

**CONTENT MANAGEMENT SYSTEM WITH  
DJANGO**

**SHIB SHANKAR SUTRADHAR**

**BACHELOR OF SCIENCE  
INFORMATION AND COMMUNICATION  
ENGINEERING  
NOAKHALI SCIENCE AND TECHNOLOGY  
UNIVERSITY**

# **CONTENT MANAGEMENT SYSTEM WITH DJANGO**

**SHIB SHANKAR SUTRADHAR**

This project proposal submitted to the Department of Information and Communication Engineering, Noakhali Science and Technology University, in partial fulfillment of the requirements for the degree of B.Sc. (Eng.) in Information and Communication Engineering.

**DEPARTMENT OF INFORMATION AND  
COMMUNICATION ENGINEERING NOAKHLAI SCIENCE  
AND TECHNOLOGY UNIVERSITY**

**SEPTEMBER 2021**

## **DECLARATION**

I submit this project proposal to the Department of Information and Communications Engineering, Noakhali Science And Technology University, Noakhali, Sonapur. So, I, hereby, declare that this project report has not been submitted elsewhere for the requirement of any kind of degree, diploma or publication.

---

Shib Shankar Sutradhar

Roll:ASH1711055M

Session:2016-17

Dept of ICE,NSTU

## **ACCEPTANCE**

This project report is submitted to the Department of Information & Communication Engineering, Noakhali Science & Technology University, Sonapur, Noakhali in partial fulfillment of the requirements for having the B.Sc. degree in ICE.

---

Prof. Dr. Md. Ashikur Rahman Khan

Professor & Chairman

Department of Information and Communication Engineering

Noakhali Science & Technology University

Sonapur, Noakhali.

## **ABSTRACT**

This paper is describe about the process of building a content management system. With CMS any approved user can quickly and easily publish webpages without complicated software or programming. And they find an integrated workflow process for reviewing and approving content. Because of the CMS enables non-technical users to easily publish content, this frees up technical developers to focus on functionality and enhanced features. User only need to change data once for it to be updated throughout your site. User can access and update your site from anywhere with an Internet connection. And the Security of this system is automatic.

## TABLE OF CONTENT

| Page Title                                | Page no. |
|-------------------------------------------|----------|
| Cover page                                | 1        |
| Title page                                | 2        |
| <u>Declaration</u>                        | 3        |
| Acceptence                                | 4        |
| Abstract                                  | 5        |
| Table of Content                          | 6        |
| List of figure                            | 7        |
|                                           |          |
| 1.Introduction                            | 8        |
| 1.1.Introduction of area of research      | 8        |
| 1.2. Literature Review                    | 9        |
| 1.3 . Statement of problem                | 15       |
| 1.4. Project objectives                   | 17       |
|                                           |          |
| 2. Methodology                            | 18       |
| 2.1. Research design                      | 18       |
| 2.1.1. E-R Diagram                        | 23       |
| 2.1.2. Database Diagram                   | 24       |
| 2.1.3. Procedure                          | 25       |
| 2.1.4. Use Case Diagram                   | 26       |
| 2.2 Instrumentation                       | 27       |
| 2.3 Data collection and analysis procedue | 29       |
|                                           |          |
| 3. Expected outcome                       | 30       |
| 4. Conclusion                             | 30       |
| 4.1. Conclution                           | 30       |
| 4.2 Future Work                           | 30       |
| 5. References                             | 31       |
| Project Timetable                         | 32       |

## LIST OF FIGURES

| Figure name      | Page no |
|------------------|---------|
| E-R Diagram      | 24      |
| Database Diagram | 25      |
| Use Case Diagram | 27      |

## **CHAPTER 1**

### **INTRODUCTION**

#### **1.1 . Introduction of the area of research :**

In this world time is most valuable things,as well as a website is the common material for any types of company.In present it's very hard to find a small or big company without website.Not only company now a day every business build their own website to perform their work and increase their business introduction.

So we can say that to build a website is the most valuable work on this real world.For develop a Website a business company need to hire a person or system which are available to make a good website,which are useable to this business.

Build a website is not an easy task,it's a costly process, with cost it's also provide the headache of time.At first we say that time is the most valuable thing in our real life.But in real to made a website we need to know about web technology,front-end technology which are HTML/XML,CSS,JS, and back-end technology like as python,PHP,JS and many other technology which are used for build a website.For consume this type of technology and also to build a website with this type of technology is very time dependent and the cost rate to build website with this is very high.

To solve this problem of time and cost we can use my web-app,which is a content management system based on Django.By using this web app a non-technical person could build his/her website easily.It will give some flexible feature/plugin which are easily understandable.By using

this flexible plugin a user could made her/his website easily without any knowledge of upper technology.

My web application decrease the rate of cost because there is no need to hire a web specialist to build a website with high cost.It is time saver also because my system give us flexibility to make a good website .

## 1.2 . Literature Review :

### I . Exponent CMS:



Exponent CMS is a free, open-source, open standards modular enterprise software framework and content management system (CMS) written in the programming language PHP.

Exponent CMS editing system allows website content to be edited on the page as it appears - without back-end administration. The default installation includes a set of modules for managing a typical website. Additional modules are developed by a community of open-source developers and can be installed via a web browser.

### II . mezzanine:



Mezzanine is a Content Management System written in Python using the Django framework. It was initially developed by Stephen McDonald in 2010, then formally released for use in 2012. McDonald wrote in a blog post that reception to Mezzanine was mostly positive, with the most notable feedback coming from GitHub users.

Wagtail is a free and open source content management system (CMS) written in Python. It is popular amongst websites using the Django framework. The project is maintained by a team of open-source contributors backed by companies around the world. The project has a focus on developer friendliness as well as ease of use of its administration interface, translated in multiple languages.

Kotti is a high-level, Pythonic web application framework based on Pyramid and SQLAlchemy. It includes an extensible Content Management System called the Kotti CMS.

Plone is a free and open source content management system (CMS) built on top of the Zope application server. Plone is positioned as an enterprise CMS and is commonly used for intranets and as part of the web presence of large organizations.

MODX (originally MODx) is an open source content management system and web application framework for publishing content on the world wide web and intranets. MODX is licensed under the GPL, is written in the PHP programming language, and supports MySQL, MariaDB and Percona Server as the database. It was awarded Packt Publishing's *Most Promising Open Source Content Management System* in 2007.

October is a self-hosted content management system (CMS) based on the PHP programming language and Laravel web application framework. It supports MySQL, SQLite and PostgreSQL for the database backend and uses a flat file database for the front end structure.

Grav is a free software, self-hosted content management system (CMS) written in the PHP programming language and based on the Symfony web application framework. It uses a flat file database for both backend and frontend.

ImpressCMS is an open source content management system for building and maintaining dynamic web sites, written in the PHP programming language and using a MySQL database. The product is released under the GNU General Public License version 2.

SPIP (*Système de Publication pour l'Internet*) is a free software content management system designed for web site publishing, oriented towards online collaborative editing.

#### 2.4 COMPARISON BETWEEN THE EXISTING SYSTEM AND OUR SYSTEM:

| SI. No | Paper Title  | Methodology                                                                                                                                                            | Advantages                                                        | Limitations                                                                                                                            | Difference to us                |
|--------|--------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------|---------------------------------|
| 1      | Exponent CMS | Exponent CMS is a free, open-source, open standards modular enterprise software framework and content management system (CMS) written in the programming language PHP. | is built-in for better functionality overall part of the website, | Before you can run <b>Exponent CMS</b> , there are four components you must have installed on your computer or web hosting environment | I use a better interface system |

|   |           |                                                                                                                                                |                                                                              |                                                                                                                  |                                              |
|---|-----------|------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------|----------------------------------------------|
| 2 | Mezzanine | Mezzanine is a Content Management System written in Python using the Django framework. It was initially developed by Stephen McDonald in 2010, | Mezzanines are efficient, quick to install and profitable                    | its takes significant ground floor space, supported by posts                                                     | I will use drag nad drop system in my system |
| 3 | Wagtail   | Wagtail is a free and open source content management system (CMS) written in Python. It is popular amongst websites using the Django framework | a lot of cool methods that you can take advantage o f to have already talked | Wagtail is a tool in the Self-Hosted Blogging / CMS category of a tech stack                                     | I made my system flexible                    |
| 4 | Kotti     | Kotti is a high-level, Pythonic web application framework based on Pyramid and SQLAlchemy                                                      | We will take advantage of the imports already there.                         | Its user interface are not good                                                                                  | I will made a understandable system          |
| 5 | Plone     | Plone is a free and open source content management system (CMS) built on top of the Zope application server.                                   | Easy to use content management.                                              | The disadvantages of Plone include sluggish performance when logged on although this may depend upon the browser | I will use djnago admin login system         |
| 6 | MODX      | MODX (originally MODx) is an open source content management                                                                                    | The advantages of MODX                                                       | the administrative panel <b>MODx</b> dif                                                                         | We knaow the flexibility of Django admin     |

|    |            |                                                                                                                                      |                                                                                                                   |                                                             |                                                         |
|----|------------|--------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------|---------------------------------------------------------|
|    |            | system and web application framework for publishing content on the world wide web and intranets                                      | are most noticeable in the work with large sites.                                                                 | fers by slow work                                           |                                                         |
| 7  | October    | October is a self-hosted content management system (CMS) based on the PHP programming language and Laravel web application framework | The other big advantage is the fact that October CMS has a modern codebase and comprehensive online documentation | Its build with PHP which security is not strong than djnago | Django security system is strong                        |
| 8  | Grav       | Grav is a free software, self-hosted content management system (CMS) written in the PHP programming language                         | get more flexibility                                                                                              | Site load speed                                             | Python is fast language so site load speed will be high |
| 9  | ImpressCMS | ImpressCMS is an open source content management system for building and maintaining dynamic web sites                                | advantage of a CMS is that is offers a number of ready to use                                                     | Maintenance                                                 | Maintenance are easy                                    |
| 10 | SPIP       | SPIP is a free software content management system designed for web site publishing, oriented                                         | It consists of a collection of files installed on a web server which allow you                                    | Site load speed                                             | Load speed will be high                                 |

|  |  |                                             |                                                                   |  |  |
|--|--|---------------------------------------------|-------------------------------------------------------------------|--|--|
|  |  | towards<br>online collaborative<br>editing. | to<br>take <b>advantage</b><br>of a number of<br>automated tasks: |  |  |
|--|--|---------------------------------------------|-------------------------------------------------------------------|--|--|

### The advantages of my CMS :

- It is easier to adapt the content management workflow with Django.
- CMS has a lower chance to get hacked
- CMS does not become slow after some time
- It is easier and faster to create extensions (plugins) with frontend edition
- No extension required to support multiple languages
- Can be integrated with any Django applications
- It is a good compromise between developer and user
- It uses one of the most popular programming languages in the world

### 1.3 . Statement of Problem :

From previous discussion about my web-application we generally can say that our system is an important feature of this web world. Some important need of my system or a cms are given below.

- **Ease of use.** Due to a graphical user interface, even those with limited technical knowledge can use the software.
- **Easy to manage content.** Not only is creating content easy, but so is removing content. A CMS makes it easy to unpublish content to keep websites up to date.
- **Accessible from anywhere .** A CMS can be cloud-based or on premises, but users can access content from anywhere with a device that's connected to the internet.
- **Allows multiple users.** A CMS makes it easy to manage publishing permissions.
- **Instant content updates.** A CMS enables users to manage and update content in real time -- without needing to wait for a developer.

In this world of internet CMS is a very common web-application which made easy way to join the world of internet. In this internet world, there have many CMS web-app, by using this we are able to make many types of website easily. In my literature review we discuss about two different types of CMS one build with python and another build with php. From previous discussion we are able to say that to make a CMS is not an unrealistic project. So we can say that making my CMS is not an unrealistic task. The probability rate for building my project is not bad.

Many developers or some company like as wordpress, Django cms made their CMS for increase flexibility in the field of website building. So building a CMS is not a new idea.

I think that as well as other CMS my system will provide a user friendly interface which increase the flexibility to build a website for that reason people or user of application , I hope that they will be able to find a good experience with my system.

I wish In backend or my source code of this project give an example of good CMS.By following this source code some types of people who want to make their own CMS or who want to update or work with my CMS they could be find a proper place for experimenting their knowledge.

I will try my best for finding a optimal result.If I find a result which is not more different from my plan then it will give some exclusive feature which could be give some excitement for my web-app user.

At first we said that our application provide some feature or plugin.By using this plugin my app user are able to make website and this is the mai summary of my system.For building this type of system I need to research about python,python libraries,python api,web technology,front end technology ,backend technology ,internet technology,different types of programming language and some other things which will be usable for my upcoming project.

From previous discussion we can understand that what is a CMS is or what is the basic work of it.On this upper system we noticed that the flexibility for user is not that good.i want to work with the flexibility which made a easy CMS for my user.Also I use Django framework,we also know that about the security system of this framework so we can say that my CMS security system will be good.I will try made a good user interface which could be understandable by any type of user.

## 1.4 . PROJECT OBJECTIVES :

The main objectives of this project is to analyze various CMS are given below:

- To build a secure login system which will made with Django admin panel login system.
- To build a good user interface with good flexibility.
- To build some usable plugin for work with different type of need of users.
- To build a flexible backend system with mysql,Django,python for made a flexible system

## CHAPTER 2

### METHODOLOGY

#### 2.1 . Research Design:

Basically to design my project which is a CMS based on Django we need to work with two type of module:

- I. Plugin module
- II. Main\_module

##### Plugin\_module:

In this type of module we set some type attribute which describe about my plugin characteristic. And that module attribute describes about the user interface and my backend interface. Plugin module attribute will be filed by the user. The input value which are given to the plugin module is changeable by the admin user. After publishing his/her website on server it should be possible to change the input value of plugin. So we can say that we are able to made a dynamic website by using my CMS. There have also a common attribute for all plugin which is page\_no and plugin\_no. By using this two attribute we will able to find our desired plugin. User will be able to set the plugin position with drag and drop feature. On this time we will describe about four basic plug\_in which are usable by the user. Which are,

- Navbar plugin.
- Footer plugin.
- Blog plugin
- Image\_upload plugin

##### Navbar\_plugin:

Whith this plugin we could made our navigationbar easily. The attribute for this navbar plugin are given bellow.

- SI\_No

- Page\_no
- Plugin\_no
- Nav\_title
- Nav\_color
- Nav\_H\_W

#### Footer plugin:

With this plugin we could made our website Footer easily.The attribute for this footer plugin are given below.

- Si\_no
- Page\_no
- Plugin\_no
- Footer\_H\_W
- Footer\_color
- Footer element

#### Blog plugin:

With this plugin we could made our website blog easily.The attribute for this Blog plugin are given below.

- Si\_no
- Plugin\_no
- Blog\_title
- Blog\_H\_W
- Text\_color
- Page\_no

#### Image\_uplad plugin:

With this plugin we could made our website Image\_uploader easily.The attribute for this Image\_upload plugin are given below.

- Si\_no

- Plugin\_no
- Blog\_tile
- Blog\_H\_W
- Text\_color
- Page\_no

By using this upper plugin and with the proper use of this plugin attribute we could made our website.

### **Main\_module:**

It's the main module for my system. Without this module my system is nothing. Main module connected with all attribute of all plugin in my system. It's made many to one relationship with my system plug\_in module. For each page it collect different data which are provided by the plugin. The main work for this plugin is to collect data from plugin and deliver required data for building website. The website which are built by the user that's are able to collect it's required data from the main module. We know that we could build dynamic website with my system. For that edit the value of main\_module is required. There should be a feature by using this we are able to remove or update the value of our required attribute. Some attribute of my main module are given below,

- Si\_no
- Page\_no
- Pplugin\_no
- Nav\_title
- Nav\_color
- Nav\_H\_W
- Blog\_title
- Blog\_H\_W
- Text\_color
- Footer\_H\_W
- Footer\_color
- Footer\_element
- Img\_name

- `Img_upload_to`
- `Image_H_W`

From upper attribute we see that `si_No`, `page_no`, `plug_in` are the common attribute like as `plugin`.

By using this attribute we could scrap data from `plugin` and able to use this data as requirement.

### 2.1.1E-R Diagram:

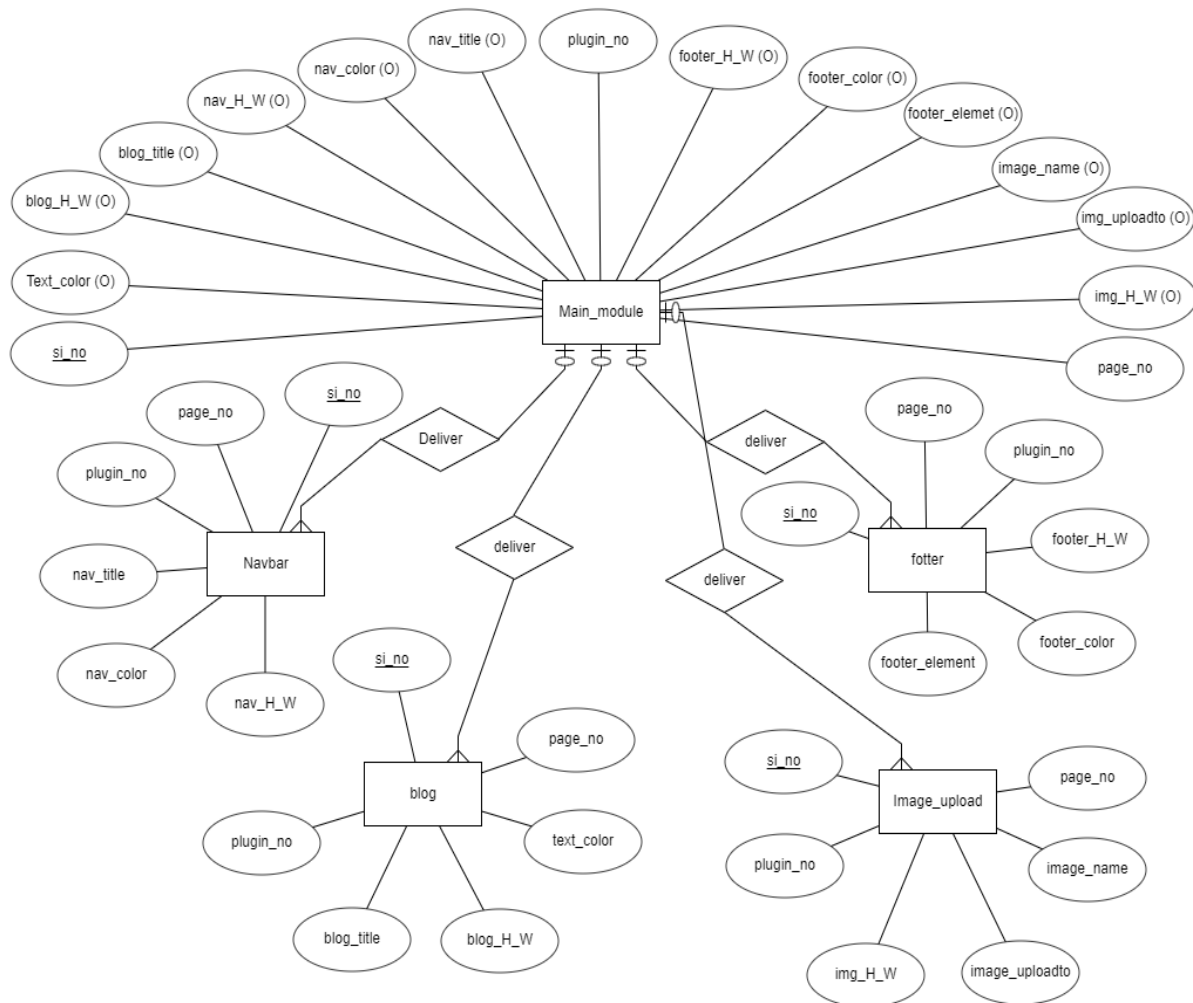
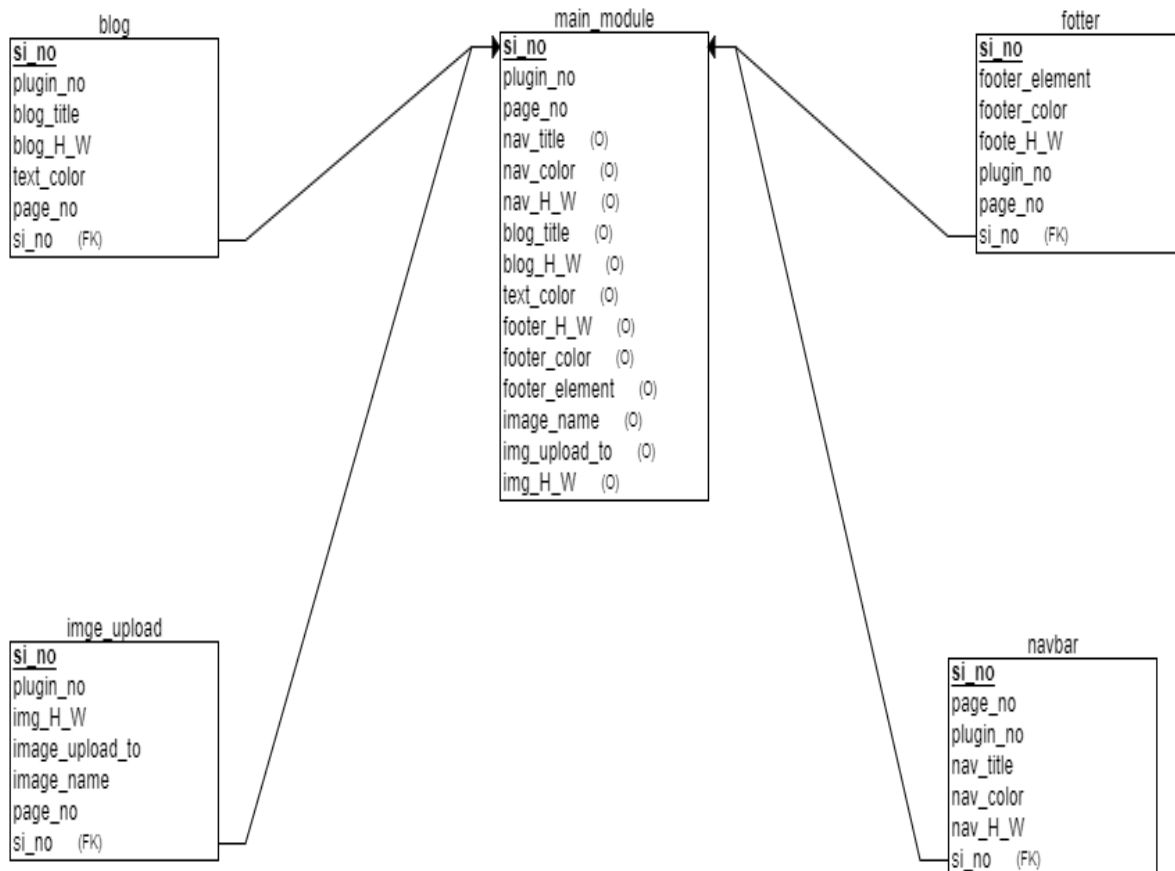


Fig: E-R Diagram for CMS

Every plugin module of this diagram make a many to one relationship module with the main module. For main module every plugin module are selectable, So main module made a relation which is one to many and take a optional option for selecting different type of plugin module.

The work of this diagram is to delivered data from plugin module to main module. So the name of the connection is deliver.

### 2.1.2 Database Diagram:



On this diagram we set a primary key or a foreign key for each table. By this type of key the table attribute data are communicate among them. Every table schema have some attribute, Every plugin foreign key attribute are same as main\_modules primary key attribute. By this they can build a relationship among them. Without page\_no, plugin\_no and si\_no all attribute are optional its defined that the value of this attribute could be null.

### 2.1.3 Procedure:

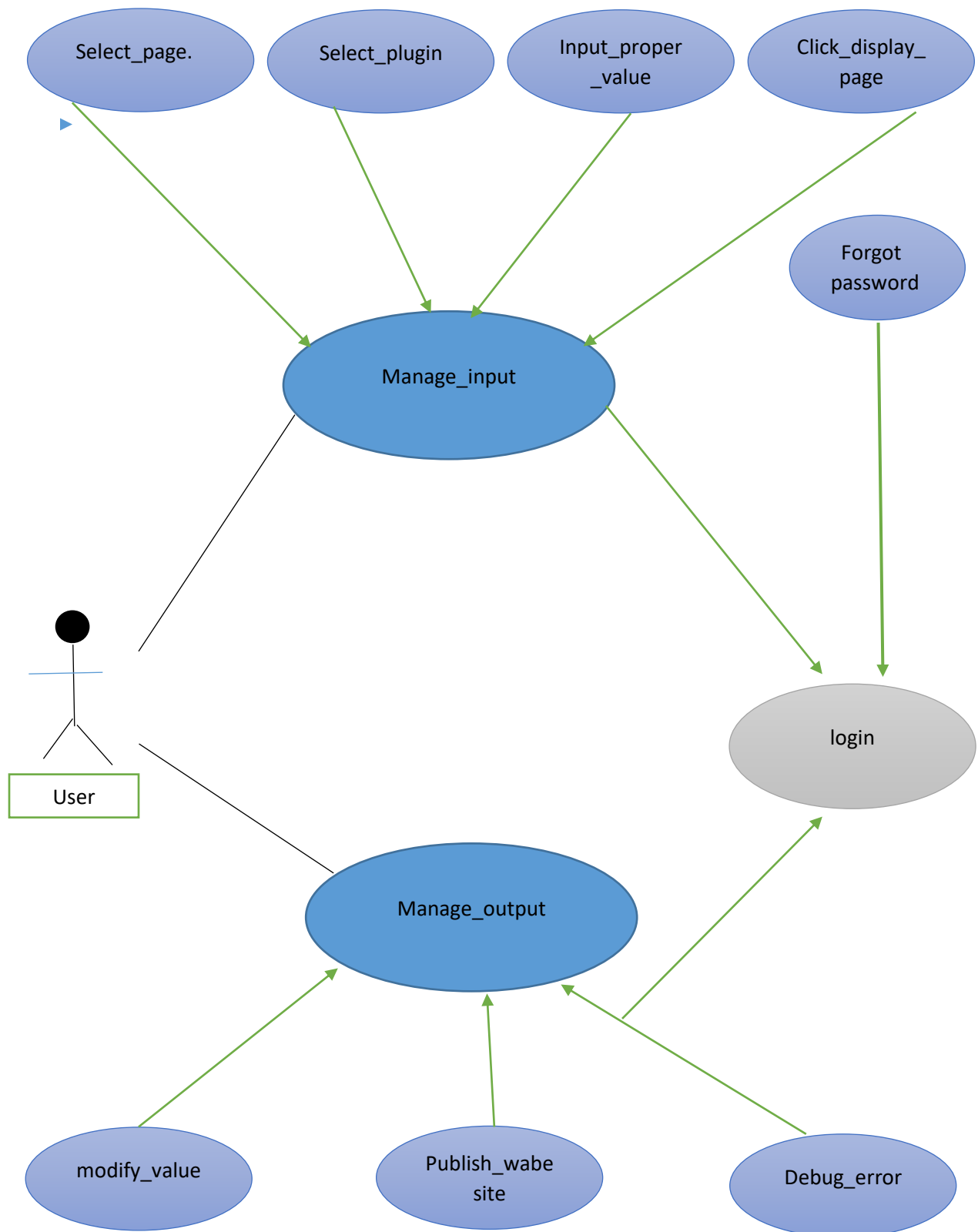
First at all a user need to download my system from cloud or I'll try to made a python package which are downloadable by using pip installer. After downloading my package or system user need type some command which are 'python manage.py createsuperuser' for set the admin password and username and after that my user need to type 'python manage.py runserver' that's time Django will provide localhost .By running this localhost on browser like google chrome my user will find a page which will give congratulation for using my system. Its also our main page which are editable. My system will provide a admin button. By using this button we could find our admin panel which could be controllable by my system user and this admin penal is password protected which is give a safe zone to user admin by using Django supper user functionality. We know about the ability of Django supperuser. From discussion we know that my system will gave some plugin for built my user desired website. This plugin will be available on my admin panel. With the use of my system admin panel any user can made his/her own website with his/her desired plugin. User are able to modify his website by update delete or select different type of plugin. There have an password reset option if my user forgot his/her passcode then he can reset this password by using his/her email. In user case diagram we will describe our system by using diagram of user and some other functionality.

There have two main functionality which are input manager and output manager.

Ther also have some non functional requirement which are

- Portability
- Security
- Maintainability
- Perfomance
- Scalability
- Flexibility

#### 2.1.4 Use Case Diagram:



## 2.2 Instrumentation :

Software instrument:

- Python : Python is an interpreted high-level general-purpose programming language. Its design philosophy emphasizes code readability with its use of significant indentation. Its language constructs as well as its object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.
- Django : Django is a Python-based free and open-source web framework that follows the model–template–views (MTV) architectural pattern. It is maintained by the Django Software Foundation (DSF), an American independent organization established as a non-profit.
- PyCharm: PyCharm is an integrated development environment (IDE) used in computer programming, specifically for the Python language. It is developed by the Czech company JetBrains . It provides code analysis, a graphical debugger, an integrated unit tester, integration with version control systems (VCSes), and supports web development with Django as well as data science with Anaconda.
- Atom: Atom is a free and open-source text and source code editor for macOS, Linux, and Microsoft Windows with support for plug-ins written in JavaScript, and embedded Git Control. Developed by GitHub, Atom is a desktop application built using web technologies. Most of the extending packages have free software

licenses and are community-built and maintained. Atom is based on Electron (formerly known as Atom Shell), a framework that enables cross-platform desktop applications using Chromium and Node.js. Atom is written in CoffeeScript and Less, but much of it has been converted to JavaScript.

- HTML: The HyperText Markup Language, or HTML is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript.
- CSS: Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language such as HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.
- JavaScript : JavaScript often abbreviated as JS, is a programming language that conforms to the ECMAScript specification. JavaScript is high-level, often just-in-time compiled, and multi-paradigm. It has curly-bracket syntax, dynamic typing, prototype-based object-orientation, and first-class functions.
- MySQL : MySQL is an open-source relational database management system (RDBMS). Its name is a combination of "My", the name of co-founder Michael Widenius's daughter, and "SQL", the abbreviation for Structured Query Language. A relational database organizes data into one or more data tables in which data types may be related to each other; these relations help structure the data. SQL is a language programmers use to create, modify and extract data from the relational database, as well as control user access to the database. In addition to

relational databases and SQL, an RDBMS like MySQL works with an operating system to implement a relational database in a computer's storage system, manages users, allows for network access and facilitates testing database integrity and creation of backups.

**Hardware instrument:**

- Processor : Standard Processor
- Ram : 4GB or more
- Hard Disk : 40GB or more
- Monitor : Standard color monitor
- Keyboard : Standard keyboard
- Monitor : Standard mouse

**2.3.Data collection and analysis procedure:**

Create a plan for building my system is a difficult task. For this I need some analysis of different types of CMS. Some of them are described in literature review. With this some other CMS designs help me to increase my knowledge about CMS. And that thing helped me to create a plan.

## **CHAPTER 3**

### **EXPECTED OUTCOME**

This system have the ability to building a website automatically or by using some user input.It will decrease the time rate which was needed to be made a website.And also it's a flexible system and user friendly system understand and word with this system will be not a hard task for my system user

## **CHAPTER 4**

### **CONCLUTION**

#### **4.1. Conclution :**

I strongly believe that within some month I will build my system perfectly.And with my system my user will find a better experience.

#### **4.2 Future work:**

I will try to made some AI releted plugin for my website.with that I will try to increase the adding flexibility of plugin.I will try to made a CMS in future which will be able to made decentralized web-based system.

## CHAPTER 5

### REFERENCES

- [1] Wagtil . 2014 . [https://docs.wagtail.io/en/stable/?\\_ga=2.21032377.363196393.1633072343-1998577556.1631180409](https://docs.wagtail.io/en/stable/?_ga=2.21032377.363196393.1633072343-1998577556.1631180409) (Accesed at 2<sup>nd</sup> September,2021)
- [2] DjangoCMS .2007 . <https://docs.django-cms.org/en/latest/>( Accesed at 1<sup>st</sup> September 2021)
- [3] Python 3.9.7 .1991 . <https://docs.python.org/3/>( Accesed at 2<sup>nd</sup> January 2021)
- [4] Zed Shaw . 2013 . Learn Python the Hard Way.1<sup>st</sup> Edition.
- [5] HTML .1993. <https://www.w3schools.com/html/>( Accesed at 8<sup>th</sup> October 2020)
- [6] CSS . 1996 . <https://www.w3schools.com/css/> ( Accesed at 7<sup>th</sup> October 2020)
- [7] Java Scripts . 1995 . <https://www.w3schools.com/js/> (Accesed at 9<sup>th</sup> October 2020)
- [8] MySQL . 1995 . <https://dev.mysql.com/doc/> ( Accesed at 10<sup>th</sup> October 2020)
- [9] Django 3.2 .2005 . <https://docs.djangoproject.com/en/3.2/>( Accesed at 11<sup>th</sup> October 2020)
- [10] William S. Vincent. Django for Beginners: Build websites with Python and Django 2<sup>nd</sup> Edition.
- [11] Douglas Crockford . 2008 . JavaScript: The Good Parts 1<sup>st</sup> Edition.
- [12] Seyed M. M. Tahaghoghi .2006 . Learning MySQL 1<sup>st</sup> Edition.

## PROJECT TIMETABLE

Here I have given a timetable to complete my project. This is not accurate timetable. It is an approximate timetable. I try my best to complete my project in this time period.

| Activity Plan                              | Month 1 | Month 2 | Month 3 |
|--------------------------------------------|---------|---------|---------|
| Selection of Project and Literature survey |         |         |         |
| Front-end Design                           |         |         |         |
| Backend Development                        |         |         |         |
| Testing and modification                   |         |         |         |
| Documentation                              |         |         |         |