Botswana International University of S&T

Computer Science, College of Science

Activity: Project

Module Code	Comp 342
Module Description	Internet Programming
Work	Project
Marks	100 (**Final Grade will be 90% from the report and 10% demo)** **(Distribution may change as per lecturer's discretion)**
Due Date	Last Week of Class
Instructor	Dr. Hlomani Hlomani
Teaching Assistant	Mr. Thamang Tedie Madile,
Method of Submission	• Submission link will be provided on Blackboard closer to submission • Demo will be last week of class

Introduction

In this module students should have developed a critical and in-depth awareness of the theory, concepts, technology, practices, and professional issues related to data driven web application development using the .NET Framework. They should have had lots of practical exercises, giving them the opportunity to explore a variety of options in terms of web technologies to solve a given problem. The purpose of this project is to assess the student's grasp of the aforementioned. This is an individual project intended to assess the individual student's approach to solving a problem given a set of tools. The students can choose any Database Management System to hold their data but must remain within the .NET family of languages, preferably C# since it was the instructional language for this course

Case Study

The Office of the Deputy Vice Chancellor for Research and Innovation (DVC R&I) as well as that of the Deputy Vice Chancellor for Academic Affairs (DVC AA) are always in constant need to collect information on the research output and relevance for academic staff employed at BIUST. They periodically send out a form to all academic departments to collect such information. As you can imagine, this is a very tedious and cumbersome process both for the academic departments and for the offices of the DVCs. Once information has been received from the departments it has to be collated into a single report. This is prone to errors since the staff collating the information will have to deal with large sums of records manually and can easily omit some records.

The offices of the DVCs need a system to automate this data collection process – as a starting point, your job is to deliver a **database design**, **application design** and **implementation of both the database and application** that should:

1. Capture information such as:
    - Details of personnel involved in research (Employee number, First name, Last Name, Department, Position held, Highest Qualification and Institution obtained from, Research Interest/Research areas)
    - Details of research projects in the department (Project ID, Title of the research project, project leader[1], funding source[2], funding amount)
    - Details of research articles in accredited journals (Authors[3], title, journal name, volume, issue, pages, year)
    - Details of research articles in accredited conference proceedings (Authors[3], year, title, conference name, pages, publisher, location)
    - Details of book chapters written (Authors[3], chapter title, book title, publisher, pages, year)

[1] Project leader would naturally be an employee in the department – HINT: this will then be a foreign key

[2] Each project could have zero, one or multiple funders – HINT: you might want to capture funders details

[3] Each article could have multiple authors.

- Details on editorial or review activities carried out by staff (staff member, activity, organization/journal/conference, URL for the organization)

2. Generate Reports (See sample attached) such as:
   - List publications for a given employee (you can apply record filters such as: All records, in a certain year, in a particular journal/conference etc.)
   - List publications in a certain department (you can apply filters such as: All records, in a certain year, in a particular journal/conference etc.)
   - Publication outputs and conferences (see attached sample report for details.) This can also be filtered per department, per college, and All (BIUST wide)
   - Personnel involved in research (see attached sample report) this should also be filtered by department, and by college
   - List of research projects at BIUST. This should also be filtered by department, and college

The above is simply an outline of the database and application requirements. You should research the attributes of a journal, conference, and book chapter and will need to make your own assumptions to interpret or even extend the scenario as you go. Use your imagination as you see fit but you must clearly document all assumptions and extensions. **An example of such a report is attached with the assignment to help you conceptualize your solution (It is just an example and may not contain all the information stated in the project requirements).**

Your Task

1. Use tools you have learnt in this course and other courses to design the database that solves the problem described above. As a guide use the following:
   a. Relationship table
   b. Entity-Relationship Model: Develop a top down design of the data in the form of an entity-relationship diagram using crow's foot notation. You should note all assumptions you make about the data and the reasoning behind your design choices. Also include any appropriate constraints and list of entities types showing their attributes and identifiers.
   c. Normalize your database up to 3rd Normal Form
2. Implement your final database using any database management system of your choice your report)
3. With your database in place, design your application
4. **Implement your solution in ASP.NET (either through MVC or Web Forms). Make sure to use Object Oriented Programming concepts (e.g. classes. HINT classes for each entity) THIS IS YOUR MAIN FOCUS. YOU HAVE TO DELIVER A WORKING APPLICATION TO GET GOOD MARKS.**

HINT: You might benefit a lot from using Stored Procedure and Functions in SQL Server to create queries for your reports. You might also want to take a look at Crystal Reports as a solution for your reports.

Important Notes

- Makes sure that your full names and student IDs are on the front page of the assessment
- Clearly label all tasks and make sure you explain all your technical work.
- For your design work (task 1 and 3) make sure your diagrams a clearly labeled
- For your implementation of SQL tasks (task 2) you must provide your SQL code listings.
- You must submit your written report in either PDF or Word format
- All your SQL codes should be in clearly named separate files and zipped into one archive files that will be submitted along with your report
- SO, YOU SUBMIT: a report, one zip file contain several SQL code files, a zip containing your project folder
- YOU WILL OF COURSE DEMO YOUR PROJECT TO COMPLETE THE GRADE FOR THIS PROJECT

THE DETAILED GRADING CRITERIA IS PROVIDED OVERLEAF (NEXT PAGE).

Grading Criteria

70-100% (Excellent/Outstanding)

- Fully complete and accurate ER Model that captures all/most semantic aspects of the case study
- A fully normalized set of 3rd Normal form relations with all intermediate steps fully annotated
- A fully populated set of tables that encapsulates all primary and foreign keys and other constraints
- Fully documented SQL code listings
- A fully functional application that captures and stores data, with all reporting facilities implemented
- All work in the reports has been fully annotated, well laid out and easy to follow with suitable headings

- Fully complete and accurate ER Model that captures most of the semantic aspects of the case study
- A fully normalized (up to 3rd normal form) set of relations with most intermediate steps fully annotated
- A fully populated set of tables that encapsulate all primary and foreign keys and as well as other constraints.
- Fully documented SQL code listings
- A functional application that captures and stores data, with most of the reporting facilities implemented
- All work in the report has been fully annotated, well laid out and easy to follow with suitable headings

- Mostly complete and accurate ER Model that captures many semantic aspects of the case study
- A largely normalized set of 3rd normal form relations with many intermediate steps fully annotated
- A fully populated set of tables that encapsulates some primary & foreign keys and other constraints
- Documented SQL code listings
- A functional application that captures and stores data, with 70% of the reporting facilities implemented
- Most work in the report has been fully annotated, well laid out and easy to follow with suitable headings

40-49 (Satisfactory/Competent)

- A very basic ER Model that captures only some semantic aspects of the case study
- A partly normalized set of 3rd normal form relations with no/little annotation
- A barely populated set of tables that encapsulates only primary and/or foreign keys
- SQL code listings for creating tables
- A functional application that captures and stores data, with some of the reporting facilities implemented
- Work is presented in a very basic (but readable) state

35-39% (Unsatisfactory)

- A very basic ER Model that captures only small semantic aspects of the case study
- A very basic set of relations with no/little annotation or serious attempt at normalization
- A barely populated set of tables that encapsulates only primary and/or foreign keys
- No SQL code listings for creating tables
- A barely functional application that captures and stores data, with some of the reporting facilities implemented
- Work is presented in a very basic (but readable) state

- A very basic (or absent) ER Model that captures only small semantic aspects of the case study
- A very basic (or absent) set of relations with no/little annotation or serious attempt at normalization
- No SQL code listings for creating tables
- A barely populated set of tables that encapsulates only primary and/or foreign keys
- A non functional application that captures and stores data, with some of the reporting facilities implemented
- Work is presented in a very basic (but readable) state