

**KARNATAK UNIVERSITY’S**

**KARNATAK SCIENCE COLLEGE,**

**DHARWAD**



**DEPARTMENT OF COMPUTER SCIENCE**

**B.C.A AND B.SC (CS)**

**PROJECTREPORT ON**

**PAID ADS WEB APPLICATION BASED ON NEWSPORTAL**

**UNDER THE GUDIENCE OF**

**SMT.KALPANA C DALAWAYI**

**SUBMITTED BY**

**MAHESH GUBBALLI (17M10016) PURUSHOTTAM PATTANASHETTY (17M10028)**

**KARNATAK UNIVERSITY’S**

**KARNATAK SCIENCE COLLEGE, DHARWAD**

**DEPARTMENT OF COMPUTER SCIENCE,**

**B.C.A /B.SC (CS)**

****

**2019-2020**

**CERTIFICATE**

This is to certify that Mr.Mahesh Gubbali and Mr. Purushottam Pattanashetty has satisfactorily completed the project work entitled “**Paid Ads Web Application Based on News Portal**” for the partial fulfillment of degree in Bachelor of Computer Science from the Karnatka University, Dharwad for the year 2019-2020.

**Project Associates Project Guide**

1. MAHESH GUBBALLI (7M10016) SMT.Kalpana.C.D

2. PURUSHOTTAM PATTANASHETTY (17M10028)

**Examiners Co-ordinator**

**1.**

**2.**

**ACKNOWLEDGEMENT**

Nothing in life is ever successful without the co-operate effort of many gifted people who are willing to network and submit their talent experience and passion for a common goal. This work is a product of countless individuals whose thoughts, ideas, perspectives and work have us the exposure to knowledge.

The satisfaction and happiness we feel at the successful completion of our project entitled “**Paid Ads Web Application Based on News Portal**” would be incomplete if did not remember the people who made it possible and crowned our efforts with success. First and foremost we thank our parents and almighty for their blessings on us at all times in all circumstances.

We are grateful to our beloved principal **Dr.Giriyappanavar** and Co-ordinator **Smt.Kalpana.C.D** Department of Computer Science for your kind support.

We express our deep sense of indebtedness to our guide **Smt.Kalpana.C.D** for her dynamic guidance and encouragement.

Finally, we thank our Department Faculties Shri. Rajashekhar.V.B, **Dr.Nagaraj.S,Smt.Lalitha.R.Naik** and **Smt.Ambika.R** for their kind support.

**ABSTRACT**

The Project titled “Paid Ads Web Application Based on News Portal” is an application designed to monitor and control the information related to news media and advertisement. The application is developed in PHP Language and MYSQL and database.

The Application is intended to deliver the news of different categories such as (Breaking News, Hubli-Dharwad, Crime, and Sports etc) and also paid advertisement based on different section of the application, the advertisement are categorized into three types namely IMAGE, MARQUEE and VIDEO

The Application is completely platform independent and can be accessed on any devices. The application has features like, adding news on regular basis with youtube video integration, advertisement management, generation of invoice and order reports.

In terms of economical feasibility apart from sms and server charges, the application does not incur any additional charges. The Application is portable which can be accessed on any platform without depending on the operating system. The Software is tested at different levels and made sure data integrity is maintained. In terms of Performance of the application process and retrieve the data in less than second, which makes the application use friendly and interactive

The Security and Robustness of the application is tested at each phase of the modules integration to ensure no redundant data is recorded and the application can be accessed through authorization.

**TABLE OF CONTENTS**

1. INTRODUCTION 1-2

1.1 PROBLEM DEFINATION 1

1.2 PROJECT OVERVIEW 1

1.3 HARDWARE SPECIFICATION………………………………………………………..

1.4 SOFTWARE SPECIFICATIONS 2

2. LITERATURE SURVEY 3-6

2.1 EXISTING SYSTEM 3

2.2 PROPOSED SYSTEM 3

2.3 FEASIBILIITY STUDY 3

3. SYSTEM ANALYSIS AND DESIGN 8-12

3.1 REQUIREMENT SPECIFICATION 3

3.2 DATA FLOW DIAGRAM 3

3.3 ER DIAGRAM 3

3.4 DATABASE DESIGN 3

4. IMPLEMENTATION 22-31

4.1 LANGUAGES USED3

4.1.1 HTML & CSS 4

4.1.2 PHP6

4.2 SOFTWARE USED 3

4.2.1 XAMPP6

4.2.2 BRACKETS4

4.3 CODE 3

5. TESTING 32-37

5.1 DETAILS OF TESTING METHODOLOGIES 3

5.2 TEST CASES 3

**6. RESULTS / SCREEN SHOTS 38-53**

7. CONCLUSION / FUTURE SCOPE 54

**8. REFERENCES 57**

**9. APPENDICES**

1. **INTRODUCTION**

The Information Technology has grown rapidly since last decade and is one of the most powerful mechanisms to provide and transfer information throughout the world using World Wide Web. Since the technology has growing day by day, it is challenging for the industries, companies and organizations to adopt and enhance their services and reach to the customers. Media Companies since earlier decade has been publishing the news and advertising information through paper publications, which result in lot of manual work and machine power. Over the time, it is observed that people tend to access the information in smart phones or devices rather than news papers or magazines.

The Software has been developed to override the problems prevailing in the practicing manual system. The software is supported to eliminate and in some cases reduce the hardship faced by the existing system. Moreover the system is designed for the particular need of the company to carry out the operations in smooth and effective manner.

The Application has been designed in accordance with Portability, Security and Robustness, the application ensures it can be accessed on all the devices without depending on any platform. The application allows only authorized users to maintain the news portal, where they add,update and modify the data.

The Application has Rich User Interface, which allows the user to easily understand and carry out the activities. The Bootstrap Framework technique is used to ensure the application is completely responsive and accessed on all devices such as smartphones, tabs and laptops.

**1.1 Problem Definition**

The Existing System is completely offline, making it complicated to maintain the advertisement and news information. The application should have features like maintaining the information at one centralized system, differentiate the news with its categories such as Crime, State, and Breaking News, Provide the platform to advertise customer business/products based on priority, Provide the platform to upload the videos on Youtube & Integrate in the application and can be accessed on any device or at any location.

**1.2 Project Overview**

**“Paid ads web application based on news portal”** has been developed mainly for the cause of advertisement & also at the same time we can see news also. In present system there is facility of advertising the business products through online. For which we are providing platform for enterprisers to advertise about their products and advertise about their shop too.

In current system there is lot of process to approach of online marketing using that environment the entrepreneurs are also interested to advertise about their business and their products.

There are mainly three types of advertisement namely,

1. Text type

2. Image type

3. Video type (audio and video).

In online advertisement there is large number of users are present. It is easy to recognise their product or business. In other way news is also important to read currently what happening around us. In news there are categories of news are present like state news, breaking news, entertainment, political news, etc.., these all are in same page by viewing news the users can view about advertisement. However, we can see news related videos too. Based on priority of customer like top header, right side, right bottom, bottom, and middle part of web page we are pricing for advertisement.

**1.3 Hardware Specification**

|  |  |
| --- | --- |
| Hardware component | Description |
| Memory | 2GB RAM and above |
| Hard Disk | 100GB and above |
| Processor | Core i3 and above |

**1.4 Software Specification**

|  |  |
| --- | --- |
| Software component | Description |
| Operating System | Windows 7 and above |
| Text Editor | Brackets |
| Scripting Languages | Java Script & JQuery |
| Language | PHP |
| Database | MYSQL |
| Server | Apache Tomcat Server |

1. **LITERATURE SURVEY**

The Web Application is designed for News Portal and is intended to deliver day to day news to the users & provide the right information regarding State, Crime, Breaking, Sports News etc. The Application also provides an advertising platform for the customers, who can advertise their business/products based on the priority. The Application allows the organization to maintain the news & advertisement at one centralized system, which can be accessed anywhere anytime.

**2.1 Existing System**

* In the present system the news are posted in news paper & and are maintained by other service providers.
* There is no platform to advertise the ads of the customers.
* The Organization manager should collect all the news at one time and visit the media to publish with lots of paper work.

**2.2 Proposed System**

* The Application provides an interface to maintain and add the news regularly.
* The Application can add or modify the categories as when required.
* The customer ads can be advertised based on the priority & sections.
* Reporters, customer details & News Details will be maintained & accessed anywhere anytime.
* The application can be accessed on any device with the help of internet.
* The application is secured & has access to known users only.

**2.3 Feasibility Study**

A feasibility analysis usually involves a thorough assessment of the operational (need), financial and technical aspects of a proposal. Feasibility study is the test of the system proposal made to identify whether the user needs may be satisfied using the current software and hardware technologies, whether the system will be cost effective from a business point of view and whether it can be developed with the given budgetary constraints. A feasibility study should be relatively cheap and done at the earliest possible time. Depending on the study, the decision is made whether to go ahead with a more detailed analysis.

When a new project is proposed, it normally goes through feasibility assessment. Feasibility study is carried out to determine whether the proposed system is possible to develop with available resources and what should be the cost consideration. Facts considered in the feasibility analysis were

* Technical Feasibility
* Economic Feasibility
* Behavioral Feasibility

**2.3.1 Technical Feasibility**

Technical feasibility includes whether the technology is available in the market for development and its availability. The assessment of technical feasibility must be based on an outline design of system requirements in terms of input, output, files, programs and procedures. This can be qualified in terms of volumes of data, trends, frequency of updating, cycles of activity etc, in order to give an introduction of technical system. Considering our project it is technically feasible.

**2.3.2 Economic Feasibility**

This feasibility study present tangible and intangible benefits from the project by comparing the development and operational cost. The technique of cost benefit analysis is often used as a basis for assessing economic feasibility. This system needs some more initial investment than the existing system, but it can be justifiable that it will improve quality of service.

Thus feasibility study should center along the following points:

* Improvement resulting over the existing method in terms of accuracy, timeliness.
* Cost comparison
* Estimate on the life expectancy of the hardware.
* Overall objective.

Our project is economically feasible. It does not require much cost to be involved in the overall process. The overall objective is in easing out the recruitment processes.

**2.3.3 Behavioral / Operational Feasibility**

This analysis involves how it will work when it is installed and the assessment of political and managerial environment in which it is implemented. People are inherently resistant to change and computers have been known to facilitate change.

1. **SYSTEM ANALYSIS AND DESIGN**

**3.1 Functional Requirements**

* The Application consists of an Administrator. The administrator has **Unique Id, Name, Contact, Address, Email, Username** and **Password**. The administrator Username and Password is used to access the application and manage the orders.
* The Application has news categories classified as,
  + - **Crime.**
    - **Sports.**
    - **Hubli-Dharwad.**
    - **State.**
    - **Breaking-News.**
    - **Political.**
    - **National.**
    - **International etc.**

There is a provision to **Add**, **Modify** and **Delete** the news categories as when required.

* The News details are maintained with the provision to **Add**, **Modify** & **Delete** the records as when required. The News Data has **Unique id, Name, News Category, Image, Description, Video URL** and **Date o**f the record Added or Modified.
* The Youtube Video URL is added into the news record to display the video uploaded on youtube.
* The customer information is maintained such as, **Customer Unique Id**, **Name**, **Contact**, **Address**, **Email**, **Username**, **Password** and **Date** of Registration.
* The Advertisement details are maintained with provision to **Add, Modify & Delete** the records as when required. The Advertisement has priority such as **TOP, MIDDLE, MIDDLE RIGHT** etc so that the customer can choose the position to view the advertisement
* The Customer will be sent a Notification regarding advertisement details.
* When the customer selects the position of the advertisement to be displayed, the respective **position price** is displayed.
* The advertisement details are maintained such as **Unique id**, **Advertisement image**, **Position of Image, Price, No of days, Total Amount & Date** of the Advertisement added.
* The Advertisement Reports are maintained and total revenue earned from advertisement
* The Reports are generated for particular month/year.

**Non-Functional Requirements**

Requirements, which are not related to functional aspect of software, fall into this category. They are implicit or expected characteristics of software, which users make assumption of.

**Performance Requirements**

* Performance: The application takes less than one second to validate the information or execute the query.
* Reliability: Since the validation is done on each stage the information is processed in a correct and accurate form.
* Safety: The backup of database is always maintained in a local server. & information is securely transferred using https(Hyper-Text Transfer Protocol)

**Software Quality Attributes**

* Security: An OTP is sent to registered mobile number in case the user reset the password.
* Maintainability: Since each model has its own requirements. Therefore changes made on the module will not affect the other.
* Portability: The Application can be accessed on any device, without depending on specific platform

**Business Attributes**

* **Cost**

In the development of the system the cost attribute requires the factors like hardware and software resources, specific amount of man power. As Windows is open source so it’s not much cost effective.

* **Quality**

This software is properly normalized to avoid the redundancy and Ambiguity and hence, all the irrelevant constraints are removed.

**Development Attributes**

* **Testability:** The design phase of the system development takes into account the acceptance criteria and the factors affecting the performance, to make the system easily testable. The system will give high performance in terms of speed and storage efficiency. The system will provide the accurate output especially in maintaining & updating the account information.

**3.2 Dataflow Diagram**

A data flow diagram (DFD) is a significant modeling technique for analyzing and constructing information processes. Data-flow diagram (DFD) is a graphical representation of the "flow" of data through an information system. DFDs can also be used for the visualization of data processing (structured design).On a DFD, data items flow from an external data source or an internal data store to an Internal data store or an external data sink, via an internal process. A DFD provides no information about the timing or ordering of processes, or about whether processes will operate in sequence or in parallel. It is therefore quite different from a flowchart.

This shows the flow of control through an algorithm, allowing a reader to determine what operations will be performed, in what order, and under what circumstances, but not what kinds of data will be input to and output from the system, nor where the data will come from and go to, nor where the data will be stored (all of which are shown on a DFD). Data-flow diagrams 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 report. How any system is developed can be determined through a data-flow diagram.

With a data-flow diagram, users are able to visualize how the system will operate, what the system will accomplish, and how the system will be implemented. A designer usually draws a context-level DFD showing the relationship between the entities inside and outside of a system as one single step. This basic DFD can be then disintegrated to a lower level diagram demonstrating smaller steps exhibiting details of the system that is being modeled. Numerous levels may be required to explain a complicated system. The different versions are Context Diagrams (Level 0), Partitioned Diagrams (single process only -- one level), functionally decomposed, leveled sets of Data Flow Diagrams.

### Data Flow Diagrams Symbols

A DFD usually comprises of four components. These four components can be represented by four simple symbols. These symbols can be explained in detail as follows: External entities (source/destination of data) are represented by squares; Processes (input-processing-output) are represented by rectangles with rounded corners; Data Flows (Physical or electronic data) are represented by arrows; and finally, Data Stores (physical or electronic like XML files) are represented by open-ended rectangles.

**Data store**



Or



A data store stores data passively for later access. A data store responds to requests to store and access data. It does not generate any operations. A data store allows values to be accessed in an order different from the order in which they were generated. Input flows indicate information or operations that modify the stored data such as adding or deleting elements or changing values. Output flows indicate information retrieved from the store; this information can be an entire value or a component of a value.

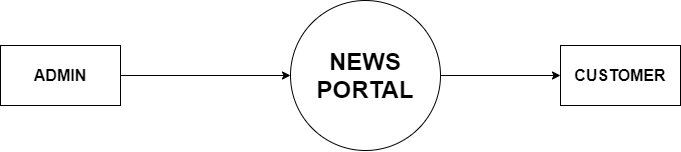
**Data flow**

A data flow moves data between processes or between processes and data stores. As such, it represents a data value at some point within a computation and an intermediate value within a computation if the flow is internal to the diagram. This value is not changed. The names of input and output flows can indicate their roles in the computation or the type of the value they move. Data names are preferably nouns. The name of a typical piece of data, the data aspect, is written alongside the arrow.

**Rules for Creating DFD**

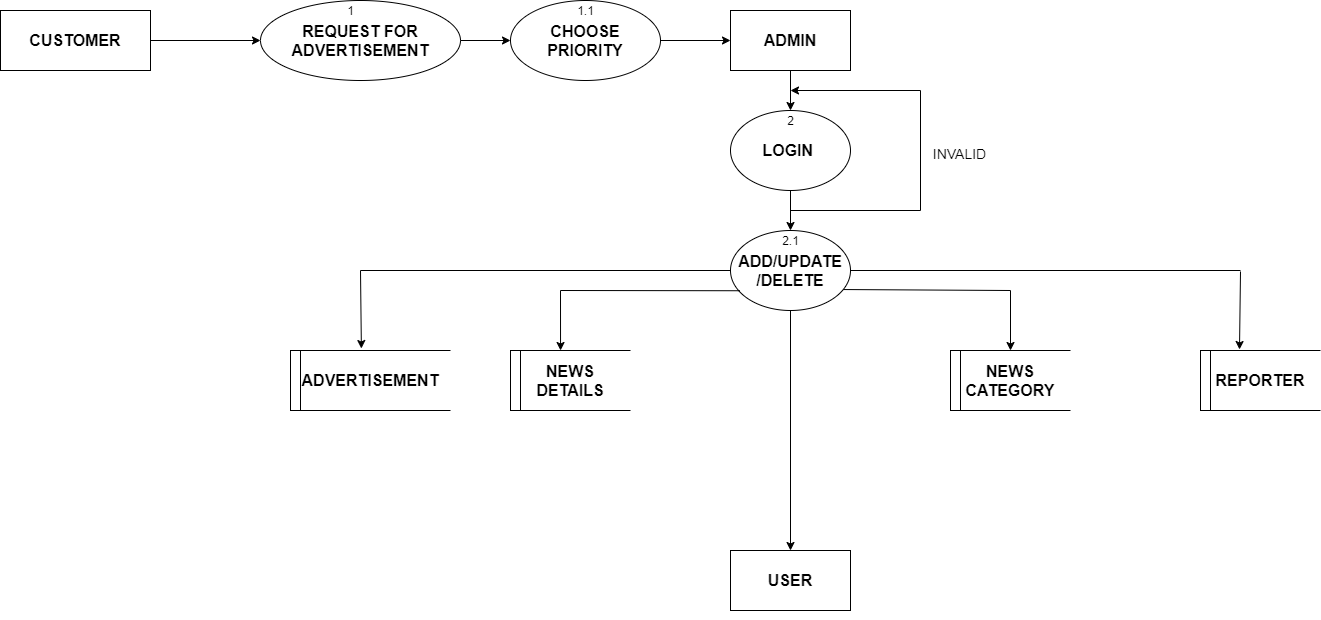
* Processes should be named and numbered for easy reference. Each name should be representative of the process.
* The direction of flow is from top to bottom and from left to right. Data traditionally flow from the source (upper left corner) to the destination although they may flow back to a source. One way to indicate this is to repeat the source symbol as a destination. Since it is used more than once in the DFD, it is marked with a short diagonal in the lower right corner.
* When a process is exploded into the lower-level details, they are numbered.
* The names of data stores, sources and destinations are written in capital letters. Process and data flow names have the first letter of each word capitalized.

**Zero Level DFD**

****

**Figure 3.2 Zero Level DFD**

**First Level DFD**

****

**Figure 3.2 First Level DFD**

**3.3 Entity Relationship Diagram**

An entity-relationship diagram is a data modeling technique that creates a graphical representation of the entities, and the relationships between entities, within an information system. An **entity-relationship model** (**ERM**) is an abstract and conceptual representation of data. Entity-relationship modeling is a database modeling method, used to produce a type of conceptual schema or semantic data model of a system, often a relational database, and its requirements in a top-down fashion. Diagrams created by this process are called **entity-relationship diagrams**, **ER diagrams**, or **ERDs**.

The first stage of information system design uses these models during the requirements analysis to describe information needs or the type of information that is to be stored in a database. The data modeling technique can be used to describe any ontology (i.e. an overview and classifications of used terms and their relationships) for a certain area of interest. In the case of the design of an information system that is based on a database, the conceptual data model is, at a later stage (usually called logical design), mapped to a logical data model, such as the relational model; this in turn is mapped to a physical model during physical design. Sometimes, both of these phases are referred to as "physical design".

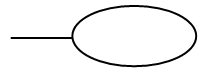
**The three main components of an E-R Diagram are:**

* The entity is a person, object, place or event for which data is collected. For example, if you consider the information system for a business, entities would include not only customers, but the customer’s address, and orders as well. The entity is represented by a rectangle and labeled with a singular noun.
* The relationship is the interaction between the entities. In the example above, the customer places an order, so the word “places” defines the relationship between that instance of a customer and the order or orders that they place. A relationship may be represented by a diamond shape, or more simply, by the line connecting the entities. In either case, verbs are used to label the relationships.
* The cardinality defines the relationship between the entities in terms of numbers. An entity may be optional: for example, a sales representative could have no customers or could have one or many customers; or mandatory: for example, there must be at least one product listed in an order. There are several different types of cardinality notations; crow’s foot notation, used here, is a common one. In crow’s foot notation, a single bar indicates one, a double bar indicates one and only one (for example, a single instance of a product can only be stored in one warehouse), a circle indicates zero, and a crow's foot indicates many. The three main cardinal relationships are: one-to-one, expressed as 1:1; one-to-many, expressed as 1: N; and many-to-many, expressed as M: N.

##### Weak Entity:



A weak entity is an entity that must defined by a foreign key relationship with another entity as it cannot be uniquely identified by its own attributes alone. **Attribute:**

****

##### Each entity has attributes or particular properties that describe the entity. Most of the data in a database consists of values of attributes. The set of all possible values of an attribute is the attribute domain. In an ER model, an attribute name appears in an oval that has a line to the corresponding entity box.

##### Key attributes:



A key attribute is the unique, distinguishing characteristic of the entity. An attribute or set of attributes that uniquely identifies a particular entity is a key. A key attribute in an ER Diagram is represented by an oval that has a line inside it and a line to the corresponding entity box. For example, an employee's social security number might be the employee's key attribute.

##### Multi-valued attribute:



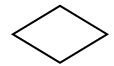
A multi-valued attribute can have more than one value. We indicate this with a double oval. For example, an employee entity can have multiple skill values.

**Derived attribute:**

****

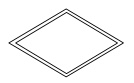
A derived attribute is based on another attribute. It is denoted by a oval and dotted line within it. For example, an employee's monthly salary is based on the employee's annual salary.

##### Relationships:



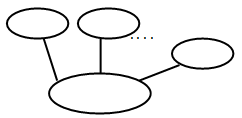
Relationships illustrate how two entities share information in the database structure. An association among entities is called a relationship. An attribute can also be a property of a relationship set. The association among the entities is described as one-to-one, one-to-many, many-to-many. A relationship is indicated by a rhombus.

##### Identifying relationship:



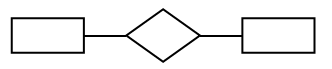
Identifying relationship is denoted by double rhombus.

##### Composite Attribute



A composite attribute has multiple components and each component is atomic or composite. We illustrate this composite nature in the ER model by branching off the component attributes.

##### Total Participation:



Total participation is represented by a double line.

##### Cardinality:

Cardinality specifies how many instances of an entity relate to one instance of another entity. Ordinarily is also closely linked to cardinality. While cardinality specifies the occurrences of a relationship, ordinarily describes the relationship as either mandatory or optional. In other words, cardinality specifies the maximum number of relationships and ordinarily specifies the absolute minimum number of relationship.

**3.3 ER Diagram**

****

**Figure 3.3 ER Diagram**

**3.4 Database Design**

**Admin**

|  |  |  |
| --- | --- | --- |
| **Column** | **Type** | **Null** |
| admin\_id *(Primary)* | int(11) | No |
| admin\_name | varchar(200) | No |
| admin\_address | varchar(500) | No |
| admin\_contact | varchar(500) | No |
| admin\_email | varchar(500) | No |
| admin\_username | varchar(500) | No |
| admin\_password | varchar(500) | No |
| admin\_date | varchar(100) | No |
| admin\_status | varchar(200) | No |

**Advertisement**

|  |  |  |
| --- | --- | --- |
| **Column** | **Type** | **Null** |
| ad\_id *(Primary)* | int(11) | No |
| cu\_id | int(11) | No |
| ap\_id | int(11) | No |
| at\_id | int(11) | No |
| ad\_name | varchar(500) | No |
| ad\_type | varchar(30) | No |
| ad\_no\_of\_days | int(11) | No |
| ad\_total\_price | int(11) | No |
| ad\_status | varchar(20) | No |
| ad\_date | date | No |
| ad\_avail\_date | date | No |

**Advertisement Pricing**

|  |  |  |
| --- | --- | --- |
| **Column** | **Type** | **Null** |
| ap\_id *(Primary)* | int(11) | No |
| at\_id | int(11) | No |
| ap\_name | varchar(50) | No |
| ap\_price | int(11) | No |
| ap\_priority | int(11) | No |

**Advertisement Type**

|  |  |  |
| --- | --- | --- |
| **Column** | **Type** | **Null** |
| at\_id *(Primary)* | int(11) | No |
| at\_type | varchar(20) | No |

**Customer**

|  |  |  |
| --- | --- | --- |
| **Column** | **Type** | **Null** |
| cu\_id *(Primary)* | int(11) | No |
| cu\_name | varchar(30) | No |
| cu\_contact | bigint(20) | No |
| cu\_email | varchar(50) | No |
| cu\_address | varchar(50) | No |
| cu\_username | varchar(50) | No |
| cu\_password | varchar(100) | No |
| cu\_date | date | No |

**News Category**

|  |  |  |
| --- | --- | --- |
| **Column** | **Type** | **Null** |
| news\_category\_id *(Primary)* | int(11) | No |
| news\_category\_name | text | No |
| news\_category\_date | date | No |
| news\_category\_status | int(11) | No |

**News Details**

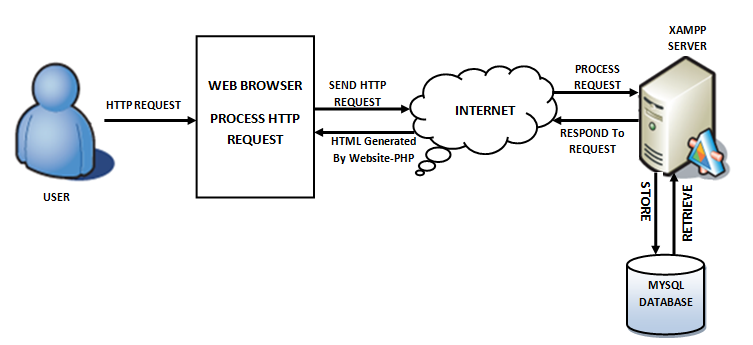
|  |  |  |
| --- | --- | --- |
| **Column** | **Type** | **Null** |
| news\_details\_id *(Primary)* | int(11) | No |
| news\_category\_id | int(11) | No |
| content\_type | varchar(100) | No |
| reporter\_id | bigint(20) | No |
| news\_details\_headlines | text | No |
| news\_details\_image | varchar(2000) | No |
| news\_details\_description | text | No |
| news\_details\_date | varchar(100) | No |
| news\_details\_video\_id | varchar(300) | No |

**Reporters**

|  |  |  |
| --- | --- | --- |
| **Column** | **Type** | **Null** |
| reporter\_id *(Primary)* | bigint(20) | No |
| fullname | varchar(100) | No |
| email | varchar(100) | No |
| contact | varchar(50) | No |
| place | varchar(50) | No |
| image | varchar(100) | No |
| regdate | varchar(50) | No |

[Open new phpMyAdmin window](http://localhost/phpmyadmin/db_datadict.php?db=news_portal&table=&server=1&target=)

1. **IMPLEMENTATION**



**FIGURE 4: XAMPP ARCHITECTURE**

**XAMPP** stands for Cross-Platform (X), Apache (A), MySQL (M), PHP (P) and Perl (P). It is a simple, lightweight Apache distribution that makes it extremely easy for developers to create a local web server for testing purposes. Everything you need to set up a web server – server application (Apache), database (MySQL), and scripting language (PHP) – is included in a simple extractable file. XAMPP is also cross-platform, which means it works equally well on Linux, Mac and Windows. Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server extremely easy

**2.1 What’s included in XAMPP?**

XAMPP has three primary components. These are:

**2.1.1. Apache:** Apache is the actual web server application that processes and delivers web content to a computer. Apache is the most popular web server online, powering nearly 54% of all websites.

**2.1.2 MySQL:** Every web application, howsoever simple or complicated, requires a database for storing collected data. MySQL, which is open source, is the world’s most popular database management system. It powers everything from hobbyist websites to professional platforms like WordPress, laraveletc

.

**2.2 Hypertext Pre-Processor (PHP):**

PHP stands for Hypertext Preprocessor created by **RasmusLerdorf**,It is a server-side scripting language that powers some of the most popular websites in the world, including WordPress and Facebook. However, PHP alone isn't enough in order to build dynamic web sites. To use PHP on a web site, we need a server that can process PHP scripts. XAMPP server allows developers to test PHP scripts locally, this makes it an invaluable piece of your local development environment.

Additionally, dynamic websites are dependent on stored information that can be added and updated easily, this is the main difference between a dynamic application and a static application. However, PHP doesn’t provide a simple, efficient way to store data. This is where a relational database management system like MySQL comes into play.

**Syntax:**

<?php

//PHP CODE

?>

PHP originally stands for “Hypertext Pre-Processor” or “Personal Home Page” and was released as a free, open source project. Over time, the language was reworked to meet the needs of its users. In 1997, PHP was renamed to the current “PHP: Hypertext Preprocessor”. PHP is generally used as a server-side scripting language; it is especially well-suited for creating dynamic web pages and client-side GUI applications. PHP generally runs on a web server, taking PHP code as its input and creating web pages as output The scripting language features integrated support for interfacing with databases such as MySQL, which makes it a prime candidate for building all manner of web applications, from simple personal web sites to complex enterprise-level applications.

**2.2.1 Usage:**

PHP primarily acts as a filter, taking input from a file or stream containing text and/or PHP instructions and outputs another stream of data; most commonly the output will be HTML. It can automatically detect the language of the user. From PHP 4, the PHP parser compiles input to produce byte code for processing by the Zend Engine, giving improved performance over its interpreter predecessor. Originally designed to create dynamic web pages, PHP’s principal focus is server side scripting, and it is similar to other server-side scripting languages that provide dynamic content from a web server to a client, such as Microsoft’s Active Server Pages, Sun Microsystems’ Java Server Pages, and mod\_perl. PHP has also attracted the development of many frameworks that provide building blocks and a design structure to promote rapid application development (RAD). Some of these include Cake PHP, Symfony, Code Igniter, and Zend Framework, offering features similar to other web application frameworks.

**2.2.2 Speed Optimization:**

As with many scripting languages, PHP scripts are normally kept as human-readable source code, even on production web servers. In this case, PHP scripts will be compiled at runtime by the PHP engine, which increases their execution time. PHP scripts are able to be compiled before runtime using PHP compilers as with other programming languages such as C (the language PHP and its extensions are written in). Code optimizers aim to reduce the computational complexity of the compiled code by reducing its size and making other changes that can reduce the execution time with the overall goal of improving performance. The nature of the PHP compiler is such that there are often opportunities for code optimization, and an example of a code optimizer is the Zend Optimizer PHP extension.

Another approach for reducing overhead for high load PHP servers is using PHP accelerators. These can offer significant performance gains by caching the compiled form of a PHP script in shared memory to avoid the overhead of parsing and compiling the code every time the script runs.

**2.3 HTML:**

HTML stands for Hyper Text Markup Language, which is the most widely used language on Web to develop web pages. HTML was created by Berners-Lee in late 1991 but "HTML 2.0" was the first standard HTML specification which was published in 1995. HTML 4.01 was a major version of HTML and it was published in late 1999. Though HTML 4.01 version is widely used but currently we are having HTML-5 version which is an extension to HTML 4.01, and this version was published in 2012.

**Syntax:**

<! DOCTYPE html>

<html>

<head>

<title>This is a Title </title>

</head>

<body>

<p>Hello World!</p>

</body>

</html>

**<! DOCTYPE>**

This tag defines the document type and HTML version.

**<HTML>**

This tag encloses the complete HTML document and mainly comprises of document header which is represented by <head>...</head> and document body which is represented by <body>...</body> tags.

**<HEAD>**

This tag represents the document's header which can keep other HTML tags like <title>, <link> etc.

**<TITLE>**

The <title> tag is used inside the <head> tag to mention the document title.

**<BODY>**

This tag represents the document's body which keeps other HTML tags like<h1>, <div>, <p> etc.

HTML Elements are the building blocks of HTML pages. With HTML constructs, images and other objects, such as [interactive forms,](https://en.wikipedia.org/wiki/Fieldset) may be embedded into the rendered page. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets. Tags such as<img/> and <input/> introduce content into the page directly. Others such as<p>…. </p>, surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.

**2.4 Cascading Style Sheet (CSS):**

**Cascading Style Sheets** was invented by **HakonWium Lie** on October 10, 1994 and maintained through a group of people within the W3C called the CSS Working Group

CSSis the language for describing the presentation of Web pages, including colors, layout, and fonts. It allows one to adapt the presentation to different types of devices, such as large screens, small screens, or printers. CSS is independent of HTML and can be used with any XML-based markup language. The separation of HTML from CSS makes it easier to maintain sites, share style sheets across pages, and tailor pages to different environments. There are three types CSS mainly

* External CSS
* Internal CSS
* Inline CSS

**External** style sheets are separate files full of CSS instructions (with the file extension .css). When any web page includes an external style sheet, its look and feel will be controlled by this CSS file. This is how you change a whole website at once. And that's perfect if you want to keep up with the latest fashion in web pages without rewriting every page.

**Internal** styles are placed at the top of each web page document, before any of the content is listed. This is the next best thing to external, because they're easy to find, yet allow you to 'override' an external style sheet -- for that special page that wants to be a nonconformist.

**Inline** styles are placed right where you need them, next to the text or graphic you wish to decorate. You can insert inline styles anywhere in the middle of your HTML code, giving you real freedom to specify each web page element. On the other hand, this can make maintaining web pages a real chore.

**Advantages of CSS**

* **CSS saves time** − You can write CSS once and then reuse same sheet in multiple HTML pages. You can define a style for each HTML element and apply it to as many Web pages as you want.
* **Pages load faster** − If you are using CSS, you do not need to write HTML tag attributes every time. Just write one CSS rule of a tag and apply it to all the occurrences of that tag. So less code means faster download times.
* **Easy maintenance** − To make a global change, simply change the style, and all elements in all the web pages will be updated automatically.
* **Superior styles to HTML** − CSS has a much wider array of attributes than HTML, so you can give a far better look to your HTML page in comparison to HTML attributes.
* **Multiple Device Compatibility** − Style sheets allow content to be optimized for more than one type of device. By using the same HTML document, different versions of a website can be presented for handheld devices such as PDAs and cell phones or for printing.
* **Global web standards** − Now HTML attributes are being deprecated and it is being recommended to use CSS. So it’s a good idea to start using CSS in all the HTML pages to make them compatible to future browsers.
* **Offline Browsing** − CSS can store web applications locally with the help of an offline cache.Using of this, we can view offline websites.The cache also ensures faster loading and better overall performance of the website.
* **Platform Independence** − The Script offer consistent platform independence and can support latest browsers as well.

**2.5 MYSQL:**

SQL is a Relational Database Management System (RDBMS) that runs exclusively under the Windows operating system. One benefit of using Windows exclusively is that we can send and receive E-mail messages based on SQL "events" and we can also let the operating system handle login security. The data base is an organized collection of data. A database management system (DBMS) such as Access, FileMaker Pro, Oracle or a flexible manner. It includes facilities to add, modify or delete data from the SQL Server provides we with the software tools we need to organize that data in database, ask questions (or queries) about the data stored in the database and produce reports summarizing selected contents.

**Database Connection**

<?php

$servername ="localhost";

$username = "username";

$password = "password";

$dbname = "database";

// Create connection

$conn =new mysqli($servername, $username, $password,$dbname);

// Check connection

if (!$conn)

{

die("Connection failed: " . $conn->connect\_error());

}

?>

**2.5.1 Features of SQL:**

* It is simple English like language and uses simple commands such as SELECT, CREATE, DROP etc.
* It is not having condition loops, variables and most of the commands are single line commands.
* To implement application logics, SQL has got extension language popularly called as PL/SQL (Procedural language of sql).
* The entire SQL has been divided into 4 major categories.
* Data Manipulation Language.
* Data Definition Language.
* Transaction Control language.
* Data Control Language.

**2.5.2 Security:**

View are basically used as a part of security, means in many organizations ,the end user will never be given original tables & all data entry will be done with the help of views only. But the database administrator will be able to see everything because all the operations done by the different users will come to the same table.

**2.6 BRACKETS**

**Brackets** is an light weight &opensource editor written in HTML, [CSS](https://en.wikipedia.org/wiki/CSS), and [JavaScript](https://en.wikipedia.org/wiki/JavaScript) with a primary focus on [web development](https://en.wikipedia.org/wiki/Web_Development). It was created by [Adobe Systems](https://en.wikipedia.org/wiki/Adobe_Systems), licensed under the [MITLicense](https://en.wikipedia.org/wiki/MIT_License), and is currently maintained on [GitHub](https://en.wikipedia.org/wiki/GitHub).Brackets is available for Mac OS X, Windows and Linux.

Brackets provide several featuresincluding:

* Quick Edit
* Live Preview

**2.7.1 Quick Edit**

Quick edit enables inline editing of CSS, Color Property, and JavaScript elements for developers. This built-in feature can be applied to multiple functions or properties simultaneously and all updates are applied directly to the file associated with the changed elements.

* [**HTML**](https://en.wikipedia.org/wiki/HTML)**File**
* Applying quick edit to [HTML elements](https://en.wikipedia.org/wiki/HTML_element) will display all corresponding [CSS](https://en.wikipedia.org/wiki/CSS) properties in a box beneath the selected element. Users can choose to create new [CSS](https://en.wikipedia.org/wiki/CSS) rules directly within the [editor](https://en.wikipedia.org/wiki/HTML_editor) and edit a [tag's](https://en.wikipedia.org/wiki/HTML_element) [CSS](https://en.wikipedia.org/wiki/CSS) properties inline without leaving the context of the [HTML](https://en.wikipedia.org/wiki/HTML) file.
* [**JavaScript**](https://en.wikipedia.org/wiki/JavaScript)**File**
* On [JavaScript](https://en.wikipedia.org/wiki/JavaScript) functions, quick edit performs the same procedure as with [HTML elements](https://en.wikipedia.org/wiki/HTML_element) but displays the selected function’s body within the drop down box. All updates to the function’s body will propagate and update directly within the corresponding [JavaScript](https://en.wikipedia.org/wiki/JavaScript) file.
* **Files containing**[**Hex**](https://en.wikipedia.org/wiki/Hexadecimal)**or**[**RGB**](https://en.wikipedia.org/wiki/RGB)**color properties**
* For color properties, quick edit will return an inline color picker for previewing and color adjustment functionality.

**2.7.2 Live Preview**

When one clicks the respective code snippet in [CSS](https://en.wikipedia.org/wiki/CSS)/[HTML](https://en.wikipedia.org/wiki/HTML) the web browser immediately shows the output relating to that code snippet in web browser. This feature is termed as Live Highlight. Also, the feature Live Preview in Brackets pushes code edits instantly to the [browser](https://en.wikipedia.org/wiki/Web_browser) to present an updated webpage as the developers modify the code. Brackets contains a [Node.js](https://en.wikipedia.org/wiki/Node.js) backend that predicts what the code does as the developer types the code

**2.7.3 Major Features**

1. **Live Preview:** It launches a new browser window showing the current page that not only does not require a manual refresh when you change a file, but also as you type. As such, Brackets feels like an extension of your browser’s Developer tools. This feature is incredibly useful, as it can combine the advantages of coding in the browser with those of an editor.
2. **Quick Edit:** When you are editing HTML code, if you click a tag with a corresponding CSS declaration situated in a linked file and hit Ctrl/Command+E, an inline editor appears allowing you to quickly edit that rule.
3. **JSLint:** It works with your JavaScript code upon saving, with its results displayed as a panel below the main editing window.
4. Brackets implements code completion especially meant for web designers and front-end development.
5. Additional features such as command line integration, multiple code selection, and quick open and quick find, add to the highlights of Brackets code editor.

**2.7.4 Brackets Extensions**

**Beautify**

[Beautify](https://github.com/brackets-beautify/brackets-beautify) Brackets extension makes your code look good. Not only does it make your files more readable, but it also gives you a handy shortcut key to beautify those files — CMD-Shift-L (Mac) and Ctrl-Shift-L (Win). JavaScript code can be consistently formatted and made more readable using Beautify.

**Autoprefixer**

[Autoprefixer](https://github.com/mikaeljorhult/brackets-autoprefixer) is a postprocessor for dealing with vendor prefixes in the best possible way. Autoprefixer parses CSS files and adds vendor prefixes to CSS rules. All you have to do is to add it to your asset building tool. Autoprefixer cleans outdated prefixes as well.

**Emmet**

[Emmet](https://github.com/emmetio/brackets-emmet) helps improve your HTML and CSS workflow by letting you type in CSS-like expressions that can be dynamically parsed and produce an output according to what you actually typed in the abbreviation.

Emmet abbreviations look like CSS selectors but are parsed in runtime and turned into a structured code block with just a single key press.

Once you learn the abbreviations, it’s a helpful way to speed up your workflow.

**Minifier**

[Minifier](https://github.com/alfredxing/brackets-minifier/) minifies JavaScript and CSS files in brackets and saves them as filename.min.ext

To minify a file, use the keyboard shortcut CMD+M or Ctrl+M. It compresses files and saves your time as it minifies the relevant JavaScript automatically so you can continue to work and have the file minified and fully ready for deployment.

**HTML Skeleton**

[HTML Skeleton](https://github.com/le717/brackets-html-skeleton) is a collection of a set of tags required by every HTML webpage you build. The tags that make up the skeleton tell browsers what kind of file it is reading and without the skeleton, HTML files will not be rendered correctly in web browsers. This is a must-have extension if you regularly use HTML to build pages in Brackets.

**HTML Wrapper**

[HTML Wrapper](https://github.com/rcaferati/brackets-html-wrapper) is a Brackets extension that formats nav and selects tags with a single command. You can use it by selecting a list of items encapsulated by the desired tag, and then run the command.

**JavaScript and CSS CDN Suggestions**

[Brackets CDN extension](https://github.com/szdc/brackets-cdn-suggestions) lets you work with CDN services. As you already know, a Content Delivery Network (CDN) saves your web server resources by loading files via cloud servers.

**4.3 Source Code**

**News details form**

<!DOCTYPE html>

<html lang="en">

<?php include('1\_metatags.php');?>

<body class="sidebar-dark">

<!-- partial:partials/\_settings-panel.html -->

<div class="settings-panel">

<ul class="nav nav-tabs" id="setting-panel" role="tablist">

<li class="nav-item">

<a class="nav-link active" id="layouts-tab" data-toggle="tab" href="#layouts-section" role="tab" aria-controls="layouts-section" aria-expanded="true"><i class="mdi mdi-settings"></i></a>

</li>

<li class="nav-item">

<a class="nav-link" id="chats-tab" data-toggle="tab" href="#chats-section" role="tab" aria-controls="chats-section"><i class="mdi mdi-account"></i></a>

</li>

<li class="nav-item">

<a class="nav-link" id="close-button" href="#"><i class="mdi mdi-window-close"></i></a>

</li>

</ul>

</div>

<!-- partial -->

<div class="container-scroller">

<!-- partial:partials/\_navbar.html -->

<?php include('2\_topnav.php');?>

<!-- partial -->

<div class="container-fluid page-body-wrapper">

<div class="row row-offcanvas row-offcanvas-right">

<!-- partial:partials/\_sidebar.html -->

<?php include('3\_sidebar.php');?>

<!-- partial -->

<div class="content-wrapper">

<h1 class="page-title">NEWS DETAILS</h1>

<div class="row">

<div class="col-lg-12 mb-4">

<div class="card">

<div class="card-body">

<!-- <h2 class="card-title">FORM DETAILS</h2>-->

<form method="post" enctype="multipart/form-data" onsubmit="return ValidateForm()" action="news\_details\_insert.php">

<div class="form-group">

<label for="inputName1" class="control-label">NAME</label>

<select name="news\_category\_id" id="news\_category\_id" class="form-control">

<option value="">--SELECT--</option>

<?php

include\_once("db\_connection.php");

$sql1=$conn->prepare("SELECT \* FROM news\_category");

$sql1->execute();

$result1=$sql1->get\_result();

while($row1=$result1->fetch\_assoc()){

?>

<option value="<?php echo $row1["news\_category\_id"];?>"><?php echo $row1["news\_category\_name"];?></option>

<?php

}

?>

</select>

<span id="news\_category\_id\_error"></span>

</div>

<div class="form-group">

<label for="inputEmail" class="control-label">CONTENT TYPE</label>

<input type="text" id="content\_type" name="content\_type" class="form-control" placeholder="Enter content type">

<span id="content\_type\_error"></span>

</div>

<div class="form-group">

<label for="textarea" class="control-label">REPORTERS NAME</label>

<select name="reporters\_id" id="reporters\_id" class="form-control">

<option value="">--SELECT--</option>

<?php

include\_once("db\_connection.php");

$sql1=$conn->prepare("SELECT \* FROM reporters");

$sql1->execute();

$result1=$sql1->get\_result();

while($row1=$result1->fetch\_assoc()){

?>

<option value="<?php echo $row1["reporters\_id"];?>"><?php echo $row1["fullname"];?></option>

<?php

}

?>

</select>

<span id="reporters\_id\_error"></span>

</div>

<div class="form-group">

<label for="textarea" class="control-label">HEADLINES</label>

<input type="text" id="news\_details\_headlines" name="news\_details\_headlines" class="form-control" placeholder="Enter headlines">

<span id="news\_details\_headlines\_error"></span>

</div>

<div class="form-group">

<label for="textarea" class="control-label">IMAGE</label>

<input type="file" id="news\_details\_image" name="news\_details\_image" class="form-control">

<span id="news\_details\_image\_error"></span>

</div>

<div class="form-group">

<label for="inputPassword" class="control-label">DESCRIPTION</label>

<textarea type="text" id="news\_details\_description" name="news\_details\_description" class="form-control" placeholder="Enter description"></textarea>

<script type="application/javascript">CKEDITOR.replace('news\_details\_description');</script>

<span id="news\_details\_description\_error"></span>

</div>

<div class="form-group">

<label for="inputPassword" class="control-label">DATE</label>

<input type="text" id="news\_details\_date" name="news\_details\_date" class="form-control" value="<?php echo date("Y-m-d");?>" readonly>

<span id="news\_details\_date\_error"></span>

</div>

<div class="form-group">

<label for="inputPassword" class="control-label">VIDEO NAME</label>

<input type="text" id="news\_details\_video\_id" name="news\_details\_video\_id" class="form-control" placeholder="Enter video name">

<span id="news\_details\_video\_id\_error"></span>

</div>

<div class="form-group">

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

</div>

</form>

</div>

</div>

</div>

</div>

</div>

<!-- content-wrapper ends -->

<!-- partial:partials/\_footer.html -->

<?php include('5\_footer\_name.php');?>

<!-- partial -->

</div>

<!-- row-offcanvas ends -->

</div>

<!-- page-body-wrapper ends -->

</div>

<!-- container-scroller -->

<!-- plugins:js -->

<?php include('4\_footer.php');?>

<script type="text/javascript">

function ValidateForm()

{

var news\_category\_id = '';

var content\_type = '';

var reporters\_id = '';

var news\_details\_headlines = '';

var news\_details\_image = '';

var news\_details\_description = '';

var news\_details\_video\_id = '';

news\_category\_id = fieldrequired('news\_category\_id', 'news\_category\_id\_error', 'NAME');

content\_type = fieldrequired('content\_type', 'content\_type\_error', 'CONTENT TYPE');

reporters\_id = fieldrequired('reporters\_id', 'reporters\_id\_error', 'REPORTERS NAME');

news\_details\_headlines = fieldrequired('news\_details\_headlines', 'news\_details\_headlines\_error', 'HEADLINES');

news\_details\_image = imagename('news\_details\_image', 'news\_details\_image\_error', 'IMAGE');

news\_details\_description = ckeditor\_val('news\_details\_description', 'news\_details\_description\_error', 'DESCRIPTION');

news\_details\_video\_id = fieldrequired('news\_details\_video\_id', 'news\_details\_video\_id\_error', 'VIDEO NAME');

if (news\_category\_id == 1 && content\_type == 1 && reporters\_id == 1 && news\_details\_headlines == 1 && news\_details\_image == 1 && news\_details\_description == 1 && news\_details\_video\_id == 1)

{

return true;

}

else

{

return false;

}

}

</script>

</body>

</html>

**News details insert**

<?php

include\_once("db\_connection.php");

include\_once("6\_function.php");

$news\_category\_id =$\_POST['news\_category\_id'];

$content\_type =$\_POST['content\_type'];

$reporters\_id=$\_POST['reporters\_id'];

$news\_details\_headlines=$\_POST['news\_details\_headlines'];

//$news\_details\_image=$\_POST['news\_details\_image'];

$news\_details\_description=$\_POST['news\_details\_description'];

$folder="photo/";

$news\_details\_image=$\_FILES['news\_details\_image']['name'];

$tmp\_news\_details\_image=$\_FILES['news\_details\_image']['tmp\_name'];

$news\_details\_image=upload\_image($news\_details\_image,$tmp\_news\_details\_image,$folder);

$news\_details\_description=$\_POST['news\_details\_description'];

$news\_details\_date=$\_POST['news\_details\_date'];

$news\_details\_video\_id=$\_POST['news\_details\_video\_id'];

$sql=$conn->prepare("INSERT INTO news\_details(news\_category\_id,content\_type,reporters\_id,news\_details\_headlines,news\_details\_image,news\_details\_description,news\_details\_date,news\_details\_video\_id)VALUES(?,?,?,?,?,?,?,?)");

$sql->bind\_param("ssisssss",$news\_category\_id,$content\_type,$reporters\_id,$news\_details\_headlines,$news\_details\_image,$news\_details\_description,$news\_details\_date,$news\_details\_video\_id);

if($sql->execute())

{

echo "<script type='text/javascript'>

alert('RECORD INSERTED');

window.location='news\_details\_view.php';

</script>";

}

else

{

echo "<script type='text/javascript'>

alert('RECORD NOT INSERTED');

window.location='news\_details\_view.php';

</script>";

}

?>

**News details edit**

<!DOCTYPE html>

<html lang="en">

<?php include('1\_metatags.php');?>

<body class="sidebar-dark">

<!-- partial:partials/\_settings-panel.html -->

<div class="settings-panel">

<ul class="nav nav-tabs" id="setting-panel" role="tablist">

<li class="nav-item">

<a class="nav-link active" id="layouts-tab" data-toggle="tab" href="#layouts-section" role="tab" aria-controls="layouts-section" aria-expanded="true"><i class="mdi mdi-settings"></i></a>

</li>

<li class="nav-item">

<a class="nav-link" id="chats-tab" data-toggle="tab" href="#chats-section" role="tab" aria-controls="chats-section"><i class="mdi mdi-account"></i></a>

</li>

<li class="nav-item">

<a class="nav-link" id="close-button" href="#"><i class="mdi mdi-window-close"></i></a>

</li>

</ul>

</div>

<!-- partial -->

<div class="container-scroller">

<!-- partial:partials/\_navbar.html -->

<?php include('2\_topnav.php');?>

<!-- partial -->

<div class="container-fluid page-body-wrapper">

<div class="row row-offcanvas row-offcanvas-right">

<!-- partial:partials/\_sidebar.html -->

<?php include('3\_sidebar.php');?>

<!-- partial -->

<div class="content-wrapper">

<h1 class="page-title">NEWS DETAILS</h1>

<div class="row">

<div class="col-lg-12 mb-4">

<div class="card">

<div class="card-body">

<!-- <h2 class="card-title">FORM DETAILS</h2>-->

<?php

include\_once("db\_connection.php");

$news\_details\_id=$\_REQUEST['id']; //TO FETCH PK

$sql=$conn->prepare("SELECT \* FROM news\_details WHERE news\_details\_id=?");

$sql->bind\_param("i",$news\_details\_id);

$sql->execute();

$result=$sql->get\_result();

$row=$result->fetch\_assoc();

?>

<form method="post" enctype="multipart/form-data" onsubmit="return ValidateForm()" action="news\_details\_update.php">

<input type="hidden" name="news\_details\_id" id="news\_details\_id" value="<?php echo $row['news\_details\_id']; ?>">

<div class="form-group">

<label for="inputName1" class="control-label">NAME</label>

<select name="news\_category\_id" id="news\_category\_id" class="form-control">

<?php

include\_once("db\_connection.php");

$sql1=$conn->prepare("SELECT \* FROM news\_category");

$sql1->execute();

$result1=$sql1->get\_result();

while($row1=$result1->fetch\_assoc()){

?>

<option value="<?php echo $row1["news\_category\_id"];?>"

<?php

if($row["news\_category\_id"]==$row1["news\_category\_id"]){

?>

selected

<?php

}

?>

><?php echo $row1["news\_category\_name"];?></option>

<?php

}

?></select>

<span id="news\_category\_id\_error"></span>

</div>

<div class="form-group">

<label for="inputEmail" class="control-label">CONTENT TYPE</label>

<input type="text" id="content\_type" name="content\_type" class="form-control" placeholder="Enter content type" value="<?php echo $row["content\_type"]; ?>">

<span id="content\_type\_error"></span>

</div>

<div class="form-group">

<label for="textarea" class="control-label">REPORTERS NAME</label>

<select name="reporters\_id" id="reporters\_id" class="form-control">

<?php

include\_once("db\_connection.php");

$sql2=$conn->prepare("SELECT \* FROM reporters");

$sql2->execute();

$result2=$sql2->get\_result();

while($row2=$result2->fetch\_assoc()){

?>

<option value="<?php echo $row2["reporters\_id"];?>"

<?php

if($row["reporters\_id"]==$row2["reporters\_id"]){

?>

selected

<?php

}

?>

><?php echo $row2["fullname"];?></option>

<?php

}

?>

</select>

<span id="reporters\_id\_error"></span>

</div>

<div class="form-group">

<label for="textarea" class="control-label">HEADLINES</label>

<input type="text" id="news\_details\_headlines" name="news\_details\_headlines" class="form-control" placeholder="Enter headlines" value="<?php echo $row['news\_details\_headlines']; ?>">

<span id="news\_details\_headlines\_error"></span>

</div>

<div class="form-group">

<label for="textarea" class="control-label">IMAGE</label>

<img src="photo/<?php echo $row["news\_details\_image"];?>" width="150" height="150">

<input type="file" id="news\_details\_image" name="news\_details\_image" class="form-control" value="<?php echo $row['news\_details\_image']; ?>">

<input type="hidden" name="old\_news\_details\_image" id="old\_news\_details\_image" value="<?php echo $row["news\_details\_image"];?>">

</div>

<div class="form-group">

<label for="inputPassword" class="control-label">DESCRIPTION</label>

<textarea type="text" id="news\_details\_description" name="news\_details\_description" class="form-control" placeholder="Enter description"><?php echo $row['news\_details\_description']; ?></textarea>

<script type="application/javascript">CKEDITOR.replace('news\_details\_description');</script>

<span id="news\_details\_description\_error"></span>

</div>

<div class="form-group">

<label for="inputPassword" class="control-label">DATE</label>

<input type="text" id="news\_details\_date" name="news\_details\_date" class="form-control" value="<?php echo date("Y-m-d");?>" readonly>

<span id="news\_details\_date\_error"></span>

</div>

<div class="form-group">

<label for="inputPassword" class="control-label">VIDEO NAME</label>

<input type="text" id="news\_details\_video\_id" name="news\_details\_video\_id" class="form-control" placeholder="Enter video name" class="form-control" value="<?php echo $row['news\_details\_video\_id']; ?>">

<span id="news\_details\_video\_id\_error"></span>

</div>

<div class="form-group">

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

</div>

</form>

</div>

</div>

</div>

</div>

</div>

<!-- content-wrapper ends -->

<!-- partial:partials/\_footer.html -->

<?php include('5\_footer\_name.php');?>

<!-- partial -->

</div>

<!-- row-offcanvas ends -->

</div>

<!-- page-body-wrapper ends -->

</div>

<!-- container-scroller -->

<!-- plugins:js -->

<?php include('4\_footer.php');?>

<script type="text/javascript">

function ValidateForm()

{

var news\_category\_id = '';

var content\_type = '';

var reporters\_id = '';

var news\_details\_headlines = '';

var news\_details\_description = '';

var news\_details\_video\_id = '';

news\_category\_id = fieldrequired('news\_category\_id', 'news\_category\_id\_error', 'NAME');

content\_type = fieldrequired('content\_type', 'content\_type\_error', 'CONTENT TYPE');

reporters\_id = fieldrequired('reporters\_id', 'reporters\_id\_error', 'REPORTERS NAME');

news\_details\_headlines = fieldrequired('news\_details\_headlines', 'news\_details\_headlines\_error', 'HEADLINES');

news\_details\_description = ckeditor\_val('news\_details\_description', 'news\_details\_description\_error', 'DESCRIPTION');

news\_details\_video\_id = fieldrequired('news\_details\_video\_id', 'news\_details\_video\_id\_error', 'VIDEO NAME');

if (news\_category\_id == 1 && content\_type == 1 && reporters\_id == 1 && news\_details\_headlines == 1 && news\_details\_description == 1 && news\_details\_video\_id == 1)

{

return true;

}

else

{

return false;

}

}

</script>

</body>

</html>

**News details delete**

<?php

include\_once("db\_connection.php");

$news\_details\_id=$\_REQUEST["id"];

$sql=$conn->prepare("DELETE FROM news\_details WHERE news\_details\_id=?");

$sql->bind\_param("i",$news\_details\_id);

if($sql->execute()){

echo"<script type='text/javascript'>

alert('Record Deleted Successfully');

window.location='news\_details\_view.php';

</script>";

}

else{

echo"<script type='text/javascript'>

alert('Record Not Deleted');

window.location='news\_details\_view.php';

</script>";

}

?>

**News details view**

<!DOCTYPE html>

<html lang="en">

<?php include('1\_metatags.php');?>

<body class="sidebar-dark" onload=display\_ct();>

<!-- partial:partials/\_settings-panel.html -->

<div class="settings-panel">

<ul class="nav nav-tabs" id="setting-panel" role="tablist">

<li class="nav-item">

<a class="nav-link active" id="layouts-tab" data-toggle="tab" href="#layouts-section" role="tab" aria-controls="layouts-section" aria-expanded="true"><i class="mdi mdi-settings"></i></a>

</li>

<li class="nav-item">

<a class="nav-link" id="chats-tab" data-toggle="tab" href="#chats-section" role="tab" aria-controls="chats-section"><i class="mdi mdi-account"></i></a>

</li>

<li class="nav-item">

<a class="nav-link" id="close-button" href="#"><i class="mdi mdi-window-close"></i></a>

</li>

</ul>

</div>

<!-- partial -->

<div class="container-scroller">

<!-- partial:partials/\_navbar.html -->

<?php include\_once('2\_topnav.php');?>

<!-- partial -->

<div class="container-fluid page-body-wrapper">

<div class="row row-offcanvas row-offcanvas-right">

<!-- partial:partials/\_sidebar.html -->

<?php include\_once('3\_sidebar.php');?>

<!-- partial -->

<div class="content-wrapper">

<h1 class="page-title">NEWS DETAILS INFORMATION</h1>

<div class="card">

<div class="card-body">

<!-- <h2 class="card-title">Data table</h2>-->

<button type="button" class="btn btn-primary" onclick="window.location.href='news\_details\_form.php'">ADD NEW RECORD</button>

<br><br>

<div class="row">

<div class="col-12">

<table id="order-listing" class="table table-striped" style="width:100%;">

<thead>

<tr>

<th>SLNO</th>

<th>CATEGORY NAME</th>

<th>CONTENT</th>

<th>REPORTER NAME</th>

<th>HEADLINES</th>

<th>IMAGE</th>

<th>DESCRIPTION</th>

<th>DATE</th>

<th>VIDEO NAME</th>

<th>EDIT</th>

<th>DELETE</th>

</tr>

</thead>

<tbody>

<?php

$sno=1;

include\_once("db\_connection.php");

$sql=$conn->prepare("SELECT \* FROM news\_details,news\_category,reporters WHERE news\_details.news\_category\_id=news\_category.news\_category\_id AND news\_details.reporters\_id=reporters.reporters\_id");

$sql->execute();

$result=$sql->get\_result();

while($row=$result->fetch\_assoc()){

?>

<tr>

<td><?php echo $sno++;?></td>

<td><?php echo $row["news\_category\_name"];?></td>

<td><?php echo $row["content\_type"];?></td>

<td><?php echo $row["fullname"];?></td>

<td><?php echo $row["news\_details\_headlines"];?></td>

<td><img src="photo/<?php echo $row["news\_details\_image"]; ?>" alt="no image" width="100" height="100"></td>

<td><?php echo $row["news\_details\_description"];?></td>

<td><?php echo $row["news\_details\_date"];?></td>

<td><?php echo $row["news\_details\_video\_id"];?></td>

<td><a class="btn btn-info" href="news\_details\_edit.php?id=<?php echo $row['news\_details\_id']; ?>">EDIT</a></td>

<td><a class="btn btn-danger" href="news\_details\_delete.php?id=<?php echo $row['news\_details\_id'];?>">DELETE</a></td>

</tr>

<?php

}

?>

</tbody>

</table>

</div>

</div>

</div>

</div>

</div>

<!-- content-wrapper ends -->

<!-- partial:partials/\_footer.html -->

<?php include('5\_footer\_name.php');?>

<!-- partial -->

</div>

<!-- row-offcanvas ends -->

</div>

<!-- page-body-wrapper ends -->

</div>

<?php include('4\_footer.php');?>

<?php include('6\_time\_script.php');?>

</body>

</html>

**News details update**

<?php

include\_once("db\_connection.php");

include\_once("6\_function.php");

$news\_details\_id=$\_POST['news\_details\_id']; //PK

$news\_category\_id=$\_POST['news\_category\_id'];

$content\_type=$\_POST['content\_type'];

$reporters\_id=$\_POST['reporters\_id'];

$news\_details\_id=$\_POST['news\_details\_id'];

$news\_details\_headlines=$\_POST['news\_details\_headlines'];

$news\_details\_description=$\_POST['news\_details\_description'];

$folder="photo/";

$news\_details\_image=$\_FILES['news\_details\_image']['name'];

if(empty($news\_details\_image)){

$news\_details\_image=$\_POST["old\_news\_details\_image"];

}

else{

$tmp\_image=$\_FILES['news\_details\_image']['tmp\_name'];

$news\_details\_image=upload\_image($news\_details\_image,$tmp\_image,$folder);

}

$news\_details\_description=$\_POST['news\_details\_description'];

$news\_details\_date=$\_POST['news\_details\_date'];

$news\_details\_video\_id=$\_POST['news\_details\_video\_id'];

$sql=$conn->prepare("UPDATE news\_details SET news\_category\_id=?,content\_type=?,reporters\_id=?,news\_details\_headlines=?,news\_details\_image=?,news\_details\_description=?,news\_details\_date=?,news\_details\_video\_id=? WHERE news\_details\_id=?");

$sql->bind\_param("isisssssi",$news\_category\_id,$content\_type,$reporters\_id,$news\_details\_headlines,$news\_details\_image,$news\_details\_description,$news\_details\_date,$news\_details\_video\_id,$news\_details\_id);

if($sql->execute())

{

echo "<script type='text/javascript'>

alert('RECORD UPDATED');

window.location='news\_details\_view.php';

</script>";

}

else

{

echo "<script type='text/javascript'>

alert('RECORD NOT UPDATED');

window.location='news\_details\_view.php';

</script>";

}

?>

1. **TESTING**

**5.1 Details of Testing Methodologies**

Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and code generation. Testing is done to see if the entire feature provided by the module are performing satisfactorily and to ensure that the process of testing is as realistic as possible. Testing part is done to major phases via unit level and module level testing. The idea of testing the software in a phased manner is to identify and isolate the bugs for any easy correction. Both phases are over and results meet with the user needs.

**Test Approaches:**

**Black box Testing:**

Black box testing is done to find:

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

**White box Testing:**

White box testing is done to find out:

* Check whether all independent paths within a module have been exercised at least once
* Exercise all logical decision on their true and false sides
* Execute all loops at their boundaries and within their bounds.
* Exercise the internal data structure to ensure their validity.
* Ensure whether all the possible validity checks and validity lookups have been provided to validate data entry.

**Testing Strategies:**

**Levels of testing:**

Client Needs Acceptance Testing

Requirements System Testing

Design Integration Testing

Coding Unit Testing

The different testing is carried out on the reflects the effectiveness and

Efficiency of different phases of software development these test help to uncover the error in the corresponding phase.

* **Unit testing** is carried out to check the coding errors and program logic.
* **Integration testing** which also known as moduletesting uncovers the system design errors.
* **Performance testing** is performed to determine how fast some aspect of a system performs under a particular workload.
* **User Acceptance testing** is done to test whether the product developed needs the client needs and acceptable to him.

**Unit Testing:**

Unit testing involves, checking all the modules in the system individually against the specification produced during the design of the module and for their performance. Unit testing also involves code produced in the coding phase and hence the internal logic of the program. Each module is tested for different test cases design to check each specific combination of conditions handled by the program. Error handlers are included in each module for each event trap and handled the errors.

Unit testing is done by inputting proper input s in each page and checking whether the data is in correct format as used to backend. Each servlet is also checked whether each method is working properly to fulfill the requirement.

**Integration testing:**

Integration testing takes as its input modules that have been checked out by unit testing, groups them in larger aggregates, applies tests defined in an Integration test plan to those aggregates, and delivers as its output the integrated system ready for system testing. The purpose of integration testing is to verify functional, performance and reliability requirements placed on major design items.

In our project we integrate all the .ASPX pages and run complete module and check for all conditions whether it’s working properly.

**System Testing:**

**System testing** of software or hardware is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. System testing falls within the scope of black box testing, and as such, should require no knowledge of the inner design of the code or logic. As a rule, system testing takes, as its input, all of the "integrated" software components that have successfully passed integration testing and also the software system itself integrated with any applicable hardware system(s). The purpose of integration testing is to detect any inconsistencies between the software units that are integrated together (called *assemblages*) or between any of the *assemblages* and the hardware. System testing is a more limited type of testing; it seeks to detect defects both within the "inter-assemblages" and also within the system as a whole.

**Security Testing:**

This attempts to verify whether the protection mechanism is built into the system. In this testing the authentication of the users is checked and authorized users are allowed to access the database.

**Performance Testing:**

Performance testing can serve different purpose. It can demonstrate that the system meets the performance criteria. It can compare two systems to find which performs better, or it can measure what parts of the system or workload cause the system to perform badly. In the diagnostic case, software engineers use tools such as profilers to measure what parts of a device or Software contributes most to the poor performance.

#### Acceptance testing

Acceptance testing can mean one of two things:

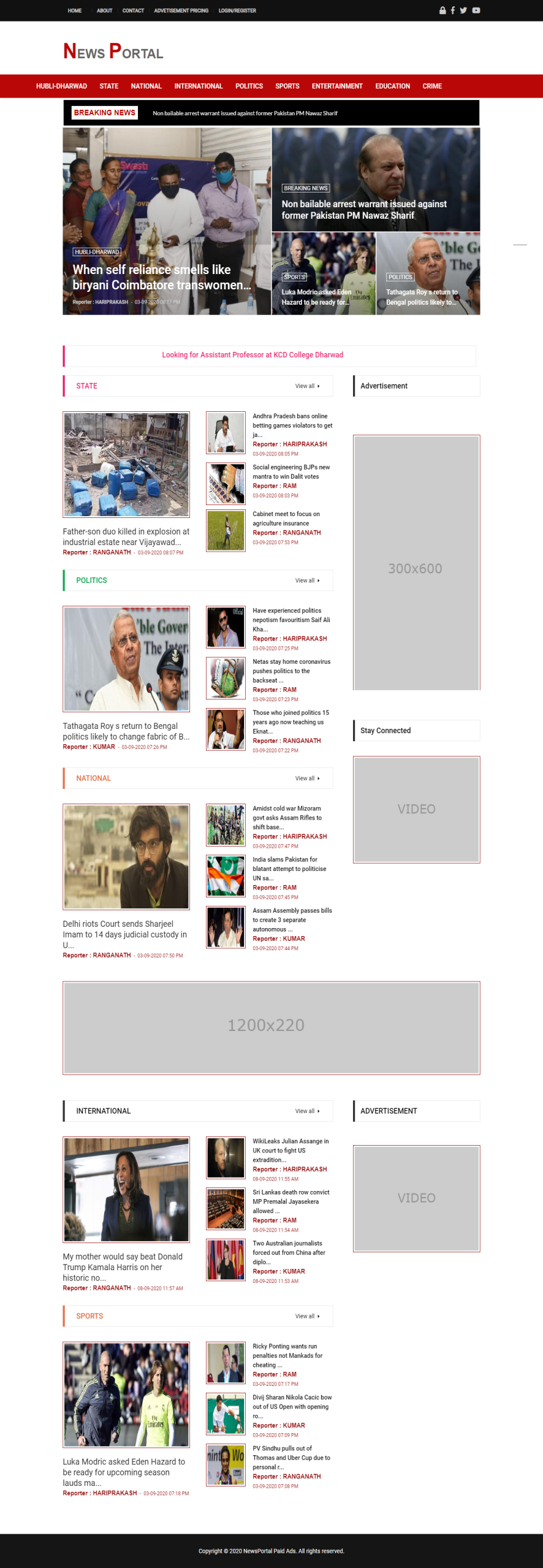
1. A smoke test is used as an acceptance test prior to introducing a new build to the main testing process, i.e. before integration or regression.
2. Acceptance testing performed by the customer, often in their lab environment on their own hardware, is known as user acceptance testing (UAT). Acceptance testing may be performed as part of the hand-off process between any two phases of development

**5.2 Test Cases**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case** | **INPUT** | **EXPECTED OUTPUT** | **ACTUAL OUTPUT** | **RESULT** |
| 1 | Valid Username and Password | Should display respective page according to user type | Respective Home is displayed | Passed |
| 2 | Invalid username and password | It should give appropriate error message saying “Username or Password is Incorrect | Error message displayed | Passed |
| 3 | Enter Registered Mobile Number | Validate Mobile Number and Redirect to OTP Page | Mobile Number Verified and OTP Page is Displayed | Passed |
| 4 | Enter OTP | Validate OTP,Allow excess to Change Password | OTP Validated and Password Changed Successfully | Passed |
| 5 | Admin can Add /Update/Delete News details | Add/Update/Delete action should take place properly | Add/Update/Delete done successfully | Passed |
| 6 | Customer can view the latest news details based on categories | Respective news category details information should be displayed and confirmation | Respective news category details is displayed | Passed |

1. **SCREENSHOTS**

**Home Page**

****

**LOGIN PAGE**

****

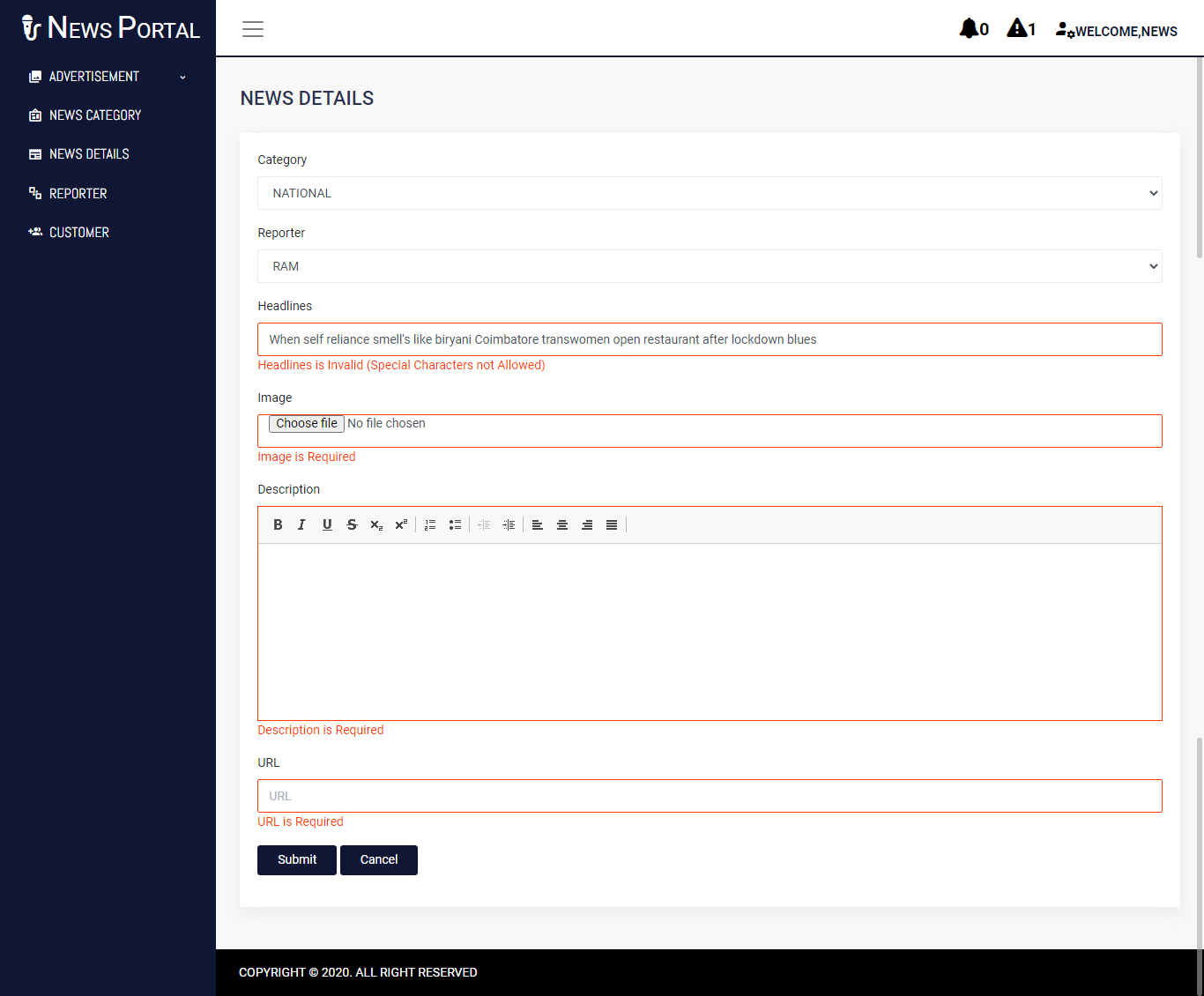
**DASHBOARD**

****

**NEWS DETAILS**

****

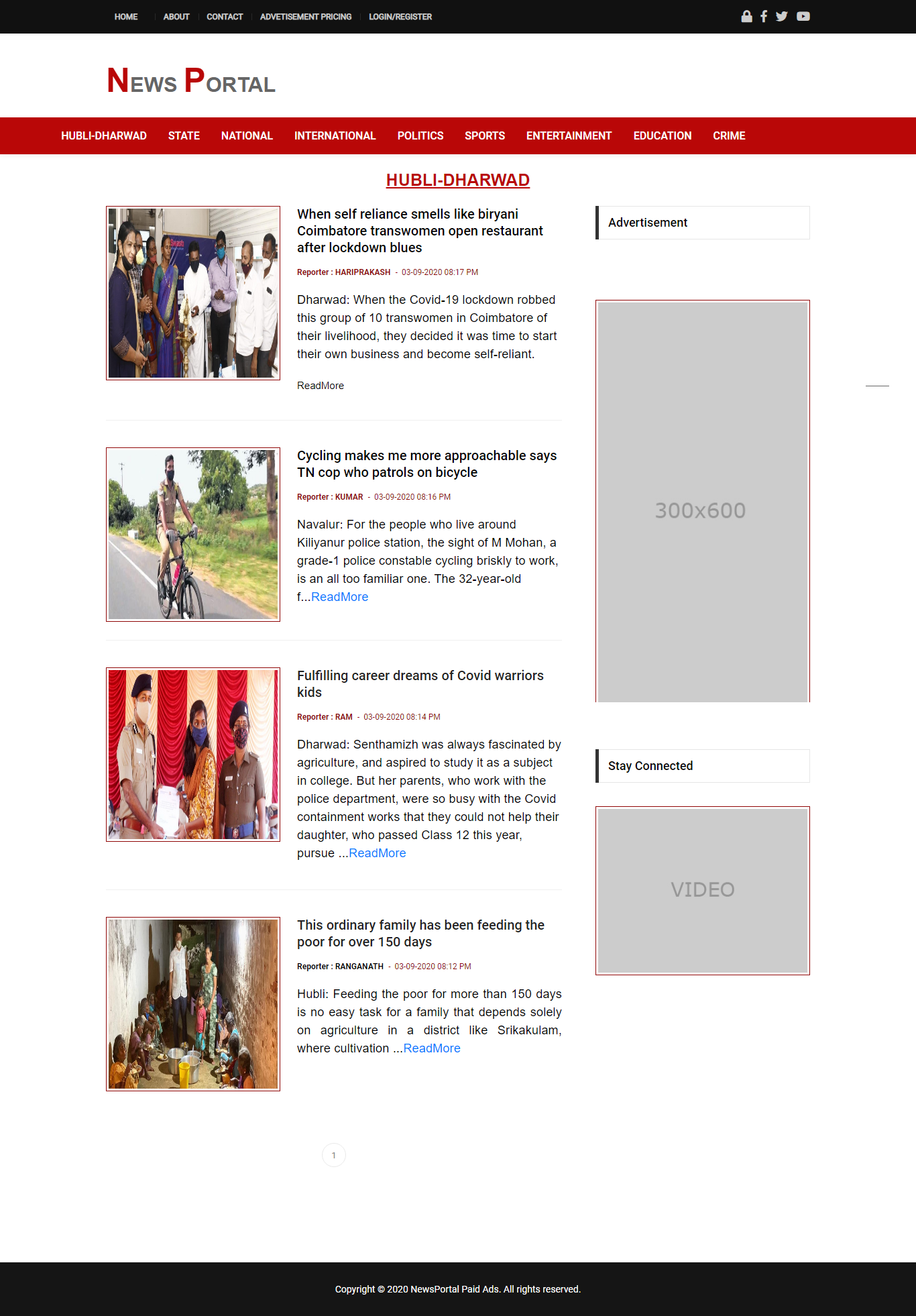
**VALIDATION PAGE**

****

**STATE NEWS**

****

**HUBLI-DHARWAD NEWS CATEGORY PAGE**

****

1. **CONCLUSION**

Software is said to have attained its objective only when it meets all the requirements of the user, further the user himself is the person to judge the success of the system.

Developing this project helped me to gain some experience in real time development procedures. Hereby I conclude that the system will surely be a valuable preposition in contrast to the changing business requirements and modern day cut-throat competition.

Each of modules designed into his project are independent to each other. So we can add other modules as and when needed. Every attempt has been made to ensure that the system is fully functional and works effectively and efficiently.

The system has been tested with simple data to cover all possible options and checked for all outputs. Since the system is flexible and modular, further modifications of this package can be easily incorporated.

**8. REFERENCES**

**Websites:**

1)http://php.net/manual/en/language.types.intro.php

*2)*<https://docs.oracle.com/cd/B19306_01/server.102/b14200.pdf>

3) http://www.w3schools.com

**Reference Books**

1. Software Engineering”, by Ian Somerville, Sixth Edition, Pearson Education Ltd 2007.
2. “Web Programming”, by ‘Chris Bates’ Wiley Dreamtech India, 2nd Edition.
3. “Data Base Management System” by Steve ‘Shamkantnavathe’And Ramezelmarsi