**DRIVEONDEMAND**

## A Minor Project Report for the Degree of

# MASTER OF COMPUTER APPLICATIONS

### By

**Akash Chaudhary** (2300290140016)

**Ajay Kaushik**(2300290140014)

**Aisha Sharma** (2300290140013)

**Arnav Kumar Gupta** (2300290140040)

**Under the Supervision of**

**Mr. Shish Pal Jatav**

**Assistant Professor**



**Submitted to**

## DEPARTMENT OF COMPUTER APPLICATIONS

**KIET Group of Institutions, Ghaziabad Uttar Pradesh-201206**

**DECEMBER 2024**

# DECLARATION

We hereby declare that the work presented in this report entitled **“DRIVEONDEMAND"**, was carried out by us. We have not submitted the matter embodied in this report for the award of any other degree or diploma of any other University or Institute.

We have given due credit to the original authors/sources for all the words, ideas, diagrams, graphics, computer programs, experiments, results, that are not my original contribution. We have used quotation marks to identify verbatim sentences and given credit to the original authors/sources.

We affirm that no portion of my work is plagiarized, and the experiments and results reported in the report are not manipulated. In the event of a complaint of plagiarism and the manipulation of the experiments and results, We shall be fully responsible and answerable.

**Akash Chaudhary** (2300290140016)

**Ajay Kaushik**(2300290140014)

**Aisha Sharma** (2300290140013)

**Arnav Kumar Gupta** (2300290140040)

**ACKNOWLEDGEMENT**

At the outset, we would like to thank our guide and advisor**, MR. Shish Pal Jatav Assistant Professor,** for giving us an opportunity to work on this challenging topic and providing us ample and valuable guidance throughout the Project.

Without his encouragement and constant guidance, we would not have able to finish this project. He has been always a source of inspiration and motivator for innovative ideas during the entire span of this work.

We are grateful **Dr. Arun Kumar Tripathi, Professor and Head, Department of Computer Applications, KIET Group of Institutions, Ghaziabad** for providing all the necessary resources to carry out this Project work.

We will be failing in our duty if we don’t acknowledge the people behind this work to give us moral and psychological support. Our special thanks to our parents for their endless care and constant support.

**Akash Chaudhary** (2300290140016)

**Ajay Kaushik**(2300290140014)

**Ayisha Sharma** (2300290140013)

**Arnav Kumar Gupta** (2300290140040)

## CERTIFICATE

Certified that **Akash Chaudhary (2300290140016), Ajay Kaushik (2300290140014), Ayisha Sharma (2300290140014), Arnav Kumar Gupta(2300290140014),** have carried out the project work having **“DRIVEONDEMAND”** for Master of Computer Applications from Dr.

A.P.J. Abdul Kalam Technical University (AKTU) (formerly UPTU), Technical University, Lucknow under my supervision. The project report embodies original work, and studies are carried outby the student himself/herself and the contents of the project report do not form the basis forth award of any other degree to the candidate or to anybody else from this or any other University/Institution.

**Akash Chaudhary** (2300290140016)

**Ajay Kaushik**(2300290140014)

**Aisha Sharma** (2300290140013)

**Arnav Kumar Gupta** (2300290140040)

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

### Date:

### Mr. Shish Pal Jatav

### Assistant Professor

### Internal Examiner External Examiner

**Dr. Arun Kumar Tripathi**

**Head, Department of Computer Application**

**KIET Group of Institutions**

# ABSTRACT

**DriveOnDemand** is a web-based platform designed to revolutionize the car rental experience by seamlessly connecting car owners with renters. The platform enables users to register and log in, providing them access to a dynamic ecosystem where they can explore cars listed by other users for rent. Registered users can book their desired vehicles for specific durations, catering to diverse travel needs with ease and convenience.

In addition to renting cars, users can also become car providers by listing their vehicles on the platform, empowering individuals to monetize their idle assets. DriveOnDemand fosters a community-driven marketplace that ensures flexibility, accessibility, and efficiency in car rentals, enhancing the overall user experience.

This project incorporates robust features, including secure authentication, user-friendly interfaces, and streamlined booking and listing processes. It aims to simplify car rentals, promote resource sharing, and create a mutually beneficial ecosystem for all stakeholders involved.

TABLE OF CONTENTS

[Declaration ii](#_TOC_250015)

Acknowledge iii

[Certificate iv](#_TOC_250014)

[Abstract v](#_TOC_250013)

Table of content vi

List of tables vii

[List of figures viii](#_TOC_250012)

Chapter 1: Introduction

* 1. Project description 9
  2. [Project Scope 10](#_TOC_250011)
  3. [Future Scope 11](#_TOC_250010)
  4. [Identification of need 12](#_TOC_250009)
  5. [Problem Statement 13](#_TOC_250008)
  6. [Software/Technology used in project 14](#_TOC_250007)
     1. Non-Functional Requirement 15
     2. Functional Requirement 16-17
  7. Project Schedule 18

Chapter 2: Feasibility Study

* 1. [Introduction 219-20](#_TOC_250003)
  2. [Main Aspects 21](#_TOC_250002)
     1. [Technical feasibility 22](#_TOC_250001)
     2. Economic Feasibility 22
     3. Operational Feasibility 22
  3. [Benefits 2](#_bookmark0)3
  4. SRS 24-25

[Chapter 3:Design](#_bookmark1)

* 1. Introduction 26
  2. Analysis 27-29
  3. SDLC 31-34
  4. Soft. Engr. Paradig 35

Chapter 4: Report

|  |  |  |
| --- | --- | --- |
| 4.1 Gist |  | 36 |
| 4.2 Some Snippets |  | 37 |
| 4.2.1 | Home Page | 37 |
| 4.2.2 | MongoDB | 41 |
|  |  |  |

Chapter 6: Testing

* 1. [Introduction](#_bookmark4) 42
     1. Testing Objectives 42
     2. Testing Principles 42-43
  2. Level of testing 44
     1. Unit Testing 45
     2. Integration Testing 46
     3. System Testing 46

Chapter 7: Conclusion & Future scope

* 1. [Conclusion](#_bookmark5) 47-48
  2. [Future Scope](#_bookmark6) 49
  3. Bibliography 50

# CHAPTER 1

**INTRODUCTION**

## PROJECT DESCRIPION

Transportation has always been a vital aspect of human activity, evolving over the centuries to provide better accessibility and convenience. With the rise of shared economy models and digital platforms, the car rental industry has witnessed a significant transformation. DriveOnDemand, our web-based application, aims to bridge the gap between car owners and renters by offering a platform for seamless car rentals and sharing.

DriveOnDemand empowers users to register, log in, and explore a diverse collection of cars posted by other users for rental purposes. Registered users can easily book vehicles based on their preferences and travel needs, ensuring a hassle-free booking experience. Simultaneously, individuals can list their vehicles for rent, providing an opportunity to generate income from unused assets.

This project caters to a broad audience, including individuals seeking temporary vehicle solutions and those looking to monetize their cars. DriveOnDemand fosters a community-driven marketplace, promoting resource sharing, sustainability, and convenience.

By leveraging modern technologies for secure authentication, intuitive design, and efficient workflows, DriveOnDemand is an innovative approach to enhancing the car rental ecosystem. It aims to redefine car rentals by creating a platform where users can connect, explore, and share with ease.

## PROJECT SCOPE

The purpose of the **DriveOnDemand** project is to create a comprehensive car rental platform that facilitates seamless communication and transactions between car owners and renters. The primary objectives of the project are as follows:

1. To provide a user-friendly platform where users can register and log in to explore available cars for rent and manage their bookings effortlessly.
2. To enable car owners to list their vehicles on the platform and generate income by renting them out.

The application is designed to cater to a wide audience, offering flexibility for various travel and rental needs. It is intended to promote convenience, accessibility, and efficient resource utilization.

While the platform provides robust features like authentication, booking management, and car listing, it currently focuses on text-based communication between users for inquiries and coordination. Advanced features, such as real-time chat or integrated payment systems, are not yet implemented.

The project has been developed with a strong emphasis on scalability, usability, and maintainability. It incorporates secure mechanisms for user authentication and authorization to ensure data confidentiality and integrity.

**Key Features:**

1. **Authentication:** Secure login and registration process for all users.
2. **Integrity:** Reliable data handling and booking processes to ensure accurate transactions.
3. **Confidentiality:** Protection of user data and sensitive information through secure communication protocols.

## FUTURE SCOPE

1. **Enhanced Authorization:** Expanding user roles and permissions for better management of access and functionalities.
2. **Database Integration:** Developing a centralized database to maintain user profiles, car listings, and booking records.
3. **Voice Communication:** Introducing voice chat features to facilitate real-time communication between users.
4. **Web Compatibility:** Extending the platform to support web applications for greater accessibility across devices

## IDENTIFICATION OF NEED

In today’s interconnected world, especially during challenging times such as the lockdown period, the demand for reliable and secure digital communication platforms has surged. While many existing chatting applications facilitate seamless communication, they often compromise user privacy by sharing data with third parties for profit, leaving users vulnerable to data misuse and privacy breaches.

The **DriveOnDemand** project addresses the need for a secure and user-centric car rental platform by focusing on confidentiality, data integrity, and privacy. Just as secure communication platforms became essential during lockdowns, a secure platform for car rentals is critical in today's sharing economy.

**Key Needs Identified:**

1. **Data Privacy and Security:** Protecting user information, car listings, and booking details with stringent security measures.
2. **End-to-End Encryption:** Ensuring that all interactions, such as inquiries between renters and car owners, remain confidential and protected from external threats.
3. **Convenience with Transparency:** Allowing users to trust the platform for renting or listing cars without fear of data misuse.

By prioritizing these aspects, DriveOnDemand ensures a trustworthy and reliable environment where users can confidently connect, transact, and interact. The project’s features aim to provide an unparalleled experience by integrating secure communication protocols with user-friendly functionalities.

## PROBLEM STATEMENT

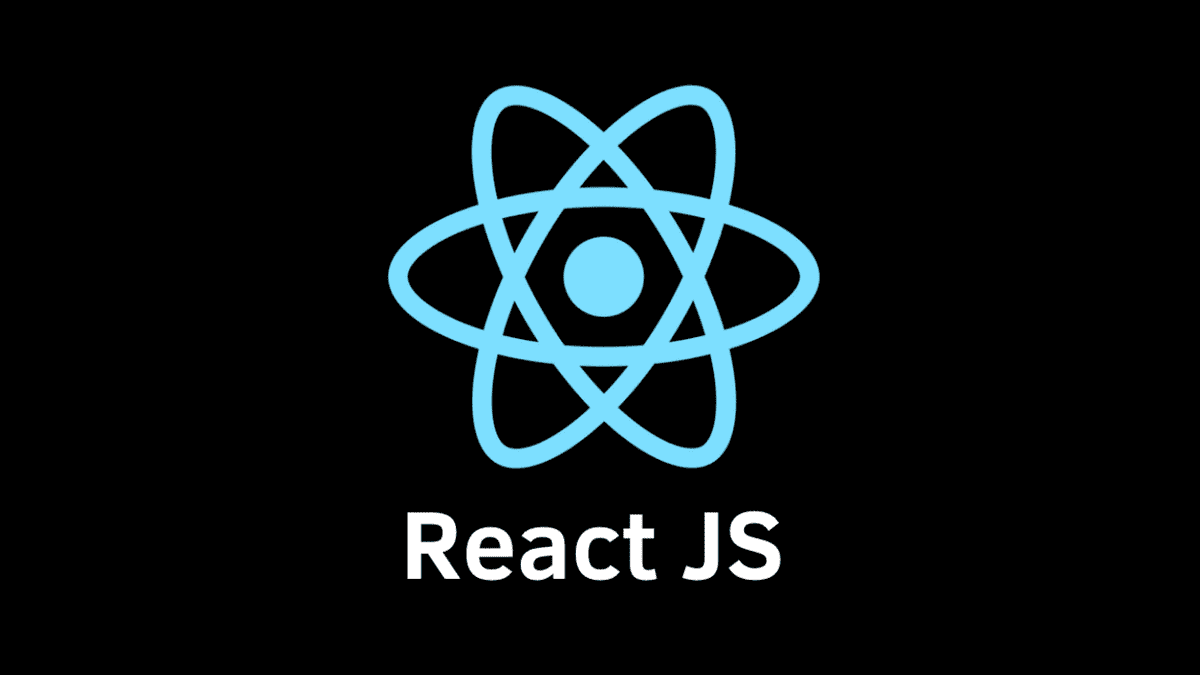
In the lockdown period, people's mobility was restricted, leaving virtual communication as the only viable option to stay connected. While chatting applications became an essential tool, many existing platforms posed significant privacy concerns, exposing users to risks such as data breaches and unauthorized access to personal conversations.

The primary problem lies in the lack of secure and private communication solutions that guarantee user data and messages are protected from external threats or misuse. To address this issue, our team has proposed developing a robust and secure Android-based chatting application that prioritizes privacy and user data security.

This application is designed to provide a secure, user-friendly solution for private communication, addressing the growing need for privacy in digital interactions.

## SOFTWARE/TECHNOLOGY USED IN PROJECT

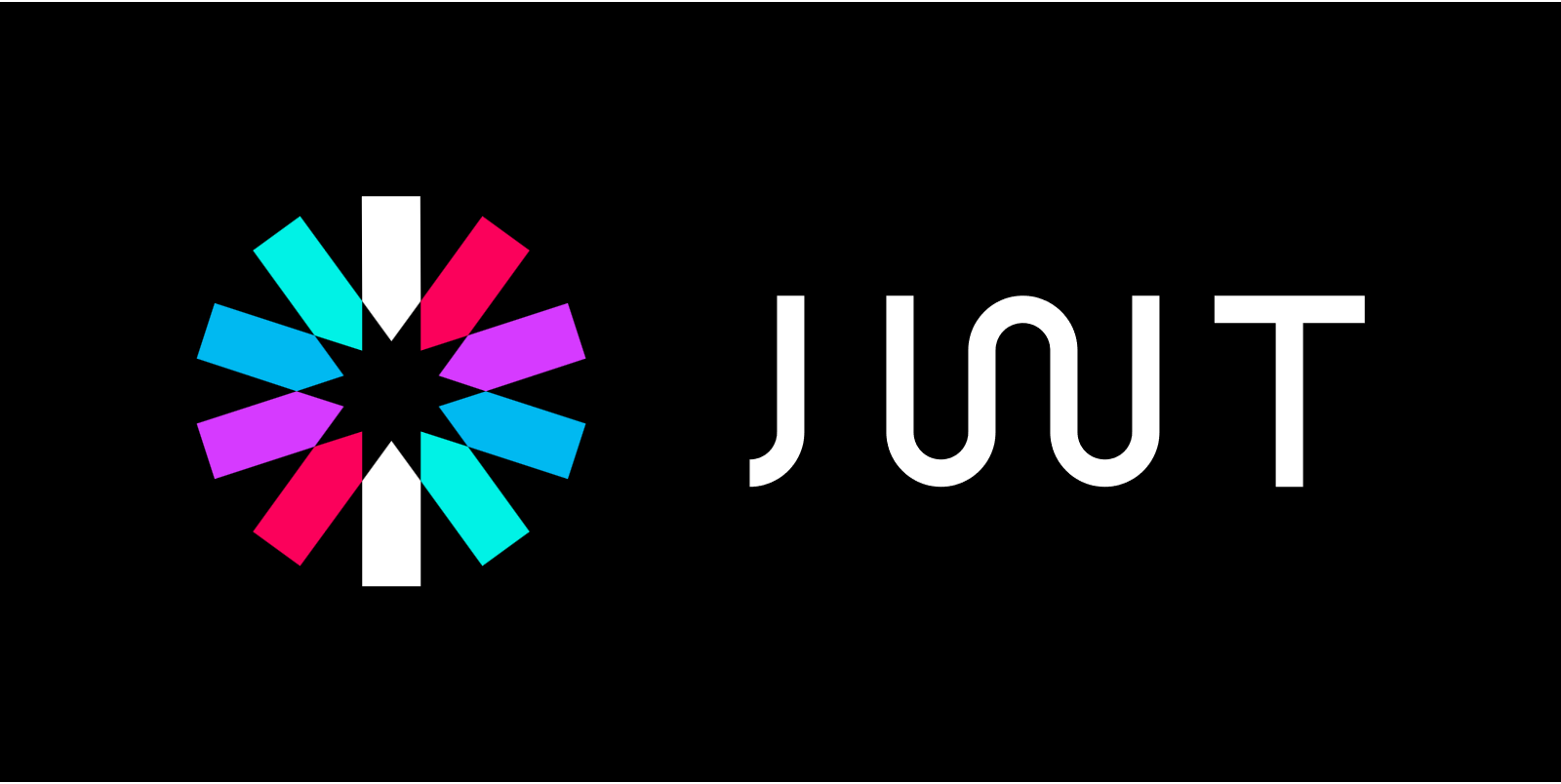
### React JS :-



[This Photo](https://dev-astra.tistory.com/565) by Unknown Author is licensed under [CC BY-NC-ND](https://creativecommons.org/licenses/by-nc-nd/3.0/)

Figure 1.1: ReactJS logo

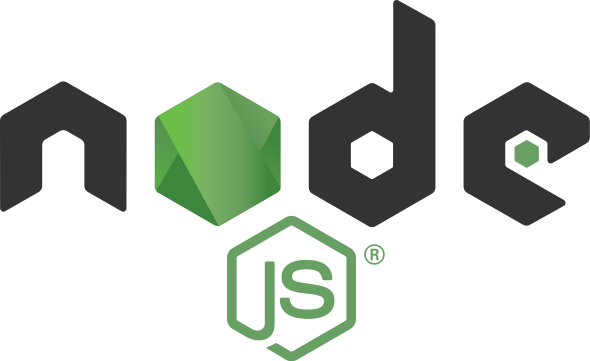
### JWT



[This Photo](https://www.sainth.de/) by Unknown Author is licensed under [CC BY-SA](https://creativecommons.org/licenses/by-sa/3.0/)

Figure 1.3: JWT logo

* + 1. **NODE.JS**



[This Photo](https://infogonzalez.com/2020/02/primeros-pasos-con-express-js-con-nodejs.html) by Unknown Author is licensed under [CC BY-SA-NC](https://creativecommons.org/licenses/by-nc-sa/3.0/)

## NON- FUNCTIONAL REQUIREMENTS

### Performance Requirements

### To ensure optimal performance and user satisfaction, the following requirements must be fulfilled:

### Scalability: The system should be easily scalable to handle an increasing number of users, car listings, and bookings without impacting performance.

### Portability: The application should function seamlessly across various hardware and software environments, including mobile and desktop platforms.

### Security: Robust security mechanisms should be implemented to ensure all user data and transactions are protected from unauthorized access, manipulation, or breaches.

### Maintainability: The system should be designed for easy updates and enhancements, enabling quick fixes to faults, performance improvements, and adaptation to changing environments.

### Reliability: The platform must consistently perform its intended functions, including user authentication, car listing, and booking, without failure under specified conditions.

### Reusability: Key components of the software, such as authentication and booking modules, should be reusable in other related systems to enhance development efficiency.

## FUNCTIONAL REQUIREMENTS

### Firebase



[This Photo](https://kogun82.tistory.com/236) by Unknown Author is licensed under [CC BY-NC-ND](https://creativecommons.org/licenses/by-nc-nd/3.0/)

Figure 1.4: MongoDB logo

**MongoDB** is a powerful, open-source NoSQL database that stores data in a flexible, JSON-like format known as BSON (Binary JSON). It is widely used for building applications that require scalability, flexibility, and high availability. MongoDB’s architecture is different from traditional relational databases, allowing for greater flexibility in terms of data structure and storage.

* + - 1. **Visual Studio Code**



[This Photo](https://code.makery.ch/library/html-css/part1/) by Unknown Author is licensed under [CC BY](https://creativecommons.org/licenses/by/3.0/)

Figure 1.6: Visual Studio Code Logo

**Visual Studio Code (VS Code)** is a free, open-source, and highly customizable code editor developed by Microsoft. It is widely used by developers for creating applications, websites, and more due to its rich feature set, speed, and versatility. It supports various programming languages, including JavaScript, Python, C++, Java, PHP, and many more, making it suitable for a wide range of projects.

## PROJECT SCHEDULE

The objective of software project planning is to provide a framework that enables the manager to make reasonable estimates of resources, costs and schedule. These estimates are made within a limited time frame at the beginning of a software project and should be updated regularly as the project progresses. In addition, estimates should attempt to define “best case” and “worst case” scenarios so that project outcomes can be bounded.

The first activity in software project planning is the determination of software scope. Function and performance allocated to software during system engineering should be assessed to establish a project scope that is ambiguous and understandable at management and technical levels. Software scope describes function, performance, constraints, interfaces and reliability.

During early stages of project planning, a microscopic schedule is developed. This type of schedule identifies all major software engineering activities and the product functions to which they are applied. As the project gets under way, each entry on the macroscopic schedule is refined into detailed schedule. Here specific software tasks are identified and scheduled.

Scheduling has following principles:

1. Compartmentalization: the project must be compartmentalized into a number of manageable activities and tasks.
2. Interdependency: the interdependencies of each compartmentalized activity or tasks must be determined.
3. Time allocation: each task to be scheduled must be allocated some number of work units.
4. Effort validation: every project has a defined number of staff members.

# CHAPTER 2

**FEASBILITY STUDY**

## INTRODUCTION

Feasibility of the system is a crucial aspect to consider while developing "DriveOnDemand." The system must adhere to the law of economics, ensuring that maximum output is achieved with minimal resources.

A feasibility analysis evaluates the potential success of the project, ensuring objectivity in assessing its viability. This objectivity is vital for stakeholders and potential investors. There are five key types of feasibility studies, each of which is described below:

1. Technical Feasibility

This assessment focuses on the technical resources available to develop the "DriveOnDemand" system. It evaluates whether existing resources meet capacity requirements and whether the technical team can implement the proposed functionalities.

1. Economic Feasibility

The economic feasibility involves a cost/benefit analysis, determining whether the project is viable and financially beneficial for stakeholders.

1. Legal Feasibility

This assessment ensures compliance with legal standards such as data protection laws by safeguarding user data through proper encryption and privacy policies. It also adheres to vehicle rental regulations, ensuring conformity with local laws, including provisions for insurance and liability. By addressing these legal requirements, the project mitigates potential regulatory conflicts.

1. Operational Feasibility

Operational feasibility examines how well "DriveOnDemand" aligns with organizational goals and user needs. This includes:

* User experience: Ensuring seamless navigation, car search, and rental request features.
* Admin capabilities: Efficient management of listings, user accounts, and payment processing.  
  Operational feasibility ensures the platform addresses the core needs of both car owners and renters.

## MAIN ASPECTS

There are three aspects of feasibility to be considered namely.

1. Technical
2. Operational
3. Economical

**TECHNICAL:**

The technical feasibility of "DriveOnDemand" focuses on system requirements, such as:

* Minimal hardware resources: Hosting the platform on a cloud service (e.g., AWS or Azure).
* Modern software tools: React.js for UI, Node.js for backend logic, and MongoDB for database management.  
  This ensures smooth performance with limited initial resources, making the system scalable and efficient.

OPERATIONAL:

Operational feasibility examines the system's ability to reduce workload and improve user satisfaction. For example:

* Car owners can easily post listings with images and rental details.
* Renters can search for cars based on location, price, and availability.  
  This ensures timely and accurate service delivery while reducing manual effort.

ECONOMICAL:

"DriveOnDemand" is economically feasible due to low initial software costs and efficient operations:

* Cost-effective development with open-source tools and frameworks.
* A revenue model based on transaction fees or subscription plans.  
  This minimizes waste and optimizes resource usage while ensuring profitability.

## Technical feasibility:

## The existing technical infrastructure is adequate for hosting the "DriveOnDemand" system. The system design includes tools like:

## React.js for creating an intuitive and responsive user interface.

## Node.js and Express for handling backend operations.

## MongoDB for a robust and scalable database. These tools are technically feasible and align with the project requirements

## Operational Feasibility:

## Operational feasibility includes:

## Availability of staff to manage platform operations and user support.

## Training modules for administrators to manage listings and monitor transactions. Behavioral feasibility also highlights user acceptance, ensuring the platform's design and workflow are intuitive.

## Economic feasibility:

## A detailed cost/benefit analysis reveals that "DriveOnDemand" is economically viable:

## Costs: Initial development, server hosting, and marketing expenses.

## Benefits: Revenue generation from service fees, enhancing financial sustainability. Economic analysis ensures the project remains financially rewarding over its .

## BENEFITS

Conducting a feasibility study for "DriveOnDemand" provides several advantages:

* Improved focus on project objectives.
* Identification of new opportunities for market growth.
* Provides a basis for the “go/no-go” decision.
* Validates the business idea with strong reasoning.
* Enhances project success by evaluating critical parameters.
* Facilitates informed decision-making and minimizes risks.

## SYSTEM REQUIREMENT SPECIFICATION

Any system can be designed after specifies the requirement of the user about that system. Fortis first of all gathered information from user by the preliminary investigation which is starting investigation about user requirement.

The data that the analysts collect during preliminary investigation are gathered through the various preliminary methods.

1. Documents Reviewing Organization

The analysts conducting the investigation first learn the organization involved in, or affected by the project. Analysts can get some details by examining organization charts and studying written operating procedures.

Collected data is usually of the current operating procedure:

* The information relating to clients, projects and students and the relationship between them was held manually.
* Managing of follow-ups was through manual forms.
* Complaints require another tedious work to maintain and solve.
* Payments details had to be maintained differently.

1. Gathering Information by Asking Questions

Interviewing is the most commonly used techniques in analysis. It is always necessary first to approach someone and ask them what their problems are, and later to discuss with them the result of your analysis.

1. Questionnaires

Questionnaires provide an alternative to interviews for finding out information about a system. Questionnaires are made up of questions about information sought by analyst. The questionnaire is then sent to the user, and the analyst analyzes the replies.

1. Electronic Data Gathering

Electronic communication systems are increasingly being used to gather information. Thus, itis possible to use electronic mail to broadcast a question to a number of users in an organization to obtain their viewpoint on a particular issue.

1. Interviews

Interview allows the analysts to learn more about the nature of the project request and reason of submitting it. Interviews should provide details that further explain the project and show whether assistance is merited economically, operationally or technically.

One of the most important points about interviewing is that what question you need to ask.

It is often convenient to make a distinction between three kinds of question that is

* + Open questions
  + Closed question
  + Probes

Open questions are general question that establish a person’s view point on a particular subject.

Closed questions are specific and usually require a specific answer.

## CHAPTER 3

## DESIGN

## INTRODUCTION

Systems are developed to address specific problems or improve existing processes. A systems approach provides an organized framework for identifying and solving these issues. In today's fast-paced world, system analysis and design play a pivotal role in the development of software solutions, including "DriveOnDemand."

The creation of a new system requires meticulous planning, and one of the most critical phases of the Software Development Life Cycle (SDLC) is **System Requirement Gathering and Analysis**. During this phase, a comprehensive understanding of the system's operational environment and user needs is established.

Analysis involves a detailed study of various operations that the system performs and their interrelationships, both internally and externally.

System analysis also involves breaking down complex processes into manageable components, identifying data storage requirements, and documenting all manual or automated workflows.

## SYSTEM ANALYSIS

System is created to solve problems. One can think of the systems approach as an organized way of dealing with a problem. In this dynamic world, the subject system analysis and design, mainly deals with the software development activities.

Since a new system is to be developed, the one most important phases of software development life cycle are system requirement gathering and analysis. Analysis involves detailed study of the current system, leading to specification of a new system. Analysis is a detailed study of various operations performed by a system and their relationship within and outside the system. Using the following steps, it becomes easy to draw the exact boundary of the new system under consideration.

Keeping in view the problems and new requirements, workout the pros and cons including new area of the system.

All procedures, requirements must be analyzed and documented in the form of detailed DFDs, logical data structure and miniature specifications.

System analyses also include sub-dividing of complex process involving the entire system, identification of data store and manual processes.

System Analysis is conducted with the following steps

* Information gathering
* The tools of structured analysis
* Identification of Need
* System Planning and initial investigation
* Feasibility study

Information Gathering:

* + Information about the firm
  + Information about the workflow
  + Various tools used are:
    - Review of literature
    - Procedure
    - Forms

Initial investigation:

* + Problem definition and project initiation
  + Determining the requirements
  + Needs identification
  + Dimension of planning
  + Determination of feasibility

Feasibility Analysis:

* + System Performance definition
  + Identification of system objectives
  + Description of outputs

Preliminary Investigation:

* + - Evaluation of project request is major purpose of preliminary investigation.
    - It is the collecting information that helps committee members to evaluate merits of the project request and make judgment about the feasibility of the proposed projects.
    - To answer the above questions, system analysts discuss with different category of person to collect facts about their business and their operations.
    - When the request is made, the first activity the preliminary investigation begins.
    - Preliminary investigation has three parts-

1. Request clarification
2. Feasibility study

An information system is intended to meet needs of an organization. Thus, the first step is in this phase is to specify these needs and requirements.

* + - The next step is to determine the requirements met by the system. Many requests from employees and users in the organizations are not clearly defined. Therefore, it become necessary that project request must examine and clarified properly before considering system investigation.
    - Information related to different needs of the System can be obtained by different users of the system. This can be done by reviewing different organization’s documents such
    - as current method of storing sales data, complaint data etc. By observing the onsite activities, the analyst can get close information related to real system.

## SDLC

Software Development Life Cycle (SDLC) is a framework that defines the steps involved in the development of software at each phase. It covers the detailed plan for building, deploying and maintaining the software.

SDLC defines the complete cycle of development i.e. all the tasks involved in planning, creating, testing, and deploying a Software Product.

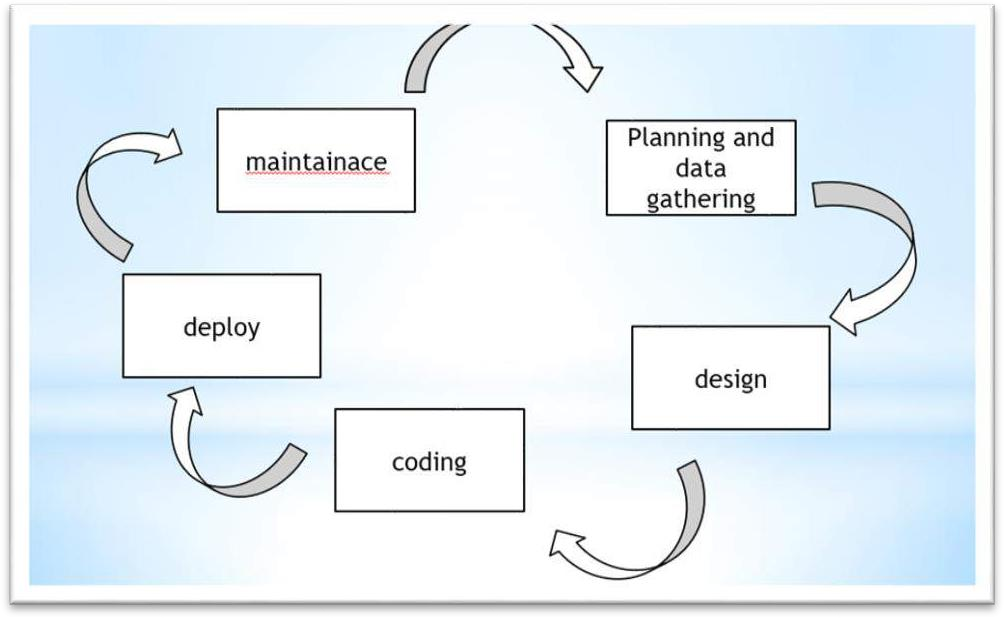


Figure 3.1: Above image depicting the planning step

### SDLC Phases

**Given below are the various phases:**

* Requirement gathering and analysis
* Design
* Implementation or coding
* Testing
* Deployment

1. Requirement Gathering and Analysis

During this phase, all the relevant information is collected from the customer to develop a product as per their expectation. Any ambiguities must be resolved in this phase only.

Business analyst and Project Manager set up a meeting with the customer to gather all the information like what the customer wants to build, who will be the end-user, what is the purpose of the product. Before building a product a core understanding or knowledge of the product is very important.

**For Example,** A customer wants to have an application which involves money transactions. In this case, the requirement has to be clear like what kind of transactions will be done, how it will be done, in which currency it will be done, etc.

1. Design

In this phase, the requirement gathered in the SRS document is used as an input and software architecture that is used for implementing system development is derived.

1. Implementation or Coding

Implementation/Coding starts once the developer gets the Design document. The Software design is translated into source code. All the components of the software are implemented in this phase.

1. Testing

Testing starts once the coding is complete and the modules are released for testing. In this phase, the developed software is tested thoroughly and any defects found are assigned to developers to get them fixed.

Retesting, regression testing is done until the point at which the software is as per the customer’s expectation. Testers refer SRS document to make sure that the software is as per the customer’s standard.

1. Deployment

Once the product is tested, it is deployed in the production environment or first [UAT](https://www.softwaretestinghelp.com/what-is-user-acceptance-testing-uat/) User Acceptances done depending on the customer expectation.

In the case of UAT, a replica of the production environment is created and the customer along with the developers does the testing. If the customer finds the application as expected, then sign off is provided by the customer to go live.

1. Maintenance

After the deployment of a product on the production environment, maintenance of the product i.e., if any issue comes up and needs to be fixed or any enhancement is to be done is taken care by the developers.

## SOFTWARE ENGG. PARADIGM APPLIED

Software engineering is a layered technology. The foundation for software engineering is the process layer. Software engineering processes the glue that holds the technology layers together and enables ratios and timely development of computer software. Process defines a framework for a set of key process areas that must be established for effective delivery of software engineering technology.

Software engineering methods provide the technical how-tops for building software.

Methods encompass a broad array of tasks that include requirements analysis, design, program construction, testing and support. Software engineering tools provide automated or semi- automated support for the process and the methods. When tools are integrated so that information created by one tool can be used by another tool, a system for the support of software development, called computer-aided software engineering is established.

### The following paradigms are available:

1. The Waterfall Model
2. The Prototyping Model

# CHAPTER 4

**REPORT**

## GIST

We have developed an Android-based chatting application to facilitate seamless communication between users while prioritizing their privacy and data security. The application ensures that user data is protected and kept confidential. For user authentication, we have implemented Google Firebase Authentication, specifically utilizing phone number verification to enhance security and streamline the login process.

To store user profile information, we have integrated Google's Firebase database, which provides a reliable and scalable solution. For real-time chatting functionality, we use Google Realtime Database, enabling users to send and receive messages instantly. The application also supports the sharing of various file types, including text files, images, and videos, ensuring all data is end-to-end encrypted for maximum privacy and security.

Additionally, the application allows users to upload and update their profiles and share status updates visible to others. Users can view the statuses uploaded by their contacts, enhancing interactivity. The application also includes the feature to send reactions to messages, adding a personalized touch to conversations and enhancing user engagement.

## SOME SNIPPETS

* + 1. HOME PAGE

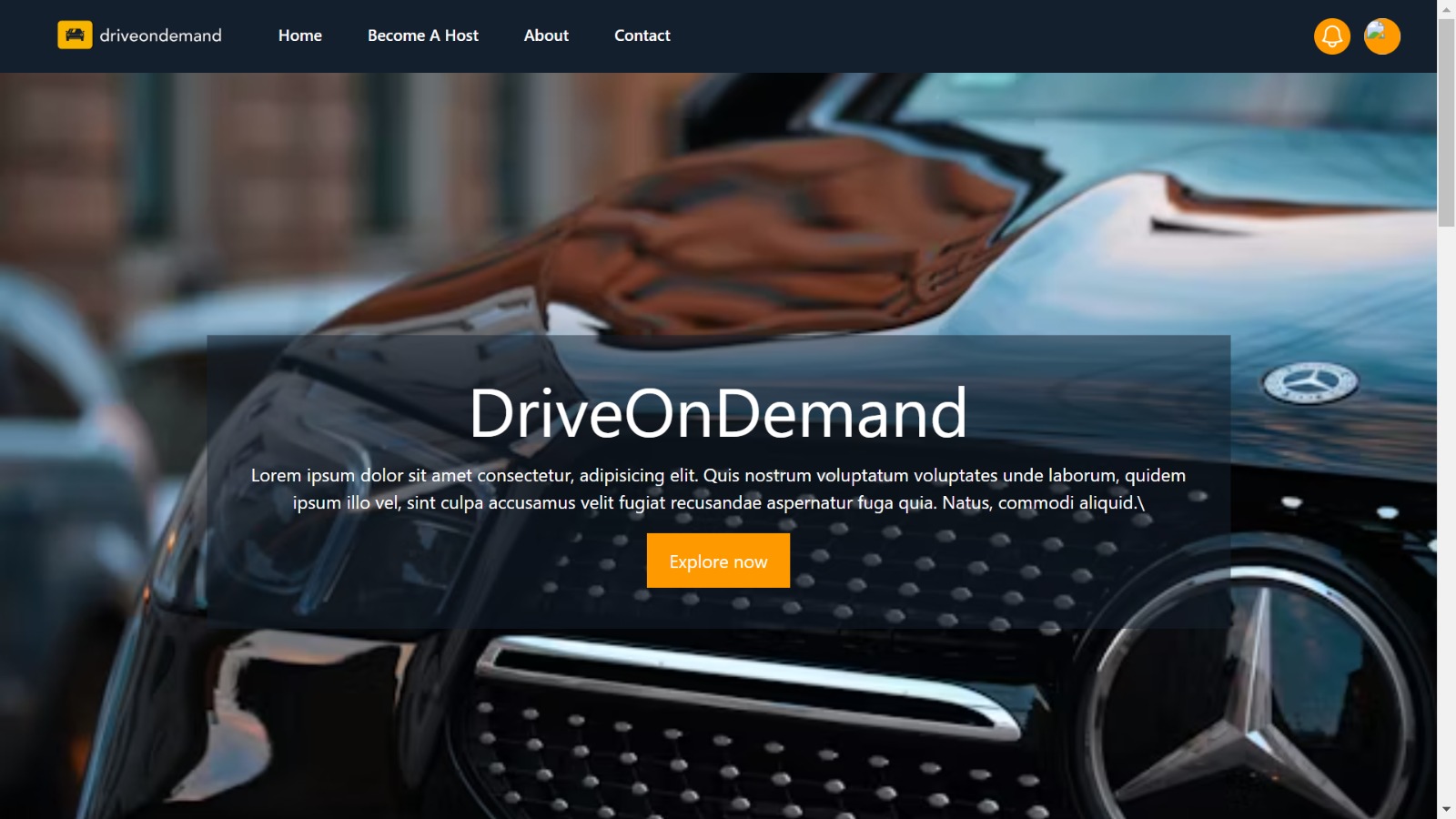
****

Figure 4.1: Home Page

* + 1. **LOGIN PAGE**

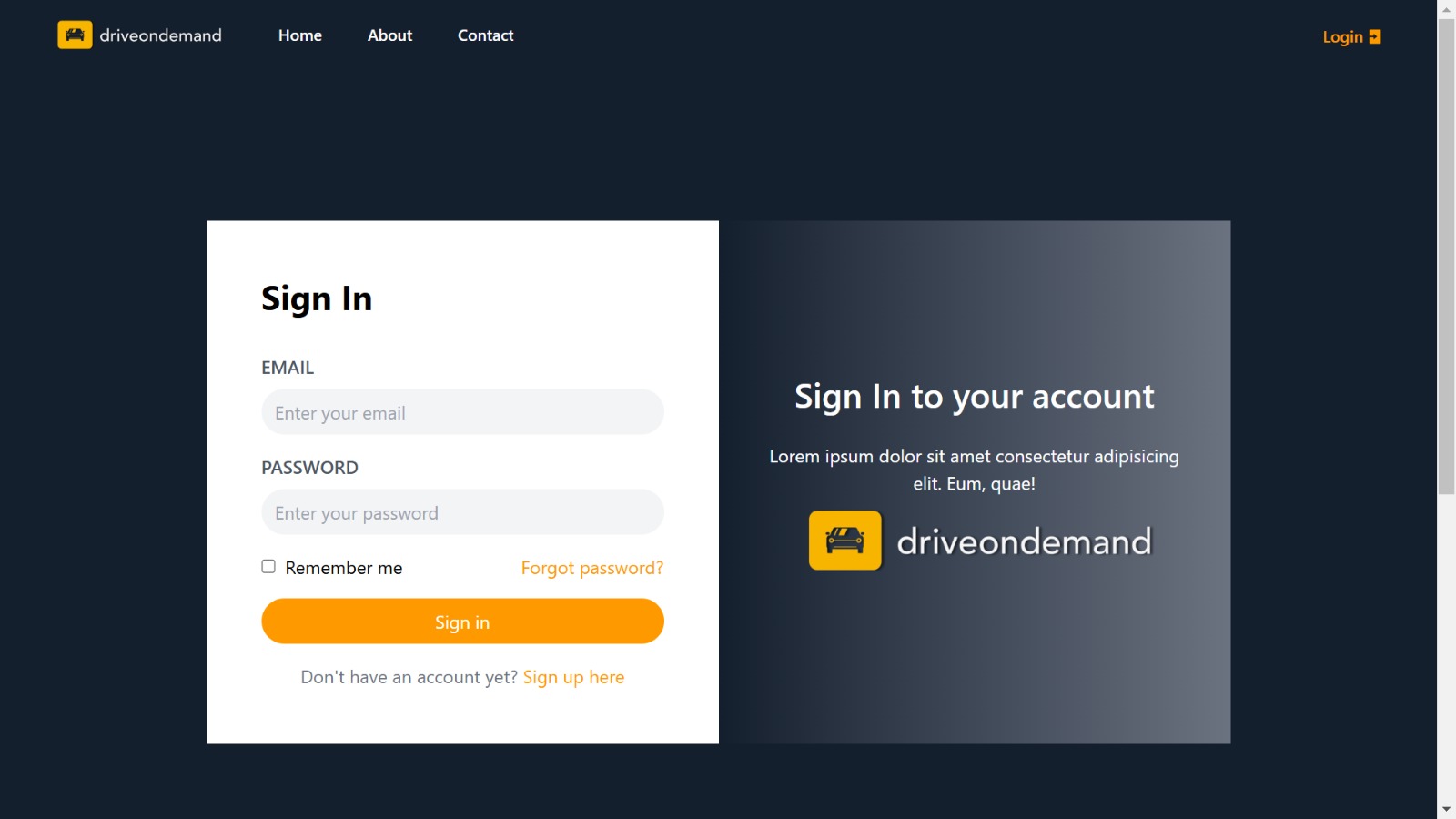
****

Figure 4.1: Login Page

* + 1. **SIGNUP PAGE**

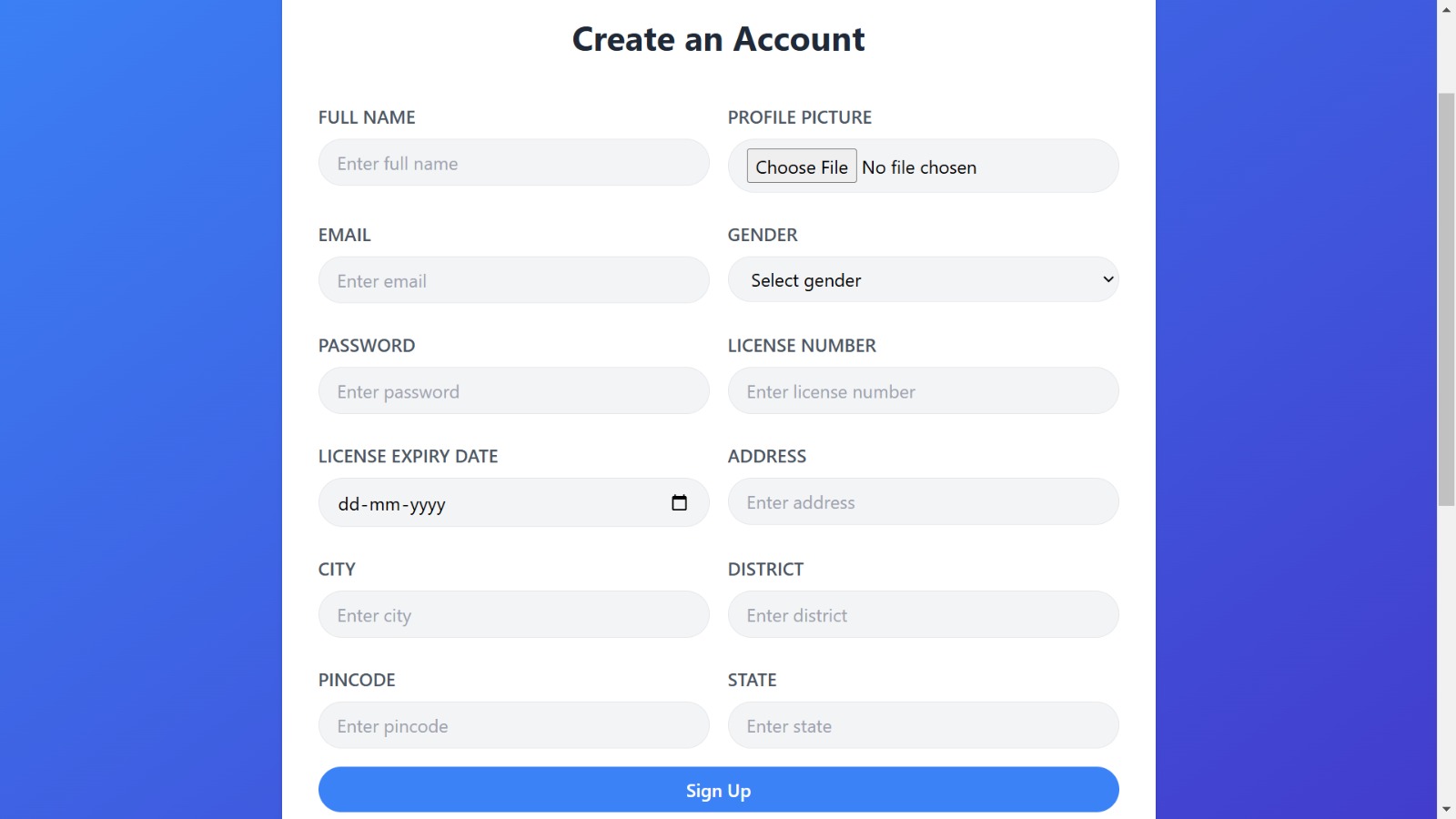
****

Figure 4.1: Signup Page

## MONGODB

In MongoDB, all data is stored in flexible, JSON-like documents, which are part of collections within a database. Unlike SQL databases, MongoDB does not use tables or rows. Instead, data is organized as documents, where each document is a key-value pair structure.

When adding data to a collection in MongoDB, each document is stored with a unique identifier, known as the \_id field, which is automatically generated unless explicitly provided. You can create custom keys, such as user IDs or semantic identifiers, or rely on MongoDB to generate an Object Ids for you.

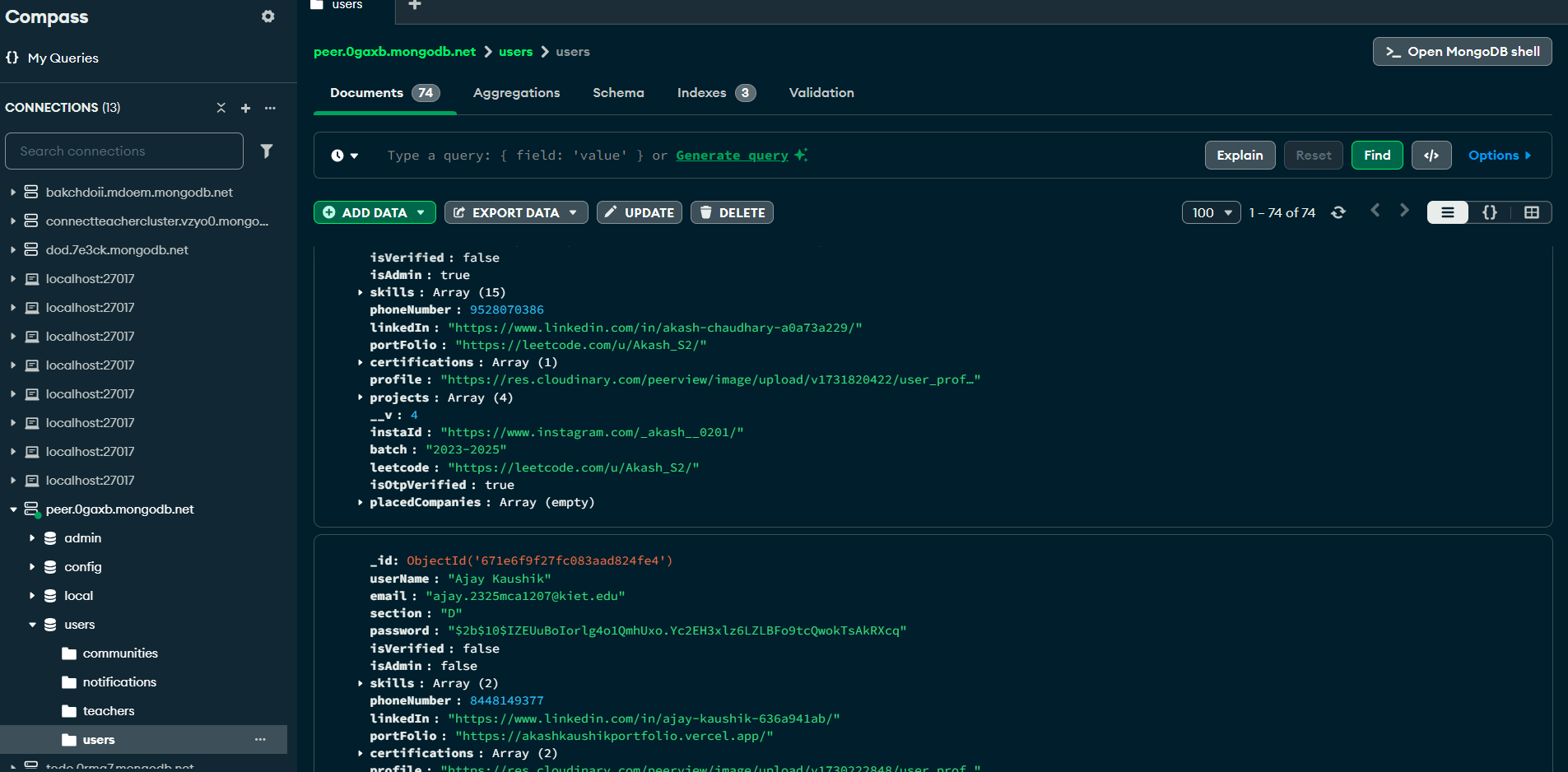


Figure 4.5: MongoDB database

## VISUAL STUDIO CODE

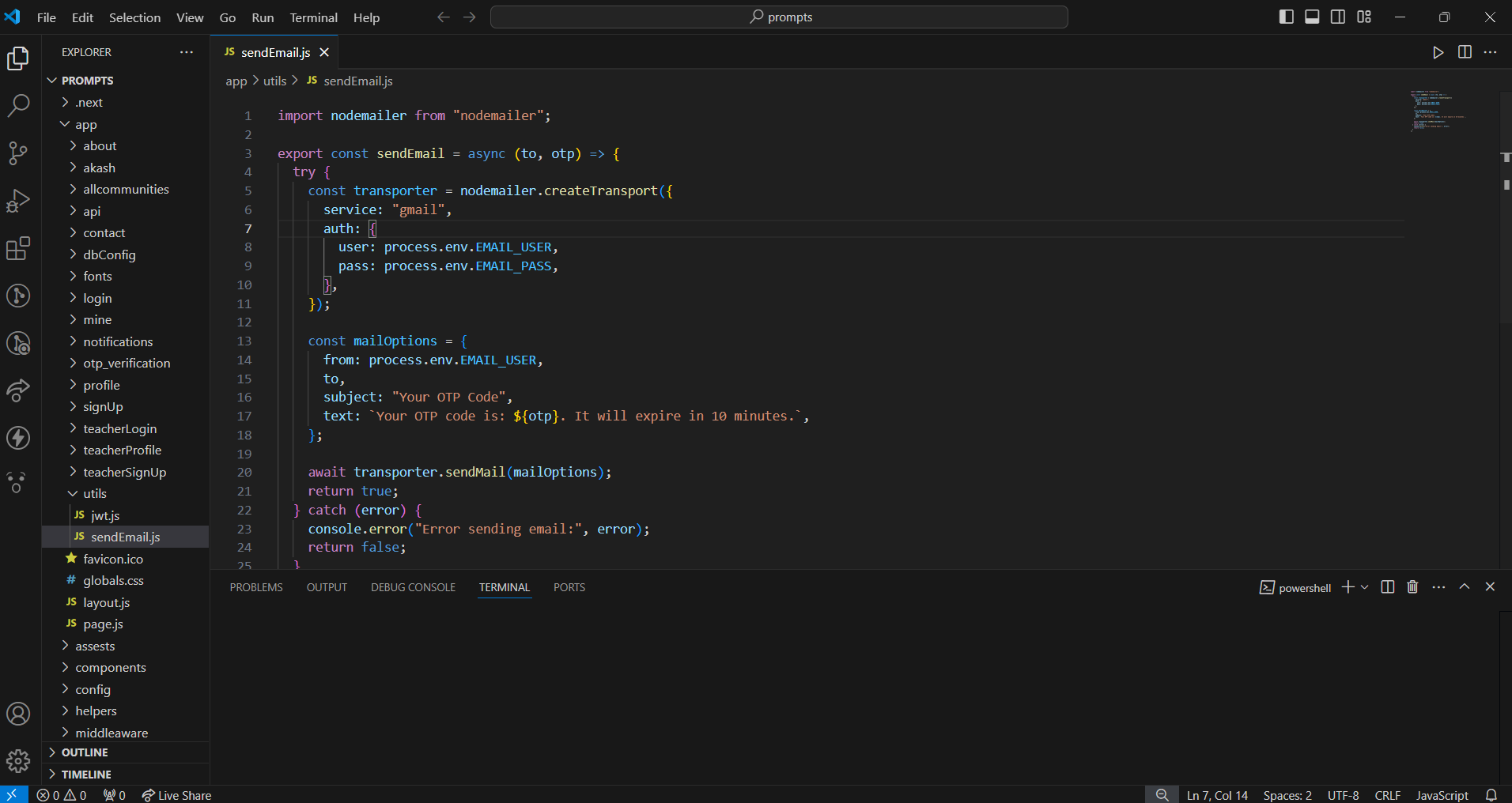
****

Figure 4.7: Visual Studio Code Main Window

1. The **toolbar** lets you carry out a wide range of actions, including running your app and launching Android tools.
2. The **navigation bar** helps you navigate through your project and open files for editing. It provides a more compact view of the structure visible in the **Project** window.
3. The **editor window** is where you create and modify code. Depending on the current file type, the editor can change. For example, when viewing a layout file, the editor displays the Layout Editor.
4. The **tool window bar** runs around the outside of the IDE window and contains the buttons that allow you to expand or collapse individual tool windows.

# CHAPTER 6

**TESTING**

## INTRODUCTION

### 6.1.1 Testing Objectives

The following are the testing objectives:

-Testing is a process of executing a program with the intent of finding an error.

-A good test case is one that has a high probability of finding an as-yet-undiscovered error

-successful test is one that uncovers an as yet undiscovered error.

### 2 Testing Principles

The basic principles that guide software testing are as follows:

* + - -All tests should be traceable to customer requirements.
    - -Tests should be planned long before testing begins.
    - -The parate principle applies to software testing.

Pareto principle states that 80 percent of all errors uncovered during testing will likely betraceable to 20 percent of all program components.

Testing should begin “in the small “and progress toward testing “in the large.” Exhaustive testing is not possible.

### LEVEL OF TESTING

There are different levels of testing

->Unit Testing

->Integration Testing

->System Testing

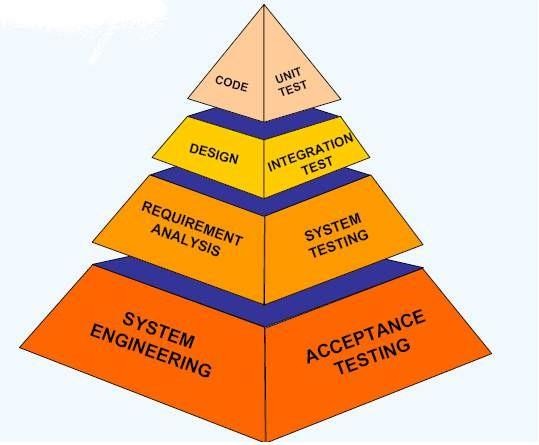


Figure 5.1: Testing pyramid

## Unit testing

Unit testing focuses verification effort on the smallest unit of software design,

the module. The important control parts are tested to uncover with in theboundary of the module. The module interface is tested to ensure that the information properly flows into and out of the program unit and boundary conditions are tested to ensure that the modules operate properly at boundaries established to limit or restrict processing. Test date is provided through testing screens.

## Integration testing

Integrating testing is a systematic technique for constructing Program structure while conducting tests to uncover error associates with interfacing. The objective isto take unit modules and built a program structure that has been directed by design.

* Integration Testing will test whether the modules work well together.
* This will check whether the design is correct.

## System testing

System testing is the process of testing the completed software as a part of the environment itwas created for. It is done to ensure that all the requirements specified by the customer are met. System testing involves functional testing and performance testing.

* System Testing will contain the following testing:
  + Functional Testing.
  + Performance Testing.
* Function Testing will test the implementation of the business needs.
* Performance Testing will test the non-functional requirements of the system like the speed, load etc.

# CHAPTER 7

**CONCLUSION AND FUTURE SCOPE**

## CONCLUSION

The primary objective of this project was to develop a secure and efficient community-based web application aimed at connecting members, local businesses, and organizations. Through thorough research and analysis, we reviewed existing platforms to identify their limitations and designed a solution to address these gaps. This project provided an opportunity to explore advanced technologies, develop robust algorithms, and understand the intricacies of secure web application development.

The web application was successfully developed using modern web technologies, incorporating features such as authentication, role-based access control, and real-time interaction. The system is designed to be extendable, portable, and maintainable while meeting critical requirements such as:

* Authentication
* Integrity
* Confidentiality

The experience gained during the development of this application has enhanced my technical expertise, and I am confident that I can further improve this system in the future by implementing advanced features and optimizing its functionality.

Future improvements may include:

* Enhancing the application with detailed analytics and reporting.
* Developing an intuitive and mobile-responsive UI/UX design.
* Expanding the application to support multi-language communication.
* Introducing notification systems for timely updates.
* Increasing scalability to handle a larger user base.

## FUTURE SCOPE

The future scope of the project is to make the system more user-friendly, feature-rich, and scalable. Planned enhancements include:

* Adding **status uploading** capabilities to allow users to share updates in real-time.
* Implementing features for **sending reactions** to posts and comments for interactive engagement.
* Introducing **audio call functionality** for improved communication.
* Expanding to include **single/group video call features** for enhanced connectivity.
* Providing complete **web support** for accessibility across all devices.
* Strengthening application security with advanced encryption techniques and robust privacy policies.
* Implementing a detailed and transparent **user privacy policy** to ensure data protection and user trust.

## BIBLIOGRAPHY

1. Enhancing JWT Authentication and Authorization in Web Applications Based on User Behavior History. (By Ahmet Bucko) [LINK](https://www.mdpi.com/2073-431X/12/4/78).
2. What is Express.js? (By Mahesh Sharma) [LINK](https://www.studytrigger.com/article/what-is-express-js/)
3. JavaScript Cookies: Managing Browser Cookies for Data Storage. (By CodeLucky) [LINK](https://codelucky.com/javascript-cookies/)
4. An Introduction to Tailwind CSS. (By Michael Gold) [LINK](https://medium.com/@msgold/an-introduction-to-tailwind-css-e2c5bbc0ae34)
5. Mastering Client-Side Storage: A Comprehensive Guide to Using Local Storage and Session Storage in JavaScript. (By Ivan Kaminskyi) [LINK](https://dev.to/ivan_kaminskyi/mastering-client-side-storage-a-comprehensive-guide-to-using-local-storage-and-session-storage-in-javascript-2fg6)

## GITHUB PROJECT LINK

<https://github.com/ajaykaushik057/DriveDemand>