Group members:

1.Brian Kipkogei: 10604811

2.Anand Singh: 10641441

3.Abdul Rehman: 10639033

4.Shubham Chaudhary: 20006082

# Introduction.

In this project we are working to create a User Management System for Data linking between the user and database through a web application where the System Administrators control all the users data by having access to read, edit and delete as well as Create, Read, Update and Delete (CRUD) Operations on user data through a web application.

# System Requirements.

1. System Administrators:

Key resonsibilities:

* Are the primary users of the system who have more access and control over the whole system.
* They have the full access control to read, edit and delete user data through the web application.
* Manage and maintain the system including user account creation, configuring system settings, configuring system settings, monitoring activities in the system, assigning roles and permissions and ensuring the security and integrity of the system and user data.

1. End users:

Key responsibilities:

* They interact with the system to perform specific tasks and access resources based on their assigned roles and permissions.
* End users include the employees, customers and any other individuals who require to use the system.
* They have limited privileges as they can only view, update or delete their own personal data from the system.
* Their access control and permissions are granted and defined by the system administrator.

# System overview.

User Management System is a system that offers data linking between the user and user data in a database through a web application by use of the CRUD operations functionalities on user accounts and read, edit and delete functionalities on user data in the database done by the system administrator.

System architecture.

| WEB APPLICATION. |
| --- |
| USER INTERFACE COMPONENTS. |
| BUSINESS LOGIC LAYER. |
| DATABASE MANAGEMENT SYSTEM. |
| DATABASE. |
| DATA LINKING. |
| INTEGRATION. |

# User Management Key Features.

1. User Accounts:

* Create
* Delete
* Update

1. User Data:

* Read/ view
* Edit
* Delete

1. Permissions and Assigned Roles:

* System Administrators: Can perform all three actions on accounts as well as on three functions on user data.
* End Users:Can only perform specified actions on the accounts and user data as specified by the system Administrator.

# Data.

User data which includes both employee and customer data. The user data information which should be captured and stored includes name, username, contact details; email and phone.

The system thus ensures maintenance of data confidentiality, security and integrity in a way that only the authorized personnel can access and modify the data.

System data which includes user accounts, assigned roles, permissions and login history.

# Search and Sorting Functionality.

User data will be searched and filtered to find specific users by various criteria based either user credentials like username and email address.

Upon login users will be given a unique Identification number that will uniquely identify them from every other user like the employee ID and user ID which helps in narrowing down to produce effective and direct search results.

The unique IDs will also help in sorting user accounts in the system effectively.

# Data Entry and Update.

The system is set to be user friendly so as to allow efficient and smooth entry and update of user data for both system administrators and customers whenever it's deemed necessary.

Validation checks are in place to ensure correct data formats are used by the users including valid email addresses and unique usernames.

# 

# Data Validation and Integrity.

System implements the validation of users as well as their inputs so as to prevent malicious data entry and logins into the system.

Data integrity, confidentiality and security rules are enforced by the system to maintain consistency and accuracy of the data.

Unique user IDs being in place does prevent duplicate entries.

Potential errors and exceptions are handled promptly providing feedback to the system administrators and the customers as well.

# Security and Access Control.

The system implements permissions and role-based access controls to authenticate and authorize users.

Data encryption and secure storage for user data and sensitive user information such as user passwords.

Regular data backups to prevent data loss and facilitate data recovery incase of system failure occurrence.

# Integration.

External systems such as Customer Relations Management system for customer relations, two- factor authentication systems for authentication processes and biometric access control system for login verifications.

User friendly Application Programming Interface to enable and ensure seamless and efficient integration and data sharing between systems.

# Reporting.

Report generation is done to provide analytical insights to system administrators so that the can be able to make informed decision making based on user data and system activities.

Report customization with various search and sorting criteria for efficient reporting.

Sharing and exporting of reports in different formats is enabled in the system.

# System Customization, optimization and Scalability.

The system is set to be customizable and scalable to accommodate future growth having the ability to handle an increase in the number of users as well as increased amounts of data.

System is also customized and optimized to be able to adapt to specific business processes and requirements as per an organization's policy and rules.

# 

# Technologies Used.

1. HTML:

* HTML was used to define the overall structure of the front end structure .

HTML forms were implemented in the system for:

* User authentication to allow the system users to login and access the system.
* Capturing user information including username, name, email address, user IDs, passwords and access levels.
* HTML tables and lists to display a list of users, their details and various access levels and display database records allowing administrators to view, edit and manage data.
* HTML forms are also used in capturing user input and passing it to the server for database queries and database updates.
* HTML input fields to create user login forms.
* HTML elements were also implemented to structure and organize the contents of the web pages in the system.

1. CSS: CSS was applied in the system for:

* To enhance visual appearance on the front end web page.
* To define styles including colors, fonts, layouts, tables, input fields, buttons, other user interface components and other visual properties in the system.
* To ensure a user-friendly design on the system.
* To style the HTML forms and provide visual feedback to the user for error validation and success in operations and activities.
* CSS styles for highlighting selected users, rows and columns on tables for improved readability, visual hierarchy and usability in the database and user interface.
* To visually indicate the login status in HTML login forms.
* To differentiate between different assigned user roles and permissions highlight current system logins.
* To format and position edited forms for a seamless editing experience.

1. MonoDB:

In this project MongoDB was the database management system that was used. It is a NoSQL database used to handle large volumes of data efficiently by providing scalability and flexibility where the system administrators have control over the user data with its JSON-like document oriented approach.

MongoDB database is used as storage for user data, assigned roles and permissions and login histories and used to compare user credentials during login and authentication processes.

MongoDB has a language query that is applied for:

* Allows the system administrator to perform the Create, Read, Update and Delete (CRUD) operations on the user data.
* To fetch and collect user data through a specific criteria same as the criteria used in the search and sorting functionality.

MongoDB update queries and MongoDBs delete operations features were used in modification/ updation and deletion of user data in the database respectively.

MongoDB also has a validation feature that is used to enforce validation regulations over at the database level.

MongoDB replication feature was also incorporated for better data availability and stability since the data stored in its database is stored in multiple servers for disaster recovery and backup hence vulnerabilities such as system failures, system crash, hardware failure and service interruptions in the system.

MongoDB has a sharding feature where large datasets are split into multiple distributed collections for better data execution by the system administrators in the system.

1. Javascript and NodeJs:

In this system Javascript and NodeJs were combined to create a system that is robust and scalable for user data linking.Javascript was used in both frontend and backend where on the frontend javascript framework was used to create interactive user interfaces for the web application. Javascript was also used on both frontend and backend to facilitate seamless communication and flow of data from the API to the Database. NodeJs non-blocking input output model feature was used for efficient handling of concurrent requests thus high performance of the web application. NodeJs was used in the implementation of the system where it was used to provide a scalable and efficient platform for building the web application used in the system.

Within NodeJs a package.json file was implemented to manage dependencies including storing descriptive information about applications, modules and packages; versions and names.

NodeJs was also implemented in the system to handle requests and responses whereby nodeJs has a feature that allows it to place requests into queues and handle responses.

Conclusion.

In conclusion a user management system plays an essential role in ensuring efficient user management and data access control in an organization. By granting the system administrator authorization to fully access to the database and having the privilege to assign roles to users it empowers the System administrator with necessary tools to maintain data integrity, confidentiality, security, consistency of the database and user data hence streamlining the administrative tasks in an organization for it to run seamlessly as it enhances operational efficiency.

References:

1. W3schools: https://www.w3schools.com/
2. Geek for geeks: https://www.geeksforgeeks.org/
3. Python.org: https://www.python.org/
4. Javascript.info :<https://javascript.info/>
5. MongoDB and node.js: <https://docs.mongodb.com/drivers/node/current/>
6. "HTML and CSS: Design and Build Websites" book by Jon Duckett
7. "Design and Implementation of User Management System for Web Applications" Research paper by Atul Kumar and Ajay Singh Yadav.