**Awais Farooq(04072212008):**

3. Specific requirements

1. 3.1  External interface requirements

3.1.1 User interfaces  
3.1.2 Hardware interfaces  
3.1.3 Software interfaces  
3.1.4 Communications interfaces

1. 3.2  Functional requirements

3.2.1 Mode 1

3.2.1.1 Functional requirement 1.1 .  
.  
.

3.2.1.*n* Functional requirement 1.*n*

3.2.2 Mode 2

.  
.  
.  
3.2.*m* Mode *m*

3.2.*m.*1 Functional requirement *m.*1 .  
.  
.

3.2.*m.n* Functional requirement *m.n*

1. 3.3  Performance requirements
2. 3.4  Design constraints
3. 3.5  Software system attributes
4. 3.6  Other requirements

**3. Specific Requirements**

**3.1. External Interfaces**

This section provides a detailed description of all inputs into and outputs from the Course Evaluation System. Each interface is defined with the following attributes:

**3.1.1. User Interface(UI):**

* **Name of Item:** User Interface(UI)
* **Purpose:** To facilitate user input , display output and interaction betweenuser and the system
* **Source of input and destination of output:** The UI serves as both
* **Valid range:** As specified by the stystem
* **Units of Measures:** N/A
* **Timing:** Responds to user inputs in a timely manner
* **Releationship:** The UI interacts with various inputs and outputs
* **Data formats:** text files, tables, charts, graphs, images, videos
* **Command Format:** button clicks, keyboard shortcuts
* **End Message:** must display end messages to inform users

**3.1.2. User Registration and Authentication:**

* **Name of Item:** User Registration and Authentication
* **Purpose:** To allow users to create more acounts and verify them
* **Source of input and destination of output:** Input: User provide information. Output: Confirmation message
* **Valid range:** must follow username strength and password rules
* **Units of Measures:** N/A
* **Timing:** The process should be efficient
* **Releationship:** Inputs:User inputs are used to create user profiles and other things. Outputs: Confirmation message
* **Data formats:** Should follow specified format
* **Command Format:** Mostly clicking(e.g: create user acount)
* **End Message:** Receive confirmation message(Registration Successful)

**3.1.3. Content Delivery Network(CDN):**

* **Name of Item:** Content Delivery Network(CDN)
* **Purpose:** To optimizes the delivery of user-generated content, such as images, videos, and other content
* **Source of input and destination of output:** Input: Upload of users’s content. Output: Optimized content delivered to users' devices
* **Valid range:** Acceptable file formats and sizes
* **Units of Measures:** N/A
* **Timing:** CDN optimizes content delivery by minimizing latency and load times,
* **Releationship:** Inputs: User’s content upload to the system Outputs: Optimized content delivered to users' devices
* **Data formats:** images (e.g., JPEG, PNG), videos (e.g., MP4, AVI),
* **Command Format:** Delivery command are mostly automated
* **End Message:** N/A

**3.1.4. Social Media Integration:**

* **Name of Item:** Social Media Integeration
* **Purpose:** Integration with social media platforms for sharing news
* **Source of input and destination of output:** Input: Shared news articles, user authentication data. Output: Posted news articles on social media platforms.
* **Valid range:** Compliance with social media API usage policies and data privacy regulations.
* **Units of Measures:** N/A
* **Timing:** Real-time interaction during news sharing and user authentication.
* **Releationship**: Enhances platform reach and user engagement through social media channels.
* **Data formats:** JSON, social media API-specific formats (e.g., Graph API for Facebook).
* **Command Format:** HTTP requests, API calls for posting content and authentication.
* **End Message:** N/A

**3.1.5. Analytics and Monitoring Tools Integration:**

* **Name of Item:** Analytics and Monitoring Tools Integration:
* **Purpose:** Integration with analytics and monitoring services for tracking user behavior.
* **Source of input and destination of output:** Input: User Interaction data. Output: Analytical reports,
* **Valid range:** Real Time Monitoring
* **Units of Measures:** N/A
* **Timing**: Continuous monitoring and periodic reporting.
* **Releationship**: Identifies issues and and enhance user experience
* **Data formats:** JSON, analytics data formats
* **Command Format:** API requests,
* **End Message:** N/A

**3.1.6. Payment Gateway Integeration:**

* **Name of Item:** Payment Gateway Integeration
* **Purpose:** To process payments and secure transactions
* **Source of input and destination of output:** Input: Payment requests. Output: Payment processing results.
* **Valid range:** Secure handling of payment data.
* **Units of Measures:** N/A
* **Timing:** Real-time payment processing.
* **Releationship**: Enable other features acess
* **Data formats:** Payment transaction formats.
* **Command Format:** API requests for payment processing
* **End Message:** N/A

**3.1.7. Email and Notification Services Integration:**

* **Name of Item:** Email and Notification Services Integration
* **Purpose:** To transfer important emails to users.
* **Source of input and destination of output:** Input: Email Content. Output: Sent Emails
* **Valid range:** Ensuring email deliverability.
* **Units of Measures:** N/A
* **Timing:** Real-time email delivery.
* **Releationship**: Enhance user communication and experience.
* **Data formats:** HTML, Email Formats.
* **Command Format:** API calls for email delivery.
* **End Message:** N/A

**3.2. Functional Requirements**

Functional requirements should define the fundamental actions that must take place in the software in accepting and processing the inputs and in processing and generating the outputs. These are generally listed as “shall” statements starting with “The system shall...”

a) **Validity checks on the inputs:**

* The system shall validate user inputs for user registration, including username, email address, password, and profile information, to ensure they meet specified criteria (e.g., valid email format, password strength requirements).
* Validity checks shall include checking for uniqueness of usernames and email addresses to prevent duplicate registrations.

**b)Exact sequence of operation**

* The system shall follow a predefined sequence of operations for user registration, which includes:

1. User provides registration information (username, email, password, profile details).
2. The system validates input data.
3. If inputs are valid, the system creates a new user account and stores registration data.
4. Confirmation message is displayed to the user upon successful registration.

**c)Respnse to abnormal situations**

1. **Overflow:**

The system handle the situation when user enter the input data outside the limit.

1. **Communication facilities:**
   * The system shall send email services for registration purposes and other tasks during communication.
   * The system shall notify for delayed notification in case unavailability of email services.
2. **Error Handling and Recovery:**

* The system shall recover error handling such as database issues, servor errors and registration issues and display error message.

**d) Effects of parameters**

* The system shall consider configurable parameters such as password complexity rules, email verification settings, and account activation policies.
* Changes to parameters (e.g., updating password strength requirements) shall be reflected immediately and enforced for new registrations and password updates.

**e)Relationship of outputs to inputs**

**1. Input/Output sequences:**

* The system shall generate messages in responds of users’s inputs and during registration of the users.
* When registration is done, the system shall redirect users to the login page.

**2. Formulas for input to output conversion:**

* To create the tokens for email verification, the system shall use cryptographic algorithms.
* The conversion of user’s data to database it should follow predefined data models.

**3.3. Performance Requirement**

1. **Static Numerical Requirements:**

a) The system shall support a 1000 terminals at a same time.

b) The system shall support a minimum of 50,000 users interacting with the plateform.

c). The system shall handle a maximum of 500 news posts per minute, including text, images, and videos.

1. **Dynamic Numerical Requirements:**

a) The system shall process a minimum of 200 user registrations per minute during normal workload conditions.

b) The system shall process a minimum of 1000 search queries per minute during peak workload conditions.

c) The system shall handle a maximum of 10,000 comments and interactions (likes, shares) per hour on posted news articles.

* 1. **Logical Database Requirements**
  2. **Types of information used by the various functions:**
* User Registration Function: (username, email, password etc.)
* News Posting Function: (News title, content, category,tags, author ID,images,videos etc.)
* Commenting Function: (Comment content, user ID etc.)
  1. **Frequency of use:**

User profile informatiom and new articles and comments sections frequently used.

* 1. **Accesibility capabilities:**
* Administrator have full access to every kind of data.
* Ordinary users have limited access related to their acount.
  1. **Data entities and their relationships:**
* User entity with attributes like user ID (primary key), username, email (unique), password (hashed), and profile details.
* News entity with attributes like news ID (primary key), title, content, category, tags, timestamp, author ID (foreign key), and multimedia paths.
* Comment entity with attributes like comment ID (primary key), user ID (foreign key), news ID (foreign key), content, timestamp.

**e)Integrity Constraints:**

* Use User ID as the primary key.
* Foreign key constraint is also used.
  1. **Non-Functional Requirements**
  2. **Performance:**
     + The system shall have less than 2 second average responses time.
     + The shall support a over 10,000 users at a time without degradation of performance.
     + The system shall hold more than 1,000 posts at the peak time
  3. **Security:**
     + Users password shall be stored.
     + The system shall ensures that only authorized users can perform administrative functions.
  4. **Reliability:**
     + The system shall ensure continuous availability of users.
     + To prevent data loss, automated backup is initated after every 24 hours.
  5. **Usability:**
     + The user interface shall be user-friendly.
     + The system shall use tooltip and guidance interface for the understanding of the user.
  6. **Compatibility:**
     + The system shall be compatible with the major web browsers and (Android,iOS).
     + The system shall support multiple languages.