. (2003). *An Integrated Approach towards Software Engineering*. Narosa Publishing House.
- Musciano, C., & Kennedy, B. (1996). *HTML, The Definitive Guide*. O'Reilly & Associates.
- Powell, T. A. (2010). *HTML & CSS: The Complete Reference*. The McGraw-Hill Companies.
- Flanagan, D. (2006). *JavaScript: The Definitive Guide*. O'Reilly Media, Inc.
- Shenoy, A., & Sossou, U. (2014). *Learning Bootstrap*. Packt Publishing Ltd.
- Kuhlman, D. (2011). *A Python Book: Beginning Python, Advanced Python, and Python Exercises*. Platypus Global Media.
- Holovaty, A., & Kaplan-Moss, J. (2008). *The Definitive Guide to Django: Web Development done right*. Apress.
- Lokhande, P. S., Aslam, F., Hawa, N., Munir, J., & Gulamgaus, M. (2015). Efficient way of Web Development using Python and Flask. *International Journal of Advanced Research in Computer Science*, 54-57.
- Thakur, M. S. (2017). Review on Structural Software Testing Coverage Approaches. *International Journal of Advance Research, Ideas and Innovations in Technology*, 281-286.
