INSTITUTE FOR ADVANCED COMPUTING AND SOFTWARE DEVELOPMENT(IACSD), AKURDI, PUNE

Documentation On

IT GALAXY

PG-DAC March 2023

Submitted By:

Group No: 01

Roll No.	Name:
233001	Abdul Ahad Sheikh
233107	Tushar Pawar

Mrs. Sonali Mogal
Project Guide

Mr. Rohit Puranik
Centre Coordinator

ABSTRACT

A blog, a complete weblog or webzine, an online journal in which an individual, group or business submits entries about their actions, thoughts, or beliefs.

Blogs primarily act as news filters, collecting various online sources and adding short comments and internet links. Other blogs focus on providing original material. Some blogs focus on providing original material. Blogging is the process of writing blog material. Although content is mostly written, images, audio, and video are important elements of many blogs. The "blogosphere" is the online world of blogs. This report is an overview of how to build a blog website using the MERN stack.

ACKNOWLEDGEMENT

We express our deep sense of gratitude towards Mr. Rohit Puranik (Center Coordinator, IACSD, Pune) and Mrs. Sonali Mogal (Guiding Faculty, IACSD, Akurdi, Pune) for their valuable support throughout the project.

We are deeply indebted to them for their guidance, encouragement and deep concern for our project. Without their critical evaluation and suggestions at every stage of the project, this project could never have reached its present form.

Last but not the least we thank the entire faculty and the staff members of IACSD, Akurdi, Pune for their support.

_ Tushar Prakash Pawar

_ Abdul Ahad Sheikh

(PG-DAC, March-2023)

CERTIFICATE

This is to certify that the project work under the title '**IT Galaxy**' is done by Abdul Ahad Sheikh and Tushar Prakash Pawar in the fulfillment of the requirement for Post Graduate Diploma in Advanced Computing Course.

Mrs. Sonali Mogal Puranik

Project Guide

Mr. Rohit

Course Co-Coordinator

Date: 31/08/2023

Table of Content

Introduction

A website is a file stored on a server, the computer that hosts the website (a fancy term for "file storage"). These servers are connected to a huge network called the Internet. A browser is a computer program that loads websites through an Internet connection, such as Google Chrome or Internet Explorer, and the computer used to access these websites is referred to as a "client". To access a website, you need to know its web protocol (IP) address. An IP address is a unique sequence of numbers. Each device has an IP address that distinguishes it from the billions of websites and devices connected through the Internet. Although you can access a website using an IP address, most Internet users prefer to use a domain name or use a search engine. HTTP (Hypertext Transfer Protocol) connects you and your website requests to a remote server that stores all website data. A set of rules (protocols) that define how messages are sent over the Internet. This allows you to navigate between the site pages and the website. When you type a website into your web browser or search for something through a search engine, HTTP provides a framework so that the client (computer) and server can speak the same language when they make requests and responses to each other over the Internet. It's essentially the translator between you and the Internet — it reads your website request, reads the code sent back from the server, and translates it for you in the form of a website. Front-end (or client-side) is the side of a website or software that you see and interact with as an Internet user. When website information is transferred from a server to a browser, front-end coding languages allow the website to function without having to continually "communicate" with the Internet. Front-end code allows users to interact with a website and play videos, expand or minimize images, highlight text, and more. Web developers working on front-end coding work on client-side development. The backend (or server side) is an aspect you don't see when using the internet. This is a digital infrastructure, and to non-developers, it looks like a bunch of numbers, letters and symbols. There are more internal programming languages than interface languages. This is because on the front end the browser understands only HTML, CSS and JavaScript, whereas the server (on the back end) can be configured to understand almost any language.

1.1 Purpose

The purpose of this document is to define the requirements for an IT site. The site will allow users to create an account, write blog posts related to IT sector, and interact with other users through comments and private messages. The web application is to be developed for where we invite writers and readers from all over the world to come together and share their thoughts, ideas, and experiences on an IT related topic. Our mission is to create a space for people to express themselves, share their expertise, and engage in meaningful conversations. We believe that everyone has a unique perspective to offer, and that by sharing our ideas, we can learn from one another and broaden our horizons.

1.2 Scope

The site will consist of a web application accessible through a web browser. Users will be able to create and manage their own accounts, create blog posts, and comment on other user posts, and can use private message feature to communicate with each other.

1.3 Definitions, Acronyms, and Abbreviations

IT Information Technology
SRS Software Requirement Specification
POJO Plain Old Java Object

1.4 References for Requirement Analysis and Design

- Google for problem solving
- http://www.javaworld.com/javaworld/jw-01-1998/jw-01-Credentialreview.html
- Database Programming with JDBC and Java by O'Reilly
- https://www.geeksforgeeks.org/how-to-connect-mongodb-with-spring-boot/

1.5 Overview

1.5.1 Product Perspective:
This site will be a standalone web application, designed to be easily accessible and usable for users of all skill levels.

1.5.2 Product Features:
✓ User registration and login
✓ Ability to create and edit blog posts
✓ Commenting system for users to interact with each other
✓ Search functionality to allow users to find posts by keyword or tag
✓ User profile pages to showcase personal information and recent activity

✓ Moderation tools to allow for easy management of content and users
✓ Responsive design to ensure compatibility across devices

1.5.3 User Classes and Characteristics
The blogging site is designed to be accessible to users of all skill levels. Users may include individuals, organizations, or businesses looking to create and share content online.

1.5.4 Operating Environment
The blogging site will be accessible through a web browser on a variety of devices, including desktop computers, laptops, tablets, and smartphones. The site will be hosted on a secure server with regular backups and maintenance.

1.5.5 Objective
The objective of a blogging site can vary depending on the goals of the blogger, but some common objectives may include:
1. Sharing information: A blog can be used to share valuable information on a particular topic or niche, whether it's information related to the IT sector, tips, opinions, reviews, or insights.
2. Building a community: By consistently publishing quality content, bloggers can attract a loyal following of readers who engage with their posts, leave comments, and share their content with others.
3. Establishing authority: Bloggers can use their platform to establish themselves as experts or thought leaders in their field by sharing their knowledge and experiences.
4. Generating leads: A blog can be an effective way to attract potential customers or clients to a business or service by providing valuable information that addresses them needs or pain points.
5. Personal branding: A blog can serve as a platform for individuals to establish them personal brand, showcase their skills or talents, and build a professional reputation.

Table of Content

1. List of Functions:

Sr. No.	Function ID	Name of the Function
1	F - 01	User Registration
2	F - 02	Log In – Log Out
3	F - 03	Blog Post Creation
4	F - 04	Blog Editing
5	F - 05	Blog Deleting
6	F - 06	Commenting System
7	F – 07	Tagging
8	F - 08	Searching
9	F - 09	User Profile Page
10	F - 10	Moderation Tools
11	F - 11	Responsive Design

2. Function Descriptions

i. Function ID : F – 01

a. Purpose : i. To provide unique identification to each user.
: ii. The track of users will be stored.
: iii. User registration is to sign up user for subscriptions, services and other programs or plans.
b. Entities Involved : User (POJO)
c. Organizational Unit :
d. Frequency : Once while creating new account

ii. Function ID : F – 02

a. Purpose : i. To allow user to Login/Logout of the system.
: ii. User will be logged out if inactive for 30 minutes.
b. Entities Involved : User (POJO)
c. Organizational Unit :
d. Frequency : Daily

iii. Function ID : F – 02

a. Purpose : i. To allow user to create blog to share their ideas and experiences.

b. Entities Involved : Blog Class

c. Organizational Unit :

d. Frequency : Daily

iv. Function ID : F – 04

a. Purposes : i. To allow users to edit there created blog if any mistake occurs.

b. Entities Involved : Blog Class

c. Organizational Unit :

d. Frequency of Use : High

v. Function ID : F – 05

a. Purposes : i. User can delete his/her own blog

b. Entities Involved : Blog Class

c. Organizational Unit :

d. Frequency of Use : Medium

vi. Function ID : F – 06

a. Purposes : i. User or Viewer are allowed to add comment on different blogs.

: ii. The comment can be deleted by viewer who commented that

particular comment or the author of the blog can also delete those comments.

b. Entities Involved : Comment, Blog (POJOs)

c. Organizational Unit :

d. Frequency of Use : Medium

vii. Function ID : F – 07

a. Purposes : i. The User can tag different user using "#".

: ii. User mention another user using tagging system and other user will be notified.

b. Entities Involved : Tagging (POJO)

c. Organizational Unit :

d. Frequency of Use : Medium

viii. Function ID : F – 08

a. Purposes : i. User can search any blog or any another user.

: ii. Accessing of blog or user profile will be faster.

b. Entities Involved : User, Blog (POJO)

c. Organizational Unit :

d. Frequency of Use : High

ix. Function ID : F – 09

a. Purposes : i. User can view his profile,

update his profile and add some personal information in profile.

: ii. Personal Information such as, Git hub profile link, twitter account link, etc.

b. Entities Involved : User (POJOs)

c. Organizational Unit :

d. Frequency of Use : Medium

x. Function ID : F – 10

a. Purposes : i. This online website should give facility to recover account in case of 'Forgot Password Scenario'.

b. Entities Involved : User (POJOs)

c. Organizational Unit :

d. Frequency of Use : Low

xi. Function ID : F – 11

a. Purposes : i. The blogging website would need to be highly concurrent. There will be multiple blogs created at particular point in time. The service should handlethis type of load gracefully and fairly.

b. Entities Involved : - - -

c. Organizational Unit :

d. Frequency of Use : - - -

3. Specific Requirements

3.1 Performance

- The site will be designed for optimal performance, with fast load times and minimal downtime. The server will be capable of handling a large number of simultaneous users.

3.2 Security

- The site will be designed with security in mind, with measures in place to protect user data and prevent unauthorized access.

3.3 Usability

- The site will be designed with a clean and intuitive user interface, with features and functionality easily accessible and understandable to users of all skill levels.

3.4 Maintainability

- The Fashion 24 x 7 (Online Clothes Shopping Website) should be able to maintain with as less efforts & changes as possible.

3.5 Portability

- The specified application must provide portability in order to change components of architecture in case of emergencies.
- It should hazel – free facility to replace the databases to enhance the efficiency in needed in future. Like replacement from MYSQL to Oracle or MYSQL to MongoDB.

3.6 Accessibility

- The online website must be accessible via desktops, laptops, smart devices including mobile phones, tablets etc.
- The UI – UX must not hamper in case of above options. It should remain uniform throughout all the devices.

3.7 Durability

- The overall application should be durable, especially in the terms of data, product availability, and uniform performance over time.

3.8 Other Requirements

Hardware: The application is expected to function on Dell G3 – 15 with 1100 MHz Pre Processor Equivalent or Above, 4 GB RAM, 512 GB HDD.

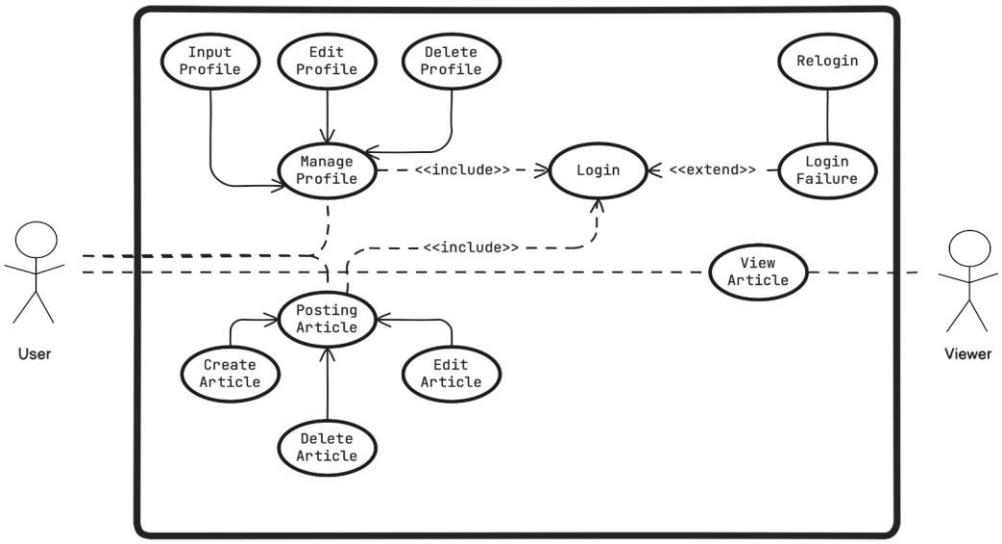
Software: The IT Galaxy Site shall work on Microsoft Windows operating systems family (MS Windows XP & Above). It configures to work with Mongo DB database. This System works on Apache Tomcat server. It uses browser Google Chrome Browser.

4. Glossary:

- Post: An article or piece of content created by a user on the site.
- Comment: A response to a blog post or another user's comment.
- Category: A classification system used to organize content based on topic or them.
- Content: Any type of media created by the user, including text, images, and videos.
- Authentication: The process of verifying a user's identity.

5. UML Diagram:

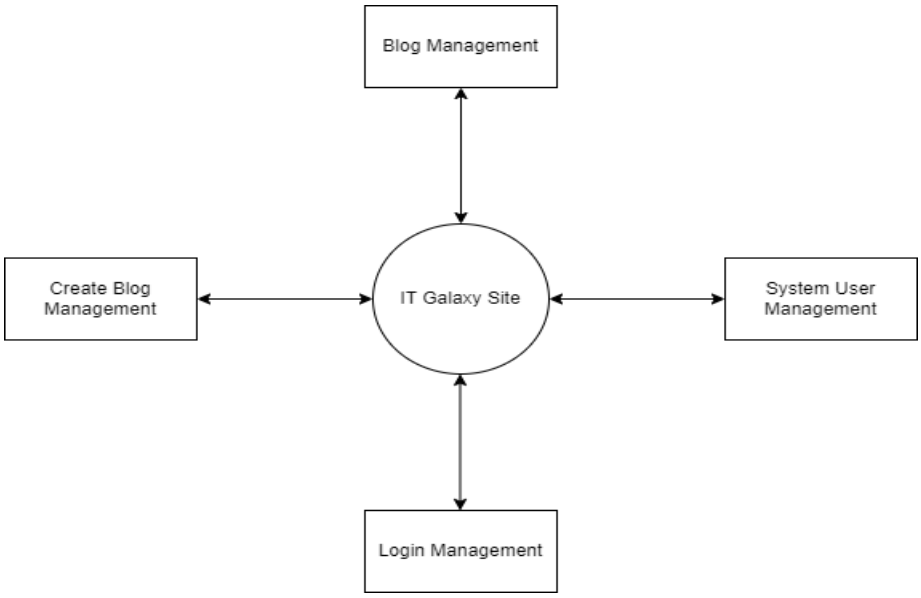
5.1 Use Case Diagram:



Use Case Diagram for IT Galaxy Site

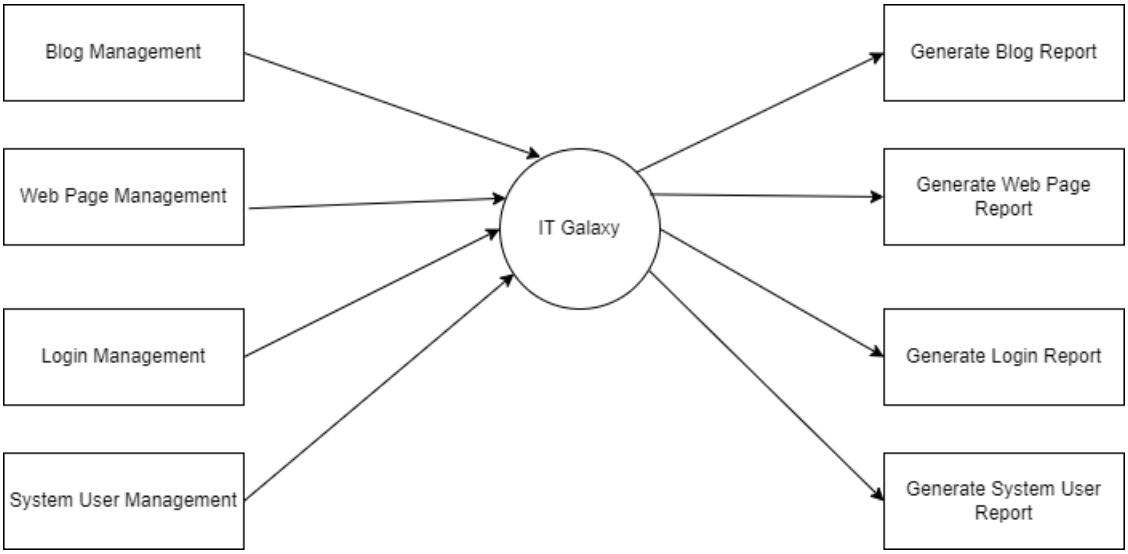
5.2 Data Flow Diagram

5.2.1 Zero-Level DFD Diagram



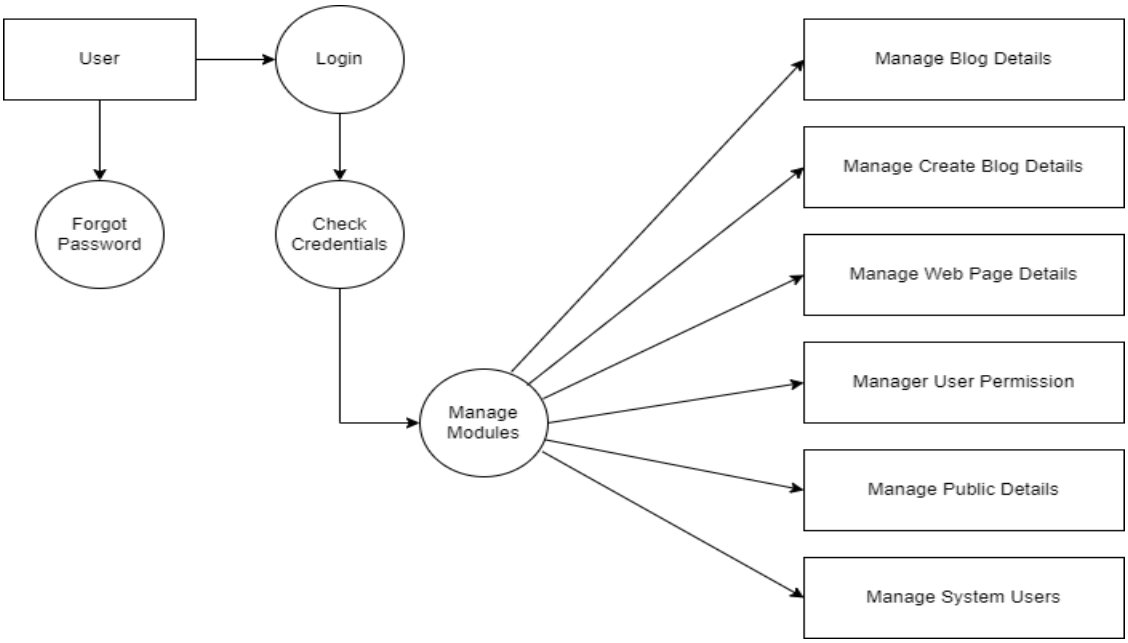
Zero Level DFD Diagram for IT Galaxy Site

5.2.2 First-Level DFD Diagram



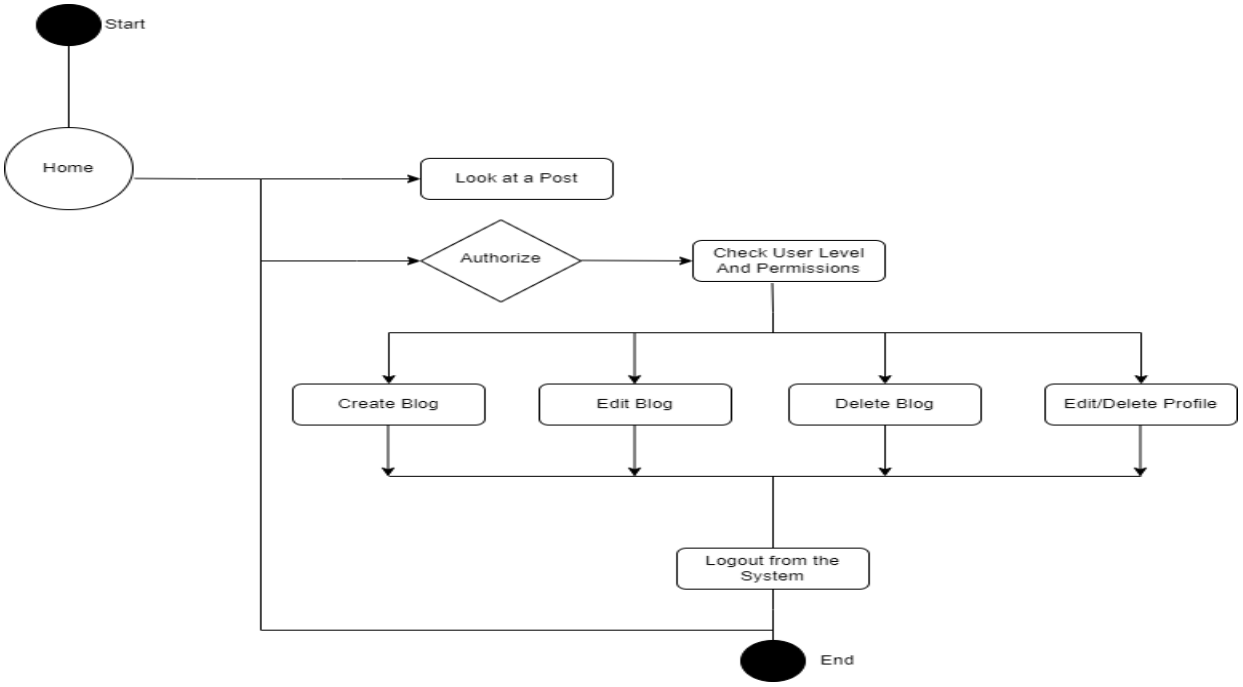
First Level DFD Diagram for IT Galaxy Site

5.2.3 Second-Level DFD Diagram



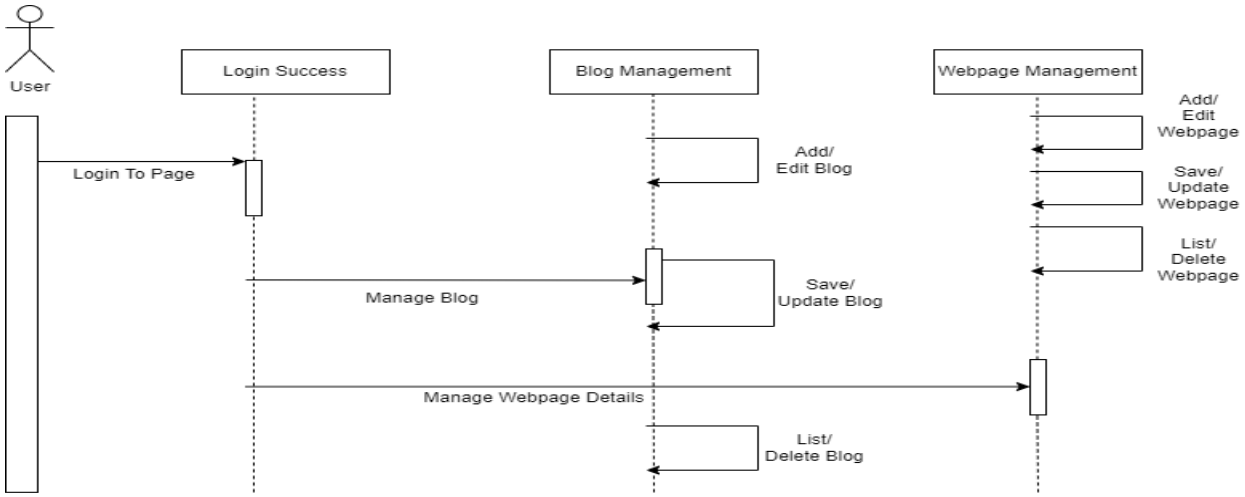
Second Level DFD Diagram for IT Galaxy Site

5.3Activity Diagram



Activity Diagram for IT Galaxy Site

5.4Sequence Diagram



Sequence Diagram for IT Galaxy Site

6. Backend Implementation:

- **Nodejs**

  An open-source, cross-platform runtime environment for JavaScript is Node.js. It is a well-liked tool for practically every project kind! The core of Google Chrome, the V8 JavaScript engine, is run by Node.js outside of the browser. Node.js can be very performant because of this. Without starting a new thread for each request, a Node.js application operates in a single process. Blocking behaviour in Node.js libraries is the exception rather than the rule since libraries in Node.js are typically created using non-blocking paradigms and because Node.js includes a set of asynchronous I/O primitives in its standard library that prevent JavaScript code from blocking.

- **Expressjs**

  The foundational library for several other well-liked Node web frameworks, including the most popular Node web framework, Express It offers methods to: Write handlers for requests using various HTTP verbs at various URL paths (routes). Integrating with "view" rendering engines to produce replies through data-inserting templates establish standard web application settings, such as the port to use for connection and the location of templates used for response rendering. At any stage of the request handling pipeline, add more "middleware" for processing requests.

  $ npm install express

7. Frontend Implementation:

- **Reactjs**

  A JavaScript-based UI development library is called React. It is controlled by Facebook and an open-source development community. React is a popular library in web development even though it isn't a language. The library made its debut in May 2013 and is currently one of the frontend libraries for web development that is most frequently used. Beyond only UI, React includes a number of extensions for supporting the architectural design of complete applications, including Flux and React Native.

- **Axios**

  A promise-based HTTP Client for the browser and node.js is called Axios. It can run in both a browser and Node.js using the same code because it is isomorphic. The client (browser) uses XMLHttpRequests, whereas the server uses the built-in node.js http module.

  $ npm install axios

8. Experimental Results:

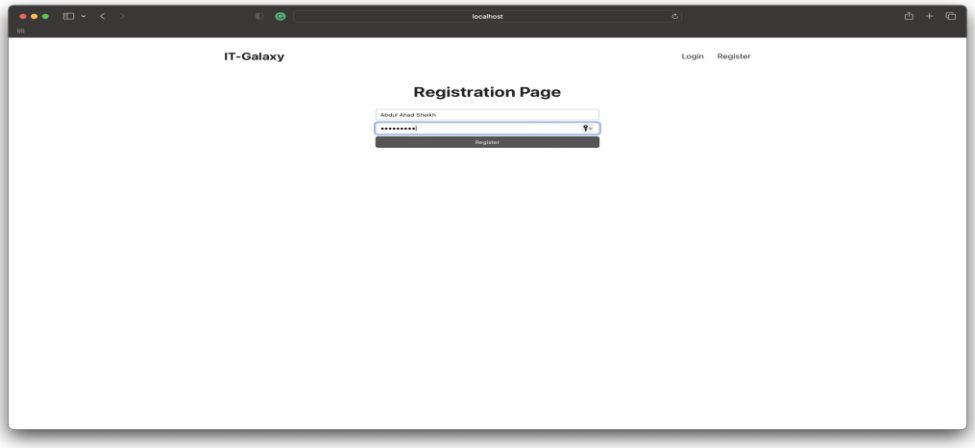


Fig1. Registration Page

Figure 1 shows the user registration page where user can register to blog before login to website

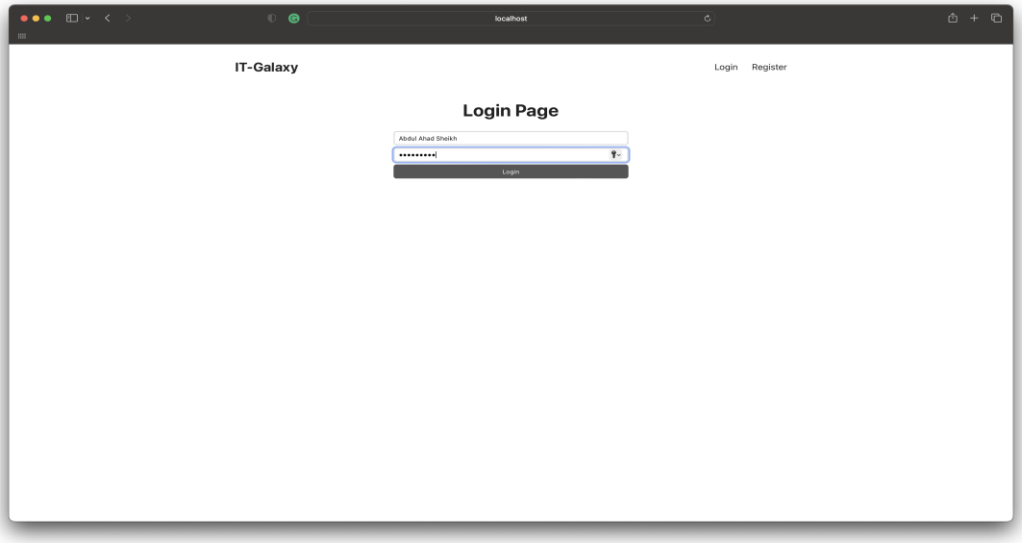


Fig 2. Login Page

Figure 2 shows the user login page where user can login to website

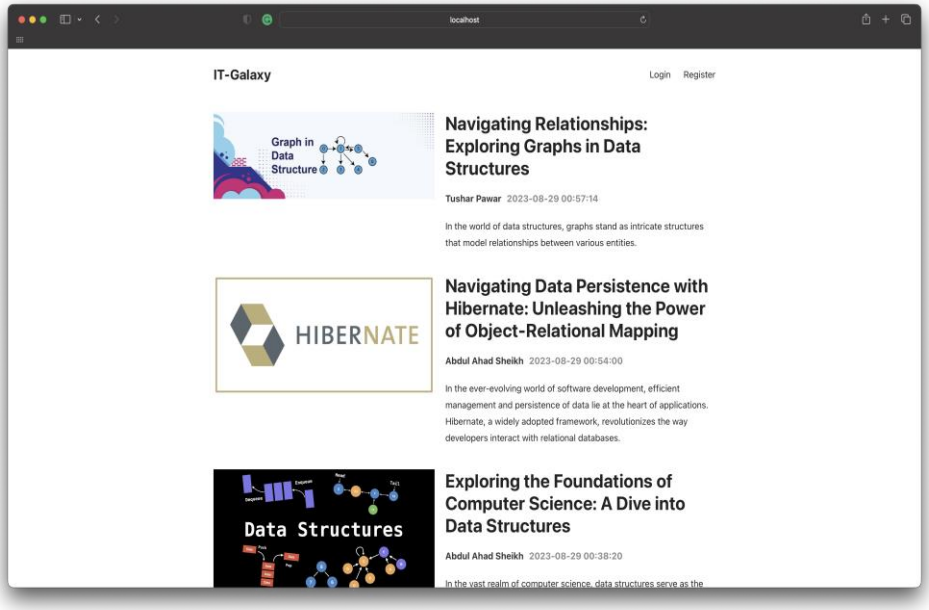


Fig 3. Home Page of IT Galaxy Website

The Figure 4 shows the home page of the website

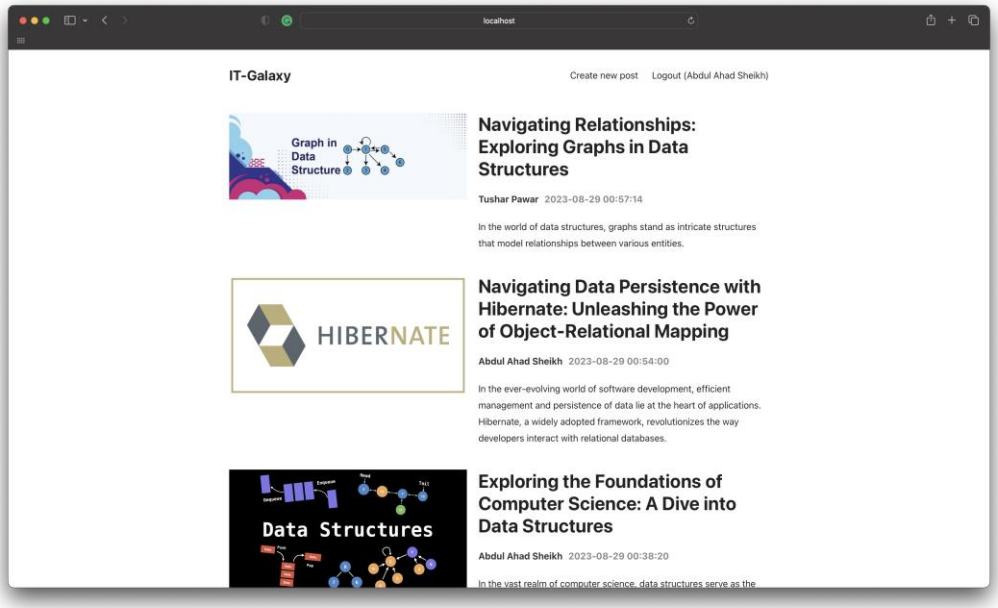


Fig 4. Home Page After Logging-in the System

Figure 4 shows the page after we logged in the website

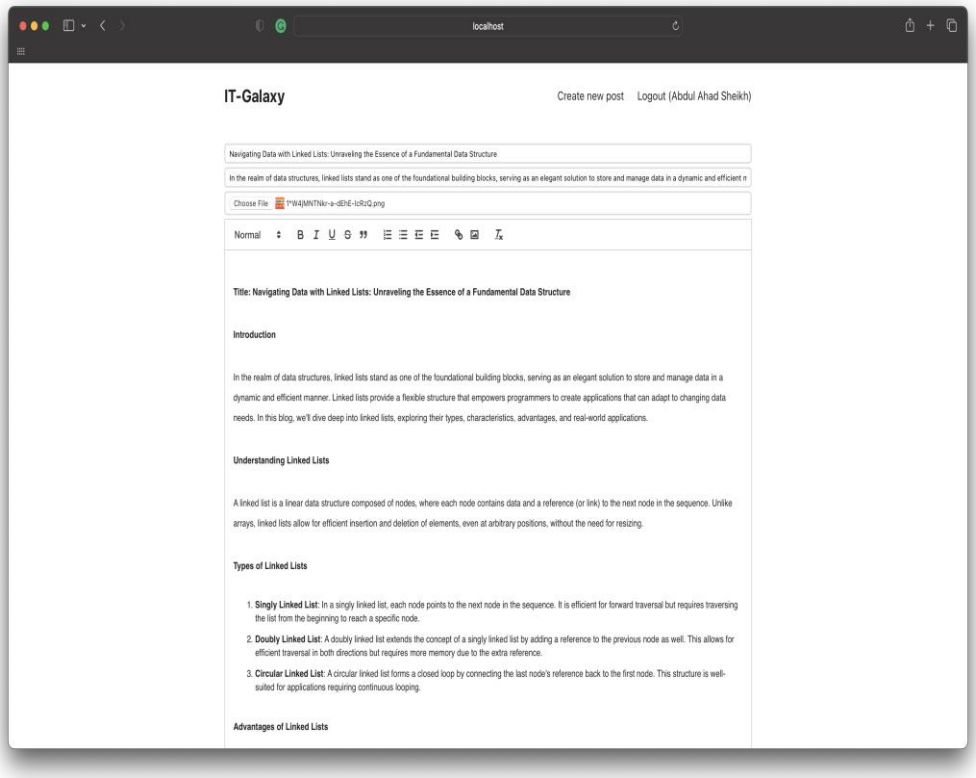


Fig 5. Creating Post in the IT Galaxy Website

Figure 5 shows the page where we can create the blog in blog website

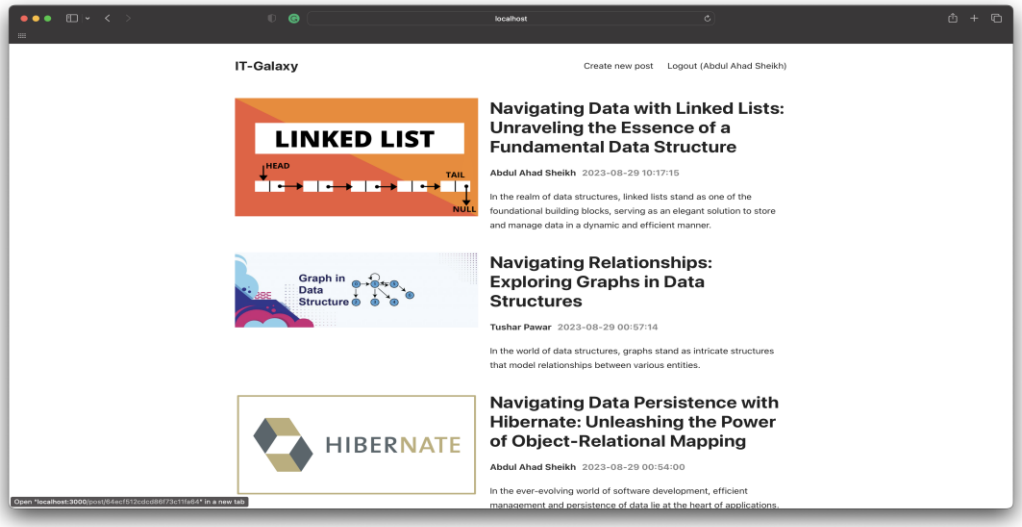


Fig 6. New Blog Created

Figure 6 shows the New Created Blog on the Home page of website.

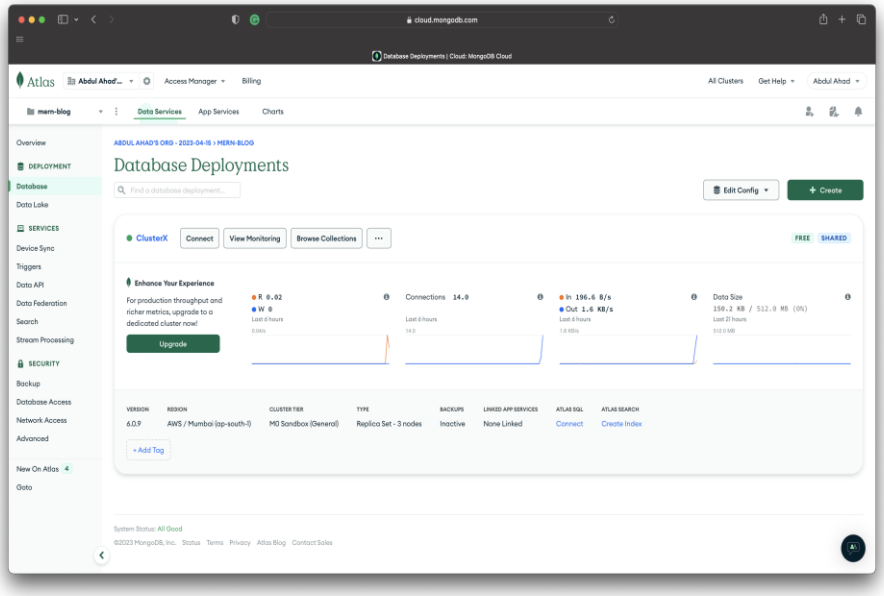


Fig 7. Database Deployment

Figure 7 shows Multi Cloud Database Service MongoDB Atlas

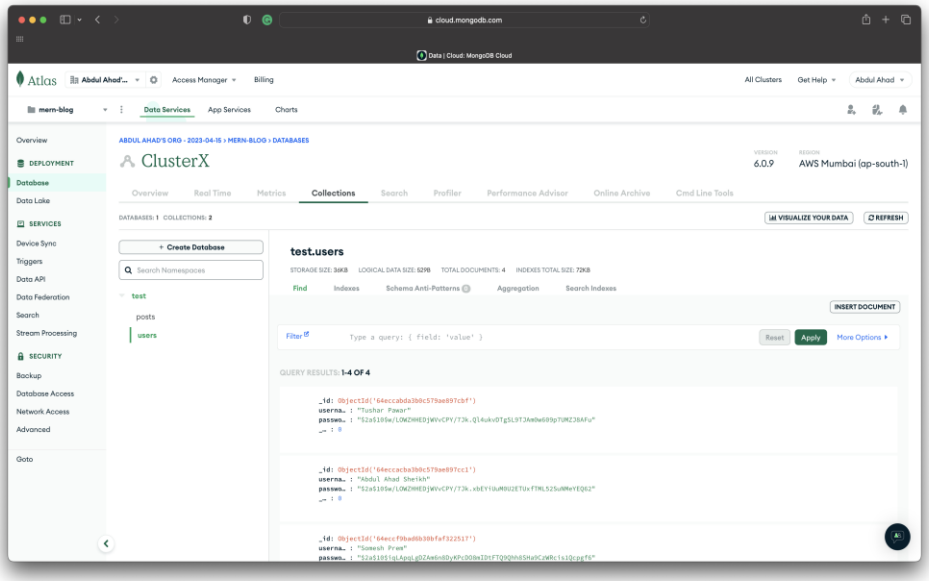


Fig 8. Users Collections

Figure 8 shows all the user's data is stored in user's collection and each data consider as document.

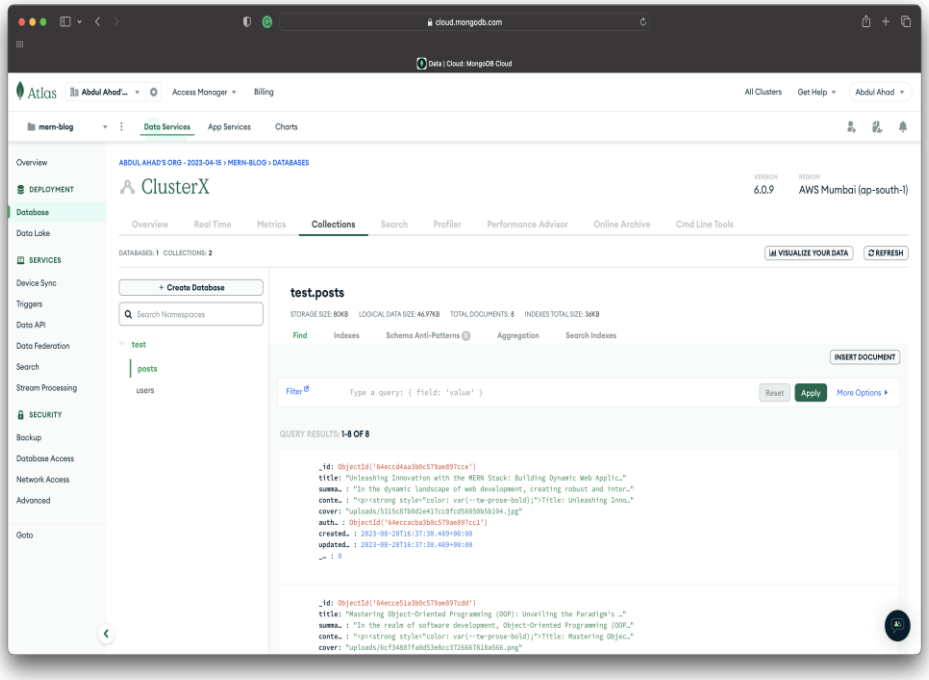


Fig 9. Posts Collections

Figure 9 shows all the posts posted by all the users and users posts is identified by user id in the post document.

9. Conclusion:

In this project the blog website can benefit us as a consumer through valuable information of around the world. We are able to know fundamental concepts and can work on MERN Stack, gain a broad understanding of web, network, server, client, cookies, session, database, front end libraries and backend libraries and are able to create meaningful website. MERN Stack provides developers the huge number of libraries and made easy to build a website. We can create powerful and professional websites by using latest MERN stack.

10. References:

- Blood, R. (2002). Introduction. In J. Rodzvilla (Ed.), We've Got Blog: How Weblogs are Changing Our Culture (pp. ix-xiii). Cambridge MA: Perseus Publishing.
- Cavanaugh, T. (2002). Let slip the blogs of war. In J. Rodzvilla (Ed.), We've Got Blog: How Weblogs are Changing Our Culture (pp. 188-197). Cambridge, MA: Perseus.
- Blood, R. (2002). The Weblog Handbook: Practical Advice on Creating and Maintaining Your Blog. Cambridge MA: Perseus Publishing.
- Döring, N. (2002). Personal home pages on the Web: A review of research. Journal of Computer-Mediated Communication, 7.
- Lasica, J. D. (2001). Blogging as a form of journalism. USC Annenberg Online Journalism Review.
- Google for problem solving
- http://www.javaworld.com/javaworld/jw-01-1998/jw-01-Credentialreview.html
- Database Programming with JDBC and Java by O'Reilly
- https://www.geeksforgeeks.org/how-to-connect-mongodb-with-spring-boot/

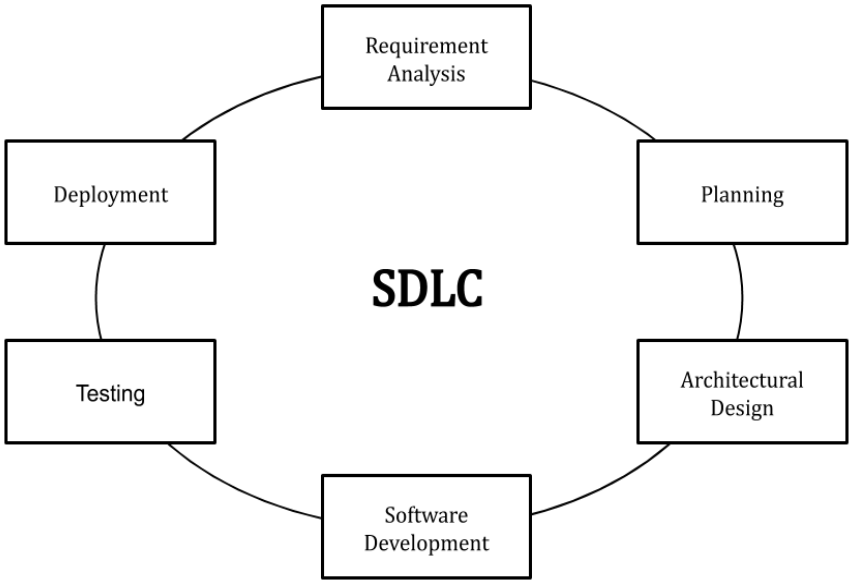
CLOSURE REPORT

11.General Information:

12.1 Productivity:

- A blog can be used to share valuable information on a particular topic or niche, whether its information related to the IT sector, tips, opinions, reviews, or insights.
- By consistently publishing quality content, bloggers can attract a loyal following of readers who engage with their posts, leave comments, and share their content with others.
- Bloggers can use their platform to establish themselves as experts or thought leaders in their field by sharing their knowledge and experiences.
- A blog can serve as a platform for individuals to establish them personal brand, showcase their skills or talents, and build a professional reputation.

12.2 Process used and Process Deviations:



- IT Galaxy Site has been successfully implemented the SDLC Lifecycle in which it went through different phases, which are described as follows:
  1. Requirement Analysis: Basic requirements were gathered as user point of view and analyzed in a proper way to set aim, objective, time constraints of the proposed project.
  2. Planning: To meet the specified requirements, planning phase is conducted in which the various technologies were taken into consideration.
  3. Architectural Design: In this phase, basic flow of the whole project was finalized.

4. Software Development: In this phase of software development, actual code is written in selected programming languages and finalized frameworks.
5. Testing: As the proposed project has been implemented using agile methodology for increasing efficiency of the proposed project. In which, testing is done in parallel with software development.
6. Deployment: The completed project has been deployed on the EC2 instance hosted by Amazon Web Service (AWS).

- During the whole process of execution, as some of the requirements were added in later phase of the Software Development Lifecycle. These changes were effectively considered and revised the structure of the proposed project by adding necessary functionalities.

13. Estimated & Actual Start and End Dates of the Project:

- **Estimated Dates:**
  - Start Date of the Software Development: 1st August, 2023
  - End Date of the Software Development: 25th August, 2023

- **Actual Dates:**
  - Start Date of the Software Development: 11th August, 2023
  - End Date of the Software Development: 31st August, 2023

14. Tools Used:

Following are the tools used effectively during the software development of IT Galaxy Site:

- GitHub: This tool is used to control versions of the application throughout the development phase. In which separate repositories were created so as to maintain modularity throughout the project.
- Visual Studio Code: Visual Studio Code is a streamlined code editor with support for development operations like debugging, task running, and version control. It aims to provide just the tools a developer needs for a quick code-build-debug cycle and leaves more complex workflows to fuller featured IDEs, such as Visual Studio IDE.
- MongoDB Atlas: MongoDB Atlas is a multi-cloud database service by the same people that build MongoDB. Atlas simplifies deploying and managing your databases while offering the versatility you need to build resilient and performant global applications on the cloud providers of your choice.
- JIRA: Jira is used for bug tracking, issue tracking, and project management.

15.Risk Management:

15.1 Risk Identified at the Start of the Project:

- Hosting Platform: At the start of the proposed project, the platform on which the website is going to be hosted was not finalized. The selection of the hosting platform depends on many parameters which will be available at the completion phase. Hence, the decision of selection of the hosting platform was finalized at the later stage of the software development.

15.2 Risk Encountered During Project:

- Security of Data: For encrypting the password we were facing the error in React Js. As password was used for authentication purpose, it is essential to use and we can't display out password, so we need to encrypt it to increase the security.

15.3 Notes on Risk Mitigation:

- Encrypting Password: Later we learned about the bcrypt Js which is used for encryption of password and implemented it in our project. bcrypt.js is a JavaScript library used for hashing passwords and generating salted password hashes using the bcrypt algorithm. Bcrypt is a widely used password hashing function that incorporates a salt to prevent rainbow table attacks and to slow down brute-force attacks.
- 'Risk Mitigation' in software development refers to the process of identifying potentials risks that could impact the success of a software project and taking proactive measures to reduce their likelihood or impact. Following are the basic steps are used to handle risks in software development:
    - Risk Identification
    - Risk Assessment
    - Risk Mitigation Planning
    - Risk Monitoring
    - Risk Response Execution
    - Documentation

16 Size:

16.1 Estimated and Actual Size (in KLOC):

- KLOC, which stands for "Thousand Lines of Code," is a metric used to measure the size or complexity of a software project by counting the number of lines of code in the project. It's worth noting that KLOC is just one of many metrics used to assess software projects, and it's not always a reliable indicator of project quality or complexity on its own.

- Estimated Size: 25000/1000 = 25
- Actual Size: 28827/1000 = 28.827

17 Defects:

17.1 SDLC Stage-Wise Defects:

17.1.1 Requirements Phase:

- Defect Category: Requirements-related defects occur when there are misunderstandings, ambiguities, or inconsistencies in the project requirements.
- Defect Management: Defects in this phase can be minimized through thorough requirements gathering, clear documentation, and regular communication with stakeholders.

17.1.2 Design Phase:

- Defect Category: Design-related defects arise when there are errors or inconsistencies in the system architecture, database design, or overall system design.
- Defect Management: Review the design documents and diagrams to identify potential issues. Conduct design reviews with experienced team members to catch and rectify design flaws.

17.1.3 Implementation (Coding) Phase:

- Defect Category: Coding defects are introduced during the implementation phase due to mistakes in the code logic, syntax errors, or violations of coding standards.
- Defect Management: Utilize code reviews, pair programming, and static code analysis tools to catch coding defects. Automated testing, including unit tests, can help identify issues early in the development process.

17.1.4 Testing Phase:

- Defect Category: Testing uncovers defects related to functionality, performance, security, and compatibility.
- Defect Management: Document defects found during testing, categorize them based on severity and impact, and assign them to the development team for resolution. Regression testing helps ensure that fixed defects don't introduce new issues.

17.1.5 Deployment and Production Phase:

- Defect Category: Production defects occur when issues arise in the live environment that weren't caught during testing.

- Defect Management: Monitor the live system for errors and user feedback. Implement proper error logging and monitoring tools to quickly identify and address production defects.

17 .1.6 Maintenance Phase:

- Defect Category: Maintenance defects emerge after the software is deployed due to changes in the environment, data, or requirements.
- Defect Management: Maintain a feedback loop with users and address defects promptly. Prioritize and manage defects alongside new feature development.

17.2 Distribution of Defects:

Defects classification as major or minor can vary based on project specific criteria and severity levels defined by the development team. Here, in this project, the defects are distributed as:

- Major Defects:
  - Requirements-related defects
  - Design-related defects
  - Coding defects
  - Testing uncovers defects
  - Maintenance defects
- Minor Defects:
  - Performance Defects