# SATISH PRADHAN DNYANASADHANA COLLEGE OF ARTS, SCIENCE AND COMMERCE, THANE

## PROJECT REPORT

ON

## "CREDIT CARD APPLICATION MANAGEMENT SYSTEM"

**IS SUBMITTED IN PARTIAL FULFILMENT OF**

**T.Y.B.SC (COMPUTER SCIENCE)**


**SUBMITTED BY**

**ANUJA MAHENDRA KAPOOR**

**UNDER THE GUIDANCE OF**

**Asst. Prof. Trupti Rongare**



**DEPARTMENT OF COMPUTER SCIENCE**


**(2023-24)**

# SATISH PRADHAN DNYANASADHANA COLLEGEOF ARTS, SCIENCE AND COMMERCE, THANE

# PROJECT REPORT

On

## "CREDIT CARD APPLICATION MANAGEMENT SYSTEM"

**IS SUBMITTED IN PARTIAL FULFILMENT OF**

**T.Y.B.SC (Computer Science)**


SUBMITTED BY

**MISS.ANUJA MAHENDRA KAOOR**

UNDER THE GUIDANCE OF

**Asst. Prof. Trupti Rongare**


**DEPARTMENT OF COMPUTER SCIENCE**

**(2023-24)**

# SATISH PRADHAN DNYANASADHANA COLLEGE OF ARTS, SCIENCE AND COMMERCE, THANE

## ( DEPARTMENT OF COMPUTER SCIENCE)

## CERTIFICATE

This is to certify that the project entitled, **" Credit Card Application Management System"** , is bonafide work **ANUJA MAHENDRA KAPOOR** bearing Seat.No: submitted in partial fulfilment of the requirements for the award of degree BACHELOR OF SCIENCE in COMPUTER SCIENCE from University of Mumbai.

| Project Guide | External Examiner | Signature of HOD | Sign of Principle |
|---|---|---|---|
| **Prof.Trupti Rongare** | | **Dr.Sujata Iyer** | **Dr.Ganesh Bhagure** |

# ABSTRACT


- Credit Card Application Management System contains data and information of credit card holder who apply for credit card online. The main purpose of CCAMS is to systematically record, store and update the details of admin/sub-admin and also manage the credit card application.
- This projects helps to those people who apply credit card online without wasting a time and also check the application status by their name/mobile number/application number.

# ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of any task would be incomplete without mentioning the people whose ceaseless cooperation made it possible, whose constant guidance and encouragement crown all the efforts with success

I am grateful to my Project Guide Asst. Prof. Trupti Rongare for the guidance, inspiration and constructive suggestions that was helpful to us in the preparation of this project.

I would like to express my sincere thanks to our Principal Dr. Ganesh Bhagure Sir and our Head of Department Dr. Sujatha Iyer for having facilitated us with the essential infrastructure and resources without which this project would not have seen light of the day.

I would also like to thank all my friends for giving me moral support and encouragement during the process of making my project. I am also thankful to my entire staff of CS & IT for their constant encouragement, suggestions and moral support throughout the duration of my project.

I sincerely appreciate the help provided by all those in the preparation of this Project.

With Sincere Regards,

**Anuja Mahendra Kapoor**

# DECLARATION

I hereby declare that the project is entitled. " **Credit Card application Management System"** done at the place where the project is done, hasnot been in any case duplicated to submit to any other university for theaward of any degree. To the best of my knowledge other than me, no one has submitted any other university

The project is done in partial fulfilment of the requirements for theaward of degree of **BACHELOR OF SCIENCE (COMPUTER SCIENCE)** to be submitted by me for SEM-V during the academic year 2023-24,,is based on actual work carried out by me under the guidance of supervision of Asst.prof.Trupti Rongare

**Signature of Student**

# TABLE OF CONTENTS

# Introduction

- Credit Card Application Management System is a web-based technology the main purpose this project is to provide all online credit card application management. User can apply credit card online and check the application status by using his/her name /mobile number/application number.
- In Credit Card Application Management System we use PHP and MySQL database. This is the project which keeps records of of admin/subadmin and also manage the credit card application. Credit Card Application Management System has three module i.e. admin, Sub-Admins and users.

## 1.1 Overview

A credit card application management system is a software platform designed to handle the entire lifecycle of credit card applications, from submission to approval or rejection, and ongoing management. Here's an overview of its functionalities:

Application Submission: Users apply for credit cards through various channels such as online forms, mobile apps, or in-person applications. The system collects applicant information including personal details, financial history, income, and more.

Data Verification: The system verifies the authenticity of the provided information by cross-referencing databases, performing credit checks, and validating documents submitted by the applicant.

Application Processing: Applications are routed through predefined workflows for evaluation. This involves risk assessment, credit scoring, and compliance checks based on predefined criteria set by the financial institution.

Decision Making: The system generates decisions (approval, rejection, or further review) based on the evaluation criteria. These decisions can be automated based on predefined rules or escalated for manual review by underwriters.

Communication: Automated notifications are sent to applicants informing them of their application status, whether approved, rejected, or pending further review. These communications may also include details about credit limits, terms, and conditions.

Account Setup: If approved, the system initiates the account setup process, generating credit card details, setting credit limits, and associating the account with the applicant.

Account Management: Once the account is active, the system facilitates ongoing management. This includes tracking transactions, payments, credit utilization, generating statements, and managing rewards or loyalty programs.

Customer Support: The system provides support to applicants and cardholders, handling inquiries, disputes, and requests for account modifications or upgrades.

Reporting and Analytics: It generates reports and analytics to track application trends, approval rates, default rates, and other key performance indicators (KPIs) for the institution's review and decision-making.

Security and Compliance: Ensuring the system complies with regulatory standards and incorporates security measures to protect sensitive customer data is paramount. This involves encryption, access controls, and adherence to industry standards.

## 1.2 Objectives

The primary objectives of a credit card management system are:

Efficiency: Streamlining the credit card application process from submission to approval/rejection and ongoing management. This includes automating tasks, reducing manual intervention, and optimizing workflows.

Risk Mitigation: Assessing the creditworthiness of applicants to minimize the risk of defaults or delinquencies. This involves thorough verification, credit scoring, and compliance checks to make informed decisions.

Customer Experience: Providing a seamless and user-friendly experience for applicants during the application process and ensuring convenience for cardholders in managing their accounts, making payments, and accessing support.

Compliance and Security: Adhering to regulatory standards and implementing robust security measures to protect sensitive customer information, prevent fraud, and ensure compliance with data protection laws.

Data Analytics and Reporting: Generating insights through data analysis for better decision-making, understanding customer behavior, identifying trends, and improving strategies related to credit card offerings and management.

Operational Excellence: Optimizing internal processes, reducing operational costs, and enhancing overall efficiency in managing credit card applications, approvals, account maintenance, and customer support.

Scalability and Integration: Building a system that can accommodate growth in application volumes, easily integrate with other banking systems or third-party services, and adapt to evolving technologies and market demands.

Revenue Generation: Maximizing revenue through effective credit card offerings, optimizing credit limits, and encouraging card usage through rewards programs while balancing risk factors.

# System Analysis

system analysis of a credit card management system involves understanding its components, functionalities, interactions, and processes. Here's a breakdown:

1. Components:

User Interface: This includes the application forms for card applicants, user dashboards for account management, and interfaces for customer support.

Database: Stores applicant information, account details, transaction history, credit limits, and other relevant data.

Processing Engine: Handles application evaluation, credit scoring, decision-making algorithms, and workflow management.

Integration Modules: Connects with credit bureaus, financial systems, fraud detection tools, and other third-party services.

2. Functionalities:

Application Submission: Allows applicants to submit their details via various channels (online forms, mobile apps, etc.).

Data Verification: Validates information provided by applicants through credit checks, document verification, and database cross-referencing.

Decision Making: Uses predefined rules or algorithms to approve, reject, or further review applications.

Account Management: Tracks transactions, payments, credit limits, and generates statements. Manages rewards, notifications, and customer support.

Reporting and Analytics: Generates reports on application trends, approval rates, default rates, and other key performance indicators.

3. Interactions:

User Interactions: Between applicants, cardholders, and customer support for inquiries, applications, and account management.

System Interactions: Communication between system components (database, processing engine, interfaces) for data retrieval, storage, and processing.

4. Processes:

Application Processing Workflow: Moves applications through stages of verification, credit scoring, decision-making, and account setup.

Account Lifecycle Management: Covers activities from account opening to ongoing account maintenance, including transaction tracking, billing, and customer support.

5. Key Considerations:

Security: Ensuring data encryption, access controls, and compliance with data protection regulations.

Scalability: Ability to handle increasing volumes of applications and account management efficiently.

Integration: Seamless integration with other systems and services for data exchange and functionality enhancement.

Performance: Optimizing speed and reliability in processing applications and managing accounts.

6. Continuous Improvement:

1. Periodic evaluations to identify bottlenecks, areas for enhancement, and adaptation to evolving technological and regulatory changes.

## 2.1 Proposed Analysis

1. System Overview:Brief description of the credit card management system, its primary objectives, and key functionalities.

2. Stakeholders:Identify stakeholders involved in the system (applicants, cardholders, customer support, administrators, etc.).

3. Functional Analysis:

Application Process:Breakdown of the application submission workflow.Verification and validation procedures for applicant data.Decision-making criteria for approval/rejection.

Account Management:Account setup process after approval.Ongoing account management functionalities (transaction tracking, statements, payments, rewards, etc.).

Customer Support:Support mechanisms provided to applicants and cardholders.Channels for communication (phone, email, chat).

4. Technical Architecture:

System Components:Description of the user interface, database, processing engine, and integration modules.Details on how these components interact and support system functionalities.

Data Handling:Data storage mechanisms (database types, encryption methods, etc.).

Data flow between different components/modules.

5. Security and Compliance:

Security Measures:

Overview of security protocols (encryption, access controls, etc.).Compliance with industry standards and data protection regulations (PCI DSS, GDPR, etc.).

Risk Management:

Measures in place to mitigate fraud, identity theft, and other risks.Monitoring and response strategies for security incidents.

6. Performance and Scalability:

System Performance:

Evaluation of system response times, reliability, and uptime.Analysis of any performance bottlenecks.

## 2.2 Requirement Analysis

Certainly, requirement analysis for a credit card management system involves understanding and documenting the needs, functionalities, and constraints of the system. Here's a structured approach:

1. Functional Requirements:Detailed forms and fields required for applicants.Multi-channel submission (online, mobile, in-person).Document upload and verification process.

Decision Making:Criteria for automated approval, rejection, or manual review.Credit scoring algorithms and parameters.Compliance checks and regulations.

Account Management:Account setup procedures after approval.Transaction tracking, statement generation, and billing.Rewards/loyalty program management.

Customer Support:Communication channels for inquiries and support.Resolution timeframes for customer queries.

2. Non-Functional Requirements: Response times for application submission and account management.System reliability and uptime expectations.

Security:Encryption standards for data transmission and storage.Access controls and user authentication mechanisms.Compliance with regulatory standards (PCI DSS, GDPR, etc.).

Scalability:Ability to handle increased application volumes and user loads.Elasticity of the system to accommodate growth.

3. User Requirements:Applicants:Intuitive and user-friendly application interface.Real-time status updates on application progress.

Cardholders:Accessible account management dashboard.Clear transaction history and billing details.Easy-to-understand reward program details.

4. System Integration Requirements:Third-Party Integration:Integration with credit bureaus for credit checks.Compatibility with financial systems for transactions.

5. Regulatory and Compliance Requirements:Legal Compliance:Adherence to financial regulations and data protection laws.Compliance with industry standards for security and privacy.

6. Reporting and Analytics Requirements: Capability to generate reports on approval rates, default rates, etc.Analytical tools for decision-making based on collected data.

7. Maintenance and Support Requirements:Procedures for updates, patches, and bug fixes.Support and maintenance service level agreements (SLAs).

8. User Acceptance Testing (UAT) Requirements:

Testing Criteria:Test scenarios for application submission, account management, etc.

## 2.3 Hardware Requirements

| | |
|---|---|
| Processor | 1.6 GHz or Faster Processor |
| RAM | 4 GB |

## 2.4 Software Requirement

| | |
|---|---|
| Operating System | Windows 10 |
| Front End | HTML, CSS, JavaScript,PHP |
| Back End | MYSQL |
| Code Editor | Visual Studio Code |

## 2.5 Identification of Need

Identifying needs within a credit card management system involves recognizing areas that require improvement, enhancement, or addressing to better serve stakeholders and improve system efficiency. Here are some key needs often found in credit card management systems:

Streamlined Application Process:

Need: Simplifying and optimizing the application process for applicants.Solution: Intuitive application forms, minimal documentation requirements, and streamlined verification processes to reduce friction and improve user experience.

Need:Efficient Decision Making: Faster and accurate decision-making for application approvals or rejections.

Solution: Implementing advanced credit scoring algorithms, automated decision-making systems, and real-time data analysis to expedite evaluations.

Enhanced Security Measures:

Need: Ensuring robust security measures to protect sensitive customer data.

Solution: Advanced encryption protocols, multi-factor authentication, and continuous monitoring for potential security threats or breaches.

Improved Customer Support:

Need: Responsive and efficient customer support for applicants and cardholders.

Solution: Implementing a multi-channel support system (phone, email, chat) with well-defined response times and comprehensive issue resolution procedures.

Personalized Account Management:

Need: Tailoring account management experiences for cardholders.

Solution: Customizable account dashboards, personalized offers, and easily accessible transaction history to enhance user engagement and satisfaction.

Scalability and Performance:

Need: Ability to handle increased application volumes and ensure system performance under load.

Solution: Scalable architecture, load testing, and continuous monitoring to maintain system performance during peak times.

Regulatory Compliance:

Need: Adherence to regulatory standards and compliance with industry regulations.

Solution: Regular audits, updates to align with changing regulations, and robust internal processes to ensure compliance.

Data Analytics for Decision-Making:

Need: Leveraging data for insights to improve decision-making processes.

Solution: Implementing advanced analytics tools to derive meaningful insights from application trends, customer behaviors, and performance indicators.

## 2.6 Software Requirements Specification (SRS)

Software Requirements Specification (SRS) for a Credit Card Management System involves outlining the functionalities, constraints, and features the system must have. Here's a template to get you started:

1. Introduction
2. Purpose: Define the purpose of the Credit Card Management System.
3. Scope: Describe the boundaries and intended users of the system.

Definitions, Acronyms, and Abbreviations: List any specific terms used throughout the document.

2. Overall Description

Product Perspective: Explain how the system interacts with other systems or applications.

Product Features: List and briefly describe the major features and functions of the system.

User Classes and Characteristics: Identify the different types of users and their roles.

Operating Environment: Specify the hardware, software, and network environments in which the system will operate.

## 3. Specific Requirements

### 3.1. External Interface Requirements

User Interfaces: Describe the interfaces users will interact with (GUI, command line, etc.).

Hardware Interfaces: Detail any hardware the system interacts with (card readers, etc.).

Software Interfaces: Specify interactions with other software systems (payment gateways, databases, etc.).

Communication Interfaces: Outline communication protocols (APIs, network protocols, etc.).

### 3.2. Functional Requirements

Authentication and Authorization: Define how users are authenticated and authorized.

Account Management: Detail how accounts are created, modified, and closed.

Transaction Processing: Explain how transactions are handled, including approvals and declines.

Credit Limit Management: Specify how credit limits are set, adjusted, and monitored.

Statement Generation: Describe how statements are generated for users.

3.3. Non-Functional Requirements

Performance: Specify response times, throughput, and system reliability.

Security: Detail security measures, encryption standards, and access controls.

Usability: Define ease of use, accessibility, and user training requirements.

Scalability: Describe how the system will handle increased load or users.

Maintainability: Outline how the system will be maintained and updated.

4. System Models (if applicable)

Use case diagrams, sequence diagrams, or any other relevant models to illustrate system behavior.

5. Other Requirements

Legal and Regulatory Requirements: Ensure compliance with relevant laws and regulations.

Documentation Requirements: Detail the documentation needed for users and developers.

Testing Requirements: Describe testing strategies, including unit testing, integration testing, etc.

6. Appendices

Additional information, diagrams, or supporting documents.

7. Glossary

Definitions of terms used in the SRS.

- Remember, this template provides a structure, but the specifics will depend on your unique project needs. Ensure clear communication between stakeholders and regular reviews to refine and update the SRS as necessary.

# Software Description

## 3.1 PHP

PHP is an open source language and all its components are free to use and distribute. PHP is server-side scripting language. It is embedded in HTML source code. PHP supports all major web servers such as Apache, Microsoft IIS and Netscape etc. All the major database such as Mysql, PostgreSQL, Oracle, Sybase, Microsoft SQL Server is supported by PHP. Following are the some major advantage:-

- Friendly With HTML - PHP and HTML are interchangeable within the page. You can put PHP outside the HTML or inside.
- Interactive Features - PHP allows you to interact with your visitors in ways HTML alone can't.
- Top-Notch Online Documentation - The PHP documentation is the best on the web. Hands down.
- Compatible With Databases - A good benefit of using PHP is that it can interact with many different database languages including MySQL.

## 3.2 MySQL

MySQL is the most popular open source relational database management system. It is one of the best RDBMS being used to develop web-based applications. It is easy to use and fast RDBMS. Following are the top reason to use MySQL:-

- High Performance
- Robust Transactional Support
- Strong Data Protection
- Open Source Freedom

## 3.3 HTML

Hypertext Markup Language (HTML) is the standard markup language for creating web pages and web applications. With Cascading Style Sheets (CSS) and JavaScript, it forms a triad of cornerstone technologies for the World Wide Web.

Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

## 3.4 CSS

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

## 3.5 JavaScript

JavaScript often abbreviated as JS, is a high-level, interpreted programming language. It is a language which is also characterized as dynamic, weakly typed, prototype-based and multi-paradigm.

## 3.6 Bootstrap

Bootstrap is an open source toolkit for developing with HTML, CSS, and JS. Quickly prototype your ideas or build your entire app with our Sass variables and mixins, responsive grid system, extensive prebuilt components, and powerful plugins built on jQuery. Build responsive, mobile-first projects on the web with the world's most popular front-end component library.

## 3.7 Paypal Payment Gateway

Collecting Online Payment for any kind of business is much eaiser with Paypal Payment Gateway. It provides a secure, PCI-compliant way to accept Debit/Credit card, and Paypal wallet payments from your customers. It also provides cancellations feature. It helps to make genuine cancellations a positive experience and maintain customer loyalty.

## 3.8 Visual Studio Code

Visual Studio Code was announced on April 29, 2015 by Microsoft at the 2015 Build conference. A Preview build was released shortly thereafter.

On November 18, 2015, Visual Studio Code was released under the MIT License and its source code posted to GitHub. Extension support was also announced.

On April 14, 2016, Visual Studio Code graduated the public preview stage and was released to web.Visual Studio Code is a source code editor developed by Microsoft for Windows, Linux and macOS. It includes support for debugging, embedded Git control, syntax highlighting, intelligent code completion, snippets, and code refactoring. It is also customizable, so users can change the editor's theme, keyboard shortcuts, and preferences. It is free and open-source, although the official download is under a proprietary license.

## 3.9 StarUML

StarUML is an open source project to develop fast, flexible, extensible and featureful diagrams . With StarUML it is very easy to make Class Diagram. StarUML is implemented to provide many user-friend features such as Quick dialog, Keyboard manipulation, Diagram overview, etc.

# System Design

**Modules:**

Admin

- **Dashboard**: In this section admin can briefly view total number of Subadmins and total number of new application, accepted application and rejected application.

- **Subadmins**: In this section, admin can manage the Sub-Admins (add/update,Delete).

- **CC Application**: In this section, admin can manage the credit card application and change the status of application according to current scenario.

- **Report**: In this section, two reports are available.

- **B/w Dates Report:** Admin can view number of credit card application received in the particular periods.

- **Search Report:** admin can search Credit Card Application by using Name/Email/Mobile Number/Application Number.

- **Pages:** In this Section, Admin can manage the content of about us and contact us page.

**Account Settings:**

- **Profile**: In this section admin can update his/her profile.
- **Change Password**: In this section admin can change his/her own passwords
- **Logout**: Through this button admin can logout.

**Forgot Password:** In this section, admin can reset his/her password by using registered email id and contact number.

.

Advantages

- A Credit Card Management System offers numerous advantages for both users and businesses, fostering efficiency, security, and convenience in financial transactions. Here are some key advantages:

- For Users:
- Convenience: Users can manage their credit cards, view transactions, and pay bills conveniently through web or mobile applications, saving time and effort.

- Transaction Tracking: Access to real-time transaction information allows users to track spending, manage budgets, and identify any fraudulent activities promptly.

Disadvantages
Debt Accumulation: Easy access to credit can lead to overspending, resulting in high-interest debt if not managed responsibly.

Interest Rates and Fees: High-interest rates and various fees (annual fees, late payment fees) can significantly increase the overall cost of borrowing.

Security Risks: Despite security measures, credit card information can be compromised, leading to fraud, identity theft, or unauthorized transactions.

Dependency on Technology: System outages, technical issues, or cyber-attacks can disrupt access to funds or account information, causing inconvenience and financial stress.

**Design Introduction:**

Design is the first step in the development phase for any techniques and principles for the purpose of defining a device, a process or system in sufficient detail to permit its physical realization.

Once the software requirements have been analyzed and specified the software design involves three technical activities - design, coding, implementation and testing that are required to build and verify the software.

The design activities are of main importance in this phase, because in this activity, decisions ultimately affecting the success of the software implementation and its ease of maintenance are made. These decisions have the final bearing upon reliability and maintainability of the system. Design is the only way to accurately translate the customer's requirements into finished software or a system.

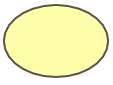
Design is the place where quality is fostered in development. Software design is a process through which requirements are translated into a representation of software. Software design is conducted in two steps. Preliminary design is concerned with the transformation of requirements into data

UML Diagrams:

Actor:

A coherent set of roles that users of use cases play when interacting with the use `cases.

Use case: A description of sequence of actions, including variants, that a system performs that yields an observable result of value of an actor.



UML stands for Unified Modeling Language. UML is a language for specifying, visualizing and documenting the system. This is the step while developing any product after analysis. The goal from this is to produce a model of the entities involved in the project which later need to be built. The representation of the entities that are to be used in the product being developed need to be designed.

**USECASE DIAGRAMS:**

Use case diagrams model behavior within a system and helps the developers understand of what theuser require. The stick man represents what's called an actor.

Use case diagram can be useful for getting an overall view of the system and clarifying who can doand more importantly what they can't do.

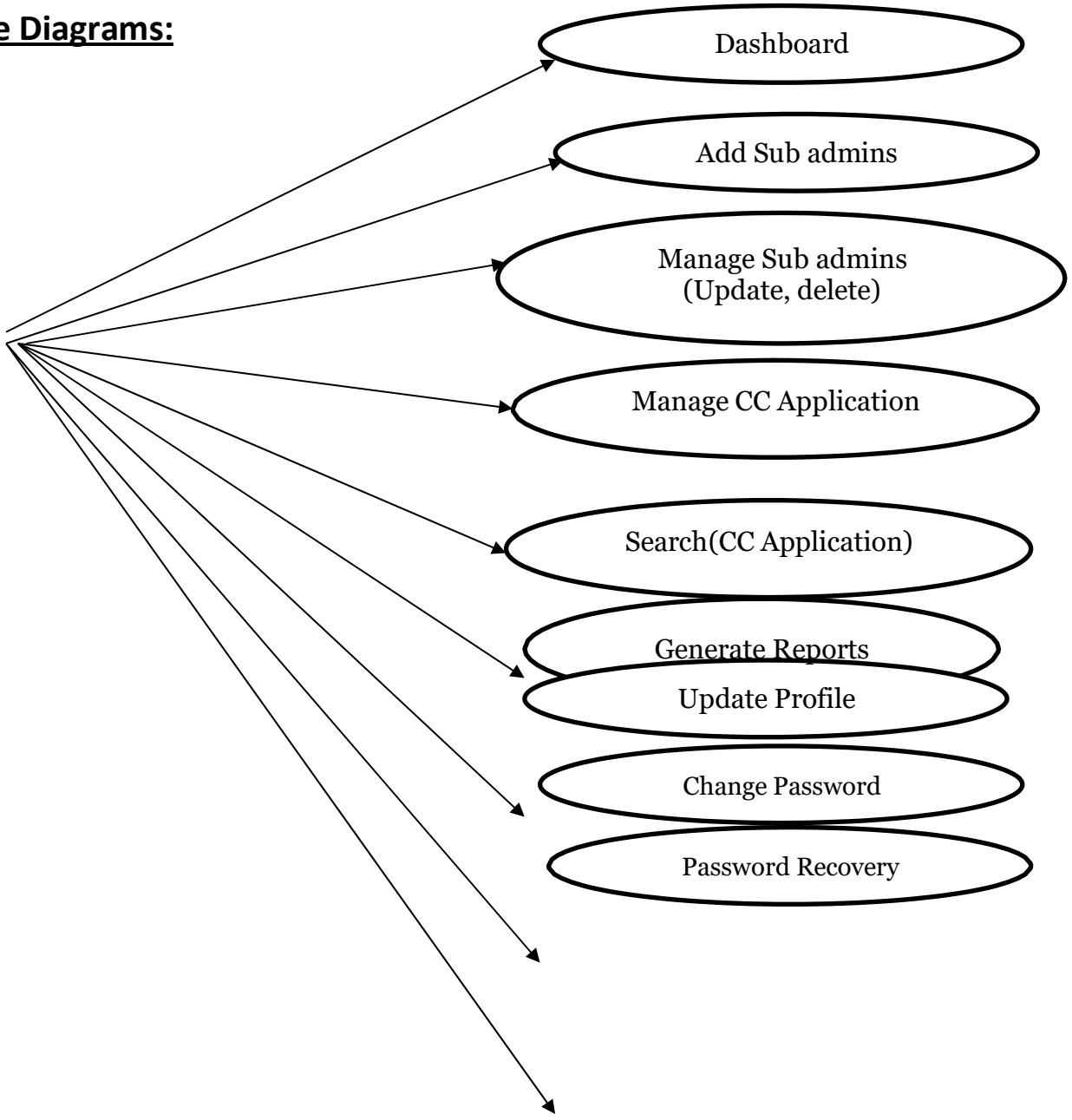
Use case diagram consists of use cases and actors and shows the interaction between the use caseand actors.

- The purpose is to show the interactions between the use case and actor.
- To represent the system requirements from user's perspective.
- An actor could be the end-user of the system or an external system.

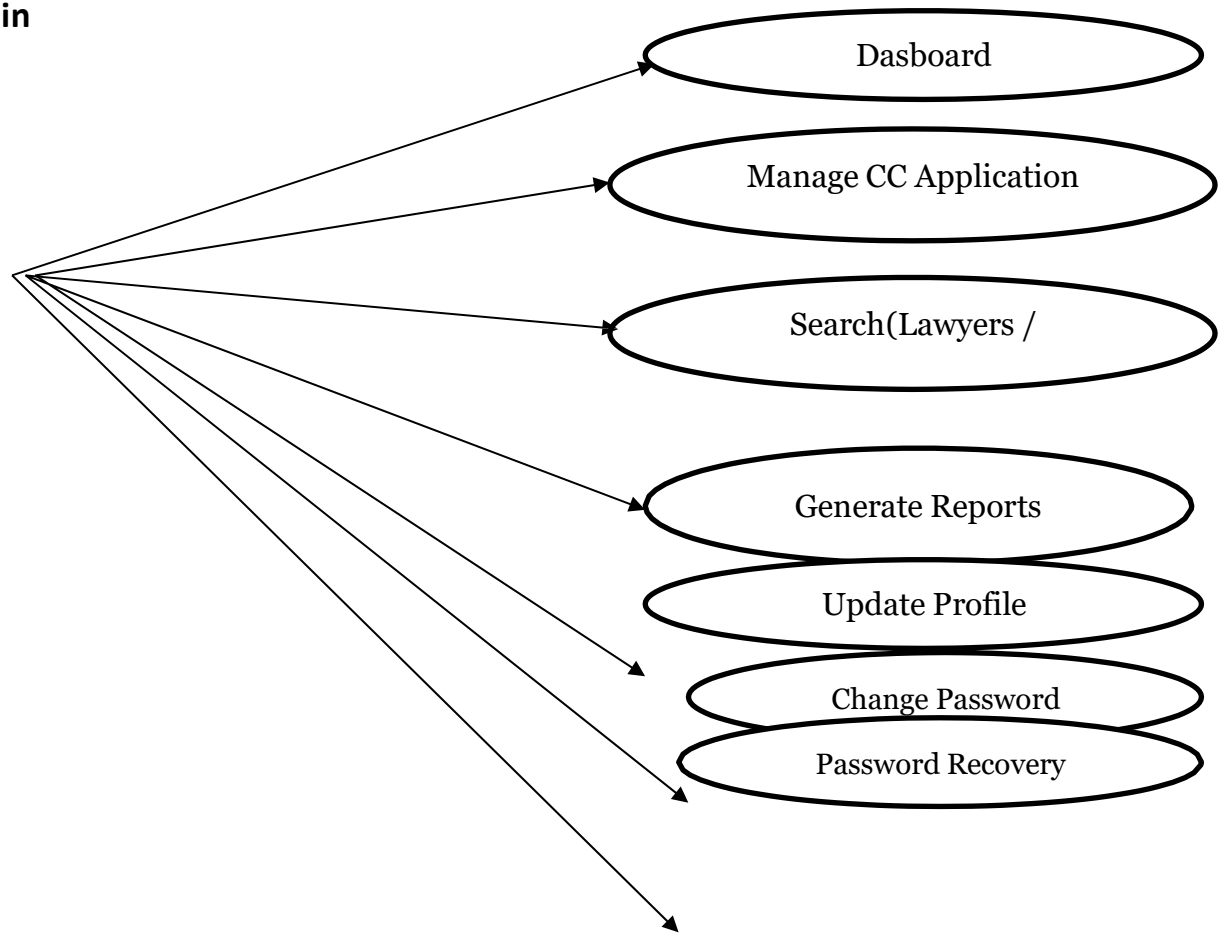
**USECASE DIAGRAM:** A Use case is a description of set of sequence of actions. Graphically it is rendered as an ellipse with solid line including only its name. Use case diagram is a behavioral diagram that shows a set of use cases and actors and their relationship. It is an association between the use cases and actors. An actor represents a real-world object. Primary Actor – Sender, SecondaryActor Receiver.

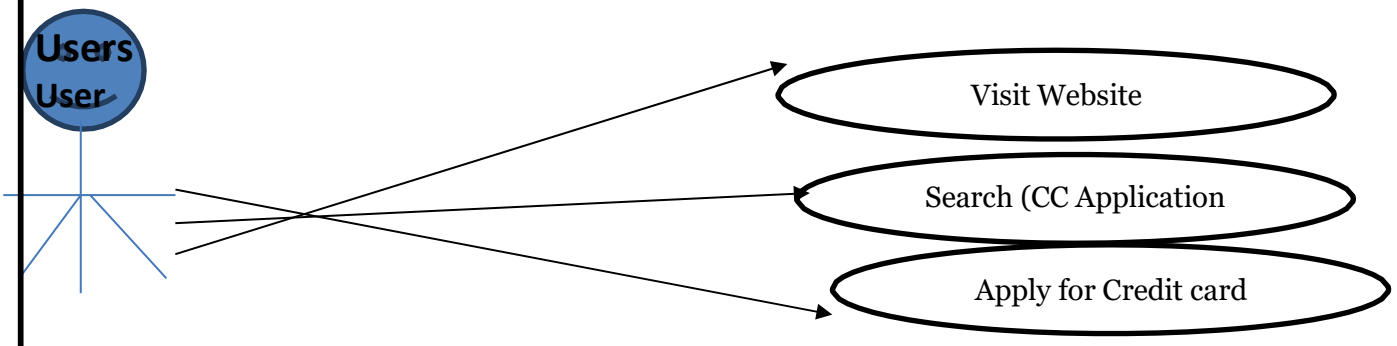
## Use Case Diagrams:

**Admin**



**Sub-Admin**


Dasboard
Manage CC Application
Search(Lawyers /
Generate Reports
Update Profile
Change Password
Password Recovery


Users
User
Visit Website
Search (CC Application
Apply for Credit card

**ER Diagram:**

The Entity-Relationship (ER) model was originally proposed by Peter in 1976 [Chen76] as a way to unify the network and relational database views. Simply stated the ER model is a conceptual data model that views the real world as entities and relationships. A basic component of the model is the Entity-Relationship diagram which is used to visually represent data objects. Since Chen wrote his paper the model has been extended and today it is commonly used for database design for the database designer, the utility of the ER model is:

- It maps well to the relational model. The constructs used in the ER model can easily be transformed into relational tables.
- It is simple and easy to understand with a minimum of training. Therefore, the model can be used by the database designer to communicate the design to the end user.
- In addition, the model can be used as a design plan by the database developer to implement a data model in specific database management software.

**ER Notation**

There is no standard for representing data objects in ER diagrams. Each modeling methodology uses its own notation. The original notation used by Chen is widely used in academics texts and journals but rarely seen in either CASE tools or publications by non-academics. Today, there are a number of notations used; among the more common are Bachman, crow's foot, and IDEFIX.

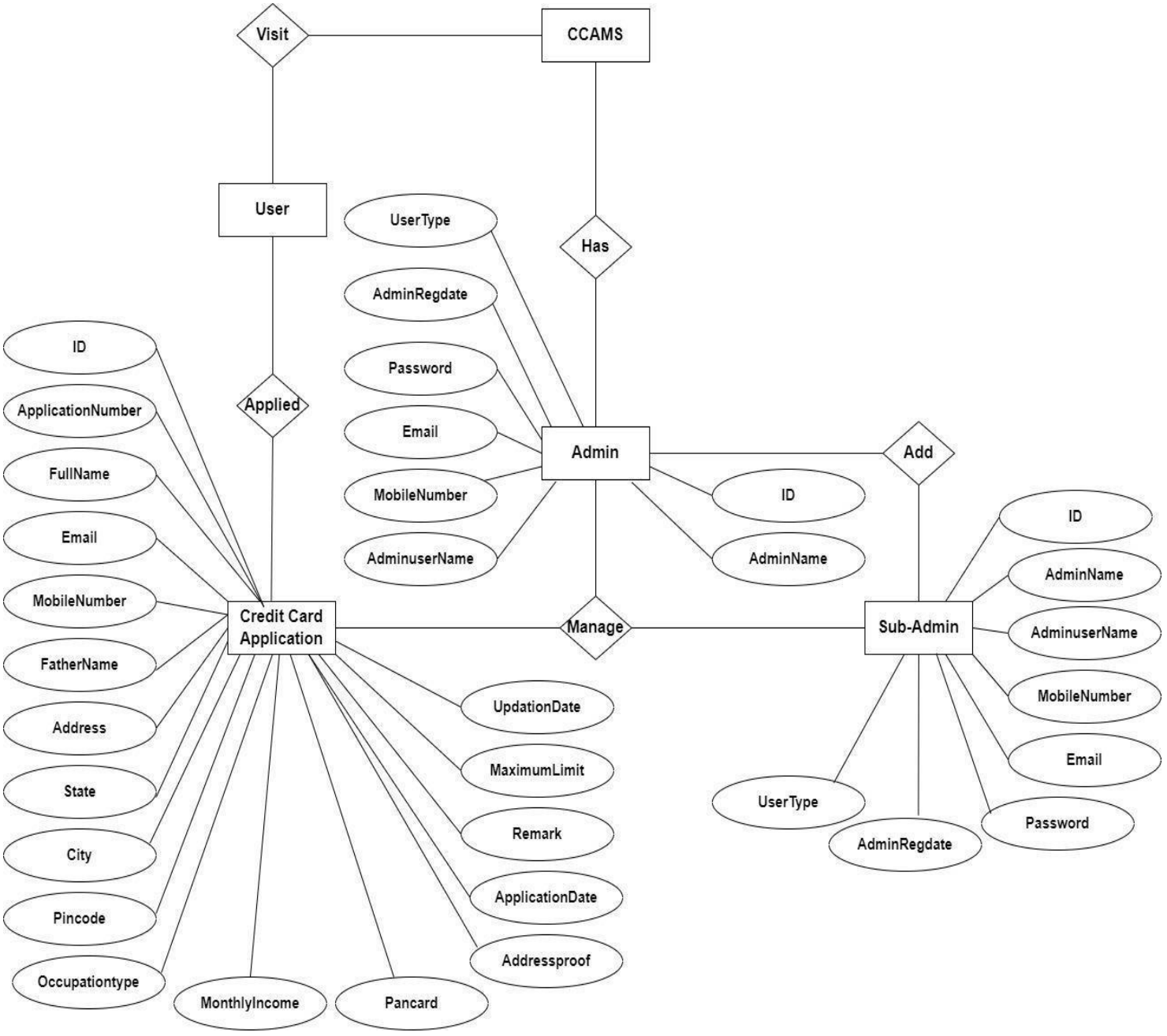
All notational styles represent entities as rectangular boxes and relationships as lines connecting boxes. Each style uses a special set of symbols to represent the cardinality of a connection. The notation used in this document is from Martin. The symbols used for the basic ER constructs are:

- **Entities** are represented by labeled rectangles. The label is the name of the entity. Entity names should be singular nouns.
- **Relationships** are represented by a solid line connecting two entities. The name of the relationship is written above the line. Relationship names should be verbs
- **Attributes**, when included, are listed inside the entity rectangle. Attributes which are identifiers

are underlined. Attribute names should be singular nouns.

- **Cardinality** of many is represented by a line ending in a crow's foot. If the crow's foot is omitted, the cardinality is one.

**Existence** is represented by placing a circle or a perpendicular bar on the line. Mandatory existence is shown by the bar (looks like a 1) next to the entity for an instance is required. Optional existence is shown by placing a circle next to the entity that is optional.


Visit
CCAMS
User
UserType
Has
AdminRegdate
ID
Password
Applied
ApplicationNumber
Email
Admin
Add
FullName
MobileNumber
ID
Email
AdminuserName
AdminName
ID
MobileNumber
AdminName
AdminuserName
Credit Card Application
Manage
Sub-Admin
FatherName
UpdationDate
Address
MaximumLimit
State
Remark
UserType
Password
City
ApplicationDate
AdminRegdate
Pincode
Addressproof
Occupationtype
MonthlyIncome
Pancard

## Data Flow Diagrams

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It can be manual, automated, or a combination of both.

It shows how data enters and leaves the system, what changes the information, and where data is stored.

The objective of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communication tool between a system analyst and any person who plays a part in the order that acts as a starting point for redesigning a system. The DFD is also called as a data flow graph or bubble chart.

**The following observations about DFDs are essential:**

1. All names should be unique. This makes it easier to refer to elements in the DFD.
2. Remember that DFD is not a flow chart. Arrows is a flow chart that represents the order of events; arrows in DFD represents flowing data. A DFD does not involve any order of events.
3. Suppress logical decisions. If we ever have the urge to draw a diamond-shaped box in a DFD, suppress that urge! A diamond-shaped box is used in flow charts to represents decision points with multiple exists paths of which the only one is taken. This implies an ordering of events, which makes no sense in a DFD.
4. Do not become bogged down with details. Defer error conditions and error handling until the end of the analysis.

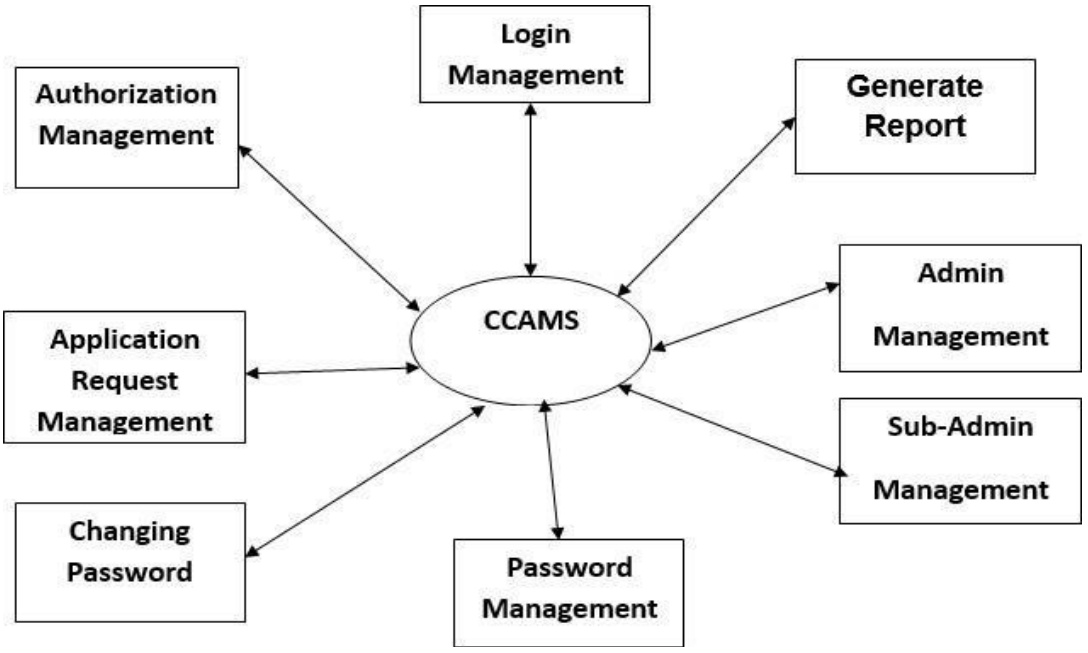
Standard symbols for DFDs are derived from the electric circuit diagram analysis and are shown in fig:

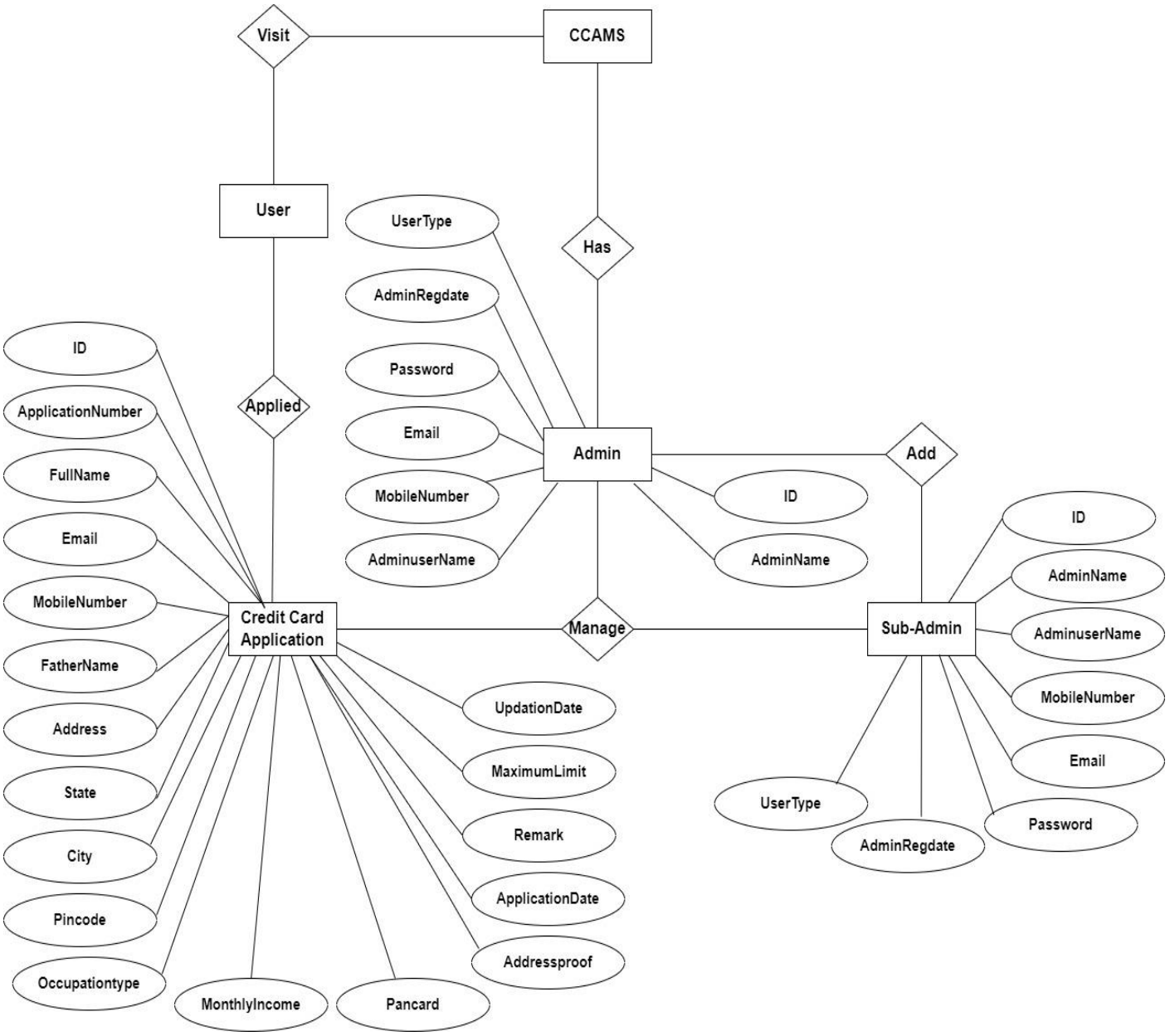
**Circle**: A circle (bubble) shows a process that transforms data inputs into data outputs.

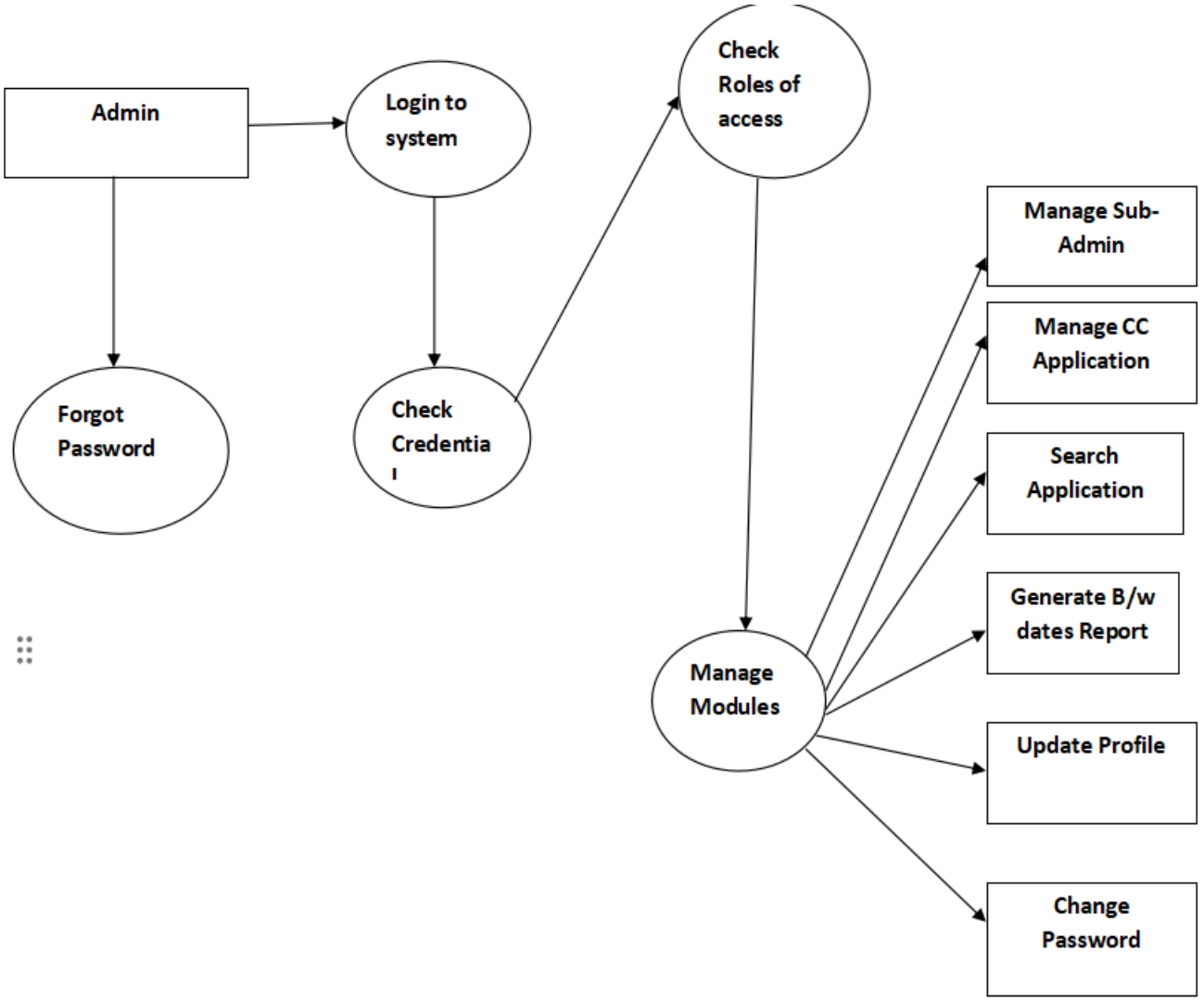
**Data Flow**: A curved line shows the flow of data into or out of a process or data store.

**Data Store**: A set of parallel lines shows a place for the collection of data items. A data store indicates that the data is stored which can be used at a later stage or by the other processes in a different order. The data store can have an element or group of elements.

**Source or Sink**: Source or Sink is an external entity and acts as a source of system inputs or sink of system outputs.


Login Management
Authorization Management
Generate Report
Admin Management
CCAMS
Application Request Management
Sub-Admin Management
Changing Password
Password Management


Visit
CCAMS
User
UserType
Has
AdminRegdate
ID
Password
Applied
ApplicationNumber
Email
Admin
Add
FullName
MobileNumber
ID
Email
AdminuserName
AdminName
ID
AdminName
MobileNumber
Credit Card Application
Manage
Sub-Admin
FatherName
AdminuserName
Address
MobileNumber
UpdationDate
State
Email
MaximumLimit
UserType
City
Remark
Password
AdminRegdate
Pincode
ApplicationDate
Occupationtype
Addressproof
MonthlyIncome
Pancard


Admin
Login to system
Check Roles of access
Forgot Password
Check Credentia l
Manage Sub-Admin
Manage CC Application
Search Application
Generate B/w dates Report
Manage Modules
Update Profile
Change Password

# MySQL Data Tables

**Admin Table :**(Table name is admin)

This table stores admin / Sub-Admins personal and login details.

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra |
|---|---|---|---|---|---|---|---|---|
| 1 | ID | int(11) | | | No | *None* | | AUTO_INCREMENT |
| 2 | AdminName | varchar(120) | latin1_swedish_ci | | Yes | *NULL* | | |
| 3 | AdminuserName | varchar(20) | latin1_swedish_ci | | Yes | *NULL* | | |
| 4 | MobileNumber | bigint(10) | | | Yes | *NULL* | | |
| 5 | Email | varchar(120) | latin1_swedish_ci | | Yes | *NULL* | | |
| 6 | Password | varchar(120) | latin1_swedish_ci | | Yes | *NULL* | | |
| 7 | AdminRegdate | timestamp | | | Yes | current_timestamp() | | |
| 8 | UserType | int(1) | | | Yes | *NULL* | | |

Indexes

| Keyname | Type | Unique | Packed | Column | Cardinality | Collation | Null | Comment |
|---|---|---|---|---|---|---|---|---|
| PRIMARY | BTREE | Yes | No | ID | 3 | A | No | |

**Credit Card Application Table:** (Table name is tblapplication)

This table stores the details of credit card application which is received by users.

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra |
|---|---|---|---|---|---|---|---|---|
| 1 | **ID** | int(5) | | | No | *None* | | AUTO_INCREMENT |
| 2 | **ApplicationNumber** | varchar(250) | latin1_swedish_ci | | Yes | *NULL* | | |
| 3 | **FullName** | varchar(250) | latin1_swedish_ci | | Yes | *NULL* | | |
| 4 | **Email** | varchar(250) | latin1_swedish_ci | | Yes | *NULL* | | |
| 5 | **MobileNumber** | bigint(10) | | | Yes | *NULL* | | |
| 6 | **FatherName** | varchar(250) | latin1_swedish_ci | | Yes | *NULL* | | |
| 7 | **Address** | mediumtext | latin1_swedish_ci | | Yes | *NULL* | | |
| 8 | **State** | varchar(250) | latin1_swedish_ci | | Yes | *NULL* | | |
| 9 | **City** | varchar(250) | latin1_swedish_ci | | Yes | *NULL* | | |
| 10 | **Pincode** | int(10) | | | Yes | *NULL* | | |
| 11 | **Occupationtype** | varchar(250) | latin1_swedish_ci | | Yes | *NULL* | | |
| 12 | **MonthlyIncome** | decimal(10,0) | | | Yes | *NULL* | | |
| 13 | **Pancard** | varchar(250) | latin1_swedish_ci | | Yes | *NULL* | | |
| 14 | **Addressproof** | varchar(250) | latin1_swedish_ci | | Yes | *NULL* | | |
| 15 | **ApplicationDate** | timestamp | | | Yes | current_timestamp() | | |
| 16 | **Status** | varchar(250) | latin1_swedish_ci | | Yes | *NULL* | | |
| 17 | **Remark** | varchar(250) | latin1_swedish_ci | | Yes | *NULL* | | |
| 18 | **MaximumLimit** | decimal(10,0) | | | Yes | *NULL* | | |
| 19 | **UpdationDate** | timestamp | | | Yes | *NULL* | | ON UPDATE CURRENT_TIMESTAMP() |

Indexes

| Keyname | Type | Unique | Packed | Column | Cardinality | Collation | Null | Comment |
|---|---|---|---|---|---|---|---|---|
| **PRIMARY** | BTREE | Yes | No | ID | 2 | A | No | |

## **Page Table:** (Table name is tblpage)

This table stores the content of about us and contact us page.

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra |
|---|---|---|---|---|---|---|---|---|
| 1 | **ID** | int(10) | | | No | *None* | | AUTO_INCREMENT |
| 2 | **PageType** | varchar(200) | utf8mb4_general_ci | | Yes | *NULL* | | |
| 3 | **PageTitle** | varchar(200) | utf8mb4_general_ci | | Yes | *NULL* | | |
| 4 | **PageDescription** | mediumtext | utf8mb4_general_ci | | Yes | *NULL* | | |
| 5 | **Email** | varchar(200) | utf8mb4_general_ci | | Yes | *NULL* | | |
| 6 | **MobileNumber** | bigint(10) | | | Yes | *NULL* | | |
| 7 | **UpdationDate** | timestamp | | | Yes | *NULL* | | ON UPDATE CURRENT_TIMESTAMP() |

Indexes

| Keyname | Type | Unique | Packed | Column | Cardinality | Collation | Null | Comment |
|---|---|---|---|---|---|---|---|---|
| **PRIMARY** | BTREE | Yes | No | ID | 2 | A | No | |

# SAMPLE CODE

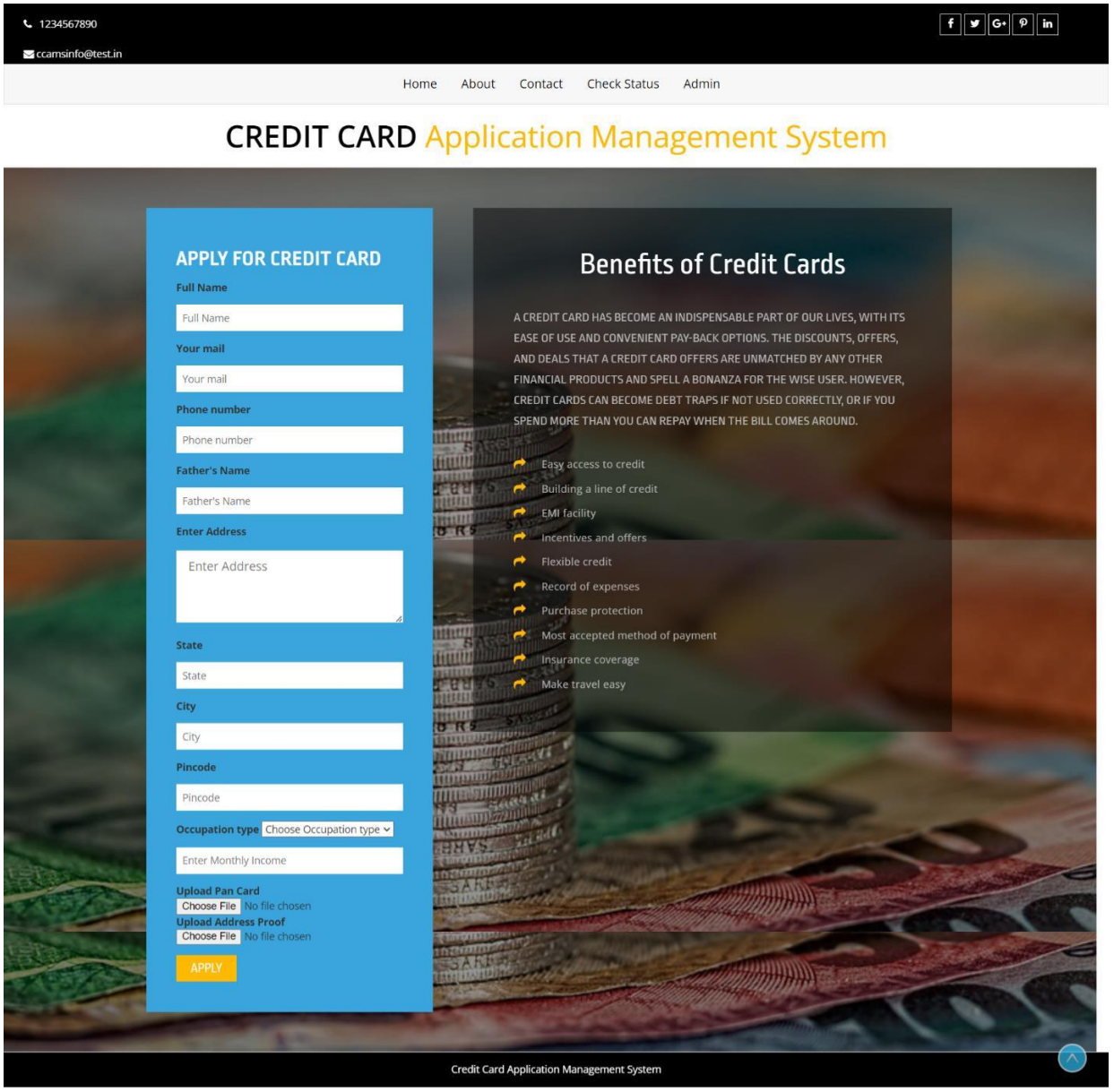
```
<?php session_start();
// Database Connection
include('includes/config.php');
//Validating Session
if(strlen($_SESSION['aid'])==0)
  {
header('location:index.php');
}
else{
// Code for update the about us content
if(isset($_POST['submit']))
  {
$pagetitle=$_POST['pagetitle'];
$pagedes=$con->real_escape_string($_POST['pagedes']);
$query=mysqli_query($con,"update tblpage set PageTitle='$pagetitle',PageDescription='$pagedes' where PageType='aboutus'");
if ($query) {
echo '<script>alert("About Us has been updated.")</script>';
}else{
echo '<script>alert("Something Went Wrong. Please try again.")</script>';
}}
?>
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>CCAMS | About us</title>

  <!-- Google Font: Source Sans Pro -->
  <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,400i,700&display=fallback">
  <!-- Font Awesome -->
  <link rel="stylesheet" href="../plugins/fontawesome-free/css/all.min.css">
  <!-- Theme style -->
  <link rel="stylesheet" href="../dist/css/adminlte.min.css">
  <!--Function Email Availabilty---->
<script src="nic.js" type="text/javascript"></script>
<script type="text/javascript">bkLib.onDomLoaded(nicEditors.allTextAreas);</script>
</head>

</head>
<body class="hold-transition sidebar-mini">
<div class="wrapper">
  <!-- Navbar -->
<?php include_once("includes/navbar.php");?>
  <!-- /.navbar -->

  <!-- Main Sidebar Container -->
 <?php include_once("includes/sidebar.php");?>

  <!-- Content Wrapper. Contains page content -->
  <div class="content-wrapper">
    <!-- Content Header (Page header) -->
    <section class="content-header">
      <div class="container-fluid">
        <div class="row mb-2">
          <div class="col-sm-6">
```

```
            <h1>About us</h1>
          </div>
          <div class="col-sm-6">
            <ol class="breadcrumb float-sm-right">
              <li class="breadcrumb-item"><a href="dashboard.php">Dashboard</a></li>
              <li class="breadcrumb-item active">About us</li>
            </ol>
          </div>
        </div>
      </div><!-- /.container-fluid -->
    </section>

    <!-- Main content -->
    <section class="content">
      <div class="container-fluid">
        <div class="row">
          <!-- left column -->
          <div class="col-md-8">
            <!-- general form elements -->
            <div class="card card-primary">
              <div class="card-header">
                <h3 class="card-title">Fill the Info</h3>
              </div>
              <!-- /.card-header -->
              <!-- form start -->
              <form name="subadmin" method="post">
                <div class="card-body">
<?php
$ret=mysqli_query($con,"select * from  tblpage where PageType='aboutus'");
$cnt=1;
while ($row=mysqli_fetch_array($ret)) {
?>
<!--Page Title--->
   <div class="form-group">
                    <label for="exampleInputFullname">Page Title</label>
    <input type="text" class="form-control" name="pagetitle" value="<?php  echo $row['PageTitle'];?>" required='true'>
                  </div>


<!--Description--->
   <div class="form-group">
                    <label for="exampleInputFullname">Page Description</label>
  <textarea  name="pagedes" class="form-control" required='true' cols="140" rows="10"><?php  echo
$row['PageDescription'];?></textarea>
                  </div>
<?php } ?>

                </div>
                <!-- /.card-body -->
                <div class="card-footer">
                  <button type="submit" class="btn btn-primary" name="submit" id="submit">Submit</button>
                </div>
              </form>
            </div>
            <!-- /.card -->



          </div>
          <!--/.col (left) -->

        </div>
```

```php
<?php session_start();
// Database Connection
include('includes/config.php');
//Validating Session
if(strlen($_SESSION['aid'])==0)
  {
header('location:index.php');
}
else{
// Code for Add New Sub Admi
if(isset($_POST['submit'])){
$username=$_POST['sadminusername'];
$fname=$_POST['fullname'];
$email=$_POST['emailid'];
$mobileno=$_POST['mobilenumber'];
$password=md5($_POST['inputpwd']);
$usertype='0';
$query=mysqli_query($con,"insert into tbladmin(AdminuserName,AdminName,MobileNumber,Email,Password,UserType )
values('$username','$fname','$mobileno','$email','$password','$usertype')");
if($query){
echo "<script>alert('Sub admin details added successfully.');</script>";
echo "<script type='text/javascript'> document.location = 'add-subadmin.php'; </script>";
} else {
echo "<script>alert('Something went wrong. Please try again.');</script>";
}
}

  ?>
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>CCAMS  | Add Sub admin</title>

  <!-- Google Font: Source Sans Pro -->
  <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,400i,700&display=fallback">
  <!-- Font Awesome -->
  <link rel="stylesheet" href="../plugins/fontawesome-free/css/all.min.css">
  <!-- Theme style -->
  <link rel="stylesheet" href="../dist/css/adminlte.min.css">
  <!--Function Email Availabilty---->
<script>
function checkAvailability() {
$("#loaderIcon").show();
jQuery.ajax({
url: "check_availability.php",
data:'username='+$("#sadminusername").val(),
type: "POST",
success:function(data){
$("#user-availability-status").html(data);
$("#loaderIcon").hide();
},
error:function (){}
});
}
</script>

</head>
<body class="hold-transition sidebar-mini">
<div class="wrapper">
```

```
  <!-- Navbar -->
<?php include_once("includes/navbar.php");?>
  <!-- /.navbar -->

<?php session_start();
// Database Connection
include('includes/config.php');
//Validating Session
if(strlen($_SESSION['aid'])==0)
  {
header('location:index.php');
}
else{ ?>
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>CCAMS | Dashboard</title>

  <!-- Google Font: Source Sans Pro -->
  <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,400i,700&display=fallback">
  <!-- Font Awesome -->
  <link rel="stylesheet" href="../plugins/fontawesome-free/css/all.min.css">
  <!-- Ionicons -->
  <link rel="stylesheet" href="https://code.ionicframework.com/ionicons/2.0.1/css/ionicons.min.css">
  <!-- Tempusdominus Bootstrap 4 -->
  <link rel="stylesheet" href="../plugins/tempusdominus-bootstrap-4/css/tempusdominus-bootstrap-4.min.css">
  <!-- iCheck -->
  <link rel="stylesheet" href="../plugins/icheck-bootstrap/icheck-bootstrap.min.css">
  <!-- JQVMap -->
  <link rel="stylesheet" href="../plugins/jqvmap/jqvmap.min.css">
  <!-- Theme style -->
  <link rel="stylesheet" href="../dist/css/adminlte.min.css">
  <!-- overlayScrollbars -->
  <link rel="stylesheet" href="../plugins/overlayScrollbars/css/OverlayScrollbars.min.css">
  <!-- Daterange picker -->
  <link rel="stylesheet" href="../plugins/daterangepicker/daterangepicker.css">
  <!-- summernote -->
  <link rel="stylesheet" href="../plugins/summernote/summernote-bs4.min.css">
</head>
<body class="hold-transition sidebar-mini layout-fixed">
<div class="wrapper">



  <!-- Navbar -->
<?php include_once('includes/navbar.php');?>
```

# RESULTS


1234567890
ccamsinfo@test.in
Home
About
Contact
Check Status
Admin
CREDIT CARD Application Management System
APPLY FOR CREDIT CARD
Full Name
Full Name
Your mail
Your mail
Phone number
Phone number
Father's Name
Father's Name
Enter Address
Enter Address
State
State
City
City
Pincode
Pincode
Occupation type
Choose Occupation type
Enter Monthly Income
Upload Pan Card
Choose File
No file chosen
Upload Address Proof
Choose File
No file chosen
APPLY
Benefits of Credit Cards
A CREDIT CARD HAS BECOME AN INDISPENSABLE PART OF OUR LIVES, WITH ITS EASE OF USE AND CONVENIENT PAY-BACK OPTIONS. THE DISCOUNTS, OFFERS, AND DEALS THAT A CREDIT CARD OFFERS ARE UNMATCHED BY ANY OTHER FINANCIAL PRODUCTS AND SPELL A BONANZA FOR THE WISE USER. HOWEVER, CREDIT CARDS CAN BECOME DEBT TRAPS IF NOT USED CORRECTLY, OR IF YOU SPEND MORE THAN YOU CAN REPAY WHEN THE BILL COMES AROUND.
Easy access to credit
Building a line of credit
EMI facility
Incentives and offers
Flexible credit
Record of expenses
Purchase protection
Most accepted method of payment
Insurance coverage
Make travel easy
Credit Card Application Management System

1234567890

ccamsinfo@test.in

Home About Contact Check Status Admin

# CREDIT CARD Application Management System

## About

ABOUT US

A credit card is a type of credit facility, provided by banks that allow customers to borrow funds within a pre-approved credit limit. It enables customers to make purchase transactions on goods and services. The credit card limit is determined by the credit card issuer based on factors such as income and credit score, which also decides the credit limit. The credit card information includes the credit card number, cardholder's name, expiration date, signature, CVC code, etc. The best part about a credit card is that it is not linked to a bank account. So, whenever you swipe your credit card, the amount is deducted from your credit card limit, not your bank account. You can use it to pay for food, and clothes, take care of medical expenses, travel expenses, and other lifestyle products and emergency services. .

Credit Card Application Management System

1234567890

ccamsinfo@test.in

Home About Contact Check Status Admin

# CREDIT CARD Application Management System

## Contact

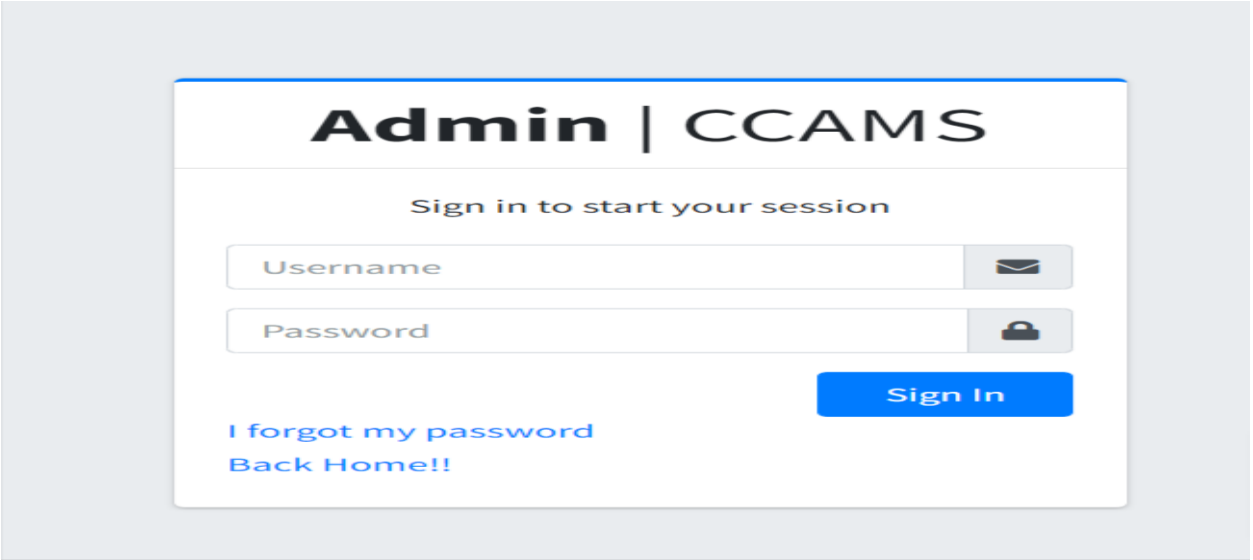
CONTACT US

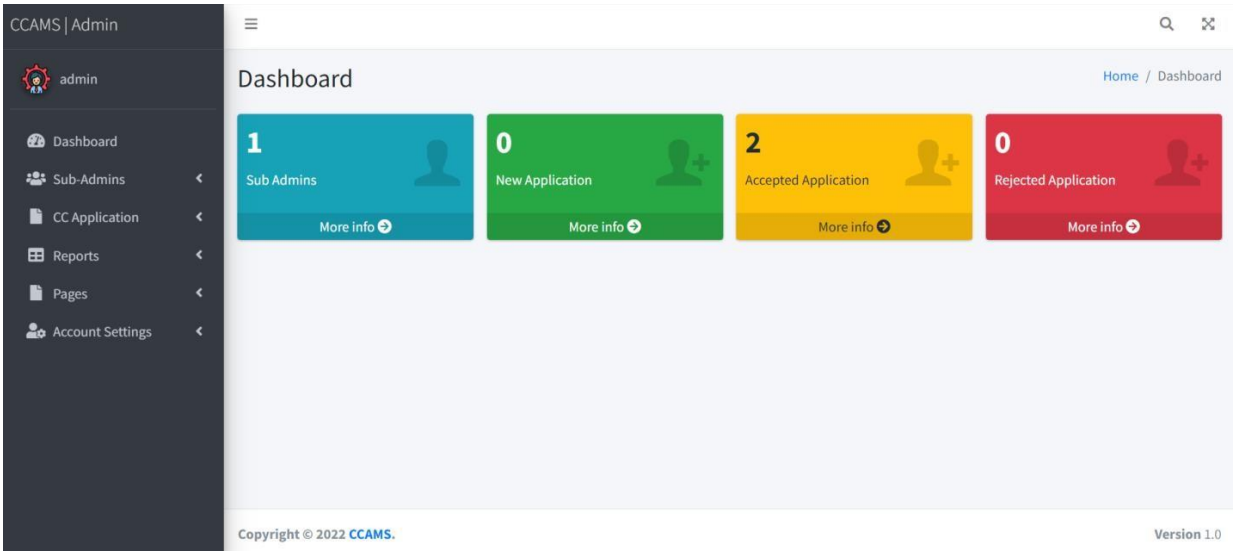
#890 CFG Apartment, Mayur Vihar, Delhi-India..

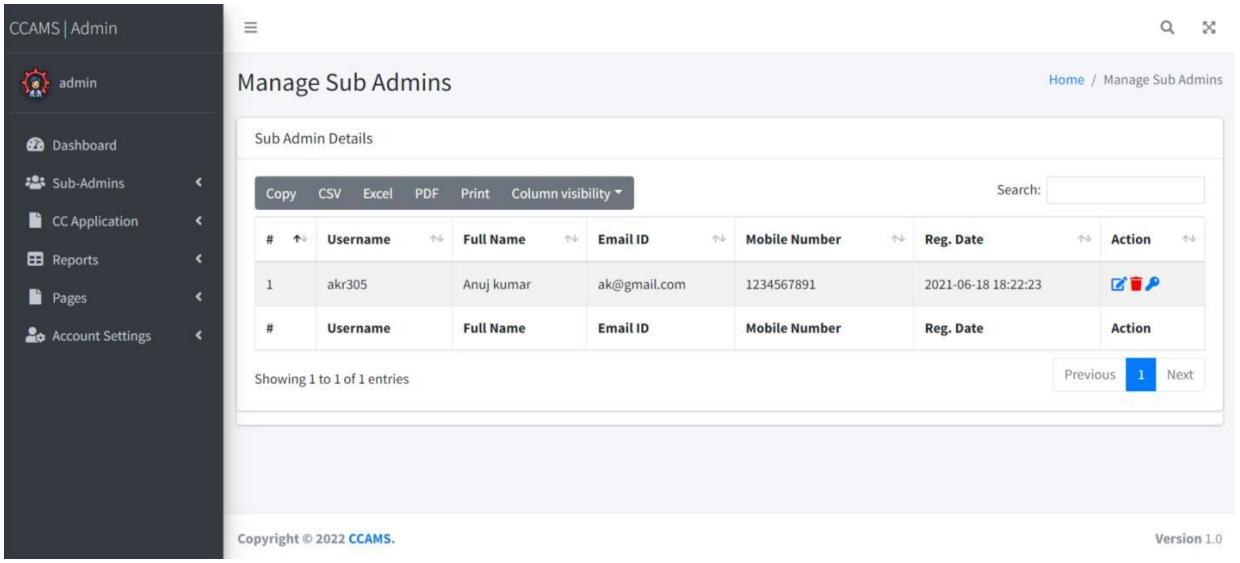
MESSAGE US

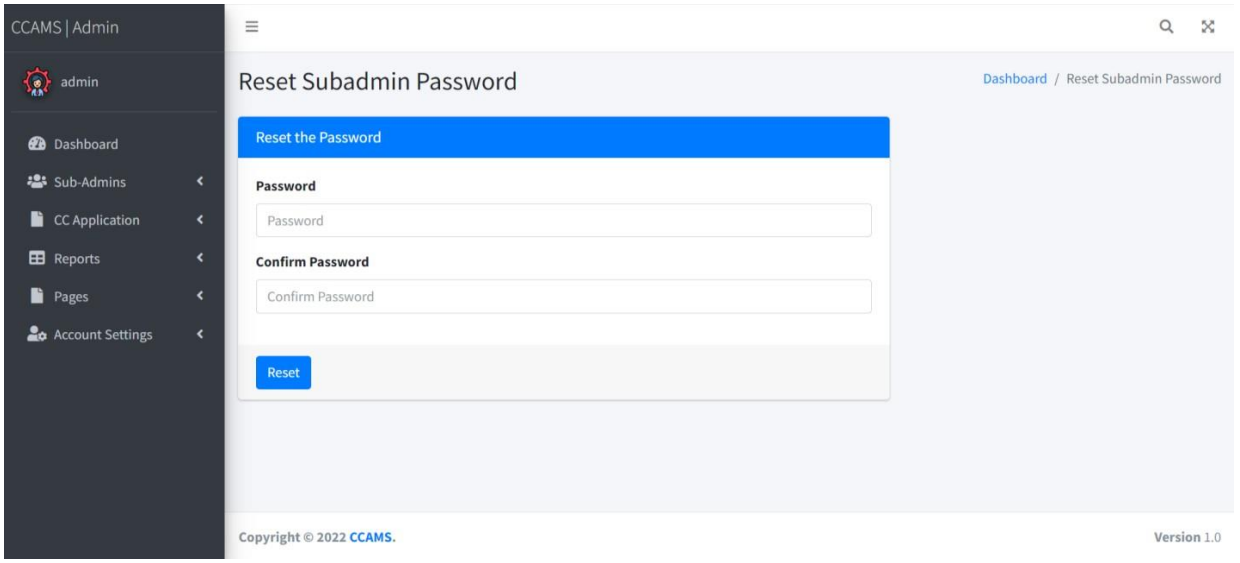
- 1234567890
- ccamsinfo@test.in

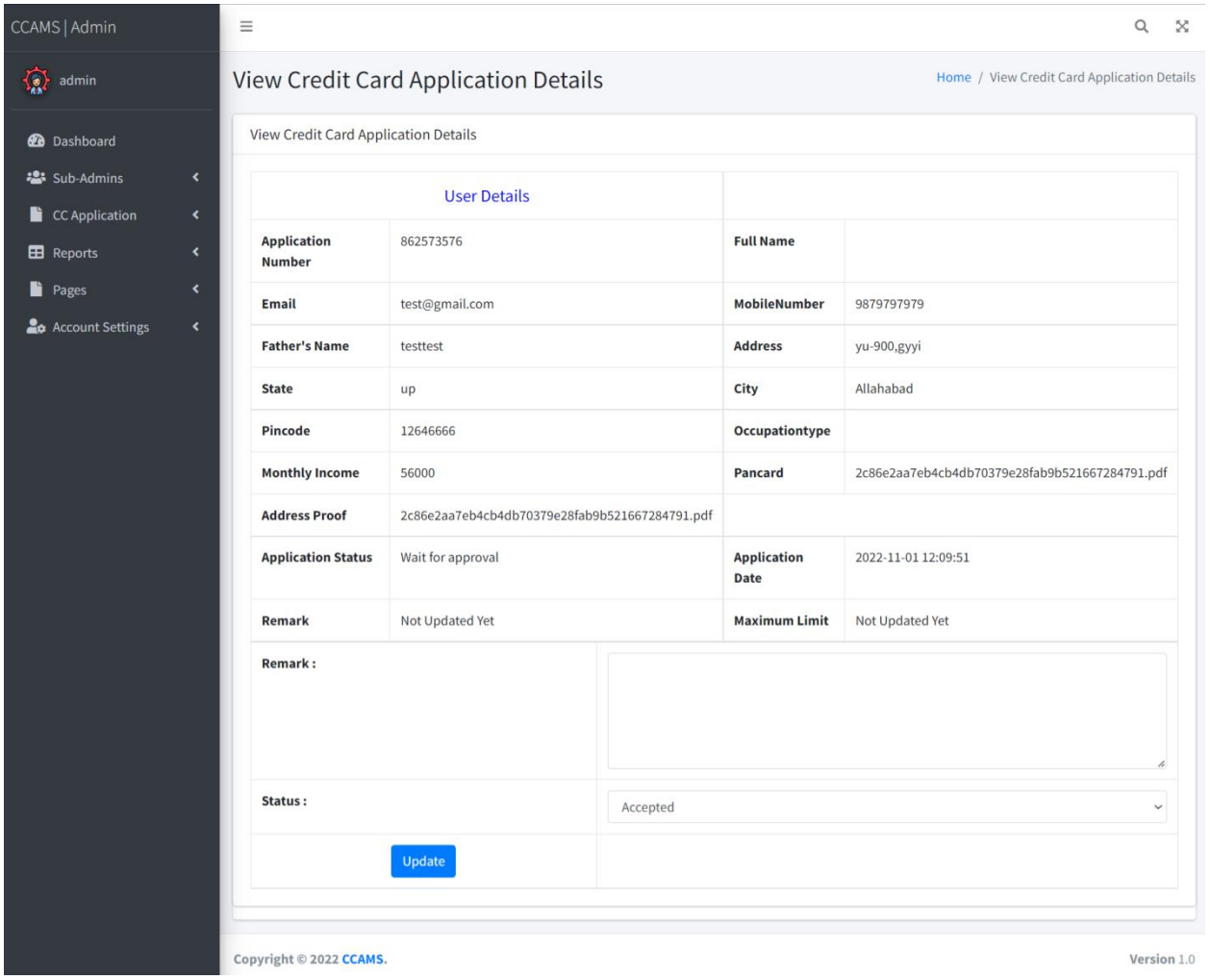
Credit Card Application Management System


Admin | CCAMS
Sign in to start your session
Username
Password
Sign In
I forgot my password
Back Home!!


CCAMS | Admin
admin
Dashboard
Sub-Admins
CC Application
Reports
Pages
Account Settings
Dashboard
Home / Dashboard
1
Sub Admins
More info
0
New Application
More info
2
Accepted Application
More info
0
Rejected Application
More info
Copyright © 2022 CCAMS.
Version 1.0


CCAMS | Admin
admin
Dashboard
Sub-Admins
CC Application
Reports
Pages
Account Settings
Manage Sub Admins
Home / Manage Sub Admins
Sub Admin Details
Copy CSV Excel PDF Print Column visibility
Search:
# Username Full Name Email ID Mobile Number Reg. Date Action
1 akr305 Anuj kumar ak@gmail.com 1234567891 2021-06-18 18:22:23
Showing 1 to 1 of 1 entries
Previous 1 Next
Copyright © 2022 CCAMS.
Version 1.0


CCAMS | Admin
admin
Dashboard
Sub-Admins
CC Application
Reports
Pages
Account Settings
Reset Subadmin Password
Dashboard / Reset Subadmin Password
Reset the Password
Password
Password
Confirm Password
Confirm Password
Reset
Copyright © 2022 CCAMS.
Version 1.0


CCAMS | Admin
admin
Dashboard
Sub-Admins
CC Application
Reports
Pages
Account Settings
View Credit Card Application Details
Home / View Credit Card Application Details
View Credit Card Application Details
User Details
Application Number
862573576
Full Name
Email
test@gmail.com
MobileNumber
9879797979
Father's Name
testtest
Address
yu-900,gyyi
State
up
City
Allahabad
Pincode
12646666
Occupationtype
Monthly Income
56000
Pancard
2c86e2aa7eb4cb4db70379e28fab9b521667284791.pdf
Address Proof
2c86e2aa7eb4cb4db70379e28fab9b521667284791.pdf
Application Status
Wait for approval
Application Date
2022-11-01 12:09:51
Remark
Not Updated Yet
Maximum Limit
Not Updated Yet
Remark :
Status :
Accepted
Update
Copyright © 2022 CCAMS.
Version 1.0

# Features of Credit Card Application Management System

- Easy to use.  It is completely secure.
- It is completely controlled by admin.
- This system is easily compatible with most of the web browsers.
- It is very interactive and saves time.
- Reduces paper works.
- Calculations are automated so it is highly accurate.
- Admin can view all the records whenever necessary with ease

# Limitation Of The Project

Security Concerns: Despite advancements in security measures, there's always a risk of data breaches or cyber attacks. Protecting sensitive customer information is crucial, and any system is vulnerable to potential threats.

Processing Delays: High volumes of applications can sometimes lead to processing delays, especially during peak times. This can result in customer dissatisfaction or lost opportunities if approvals take too long.

Scalability Issues: As the number of applications grows, the system might struggle to scale up efficiently, leading to performance issues or system downtime.

Complexity in Integration: Integrating with other systems or third-party services can be complex. Compatibility issues or the need for constant updates to ensure smooth functioning can be challenging.

Regulatory Compliance: Adhering to changing regulations and compliance standards can be demanding. Keeping the system updated with the latest requirements can be both time-consuming and resource-intensive.

Risk Assessment Accuracy: While systems use algorithms to assess creditworthiness, there can still be limitations in accurately predicting risks associated with individual applicants, leading to potential errors in approvals or rejections.

Customer Experience: Automated systems might lack the personal touch that some customers prefer. The inability to handle nuanced situations or complex inquiries might impact customer satisfaction.

Maintenance and Upkeep: Maintenance costs and the need for continuous updates to keep up with technological advancements can be high. Outdated systems might become inefficient or vulnerable to security threats.

Fraud Detection: While systems are equipped with fraud detection mechanisms, some sophisticated fraudulent activities might slip through the cracks, leading to financial losses.

Dependency on Technology: Any system outage or technological failure could disrupt the entire application management process, affecting business operations and customer service.Future

# Scope

Application Processing: Managing the entire lifecycle of credit card applications, from submission to approval or rejection. This includes data collection, verification, credit assessment, and decision-making.
Data Management: Storing and organizing applicant information securely, including personal details, financial data, credit history, and other relevant information necessary for the application process.
Automated Decision-making: Utilizing algorithms and scoring models to assess an applicant's creditworthiness based on predefined criteria, which helps in making informed decisions regarding approvals or rejections.
Documentation and Compliance: Ensuring all necessary documents and disclosures are provided by applicants and complying with regulatory requirements, such as Know Your Customer (KYC) and Anti-Money Laundering (AML) regulations.
Communication and Notifications: Facilitating communication with applicants regarding the status of their applications, providing notifications for approvals, rejections, or pending requirements, and offering support throughout the process

# Conclusion

This Application provides an online version of Credit Card Application Management System which will benefit the bank firm who want to maintain records of credit card holder and also help to customer to apply credit card online without wasting a time and apply with their convenience.

It makes entire process online and can generate reports.

The Application was designed in such a way that future changes can be done easily. The following conclusions can be deduced from the development of the project.

- Automation of the entire system improves the productivity.
- It provides a friendly graphical user interface which proves to be better when compared to the existing system.
- It gives appropriate access to the authorized users depending on their permissions.
- It effectively overcomes the delay in communications.
- Updating of information becomes so easier.
- System security, data security and reliability are the striking features.
- The System has adequate scope for modification in future if it is necessary.

# References

**For PHP**

- https://www.w3schools.com/php/default.asp
- https://www.sitepoint.com/php/
- https://www.php.net/

**For MySQL**

- https://www.mysql.com/
- http://www.mysqltutorial.org

**For XAMPP**

- https://www.apachefriends.org/download.html