**Minor Project Ocean of Notes**

**Project Report**

**On**

**OCEAN OF NOTES**

*In partial fulfillment of requirements for the degree*

Of

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

***Submitted By:***

AKSHATA GUPTA (18100BTCSAII02828)

KHUSHBOO SOLANKI (18100BTCSAII02842)

UTKARSH TIWARI (18100BTCSAII02871)

***Under the Guidance of***

PROF. SHUBHAM KOTHARI



**SHRI VAISHNAV VIDYAPEETH VISHWAVIDYALAYA, INDORE SHRI VAISHNAV INSTITUTE OF INFORMATION TECHNOLOGY DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING JANUARY-JUNE 2021**

**Minor Project Ocean of Notes**

**SHRI VAISHNAV VIDYAPEETH VISHWAVIDYALAYA, INDORE SHRI VAISHNAV INSTITUTE OF INFORMATION AND TECHNOLOGY DEPARTMENT OF COMPUTER SCIENCE AND ENGINEEERING**

**DECLARATION**

We hereby declare that the work which is being presented in the project entitled **“OCEAN OF NOTES”** in partial fulfillment of degree of **Bachelor of Technology in Computer Science and Engineering** is an authentic record of our word carried out under the supervision and guidance of **Mr. Shubham Kothari,** Asst. Professor of Computer Science and Engineering. The matter embodied in this project has not been submitted for the award of any degree.

**Team:**

AKSHATA GUPTA

KHUSHBOO SOLANKI

UTKARSH TIWARI

Date:

i

**Minor Project Ocean of Notes**

**SHRI VAISHNAV VIDYAPEETH VISHWAVIDYALAYA, INDORE SHRI VAISHNAV INSTITUTE OF INFORMATION TECHNOLOGY DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**PROJECT APPROVAL SHEET**

Following team has done the appropriate work related to the **“OCEAN OF NOTES”** in the partial fulfillment for the award of **Bachelor of Technology in Computer Science and Engineering** of “SHRI VAISHNAV INSTITUTE OF INFORMATION TECHNOLOGY” and is being submitted to “SHRI VAISHNAV VIDYAPEETH VISHWAVIDYALAYA, INDORE.”

**TEAM:**

AKSHATA GUPTA

KHUSHBOO SOLANKI

UTKARSH TIWARI

**Internal Examiner External Examiner** Date:

ii

**Minor Project Ocean of Notes**

**SHRI VAISHNAV VIDYAPEETH VISHWAVIDYALAYA, INDORE SHRI VAISHNAV INSTITUTE OF INFORMATION TECHNOLOGY DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**CERTIFICATE**

This is to certify that **Ms. Akshata Gupta, Ms. Khushboo Solanki** and **Mr. Utkarsh Tiwari** working in a team have satisfactorily completed the project entitled **“OCEAN OF NOTES”** under the guidance of **Prof. Shubham Kothari** in the partial fulfillment for the award of **Bachelor of Technology in Computer Science and Engineering** of “SHRI VAISHNAV INSTITUTE OF INFORMATION TECHNOLOGY” and is being submitted to “SHRI VAISHNAV VIDYAPEETH VISHWAVIDYALAYA, INDORE” during the academic year January 2021 – June 2021.

Prof. Shubham Kothari Dr. Anand Rajavat **Project Coordinator Director and Head And Project Guide Department of Computer Science and Engineering**

iii

**Minor Project Ocean of Notes**

**ACKNOWLEDGEMENT**

We are grateful to several people for their advice and support during the time of completion of our project. First and foremost, we would like to thank **Dr. Anand Rajavat,** Head of the Department of Computer Science & Engineering and **Mr. Shubham Kothari** the mentor of our project for providing us valuable support and necessary help whenever required and helping us in exploring th e new technologies by the help of their technical expertise. Their direction, supervision and constructive criticism were indeed the source of inspiration for us.

We would also like to express our sincere gratitude towards our Director **Dr. Anand Rajavat** for providing us valuable support.

We are really indebted to **Prof. Shubham Kothari,** project coordinator for helping us in each aspect of our academic activities. We also owe our sincere thanks to all the **faculty members** of Computer Science & Engineering Department who have always been helpful.

We forward our sincere thanks to all **teaching and non-teaching staff** of Computer Science & Engineering department, SVVV Indore for providing necessary information and their kind co operation.

We would also like to thank our parents and family members, our classmates and our friends for their motivation and their valuable suggestion during the project. Last, but not the least, we thank all those people, who have helped us directly or indirectly in accomplishing this work. It has been a privilege to study at SHRI VAISHNAV VIDYAPEETH VISHWAVIDYALAYA, INDORE.

iv

**Minor Project Ocean of Notes**

**ABSTRACT**

*Current state in a student’s life mostly revolves around the creating, managing, and organizing notes and this is not less than any hectic task. Firstly, time plays an important role which is scarce. Everybody has a hectic schedule in their respective lives and adding this one on the top of that really overburdens their schedule.*

*Studies are major concern in today’s world and any compromise in that sector is neither accepted by the student nor their parents. As we know that, if there is no proper management of notes, students may get tensed and might not perform well in studies as they could. Already, there are so much that a student must do in order to perform well in their field, then why add to the list?*

*Not only students, even the faculty needs to ensure that their study material is provided to the students on time and efficiently. That might be a concern for them.*

*Our project mainly focuses on this aspect of a student’s and a teacher’s life. Here, we propose them a platform which ensures the availability of notes. This project allows both teachers and students to upload their respective notes on a platform, which guarantees the availability of them to others. It provides them multiple options if they are not satisfied with one of them. This way one can help many and take help from many.*

v

**Minor Project Ocean of Notes**

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **Figure No.** | **Figure Name** | **Page No.** |
| 4.1 | Use Case Diagram | 18 |
| 4.2 | Class Diagram | 20 |
| 4.3 | Data Flow Diagram (L-0) | 22 |
| 4.4 | Data Flow Diagram (L-1) | 22 |
| 5.1 | Sequence Diagram | 23 |
| 5.2 | Activity Diagram | 24 |
| 5.3 | Entity-Relationship Diagram | 25 |
| 5.4 | Agile Model | 26 |
| 7.1 | Home Page | 31 |
| 7.2 | Authentication (Sign-Up) | 31 |
| 7.3 | Authentication (Sign-In) | 32 |
| 7.4 | Account Page | 32 |
| 7.5 | Courses Page | 33 |
| 7.6 | Course-Upload | 33 |
| 7.7 | Course-Download | 34 |

7.8 About Page 34

vi

**Minor Project Ocean of Notes**

**LIST OF TABLES**

|  |  |  |
| --- | --- | --- |
| **Table No.** | **Table Name** | **Page No.** |
| 2.1 | Comparative Chart | 11 |
| GLOSSARY | | 35 |

vii

**Minor Project Ocean of Notes**

**TABLE OF CONTENT**

**TITLE PAGE NO.** Declaration i Project Approval Sheet ii Certificate iii Acknowledgement iv Abstract v List of Figures vi List of Tables vii

CHAPTER 1 – **INTRODUCTION 1-8** 1.1 Introduction 1 1.2 Problem Statement 2 1.3 Need for the proper System 2 1.4 Objective 3 1.5 Modules of the System 3 1.6 Scope 9

CHAPTER 2 – **LITERATURE SURVEY 10-12** 2.1 Existing System 10 2.2 Comparative Chart 11 2.3 Proposed System 12

CHAPTER 3 – **REQUIREMENT ANALYSIS 13-15** 3.1 Method used for Requirement Analysis 13 3.2 Data Requirements 13 3.3 Function Requirements 13 3.4 Non-Functional Requirements 14 3.5 System Specification 14

3.5.1 Hardware Specification 14 3.5.2 Software Specification 15

CHAPTER 4 – **DESIGN 16-22** 4.1 Design 16 4.2 Software Requirements Specification 17

4.2.1 Supplementary Specifications 17 4..2.2 Use Case Model 18 4.3 Class Diagram 20 4.4 Data Flow Diagram (Level 0, 1) 22

CHAPTER 5 – **SYSTEM MODELLING 23-27** 5.1 Sequence Diagram 23 5.2 Activity Diagram 24 5.3 Entity-Relationship Diagram 25 5.4 System Model 26

viii

**Minor Project Ocean of Notes**

CHAPTER 6 – **CONCLUSION AND FUTURE WORK 28-29** 6.2 Conclusion 28 6.2 Limitation of Project 28 6.3 Future Enhancement 28

CHAPTER 7 **– BIBLIOGRAPHY & REFERENCES 30-34** 7.1 References 30 7.2 Snapshot 31

**Glossary** 35

ix

**Minor Project Ocean of Notes**

**CHAPTER -1**

**INTRODUCTION**

**Minor Project Ocean of Notes**

**INTRODUCTION**

**1.1 INTRODUCTION**

In this academic world, every aspect plays an important role in a student’s life. When a child is in primary school, things can still be in control as they are completely under the guidance of their parents and teachers. But as soon as they enter secondary school or higher education, things start to become more complex. Everything is not an easy task when it comes to studies. And thus, it becomes very complicated to manage them.

Notes play a significant role in every student’s life. And them being reliable and easily available is something very important. Like now-a-days, it is very crucial to gain proper and reliable knowledge about any topic. There may be many sources for this task but the ones giving relevant and precise information are rare. Also, to maintain and organize it is difficult. There must a systematic way which ensures us to give proper way to manage our notes. As we know that COVID-19 crises hit us hard, due to which many students and teachers are troubled to manage their notes. Like teachers are concerned for distributing their content, and students are troubled how to access them.

Using a platform which allows teachers to deliver their notes to students and making it easy for students to access them, we can rectify this problem. This could ensure both students and teachers that there is mutual coordination between the two. As one can upload the notes on the platform and download them.

1

**Minor Project Ocean of Notes**

**1.2 PROBLEM STATEMENT**

❖ We need to design a platform which can be both accessed by teachers and students for the availability of notes. It must provide secure and easier way for the both to manage their notes and make good use of it.

❖ The aim of this project is to make it easy for teachers to share notes and for students to gain them. A student here gets a variety of notes on any topic. There can be numerous of notes for a single topic. A student is free to choose from all of them and download it or read it from there. Also, if a student does excellent job in making notes, he/she is welcome to upload their notes, but all the notes would get verified by the admin.

**1.3 NEED FOR THE PROPER SYSTEM**

❖ Time being the most significant element is every person’s life, is we know that we always lack. And same is the case with students. They need ample of time to make notes and if they are not provided on time, this becomes a grave issue. As they do not have reliable source of knowledge.

❖ Various systems are present which can be helpful in this context and they provide proper services for that. But the issue comes when they are not free of cost. Usually these systems charge certain amount from their users which is not something preferred either by the students or the teachers.

❖ Hence, students are thereby left with no choice other than to study with whatever the material they are left with, which may or may not be reliable. This does gravely affect their academic result.

2

**Minor Project Ocean of Notes**

**1.4 OBJECTIVE**

The objectives of the proposed system are as follows:

❖ To provide both students and teachers the facility of upload and download notes. ❖ Notes of any type like presentations, word document, pdf or handwritten in pdf format can be uploaded.

❖ To facilitate the task of accessing notes.

❖ To make user access the notes anytime as they prefer provided there is network connectivity.

❖ To make the service free of cost so that it is viable for users.

❖ To provide users with login system so that authenticity is maintained.

**1.5 MODULES OF THE SYSTEM**

In our proposed method to develop the efficient system we have different modules that are interlinked to perform all the mentioned tasks of the system properly. These modules are:

**1.5.1 HTML**

**HTML** stands for Hyper Text Markup Language. It is used to design web pages using a markup language. HTML is the combination of Hypertext and Markup language. Hypertext defines the link between the web pages. A markup language is used to define the text document within tag which defines the structure of web pages. This language is used to annotate (make notes for the computer) text so that a machine can understand it and manipulate text accordingly. Most markup languages (e.g. HTML) are human-readable. The language uses tags to define what manipulation must be done on the text.

In the late 1980's , a physicist, Tim Berners-Lee who was a contractor at CERN, proposed a system for CERN researchers. In 1989, he wrote a memo proposing an internet-based hypertext system.

3

**Minor Project Ocean of Notes**

**Tim Berners-Lee** is known as the father of HTML. The first available description of HTML was a document called "HTML Tags" proposed by Tim in late 1991. The latest version of HTML is HTML5.

**Features of HTML**

1) It is a very **easy and simple language**. It can be easily understood and modified. 2) It is very easy to make an **effective presentation** with HTML because it has a lot of formatting tags.

3) It is a **markup language**, so it provides a flexible way to design web pages along with the text.

4) It facilitates programmers to add a **link** on the web pages (by html anchor tag), so it enhances the interest of browsing of the user.

5) It is **platform independent** because it can be displayed on any platform like Windows, Linux, and Macintosh, etc.

6) It facilitates the programmer to add **Graphics, Videos, and Sound** to the web pages which makes it more attractive and interactive.

7) HTML is a case-insensitive language, which means we can use tags either in lower-case or upper-case.

**1.5.2 CSS**

**C**ascading **S**tyle **S**heets, fondly referred to as CSS, is a simple design language intended to simplify the process of making web pages presentable.

CSS handles the look and feel part of a web page. Using CSS, you can control the color of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colors are used, layout designs, variations in display for different devices and screen sizes as well as a variety of other effects.

4

**Minor Project Ocean of Notes**

CSS is easy to learn and understand but it provides powerful control over the presentation of an HTML document.

**Features of CSS**

1) **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.

2) **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.

3) **Easy maintenance** − To make a global change, simply change the style, and all elements in all the web pages will be updated automatically.

4) **Superior styles to HTML** − CSS have 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.

5) **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.

6) **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.

5

**Minor Project Ocean of Notes**

**1.5.3 Bootstrap**

Bootstrap is a free front-end framework for faster and easier web development. Bootstrap includes HTML and CSS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels and many other, as well as optional JavaScript plugins. Bootstrap also gives you the ability to easily create responsive designs.

Responsive web design is about creating web sites which automatically adjust themselves to look good on all devices, from small phones to large desktops.

**Features of Bootstrap**

1) **Easy to use:** Anybody with just basic knowledge of HTML and CSS can start using Bootstrap

2) **Responsive features:** Bootstrap's responsive CSS adjusts to phones, tablets, and desktops 3) **Mobile-first approach:** In Bootstrap 3, mobile-first styles are part of the core framework 4) **Browser compatibility:** Bootstrap is compatible with all modern browsers (Chrome, Firefox, Internet Explorer, Edge, Safari, and Opera)

**1.5.4 Angular**

Angular is an open-source, JavaScript framework written in TypeScript. Google maintains it, and its primary purpose is to develop single-page applications. As a framework, Angular has clear advantages while also providing a standard structure for developers to work with. It enables users to create large applications in a maintainable manner.

Angular” is the catch-all term for the various framework versions out there. Angular was developed in 2009, and as a result, there have been many iterations.

First, there was the original Angular, called Angular 1 and eventually known as AngularJS. Then came Angular 2, 3, 4, 5, until finally, the current version, Angular 11, released on 11/11/2020. Each subsequent Angular version improves on its predecessor, fixing bugs, addressing issues, and accommodating increasing complexity of current platforms.

6

**Minor Project Ocean of Notes**

**Features of Angular**

**1) Custom Components**: Angular enables users to build their own components that can pack functionality along with rendering logic into reusable pieces. It also plays well with web components.

**2) Data Binding:** Angular enables users to effortlessly move data from JavaScript code to the view and react to user events without having to write any code manually. **3) Dependency Injection:** Angular enables users to write modular services and inject them wherever they are needed. This improves the testability and reusability of the same services.

**4) Testing:** Tests are first-class tools, and Angular has been built from the ground up with testability in mind. You will have the ability to test every part of your application—which is highly recommended.

**5) Comprehensive:** Angular is a full-fledged framework and provides out-of-the-box solutions for server communication, routing within your application, and more. **6) Browser Compatibility:** Angular is cross-platform and compatible with multiple browsers. An Angular application can typically run on all browsers (E.g.: Chrome, Firefox) and OSes, such as Windows, macOS, and Linux.

**1.5.5 No SQL(Firebase)**

Firebase is a product of Google which helps developers to build, manage, and grow their apps easily. It helps developers to build their apps faster and in a more secure way. No programming is required on the firebase side which makes it easy to use its features more efficiently. It provides services to Android, IOS, web, and unity. It provides cloud storage. It uses NoSQL for the database for the storage of data.

Features of Firebase:

1. **Incredibly Built-In Analytics:** The analytics dashboard is one of the best features of Firebase, which is equipped with. It is free and can report 500 event types, each with 25

7

**Minor Project Ocean of Notes**

attributes. The dashboard is top-notch for observing user behavior and measuring various user characteristics. Ultimately it helps us to understand how people use our app so that we can better optimize it in the future.

2. **App Development Made Easy:** With Firebase, we can focus our time and attention on developing the best possible applications for our business. The operation and internal functions are very solid. They have taken care of the Firebase Interface. We can spend more time in developing high-quality apps that users want to use.

3. **Growth and User Engagement:** One of the most important aspects of application development is being able to develop and engage with users over time. Firebase has a lot of built in features, which ensures that it is exactly what we do. With the platform leading to commercial apps, it is really at the center of what makes Firebase so great.

4. **Increase Your Earnings:** Of course, the thing about having an app or any other business strategy is that we can increase our earnings. With the feature of AdMob, we can monetize our app, considering the best possible experience for our users. Showing real-time ads to millions of Google advertisers, choosing a format which suits our app, and working with over 40 top ads networks using AdMob Mediation, we can make app development well worth it, while speaking silently.

**1.6 SCOPE**

The scope of the project is as follows:

❖ To make notes available to the students in an easy way (provided stable internet connection).

❖ To make sure that it is viable system for all the users.

❖ To make sure that the authenticity is maintained.

❖ To make sure that no unwanted or wrong data is provided. The notes provided to the user should be reliable.

❖ To facilitate the download and upload feature for the users.

❖ To provide user with all type of notes formats:

• Word Document

• Power Point Presentations

8

**Minor Project Ocean of Notes**

• Portable Document Format • Handwritten Notes(pdfs)

9

**Minor Project Ocean of Notes**

**CHAPTER-2**

**LITERATURE SURVEY**

**Minor Project Ocean of Notes**

**LITERATURE SURVEY**

**2.1 EXISTING SYSTEM**

The existing systems like our proposed models are as follows:

**1. Hopin**

**2. Student-Focus**

**3. Lecture-Notes**

We will now Study the three of them in detail.

**1. Hopin**

Hopin enables over 37 million students to Discover, Share & Subscribe to their course contents online from trusted peers, faculty and academic publishers together on a dedicated platform at affordable prices. A team of Engineers, Designers and Marketers who’ve rallied together to build India’s largest community of college students enabling you to learn and ace your exams - Faster, Cheaper, Better.

**2. Student-Focus**

They have collected some notes, study materials and question papers based on focus areas for plus two students for all subjects. Students preparing for the Higher Secondary examination, UGC of all semester can get benefitted from these materials. They're providing efficient notes from various universities, you can easily download it. It is a One Stop Solution for all study materials.

**3. Lecture Notes**

Lecture Notes enables people in the field of education to connect. It provides the features of uploading and downloading note by both the students and teachers. It gives the facility of login system which ensures the authenticity of the system. Along with notes, it provides video lectures, sample papers, MCQs for the students.

10

**Minor Project Ocean of Notes**

**2.2COMPARATIVE CHART**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CRITERIA HOPINSTUDY FOCUSLECTURE  OCEAN OF  NOTES  NOTES | | | | |
| Free | **YES** | **YES** | **YES** | **YES** |
| Online | **YES** | **YES** | **YES** | **YES** |
| Sample papers | **YES** | **YES** | **YES** | **YES** |
| Previous year  question papers | **YES** | **YES** | **NO** | **YES** |
| Practical files | **NO** | **YES** | **NO** | **YES** |
| Reliability of  content | **NO** | **YES** | **YES** | **YES** |
| Ease of access | **YES** | **YES** | **YES** | **YES** |
| Ad-banner | **YES** | **YES** | **YES** | **NO** |
| Ensured | **YES** | **YES** | **YES** | **YES** |
| Clutter-free | **YES** | **YES** | **YES** | **YES** |
| Login Required | **YES** | **No** | **YES** | **YES (For reading purpose no need to login)** |
| Internet  connection  Needed | **YES** | **YES** | **YES** | **YES** |

**Table 2.1: Comparative Chart**

11

**Minor Project Ocean of Notes**

**2.3 PROPOSED SYSTEM**

The proposed system is a basic level of the existing systems. More functionalities can be added to it in future. This system emphasizes mainly on the feature of downloading and uploading notes or study material.

In this, the user can access the web application anytime, anywhere provided there is stable network connectivity. Each user will have their accounts if they want to download or upload notes.

Login system enables us to know that the system is in demand and needed, as a user can view any notes online and study from there. But if they want to download it or upload something, they need to be an authorized user. The type of study material is up to users that what they want to study.

For any query or any feedback, they can contact us in the section provided in the name of ‘Contact us’. Even if they demand for notes for any subject, they can contact us, and we can arrange it for them. This would add to the productivity of the system and make its functionality better.

The architecture of the system is simple and small because there is always a scope of improvement. Any functionality which is needed by the users would be taken under consideration in future, that might add value to our system.

12

**Minor Project Ocean of Notes**

**CHAPTER-3**

**REQUIREMENT ANALYSIS**

**Minor Project Ocean of Notes**

**REQUIREMENT ANALYSIS**

**3.1 METHOD USED FOR REQUIREMENT ANALYSIS**

❖ The major requirement for the proposed system is stable internet connection to access the web application.

❖ For viewing purpose, the user can just navigate through the download section and read the file (notes) by clicking View.

❖ In order to download or upload the file (notes), the user must be an authenticated first. ❖ The new user must create an account and then use the services of the proposed system. ❖ If the user already has an account, then he/she must login-in and then use the services of the proposed system.

**3.2 DATA REQUIREMENTS**

For the proposed system, data is the content uploaded by the users. So, this requirement is fulfilled by the user itself as they upload the study material. As the number users increases and the content uploaded by them increases, the data is thereby collected and offered back the users.

So, in this system data requirements are achieved after the deployment of the project. **3.3 FUNCTIONAL REQUIREMENTS**

❖ **SIGNUP**: New user have to sign-up for accessing the file on site.

❖ **REGISTERED USERS**: They must simply login to the site and access the files. ❖ **ADMIN:** Admin has power to edit or change any file by using his admin id. ❖ **UPLOAD:** User can upload their files by logging - in with their user id and password. ❖ **DOWNLOAD:** User can download files by login with their user id and password. ❖ **READ**: User can read the file by login with their user-id and password. ❖ **LOGOUT**: After using the website user can logout safely.

13

**Minor Project Ocean of Notes**

**3.4 NON-FUNCTIONAL REQUIREMENTS**

❖ **PERFORMANCE**: any low-end systems can handle the site it is not affect the user experience

❖ **SCALABILITY**: Sessions of different users can’t affect another user. ❖ **COMPATIBILITY**: Website is compatible to almost every browsers and devices present.

❖ **RELIABILITY**: Website is live 24\*7 and 85%chances of not failing. ❖ **MAINTAINABILITY**: On failure site will be live again in timespan of 24h and every day at time of 12:00am to 1:00pm is for maintenance there is slight problems user can face between maintenance period.

**3.5 SYSTEM REQUIREMENT**

**3.5.1 HARDWARE SPECIFICATION**

It should run on every standard pc currently in use with minimum system specifications as follows-

❖ **PROCESSOR:** Pentium IV or higher

❖ **RAM:** 256mb

❖ **SECONDARY MEMORY:** 1gb free space

❖ **SCREEN RESOLUTION:** 720p

14

**Minor Project Ocean of Notes**

**3.5.2 SOFTWARE SPECIFICATION**

The minimum Software specification for this web application is:

❖ **OPERATING SYSTEM:** Window XP/Window vista /Window 7 ultimate or newer /AC 8.0 or Newer / Linux distribution i.e.; Ubuntu / Fedora /Linux Mint /Debian or more / Android

❖ **LANGUAGE:** HTML, CSS.

❖ **WEB BROWSER:** Google Chrome

15

**Minor Project Ocean of Notes**

**CHAPTER-4**

**DESIGN**

**Minor Project Ocean of Notes**

**DESIGN**

**4.1 DESIGN**

Software design is a process to transform user requirements into some suitable form, which helps the programmer in software coding and implementation. Software design is the first step in SDLC (Software Design Life Cycle), which moves the concentration from problem domain to solution domain. It tries to specify how to fulfil the requirements mentioned in SRS.

The main aim of design engineering is to generate a model which shows firmness, delight and commodity. Software design is an iterative process through which requirements are translated into the blueprint for building the software.

The attributes of design name as 'FURPS' are as follows:

❖ Functionality: It evaluates the feature set and capabilities of the program. ❖ **Usability:** It is accessed by considering the factors such as human factor, overall aesthetics, consistency, and documentation.

❖ **Reliability:** It is evaluated by measuring parameters like frequency and security of failure, output result accuracy, the mean-time-to-failure (MTTF), recovery from failure and the program predictability.

❖ **Performance:** It is measured by considering processing speed, response time, resource consumption, throughput and efficiency.

❖ **Supportability:** It combines the ability to extend the program, adaptability, serviceability. These three terms define the maintainability.

❖ **Testability,** compatibility, and configurability are the terms using which a system can be easily installed and found the problem easily.

❖ **Supportability** also consists of more attributes such as compatibility, extensibility, fault tolerance, modularity, reusability, robustness, security, portability, scalability.

16

**Minor Project Ocean of Notes**

**4.2 SOFTWARE REQUIREMENT SPECIFICATION**

Software Requirement Specification (SRS) is a technical specification of requirements for the software product. SRS represents an overview of products, features and summaries the processing environments for development operation and maintenance of the product. The goal of the requirement specification phase is to produce the software specification document also called requirement document.

**REQUIREMENT SPECIFICATION**

This requirement specification must have the system properties. Conceptually, every SRS should have the components:

• Functionality

• Performance

• Design constraints imposed on an implementation.

• External interfaces

**4.2.1 SUPPLEMENTARY SPECIFICATION**

The purpose of this Specifications is to define requirements of the system. This Supplementary Specification lists the requirements that are not readily captured in the use cases of the use-case model. The Supplementary Specifications and the use-case model together capture a complete set of requirements on the system. This specification defines the non-functional requirements of the system, such as reliability, usability, performance, and supportability as well as functional requirements that are common across several use cases. The non-functional requirements of the software are:

I. **ACCESS SECURITY**: It is evaluated by measuring parameters like frequency and security of failure, output result accuracy, the mean-time-to-failure (MTTF), recovery from failure and the program predictability.

17

**Minor Project Ocean of Notes**

II. **AVAILABILITY**: It is accessed by considering the factors such as human factor, overall aesthetics, consistency, and documentation. Unless the system is non-operational, the system shall present a user with notification informing them that the system is unavailable.

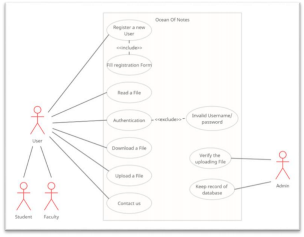
III. **EFFICIENCY**: The system restart cycle must execute completely in less than 60 seconds. Routine maintenance that is executed while users are active shall not cause a perceptible increase in response time for any function of more than 5% over the response time when no maintenance process is executing.

IV. **USABILITY**: The new product shall be easy to use by any age group. The product shall be self-explanatory and intuitive.

**4.2.2 USE CASE MODEL**

A use-case model is a model of how different types of users interact with the system to solve a problem. As such, it describes the goals of the users, the interactions between the users and the system, and the required behavior of the system in satisfying these goals.

In software and system engineering, a use case is a list of actions or event steps, typically defining the interactions between a role (known in the Unified Modelling Language as an actor) and a system, to achieve a goal. The actor can be a human or other external system.



**Figure 4.1 Use Case Diagram**

18

**Minor Project Ocean of Notes**

So, in brief, the purposes of use case diagrams can be as follows:

● Used to gather requirements of a system.

● Used to get an outside view of a system.

● Identify external and internal factors influencing the system.

● Show the interacting among the requirements are actors.

**4.3 CLASS DIAGRAM**

Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application. Class diagram describes the attributes and operations of a class and the constraints imposed on the system. Class diagram shows a collection of classes, interfaces, associations, orations, and constraints. It is also known as a structural diagram.

The purpose of class diagram is to model the static view of an application. Class diagrams are the only diagrams which can be directly mapped with object-oriented languages and thus widely used at the time of construction. UML diagrams like activity diagram, sequence diagram can only give the sequence flow of the application, however class diagram is a bit different. It is the most popular UML diagram in the coder community.

The purpose of the class diagram can be summarized as −

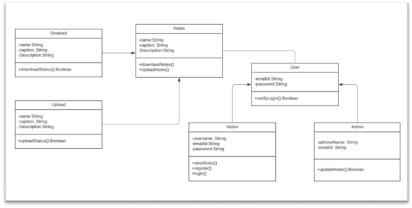
❖ Analysis and design of the static view of an application.

❖ Describe responsibilities of a system.

❖ Base for component and deployment diagrams.

❖ Forward and reverse engineering

19

**Minor Project Ocean of Notes **

**Figure 4.2 Class Diagram**

**4.4 DATA FLOW DIAGRAM**

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. It shows how data enters and leaves the system, what changes the information, and where data is stored.

DFD may be used to perform a system or software at any level of abstraction. In fact, DFDs may be partitioned into levels that represent increasing information flow and functional detail. Levels in DFD are numbered 0, 1, 2 or beyond. There are primarily three levels in the data flow diagram, which are: 0-level DFD, 1-level DFD, and 2-level DFD.

**Data Flow Diagram Symbols**

❖ **Process**

Processes are represented by circle. The name of the process is written into the circle. The name of the process is usually given in such a way that represents the functionality of the process. More detailed functionalities can be shown in the next Level if it is required.

20

**Minor Project Ocean of Notes** ❖ **External Entity**

External entities are represented by a rectangle and the name of the external entity is written into the shape. These send data to be processed and again receive the processed data. 

❖ **Data Flow**

Data flows are shown as a directed edge between two components of a Data Flow Diagram. Data can flow from external entity to process, data store to process, in between two processes and vice versa.

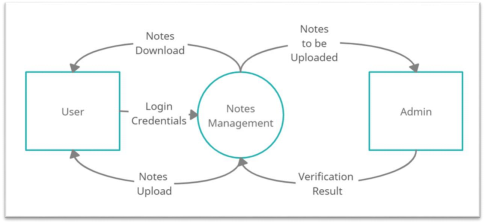


❖ **Data Store**

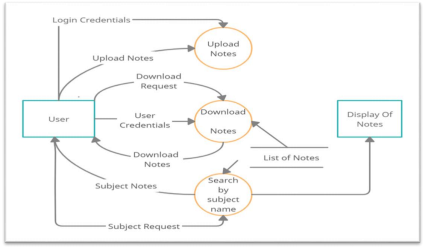
Data stores are represented by a left-right open rectangle. Name of the data store is written in between two horizontal lines of the open rectangle. Data stores are used as repositories from which data can be flown in or flown out to or from a process.



21

**Minor Project Ocean of Notes **

**Figure 4.3 Data Flow Diagram (Level-0)**

**Figure 4.4 Data Flow Diagram (Level-1)**

22

**Minor Project Ocean of Notes**

**CHAPTER-5**

**SYSTEM MODELLING**

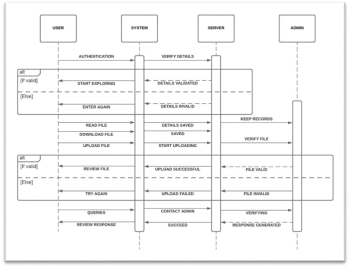
**Minor Project Ocean of Notes**

**SYSTEM MODELLING**

**5.1 SEQUENCE DIAGRAM**

A Sequence diagram is an interaction diagram that show processes operate with one another and in what order. It is a construct of a Message Sequence Chart. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development. Sequence diagram are sometimes called event diagrams or event scenarios. The purpose of a sequence diagram in UML is to visualize the sequence of a message flow in the system.

The sequence diagram shows the interaction between two lifelines as a time-ordered sequence of events. A sequence diagram shows, as parallel vertical lines (lifelines), different processes or objects that live simultaneously, and as horizontal arrows, the messages exchanged between them, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner.



**Figure 5.1 Sequence Diagram**

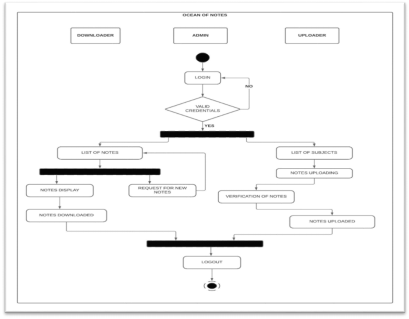
23

**Minor Project Ocean of Notes**

**5.2 ACTIVITY DIAGRAM**

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. The control flow is drawn from one operation to another. The basic purposes of activity diagrams is similar to other four diagrams. It captures the dynamic behavior of the system. Other four diagrams are used to show the message flow from one object to another, but activity diagram is used to show message flow from one activity to another.

An activity diagram is a behavioral diagram i.e. it depicts the behavior of a system. An activity diagram portrays the control flow from a start point to a finish point showing the various decision paths that exist while the activity is being executed.

**Figure 5.2 Activity Diagram**

24

**Minor Project Ocean of Notes**

**5.3 ENTITY-RELATIONSHIP MODEL**

The ER or (Entity Relational Model) is a high-level conceptual data model diagram. Entity Relation model is based on the notion of real-world entities and the relationship between them. ER modelling helps you to analyze data requirements systematically to produce a well-designed database. So, it is considered a best practice to complete ER modelling before implementing your database.

Entity relationship diagram displays the relationships of entity set stored in a database. In other words, we can say that ER diagrams help you to explain the logical structure of databases. At first look, an ER diagram looks very similar to the flowchart. However, ER Diagram includes many specialized symbols, and its meanings make this model unique.

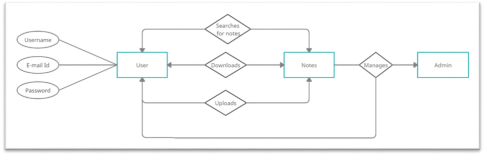
**Components of the ER Diagram**

This model is based on three basic concepts:

• Entities

• Attributes

• Relationships

**Figure 5.3 Entity-Relationship Diagram**

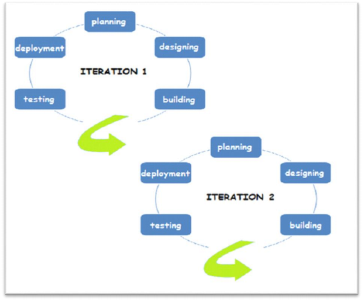
25

**Minor Project Ocean of Notes**

**5.4 System Model**

In this system, we have used agile model. 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 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.

**Figure 5.4 Agile Model**

26

**Minor Project Ocean of Notes**

❖ **Phases of Agile Model:**

Following are the phases in the Agile model are as follows:

1. Requirements gathering.

2. Design the requirements.

3. Construction/ iteration

4. Testing/ Quality assurance

5. Deployment

6. Feedback

❖ **Benefits of Agile model**

1. Frequent Delivery

2. Face-to-Face Communication with clients.

3. Efficient design and fulfils the business requirement. 4. Anytime changes are acceptable.

5. It reduces total development time.

27

**Minor Project Ocean of Notes**

**CHAPTER -6**

**CONCLUSION AND FUTURE WORK**

**Minor Project Ocean of Notes**

**CONCLUSION AND FUTURE WORK**

**6.1 CONCLUSION**

The project mainly focuses on the welfare of both the students and the faculties. Its is safe and secure. Its ease of access makes wide range of users. Verification of the study material makes only reliable data to be reached to the respective user.

Students will have great advantage on it as it provides the study material free of costs. They will be able to view, download and upload notes, thus make use of the product, and also will help others in the process.

**6.2 LIMITATIONS OF PROJECT**

There are certain limitations in this project which are as follows:

1. If a user fails to remember his or her username or password, then “Forgot password” feature is not available.

2. For Downloading purpose, the person will have to go to encounter the “View” section first.

**6.3 FUTURE ENHANCEMENT**

Apart from the features which the system delivers us, there is still a scope for improvement and adding on of new features.

❖ One such feature is, that we can make use of the uploading feature for not only uploading the notes but also the assignments.

❖ Setting the deadline for both submission and evaluation along with the marks displaying feature after the evaluation., this can be added as a new feature.

❖ To make it different to some extent, we can add a new feature for avoiding procrastination. We can add a feature of “REMINDER!!” message, through email, which will remind the

28

**Minor Project Ocean of Notes**

user of the pending assignment and that can also be set by the user that how many reminders does the user wants and on what date.

29

**Minor Project Ocean of Notes**

**CHAPTER – 7**

**BIBLIOGRAPHY AND REFERENCES**

**Minor Project Ocean of Notes**

**BIBLIOGRAPHY AND REFERENCES**

**7.1 REFERENCES**

**https://search.yahoo.com/search?fr=mcafee&type=E210US714G0&p=stackov erflow**

**https://angular.io/docs**

**https://firebase.google.com/docs**

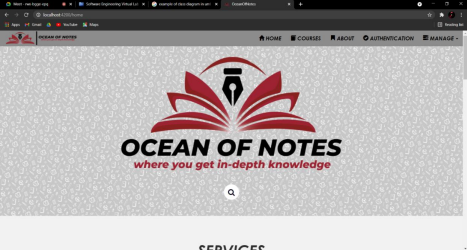
**https://www.w3schools.com/**

30

**Minor Project Ocean of Notes**

**7.2 SNAPSHOT**

**1. HOME PAGE**

**Figure 7.1 Home page**

**2. AUTHENTICATION (SIGN-UP)**

**Figure 7.2 Authentication (Sign-up)**

31

**Minor Project Ocean of Notes**

**3. AUTHENTICATION (SIGN-IN)**

**Figure 7.3 Authentication (Sign-in)**

**4. ACCOUNT PAGE**

**Figure 7.4 Account page**

32

**Minor Project Ocean of Notes**

**5. COURSES PAGE**

**Figure 7.5 Courses page**

**6. COURSES – UPLOAD**

**Figure 7.6 Course - Upload**

33

**Minor Project Ocean of Notes**

**7. COURSES – DOWNLOAD**

**Figure 7.7 Course - Download**

**8. ABOUT PAGE**

**Figure 7.8 About page**

34

**Minor Project Ocean of Notes GLOSSARY**

**Minor Project Ocean of Notes** ❖**GLOSSARY**

|  |  |
| --- | --- |
| **TERM** | **DEFINITION** |
| Admin | The person who manages the project. |
| User | The person who uses the project. |
| System | Project |
| Product | Project |
| SRS | Software Requirement Specification |
| pdf | Portable Document Format |
| SDLC | Software Development Life Cycle |

35