

**AMERICAN INTERNATIONAL UNIVERSITY–BANGLADESH (AIUB)**

**FACULTY OF SCIENCE & TECHNOLOGY**

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

WEB TECHNOLOGIES

**FALL 2023-2024**

**Section: A**

**ASSIGNMENT ON**

***MVC FOR BOOK BORROWING SYSTEM***

**Supervised By**

**Nazmus Sakib Shan**

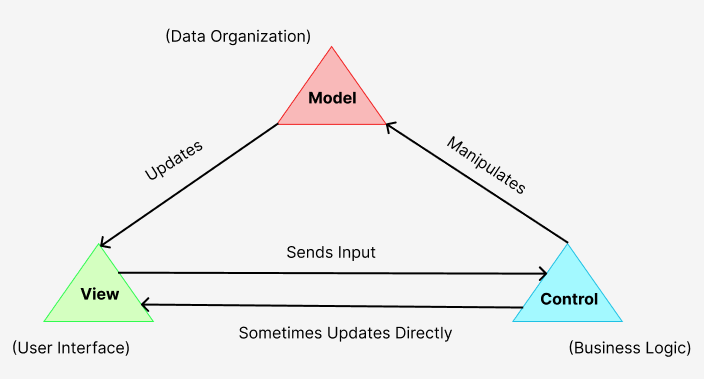
**Submitted By**

**FATEMA AKTER TRIPTY**

**ID:** 22-46989-1

**Date of Submission:** July 3, 2024

**MVC For Book Borrowing System:**

****

## Figure: MVC

## Models

### **Book.php**

* **Purpose:** Represents a book in the library system.
* **Attributes:**
  + **book\_id:** Unique identifier for the book.
  + **title:** Title of the book.
  + **author:** Author of the book.
  + **publisher:** Publisher of the book.
  + **year:** Year of publication.
* **Specification:**
  + Retrieves details of a specific book from the database.
  + Updates the availability status of a book in the database.

### **Student.php**

* **Purpose:** Represents a student who can borrow books.
* **Attributes:**
  + student\_id: Unique identifier for the student.
  + name: Name of the student.
  + phone: Phone number of the student.
  + email: Email address of the student.
  + dob: Date of birth of the student.
* **Specification:**
  + Retrieves details of a specific student from the database.
  + Registers a new student in the database.

### **BorrowRecord.php**

* **Purpose:** Represents a borrowing record of books.
* **Attributes:**
  + record\_id: Unique identifier for the borrowing record.
  + student\_id: Identifier of the student who borrowed the book.
  + book\_id: Identifier of the borrowed book.
  + borrow\_date: Date when the book was borrowed.
  + return\_date: Date when the book was returned.
* **Specification:**
  + Creates a new borrowing record in the database.
  + Retrieves borrowing records for a specific student.
  + Updates the return date for a borrowing record in the database.

## Views

### **index.html**

* **Purpose:** Introduction page requested by the librarian.
* **Contents:**
  + A form with fields for Student ID, Name, Phone Number, Email, and Date of Birth.
  + Links to other pages such as login, signup, borrow book, and return book.

### **login.html**

* **Purpose:** Login page for students and librarians.
* **Contents:**
  + A form with fields for Email and Password.

### **signup.html**

* **Purpose:** Signup page for new students.
* **Contents:**
  + A form with fields for Name, Phone Number, Email, Date of Birth, and Password.

### **borrow\_book.html**

* **Purpose:** Page for borrowing a book.
* **Contents:**
  + A form with fields for Student ID, Book ID, Borrow Date, and Return Date.

### **return\_book.html**

* **Purpose:** Page for returning a borrowed book.
* **Contents:**
  + A form with fields for Record ID and Return Date.

## Controllers

### **StudentController.php**

* **Purpose:** Handles student-related requests and updates.
* **Specification:**
  + Handles student registration, validates input. If registration is successful, it redirects to login.html.
  + Handles student login, validates input, and authenticates the student.

### **BookController.php**

* **Purpose:** Handles book-related requests and updates.
* **Specification:**
  + Retrieves details of a specific book and updates the view.
  + Updates the availability status of a book based on borrow/return actions.

### **BorrowRecordController.php**

* **Purpose:** Handles borrowing and returning of books.
* **Specification:**
  + Validates input, creates a borrowing record, and updates book availability.
  + Validates input, updates the return date in the borrowing record, and updates book availability.

### **DatabaseController.php**

* **Purpose:** Manages database operations.
* **Specification:**
  + Executes SQL queries with parameters and returns results.

## Interaction Flow

1. **Student Registration:**
   * User accesses signup.html.
   * User fills out the registration form and submits it.
   * StudentController.php handles the form submission, and validates input.
   * If registration is successful, the user is redirected to login.html.
2. **Student Login:**
   * User accesses login.html.
   * User fills out the login form and submits it.
   * StudentController.php handles the form submission, validates input, and authenticates the student.
   * If authentication is successful, the user is redirected to index.html.
3. **Borrowing a Book:**
   * User accesses borrow\_book.html.
   * User fills out the borrow book form and submits it.
   * BorrowRecordController.php handles the form submission, and validates input.
   * BookController.php updates the book's availability status.
4. **Returning a Book:**
   * User accesses return\_book.html.
   * User fills out the return book form and submits it.
   * BorrowRecordController.php handles the form submission, validates input
   * BookController.php updates the book's availability status.