**News Portal**

**A PROJECT REPORT**

**Submitted By**

**GAURAV SHAKYA**

**University Roll No 1802914004**

**Submitted in partial fulfillment of the**

**Requirements for the Degree of**

**MASTER OF COMPUTER APPLICATION**

**Under the Supervision of**

**Dr. SANGEETA ARORA**

**ASSOCIATE PROFESSOR**



**Submitted to**

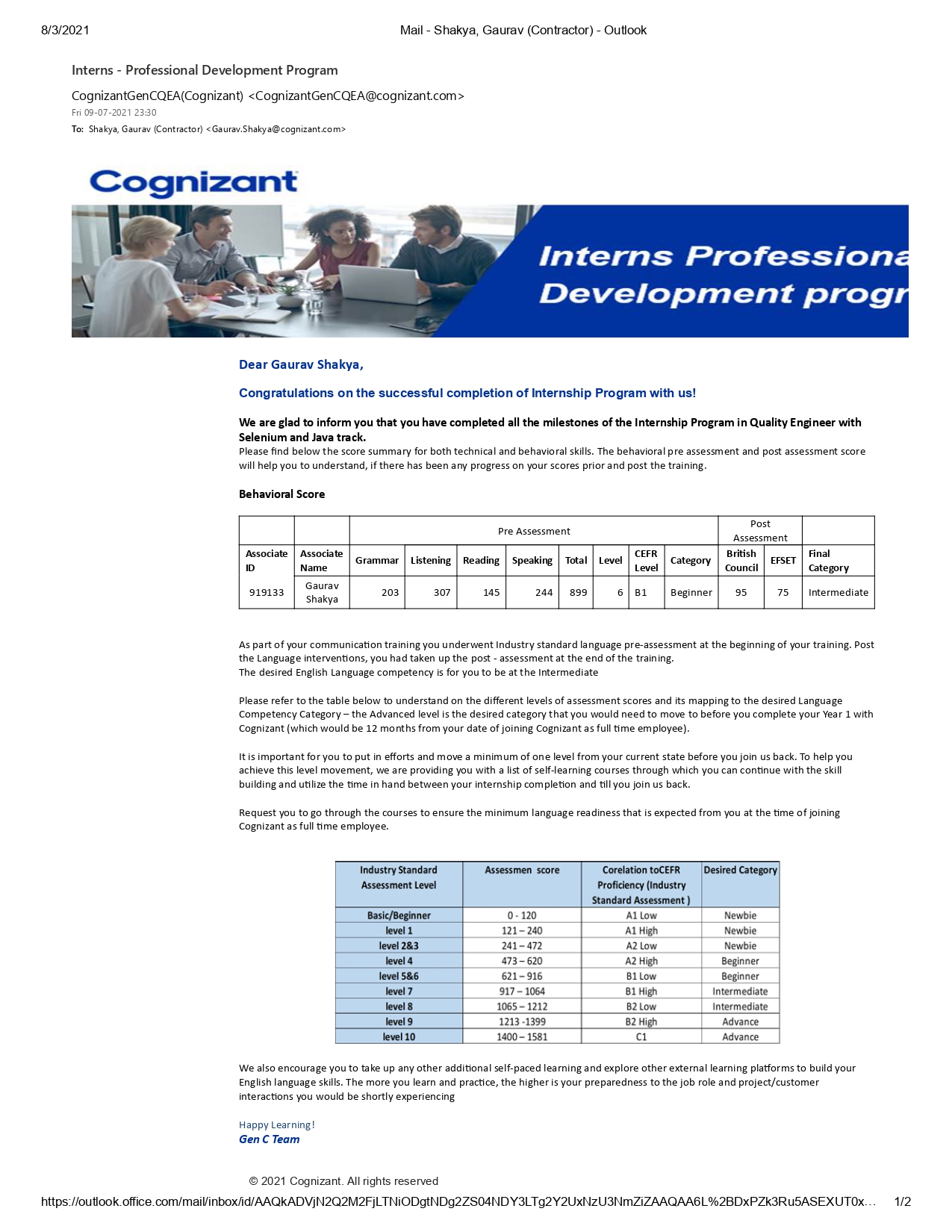
**Department Of Computer Applications**

**KIET Group of Institutions, Ghaziabad**

**Uttar Pradesh-201206**

**(AUGUST 2021)**

**Training Certificate**

**CERTIFICATE**

Certified **that Gaurav Shakya (University Roll No 1802914004)** has carried out the project work having “**News Portal**” for Master of Computer Applications from Dr. A.P.J. Abdul Kalam Technical University (AKTU**)** (formerly UPTU),Technical University, Lucknow under my supervision. The project report embodies original work, and studies are carried out by the student himself/herself and the contents of the project report do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University/Institution.

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Date:

**Dr. Sangeeta Arora**

**Associate Professor**

**Department of Computer Applications**

**KIET Group of Institutions, Ghaziabad**

**Signature of Internal Examiner Signature of External Examiner**

**Dr. Ajay Srivastava**

**Head, Department of Computer Applications**

**KIET Group of Institutions, Ghaziabad**

**NEWS PORTAL**

**GAURAV SHAKYA**

**ABSTRACT**

The “Online News Portal” has been developed to override the problems prevailing in the practicing manual system. This software is supported to eliminate and in some cases reduce the hardships faced by this existing system. Online news portal can lead to error free, secure and reliable fast news management system.

Online news portal allows users to read up to date news related to many fields like entertainment, national, international, business, sports etc. without any payment or login. He can also contact us to give suggestions and can also give us feedback related to our site.

New news can be added only by the admin and only admin have the right to update or delete any news.

The objective of this project is to develop a web application for Online News Paper website that can aware the people

 The objective of this project is to provide the daily news.

 The objective of this project is to provide the breaking news.

 It makes use of various technologies to get required crime oriented information more quickly, easily, colorfully and attractively.

**ACKNOWLEDGEMENT**

Success in life is never attained single handedly. My deepest gratitude goes to my Project supervisor, **Dr. Sangeeta Arora** for her guidance, help and encouragement throughout my research work. Their enlightening ideas, comments, and suggestions.

Words are not enough to express my gratitude to **Dr. Ajay Kumar Srivastava, Professor and Head, Department of Computer Applications**, for his insightful comments and administrative help at various occasions.

Fortunately, I have many understanding friends, who have helped me a lot on many critical conditions.

Finally, my sincere thanks go to my family members and all those who have directly and indirectly provided me moral support and other kind of help. Without their support, completion of this work would not have been possible in time. They keep my life filled with enjoyment and happiness.

**Gaurav Shakya**

**(1802914004)**

**Table of Contents**

Trainng Certificate ii

Certificate iii

Abstract iv

Acknowledgement v

Table of content vi-viii

List of Figures ix

CHAPTER 1 Introduction 1-7

1.1 Project description 1

1.2 Project Purpose 1

1.3 Software Requirement Specification of “News Portal” 2-7

1.3.1 Introduction 2

1.3.1.1 Purpose of this topic 2

1.3.1.2 Audience and Reading Suggestions 2

1.3.1.3 Scope of Project and Document 3

* + 1. General Description 3-4
       1. Quality Function Deployment of “News Portal” 3-4
    2. Specific Requirement 4-5

1.3.3.1External Interface Requirements of the System 4

1.3.3.2 User Characteristics for the System 5

1.3.4 Project Model Used 6-7

CHAPTER 2 Literature Review 8-14

CHAPTER 3 Feasibility Study 15-17

3.1 Introduction 15

3.1.1 Technical feasibility 15-16

3.1.2 Economical Feasibility 16

3.1.3 Operational Feasibility 16

3.1.4 Social and Behavioral Feasibility 16

3.1.5 Legal Feasibility 16-17

CHAPTER 4 System Design 18-23

4.1 Introduction 18

4.2 System Architecture 18

4.3 Module in the System 18-19

4.4 Methodology Used 19-20

4.5 Data Flow Diagrams 21-23

4.5.1 0-level DFD 22

4.5.2 1-level DFD 23

4.6 Entity Relationship Diagram 23

CHAPTER 5 Source Code 24-57

CHAPTER 6 Snapshot 58-62

CHAPTER 7 Testing 63-66

7.1 Introduction 63

7.2 Types of Testing 63-66

CHAPTER 8 Conclusion and Future Scope 67

8.1 Conclusion 67

8.2 Future Scope 67

CHAPTER 9 References 68-70

**List of Figures**

[**Figure 1**-Phases of Iterative Model 7](#_Toc42360805)

[**Figure 2**-Use Case Diagram 20](#_Toc42360806)

[**Figure 3**-0-Level DFD 22](#_Toc42360807)

[**Figure 4**-1 Level DFD](#_Toc42360808) 22

[**Figure 5**-1-ERD](#_Toc42360809) 23

[**Figure 6**-Home Page 58](#_Toc42360810)

[**Figure 7**-Admin Dashboard 59](#_Toc42360811)

[**Figure 8**-News Category 60](#_Toc42360812)

[**Figure 9**-Manage Category Page 61](#_Toc42360813)

**Figure 10-**Add Post 62

**CHAPTER 1**

**INTRODUCTION**

* 1. **PROJECT DESCRIPTION**

The “Online News Portal” has been developed to override the problems prevailing in the practicing manual system. This software is supported to eliminate and in some cases reduce the hardships faced by this existing system. Online news portal can lead to error free, secure and reliable fast news management system.

Online news portal allows users to read up to date news related to many fields like entertainment, national, international, business, sports etc. without any payment or login. He can also contact us to give suggestions and can also give us feedback related to our site.

New news can be added only by the admin and only admin have the right to update or delete any news.

.

Managing your online news portal management system may seem tricky, but this is part of user service system.

* 1. **PURPOSE OF PROJECT**

 The objective of this project is to develop a web application for Online News Paper website that can aware the people

 The objective of this project is to provide the daily news.

 Anytime, anywhere, anyone can know about the news or information by internet at low cost

* 1. **Software Requirements Specification of “News Portal”**

This topic covers the requirements specification of our “News Portal”. It includes the specification of this documentation with general description, specific requirements, and analysis of models. It also includes changes management of this requirements specification in case of any change

* + 1. **Introduction**

In this section, the documentation of this report is specified. It specifies the document convention, document scope and provides a suggestion for the readers of the document.

**1.3.1.1Purpose of this Topic**

This Software Requirements Specification (SRS) part is intended to give a complete overview of our Project the “News Portal” including the action flow, initial user interface and story therein. The SRS document details all features upon which we have currently decided with reference to the manner and importance of their implementation.

**1.3.1.2 Audience and Reading Suggestions**

This project is a prototype for News Portal. This has been implemented under the guidance of college professors. This project is useful for anyone to check daily activities around the world.

Although the document may be read from front to back for a complete understanding of the project, it was written in section and hence can be read as such. For an overview of the document and the project itself, see Overall Description. For a detailed description of the system elements and their interaction with the user, read System Features. Readers interested in the system interface and navigation between different front-end menus should go through External Interface Requirements.

* + - 1. **Scope of Project and Document**

The project has a wide scope, as it is not intended to a particular organization. This project is going to develop generic software, which can be applied by any businesses organization. More over it provides facility to its users. Also the software is going to provide a huge amount of summary data. It helps to user to read up to date news related to many fields like entertainment, national, international, business, sports etc. without any payment or login. The purpose of project is to online management of news i.e adds latest news and also categorizes them

**1.3.2 General Description**

It specifies the QFD (Quality Function Deployment) of our software and the User Story of it.

**1.3.2.1Quality Function Deployment of “News Portal”**

Quality Function Deployment is a technique that translates the needs of the customer into technical requirements for software. It concentrates on maximizing customer satisfaction from the Software engineering process. With respect to our project the following requirements are identified by a QFD.

* Normal Requirements.
* Expected Requirements.

* **Normal Requirements**

Normal requirements consist of objectives and goals that are stated during the meeting with the relevant people. Normal requirements of our project are:

1. Minimum maintenance cost.
2. Availability of expected requirements within the PC configuration.
3. Easy to operate.
4. The software with measured coding, professional thinking.

* **Expected Requirements**

These requirements are implicit to the system and may be so fundamental that the relevant person does not explicitly state them. Their absence will be a cause for dissatisfaction.

1. Maximum high regulation with minimum hardware.
2. We may provide a user friendly interface to users for easy commenting and form filling.
3. Easy to update.

**1.3.3 SPECIFIC REQUIREMENTS**

This section covers the project external requirements of our software and indicates the User Characteristics for this project.

**1.3.3.1External Interface Requirements of the System**

* **User Interfaces**

Every website must have a home page and a site map to make the software user friendly enough and user can easily fulfill their need. Nav Bar is also an important component. We have added the Home Page, different Forms’ snapshots.

* **Hardware Interfaces**

“News Portal” application designed specifically for Windows platform and is functional on both Desktop and Laptop. And a browser which supports HTML& Javascript.

* **Software Interface**

“News Portal” has been developed using a series of Website Development with Django platform. Working tools and Platform

* **Operating System-** Windows/Linux
* **Technology Used-** Python Django
* **Front-end Technologies-**HTML5,CSS,Bootstrap,Javascript
* Python3 (3.8 32bit)
* **Code Editor-** PyCharm 2020.1.1
* **Database-** SQLite

**1.3.3.2 User Characteristics for the System**

It helps to user to read up to date news related to many fields like entertainment, national, international, business, sports etc. without any payment or login.

* + 1. **PROJECT MODEL USED**

**Iterative Enhancement Model**

* This model has the same phases as the waterfall model, but with fewer restrictions.
* Generally the phases occur in the same order as in the waterfall model, but they may be conducted in several cycles.
* Useable product is released at the end of the each cycle, with each release providing additional functionality. Customers and developers specify as many requirements as possible and prepare a SRS document. Developers and customers then prioritize these requirements. Developers implement the specified requirements in one or more cycles of design, implementation and test based on the defined priorities.

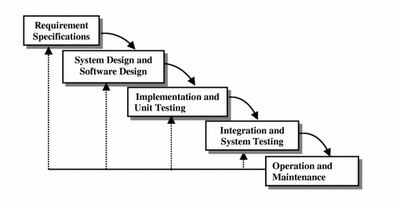
The procedure itself consists of the initialization step, the iteration step, and the Project Control List. The initialization step creates a base version of the system. The goal for this initial implementation is to create a product to which the user can react. It should offer a sampling of the key aspects of the problem and provide a solution that is simple enough to understand and implement easily. To guide the iteration process, a project control list is created that contains a record of all tasks that need to be performed. It includes such items as new features to be implemented and areas of redesign of the existing solution. The control list is constantly being revised as a result of the analysis phase.

The iteration involves the redesign and implementation of iteration is to be simple, straightforward, and modular, supporting redesign at that stage or as a task added to the project control list. The level of design detail is not dictated by the iterative approach. In a light-weight iterative project the code may represent the major source of [documentation](http://en.wikipedia.org/wiki/Software_documentation) of the system; however, in a critical iterative project a formal [Software Design Document](http://en.wikipedia.org/wiki/Software_Design_Document) may be used. The analysis of an iteration is based upon user feedback, and the program analysis facilities available. It involves analysis of the structure, modularity, [usability](http://en.wikipedia.org/wiki/Usability), reliability, efficiency, & achievement of goals. The project control list is modified in light of the analysis results.

### PHASES:

Incremental development slices the system functionality into increments (portions). In each increment, a slice of functionality is delivered through cross-discipline work, from the requirements to the deployment. The unified process groups increments/iterations into phases: inception, elaboration, construction, and transition.

* Inception identifies project scope, requirements (functional and non-functional) and risks at a high level but in enough detail that work can be estimated.
* Elaboration delivers a working architecture that mitigates the top risks and fulfills the non-functional requirements.
* Construction incrementally fills-in the architecture with production-ready code produced from analysis, design, implementation, and testing of the functional requirements.
* Transition delivers the system into the production operating environment.

****

**Figure 1Phases of Iterative Model**

**CHAPTER 2**

**LITERATURE REVIEW**

# 2.1 Configurable hardware components generator in Python

Traditional hardware description languages are limited when describing highly configurable and reusable hardware components. The paper introduces methodology based on a Python language for design of hardware component generators on higher abstraction level. The scripting language is used to produce customizable cycle accurate hardware behavior and open-source tools provide automatic conversion to register transfer level. A case study presents configurable graphics processing component design. The automatic interface insertion, state machine synthesis and pipeline configuration provided by the proposed methodology enables efficient hardware description and design space exploration.

**2.2 A Survey: How Python Pitches in IT-World**

This paper lights on Python amongst other different programming paradigms used in the IT World, which enhances development speed. Although, Python was conceptualized in the late 1980s and after its implementation in 1989, it has emerged as a new multi-paradigm language platform with advent of Big Data. Python includes various data structures, standard libraries with the implementation of sentiment analysis and data science code. The real aim is to provide awareness to all the programmers about various facts of python language. It tells how Python works with various commercial and social communities and provides complete and desirable results. There are many areas and applications where Python makes its own stand as compared to other programming languages.

**2.3 JS Optimizer: An Extensible Framework for JavaScript Program Optimization**

JavaScript has become a popular programming language. It is widely used in both client-side and server-side programming in web applications. The robustness and performance of JavaScript programs become vital. Unfortunately, real-world JavaScript programs often suffer from various issues. In this work, we present nine issue patterns derived from open-source projects and propose a general static analysis framework, JSOptimizer, to help detect such patterns of issues and optimize the code accordingly. Comparing to existing work, JSOptimizer is not only highly extensible but also performs code optimizations automatically. We applied JS Optimizer to seven real open-source JavaScript projects and five bugs detected by it have been confirmed by developers. Besides, we conducted a case study based on a popular project and found that addressing the issues detected by our framework can speed up the original project by over 300%. This shows the usefulness of JS Optimizer.

**2.4 Javascript ahead-of-time compilation for embedded web platform**

Web applications (apps) programmed using HTML, CSS, and JavaScript have advantages in portability and productivity, compared to Android or IOS apps. However, web apps are involved with some performance issue, due to JavaScript with its dynamic typing and prototypes which are difficult to execute efficiently. One popular way of accelerating JavaScript is using the just-in-time compilation (JITC), which translates the JavaScript source code to the machine code at runtime. Unfortunately, JavaScript JITC for web apps suffers from the parsing and compilation overhead seriously, which offsets the performance gain of executing the compiled code. In this paper, we propose ahead-of-time compilation (AOTC) of JavaScript at the client device. We save the code generated by the JITC at the first run of the web app, so that we can reuse the code in the next runs to remove the parsing and the compilation overhead.

**2.5 Modern JavaScript frameworks: A Survey Study**

With the increasing popularity of the web, some new web technologies emerged and introduced dynamics to web applications, in comparison to HTML, as a static programming language. JavaScript is the language that provided a dynamic web site which actively communicates with users. JavaScript is used in today's web applications as a client script language and on the server side. The JavaScript language supports the Model View Controller (MVC) architecture that maintains a readable code and clearly separates parts of the program code. The topic of this research is to compare the popular JavaScript frameworks: AngularJS, Ember, Knockout, Backbone. All four frameworks are based on MVC or similar architecture. In this paper, the advantages and disadvantages of each framework, the impact on application speed, the ways of testing such JS applications and ways to improve code security are presented.

**2.6 Integration of HTML pages in Web Pages**

The growing number of Web pages on the Internet introduces a need to combine and integrate information from HTML tables of different Web pages that contain similar information into a single Web page, especially information from the same domain of interest. This paper presents an approach of HTML table integration by combining several existing methods that are proved to solve different issues in the integration processes. The integration of HTML table consists of three phases: (1) extraction of the structure of the tables; (2) integration of the tables' schema; (3) integration of the data values. To solve the conflicts in semantics and naming in the tables schema, domain-ontology is used.

**2.7 Extracting Elements of HTML Documents**

Information on the Web, which are conglomeration of heterogeneous data, such as texts, images and audio clips, are often accessed through documents written according to the HTML specification. According to the HTML specification, HTML documents are semistructured in nature. We propose a high-level stack machine (HSM) which accesses an HTML document through its URL and constructs a semistructured data graph (SDG) of the document. The SDG of an HTML document H precisely captures the structure of the semistructured data embedded in H based on the dependency relationship among the data objects in H. HSM is configurable to accommodate a user's interest with respect to the HTML elements in H to be considered during the construction process of the SDG of H.

**2.8 A Webpage Data Hiding Method by Using Tag and CSS Attribute Setting**

Computer networks connection becomes the most important way for people to contact each other, share information, and transmit privacy data. Because the Internet is not secure enough, data hiding techniques provide a good manner to deliver secret data with security. HTML webpage not only can be used to advertise a company's product but also used to share someone experience or knowledge. The HTML file is different from a digital image because it is composed of tags but pixels. CSS provides more options and assistance to help HTML file coding to create colorful web pages. The proposed method utilizes both HTML and CSS's properties to achieve the goal of secret data delivery. The experimental results indicate that the proposed method has a larger embedding capacity than others.

**2.9 High performance PL/SQL programming**

Performance engineering is a vital aspect in PL/SQL programming, as most of the Database associated applications are built with PL/SQL Code. There subsists many ways of writing PL/SQL statement to retrieve same result set, but the approach which levies minimum impact on DBMS engine is always esteemed. Most commercial transaction scripts are written using PL/SQL code. The proposed idea(s) are envisioned to oblige as tuning utility and benchmark for tuning PL/SQL queries. This paper enunciates different techniques that can reduce time and space complexity of a native SQL query and PL/SQL script. Our analysis reduced the rate of Context switching (an overhead) among SQL engine and PL/SQL engine.

**2.10 On the IO Characteristics of the SQLite Transactions**

This work is dedicated to study the IO characteristics of SQLite transaction in Android platform. We collect the block level IO trace from for six months. We develop an elaborate pattern matching algorithm. It allows us to identify the individual SQLite transactions from the raw IO trace, which is essentially an interleaved mixture of IO requests from concurrently running smartphone applications. Among the various observations obtained from the study, we can summarize the key findings as follows. We carefully believe that these deserve special attention. First, SQLite transaction is under extreme inefficiency. In an SQLite transaction, the IO's for SQLite journaling and EXT4 filesystem journaling account for over 75% of the entire IO volume in a transaction. Second, the suspend and the wakeup feature of the smartphone can leave the SQLite transaction to an extreme delay, a few minutes.

**2.11 A Method of Optimizing Django Based on Greedy Strategy**

Django on Python does well in agile web development. Python is dynamic programming language, as is known, its runtime efficiency is low. the question is how it will affect Django's efficiency. This Paper answered the question from the angle of memory optimization, by improving the efficiency of Python in memory use, we will see how it will affect django. Python manages non-container objects with the technology of memory pool, after a consequence of allocate and free operations, it will produce memory fragment, and the garbage collection module can hardly handle it. This paper proposed a new non-container object management greedy algorithm, it can minimize the probability of generating fragments, thus the garbage collection module can collect garbage non-container objects in time. We conduct our experiment on the benchmark of an open-source project named Unladen-Swallow. When Django renders a template, it will save about 10% memory, with almost the same time consuming.

**2.12 Development of a Continuous System Simulation Engine in Python Programming Language**

This paper portrays the development of an simulation engine in Python programming language with focus on the continuous system simulation. Features of the engine are based on CSMP block-oriented language and software for continuous system simulation. The aim of the paper is threefold. Firstly, the paper describes the need and usage of such engine. Second goal is to analyze Python concepts used for the implementation of the simulation engine - API development, communication with smart systems, real-time data processing, integration of differential equations, threading and parallel programming. And last, this paper will depict the engine architecture, use cases and usage of the analyzed concepts in the development process.

**2.13 Extraction of information from log files Using Python Programming and Tableau**

Application servers generate daily log files with a significant part of their activity. This information is recorded sequentially over time but mixes various types of information. The absence of a standard for formatting the data record and the respective volume, make it difficult to extract the corresponding information. The lack of work, specifically in the treatment of SOA server log files, did not allow the comparisson with pre-existing Key Performance Indicators (KPI) or a set of best practices that could be followed. This work results in a description of the process that can serve as a guide for: definition of a logging structure; construction of a data extraction process; definition of a data structure to support the extracted information; definition of control metrics; definition of analysis and control processes for the extracted data.

**2.14 A new model for measuring the complexity of SQL commands**

Software that uses database system has different complexity with software without database system, therefore required model that can pay attention to attribute or parameter in software with the database system. In this study, we propose a new model for measuring software complexity by taking into account the SQL query attributes in the database system. The model is divided into five stages, among others: reading program module, forming SQL query model, giving SQL query weight, calculating SQL complexity, and module complexity result. The results of a series of tests performed, showing the accuracy value in accordance with the assessment made by the expert. Based on the test results, the proposed model can be applied to measure the complexity of SQL queries.

**2.15 Automatically Repairing SQL Faults**

SQL is the standard database language, yet SQL statements can be complex and expensive to debug by hand. Automatic program repair techniques have the potential to reduce cost significantly. A previous attempt to repair SQL faults automatically used a decision tree (DT) algorithm that succeeded in some cases, but also generated many patches that passed the automated tests but that were not acceptable to the engineers. This paper proposes a novel fault localization and repair technique to repair faulty SQL statements. It targets faults in two common SQL constructs, JOIN and WHERE. It identifies the fault location and type precisely, and then creates a patch to fix the fault. We implemented this technique in a tool, and evaluated it on five medium to large-scale databases using 825 faulty queries with various complexity and faulty types.

**CHAPTER 3**

**FEASIBILITY STUDY**

**3.1 INTRODUCTION**

A feasibility study is an analysis that takes all of a project's relevant factors into account—including economic, technical, legal, and scheduling considerations—to ascertain the likelihood of completing the project successfully. Project managers use feasibility studies to discern the pros and cons of undertaking a project before they invest a lot of time and money into it.

The feasibility study is conducted to check whether the candidate system is feasible. The system which is selected to be the best against the criteria is there after designed and developed. The feasibility study takes in to consideration, the risks involved in the project development beforehand. Therefore in this phase we have to do feasibility study which is the test of the website according to its work ability, impact on the organization, ability to meet user need and effective use of resources. We do the feasibility study for website to analyze the risks, costs and benefits relating to economics, technology and user organization. There are several types of feasibility depending on the aspect they cover. Import of these includes:

**3.1.1 TECHNICAL FEASIBILITY**

This is an important outcome of preliminary investigation. It comprise of following questions:-

* Can the work of project be one with the current equipment, existing software and available man power resource?
* If Technology is required what are the possibilities that it can be developed?

We can strongly say that it is technically feasible, since there will not be much difficulty in getting resources for the development and maintaining the system as well. All the resources needed for the development of the software as well as the maintenance of the same is available in the organization.

**3.1.2 ECONOMICAL FEASIBILITY**

It deals with question related to the economy. It comprise of the following questions:-

* Are there sufficient benefits in creating the system to make the cost acceptable?
* Are the costs of not creating the system so great that the project must be undertaken?

Development of this application is highly economically feasible. The organization needed not spend much money for the development of the system already available. The only thing is to be done is making an environment for the development with an effective supervision. Even after the development, the organization will not be in condition to invest more in the organization.

**3.1.3 OPERATIONAL FEASIBILITY**

The operational feasibility consists of the following activity:-

* + - * Will the system be useful if it is developed &implemented?

**3.1.4 SOCIAL AND BEHAVIORAL FEASIBILITY**

It deals with the various issues related to the human behavior like:-

* + - * Whether the user be able to adapt a new change or not?
      * Whether the ambiance we are providing suits the user or not?

**3.1.5 LEGAL FEASIBILITY**

It deals with the question related to the legal issues. It comprise of the following questions:-

* Contract Signing
* Software License agreement
  + - * Issues related to cyber laws.
      * Legal issues relating to the man power contract.

**CHAPTER 4**

**SYSTEM DESIGN**

**4.1 INTRODUCTION**

**System design** is the phase that bridges the gap between problem domain and the existing system in a manageable way. This phase focuses on the solution domain, i.e. *“how to implement?”*

**4.2SYSTEM ARCHITECTURE**

System design is the process of defining system architecture, modules and interfaces for the proposed system to satisfy specified requirement.

**4.3 Modules in the System**

The system comprises of 2 major modules as follows:

1. **Admin:**
   1. Secure admin login system
   2. Admin dashboard
   3. **Category** – In this section admin can add/update/delete the category. Admin can also restore deleted category
   4. **Sub- Category** – In this section admin can add/update/delete the Subcategory. Admin can also restore deleted Subcategory
   5. **Post –** Admin can add /update / delete news posts. admin can also view deleted news post in trash post section and restore deleted posts.
   6. **Pages –**Admin can manage the contact of about us and contact us page.
   7. **Comments –**Admin can approve/ unapprove / delete reader comments.
2. **User:**

Anyone can read the news and also search for particular news. The reader can leave comments on the particular news

**4.4METHODOLOGY USED**

**4.4.1 Python**

Python is a widely used general-purpose, high level programming language. It was initially designed by Guido van Rossum in 1991 and developed by Python Software Foundation. It was mainly developed for emphasis on code readability, and its syntax allows programmers to express concepts in fewer lines of code.

Python is a programming language that lets you work quickly and integrate systems more efficiently.

Python is dynamically typed and garbage-collected. It supports multiple programming paradigms, including procedural, object-oriented, and functional programming. Python is often described as a "batteries included" language due to its comprehensive standard library.

**4.4.2 HTML**

HTML (Hypertext Markup Language) is the set of markup symbols or codes inserted in a file intended for display on a World Wide Web browser page. The markup tells the Web browser how to display a Web page's words and images for the user. Each individual markup code is referred to as an element (but many people also refer to it as a tag). Some elements come in pairs that indicate when some display effect is to begin and when it is to end.

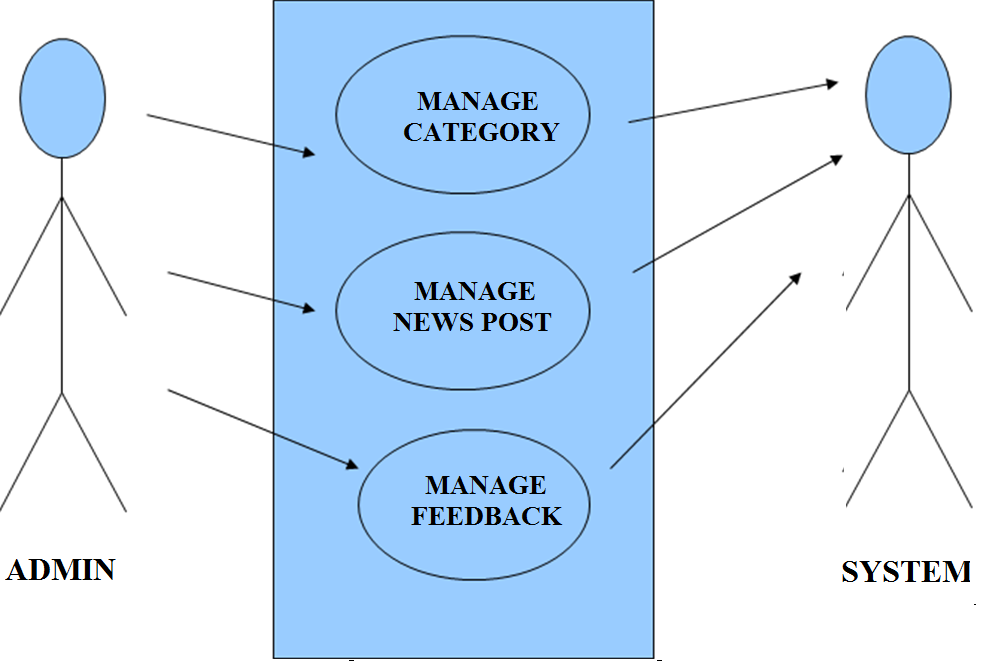
**4.4.3.Javascript**

JavaScript is a programming language commonly used in web development. It was originally developed by Netscape as a means to add dynamic and interactive elements to websites. While JavaScript is influenced byJava, the syntax is more similar to C and is based on ECMAScript, a scripting language developed by Sun Microsystems.

JavaScript is a client-side scripting language, which means the source code is processed by the client's web browser rather than on the web server. This means JavaScript functions can run after a webpage has loaded without COMMUNICATING with the server. For example, a JavaScript function may check a web form before it is submitted to make sure all the required fields have been filled out. The JavaScript code can produce an error message before any information is actually transmitted to the server.

**4.4.4.Django**

Django is a web application framework written in Python programming language. It is based on MVT (Model View Template) design pattern. The Django is very demanding due to its rapid development feature. It takes less time to build application after collecting client requirement.



**Figure 2Use Case Diagram**

**Data Flow Diagram**

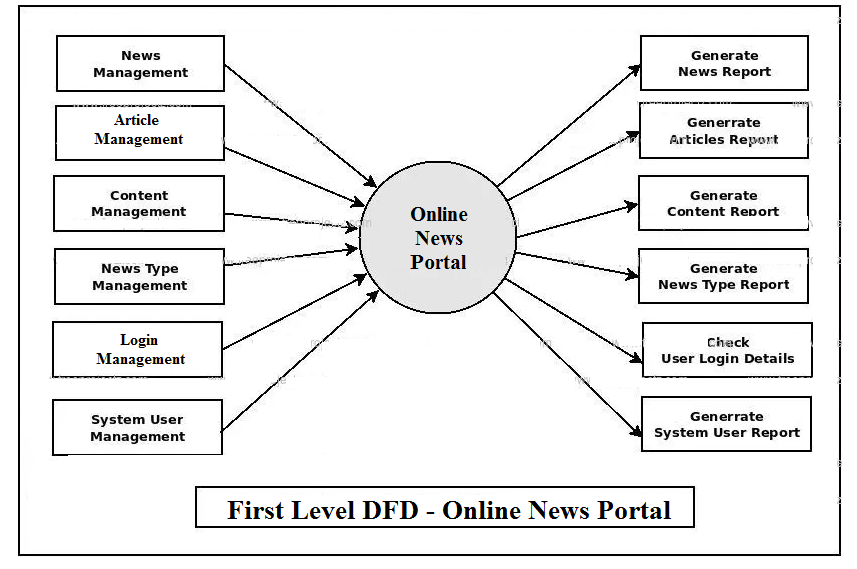
A Data Flow Diagram (DFD) is a graphical representation of the "flow" of data through an Information System. A data flow diagram can also be used for the visualization of Data Processing. It is common practice for a designer to draw a context-level DFD first which shows the interaction between the system and outside entities. This context-level DFD is then "exploded" to show more detail of the system being modeled.

A DFD represents flow of data through a system. Data flow diagrams are commonly used during problem analysis. It views a system as a function that transforms the input into desired output. A DFD shows movement of data through the different transformations or processes in the system.

Dataflow diagrams can be used to provide the end user with a physical idea of where the data they input ultimately has an effect upon the structure of the whole system from order to dispatch to restock how any system is developed can be determined through a dataflow diagram. The appropriate register saved in database and maintained by appropriate

authorities.

**DFD Level 0 Diagram**

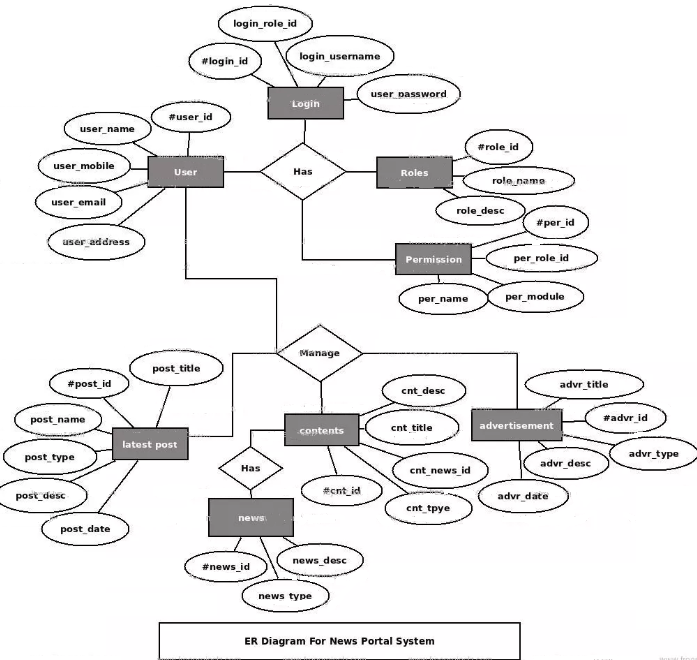
****

**Figure 4- First level DFD**

**Entity Relationship Diagrams (ER-Diagrams):**

An entity-relationship (ER) diagram is a specialized graphic that illustrates the interrelationships between entities in a database. ER diagrams often use symbols to represent three different types of information. Boxes are commonly used to represent entities. Diamonds are normally used to represent relationships and ovals are used to represent attributes

**ENTITY RELATIONSHIP(ER) DIAGRAM**



**Figure 5- ERD**

**CHAPTER 5**

**SOURCE CODE**

**5.1 Home Page Coding**

**{% load static %}**

**{% block body %}**

**<html lang="en">**

**<head>**

**<meta charset="utf-8">**

**<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">**

**<meta name="description" content="">**

**<meta name="author" content="">**

**<title>News Portal | Home Page</title>**

**<!-- Bootstrap core CSS -->**

**<link href="{% static 'vendor/bootstrap/css/bootstrap.min.css' %}" rel="stylesheet">**

**<!-- Custom styles for this template -->**

**<link href="{% static 'css/modern-business.css rel=stylesheet' %}">**

**<!--old start-->**

**<link href="{% static 'old/css/bootstrap.css' %}" rel='stylesheet' type='text/css' />**

**<script src="{% static 'old/js/jquery.min.js' %}"></script>**

**<link rel="stylesheet" href="{% static 'old/css/style.css' %}" type="text/css" media="all">**

**<!-- Custom Theme files -->**

**<link rel="stylesheet" href="{% static 'old/css/touchTouch.css' %}" type="text/css" media="all" />**

**<!-- Custom Theme files -->**

**<script type="application/x-javascript"> addEventListener("load", function() { setTimeout(hideURLbar, 0); }, false); function hideURLbar(){ window.scrollTo(0,1); } </script>**

**<!--webfont-->**

**<link href='http://fonts.googleapis.com/css?family=Monoton' rel='stylesheet' type='text/css'>**

**<link href='http://fonts.googleapis.com/css?family=Raleway' rel='stylesheet' type='text/css'>**

**<!---//End-css-style-switecher----->**

**<script type="text/javascript" src="{% static 'old/js/jquery.fancybox.js' %}"></script>**

**<link rel="stylesheet" type="text/css" href="{% static 'old/css/jquery.fancybox.css' %}" media="screen" />**

**<script type="text/javascript">**

**$(document).ready(function() {**

**/\***

**\* Simple image gallery. Uses default settings**

**\*/**

**$('.fancybox').fancybox();**

**});**

**</script>**

**<!--old end-->**

**</head>**

**<body>**

**<!-- Navigation -->**

**{% include 'header.html' %}**

**<!-- Page Content -->**

**<div class="container">**

**<div class="row" style="margin-top: 4%">**

**<!-- Blog Entries Column -->**

**<div class="col-md-8">**

**<!-- Blog Post -->**

**{% for i in posts %}**

**<div class="card mb-4">**

**<img class="card-img-top" src="{{i.postimage.url}}" alt="{{i.posttitle}}">**

**<div class="card-body">**

**<h2 class="card-title">{{i.posttitle}}</h2>**

**<p><b>Category : </b> <a href="{% url 'categorynews' i.category.id %}">{{i.category.categoryname}}</a> </p>**

**<a href="{% url 'news\_detail' i.id %}" class="btn btn-primary">Read More &rarr;</a>**

**</div>**

**<div class="card-footer text-muted">**

**Posted on {{i.postdate}}**

**</div>**

**</div>**

**{% endfor %}**

**<!-- Pagination -->**

**{% if posts.has\_other\_pages %}**

**<ul class="pagination">**

**{% if posts.has\_previous %}**

**<li><a href="?page={{ posts.previous\_page\_number }}">&laquo;</a></li>**

**{% else %}**

**<li class="disabled"><span>&laquo;</span></li>**

**{% endif %}**

**{% for i in posts.paginator.page\_range %}**

**{% if posts.number == i %}**

**<li class="active"><span>{{ i }} <span class="sr-only">(current)</span></span></li>**

**{% else %}**

**<li><a href="?page={{ i }}">{{ i }}</a></li>**

**{% endif %}**

**{% endfor %}**

**{% if posts.has\_next %}**

**<li><a href="?page={{ posts.next\_page\_number }}">&raquo;</a></li>**

**{% else %}**

**<li class="disabled"><span>&raquo;</span></li>**

**{% endif %}**

**</ul>**

**{% endif %}**

**</div>**

**<!-- Sidebar Widgets Column -->**

**{% include 'sidebaruser.html' %}**

**</div>**

**<!-- /.row -->**

**</div>**

**<!-- /.container -->**

**<!-- Footer -->**

**{% include 'footer.html' %}**

**<!-- Bootstrap core JavaScript -->**

**<script src="{% static 'vendor/jquery/jquery.min.js' %}"></script>**

**<script src="{% static 'vendor/bootstrap/js/bootstrap.bundle.min.js' %}"></script>**

**</head>**

**</body>**

**</html>**

**{% endblock %}**

**NEWS DETAIL PAGE CODING**

**{% load static %}**

**{% block body %}**

**<html lang="en">**

**<head>**

**<meta charset="utf-8">**

**<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">**

**<meta name="description" content="">**

**<meta name="author" content="">**

**<title>News Portal | Home Page</title>**

**<!-- Bootstrap core CSS -->**

**<link href="{% static 'vendor/bootstrap/css/bootstrap.min.css' %}" rel="stylesheet">**

**<!-- Custom styles for this template -->**

**<link href="{% static 'css/modern-business.css rel=stylesheet' %}">**

**<!--old start-->**

**<link href="{% static 'old/css/bootstrap.css' %}" rel='stylesheet' type='text/css' />**

**<script src="{% static 'old/js/jquery.min.js' %}"></script>**

**<link rel="stylesheet" href="{% static 'old/css/style.css' %}" type="text/css" media="all">**

**<!-- Custom Theme files -->**

**<!--webfont-->**

**<!---//End-css-style-switecher----->**

**<!--old end-->**

**</head>**

**<body>**

**<!-- Navigation -->**

**{% include 'header.html' %}**

**<!-- Page Content -->**

**<div class="container">**

**<div class="row" style="margin-top: 4%">**

**<!-- Blog Entries Column -->**

**<div class="col-md-8">**

**<!-- Blog Post -->**

**<div class="card mb-4">**

**<div class="card-body">**

**<h2 class="card-title">{{newspost.posttitle}}</h2>**

**<p><b>Category : </b> <a href="{% url 'categorynews' newspost.category.id %}">{{i.category.categoryname}}</a> |**

**<b> Posted on </b>{{newspost.postdate}}</p>**

**<hr />**

**<img class="img-fluid rounded" src="{{newspost.postimage.url}}" alt="{{newspost.posttitle}}">**

**<p class="card-text">**

**<br>**

**{{newspost.postdetail | safe}}**

**<br><br><br>**

**</p>**

**</div>**

**<div class="card-footer text-muted">**

**</div>**

**</div>**

**</div>**

**<!-- Sidebar Widgets Column -->**

**{% include 'sidebaruser.html' %}**

**</div>**

**<!-- /.row -->**

**<!---Comment Section --->**

**<div class="row" style="margin-top: -8%">**

**<div class="col-md-8">**

**<div class="card my-4">**

**<h5 class="card-header">Leave a Comment:</h5>**

**<div class="card-body">**

**<form name="Comment" method="post">**

**{% csrf\_token %}**

**<div class="form-group">**

**<input type="text" name="name" class="form-control" placeholder="Enter your fullname" required>**

**</div>**

**<div class="form-group">**

**<input type="email" name="emailid" class="form-control" placeholder="Enter your Valid email" required>**

**</div>**

**<div class="form-group">**

**<textarea class="form-control" name="commentmsg" rows="3" placeholder="Comment" required></textarea>**

**</div>**

**<button type="submit" class="btn btn-primary" name="submit">Submit</button>**

**</form>**

**</div>**

**</div>**

**<!---Comment Display Section --->**

**{% for i in comment %}**

**<div class="media mb-4">**

**<img class="d-flex mr-3 rounded-circle" src="images/usericon.png" alt="">**

**<div class="media-body">**

**<h5 class="mt-0">{{i.name}} <br />**

**<span style="font-size:11px;"><b>at</b> {{i.cdate}}</span>**

**</h5>**

**{{i.commentmsg}} </div>**

**</div>**

**{% endfor %}**

**</div>**

**</div>**

**</div>**

**<!-- Footer -->**

**{% include 'footer.html' %}**

**<!-- Bootstrap core JavaScript -->**

**<script src="{% static 'vendor/jquery/jquery.min.js' %}"></script>**

**<script src="{% static 'vendor/bootstrap/js/bootstrap.bundle.min.js' %}"></script>**

**</body>**

**</html>**

**{% ifequal error "no" %}**

**<script>**

**alert('Comment successfully submit. Comment will be display after admin review');**

**</script>**

**{% endifequal %}**

**{% ifequal error "yes" %}**

**<script>**

**alert('Something went wrong , Try Again');**

**</script>**

**{% endifequal %}**

**{% endblock %}**

**ADD CATEGORY PAGE CODING**

**{% load static %}**

**{% block body %}**

**<html lang="en" class="no-focus"> <!--<![endif]-->**

**<head>**

**<title>News Portal - Add Category</title>**

**<link rel="stylesheet" id="css-main" href="{% static 'adminassets/css/codebase.min.css' %}">**

**</head>**

**<body>**

**<div id="page-container" class="sidebar-o sidebar-inverse side-scroll page-header-fixed main-content-narrow">**

**{% include 'sidebar.html' %}**

**{% include 'admin\_nav.html' %}**

**<!-- Main Container -->**

**<main id="main-container">**

**<!-- Page Content -->**

**<div class="content">**

**<!-- Register Forms -->**

**<h2 class="content-heading">Add Category</h2>**

**<div class="row">**

**<div class="col-md-12">**

**<!-- Bootstrap Register -->**

**<div class="block block-themed">**

**<div class="block-header bg-gd-emerald">**

**<h3 class="block-title">Add Category</h3>**

**<div class="block-options">**

**<button type="button" class="btn-block-option" data-toggle="block-option" data-action="state\_toggle" data-action-mode="demo">**

**<i class="si si-refresh"></i>**

**</button>**

**<button type="button" class="btn-block-option" data-toggle="block-option" data-action="content\_toggle"></button>**

**</div>**

**</div>**

**<div class="block-content">**

**<form method="post">**

**{% csrf\_token %}**

**<div class="form-group row">**

**<label class="col-12" >Category Name:</label>**

**<div class="col-12">**

**<input type="text" class="form-control" name="catname" value="" required='true'>**

**</div>**

**</div>**

**<div class="form-group row">**

**<div class="col-12">**

**<button type="submit" class="btn btn-alt-success" name="submit">**

**<i class="fa fa-plus mr-5"></i> Add**

**</button>**

**</div>**

**</div>**

**</form>**

**</div>**

**</div>**

**<!-- END Bootstrap Register -->**

**</div>**

**</div>**

**</div>**

**<!-- END Page Content -->**

**</main>**

**<!-- END Main Container -->**

**{% include 'adminfooter.html' %}**

**</div>**

**<!-- END Page Container -->**

**<!-- Codebase Core JS -->**

**<script src="{% static 'adminassets/js/core/jquery.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/core/popper.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/core/bootstrap.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/core/jquery.slimscroll.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/core/jquery.scrollLock.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/core/jquery.appear.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/core/jquery.countTo.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/core/js.cookie.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/codebase.js' %}"></script>**

**</body>**

**</html>**

**{% ifequal error "no" %}**

**<script>**

**alert('Record Saved Successfully');**

**window.location=('{% url 'view\_category' %}');**

**</script>**

**{% endifequal %}**

**{% ifequal error "yes" %}**

**<script>**

**alert('Something went wrong , Try Again');**

**</script>**

**{% endifequal %}**

**{% endblock %}**

**MANAGE CATEGORY PAGE CODING**

**{% load static %}**

**{% block body %}**

**<html lang="en" class="no-focus"> <!--<![endif]-->**

**<head>**

**<title>News Portal - Manage Category</title>**

**<link rel="stylesheet" href="{% static 'adminassets/js/plugins/datatables/dataTables.bootstrap4.min.css' %}">**

**<link rel="stylesheet" id="css-main" href="{% static 'adminassets/css/codebase.min.css' %}">**

**</head>**

**<body>**

**<div id="page-container" class="sidebar-o sidebar-inverse side-scroll page-header-fixed main-content-narrow">**

**{% include 'sidebar.html' %}**

**{% include 'admin\_nav.html' %}**

**<!-- Main Container -->**

**<main id="main-container">**

**<!-- Page Content -->**

**<div class="content">**

**<h2 class="content-heading">Manage Category</h2>**

**<!-- Dynamic Table Full Pagination -->**

**<div class="block">**

**<div class="block-header block-header-default">**

**<h3 class="block-title">Manage Category</h3>**

**</div>**

**<div class="block-content block-content-full">**

**<!-- DataTables init on table by adding .js-dataTable-full-pagination class, functionality initialized in js/pages/be\_tables\_datatables.js -->**

**<table class="table table-bordered table-striped table-vcenter js-dataTable-full-pagination">**

**<thead>**

**<tr>**

**<th class="text-center"></th>**

**<th>Category Name</th>**

**<th class="d-none d-sm-table-cell" style="width: 15%;">Action</th>**

**</tr>**

**</thead>**

**<tbody>**

**{% for i in category %}**

**<tr>**

**<td class="text-center">{{forloop.counter}}</td>**

**<td class="font-w600">{{i.categoryname}}</td>**

**<td class="d-none d-sm-table-cell"><a href="{% url 'delete\_category' i.id %}" onclick="return confirm('Do you really want to Delete ?');"><i class="fa fa-trash fa-delete" aria-hidden="true"></i></a></td>**

**</tr>**

**{% endfor %}**

**</tbody>**

**</table>**

**</div>**

**</div>**

**<!-- END Dynamic Table Full Pagination -->**

**<!-- END Dynamic Table Simple -->**

**</div>**

**<!-- END Page Content -->**

**</main>**

**<!-- END Main Container -->**

**{% include 'adminfooter.html' %}**

**</div>**

**<!-- END Page Container -->**

**<!-- Codebase Core JS -->**

**<script src="{% static 'adminassets/js/core/jquery.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/core/popper.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/core/bootstrap.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/core/jquery.slimscroll.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/core/jquery.scrollLock.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/core/jquery.appear.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/core/jquery.countTo.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/core/js.cookie.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/codebase.js' %}"></script>**

**<!-- Page JS Plugins -->**

**<script src="{% static 'adminassets/js/plugins/datatables/jquery.dataTables.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/plugins/datatables/dataTables.bootstrap4.min.js' %}"></script>**

**<!-- Page JS Code -->**

**<script src="{% static 'adminassets/js/pages/be\_tables\_datatables.js' %}"></script>**

**</body>**

**</html>**

**{% endblock %}**

**ADD POST PAGE CODING**

**{% load static %}**

**{% block body %}**

**<html lang="en" class="no-focus"> <!--<![endif]-->**

**<head>**

**<title>News Portal - Add Post</title>**

**<link rel="stylesheet" id="css-main" href="{% static 'adminassets/css/codebase.min.css' %}">**

**<script src="https://cdn.ckeditor.com/4.14.1/standard/ckeditor.js"></script>**

**</head>**

**<body>**

**<div id="page-container" class="sidebar-o sidebar-inverse side-scroll page-header-fixed main-content-narrow">**

**{% include 'sidebar.html' %}**

**{% include 'admin\_nav.html' %}**

**<!-- Main Container -->**

**<main id="main-container">**

**<!-- Page Content -->**

**<div class="content">**

**<!-- Register Forms -->**

**<h2 class="content-heading">Add Post</h2>**

**<div class="row">**

**<div class="col-md-12">**

**<!-- Bootstrap Register -->**

**<div class="block block-themed">**

**<div class="block-header bg-gd-emerald">**

**<h3 class="block-title">Add Post</h3>**

**<div class="block-options">**

**<button type="button" class="btn-block-option" data-toggle="block-option" data-action="state\_toggle" data-action-mode="demo">**

**<i class="si si-refresh"></i>**

**</button>**

**<button type="button" class="btn-block-option" data-toggle="block-option" data-action="content\_toggle"></button>**

**</div>**

**</div>**

**<div class="block-content">**

**<form method="post" enctype="multipart/form-data">**

**{% csrf\_token %}**

**<div class="form-group row">**

**<label class="col-12" >Post Title:</label>**

**<div class="col-12">**

**<input type="text" class="form-control" name="posttitle" value="" required='true'>**

**</div>**

**</div>**

**<div class="form-group row">**

**<label class="col-12" >Category:</label>**

**<div class="col-12">**

**<select type="text" class="form-control" name="categoryname" required="true" >**

**<option value="">--Choose Category--</option>**

**{% for i in category %}**

**<option value="{{i.categoryname}}">{{i.categoryname}}</option>**

**{% endfor %}**

**</select>**

**</div>**

**</div>**

**<div class="form-group row">**

**<label class="col-12" >Post Details:</label>**

**<div class="col-12">**

**<textarea class="form-control" name="postdetail" required='true'></textarea>**

**</div>**

**</div>**

**<script>**

**CKEDITOR.replace('postdetail');**

**</script>**

**<div class="form-group row">**

**<label class="col-12" >Feature Image:</label>**

**<div class="col-12">**

**<input type="file" class="form-control" name="postimage" value="" required='true'>**

**</div>**

**</div>**

**<div class="form-group row">**

**<div class="col-12">**

**<button type="submit" class="btn btn-alt-success" name="submit">**

**<i class="fa fa-plus mr-5"></i> Add**

**</button>**

**</div>**

**</div>**

**</form>**

**</div>**

**</div>**

**<!-- END Bootstrap Register -->**

**</div>**

**</div>**

**</div>**

**<!-- END Page Content -->**

**</main>**

**<!-- END Main Container -->**

**{% include 'adminfooter.html' %}**

**</div>**

**<!-- END Page Container -->**

**<!-- Codebase Core JS -->**

**<script src="{% static 'adminassets/js/core/jquery.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/core/popper.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/core/bootstrap.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/core/jquery.slimscroll.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/core/jquery.scrollLock.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/core/jquery.appear.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/core/jquery.countTo.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/core/js.cookie.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/codebase.js' %}"></script>**

**</body>**

**</html>**

**{% ifequal error "no" %}**

**<script>**

**alert('Record Saved Successfully');**

**window.location=('{% url 'view\_post' %}');**

**</script>**

**{% endifequal %}**

**{% ifequal error "yes" %}**

**<script>**

**alert('Something went wrong , Try Again');**

**</script>**

**{% endifequal %}**

**{% endblock %}**

**MANAGE POST PAGE CODING**

**{% load static %}**

**{% block body %}**

**<html lang="en" class="no-focus"> <!--<![endif]-->**

**<head>**

**<title>News Portal - View Post</title>**

**<link rel="stylesheet" href="{% static 'adminassets/js/plugins/datatables/dataTables.bootstrap4.min.css' %}">**

**<link rel="stylesheet" id="css-main" href="{% static 'adminassets/css/codebase.min.css' %}">**

**</head>**

**<body>**

**<div id="page-container" class="sidebar-o sidebar-inverse side-scroll page-header-fixed main-content-narrow">**

**{% include 'sidebar.html' %}**

**{% include 'admin\_nav.html' %}**

**<!-- Main Container -->**

**<main id="main-container">**

**<!-- Page Content -->**

**<div class="content">**

**<h2 class="content-heading">Manage Post</h2>**

**<!-- Dynamic Table Full Pagination -->**

**15%;">Action</th>**

**</tr>**

**</thead>**

**<tbody>**

**{% for i in newspost %}**

**<tr>**

**<td class="text-center">{{forloop.counter}}</td>**

**<td class="font-w600">{{i.posttitle}}</td>**

**<td class="font-w600">{{i.category.categoryname}}</td>**

**<td class="d-none d-sm-table-cell">**

**<a href="{% url 'post\_detail' i.id %}"><i class="fa fa-eye" aria-hidden="true"></i></a>**

**<a href="{% url 'delete\_post' i.id %}" onclick="return confirm('Do you really want to Delete ?');"><i class="fa fa-trash fa-delete" aria-hidden="true"></i></a></td>**

**</tr>**

**{% endfor %}**

**</tbody>**

**</table>**

**</div>**

**</div>**

**<!-- END Dynamic Table Full Pagination -->**

**<!-- END Dynamic Table Simple -->**

**</div>**

**<!-- END Page Content -->**

**</main>**

**<!-- END Main Container -->**

**{% include 'adminfooter.html' %}**

**</div>**

**<!-- END Page Container -->**

**<!-- Codebase Core JS -->**

**<script src="{% static 'adminassets/js/core/jquery.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/core/popper.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/core/bootstrap.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/core/jquery.slimscroll.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/core/jquery.scrollLock.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/core/jquery.appear.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/core/jquery.countTo.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/core/js.cookie.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/codebase.js' %}"></script>**

**<!-- Page JS Plugins -->**

**<script src="{% static 'adminassets/js/plugins/datatables/jquery.dataTables.min.js' %}"></script>**

**<script src="{% static 'adminassets/js/plugins/datatables/dataTables.bootstrap4.min.js' %}"></script>**

**<!-- Page JS Code -->**

**<script src="{% static 'adminassets/js/pages/be\_tables\_datatables.js' %}"></script>**

**</body>**

**</html>**

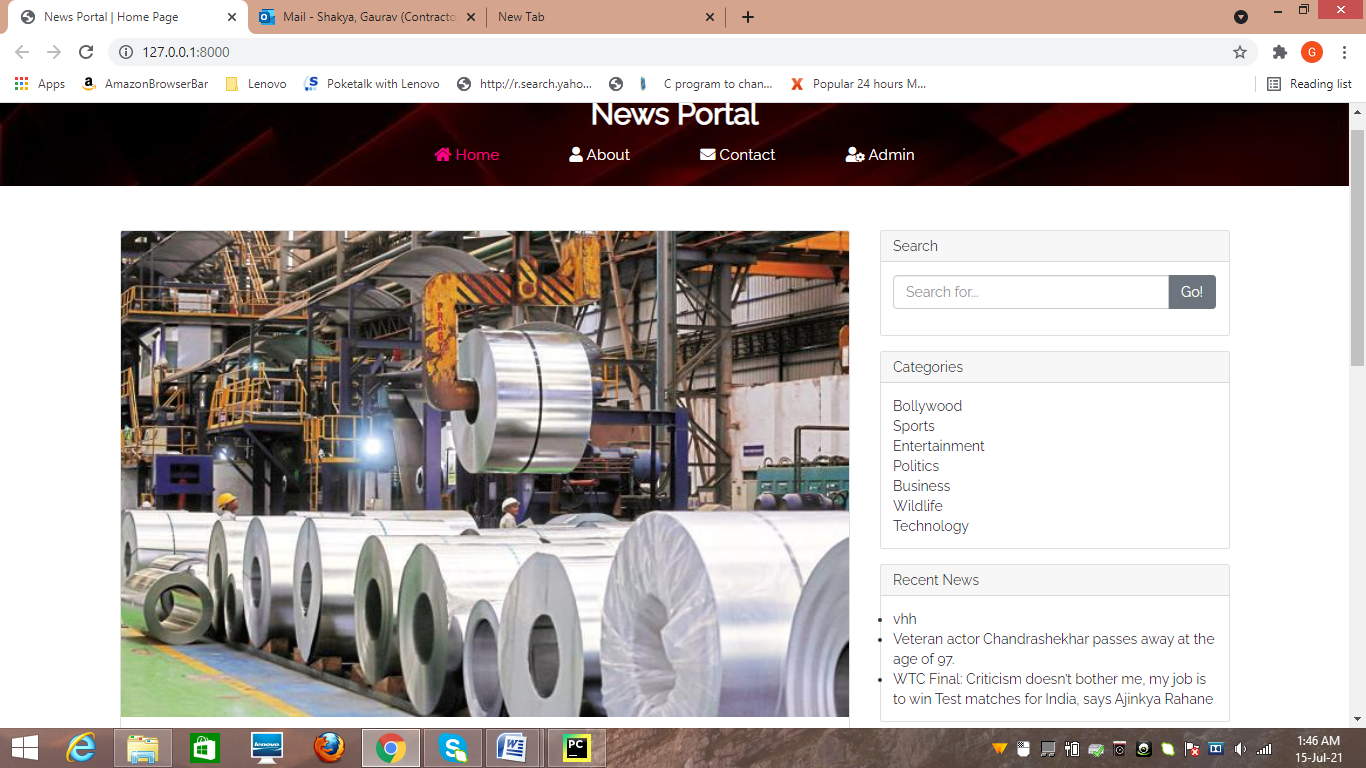
**{% endblock %}**

**CHAPTER 6**

**SNAPSHOTS**

**Home Page**

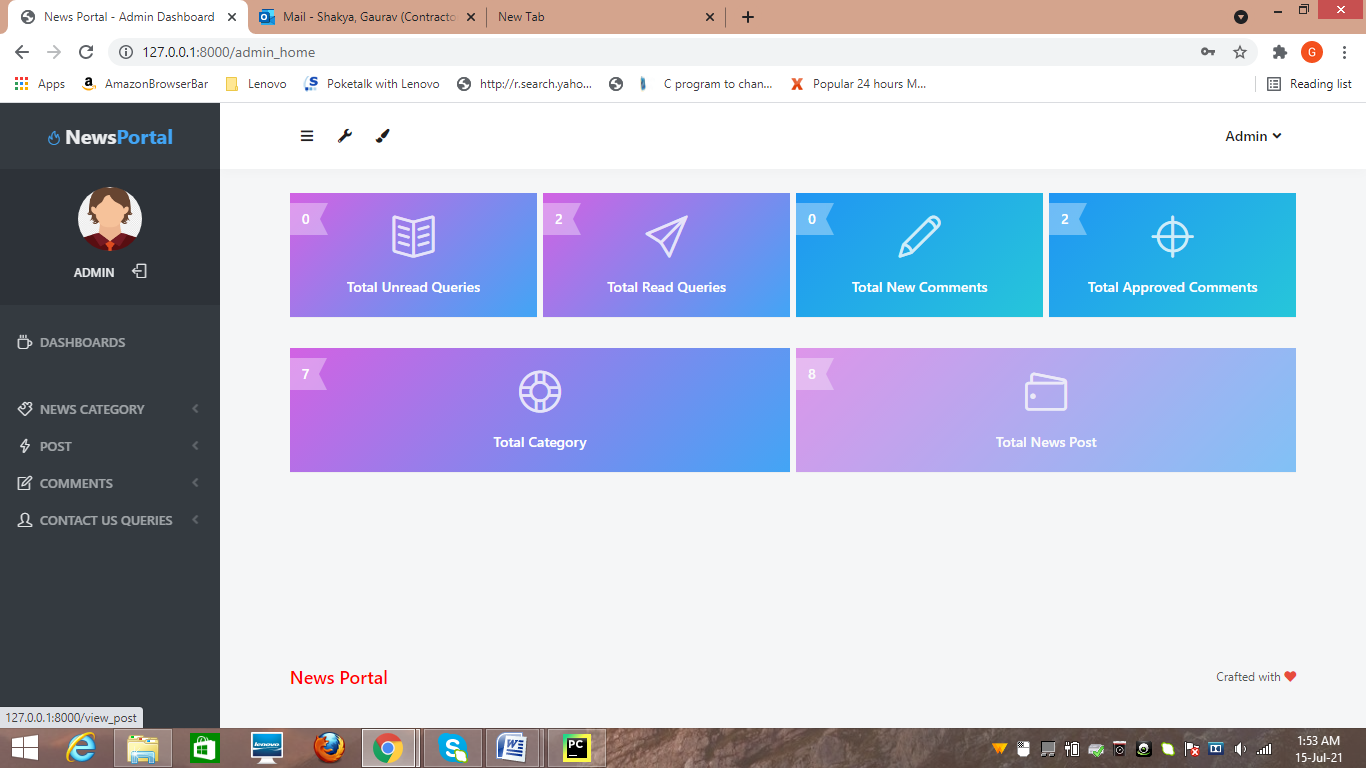
On this home page, latest news will be displayed. All the categories and recent news will also be displayed.



**Figure 6-Home Page**

**Admin Dashboard Page**

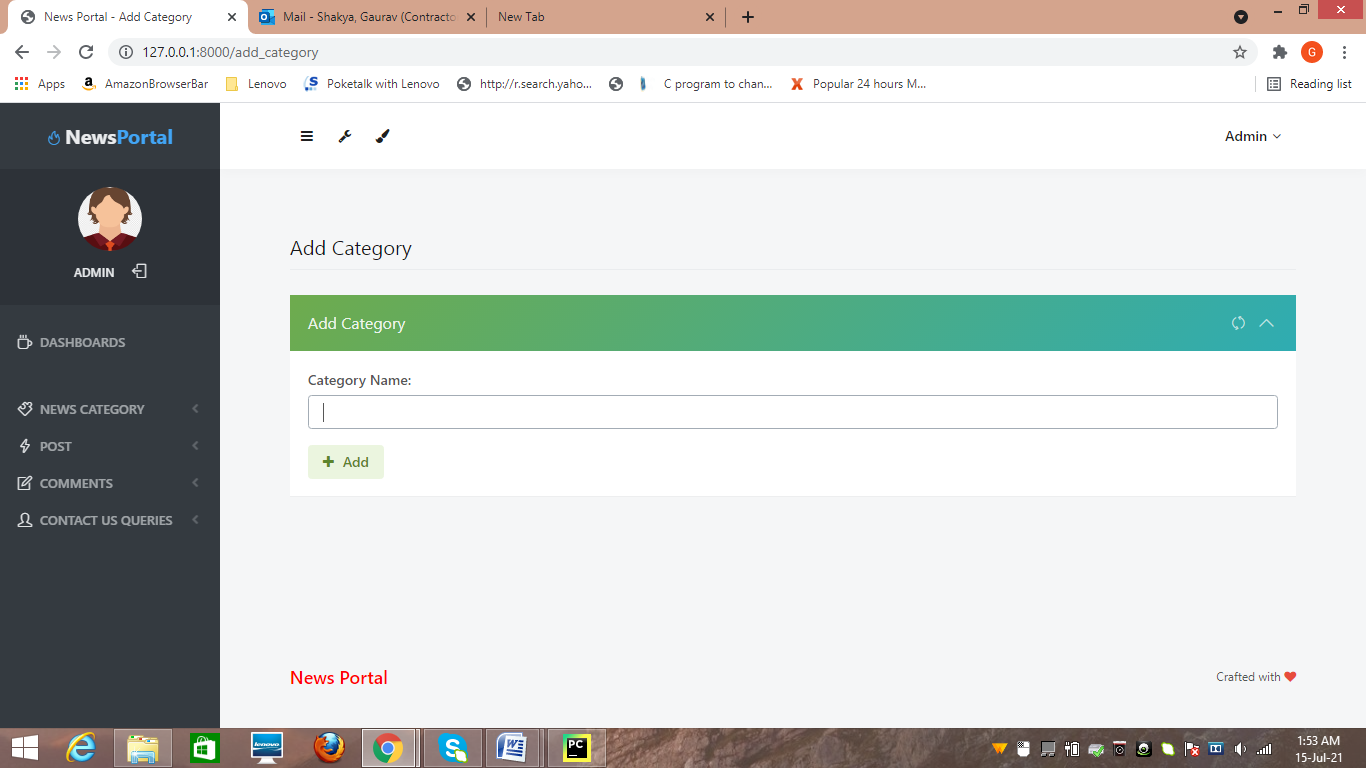
This section admin can add/update/delete the category/post/comments.. Admin can also restore deleted category.



**Figure 7-Admin Dashboard**

**Add News Category Page**

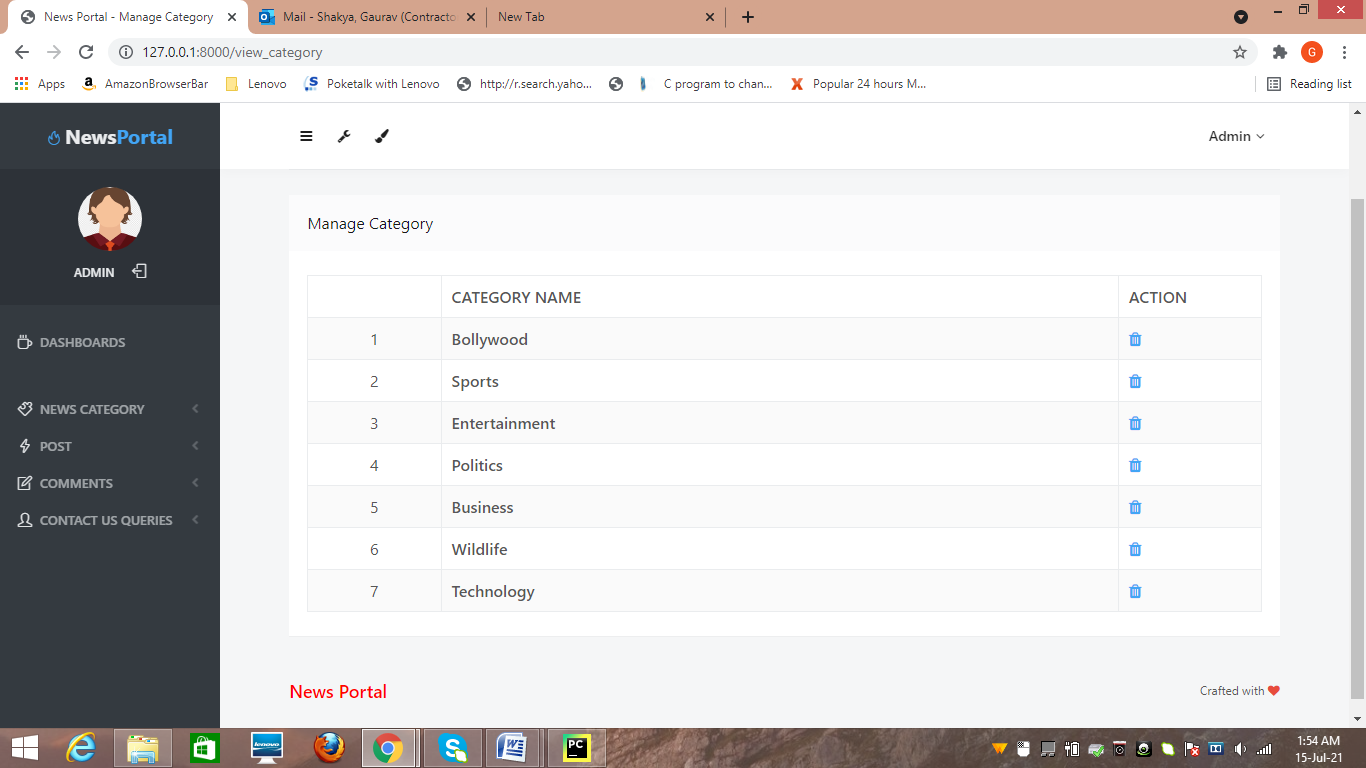
In this section admin can add/update/delete the category. Admin can also restore deleted category.



**Figure 8-News Category**

**Manage Category Page**

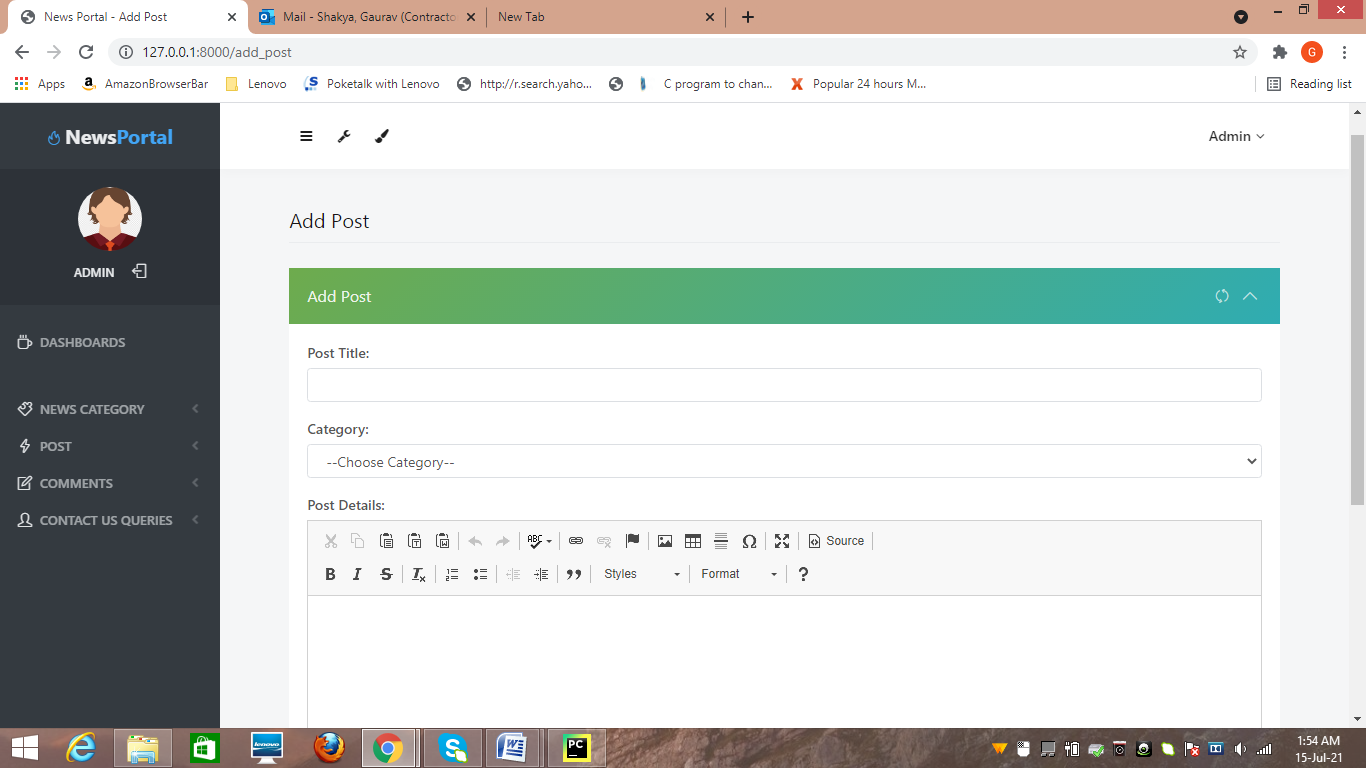
In this section admin can add/update/delete the Subcategory. Admin can also restore deleted Subcategory **.**



**Figure 9-Manage Category Page**

**Add Post Page**

Admin can add /update / delete news posts. admin can also view deleted news post in trash post section and restore deleted posts.

****

**Figure 10-Add post**

**CHAPTER 7**

**TESTING**

**7.1 INTRODUCTION**

Testing is the integral part of any System Development Life Cycle insufficient and interested application tends to crash and result in loss of economic and manpower investment besides user’s dissatisfaction and downfall of reputation.

“Software Testing can be looked upon as one among much process, an organization performs, and that provides the last opportunity to correct any flaws in the developed system. Software Testing includes selecting test data that have more probability of giving errors.” The first step in System testing is to develop the plan that all aspect of system .Complements, Correctness, Reliability and Maintainability.

Software is to be tested for the best quality assurance, an assurance that system meets the specification and requirement for its intended use and performance.

System Testing is the most useful practical process of executing the program with the implicit intention of finding errors that makes the program fail.

**7.2 Types of Testing**

**Black Box (Functional) Testing:**

Testing against specification of system or components. Study it by examining its inputs and related outputs. Key is to devise inputs that have a higher likelihood of causing outputs that reveal the presence of defects. Use experience and knowledge of domain to identify such test cases. Failing this a systematic approach may be necessary. Equivalence partitioning is where the input to a program falls into a number of classes, e.g. positive numbers vs. negative numbers. Programs normally behave the same way for each member of a class. Partitions exist for both input and output. Partitions may be discrete or overlap. Invalid data (i.e. outside the normal partitions) is one or more partitions that should be tested.

Internal System design is not considered in this type of testing. Tests are based on requirements and functionality.

This type of test case design method focuses on the functional requirements of the software, ignoring the control structure of the program. Black box testing attempts to find errors in the following categories:

* Incorrect or missing functions.
* Interface errors.
* Errors in data structures or external database access.
* Performance errors.
* Initialization and termination errors.

**White Box (Structural) Testing:**

Testing based on knowledge of structure of component (e.g. by looking at source code). Advantage is that structure of code can be used to find out how many test case need to be performed. Knowledge of the algorithm (examination of the code) can be used to identify the equivalence partitions. Path testing is where the tester aims to exercise every independent execution path through the component. All conditional statements tested for both true and false cases. If a unit has no control statements, there will be up to 2n possible paths through it. This demonstrates that it is much easier to test small program units than large ones. Flow graphs are a pictorial representation of the paths of control through a program (ignoring assignments, procedure calls and I/O statements). Use flow graph to design test cases that execute each path. Static tools may be used to make this easier in programs that have a complex branching structure. Tools support. Dynamic program analyzers instrument a program with additional code. Typically this will count how many times each statement is executed. At end print out report showing which statements have and have not been executed. Problems with flow graph derived testing:

# Data complexity could not take into account.

# We cannot test all paths in combination.

# In really only possible at unit and module testing stages because beyond that complexity is too high.

This testing is based on knowledge of the internal logic of an application’s code. Also known as a Glass Box Testing .Internal software and code working should be known for this type of testing. Tests are based on coverage of code statements, branches, paths, conditions.

**Unit Testing:**

Unit testing concentrates on each unit of the software as implemented in the code. This is done to check syntax and logical errors in programs. At this stage, the test focuses on each module individually, assuring that it functions properly as a unit. In our case, we used extensive white-box testing at the unit testing stage.

A developer and his team typically do the unit testing do the unit testing is done in parallel with coding; it includes testing each function and procedure.

**Incremental Integration Testing:**

Bottom up approach for testing i.e. continuous testing of an application as new functionality is added; Application functionality and modules should be independent enough to test separately done by programmers or by testers.

**Integration Testing:**

Testing of integration modules to verify combined functionality after integration .Modules are typically code modules, individual applications, client and server and distributed systems.

**Functional Testing:**

This type of testing ignores the internal parts and focus on the output is as per requirement or not .Black box type testing geared to functionality requirements of an application.

**System Testing:**

Entire system is tested as per the requirements. Black box type test that is based on overall requirement specifications covers all combined parts of a system.

**CHAPTER 8**

**CONCLUSION &FUTURE SCOPE**

**8.1 CONCLUSION-**

The project entitled “Online News Portal” is developed using HTML, CSS and Bootstrap as front end and Python, Sqlite database in back end to computerize the process of online management of news post. This project covers only the basic features required

**8.2 FUTURE SCOPE-**

Thisweb application involves almost all the features of the online news posting. The future implementation will be online help for the users and chatting with website administrator.

**CHAPTER 9**

**REFERENCES**

1. Trost and A. Žemva, "Configurable hardware components generator in Python," 2015 4th Mediterranean Conference on Embedded Computing (MECO), 2015, pp. 96-99, doi: 10.1109/MECO.2015.7181876.
2. A. Kumar and S. P. Panda, "A Survey: How Python Pitches in IT-World," 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon), 2019, pp. 248-251, doi: 10.1109/COMITCon.2019.8862251
3. Y. Liu, "JSOptimizer: An Extensible Framework for JavaScript Program Optimization," 2019 IEEE/ACM 41st International Conference on Software Engineering: Companion Proceedings (ICSE-Companion), 2019, pp. 168-170, doi: 10.1109/ICSE-Companion.2019.00069.
4. H. Park, W. Jung and S. Moon, "Javascript ahead-of-time compilation for embedded web platform," 2015 13th IEEE Symposium on Embedded Systems For Real-time Multimedia (ESTIMedia), 2015, pp. 1-9, doi: 10.1109/ESTIMedia.2015.7351768.
5. S. Delcev and D. Draskovic, "Modern JavaScript frameworks: A Survey Study," 2018 Zooming Innovation in Consumer Technologies Conference (ZINC), 2018, pp. 106-109, doi: 10.1109/ZINC.2018.8448444.
6. M. Akbar, F. N. Azizah and G. A. P. Saptawati, "Integration of HTML tables in web pages," 2015 International Conference on Data and Software Engineering (ICoDSE), 2015, pp. 132-137, doi: 10.1109/ICODSE.2015.7436985.
7. S. -. Lim and Y. -. Ng, "Extracting structures of HTML documents," Proceedings Twelfth International Conference on Information Networking (ICOIN-12), 1998, pp. 420-426, doi: 10.1109/ICOIN.1998.648420
8. Y. C. Chou and H. C. Liao, "A Webpage Data Hiding Method by Using Tag and CSS Attribute Setting," 2014 Tenth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, 2014, pp. 122-125, doi: 10.1109/IIH-MSP.2014.37.
9. Vamsi Krishna Myalapalli and Bhupati Lohith Ravi Teja, "High performance PL/SQL programming," 2015 International Conference on Pervasive Computing (ICPC), 2015, pp. 1-5, doi: 10.1109/PERVASIVE.2015.7087001.
10. T. Q. Dam, S. Cheon and Y. Won, "On the IO Characteristics of the SQLite Transactions," 2016 IEEE/ACM International Conference on Mobile Software Engineering and Systems (MOBILESoft), 2016, pp. 214-224, doi: 10.1109/MobileSoft.2016.047.
11. J. Chou, L. Chen, H. Ding, J. Tu and B. Xu, "A Method of Optimizing Django Based on Greedy Strategy," 2013 10th Web Information System and Application Conference, 2013, pp. 176-179, doi: 10.1109/WISA.2013.41.
12. T. Naumovic, M. Despotovic-Zrakic, B. Radenkovic, L. Zivojinovic and I. Jezdovic, "Development of a Continuous System Simulation Engine in Python Programing Language," 2020 19th International Symposium INFOTEH-JAHORINA (INFOTEH), 2020, pp. 1-5, doi: 10.1109/INFOTEH48170.2020.9066334.
13. F. Rigueira, J. Bernardino and I. Pedrosa, "Extraction of information from log files Using Python Programming and Tableau," 2020 15th Iberian Conference on Information Systems and Technologies (CISTI), 2020, pp. 1-7, doi: 10.23919/CISTI49556.2020.9140844.
14. M. A. P. Subali and S. Rochimah, "A new model for measuring the complexity of SQL commands," 2018 10th International Conference on Information Technology and Electrical Engineering (ICITEE), 2018, pp. 1-5, doi: 10.1109/ICITEED.2018.8534782.
15. Y. Guo, N. Li, J. Offutt and A. Motro, "Automatically Repairing SQL Faults," 2018 IEEE International Conference on Software Quality, Reliability and Security (QRS), 2018, pp. 500-511, doi: 10.1109/QRS.2018.00063.