**CMP 403 Software Engineering: GROUP OPEN TEST**

MEMBERS:

1. EJIGA ALEXANDER ACHILE. BHU/20/04/05/0046
2. OBENTA CHRISTOPHER OWOICHO. BHU/20/04/05/0090
3. IKEMKA ROMASON ROMANUS. BHU/20/04/05/0016
4. OGUH CHUBIYOJO FAVOUR. BHU/20/04/05/0048
5. KPONGO BEMSEN CHUKWUMA. BHU/20/04/05/0115
6. ADAMANI MIVANYI ENAN. BHU/20/04/05/0044
7. YOROM JOSHUA RIMAN BHU/20/04/05/0014
8. IDAMINABO CLEMENT-DAVIS BHU/20/04/05/0054
9. ADEJORI OLUWAFERANMI BHU/20/04/05/0080

QUESTION 1

Compare and contrast between plan-driven/Agile methods.

QUESTION 2

Write the requirements for the four (4) software projects.

ANSWERS

Q1:

|  |  |
| --- | --- |
| Differences Between Plan-driven and Agile Development Methods | |
| Plan-driven | Agile |
| Extensive initial planning with focus on documentation and developing predetermined requirements. | Minimal initial planning with a focus on iterations of adaptations of new requirement |
| Follows a sequential, stage-based approach to development. | Follows iterative and incremental development cycles. |
| Clients are often involved primarily at the beginning and end of the project. | Clients are actively involved throughout the development process. |

Similarities Between Plan-driven and Agile Development Methods:

1. Both methods require project management and tracking of progress.
2. Ultimately, both aim to satisfy customer needs and expectations.
3. Collaboration among team members is essential for success in both methods.
4. Effective communication with stakeholders is crucial in both approaches.

Q2

1. **Employee Management System (EMS)**

Requirements:

**Functional Requirements:**

1. User Authentication and Authorization:

Administrators and employees should have secure login access with unique usernames and passwords.

Administrators should have higher privileges compared to employees.

1. Admin Functions:

Admins should be able to add, edit, and delete employee records.

Admins should manage leave requests, approve or reject leaves, and maintain leave records.

Admins should be able to update employee qualification details.

Admins should have access to promotion history records.

1. Employee Functions:

Employees should be able to check their leave status, including available leave balances and pending requests.

Employees should view and update their qualification details.

Employees should access a yearly holiday list to plan their vacations.

Employees should be able to submit grievances or resignations through the system.

1. Notifications:

The system should send notifications to admins when new leave requests are submitted.

Employees should receive notifications when leave requests are approved or rejected.

1. Reporting:

The system should generate reports on employee information, leave records, and promotions history for administrators.

**Non-Functional Requirements:**

1. Security:

The system should ensure data security and access control, protecting sensitive employee information.

Passwords should be securely stored, and data transmission should be encrypted.

1. Performance:

The system should be responsive and provide quick access to data.

It should handle concurrent users without performance degradation.

1. Scalability:

The system should be scalable to accommodate a growing number of employees and administrators.

1. Data Backup and Recovery:

Regular data backups and a recovery mechanism should be in place to prevent data loss.

Audit Trail:

The system should maintain an audit trail of important actions taken, including who performed them and when.

1. Data Privacy:

The system should comply with data privacy regulations and protect sensitive employee information.

1. **Payroll Management System**

Requirements:

**Functional Requirements:**

1. User Authentication and Authorization:

Users (employees and administrators) should have secure login access with unique IDs and passwords.

Admins should have higher privileges compared to employees.

1. Employee Registration and Profile Management:

Admins should be able to create and manage employee profiles, including relevant personal and work-related information.

1. Payroll Management:

The system should calculate and manage employee salaries, including deductions, bonuses, and taxes.

HR should have the capability to adjust payroll based on accurate working days and other factors.

1. Location Tracking:

The app should capture user images and GPS locations.

Employee locations should be constantly updated to the admin every 5 minutes while they are logged in and upon log out.

1. HR Functions:

HR personnel should be able to keep accurate records of working days for each employee.

HR should have the ability to make adjustments to payroll based on attendance and other relevant factors.

1. Notifications:

The system should notify administrators when employees log in and log out.

The admin should receive regular updates on employee locations.

1. Data Storage and Security:

Employee data, including personal and financial information, should be securely stored and protected.

Passwords should be securely stored, and data transmission should be encrypted.

**Non-Functional Requirements:**

1. Security:

The system should ensure data security and access control to protect sensitive employee information.

Employee images and GPS locations should be stored securely.

1. Scalability:

The system should be scalable to accommodate a growing number of employees and administrators.

1. Compliance To Local Laws:

The system should comply with relevant labor laws and regulations, especially regarding employee location tracking.

1. Audit Logging:

The system should maintain an audit trail of important actions taken, including who performed them and when.

1. Data Privacy:

The system should comply with data privacy regulations and protect sensitive employee information.

1. **Fingerprint-Based ATM Card**

**Functional Requirements:**

1. A Database/record of all transaction with a query for the last five transactions specific to each or the logged in User
2. **Security:** Heavy/Extremely complex encryption to protect user pins as fingerprint technology is not completely secure
3. **Proof of Transaction:** there should be a system for receipts either printed or digital containing details about the current transactions
4. **History:** A Detailed History of transactions with dates, location and other metadata
5. **Fingerprint Authentication:** capture fingerprints at a vey high level of detail and verify user fingerprints accurately

**Non-Functional Requirements**

1. **Localization**:Should be available in multiple languages **and should have options for other currencies**
2. **Auditability:** The system should have clear and concise records for precise cleaning and collection/traversing of transaction data in the system
3. **Availability**: the system should be available around the clock as an important component of a very fast life
4. **Performance:** Extremely fast response times for even greater or larger user bases
5. **Multi-platform Compatibility:** the system should be able to run seamlessly on a multitude for a variety of devices .
6. **Android Local Ticketing System**

**Functional Requirements:**

1. User Authentication and Authorization:

Users and administrators should have secure login access with unique credentials.

Admins should have privileges to manage user accounts and view processed tickets.

1. User Account Management:

Users should be able to create, update, and delete their accounts.

Admins should have the capability to recharge user accounts.

1. Ticket Booking:

Users should be able to fill out a ticket booking form, including details like destination, travel class, and journey type (single route or return journey).

Users should receive booking receipts that can be viewed and printed.

1. Admin Ticket Management:

Admins should be able to view and manage processed tickets, including refunding and canceling tickets as necessary.

1. Journey Options:

The app should provide options for single route journeys and return journeys.

1. Travel Class Options:

Users should be able to choose from different travel classes (e.g., economy, business) when booking tickets.

1. Notifications:

Users should receive booking confirmations, ticket details, and notifications for changes or cancellations.

Admins should receive notifications about successful recharges and user activities.

1. Payment Integration:

The system should integrate with payment gateways to facilitate secure payments for ticket bookings.

**Non-Functional Requirements:**

1. Security:

The system should ensure data security and access control, protecting sensitive user information and payment details.

Passwords and payment information should be securely stored, and data transmission should be encrypted.

1. Scalability:

The system should be scalable to accommodate a growing number of users and administrators.

1. Reliability:

The system should be available and reliable, with minimal downtime for maintenance or updates.

1. Data Backup and Recovery:

Regular data backups and a recovery mechanism should be in place to prevent data loss.

1. Compliance:

The system should comply with relevant local transportation regulations and security standards for online payments.

1. Audit Logging:

The system should maintain an audit trail of important actions taken, including who performed them and when.

1. Data Privacy:

The system should comply with data privacy regulations and protect sensitive user information.