Tutorium – The Perfect Tutor, A Click Away

Second Deliverable



**GROUP MEMBERS**

**Hashir Ahmed** (BCSF20M009)Team Lead

**Muhammad Huzaifa Khawar** (BCSF20M023)

**Muhammad Shazil** (BCSF20M030)

**Asim Ali** (BCSF20M036)

**Muhammad Qadeer** (BCSF20M037)

# SUPERVISOR

## Mam MAdiha

## Department of Information Technology

## Faculty of Computing & Information Technology Friday, 10 November 2023

**TABLE OF CONTENTS**

[SUPERVISOR 1](#_bookmark0)

1. [Introduction 3](#_bookmark1)
2. [Use Case Modeling 3](#_bookmark2)
3. [Design Mapping 16](#_bookmark3)
4. [Design Class Diagram 18](#_bookmark4)
5. [Data Model 20](#_bookmark5)
6. **Introduction**

Our Platform is a three-tier architecture web-based system designed to revolutionize how organizations hire. By automating CV screening, generating tailored tests using a Large Language Model (LLM), and prioritizing data security, AI-Recruiter streamlines the recruitment journey. It offers a solution that enhances efficiency, reduces bias, and enables recruiters to make more informed decisions, ultimately redefining how businesses identify and select top-tier candidates in an increasingly competitive job market.

1. **Use Case Modeling**

**Use Case:** UC\_Register\_Google\_Account

**Primary Actor:** User

**Brief Description**: The user registers within the system using Google Accounts.

**Pre-conditions:** The user has an existing google account and is connected to the internet.

**Basic Flow:**

User selects the option to register using Google Accounts.

System redirects the user to Google's authentication page.

User logs in with Google credentials.

System retrieves necessary user information from Google.

User account is created within the system.

**Alternate Flow:**

If there are authentication issues or connection problems, the system provides an error message and prompts the user to try again.

**Post-conditions:** User's account is successfully registered within the system.

### Use Case: UC\_Login\_Recruiter

* **Primary Actor:** Recruiter
* **Brief Description:** The recruiter logs into the application using google authentication.
* **Pre-conditions:** The recruiter has an existing google account in the system.
* **Basic Flow:**
  1. Recruiter navigates to the login page of the application.
  2. Recruiter enters their registered email and password.
  3. System verifies the credentials.
  4. Upon successful verification, the system grants access to the recruiter's dashboard.
* **Alternate Flow:**
  1. If the credentials entered are incorrect, the system prompts the recruiter to re-enter the correct information.
* **Post-conditions:** Recruiter gains access to the application's features and functionalities.

### Use Case: UC\_Purchase\_with\_Stripe

* **Primary Actor:** User
* **Brief Description:** The user initiates a purchase transaction using the Stripe payment service.
* **Pre-conditions:**
  + User is logged into the system.
  + The system has integrated the Stripe payment gateway.
* **Basic Flow:**
  + User navigates to the purchase section within the system.
  + User selects the desired software or package for purchase.
  + System prompts the user to enter payment details.
  + User inputs necessary payment information (e.g., credit card details).
  + System communicates with Stripe's API to process the payment securely.
  + Stripe validates and authorizes the payment.
  + System confirms successful payment and updates user access privileges to the purchased software.
* **Alternate Flow:**
  + If payment authorization fails, the system prompts the user to recheck payment details or try another payment method.
* **Post-conditions:**
  + User gains access to the purchased software or service within the system.
  + Payment transaction details are stored securely.

### Use Case: UC\_Manage\_Profile

* **Primary Actor:** User
* **Brief Description:** The user can manage and update their profile information within the system.
* **Pre-conditions:** User is logged into the system.
* **Basic Flow:**
  1. User navigates to the profile management section.
  2. User selects the option to edit profile details.
  3. System displays the current user information.
  4. User modifies relevant profile fields (e.g., contact information, skills).
  5. User saves the updated information.
* **Alternate Flow:**
  1. If there's an issue in saving information, the system provides an error message and prompts the user to reattempt the save action.
* **Post-conditions:** User's profile information is updated and saved in the system.

### Use Case: UC\_User\_Interface

* **Primary Actor:** User
* **Brief Description:** The system provides an intuitive and user-friendly interface for efficient navigation.
* **Pre-conditions:** User has access to the system.
* **Basic Flow:**
  1. User logs into the system.
  2. System displays a clear and navigable interface with accessible functionalities.
  3. User interacts with various features seamlessly.
* **Alternate Flow:**
  1. If the user encounters interface issues or navigational difficulties, there might be a need for user guidance or system improvements.
* **Post-conditions:** User successfully interacts with the system through the provided interface.

### Use Case: UC\_CV\_Upload\_Interface

* **Primary Actor:** User
* **Brief Description:** The system offers a straightforward and user-friendly interface for uploading CVs.
* **Pre-conditions:** User is logged into the system and possesses multiple CVs to upload.
* **Basic Flow:**
  1. User navigates to the CV upload section.
  2. System prompts the user to choose the CV file.
  3. User selects the desired CV file from their device.
  4. System initiates the CV upload process.
* **Alternate Flow:**
  1. If the selected file format is unsupported, the system notifies the user and requests a compatible file format.
* **Post-conditions:** User successfully uploads their CV into the system.

### Use Case: UC\_Upload\_Candidate\_CVs

* **Primary Actor:** User
* **Brief Description:** The system allows users to easily upload candidate CVs.
* **Pre-conditions:** User is logged into the system and has access to candidate CVs.
* **Basic Flow:**
  1. User accesses the CV upload section.
  2. User selects the option to upload candidate CVs.
  3. User selects and uploads multiple candidate CVs or enter the drive link where all CVs are present.
* **Alternate Flow:**
  1. If there are issues with uploading multiple files simultaneously, the system might handle files sequentially or provide instructions for successful bulk upload.
* **Post-conditions:** Multiple candidate CVs are successfully uploaded into the system.

### Use Case: UC\_Support\_PDF\_Formats

* **Primary Actor:** System
* **Brief Description:** The system supports various PDF file formats for CV uploads.
* **Pre-conditions:** System is operational and accessible to users.
* **Basic Flow:**
  1. User attempts to upload a CV in PDF format.
  2. System detects the file format as PDF.
  3. System verifies compatibility and accepts the PDF file for upload.
* **Alternate Flow:**
  1. If the PDF file is corrupted or encrypted or any other file,than the system notifies the user of the issue and requests an uncorrupted, accessible PDF file.
* **Post-conditions:** System successfully processes and accepts PDF format CVs for upload.

### Use Case: UC\_Extract\_Skills

* **Primary Actor:** System
* **Brief Description:** The system automatically extracts skills from candidate CVs.
* **Pre-conditions:** Candidate CV in a readable format (PDF) is uploaded into the system.
* **Basic Flow:**
  1. System analyzes the uploaded CV for skill-related keywords and phrases.
  2. Extracts and compiles a list of skills mentioned in the CV.
* **Alternate Flow:**
  1. If the CV format is not supported or unreadable, the system prompts the user to upload a different format or a more accessible file.
* **Post-conditions:** System generates a list of extracted skills from the candidate's CV.

### Use Case: UC\_Extract\_Email

* **Primary Actor:** System
* **Brief Description:** The system extracts the user's email from a candidate's CV.
* **Pre-conditions:** Candidate CV in a readable format (PDF) is uploaded into the system.
* **Basic Flow:**
  1. System scans the CV for extracting email patterns or identifiers of the candidate.
* **Alternate Flow:**
  1. If the email format is not identifiable or missing, the system notifies the user
* **Post-conditions:** System successfully extracts and confirms the candidate's email address.

### Use Case: UC\_Generate\_Custom\_Tests

* **Primary Actor:** User (Recruiter)
* **Brief Description:** The user generates customized tests for candidates based on their skills or designated roles.
* **Pre-conditions:** User is logged into the system and has access to candidate profiles or have a position for a designated role.
* **Basic Flow:**
  1. User selects the option to create a custom test.
  2. User specifies criteria based on candidate skills or roles for test generation.
  3. System compiles test questions relevant to the specified criteria.
* **Alternate Flow:**
  1. If the criteria selection process encounters errors or lacks specificity, the system prompts the user to refine the parameters.
* **Post-conditions:** User creates a customized test for candidates based on specified criteria.

### Use Case: UC\_Generate\_Test

* **Primary Actor:** System
* **Brief Description:** The system generates standardized tests for candidates.
* **Pre-conditions:** System contains the data of the skillset for every employee
* **Basic Flow:**
  1. System accesses a database to get candidate skills or pre-set parameters for test generation.
  2. Constructs a standardized test based on the information.
* **Alternate Flow:**
  1. If there are issues retrieving test parameters or constructing the test, the system provides an error message or defaults to a general test format.
* **Post-conditions:** System generates a standardized test for candidate assessment.

### Use Case: UC\_Customize\_Test

* **Primary Actor:** User (Recruiter)
* **Brief Description:** The user customizes test questions and format.
* **Pre-conditions:** The test is generated by the system and now user is accessing the test customization interface.
* **Basic Flow:**
  1. User selects the option to customize a test.
  2. User modifies test questions, sections, or formats according to preferences.
* **Alternate Flow:**
  1. If there are issues with modifying test elements or formatting, the system displays an error message or reverts to the original settings.
* **Post-conditions:** User successfully customizes the test according to their preferences.

### Use Case: UC\_Test\_Editor

* **Primary Actor:** System
* **Brief Description:** The system provides an intuitive test editor interface for test creation and customization.
* **Pre-conditions:** The user has access to the test creation functionality within the system.
* **Basic Flow:**
  1. User accesses the test editor feature.
  2. System presents a user-friendly interface to create, modify, or format tests.
  3. User customizes test questions and formats as required.
* **Alternate Flow:**
  1. If there are technical issues or unexpected errors, the system displays an error message and prompts the user to retry or contact support.
* **Post-conditions:** Test is successfully created or modified within the system.

### Use Case: UC\_Send\_Secure\_Test\_Links

* **Primary Actor:** System
* **Brief Description:** The system securely generates and sends test links to candidates' email addresses.
* **Pre-conditions:** Test creation is completed within the system and email is extracted through the CVs
* **Basic Flow:**
  1. System generates secure test links based on extracted emails through CVs
  2. System sends unique test links to candidates' provided email addresses.
* **Alternate Flow:**
  1. If there are issues with email delivery or security concerns, the system alerts administrators and logs the error.
* **Post-conditions:** Candidates receive secure test links in their email for assessment.

### Use Case: UC\_Take\_Test

* **Primary Actor:** Candidate
* **Brief Description:** Candidates take tests based on their extracted skills and qualifications or as provided by the User.
* **Pre-conditions:** Candidates have received the test link and are ready to undertake the assessment.
* **Basic Flow:**
  1. Candidate opens the test link received via email.
  2. System presents the test interface with instructions and questions.
  3. Candidate completes the test within the allotted time.
* **Alternate Flow:**
  1. If there are interruptions or technical issues during the test, the system allows candidates to contacts support.
* **Post-conditions:** Test responses are submitted to the system for evaluation.

### Use Case: UC\_Secure\_Test\_Links

* **Primary Actor:** System
* **Brief Description:** The system ensures the security of test links to prevent candidates from accessing external resources during tests.
* **Pre-conditions:** Test links have been generated and sent to candidates.
* **Basic Flow:**
  1. System generates secure test links preventing candidates from switching tabs or accessing external resources.
  2. Candidates open the test link in a secure browser window provided by the system.
* **Alternate Flow:**
  1. If candidates attempt to bypass security measures, the system terminates the test session and marks candidate’s score to 0.
* **Post-conditions:** Candidates take tests within a secure environment provided by the system.

### Use Case: UC\_Auto\_Evaluate\_Tests

* **Primary Actor:** System
* **Brief Description:** The system automatically evaluates test responses and provides objective results.
* **Pre-conditions:** Candidates have completed and submitted their test responses.
* **Basic Flow:**
  1. System processes and analyzes the submitted test responses.
  2. System Matches the given answers with the correct answers provided during the test generation through LLM model.
  3. System generates objective results based on the evaluation.
* **Alternate Flow:**
  1. If there are errors in the evaluation process, the system flags the issue for manual review or correction.
* **Post-conditions:** Objective test results are available for review by authorized users.

### Use Case: UC\_Evaluation\_Integrity

* **Primary Actor:** System
* **Brief Description:** Ensuring the integrity and accuracy of the evaluation process within the system.
* **Pre-conditions:** Evaluation criteria and parameters are established by the recruiter and accessible within the system.
* **Basic Flow:**
  1. System receives test submissions or evaluations.
  2. Validates the authenticity and completeness of test submissions.
  3. Runs internal checks to prevent tampering or bias in the evaluation process.
  4. Conducts evaluations based on predetermined criteria.
* **Alternate Flow:**
  1. If inconsistencies or anomalies are detected during evaluation, the system flags and reports them for manual review.
* **Post-conditions:** Evaluation results are maintained with integrity, minimizing errors or manipulation.

### Use Case: UC\_Present\_Results

* **Primary Actor:** System
* **Brief Description:** Presenting candidate assessment or test results in an easily understandable format for users.
* **Pre-conditions:** Evaluation or assessment process has been completed for candidates.
* **Basic Flow:**
  1. System generates assessment results based on predefined criteria.
  2. Organizes and formats the results in a clear and concise manner.
  3. Presents the results through a user-friendly interface or dashboard.
* **Alternate Flow:**
  1. If there are any technical issues in presenting the results, the system provides error handling or prompts to retry.
* **Post-conditions:** Users can access and interpret candidate assessment results effectively.

### Use Case: UC\_CV\_Ratings

* **Primary Actor:** System
* **Brief Description:** Providing clear and understandable ratings for candidate CVs.
* **Pre-conditions:** CVs are uploaded and processed within the system.
* **Basic Flow:**
  1. System analyzes uploaded CVs for relevant criteria using Api.
  2. Assigns ratings or scores based on predefined evaluation metrics.
  3. Displays ratings alongside candidate profiles or CV summaries.
* **Alternate Flow:**
  1. If the system encounters difficulties in analyzing or rating a CV, it flags the issue for manual review or requests additional information.
* **Post-conditions:** Users can view and consider ratings while assessing candidate profiles.

### Use Case: UC\_Secure\_Candidate\_Data

* **Primary Actor:** System
* **Brief Description:** Ensuring the secure storage and handling of candidate data within the system.
* **Pre-conditions:** Candidate data is entered or uploaded into the system.
* **Basic Flow:**
  1. System encrypts and securely stores candidate information.
  2. Access to candidate data is limited to authorized personnel through authentication protocols.
* **Alternate Flow:**
  1. If there are breaches or unauthorized access attempts, the system activates security protocols and alerts administrators.
* **Post-conditions:** Candidate data remains confidential and protected within the system.

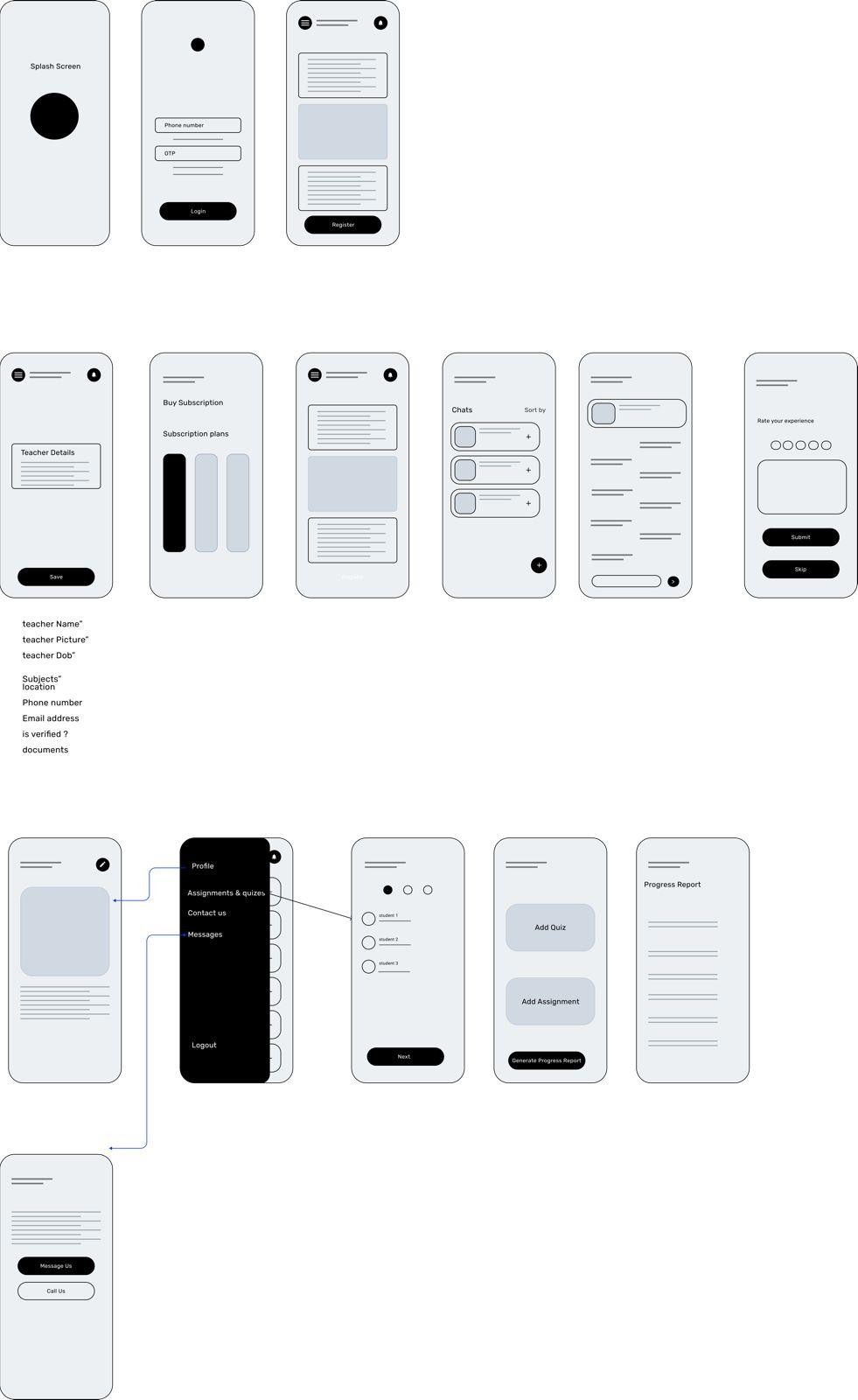
### Use Case: UC\_System\_Availability

* **Primary Actor:** System
* **Brief Description:** Ensuring continuous system accessibility for recruiters.
* **Pre-conditions:** System infrastructure is operational and connected to the internet.
* **Basic Flow:**
  1. System will be deployed on the server so it provides 24/7 service.
* **Alternate Flow:**
  1. If there are unexpected system downtimes, the system notifies users and initiates recovery procedures.
* **Post-conditions:** Recruiters can access the system at any time, facilitating uninterrupted operations.
  1. **Design Mapping**

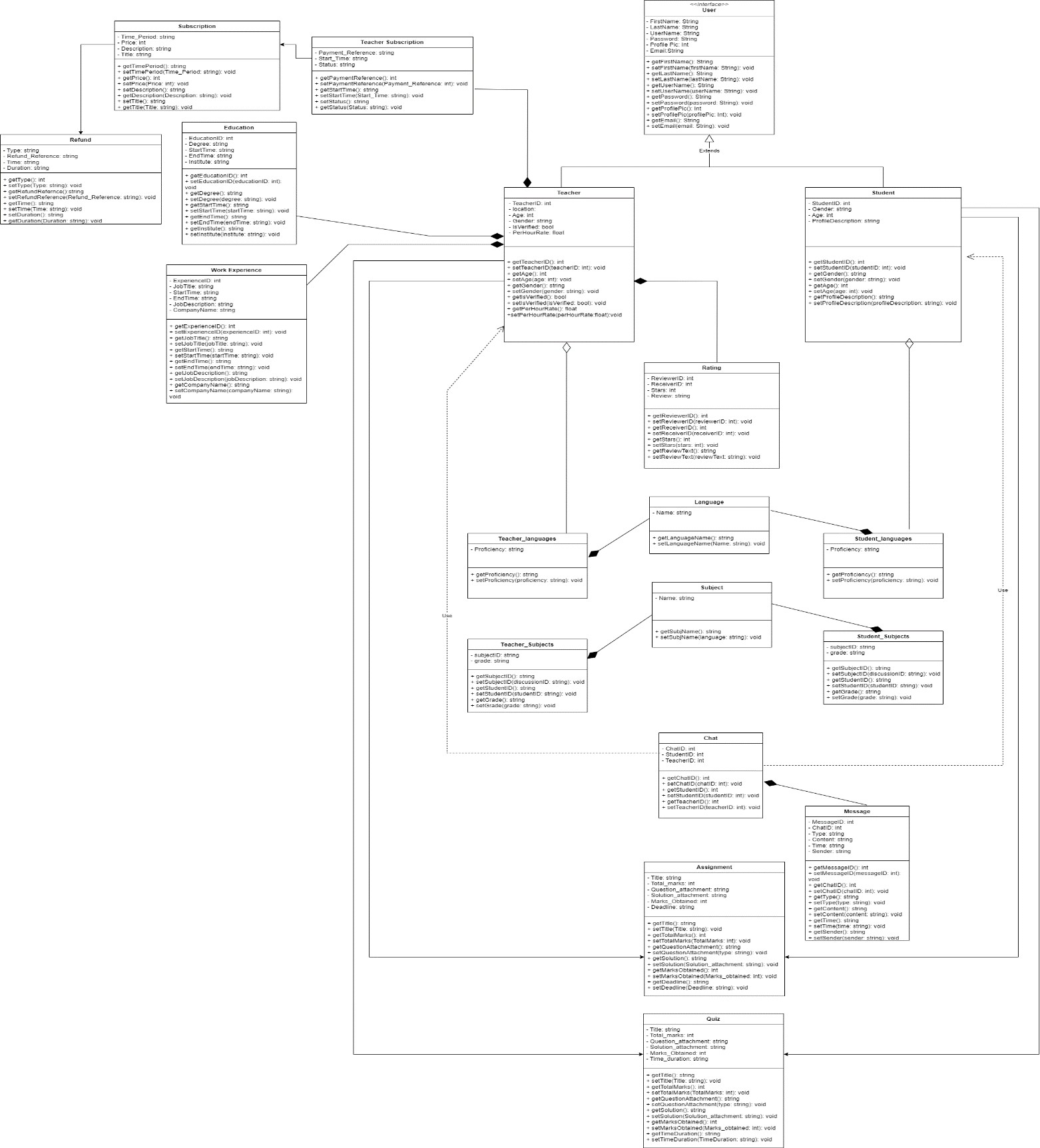
**STUDENT**



**TEACHER**



* 1. **Design Class Diagram**



* 1. **Data Model**

