SRM UNIVERSITY AP
Amaravati

OBE Implementation

Module-7:Courses

Team Name: SteveRogers

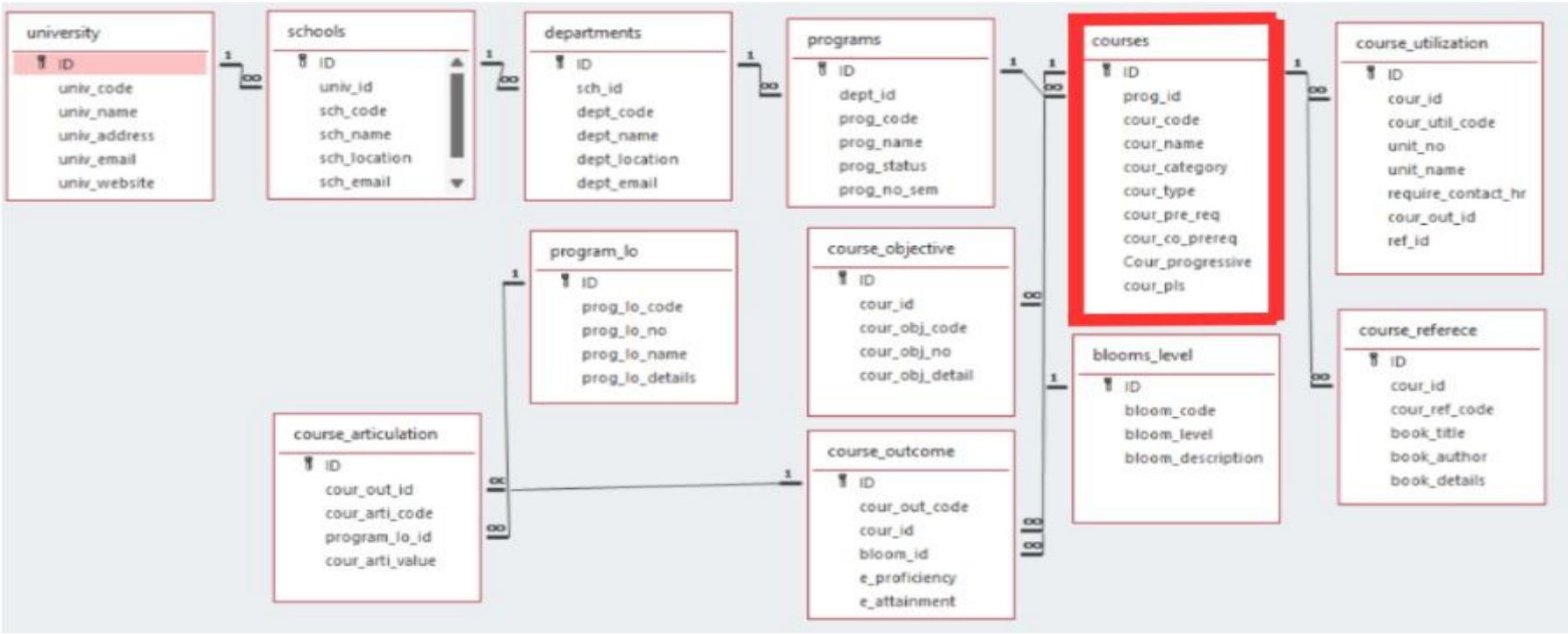
Submitted By

Vidhathri - AP22110010196
Darvini - AP22110010167
Sindhu - AP22110010180
Anjana - AP22110010181
Manasa - AP22110010606
CSE | 6th Semester | 2022

Introduction to Project

Our University (herewith considered as SRM-AP) is going to implement OBE(Outcome Based Education) in their university and you are assigned in the project to develop a CURD(Create,Update,Retrieve and Delete) windows and mobile application using JAVA programming and Android studio for the same.

Architecture Diagram



Module Description : Courses Setting

- This module is used to create,Update,Retrieve,Delete(hereafter known as CRUD) details of the module and storing the details in the mysql table.
- It provides a user-friendly interface for managing course information such as course code, name, category, prerequisites, and type.
- It ensures data accuracy and consistency by validating user input before performing database operations.
- All changes are reflected in real-time using a dynamic table display to ensure up-to-date data visibility.

Courses Setting:Field/table details

Field Name	Data type
id	Integer
programId	Integer
code	String
name	String
category	String
type	String
preReq	boolean or String
coReq	boolean or String
progressive	boolean or String
pls	boolean or String

Courses Setting:Programming Details

- **File name:** SteveRogers_courses

- **Function/method name**

  - **Create:** AP22110010167_courses_create
  - **Update:** AP22110010180_courses_update
  - **Retrieve:** AP22110010181_courses_retrive
  - **Delete:** AP22110010606_courses_delete

Sample Source Code

SRM UNIVERSITY AP
Amaravati

```
public static void AP22110010180_courses_retrieve() {
    String sql = "SELECT * FROM courses";
    try (Statement stmt = conn.createStatement();
         ResultSet rs = stmt.executeQuery(sql)) {
        boolean hasData = false;
        while (rs.next()) {
            hasData = true;
            System.out.println("\n Course ID: " + rs.getInt("ID") +
                    "\n Program ID: " + rs.getInt("prog_id") +
                    "\n Code: " + rs.getString("cour_code") +
                    "\n Name: " + rs.getString("cour_name") +
                    "\n Category: " + rs.getString("cour_category") +
                    "\n Type: " + rs.getString("cour_type") +
                    "\n Pre-req: " + rs.getString("cour_pre_req") +
                    "\n Co-prereq: " + rs.getString("cour_co_prereq") +
                    "\n Progressive: " + rs.getString("cour_progressive") +
                    "\n PLS: " + rs.getString("cour_pls") + "\n------------------");
        }
        if (!hasData) {
            System.out.println("⚠ No courses found.");
        }
    } catch (SQLException e) {
        System.out.println("❌ Error retrieving courses: " + e.getMessage());
    }
}
```

```
public static void AP22110010181_courses_update(int id, Integer progId, String code, String name, String category,
                                                String type, String preReq, String coReq, String progressive, String pls) {
    StringBuilder sql = new StringBuilder("UPDATE courses SET ");
    List<Object> values = new ArrayList<>();
    if (progId != null) {
        sql.append("prog_id = ?, ");
        values.add(progId);
    }
    if (code != null && !code.isEmpty()) {
        sql.append("cour_code = ?, ");
        values.add(code);
    }
    if (name != null && !name.isEmpty()) {
        sql.append("cour_name = ?, ");
        values.add(name);
    }
    if (category != null && !category.isEmpty()) {
        sql.append("cour_category = ?, ");
        values.add(category);
    }
    if (type != null && !type.isEmpty()) {
        sql.append("cour_type = ?, ");
        values.add(type);
    }
    if (preReq != null && !preReq.isEmpty()) {
        sql.append("cour_pre_req = ?, ");
        values.add(preReq);
    }
    if (coReq != null && !coReq.isEmpty()) {
        sql.append("cour_co_prereq = ?, ");
        values.add(coReq);
    }

    if (progressive != null && !progressive.isEmpty()) {
        sql.append("cour_progressive = ?, ");
        values.add(progressive);
    }
    if (pls != null && !pls.isEmpty()) {
        sql.append("cour_pls = ?, ");
        values.add(pls);
    }
    if (values.isEmpty()) {
        System.out.println("⚠ No fields provided for update.");
        return;
    }
    sql.setLength(sql.length() - 2); // Remove trailing comma
    sql.append(" WHERE ID = ?");
    values.add(id);

    try(Connection conn = DriverManager.getConnection("jdbc:sqlite:C:/Users/gvidh/Desktop/Apps/mydatabase.db");
        PreparedStatement pstmt = conn.prepareStatement(sql.toString())) {
        for (int i = 0; i < values.size(); i++) {
            pstmt.setObject(i + 1, values.get(i));
        }
        int updated = pstmt.executeUpdate();
        System.out.println(updated > 0 ? "✅ Updated successfully!" : "❌ No such course with ID " + id);
        } catch (SQLException e) {
            System.out.println("❌ SQL Error: " + e.getMessage());
        }
}
```

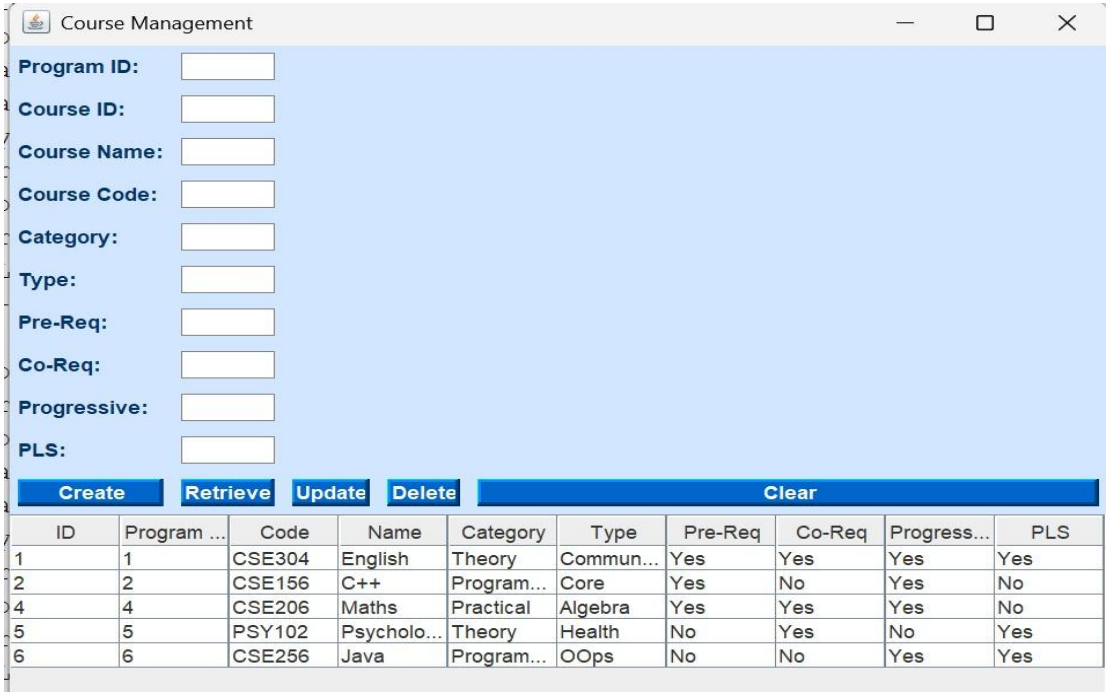
Sample Source Code

SRM UNIVERSITY AP
Amaravati

```java
public static void AP22110010167_courses_create(int progId, String code, String name,
        String category, String type, String preReq, String coPrereq, String progressive, String pls) {
    String sql = "INSERT INTO courses (prog_id, cour_code, cour_name, cour_category, "
            + "cour_type, cour_pre_req, cour_co_prereq, cour_progressive, cour_pls) VALUES (?, ?, ?, ?, ?, ?, ?, ?,
    try (PreparedStatement pstmt = conn.prepareStatement(sql)) {
        pstmt.setInt(1, progId);
        pstmt.setString(2, code);
        pstmt.setString(3, name);
        pstmt.setString(4, category);
        pstmt.setString(5, type);
        pstmt.setString(6, preReq);
        pstmt.setString(7, coPrereq);
        pstmt.setString(8, progressive);
        pstmt.setString(9, pls);
        pstmt.executeUpdate();
        System.out.println("✅ Course added successfully.");
    } catch (SQLException e) {
        System.out.println("❌ Error inserting course: " + e.getMessage());
    }
}
```

```java
public static void AP22110010606_courses_delete(int id) {
    String sql = "DELETE FROM courses WHERE ID = ?";
    try (PreparedStatement pstmt = conn.prepareStatement(sql)) {
        pstmt.setInt(1, id);
        int rowsDeleted = pstmt.executeUpdate();
        if (rowsDeleted > 0) {
            System.out.println("✅ Course deleted successfully.");
        } else {
            System.out.println("⚠️ Course not found.");
        }
    } catch (SQLException e) {
        System.out.println("❌ Error deleting course: " + e.getMessage());
    }
}
```

Sample Screen Shots



ID	Program ...	Code	Name	Category	Type	Pre-Req	Co-Req	Progress...	PLS
1	1	CSE304	English	Theory	Commun...	Yes	Yes	Yes	Yes
2	2	CSE156	C++	Program...	Core	Yes	No	Yes	No
4	4	CSE206	Maths	Practical	Algebra	Yes	Yes	Yes	No
5	5	PSY102	Psycholo...	Theory	Health	No	Yes	No	Yes
6	6	CSE256	Java	Program...	OOps	No	No	Yes	Yes

Conclusion

The Courses Module of the Outcome-Based Education (OBE) Implementation System has been successfully developed using Java Swing for the graphical user interface and SQLite for the backend database. This module allows users to perform essential CRUD (Create, Retrieve, Update, Delete) operations on course-related data through an intuitive and responsive GUI. The inclusion of a JTable provides a dynamic and interactive view of the stored course information, enhancing data visibility and user experience.

Key course details such as Program ID, Course Code, Course Name, Category, Type, Prerequisites, Corequisites, Progressive nature, and PLS have been seamlessly integrated and are displayed in real-time. This ensures that administrators can efficiently manage academic courses in alignment with OBE standards.

Overall, the project lays a solid foundation for building a comprehensive and scalable education management system. It improves academic planning and transparency while simplifying course handling for faculty and administrative staff.

SRM UNIVERSITY AP
Amaravati

Thank You