

# *Content Management System*

*University Practical Examination*

*18CSC206J – Software Engineering & Project Management*

*School of Computing*

*SRM Institute of Science and Technology*

...

**Team 5**

B Sai Pranav- RA1911026010019

...



Edit with WPS Office

# **WEEK 1**

## **Business Case and Project Charter**



Edit with WPS Office

## 1. Executive Summary

The aim of our project is to create an innovative platform for content creators to upload their blogs in a way that can be managed in a relatively simple manner, the users can view their logs anytime and from anywhere in the world, the content won't have any limit and the data will be managed through meticulous database system. The users also have an option to monetize their content through ads and paid premium subscriptions provided by Google Ads.

## 2. Strategic Business Context

### 2.1. Business Need

In the aftermath of the covid-19 pandemic there has been a tremendous rise in the number of new content creators but these creators are having a tough time to grow and gain recognitions on the already existing content creation platforms due to them being saturated with already existing creators who have huge followings and to stand out in a such a scenario for a new creator is extremely difficult. Our platform aims to provide a new space where creators can express themselves while efficiently managing their content so they have more time to create and don't spend too much time on managing their content.

### 2.2. Business Outcomes

Our business outcome and target is to have sustainable growth and eventually become profitable in the long run all while maintaining an easy to use intuitive platform for our users and aspire to become the industry standard in the future.

## 3. Detailed Business Problem

### 3.1. Problem/Opportunity Statement

Many content creators have a tough time managing their content while creating. Our platform will help them keep track of their content and interpret the feedback in an accurate manner

### 3.2. High Level Requirements

<Specify the key requirements to fully address the business need. Though the requirements are simply an elaboration of the business need, keep to a reasonable level of detail that allows for their comparison against the options >

### 3.3. Assumptions

<List and describe all the assumptions associated with the ability to address the key requirements. >

State assumptions that, for planning purposes, are considered to be true, real, or certain. These



Edit with WPS Office

assumptions will be validated during the planning process. Inaccurate, inconsistent, or incomplete assumptions result in project risks.

S.No	Assumptions

### 3.4. Constraints

S.No	Category	Constraints
1	Budget	As we are college students, we don't have a huge budget for servers.
2	Security/Privacy	A high encryption website needs a high budget, since we are a low budget CMS, our platform will suffer from lack of security.

### 3.5. Dependencies

Our project is related and dependent on server side technology like HEROKU, AWS, and MongoDB.

- These Technologies updates in a short moment of time, and sometimes the update makes it impossible for us to run the technology without modifying the code.
- This project requires the deliverable from the servers, if they crash, our project also becomes useless
- This project also requires the deliverable from some APIs(Application programming interface), which provides some functionality in program.

Dependency Description	Critical Date	Contact
Server Technologies, updation and modifications.	N/A	CTO

### 3.6. Stakeholder Analysis

Name	Designation	Role in Project
Abhishek Hindocha	Corporate Head for Sales & Marketing	Supervising Sales and Marketing
Aditya Dixit	Chief Executive Officer (CEO)	Responsible for the entire project



		as a whole
Sai Pranav	Chief of technology Operations (CTO)	Cost Approval and administrative work
Kinjalk Parth, Kritika Sharma	Department Head(s)	Scope / Requirement Approver
Vaibhav Singh, Vanshika Dixit, Sarthak Gupta, Harshdeep Kaur	Business User(s) and Tester(s)	Validate the functionalities and handle the testing phase

## 4. Detailed Analysis

### 4.1. Evaluation Criteria

Evaluation Criteria	Deal Breakers (5)	Minimum Requirement (3)	Non-essential (1)	Score
Eg. Personal Identifiable Information (PII): Data must be encrypted 'At Rest and 'On Transit'	Y	N	N	5
Capital from investors and sponsors	Y	Y	N	8
Servers for expansion of platform	N	Y	N	3



## 4.2.Cost of each Possible Options

Options (#)	One Time [CapEx]		Operational [OpEx]			Total Cost in INR
	Effort (Cost)	Infrastructure Cost	License Cost	Maintenance Cost	Infrastructure Increment	
APIs	10 USD	2USD	N/A	1 USD	N/A	Rs 950
Content Server (Heroku)	5 USD	N/A	N/A	2 USD	N/A	Rs 510
Database Server (AWS)	6 USD	N/A	N/A	2 USD	N/A	Rs 580

Category	Cost in INR
One Time (CapEx)	Rs 1500
Operational (OpEx)	Rs 700

## 4.3.Risks

Risk ID (#)	Risk Description	Risk Category [Low/Medium/High]	Risk Appetite [ Accept/ Mitigate/ Transfer/Transfer]
R01	Glitches	Low	Transfer
R02	Change in Economic Policies	Medium	Transfer
R03	Server Breakdown	High	Mitigate

## 5. Implementation & Governance

### 5.1.Required Skills



Edit with WPS Office

Skills	More Info
UX Designer	Designing experience of user
Frontend Development	Design and Develop UI and frontend layer
Backend Development	Design Database and Develop Service / API
Testing	Develop Test Cases
Project Management	Project Planning, Scheduling, Executing, Monitoring and Controlling

## 5.2.Milestone

S.N o	Project Milestone	Description	Expected Date
1	Working Prototype of Front End	To create the working Front End of the platform, the whole user interface and structure of content on the page	Within one month
2	Working Prototype with backend	To obtain feedback from the user/client	Within 3-4 months
3	Full Working Model	Initial roll out	Within 6 months
4	Obtain a Brand Partner	Getting a major market player to help promote the platform	Within 3 months of initial roll out
5	Obtain a payment/banking partner	To obtain a bank as a partner to help with transaction and managing funds for a smoother monetisation process	Within 10-12 months of initial roll out

## 5.3.Change Management

- ▶ The decisions affecting the projects will only be accepted or rejected if it directly interferes with the aim or the normal workflow of the project.
- ▶ The project lead will be instrumental in determining which team member should take over what process when the team reaches that part of the project.



## 5.4. Performance Measurement

<N/A >

Return in timeline	Return in INR	Investment (INR)	Remaining Investment
Return on 1 <sup>st</sup> year	15000	10000	5000
Return on 2 <sup>nd</sup> year	25000	10000	
Return on 3 <sup>rd</sup> year	40000	12000	
Return on 4 <sup>th</sup> year	50000	15000	
Return on 5 <sup>th</sup> year	100000	20000	

<Indicate how the organization will measure performance for project implementation and benefits realization>

<Intangible benefits (brand value increase) to be documented>

## 6. Project Charter

### 6.1. Simplified Project Charter

Section	Details
Project Scope	To create a CMS system, where user can create and edit content and can access the content from anywhere in the world.
Project Schedule	A full working model within 6 months
Project Cost	Within Rs 3000 for the low budget working model.
Constraints	<ul style="list-style-type: none"><li>• Low Budget</li><li>• Low security</li><li>• Less APIs</li></ul>
ROI	N/A
Intangible Benefit	Increased knowledge of project management and



Edit with WPS Office

	database management
--	---------------------

## 6.2.Project Team Structure

### 6.2.1. Roles & Responsibilities

Project Role	Responsibilities	Assigned To
Project Steering Committee (Executive Sponsor)	To review and check the progress of the project while providing insightful information.	Review the progress and performance of project
Project Sponsor	Evaluate the actual progress of the project and compares it to what was initially planned	Assigned to look over the construction of program and report to the project manager
Project Manager	Responsible for planning, organizing, and directing the completion	Responsible for planning, organizing, and directing the completion
Developer	Ensure the smooth running of the application and provide timely updates to deal with any glitches.	Creating, maintaining and updating the application
Tester	Provide feedback about the working of different aspects of the application	Creating Test Plan and Develop Test Cases

## 6.3.Approval

Name	Designation	Role in Project	Signature
Ms.P.Mahalakshmi	Faculty -Incharge	Evaluator	



# **WEEK 2**

## **Project Methodology and Stakeholder Analysis**



Edit with WPS Office

## 1. Executive Summary

A methodology is a system of practices, techniques, procedures and rules used by those who work in a discipline. As project managers, there are many different ways to deliver projects. Broadly speaking, these ways are our project management models—applying different principles, themes, frameworks, processes and standards to help provide structure to the way we deliver projects.

A stakeholder is an individual, group, or organization that may affect be affected by or perceive itself to be affected by a decision, activity, or outcome of a project.

Put simply, if someone has any interest or is affected by your project, they are your stakeholder. Examples include the project manager, project sponsor, higher management, and team members.

After gaining all the knowledge regarding Project Methodologies, "Agile" Model will be the best suited and optimised model for our project, it will give us enough time to improve, learn and move forward, taking small steps.

## 2. Selection of Methodology

The meaning of Agile is swift or versatile. "**Agile process model**" refers to a software development approach based on iterative development. Agile methods break tasks into smaller iterations, or parts do not directly involve long term planning. The project scope and requirements are laid down at the beginning of the development process. Plans regarding the number of iterations, the duration and the scope of each iteration are clearly defined in advance.

Each iteration is considered as a short time "frame" in the Agile process model, which typically lasts from one to four weeks. The division of the entire project into smaller parts helps to minimize the project risk and to reduce the overall project delivery time requirements. Each iteration involves a team working through a full software development life cycle including planning, requirements analysis, design, coding, and testing before a working product is demonstrated to the client.

- Agile model is a combination of iterative and incremental process models with focus on process adaptability and customer satisfaction by rapid delivery of working software product.
- Agile Methods break the product into small incremental builds. These builds are provided in iterations. Each iteration typically lasts from about one to three weeks. Every iteration involves cross functional teams working simultaneously on various areas like –
  - Planning
  - Requirements Analysis
  - Design
  - Coding



- ▶ Unit Testing and
- ▶ Acceptance Testing.
- ▶ At the end of the iteration, a working product is displayed to the customer and important stakeholders.
- ▶ Iterative approach is taken and working software build is delivered after each iteration. Each build is incremental in terms of features; the final build holds all the features required by the customer.

## 2.1. Roles and Methods

- ▶ *Scrum is a methodology in the agile framework which works on transparency, keen observation of process, and adaptability. This method is used in product development.*
- ▶ *The aim of creating a scrum is to provide high performance within a small team, and assigning everyone an objective task to work on to provide effective results. Hence keeping this concept, the project methodology has been broken down into sprints.*
- ▶ *Sprints can go from a week to months, which is dependent on the complexity. Sprint logs consist of-*
  - ▶ Forecast
  - ▶ To-Do
  - ▶ In-Progress and
  - ▶ Done

*The structure is:-*

- *3 roles: Product Owner, Scrum Master and the Team.*
- *5 events: Sprint, Sprint Planning, Daily Scrum, Sprint Review and Sprint retrospective.*
- *3 artifacts: Product Backlog, Sprint Backlog and Increment.*

*Roles:-*

- ▶ *Product Owner*

*The Product Owner (PO) is a member of the Agile Team responsible for defining Stories and prioritizing the Team Backlog to streamline the execution of program priorities while maintaining the conceptual and technical integrity of the Features or*



Edit with WPS Office

*components for the team.*

► **Scrum Master**

*The scrum master is a facilitator of the team. He is an expert in agile methodologies and works in accordance with the principles. The scrum master manages the information collaboration within the different departments in a project team. Any changes in the project stage will be quickly brought in by the scrum master. The scrum master will have the responsibility of providing high yield and to make sure everything falls in the right place using the scrum values and principles.*

► **Developers**

*Agile developers focus on sustainable development—not heroics. Sustainability is about good estimation, effective branching strategies for managing code, automated testing to protect quality, and continuous deployment to get fast feedback from users. Adopting sustainable development practices requires a discipline most of us aspire to—but often struggle to realize—as individuals.*

### *Events:* -

► **Sprint Planning:**

► This planning stage takes place on the first day of the sprint. The team is involved in the backlog work that they should work on during sprints. The Team decides what to work on for the current period. The team consists of a Scrum Master, the developer's team and the product owner. Only after the completion of the previous sprint review and retrospection, the next new sprint is taken place. Based on the previous sprint discussion, new sprint planning is considered.

► **Daily Scrum:**

► The scrum team meets for a quick 15 minutes every day of the Sprint to inspect progress and decide what to work on for the next 24 hours. The daily scrum is held every day in sprints. The intention here is to create a planned work for an effective understanding and communication between the teams.

► **Sprint Review:**



- This is the stage where the development team present to the product owner. The Team collaborates about what was done and adapts the Backlog as needed. They present the work which is “done.” This is the end of the sprint. At an ideal stage, the sprint review meeting can go for a duration of one hour per week.
- **Sprint Retrospective:**
  - Sprint retrospective stage of the meeting is headed by the scrum master, which is done after the sprint review. The Team discusses how to improve in the next sprint. This means it gives the look back on past events and ways to improve. It gives a chance to the team to give feedback to the management on the progress of the project. This creates a plan for improvement in the next sprint.

### *Artifacts:* -

- **Product Backlog:**
  - The product backlog is the detailed document of the scrum team. It's the final document that is referred for any details regarding the product. The product backlog should contain priorities. This helps in delivering as per priority. The product backlog should not contain any detailed statement, it's a top requirements document. Product backlog must be dynamic, it should change as per when the requirement gets into detailed work. As there are changes per requirement, the document changes and a new requirements area are added upon.
- **Sprint Backlog:**
  - Making decisions in a group: Each and every state should be decided by the whole team and grouped.
  - Organize task: The Scrum team should not need the assigned task from the Scrum master. They are self-organized to pick and work on tasks.
- **Product Increments:**
  - Product Increments, in general, is a sum of all product backlog committed during the sprint. This is generally a piece of software collaborating across the organization, creating transparency. This can also be in the type of task boards or charts.

## 2.2. Agile Development and Testing[optional]

- **What is Agile Testing?**
- AGILE TESTING is a testing practice that follows the rules and principles of agile



Edit with WPS Office

software development. Unlike the Waterfall method, Agile Testing can begin at the start of the project with continuous integration between development and testing. Agile Testing methodology is not sequential (in the sense it's executed only after coding phase) but continuous.

- **Agile Test Plan**
- **Agile test plan** includes types of testing done in that iteration like test data requirements, infrastructure, test environments, and test results. Unlike the waterfall model, in an agile model, a test plan is written and updated for every release. Typical test plans in agile includes
  1. Testing Scope
  2. New functionalities which are being tested
  3. Level or Types of testing based on the feature's complexity
  4. Load and Performance Testing
  5. Infrastructure Consideration
  6. Mitigation or Risks Plan
  7. Resourcing
  8. Deliverables and Milestones

## 3. Stakeholder Management

### 3.1. Identification of Stakeholders

Stakeholder management is the process of maintaining good relationships with the people who have most impact on your work. Communicating with each one in the right way can play a vital part in keeping them "on board."

*The way that you manage your stakeholders depends on:*

- *The size and complexity of your project. You could assess this by comparing it to past projects, by examining the project's milestones, by the number of resources required, or by the time allocated to it, for example.*
- *The amount of help you need to achieve the results you want. This could include sponsorship, advice and expert input, physical resources, reviews of material to increase quality, and so on.*
- *The time you have available to communicate. You need to consider how to manage the time you expect to spend on communication, particularly if your project requires a lot of stakeholder input. It is often better to allocate more time*
- *to communicating with stakeholders, rather than trying to "get by" without all the help or input that you need.*

**Stakeholders in our Project: -**



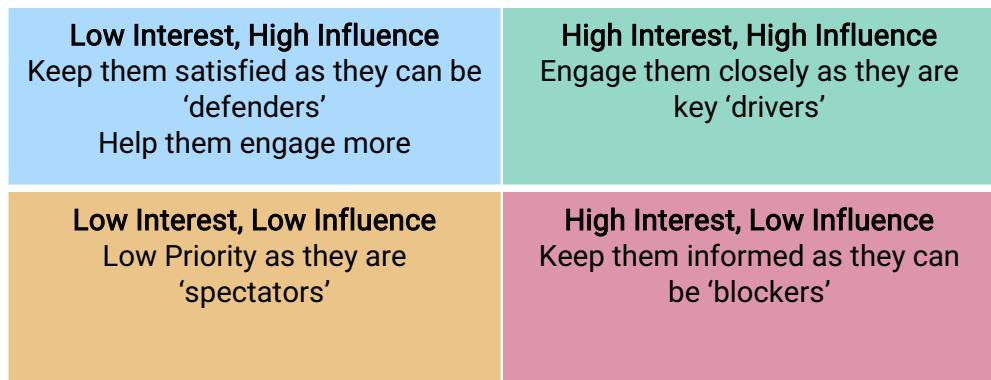
Edit with WPS Office

Name	Designation	Role in project
Aditya Dixit	CEO	Execution of project
Abhishek Hindocha	CFO	Cost Approval and administrative work
Sai Pranav	CTO	Handling the technology and engineering department
Aditya Dixit, Abhishek Hindocha, Sai Pranav	Developers/Department Heads	Handling various verticals
Sarthak Gupta, Kushagra Gupta	Employees	Carrying out the designated work
Bank and hedge fund	Credit Provider	Providing Credit
Vaibhav Singh	Investor	Providing Capital
Kinjalk Parth	Tester	Testing the various functionalities of the application
AWS	Server Provider	Provide Servers for application use
Renu Jadhav, Sudhanwa Bandi,etc (Customers) Consultant	Users External Advice	Validating functionalities and making beneficial use of the software. Guidance for optimization of plan And resources



### 3.2. Interest and Influence matrix

Interest	Influence
High	High
Low	Low
Low	High
High	Low



Stakeholder Name	Activity / Area / Phase	Interest	Influence	Priority (High / Medium/Low)
Aditya Dixit (CEO) Abhishek Hindocha(CFO) Pranav Sai (CTO)	Subscription using Web App	High	High	1
Finance Account Receivable consultant	Multiple Currency Payment	High	Low	3
Sponsor (BANK)	Provide new market and capital for expansion	Low	High	2
Team Members (Developers)	Retain and expand skill level	Low	Low	3
Project Manager	Responsible for making sure that all processes are being carried out smoothly	Low	High	2



Edit with WPS Office

Consultant	Advice	low	Low	3
Server Provider	Providing cloud services	low	Low	3

### 3.3. Communication Plan for Stakeholders

- ***Summarize Each Stakeholder's Status***

***Key Interests and Issues:*** - add each stakeholder's level of influence and area of interest in your task or project. Then, add your assessment of where they stand in respect to it: "Advocate," "Supporter," "Neutral," "Critic," or "Blocker."

- ***Decide What You Want from Each Stakeholder***

***Look at your list of stakeholders and think about the level of support that you want from each one: is it High, Medium or Low?***

***And, what role would you like each one to play in your project (if any)? Will you need full-time technical support, for instance, or just "ad hoc" advice?***

***Try to be as detailed as possible about what you want from your stakeholders. If there are specific actions that you need them to take to move the project forward, add that in your plan.***

- ***Identify Your Key Message to Each Stakeholder***

***Next, think about what you need to say to persuade your stakeholders to support you and to engage***

***with your project. What's in it for them? Highlight the benefits that your project will bring to the organization or the individuals concerned, and focus on key performance drivers, such as increasing profitability or delivering real improvements.***

- ***Identify Your Stakeholder Communication Approach***

***Focus on the most important stakeholders first, and the less crucial ones later (refer back to your Power/Interest Grid, if you need to). Devise a plan that communicates with them as simply and efficiently as possible, with just the right amount of appropriate information.***

***Consider how often each stakeholder will want to receive updates, and in what form. Would they prefer email or face-to-face meetings, or visual update?***

- ***Implement Your Stakeholder Management Plan***

***Once you have prepared your plan, you can start to implement it!***

***Aim to make Stakeholder Management an integral part of your project, rather than***



*treating it as a side task. As with all plans, it will be easier to implement if you break it down into a series of small, achievable steps which you action one by one.*



Edit with WPS Office

# **WEEK 3**

# **Project Scope and Requirements Management**



Edit with WPS Office

## 1. Executive Summary

This document produces an overview of the project scope of a Content Management system. The Project Scope deals with the deliverables produced by the end of the development stage. There are two types of scopes: in and out. Additionally, it also covers the major functions of the product developed.

A detailed analysis of the requirements to develop the product is tabulated in the form of functional and non-functional requirements with a unique code associated with each requirement. Team members are allotted to handle a particular requirement according to domain.

A comprehensive user story for each and every requirement from the user's perspective is analysed.

In addition, an acceptance-criteria is devised to integrate each of these requirements into the Project.

## 2. Project Scope

Project scope is an outline of all the deliverables covered by this process. During development, there are certain functionalities that are covered during the development phase, whereas others are not covered. These factors are referred to as Activities in Scope and Activities out of Scope.

During the project, this documentation helps the project team remain focused and on task. The scope statement also provides the team with guidelines for making decisions about change requests during the project. Note that a project's scope statement should not be confused with its charter; a project's charter simply documents that the project exists. Out of scope activities are identified to reduce ambiguity.

S.No	Activities In Scope	Activities Out of Scope
1.	Provide an option to the user for a quick enquiry about the platform	The company staff needs to reply to the queries.
2.	The website will be available 24X7 to satisfy requirements of users.	Some services may be unavailable during maintenance/backup windows.
3.	Online help will be available that guides the content contributors and approvers through procedures that they are able to perform.	A detailed list of help topics will be developed and documented by the company staff as part of this effort.
4.	Provide a new space to the users so that they can express	The Database server(AWS) needs to function properly without any glitches



	themselves and create the content they like	
5.	Provide the users an option to monetize their content through ads provided by Google Ads	Google Ads needs to pay the users.
6.	Provide credentials after purchase of subscription	Registration through purpose of subscription
7.	Allow registered payment gateways to make payment on the software	Denial of obscure/non-registered gateways

## 2.1.In Scope

- 1) Provide an option to the user for a quick enquiry about the platform
- 2) The website will be available 24X7 to satisfy requirements of users
- 3) Online help will be available that guides the content contributors and approvers through procedures that they are able to perform.
- 4) Provide a new space to the users so that they can express themselves and create the content they like
- 5) Provide the users an option to monetize their content through ads provided by Google Ads
- 6) Provide credentials after purchase of subscription
- 7) Allow registered payment gateways to make payment on the software

## 2.2.Out of Scope

<Define what is not covered in the deliverable and doesn't address as part of the outcome >

- 1) The company staff needs to reply to the queries.
- 2) Some services may be unavailable during maintenance/backup windows.
- 3) A detailed list of help topics will be developed and documented by the company staff as part of this effort.
- 4) The Database server(AWS) needs to function properly without any glitches
- 5) Google Ads needs to pay the users.
- 6) Registration through purpose of subscription
- 7) Denial of obscure/non-registered gateways

## 3. Epics [Major Functions]

<List down the modules of the product / application >

Epic (#)	Epic Description
E1	Multi-currency Payment Integration
E2	Registration



E3	Email Notifier
E4	Entity Register
E5	Localization
E6	Search Engine
E7	Redundancy
E8	Performance monitoring

## 4. Requirements

### 4.1. Functional Requirements

Functional Requirements can also be expressed in the form of “user story” which is the smallest unit of work in an agile framework. It’s an end goal, not a feature, expressed from the software user’s perspective.

Requirement (#)	Requirement Specification	Department	Name of Business User	Status
E1FR1	Allowing the use of multiple payment gateways for payment	Functional Coding Unit	Coder	Pending
E2FR1	Allowing the customer to enter their details and register to the platform	Interface Coding Unit	Developer	Pending
E2FR2	Allowing the customer to change their personal details	Functional Coding Unit	Coder	Pending
E3FR1	A Verification email is sent to user whenever he/she registers for the first time on some software system.	Functional Coding Unit	Coder	Pending
E3FR2	Provide regular email updates to the customer to inform them of their order location	Support Team	Customer support executive	Pending
E4FR1	User can add or edit the content from their account dynamically	Functional Coding Unit	Coder	Pending
E5FR1	Ability to recommend the customer the	Algorithms Unit	Designer	Pending



	content they like according to their interests			
E6FR1	As a customer, being able to find the required items with a quick search	Algorithms Unit	Designer	Pending
E7FR1	The web will have adequate redundancies built-in to the hardware and software architecture to ensure that maintenance requirements or unexpected failures will not take the system completely off-line.	Functional Coding Unit	Coder	Pending
E8FR1	Monitoring requirements will be developed to ensure uptime and availability of service and analyze server health, server traffic, individual page requests, etc...	Functional Coding Unit	Coder	Pending

## 4.2. Non-Functional Requirements

Requirement (#)	Category of NFR	Requirement Specification	Department	Name of Business User	Status
NFR1	Speed	All pages should load within 3 seconds	Algorithms Unit	Designer	Pending
NFR2	Performance	Search should bring the results less than 7 seconds	Algorithms Unit	Designer	Pending
NFR3	Availability	Application should be available for 24x7	Support team	Technical Support Executive	Pending
NFR4	Scalability	Registration Service should scale to serve 1000 request per	Algorithms Unit	Designer	Pending



		second over 5 minutes timespan			
NFR5	Confidentiality	User's privacy must be respected by encrypting the user details	Functional Coding Unit	Coder	Pending
NFR6	Compliance	Application should be able to comply with the existing software standards in order to make it free of bugs, weakness and flaws	Interface coding Unit	Developer	Pending
E1NFR1	Speed	OTP should be sent for verification within 60 seconds	Functional Coding Unit	Developer	
E1NFR2	Security	Payment credentials must be protected using various APIs	Functional Coding Unit	Developer	Pending
E1NFR3	Flexibility	The application must support all payment gateways	Interface coding Unit	Developer	Pending
E4NFR1	Extensibility	The application should be extensible to allow additional modules, extensions and changes in technology	Support team	Technical Support Executive	Pending
E4NFR2	Interoperability	Application should be able to transfer data between the content creators and the content viewers	Interface coding Unit	Designer	Pending



E7NFR1	Reliability	Application should be able to run 24X7 smoothly without any unexpected failures	Support team	Technical Support Executive	Pending
--------	-------------	---	--------------	-----------------------------	---------

#### 4.3.Infrastructure Requirements

Requirement (#)	Requirement Specification	Department	Name of Business User / Project Team Member	Status
IR1	Development Machine with 6 GB Ram and 4 Cores	Management	Project Manager	Procurement
IR2	Code Repository	Development Team	Coder	Completed
IR3	AWS S3 Bucket	Planning	Project Manager	Procurement
IR4	IDE – VSCode	Development Team	Coder	Procurement
IR5	Database – MongoDB Heroku , AWS	Testing Team	Tester	Procurement
IR6	UI – React Native	Development Team	Designer	Procurement
IR7	APIs	Development Team	Designer	Procurement

#### 4.4.Requirement definition in Agile [Optional ... Use according to methodology chosen by student]

User story is the smallest unit of work in an agile framework. It's an end goal, not a feature, expressed from the software user's perspective.

User Story	Acceptance Criteria	Size of User Story
As customer, I can view all available features related to	View all the available features in the platform	Big



content management . So, I can use a certain feature I like and start creating content.	Allow to select any feature	
As a customer, I can avail multiple services according to my preference. These services will help me deliver the content which I desire to deliver	Avail multiple services for usage	Big
As a customer, I can use multiple payment options to request my subscription orders	Any nationally registered method	Big
As a customer , I can change my preferences and accordingly avail the services which I require	Flexibility to use any service as per requirement	Big
AS a customer, I can view other user's content and can share, comment and like their content	Features such as like, share ,comment and save are available	Medium
As a customer , I can view the engagement on my profile and can view stats such as number of views, number of comments etc.	Detailed analysis of the user's content is being provided	Big
As a customer , I can customize the way my profile looks , in order to attract more viewers and increase engagement	Any mode of display provided by the UI/UX of the application	Small
As a customer , I can view my history of my content produced	Past content created should be present	Smaill
As a customer , I can cancel my membership whenever I	Verification of cancellation by providing personal details	Medium



wish



Edit with WPS Office

# **WEEK 4**

# **Project Scope and Requirements Management**



Edit with WPS Office

## 1. Executive Summary

Effort and Cost estimation is an important part of software development both from the academic and industrial viewpoint. In this report we will go over the project management plan followed by the effort and cost estimation process. For any new software project, it is necessary to know how much it will cost to develop and how much development time will it take.

## 2. Project Management Plan

Focus Area	Details
Integration Management	Governance Framework Project Team Structure Roles & Responsibilities of Team Change Management (Change Control, Issue Management) Project Closure
Scope Management	Scope Statement Requirement Management (Gathering, Control, Assumption, Constraint Stakeholder) Define Deliverable Requirement Change Control Activities and Sub-Tasks
Schedule Management	Define Milestones and create a proper structured schedule to cross those milestones within the allotted time.
Cost Management	Estimate Effort Assign Team Budget Control
Quality Management	Quality Assurance: Quality assurance will be managed including governance, roles and responsibilities, tools and techniques and reporting Quality Control: Specify the mechanisms to be used to measure and control the quality of the work products
Resource Management	Estimate and Manage the need People: People with expertise in frontend and backend development Finance: Upto Rs.3,000 Physical: Facilities, IT Infrastructure
Stakeholder	Identifying, Analyzing, Engaging Stakeholders



Communication Management	Determine communication requirements, roles and responsibilities, tools and techniques. [Type of Communication, Schedule, Mechanism Recipient]
Risk Management	Identifying, analysing, and prioritizing project risks
Procurement Management	Adhering to organization procurement process

### 3. Estimation

#### 3.1. Effort and Cost Estimation

WBS	Activity	Activity Description	Sub-Task	Sub-Task Description	Effort (in hours)	Cost in INR
E1FR1	E1R1A1	Design the user screen	E1R1A1T1	Confirm the user requirements (acceptance criteria)	3	1500
			E1R1A1T2	Intuitive Design for the user interface	5	2500
E2R2	E2R2A2	Identify Data Source for displaying units of Energy Consumption	E2R2A2T1	Go through Interface contract (Application Data Exchange) documents	5	
				Document		
E3R3	E3R3A2					

Effort (hr)	Cost (INR)
1	500



### 3.2. Infrastructure/Resource Cost [CapEx]

Infrastructure Requirement	Qty	Cost per qty	Cost per item
Server/Storage	20	500	10000

### 3.3. Maintenance and Support Cost [OpEx]

Category	Details	Qty	Cost per qty per annum	Cost per item
People	Network, System, Middleware and DB admin  Developer , Support Consultant	3	2,000	6,000
License	Operating System Database Middleware IDE	4	1,000	4,000
Infrastructures	Server, Storage and Network	20	500	10,000

## 4. Project Team Formation

### 4.1. Identification Team members

Name	Role	Responsibilities
Content Creators, Bloggers	Key Business User (Product Owner)	Provide clear business and user requirements
Aditya Dixit	Project Manager	Manage the project
Abhishek Hindocha	Business Analyst	Discuss and Document Requirements
Sai Pranav	Technical Lead	Design the end-to-end architecture
Aditya Dixit	UX Designer	Design the user experience
Sai Pranav	Frontend Developer	Develop user interface
Abhishek Hindocha	Backend Developer	Design, Develop and Unit Test Services/API/DB
Aditya Dixit, Abhishek	Tester	Define Test Cases and Perform



Hindocha, Sai Pranav		Testing
----------------------	--	---------

## 5. Responsibility Assignment Matrix

RACI Matrix		Team Members			
Activity		Abhishek Hindocha(BA)	Sai Pranav (Developer)	Aditya Dixit (Project Manager)	Key Business User
User Requirement Documentation	A	C/I	I	R	
Application Design Process	R	R	C/I	I	
Creating working prototype of actual application	C/I	R/A	C/A	I	
Testing	I	R	A	C	
Create a business model	R	I	C/A	C	

A	Accountable
R	Responsible
C	Consult
I	Inform



# **WEEK 5**

# **WBS and Risk Management Plan**



Edit with WPS Office

## 1. Executive Summary

In this document, we will discuss the WBS (Work breakdown structure) and Risk Management of Content Management System (CMS).

Work breakdown structure or WBS in project management is a method for getting a complex, multi-step project done. It's a way to divide and conquer large projects so you can get things done faster and more efficiently. The goal of a WBS is to make a large project more manageable. Breaking it down into smaller chunks means work can be done simultaneously by different team members, leading to better team productivity and easier project management.

Project risk management is the process of identifying, analysing and responding to any risk that arises over the life cycle of a project to help the project remain on track and meet its goal. Risk management isn't reactive only; it should be part of the planning process to figure out risk that might happen in the project and how to control that risk if it in fact occurs.

## 2. WBS With Project Schedule

Module (#)	Activity (#)	Sub-Task(#)	Assignee(s)	Planned Start Date	Planned End Date	Actual Start Date	Actual End Date	Status
1.	Designing a sketch of Web UI using <a href="#">Balsamiq Cloud</a>	Finalise a colour-scheme	Colour-theory expert (Outsourcing)	15-03-2021	17-03-2021			Pending
		Finalise a font scheme	Font-theory expert (Outsourcing)					
		Finalise a wireframe for a website	Designer Team, CTO (Pranav Sai) and CEO (Aditya Dixit)					
2.	Creating the UI using front end technologies		Front-End Developers	18-03-2021	7-04-2021			Pending
3.	Developing Back-End for the Project	Server-Side Coding	Back-end team and CTO	8-04-2021	10-05-2021			Pending
		Security Patches						



4.	Outsourcing security and IT solutions		CTO	11-05-2021	15-05-2021			Pending
5.	Optimising the Website code for faster response	Optimising the front end	Front end Developers	11-05-2021	15-05-2021			Pending
		Optimising the back end	Back-end Developers					
6.	Purchasing Servers for Data Management and web application Hosting		CEO (Aditya Dixit), CFO (Abhishek Hindocha)	17-05-2021	20-05-2021			Pending
7.	Testing	Feedback for Front end from Normal People	PR Team and Design Team	21-05-2021	30-05-2021			Pending
		Hiring professional tester for Back End testing and encryption lag	CTO (Pranav Sai) and CFO (Abhishek Hindocha)					
8.	Final tweaks according to testing results	Colours and Themes are changed for Front end	CEO (Aditya Dixit), Developers, Testers	1-06-2021	15-06-2021			Pending
		Back End code is optimised						
9.	Final Deployment of Application for public		CEO, CFO, CTO	20-06-2021	20-06-2021			Pending



### 3. Risk Identification

- Structured discussions between developers and department heads for exchange of ideas on how to manage the risks.
- Hierarchical flow of information to maintain integrity of the structure of the management and to ensure proper handling of risks.
- Frequent assessment of risks to ensure quick and abject responses to arising risks.
- Analysing risk using modified SWOT (Strength /Weakness /Opportunity /Threat) analysis with the options of Mitigate, Accept, Avoid and Transfer as the driving qualities.
- Working backwards using this method from the grassroot level going up to ensure proper time efficiency of queue management.

#### 3.1.List Register

Risk ID (#)	Risk Description	Impact Description
R01	Data Hacking by third party hackers	Low chance of getting hacked but mitigable risk as we can shift the data to offline databases.
R02	Server breakdown due to load	Shifting data to local memory-based servers instead of cloud-based servers.
R03	High Competition from already existing companies like Wix, blogger etc	Since there are already huge companies which are providing the same services, Our company can face a lot of competition.
R04	Change in Subscription Policies of Google Ads	Might face issues on the platform due to changes in Google Ads policies.



	and other payment portals	
R05	System Bugs	Minor Bugs found will be eliminated after each iterative update and Major Bugs can be eliminated after informing the company.
R06	Major changes in third-party software and APIs	Due to the changes and updation in third-party software and APIs, changes will be required frequently in the code, hence increasing the chances of bugs

### 3.2.Managing Risk

Risk ID (#)	Status [Open / Closed]	Risk Appetite [ Accept/ Mitigate/ Transfer/Avoid]	Action	Action Owner	Target Date	Remarks
R01	Open	Avoid	Data Hacking by third party hackers and transferring data to local servers	Abhishek Hindocha (CFO)	2-3 months after initial policy roll-out	Significant (3)
R02	Open	Transfer	Cloud based Servers.	Pranav Sai (CTO)	Situation dependent	Moderate (2)
R03	Closed	Mitigate	High Competition from already existing companies like Wix, blogger etc	Aditya Dixit (CEO)	Situation Dependent	Moderate (2)
R04	Open	Transfer	Change in Subscription Policies of	Aditya Dixit (CEO)	Same day as the problem	Moderate (1)



			Google Ads		arises, to ensure continuity	
R05	Closed	Mitigate	Iterative update	Pranav Sai (CTO)	-	Significant (2)
R06	Closed	Mitigate	Frequent Changes in third-party applications	Pranav Sai (CTO)		Significant (1)

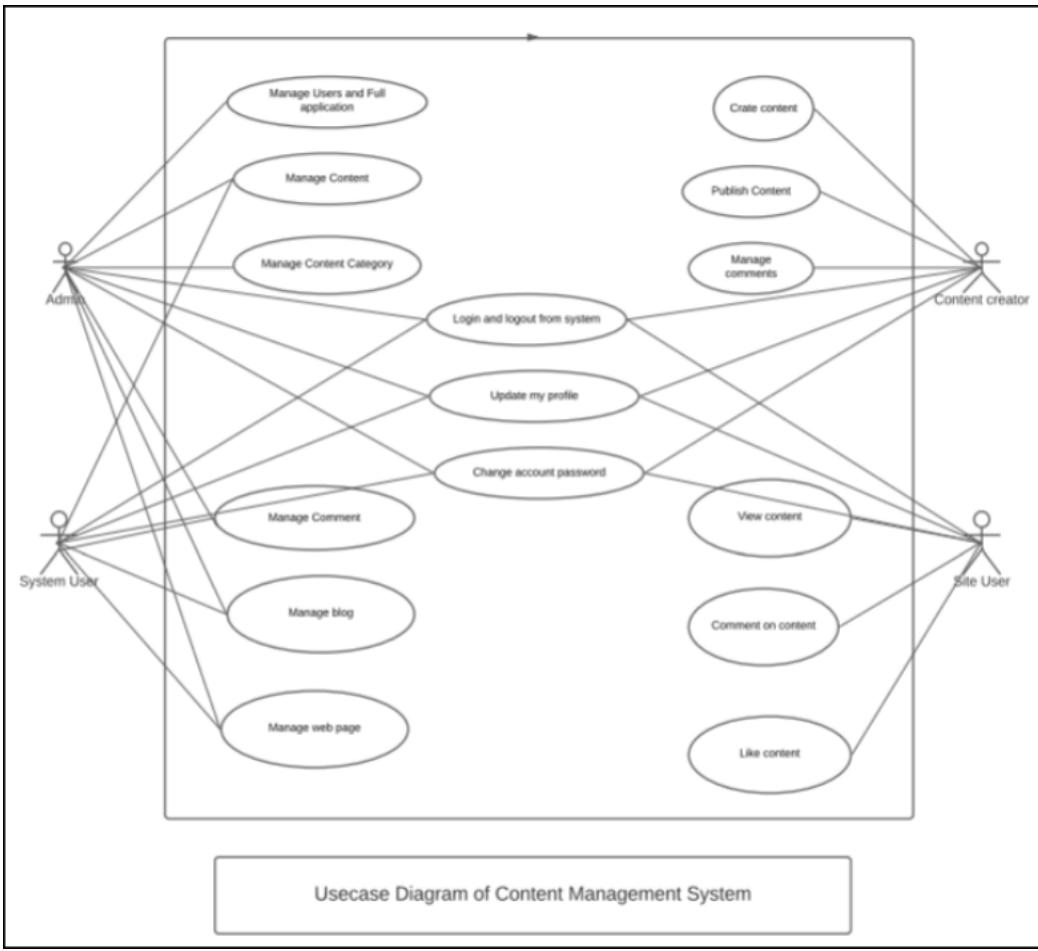
# WEEK 6

## Architecture Diagrams



Edit with WPS Office

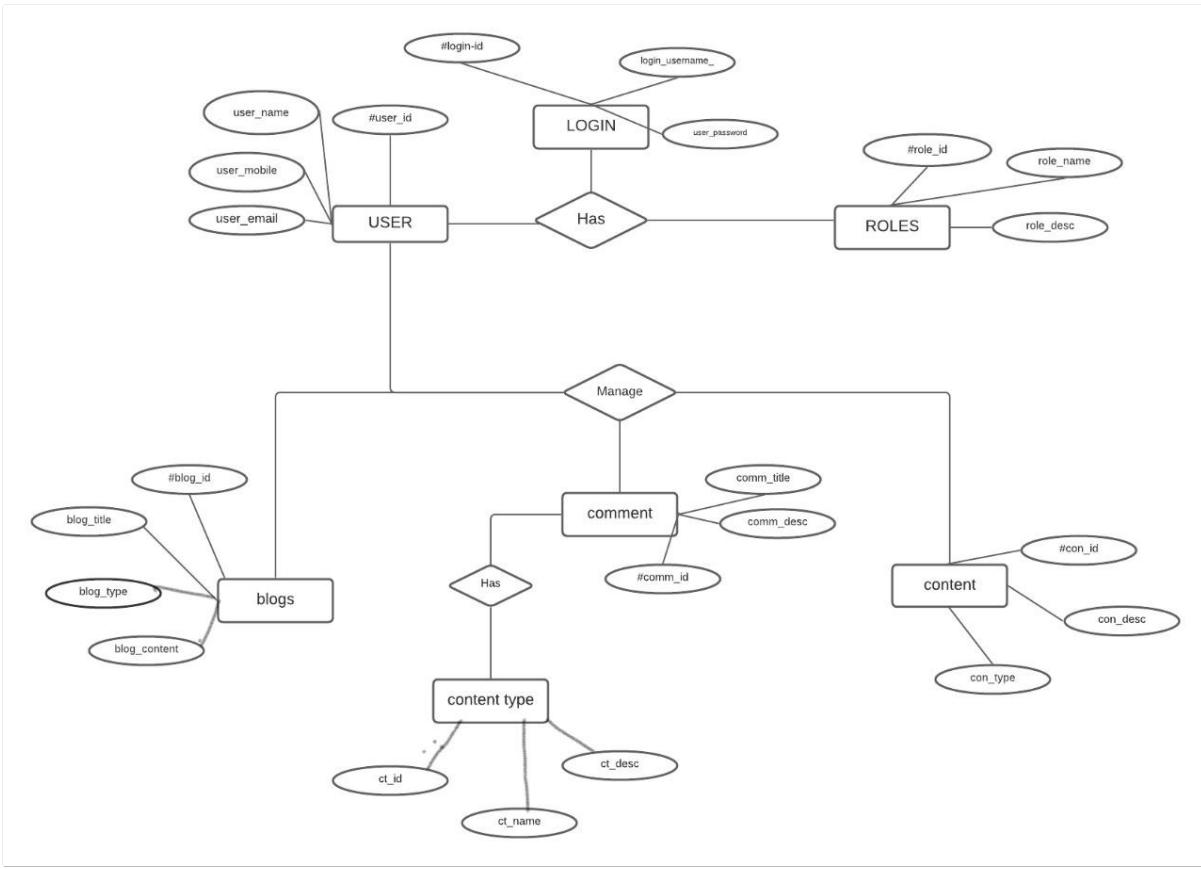
## USE CASE DIAGRAM



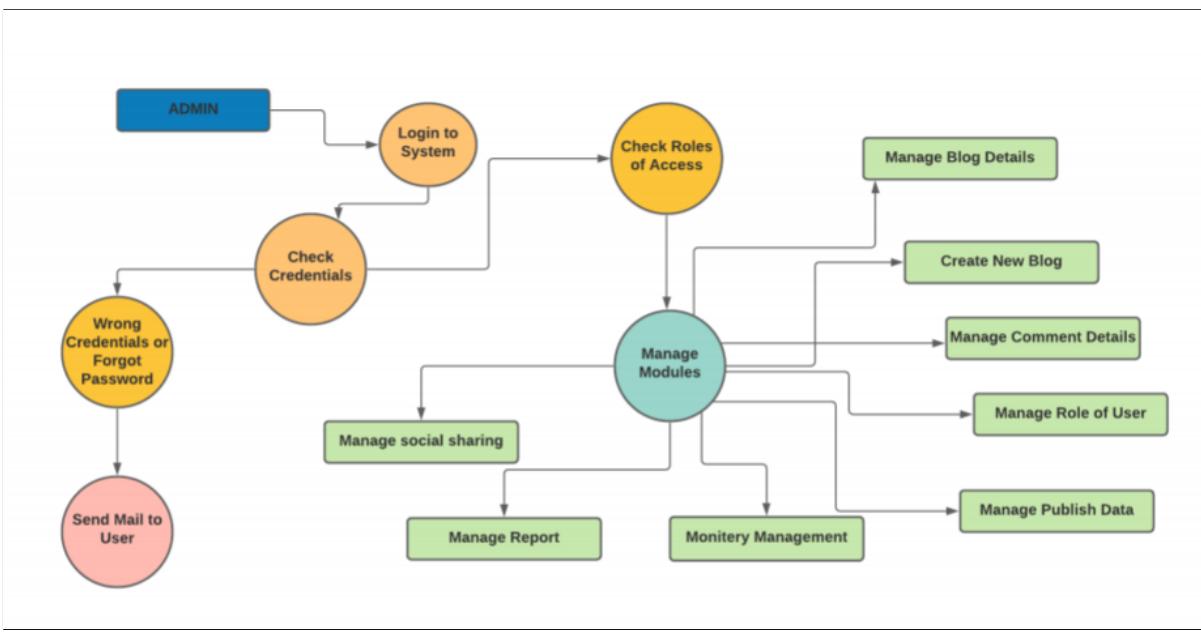
## ER DIAGRAM



Edit with WPS Office

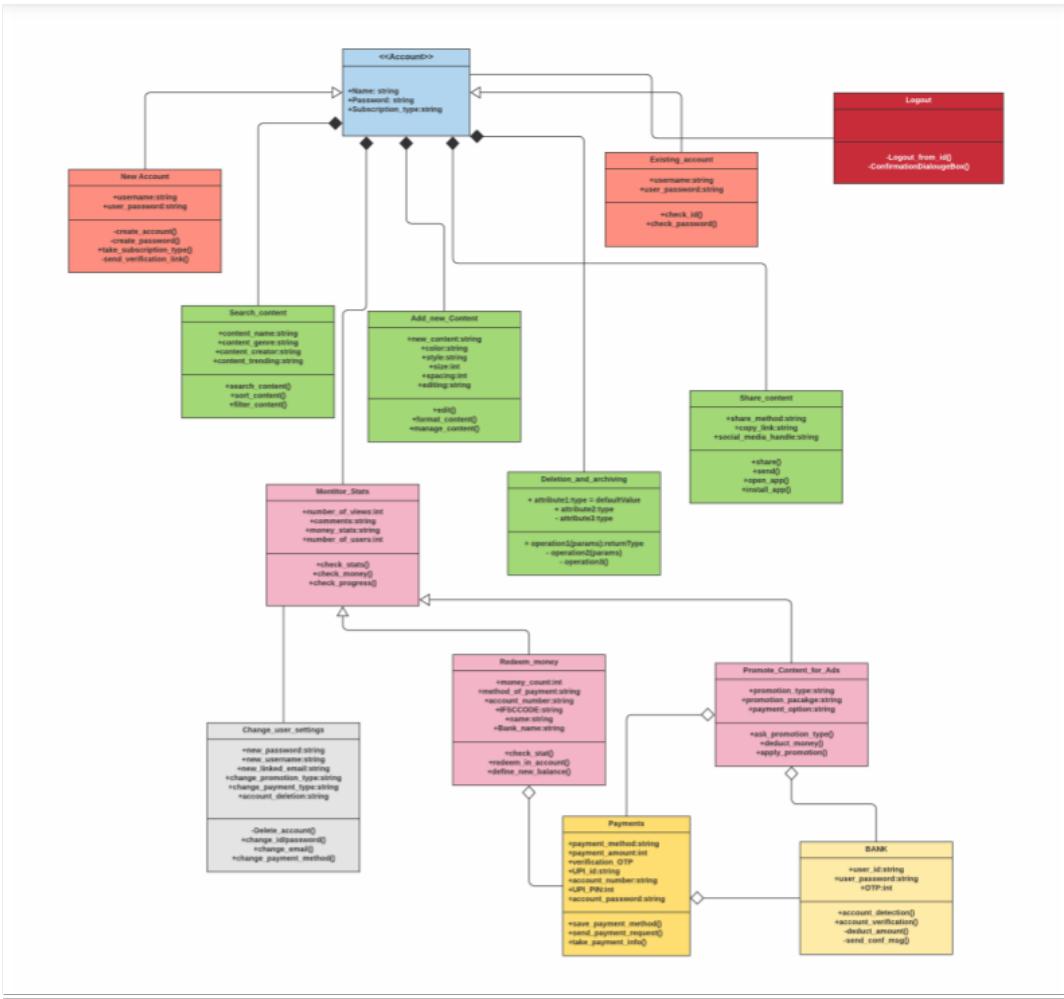


## DFD DIAGRAM

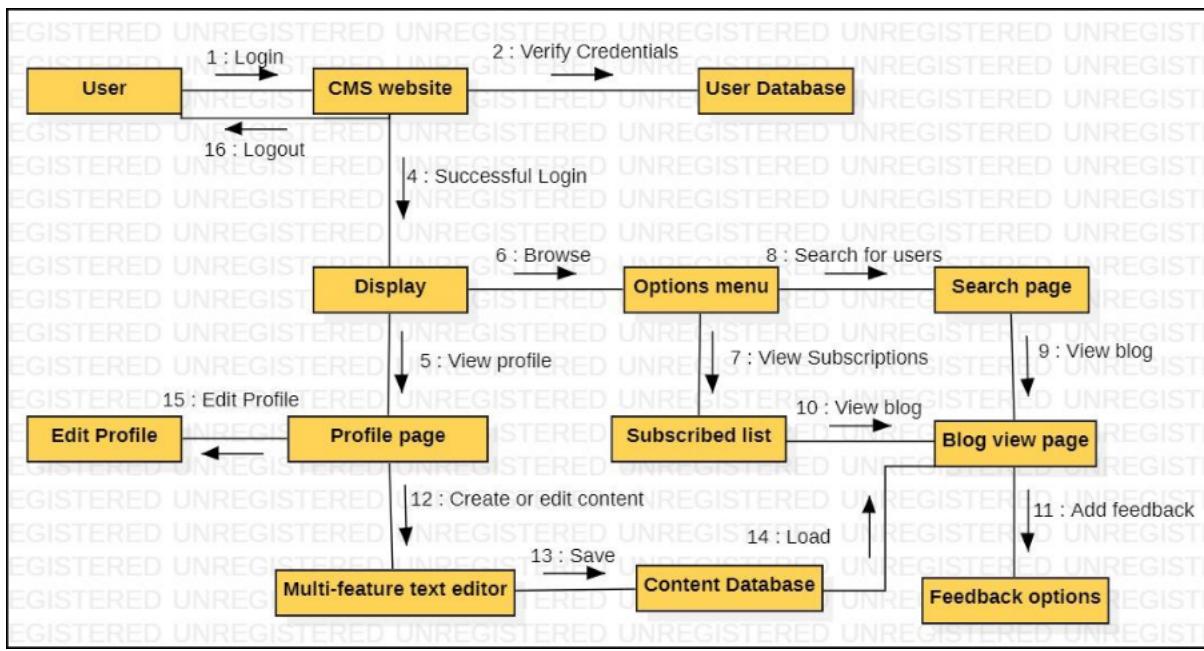


Edit with WPS Office

# Class Diagram



## Collaboration Diagram



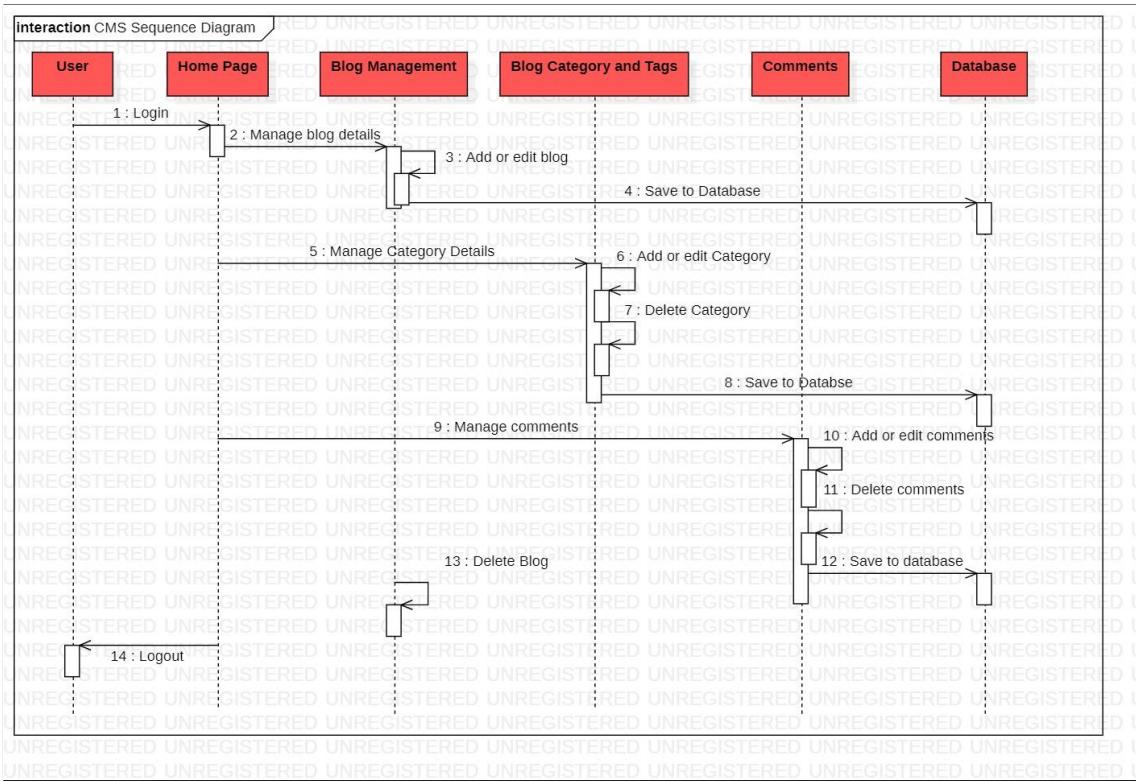
## **WEEK 7**

# **Architecture Diagrams and Front End**



Edit with WPS Office

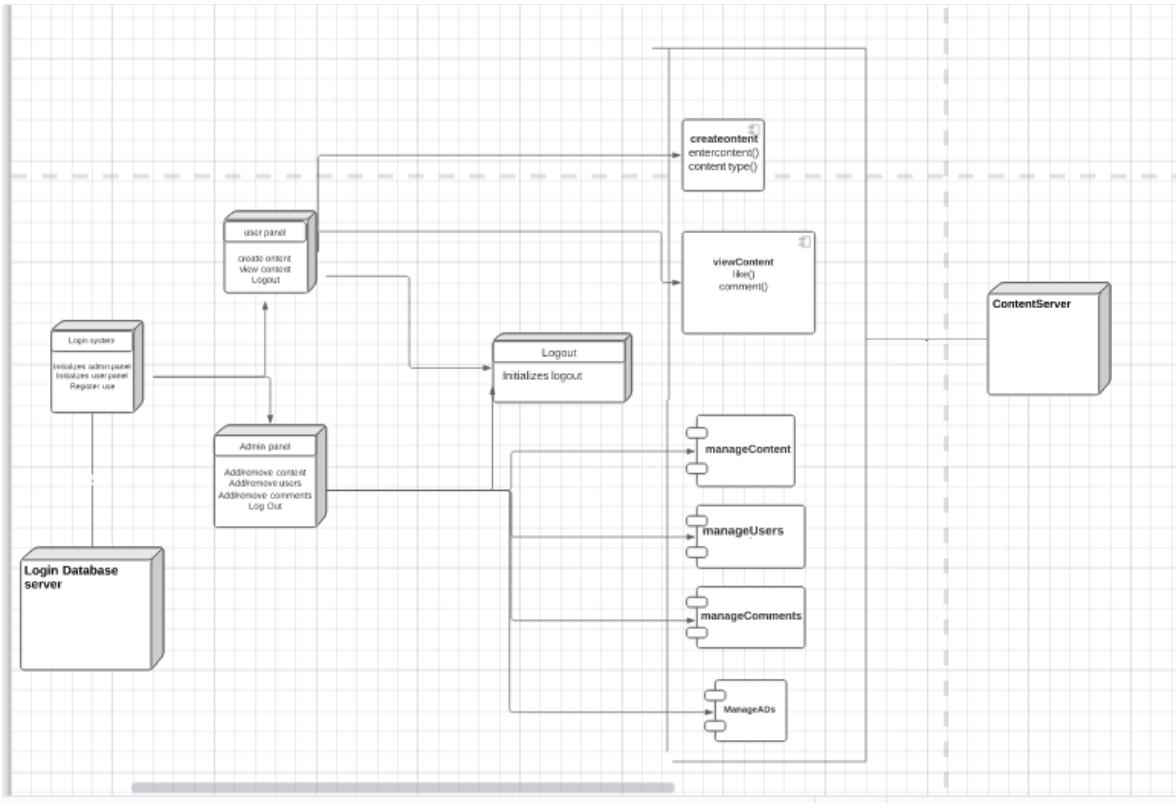
## SEQUENCE DIAGRAM



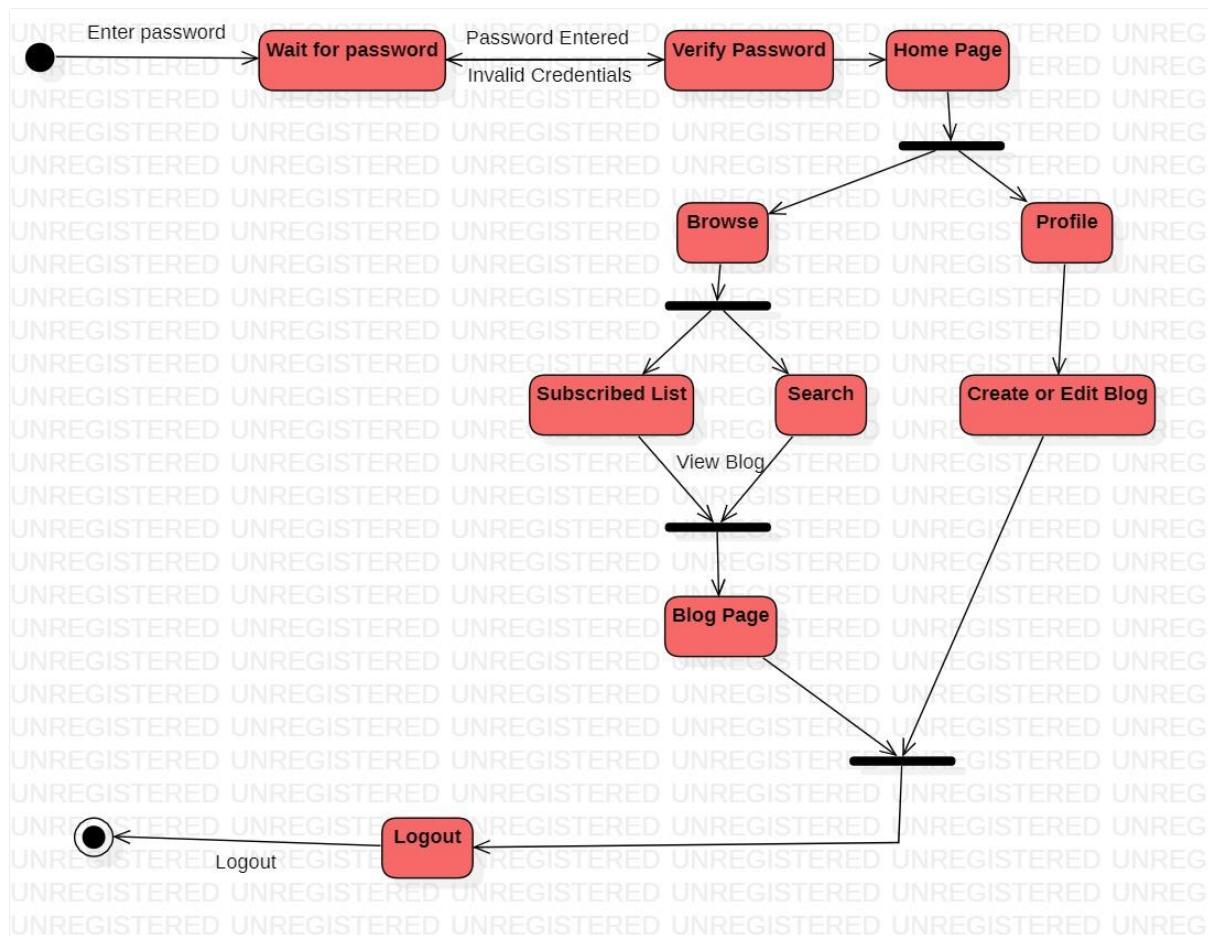
## DEPLOYMENT DIAGRAM



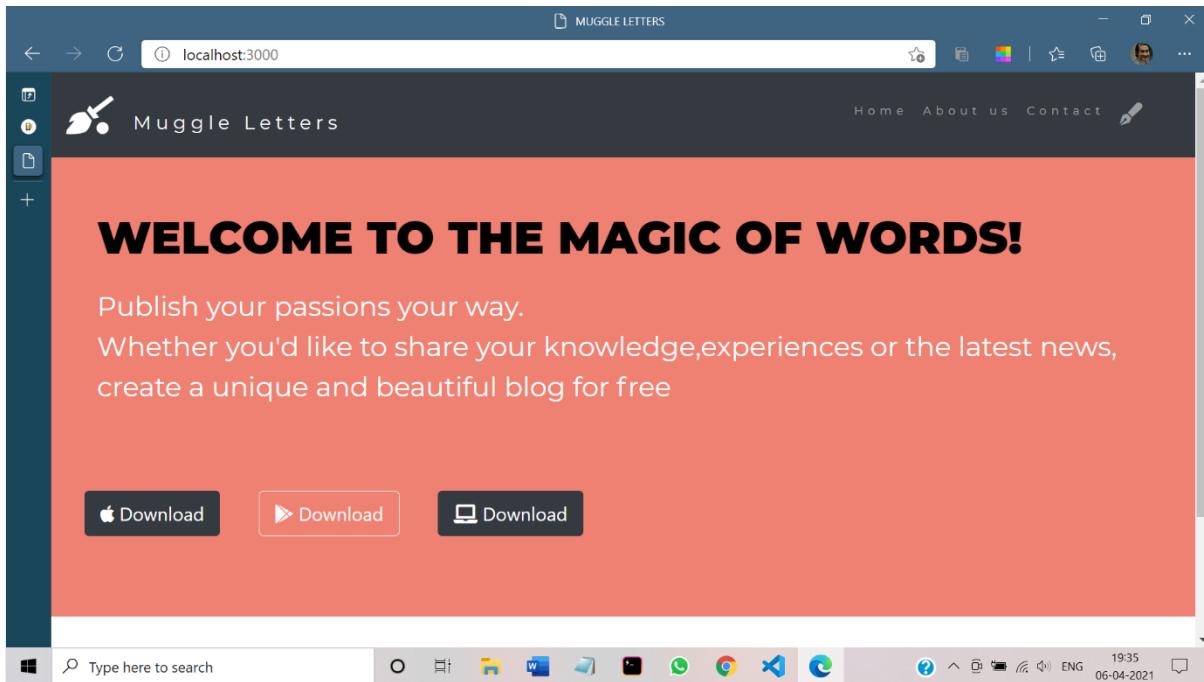
Edit with WPS Office



STATE DIAGRAM



## FRONT END OF WEBSITE: -



Edit with WPS Office

MUGGLE LETTERS

localhost:3000/about

# About-us



### Easy to use.

Give your blog the perfect home. Get a free blogspot.com domain or buy a custom domain with just a few clicks.



### Earn money

Get paid for your hard work. Google AdSense can automatically display relevant targeted ads on your blog



### Know your audience

Find out which posts are a hit with Blogger's built-in analytics. You'll see where your audience is coming from and what they're interested in.

Type here to search

19:36 ENG 06-04-2021

MUGGLE LETTERS

localhost:3000/about

interested in.

"Its feature to publish and unpublish articles and blogs anytime has helped me alot."



Kinjalk, Mumbai

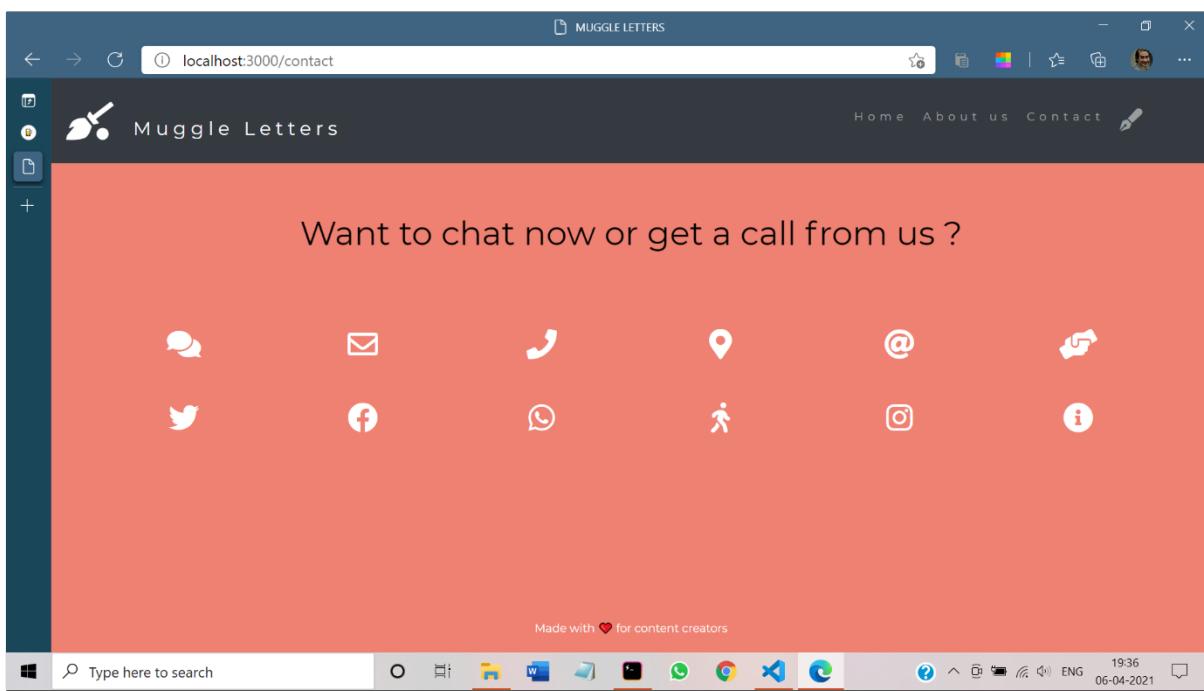
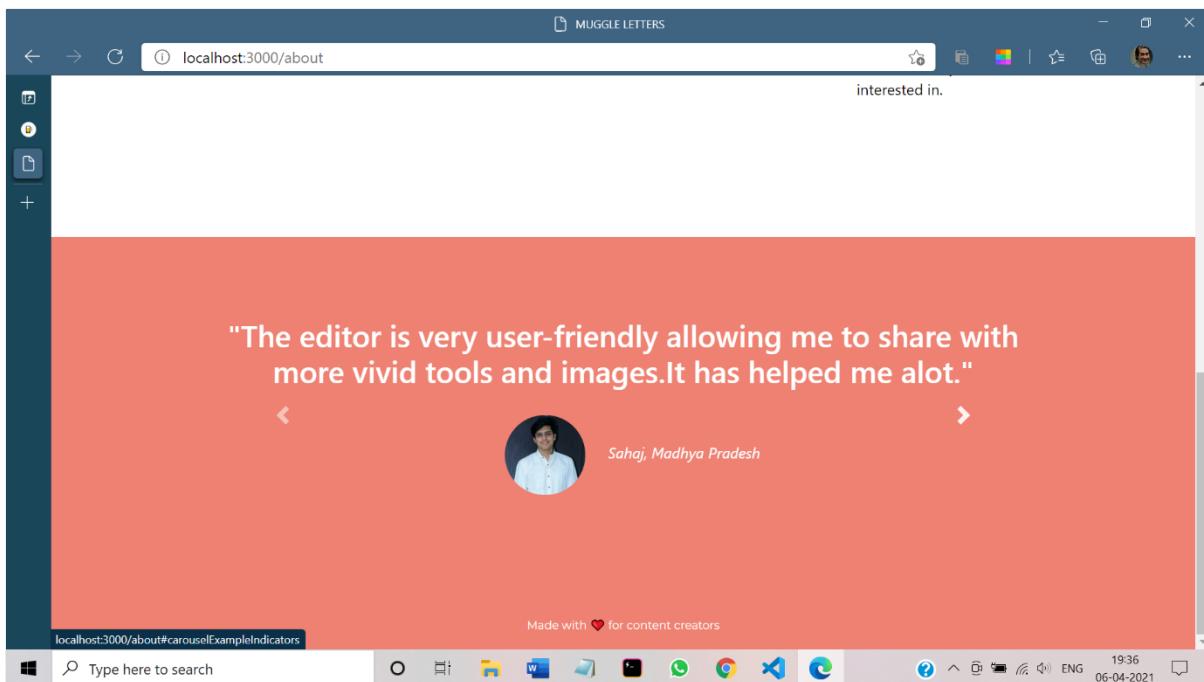
Made with ❤ for content creators

Type here to search

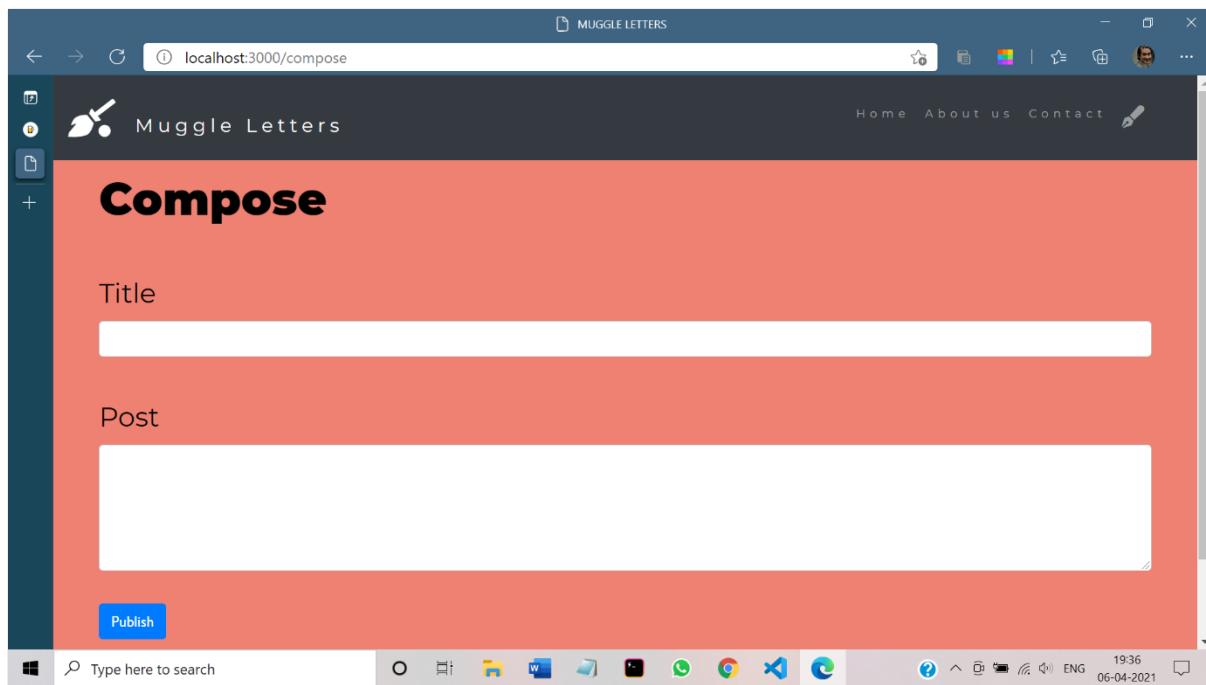
19:36 ENG 06-04-2021



Edit with WPS Office



Edit with WPS Office



Edit with WPS Office

## **WEEK 8**

# **Module Description and Implementation**



Edit with WPS Office

## Code of Module 1

### Home.ejs

```
<!DOCTYPE html>
<html lang="en" dir="ltr">

<head>
  <meta charset="utf-8">
  <title>MUGGLE LETTERS</title>
  <meta charset="utf-8">

  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css" integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm" crossorigin="anonymous">

  <script src="https://kit.fontawesome.com/abba4193f8.js" crossorigin="anonymous"></script>

  <link rel="preconnect" href="https://fonts.gstatic.com">

  <link href="https://fonts.googleapis.com/css2?family=Montserrat:ital,wght@0,400;0,900;1,900&family=Ubuntu:wght@300&display=swap" rel="stylesheet">

  <link rel="stylesheet" href="/css/styles.css">

<!--
  <nav id="nav-color" class="navbar navbar-default">
    <div class="container">
      <div class="navbar-header">
        <p class="navbar-brand">MUGGLE LETTERS</p>
        <i class="fas fa-quidditch fa-2x navbar-brand"></i>
      </div>
      <ul class="nav navbar-nav navbar-right">
        <li id="home"><a href="/">HOME</a></li>
        <li id="about"><a href="/about">ABOUT US</a></li>
        <li id="contact"><a href="/contact">CONTACT US</a></li>
        <li id="Compose" ><a href="/compose" ><i class="fas fa-pen-fancy fa-3x"></i></a></li>
      </ul>
    </div>
  </nav>
-->
<nav class="navbar navbar-expand-xl navbar-dark bg-dark navigation">
  <span class="navbar-brand mb-0 h1"><i class="fas fa-quidditch fa-2x navbar-brand"></i>Muggle Letters</span>
```



Edit with WPS Office

```

<button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNavAltMarkup" aria-controls="navbarNavAltMarkup" aria-expanded="false" aria-label="Toggle navigation">
  <span class="navbar-toggler-icon"></span>
</button>
<div class="collapse navbar-collapse" id="navbarNavAltMarkup">
  <div class="navbar-nav">
    <a class="nav-item nav-link" style="margin-left: 520px;" href="/">Home <span class="sr-only">(current)</span></a>
    <a class="nav-item nav-link" href="/about">About us</a>
    <a class="nav-item nav-link" href="/contact">Contact</a>
    <a class="nav-item nav-link" href="/compose"><i class="fas fa-pen-fancy fa-2x"></i></a>
  </div>
</div>
</nav>

<body class="body-color">
  <div>

```

## Footer.ejs

```

</div>
<div class="footer-padding"></div>
<div class="footer">
  <p>Made with     for content creators</p>
</div>
</div>

<script src="https://code.jquery.com/jquery-3.2.1.slim.min.js" integrity="sha384-KJ3o2DKtlkvYIK3UENzmM7KCkRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hXpG5KkN" crossorigin="anonymous"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/popper.min.js" integrity="sha384-ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPskvXusvfa0b4Q" crossorigin="anonymous"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js" integrity="sha384-JZR6Spejh4U02d8j0t6vLEHfe/JQGiRRSQQxSfFWpi1MquVdAyUar5+76PVCmYI" crossorigin="anonymous"></script>

</body>
</html>

```

## CSS OF ALL FRONT END



Edit with WPS Office

```
/* MID SECTION */

table, th, td {
  padding-left: 90px;
  padding-right: 90px;
}
.home-mid-sec{
  background-color: #fff;
  text-align: center;
  font-size: 2rem;
  font-family: 'Montserrat', sans-serif;
  padding: 3% 3% 3% 3%;
  font-weight: bold;
}
.download-button{
  margin: 2% 3% 7% 0;
}

/* HOME PAGE */
.home-title{
  font-size: 3rem;
  font-family: 'Montserrat', sans-serif;
  color:black;
  font-weight: bold;
}

.home-body{
  font-size: 1.8rem;
  text-align: left;
  background-color: #ef8172;
  padding: 4% 4% 4% 4%;
  font-family: 'Montserrat', sans-serif;
  color: #fff;
}

/* ABOUT PAGE */
.about-title{
  font-size: 3rem;
  font-family: 'Montserrat', sans-serif;
  color: black;
  font-weight: bold;
  text-align: center;
  padding: 4% 4% 4% 4%;
}
```



Edit with WPS Office

```
.about-body{
  font-size: 1.3rem;
  text-align: center;
  background-color: #ef8172;
  padding: 4% 4% 4% 4%;
  font-family: 'Montserrat', sans-serif;
  color: #fff;
}

/* CONTACT PAGE*/
.contact-title{
  font-size: 2.2rem;
  font-family: 'Montserrat', sans-serif;
  color: black;
  font-weight: 300;
  margin-bottom: 20px;
  text-align: center;
}

.contact-body{
  font-size: 2rem;
  text-align: center;
  background-color: #ef8172;
  font-family: 'Montserrat', sans-serif;
  color: #fff;
}

.form-group{
  margin-bottom: 30px;
}

/* COMPOSE PAGE*/

.compose-title{
  font-size: 3rem;
  font-family: 'Montserrat', sans-serif;
  color: black;
  font-weight: bold;
}

.compose-body{
  font-size: 1.8rem;
```



Edit with WPS Office

```
text-align: left;
background-color: #ef8172;
padding: 1% 4% 4% 4%;
font-family: 'Montserrat', sans-serif;
color: black;
}

/* Posts separate page */

.post-title{
font-size: 2.3rem;
font-family: 'Montserrat', sans-serif;
color: #1b1717;
font-weight: bold;
margin-bottom: 20px;

}

.post-body{
font-size: 1rem;
text-align: left;
background-color: #ef8172;
padding: 3% 3% 3% 3%;
font-family: 'Montserrat', sans-serif;
color: #fff;
}

/* NEW POSTS */

.post-home-title{
font-size: 2.3rem;
font-family: 'Montserrat', sans-serif;
color: #1b1717;
font-weight: bold;
margin-bottom: 20px;

}

.post-home-body{
font-size: 1rem;
text-align: left;
background-color: #ef8172;
padding: 4% 3% 3% 3%;
font-family: 'Montserrat', sans-serif;
color: #fff;
/* border-style: solid;
```



Edit with WPS Office

```
border-width: 2px;
margin-bottom: 3px; */

}

/* -----
#testimonials{
padding: 7% 15%;
text-align: center;
background-color: #ef8172;
color: #fff;

}
.testimonial-image{
width: 10%;
border-radius: 100%;
margin: 20px;
}

.navigation{
font-size: 16px;
font-family: 'Montserrat', sans-serif;
}
.a-class{
width: 120px;
height: 120px;
margin-top: 40px;
margin-left: 50px;
margin-right: 50px;

}
.c-image{
width: 120px;
height: 120px;
margin-left: 20px;

}

.body-color{
background-color: #ef8172;
}

/* Don't Change these below*/
```



Edit with WPS Office

```
html {
  min-height: 100%;
  position: relative;
}

.container-fluid {
  padding-top: 70px;
  padding-bottom: 70px;
}

.navbar {
  padding-top: 15px;
  padding-bottom: 15px;
  border: 0;
  border-radius: 0;
  margin-bottom: 0;
  font-size: 12px;
  letter-spacing: 5px;
}

.navbar-nav li a:hover {
  color: white !important;
}

.footer-padding {
  padding-bottom: 60px;
}

.footer {
  position: absolute;
  text-align: center;
  bottom: 0;
  width: 100%;
  height: 60px;
  background-color: #ef8172;
  font-family: 'Montserrat', sans-serif;
  font-weight: 300;
}

.footer p {
  margin-top: 25px;
  font-size: 12px;
  color: #fff;
}

/* */
```



```
#features{
  padding: 7% 15%;
  background-color: white;
  position: relative;
  z-index: 1;
  color: black;
}

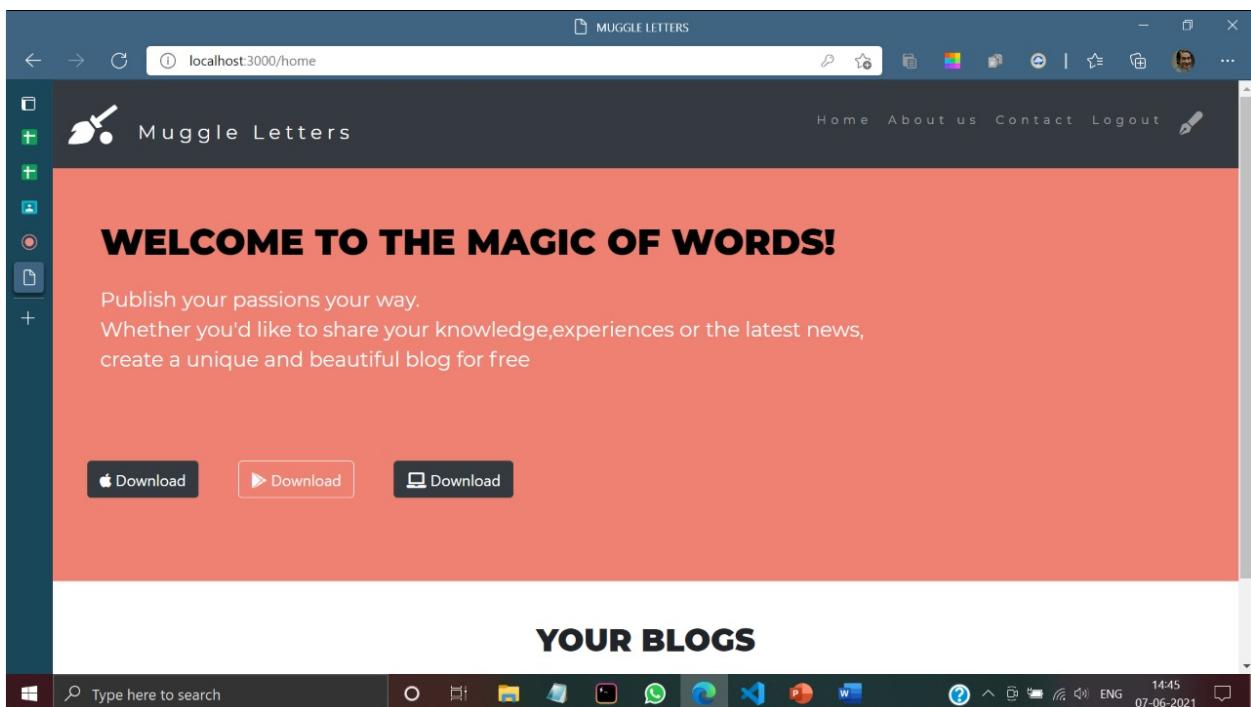
.feature-box{
  text-align: center;
  padding: 5%;
  color: black;
}

.fa-4x{
  color: #ef8172;
  margin-bottom: 1rem;
}

.fa-4x:hover{
  color: #ff4c68;
}

/* */
```

## Result of Module 1



Edit with WPS Office

MUGGLE LETTERS

localhost:3000/about

Muggle Letters

Home About us Contact Logout

## About-us

 Easy to use.

Give your blog the perfect home. Get a free blogspot.com domain or buy a custom domain with just a few clicks.

 Earn money

Get paid for your hard work. Google AdSense can automatically display relevant targeted ads on your blog posts.

 Know your audience

Find out which posts are a hit with Blogger's built-in analytics. You'll see where

Type here to search

14:45 07-06-2021

MUGGLE LETTERS

localhost:3000/about

interested in.

"Its feature to publish and unpublish articles and blogs anytime has helped me alot."

 Kinjalk, Mumbai

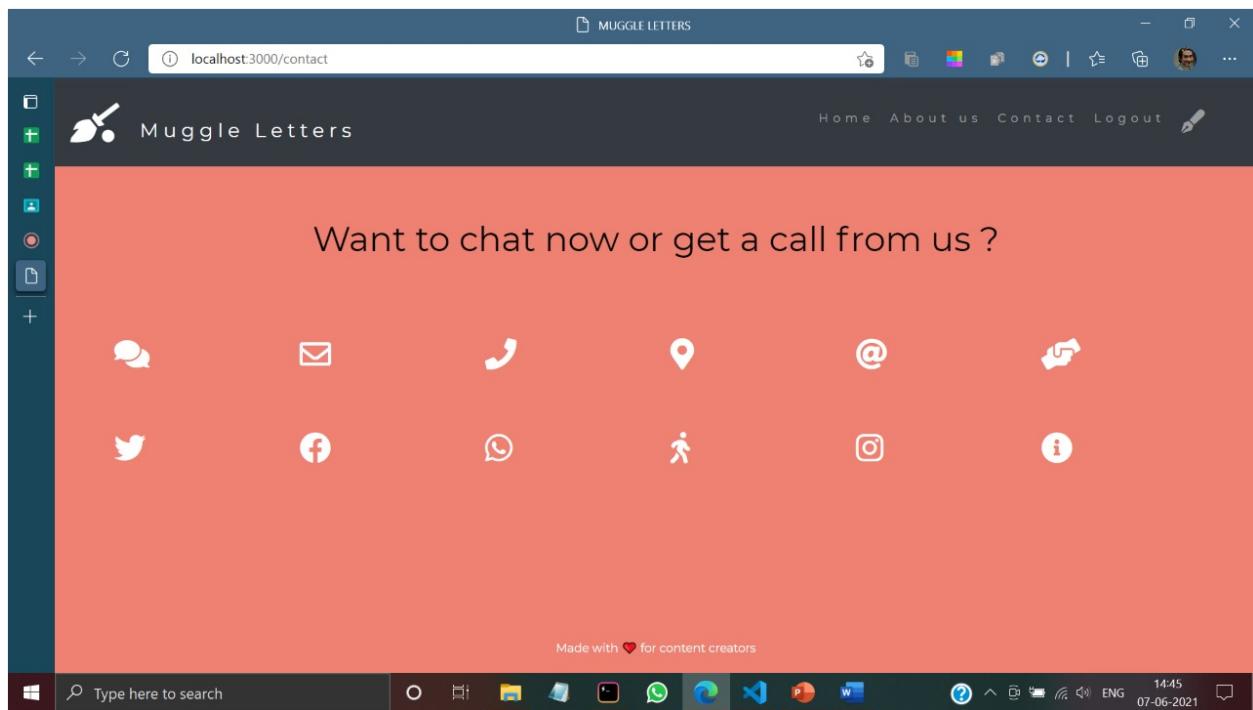
Made with ❤ for content creators

Type here to search

14:45 07-06-2021



Edit with WPS Office



Edit with WPS Office

# WEEK 9

## Module 2 Implementation



Edit with WPS Office

## Code of Module 2

### HOME PAGE

```
<%- include("partials/header"); -%>

<div class="home-body">
  <p class="home-title">WELCOME TO THE MAGIC OF WORDS!</p>
  <p>Publish your passions your way. <br> Whether you'd like to share your knowledge,experiences or the latest news,
    <br>create a unique and beautiful blog for free</p>
</div>

<button type="button" class="btn btn-dark download-button" style="margin-left: 35px;"><i class="fab fa-apple"></i> Download</button>
<button type="button" class="btn btn-outline-light download-button"><i class="fab fa-google-play"></i> Download</button>
<button type="button" class="btn btn-dark download-button"><i class="fas fa-laptop home-icon"></i> Download</button>

<div class="home-mid-sec">
  <p>YOUR BLOGS</p>
</div>
<% posts.forEach(function(post){ %>

  <div class="post-home-body">
    <h1 class="post-home-title"><%=post.title%></h1>
    <br>
    <p>
      <%=post.content.substring(0, 300) + " ..." %>
      <a href="/posts/<%=post._id%>">Read More</a>
      <a class="btn btn-outline-danger btn-sm" href="/delete/<%=post._id%>">Delete</a>
    </p>
  </div>

  <% }) %>

<%- include("partials/footer"); -%>
```

### ABOUT PAGE



Edit with WPS Office

```

<%- include("partials/header"); -%>
<h1 class="about-title">About-us</h1>

<section id="features">
  <div class="row">
    <div class="feature-box col-lg-4">
      <i class="fas fa-check-circle fa-4x"></i>
      <h3>Easy to use.</h3>
      <p>Give your blog the perfect home. Get a free blogspot.com domain or buy a custom domain with just a few clicks.</p>
    </div>

    <div class="feature-box col-lg-4">
      <i class="fas fa-bullseye fa-4x"></i>
      <h3>Earn money</h3>
      <p>Get paid for your hard work. Google AdSense can automatically display relevant targeted ads on your blog</p>
    </div>

    <div class="feature-box col-lg-4">
      <i class="fas fa-heart fa-4x"></i>
      <h3>Know your audience</h3>
      <p>Find out which posts are a hit with Blogger's built-in analytics. You'll see where your audience is coming from and what they're interested in.</p>
    </div>

  </div>
</section>

<section id="testimonials">
  <div id="carouselExampleIndicators" class="carousel slide" data-ride="false" data-interval="1000" data-pause="hover">
    <div class="carousel-inner">
      <div class="carousel-item active">
        <h2>"Its feature to publish and unpublish articles and blogs anytime has helped me alot."</h2>
        
        <em>Kinjalk, Mumbai</em>
      </div>
      <div class="carousel-item">
        <h2 class="testimonial-text">"I think MUGGLE LETTERS is a reliable and easy system for someone who wants to make a Blog."</h2>
        
      </div>
    </div>
  </div>
</section>

```



```

<em>Beverly, Illinois</em>
</div>
<div class="carousel-item">
  <h2 class="testimonial-text">"The editor is very user-
friendly allowing me to share with more vivid tools and images.It has helped me alot."</h2>
  
  <em>Sahaj, Madhya Pradesh</em>
</div>

</div>
<a class="carousel-control-prev" href="#carouselExampleIndicators" role="button" data-
slide="prev">
  <span class="carousel-control-prev-icon"></span>
</a>
<a class="carousel-control-next" href="#carouselExampleIndicators" role="button" data-
slide="next">
  <span class="carousel-control-next-icon"></span>
</a>
</div>

</section>

<%- include("partials/footer"); -%>

```

## CONTACT PAGE

```

<%- include("partials/header"); -%>
<div class="contact-body">

  <br>
  <p class="contact-title"> <%= contactContent %> </p>
  <br>

<div class="container">
  <div class="row">

```



Edit with WPS Office

```
<div class="col">
    <i class="fas fa-comments"></i>
</div>
<div class="col">
    <i class="far fa-envelope"></i>
</div>
<div class="col">
    <i class="fas fa-phone"></i>
</div>
<div class="col">
    <i class="fas fa-map-marker-alt"></i>
</div>
<div class="col">
    <i class="fas fa-at"></i>
</div>
<div class="col">
    <i class="fas fa-hands-helping"></i>
</div>

</div>

<br>

<div class="row">
    <div class="col">
        <i class="fab fa-twitter"></i>
    </div>
    <div class="col">
        <i class="fab fa-facebook"></i>
    </div>
    <div class="col">
        <i class="fab fa-whatsapp"></i>
    </div>
    <div class="col">
        <i class="fas fa-walking"></i>
    </div>
    <div class="col">
        <i class="fab fa-instagram"></i>
    </div>
    <div class="col">
        <i class="fas fa-info-circle"></i>
    </div>
</div>

</div>
```



Edit with WPS Office

```
</div>

<%- include("partials/footer"); -%>
```

## COMPOSE PAGE

```
<%- include("partials/header"); -%>



<h1 class="compose-title">Compose</h1>
  <br>
  <form action="/compose" method="post">
    <div class="form-group">
      <label>Title</label>
      <br>
      <input class="form-control" type="text" name="postTitle">
      <br>
      <label>Post</label>
      <textarea class="form-control" name="postBody" rows="5" cols="30"></textarea>
    </div>

    <button class="btn btn-primary" type="submit" name="button" style="font-family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS', sans-serif;">Publish</button>
  </form>
</div>


```

```
<%- include("partials/footer"); -%>
```

## POST PAGE

```
<%- include("partials/header"); -%>


<h1 class="post-title"><%=title%></h1>
  <p><%=content%></p>
</div>

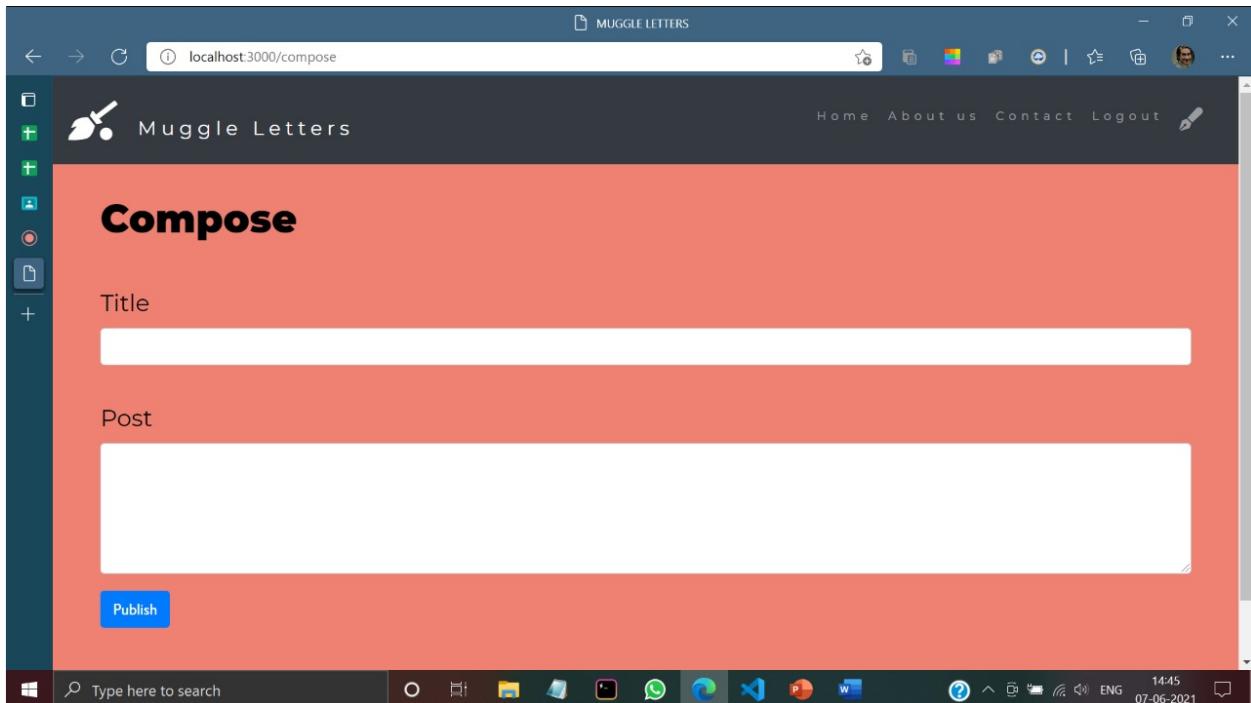

```

```
<%- include("partials/footer"); -%>
```

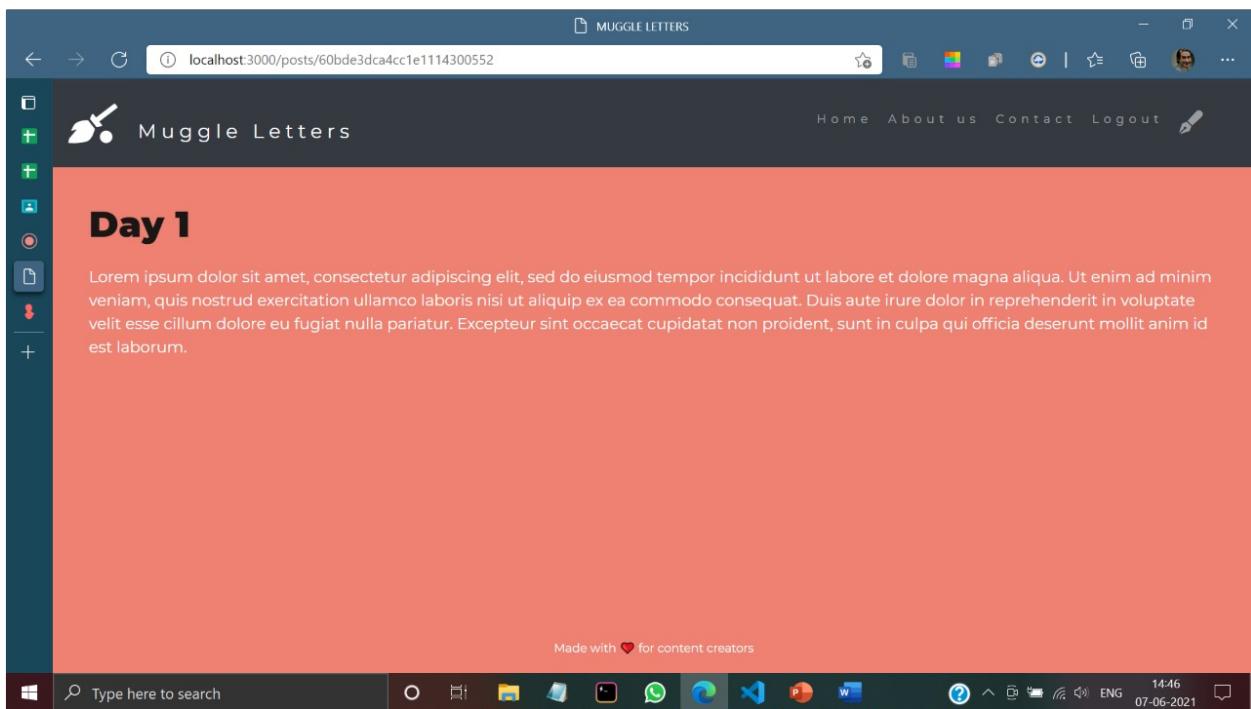


Edit with WPS Office

# Result of Module 2



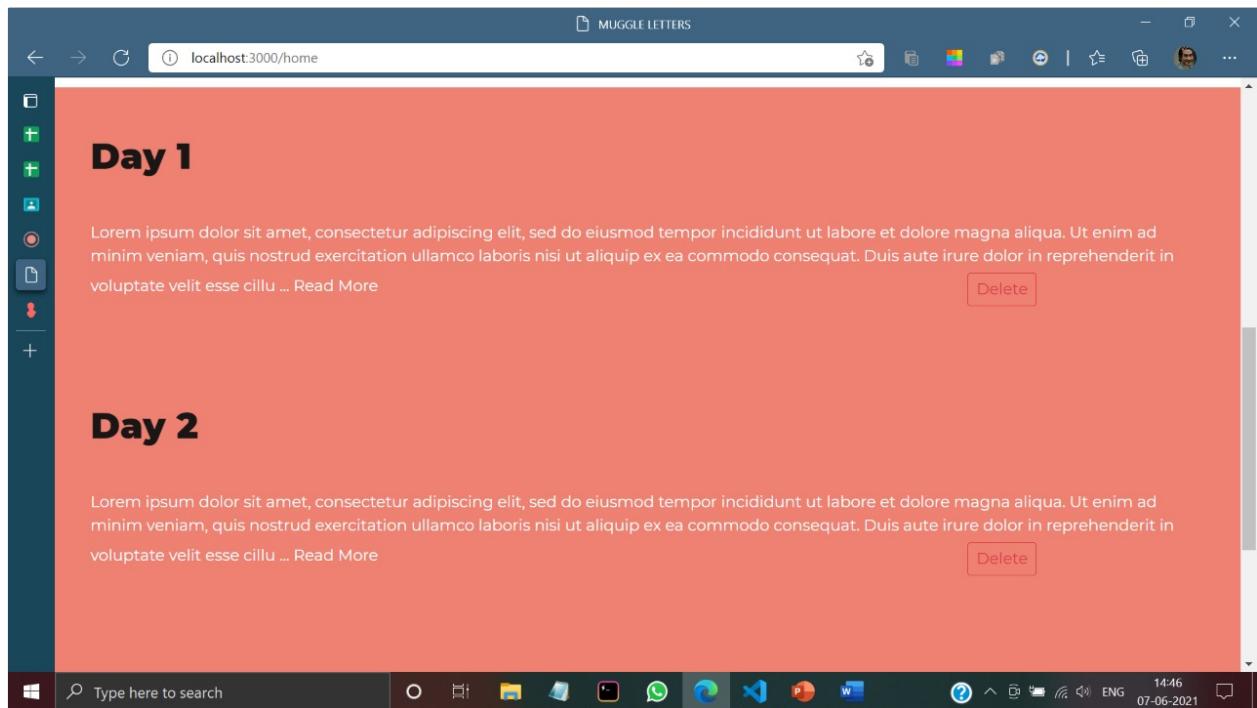
A screenshot of a web browser window showing the 'Compose' page of a website called 'Muggle Letters'. The URL in the address bar is 'localhost:3000/compose'. The page has a dark header with the site logo and navigation links for Home, About us, Contact, Logout, and a pencil icon. On the left, there's a sidebar with icons for file operations like New, Open, Save, and Delete. The main content area has a red background and contains two input fields: 'Title' and 'Post', both currently empty. A blue 'Publish' button is located at the bottom of the form. The browser's taskbar at the bottom shows various open tabs and system status.



A screenshot of a web browser window showing a post on the 'Mettle Letters' website. The URL in the address bar is 'localhost:3000/posts/60bde3dca4cc1e1114300552'. The page has a dark header with the site logo and navigation links. The main content area has a red background and displays the title 'Day 1' in large bold letters. Below the title is a paragraph of placeholder text (Lorem ipsum) and a note at the bottom right: 'Made with ❤️ for content creators'. The browser's taskbar at the bottom shows various open tabs and system status.



Edit with WPS Office



Edit with WPS Office

# WEEK 10

## Module 3 Implementation



Edit with WPS Office

## Code of Module 3

### APP.JS

```
//jshint esversion:6

const express = require("express");
const bodyParser = require("body-parser");
const ejs = require("ejs");
const _ = require("lodash");
const mongoose = require('mongoose');

const contactContent = "Want to chat now or get a call from us ?"
const app = express();

app.set('view engine', 'ejs');

app.use(bodyParser.urlencoded({extended: true}));
app.use(express.static("public"));

mongoose.connect("mongodb://localhost:27017/blogDB", {useNewUrlParser: true, useUnifiedTopology: true});

const PostSchema = {
  title: String,
  content: String
}

const Post = mongoose.model("Post", PostSchema);

app.get("/home", function(req, res){

  Post.find({}, function(err, posts){
    res.render("home", {
      posts: posts
    });
  });
});

app.get("/about", function(req, res){
  res.render("about");
});
```



Edit with WPS Office

```
app.get("/contact", function(req, res){
  res.render("contact", {contactContent: contactContent});
});

app.get("/compose", function(req, res){
  res.render("compose");
});

app.post("/compose", function(req, res){
  const post = new Post({
    title: req.body.postTitle,
    content: req.body.postBody
  });

  post.save(function(err){
    if (!err){
      res.redirect("/home");
    }
  });
});

app.get("/posts/:postId", function(req, res){

  Post.findOne({_id: requestedPostId}, function(err, post){
    res.render("post", {
      title: post.title,
      content: post.content
    });
  });
});

app.get("/delete/:postId", function(req, res){

  const requestedPostId = req.params.postId;

  Post.findByIdAndRemove(requestedPostId, function(err){
    if (!err) {
      console.log("Successfully deleted checked item.");
      res.redirect("/home");
    }
  });
});
```



Edit with WPS Office

```
        }
    });

});

const userSchema = new mongoose.Schema({
  email: String,
  password: String
});

const User = new mongoose.model("User",userSchema);

app.get("/",function(req,res){
  res.render("default");
});

app.get("/login",function(req,res){
  res.render("login");
});

app.get("/register",function(req,res){
  res.render("register");
});

app.post("/login",function(req,res){

  const username = req.body.username;
  const password = req.body.password;

  User.findOne({email: username}, function(err, foundUser){
    if(err){
      console.log(err);
    } else{
      if(foundUser) {
        if(foundUser.password === password){
          res.redirect("/home");
        }
        else{
          res.redirect("/");
        }
      }
      else{
    }
  }
});
```



Edit with WPS Office

```

        res.redirect("/");
    }
});
});

app.post("/register", function(req,res){
  const newUser = new User({
    email: req.body.username,
    password: req.body.password
});

newUser.save(function(err){
  if(err){
    console.log(err);
  } else{
    res.redirect("/home");
  }
});

app.listen(3000, function() {
  console.log("Server started on port 3000");
});

```

## ROUTE PAGE

```

<%- include('partials2/header2') %>

<div class="jumbotron centered">
  <div class="container">
    <i class="fas fa-key fa-6x"></i>
    <h1 class="display-3">MUGGLE LETTERS</h1>
    <p class="lead">Create a unique and beautiful blog. It's easy and free.</p>
    <hr>
    <a class="btn btn-light btn-lg" href="/register" role="button">Register</a>
    <a class="btn btn-dark btn-lg" href="/login" role="button">Login</a>

  </div>
</div>

```



Edit with WPS Office

```
<%- include('partials2/footer2') %>
```

## LOGIN PAGE

```
<%- include('partials2/header2') %>

<div class="container mt-5">
  <h1>Login</h1>

  <div class="row">
    <div class="col-sm-8">
      <div class="card">
        <div class="card-body">

          <!-- Makes POST request to /login route -->
          <form action="/login" method="POST">
            <div class="form-group">
              <label for="email">Email</label>
              <input type="email" class="form-control" name="username">
            </div>
            <div class="form-group">
              <label for="password">Password</label>
              <input type="password" class="form-control" name="password">
            </div>
            <button type="submit" class="btn btn-dark">Login</button>
          </form>

        </div>
      </div>
    </div>

    <!-- <div class="col-sm-4">
      <div class="card">
        <div class="card-body">
          <a class="btn btn-block" href="/auth/google" role="button">
            <i class="fab fa-google"></i>
            Sign In with Google
          </a>
        </div>
      </div>
    </div> -->
```



Edit with WPS Office

```
</div>
</div>

<%- include('partials2/footer2') %>
```

## REGISTRATION PAGE

```
<%- include('partials2/header2') %>
<div class="container mt-5">
  <h1>Register</h1>

  <div class="row">
    <div class="col-sm-8">
      <div class="card">
        <div class="card-body">

          <!-- Makes POST request to /register route -->
          <form action="/register" method="POST">
            <div class="form-group">
              <label for="email">Email</label>
              <input type="email" class="form-control" name="username">
            </div>
            <div class="form-group">
              <label for="password">Password</label>
              <input type="password" class="form-control" name="password">
            </div>
            <button type="submit" class="btn btn-dark">Register</button>
          </form>

        </div>
      </div>
    </div>

    <!-- <div class="col-sm-4">
      <div class="card social-block">
        <div class="card-body">
          <a class="btn btn-block" href="/auth/google" role="button">
            <i class="fab fa-google"></i>
            Sign Up with Google
          </a>
        </div>
      </div>
    </div> -->
```



Edit with WPS Office

```
</div>
</div>

<%- include('partials2/footer2') %>
```

## CSS FOR LOGIN/REGISTER

```
body {
    background-color: #E8ECEF;
}

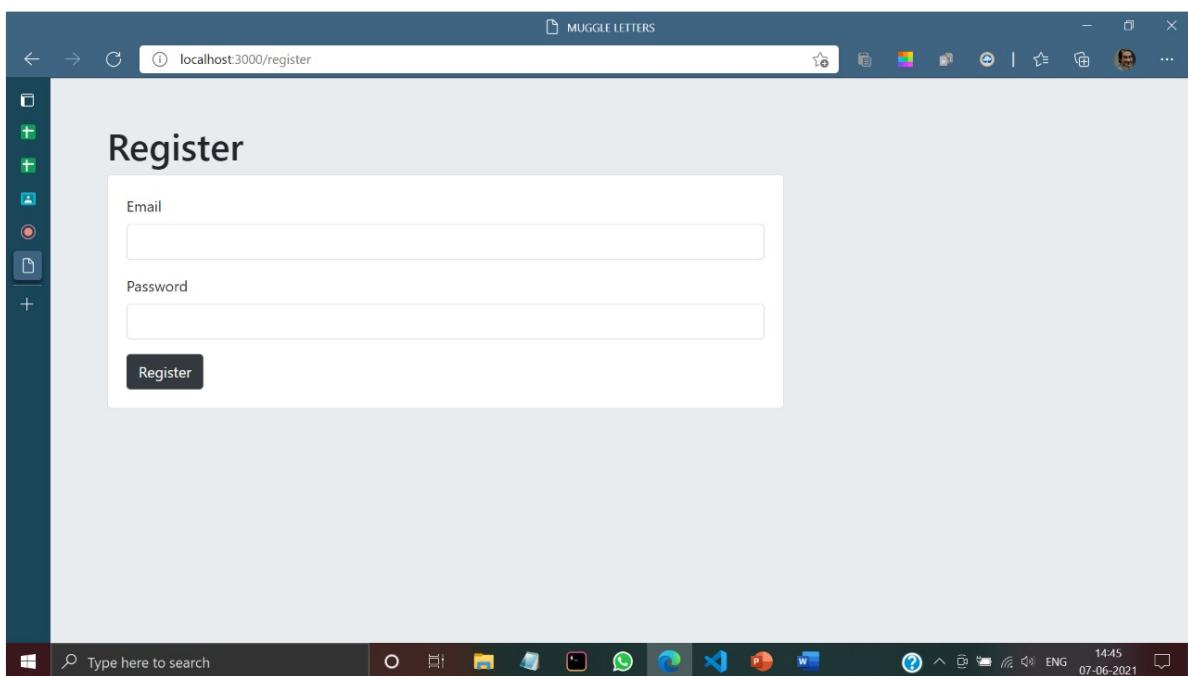
.centered {
    padding-top: 200px;
    text-align: center;
}

.secret-text {
    text-align: center;
    font-size: 2rem;
    color: #fff;
    background-color: #000;
}
```

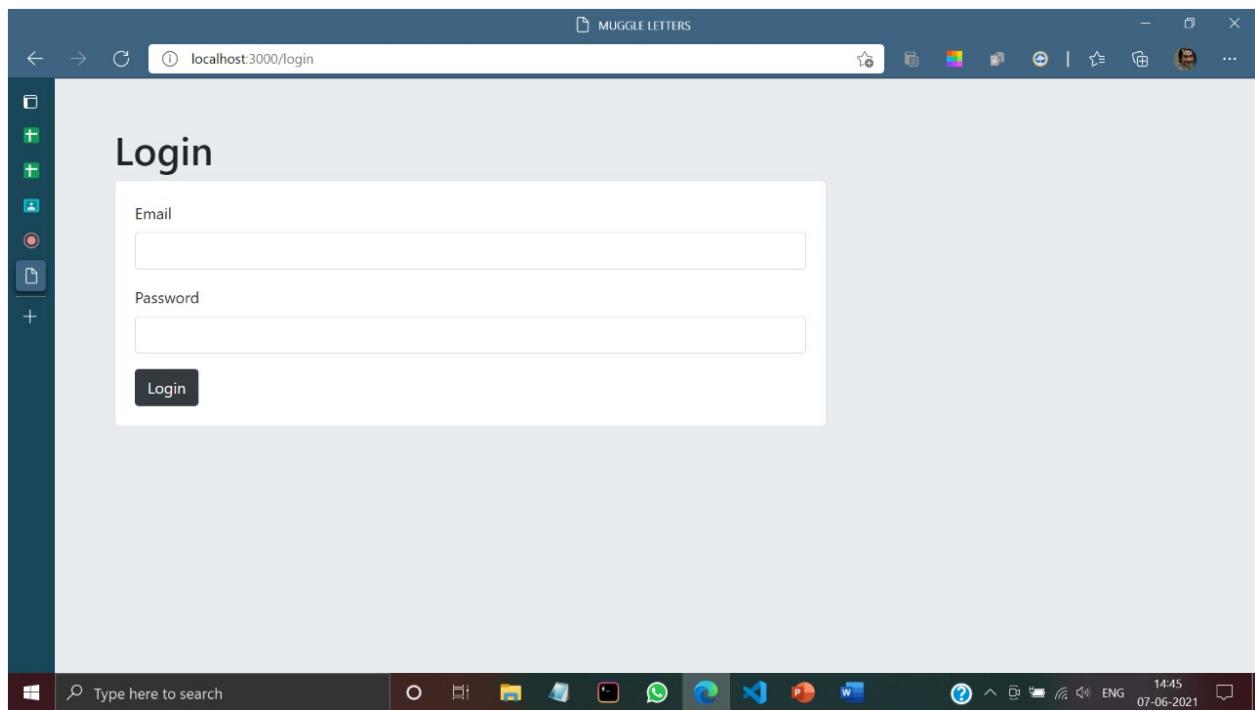


Edit with WPS Office

# Result of Module 3



Edit with WPS Office



Edit with WPS Office

# WEEK 11

## Test Case, Test Cases



Edit with WPS Office

## 1. Executive Summary

Software Testing is a method to check whether the actual software product matches expected requirements and to ensure that software product is Defect free. It involves execution of software/system components using manual or automated tools to evaluate one or more properties of interest. The purpose of software testing is to identify errors, gaps or missing requirements in contrast to actual requirements. In this report we are going to cover our test plan and methodology along with the different use cases and finally generate a test analysis report.

## 2. Test Plan

Test Management is a process of managing the testing activities in order to ensure high quality and high-end testing of the software application. The method consists of organizing, controlling, ensuring traceability and visibility of the testing process in order to deliver the high quality software application. It ensures that the software testing process runs as expected.

We plan to have a manual approach to testing and use methodologies such as unit testing and smoke testing. Following is our general plan to go about the testing phase

- The content model for the Web App is reviewed to uncover errors.
- The interface model is reviewed to ensure that all use cases can be accommodated.
- The design model for the Web App is reviewed to uncover navigation errors.
- The user interface is tested to uncover errors in presentation and/or navigation mechanics.
- Each functional component is unit tested.
- Navigation throughout the architecture is tested.
- Performance tests are conducted.

### 2.1. Scope of Testing

Software Testing is an investigation conducted to provide stakeholders with information about the quality of a particular product or service under test. In other words, software testing is a process of verification and validation.

We plan to include test cases related to both the functional and nonfunctional requirements and ensure that all modules are covered and any exceptions are well documented.

### 2.2. Types of Testing , Methodology , Tools



Category	Methodology	Tools Required
Functional Requirements	Manual (Unit and smoke testing)	Word Template
Non Functional Requirements	Manual	Word Template

## 2.3. Test Deliverables

The test deliverables include items such as Test case documentation, defect log and Test summary report.

The test case documentation involves a list of all the functional and non functional test cases and the outcomes of those test cases.

The defect log contains the list of all the exceptions and defects encountered during the testing process.

The test summary report is the final output of the testing process and is used to determine if the software is ready and functions without any bugs, this report lets the developers know if any further modifications are needed.

## 3. Test Case

### 3.1. Functional Test Cases

Test ID (#)	Test Scenario	Test Case	Execution Steps	Expected Outcome	Actual Outcome	Status	Remarks
1	Adding the user's details once an account is created.	Accept the details entered by the user and adding them to database	1. User clicks on User Registration link 2. Enters the Email id and Password to be set on the text box 3. Click Register button	User should be taken to the home page and alerted with a message "Account successfully created"	User is taken to the home page of the software .	Pass	success
2	Hiding the password while the user is	Password must be hidden while typing	1. User enters the Email Id in the	User should be taken to the home page of	User is taken to the home page of	Pass	success



	signing in to the platform		2. given text box User then moves to the next text box to enter his Password	the platform.	the platform		
3	Displaying error message when wrong email is being entered	When wrong email id is entered user will be alerted,	1. User enters the login page. 2. Enters his/her Email ID in the given text box	User should be alerted that the entered Email ID is wrong.	User is alerted that the Email ID is wrong.	Pass	success
4	Saving the written blogs to the database and displaying them whenever user wants to view	Blogs written by the user will be saved in the database.	1.User enters the compose section 2.Enters the title of the blog in the text box given 3.Composes the blog in the given text box. 4.Clicks on the Submit button to post the composed blog.	Blog must be saved in the database and must be visible in the home page	Blog is saved in the database and is displayed in the home page	Pass	Success
5	Password field takes a minimum of 6 characters, a maximum of 8 characters, numbers (0-9), letters (a-z, A-Z),	The Application must take a minimum of 6 characters and a maximum of 8 characters including all the special characters	User enters the Email Id in the given text box 2.User then moves to the next text box to enter his Password	The application should display an alert saying Password exceeded the limit / password is too short	No message is displayed saying that Password is too short or too long	Fail	Not successful



	special characters (all), and cannot be blank.						
--	--	--	--	--	--	--	--

## 3.2. Non-Functional Test Cases

Test ID (#)	Test Scenario	Test Case	Execution Steps	Expected Outcome	Actual Outcome	Status	Remarks
1	Application must be accessible on all types of devices	Application must be compatible to all versions of Windows and Mac	Application will be opened on various versions of Windows and Mac to check the app's Performance	App must function Smoothly on all versions of Windows and Mac	Application functioned smoothly on all the devices	Pass	Success
2	All web images should have alt tags	In case of a problem while loading the images an alternative text must be displayed	1.Path to the image is wrongly specified such that problem will occur in loading the picture 2.By using alt tag in the code, an alternate text will be displayed	Whenever there was a problem in loading images , alternate text must be displayed	Alternate text was displayed when the images weren't loaded	Pass	Success
3	Ensuring that a database replicates	Maintaining a secondary database	1.We try to simulate a real life scenario	In the event of failure, the	In the event of failure, data is	Fail	Can be resolved with more



	to a standby instance , and that in the event of failure the standby instance can takeover with no loss of service	as a standby such that data won't be lost in case of any failures	where the database gets deleted. 2.We check if the data stored is still available	standby database must be able to takeover without any loss of service	lost as there is no standby database.		time and resources
4	Must withstand and recover from a Denial-of-service(Dos or DDos) attack, where external systems attempt to flood the bandwidth or resources	Must be able to defend external systems which tries to flood the bandwidth or resources or target firewall.	Must be able to defend external systems which tries to flood the bandwidth or resources or target firewall.	System must be stable and reliable even when under extremely heavy load conditons	As the amount of load increased, the application couldn't cope with the extreme demand and led to crashing of servers or slowing them down.	Fail	Not successful as there's no firewall.



# **WEEK 12**

# **Manual Testing With Report**



Edit with WPS Office

## 1. Executive Summary

Software Testing is a method to check whether the actual software product matches expected requirements and to ensure that software product is Defect free. It involves execution of software/system components using manual or automated tools to evaluate one or more properties of interest. The purpose of software testing is to identify errors, gaps or missing requirements in contrast to actual requirements. In this report we are going to cover our test plan and methodology along with the different use cases and finally generate a test analysis report.

## 2. Test Plan

Test Management is a process of managing the testing activities in order to ensure high quality and high-end testing of the software application. The method consists of organizing, controlling, ensuring traceability and visibility of the testing process in order to deliver the high quality software application. It ensures that the software testing process runs as expected.

We plan to have a manual approach to testing and use methodologies such as unit testing and smoke testing. Following is our general plan to go about the testing phase

- The content model for the Web App is reviewed to uncover errors.
- The interface model is reviewed to ensure that all use cases can be accommodated.
- The design model for the Web App is reviewed to uncover navigation errors.
- The user interface is tested to uncover errors in presentation and/or navigation mechanics.
- Each functional component is unit tested.
- Navigation throughout the architecture is tested.
- Performance tests are conducted.

### 2.1. Scope of Testing

Software Testing is an investigation conducted to provide stakeholders with information about the quality of a particular product or service under test. In other words, software testing is a process of verification and validation.

We plan to include test cases related to both the functional and nonfunctional requirements and ensure that all modules are covered and any exceptions are well documented.

### 2.2. Types of Testing , Methodology , Tools

Category	Methodology	Tools Required
Functional Requirements	Manual (Unit and smoke testing)	Word Template



Non Functional Requirements	Manual	Word Template
-----------------------------	--------	---------------

## 2.3. Test Deliverables

The test deliverables include items such as Test case documentation, defect log and Test summary report.

The test case documentation involves a list of all the functional and non functional test cases and the outcomes of those test cases.

The defect log contains the list of all the exceptions and defects encountered during the testing process.

The test summary report is the final output of the testing process and is used to determine if the software is ready and functions without any bugs, this report lets the developers know if any further modifications are needed.

## 3. Test Case

### 3.1. Functional Test Cases

Test ID (#)	Test Scenario	Test Case	Execution Steps	Expected Outcome	Actual Outcome	Status	Remarks
1	Adding the user's details once an account is being created.	Accept the details entered by the user and adding them to database	<ol style="list-style-type: none"> <li>User clicks on User Registration link</li> <li>Enters the Email id and Password to be set on the text box</li> <li>Click Register button</li> </ol>	User should be taken to the home page and alerted with a message "Account successfully created"	User is taken to the home page of the software .	Pass	success
2	Hiding the password while the user is signing in to the platform	Password must be hidden while typing	<ol style="list-style-type: none"> <li>User enters the Email Id in the given text box</li> <li>User then</li> </ol>	User should be taken to the home page of the platform.	User is taken to the home page of the platform	Pass	success



			moves to the next text box to enter his Password				
3	Displaying error message when wrong credentials are entered	When wrong login credentials are entered user will be alerted,	1. User enters the login page. 2. Enters his/her Email ID and password in the given text box	User should be alerted that the entered Email ID/password is wrong.	User is alerted that the Email ID/password is wrong.	Pass	success
4	Saving the written blogs to the database and displaying them whenever user wants to view	Blogs written by the user will be saved in the database.	1.User enters the compose section 2.Enters the title of the blog in the text box given 3.Composes the blog in the given text box. 4.Clicks on the Submit button to post the composed blog.	Blog must be saved in the database and must be visible in the home page	Blog is saved in the database and is displayed in the home page	Pass	Success
5	Password field takes a minimum of 6 characters, a maximum of 8 characters, numbers (0 -9), letters (a-z, A-Z), special characters (all), and cannot be blank.	The Application must take a minimum of 6 characters and a maximum of 8 characters including all the special characters	1. User enters the Email Id in the given text box 2.User then moves to the next text box to enter his Password	The application should display an alert saying Password exceeded the limit / password is too short	No message is displayed saying that Password is too short or too long	Fail	Not successful
6	Checking whether	The format of the email	1.User enters his/her email in	An alert must be	An alert is displayed	Pass	Success



	the entered email id is valid or not.	must be checked and if it's wrong an error message must be displayed	the registration form. 2.The app checks the format of the email	displayed saying entered email isn't valid,	when an invalid email is entered		
7	Checking the delete functionality of the application	The blog must be deleted when the user clicks on the delete option	1.User will have to click the delete button in order to delete the blog he desires to delete	The blog must be deleted when the delete button is clicked	The blog is deleted when the delete button is pressed.	Pass	success
8	Checking the logout functionality of the application	The user must be logged out from his/her account by clicking on the log out button	1.User will have to click the logout button in the navigation bar	User must be logged out of his/her account when logout button is clicked	User is logged out when he/her clicked logout button	Pass	Success

### 3.1. Non-Functional Test Cases

Test ID (#)	Test Scenario	Test Case	Execution Steps	Expected Outcome	Actual Outcome	Status	Remarks
1	Application must be accessible on all types of devices	Application must be compatible to all versions of Windows and Mac	Application will be opened on various versions of Windows and Mac to check the app's Performance	App must function smoothly on all versions of Windows and Mac	Application functioned smoothly on all the devices	Pass	Success
2	All web images should have alt tags	In case of a problem while loading the images an	1.Path to the image is wrongly specified such that problem will	Whenever there was a problem in loading	Alternate text was displayed when the images weren't	Pass	Success



		alternative text must be displayed	occur in loading the picture 2.By using alt tag in the code, an alternate text will be displayed	images , alternate text must be displayed	loaded		
3	Ensuring that a database replicates to a standby instance , and that in the event of failure the standby instance can takeover with no loss of service	Maintaining a secondary database as a standby such that data won't be lost in case of any failures	1.We try to simulate a real life scenario where the database gets deleted. 2.We check if the data stored is still available	In the event of failure, the standby database must be able to takeover without any loss of service	In the event of failure, data is lost as there is no standby database.	Fail	Can be resolved with more time and resources
4	Must withstand and recover from a Denial-of-service(Dos or DDos) attack, where external systems attempt to flood the bandwidth or resources	Must be able to defend external systems which tries to flood the bandwidth or resources or target firewall.	Implemented by hitting the target resource such as a web server with too many requests at the same time.	System must be stable and reliable even when under extremely heavy load conditon s	As the amount of load increased, the application couldn't cope with the extreme demand and led to crashing of servers or slowing them down.	Fail	Not successful as there's no firewall.



## 4. Defect Log

Requirement #	Defect ID #	Defect Description	Assignee	Status
M1R1	1	There's no standby database such that such that data won't be lost in case of any failures	Aditya Dixit	Not Started
M1R2	2	There's no minimum and maximum limit for password	Abhishek Hindocha	In-Progress
M1R3	3	All the personal details of users is not encrypted	Sai Pranav	Not started
M1R4	4	Configuring firewalls to prevent DoS attacks by blocking all traffic coming from an attacker by identifying his IP	Aditya Dixit	Not started

## 5. Test Report

Category	Progress Against Plan	Status
Functional Testing	Green	Completed
Non-Functional Testing	Green	Completed

Functional	Test Case Coverage (%)	Status
M1 (Front End)	100%	Completed
M2 (Adding and deleting data)	100%	Completed
M3 (Database connectivity)	100%	Completed
M4 (Account creation and login functionality)	100%	Completed





Edit with WPS Office