Mini Project - 1

|  |  |
| --- | --- |
| **Student Name/ID Number:** | Ida Bagus Ketut Yoghantara |
| **Unit Number and Title:** | ACWD Module 4 – Database Design & Implementation |
| **Academic Year:** | 2022 |
| **Unit Assessor:** | Arvinder Kaur |
| **Project Title:** | Database Design for Community Portal |
| **Issue Date:** |  |
| **Submission Date:** |  |
| **Internal Verifier Name:** |  |
| **Date:** | 25 July 2022 |

|  |
| --- |
| **Learner declaration** |
| I certify that the work submitted for this assignment is my own and research sources are fully acknowledged.  Student signature:  Date: 25 July 2022 |

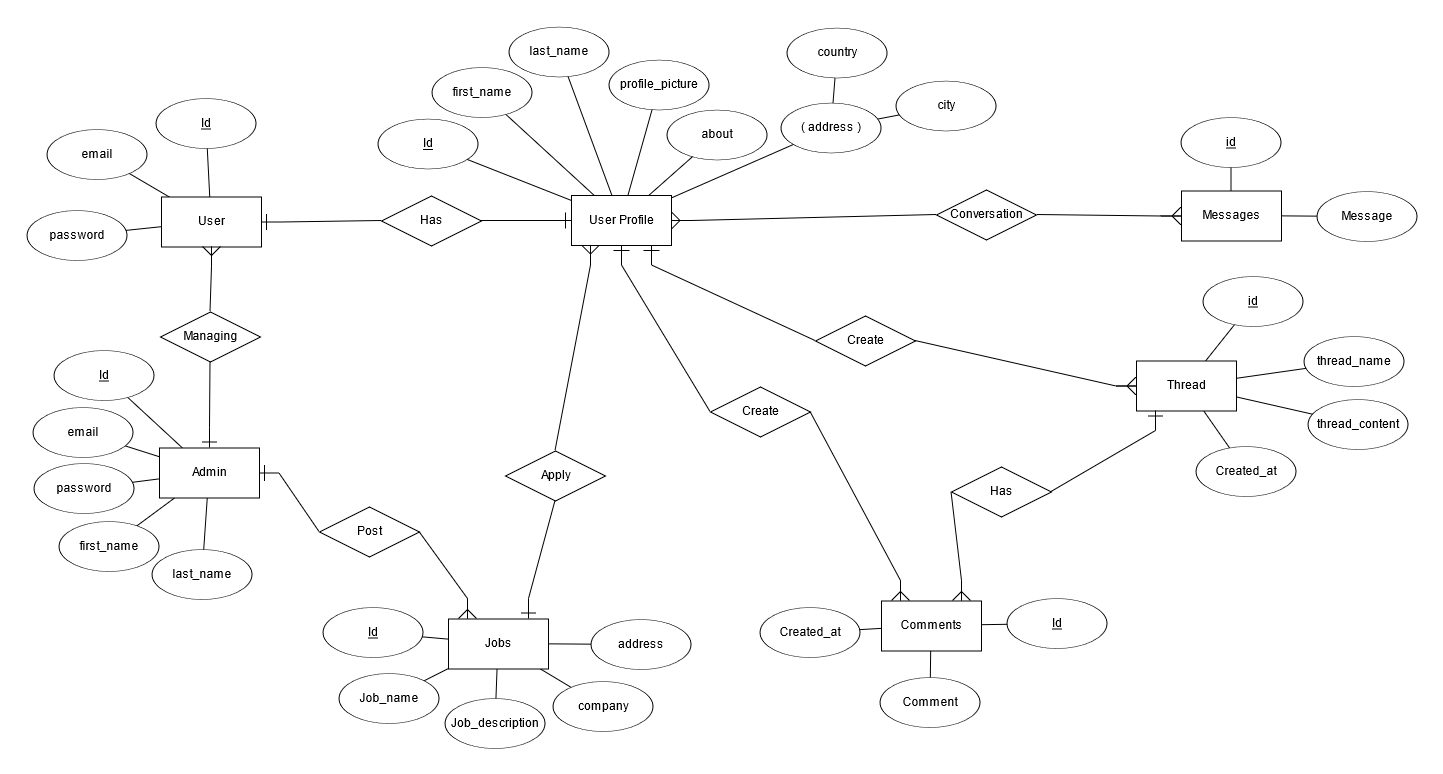
|  |
| --- |
| **Purpose of this project** |
| **Purpose of this project**  To demonstrate your capabilities in the following areas:   * Design a Database for Community Portal |
| **Submission Format** |
| 1. Description of Entities in the database 2. Screen capture of the ER Diagram 3. Screen capture of Data 4. Screen capture of Normalized Database Design 5. Documentation of Relationships |
| **Project Brief & Guidance** |
| **Scenario:**  **Refer to the Project Scenario for the Module Project**  You have been approached by ‘ABC Jobs Pte Ltd’ as a website developer to develop a community portal for Software Developers. The project will be carried over through Module 3, Module 4, Module 5 and Capstone project. For this module the scope is to Design, Develop, Implement & Document Apache Struts Framework Website.  The Scope of the Project is to design a Community Portal Similar to Linkedin.com. Users will be able to register in the portal using the Registration Page. Users of the portal can search for other users using various parameters such as First Name, Last Name, Company Name, City & Country. Users will be able to view the Public Profile of users after searching them. The portal allow users to login, request for forgotten password and Update their profile information  The scope of the mini project is to design the database.  **The overview of the project is as below**  There are 2 types of users in this Community portal. They are   1. Software Programmer 2. Administrator   **Software Programmer should be able to perform following functions in the portal** |

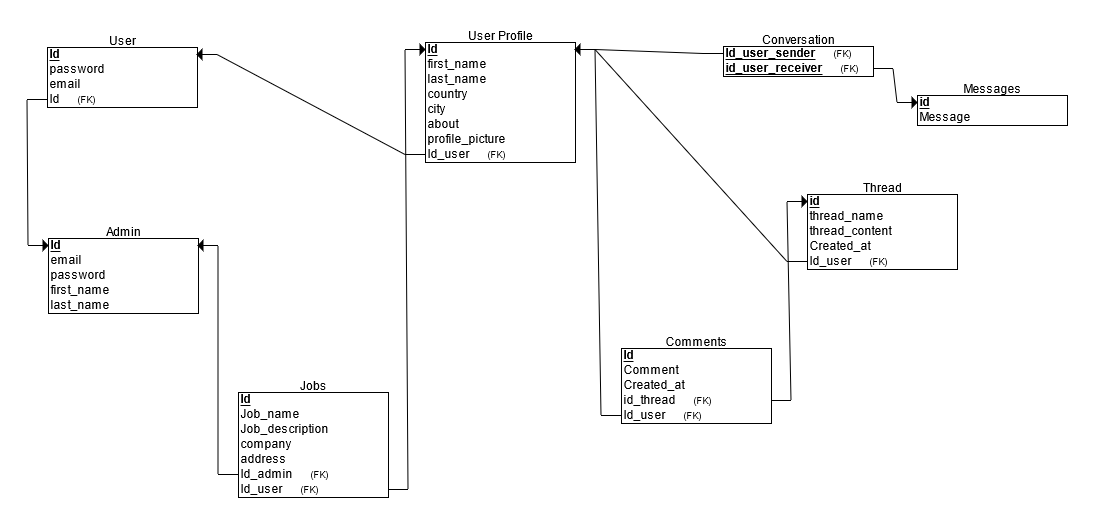
|  |
| --- |
| 1. Allow the programmers to register in the portal, show a thank you page & send a registration confirmation email. 2. Search & Find Other Programmers after login and view their profile. 3. Provide Login Page 4. Provide password retrieval functionality. 5. Update their Profile after logging in.   Following Functionality is part of Database Design & Will be enhanced as features as required in the Capstone Project, Depending on time availability.   1. Send Messages to Each Other on the Portal 2. Create Threads & Post Replies to a Thread 3. Post Job Opportunities in the Portal   **Administrator should be able to perform following functions in the portal**   1. Administer user data. 2. Send bulk email inviting programmers to register on the community portal   **The portal consist of the following Key pages (For Reference)**   1. Community Portal Home Page 2. Registration Page 3. Registration Confirmation Page 4. Update Profile Page 5. Search Users Page 6. List Search Results 7. Public Profile Page 8. Registration Confirmation Email 9. Login Page 10. Forget Password Page 11. Design the Forget Password Confirmation Page   Following Functionality is part of Database Design and will not be part of development & Will be enhanced as features are required in the Capstone Project, Depending on time availability.   1. Send Messages 2. Read Messages 3. Post in Message Board 4. List Message Board 5. Read A Thread 6. Post Job Opportunities 7. List Job Opportunities & Responses   **The scope of this assignment**  The scope is to design the community portal database.   * 1. Identify the Entities in the Community Portal Database   2. Create ER Diagram for the Community Portal Database |

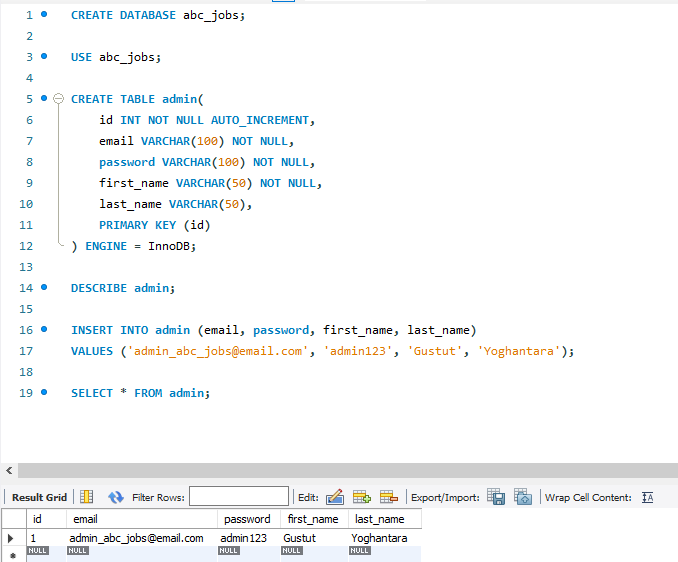
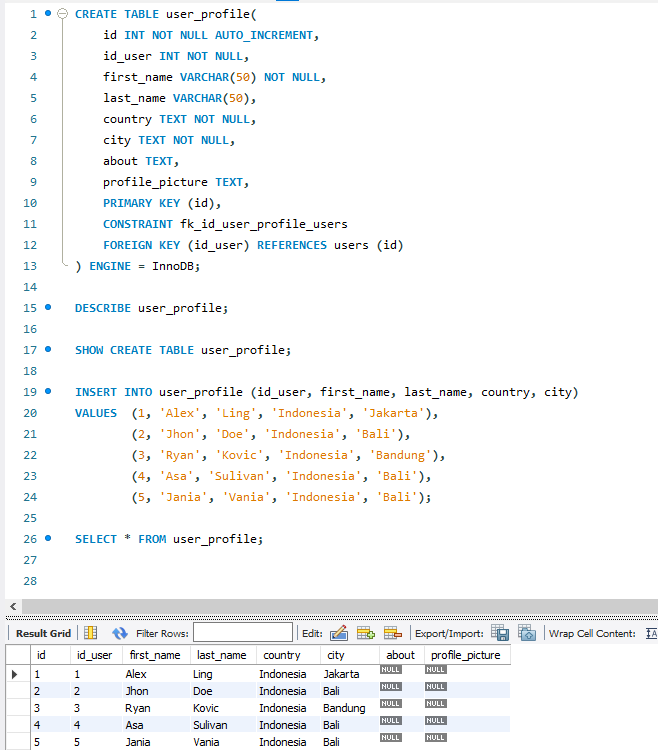
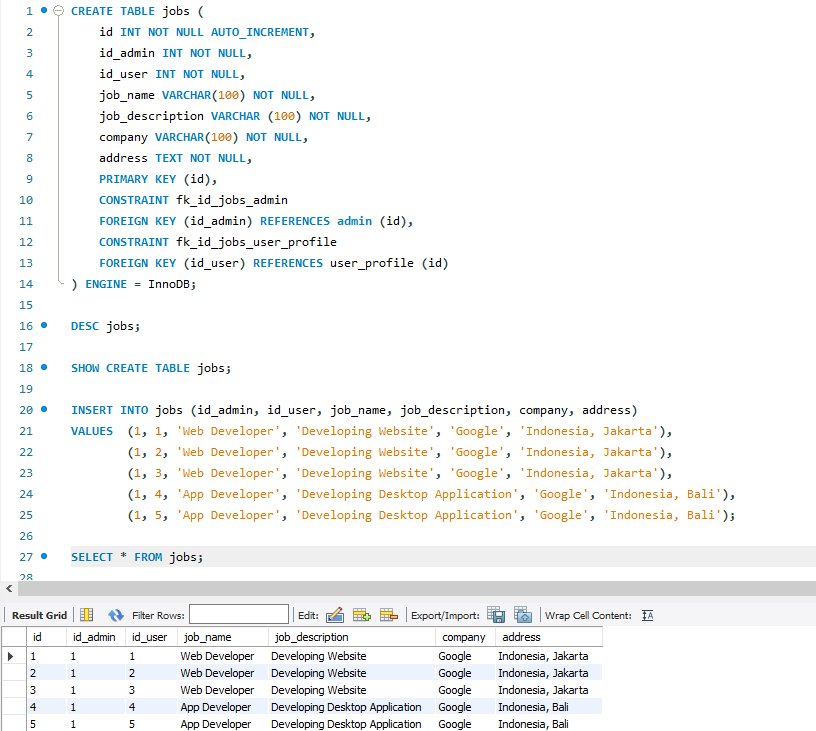
1. Create Sample data for all tables in the community portal (At least 3 tables)
2. Normalize the complete database to 3rd normal form (Provide screen capture of the normalization done)
3. Identify the relationships in tables & document them
4. Identify the entities in the community portal

|  |  |  |
| --- | --- | --- |
| Entities Name | Description | Column |
| Users | The person that using this website to find a job or whatever the website provides | * Id (PK) * Email * Password |
| User Profile | Information about users who registered on the website | * Id (PK) * Id\_company (FK) * Id\_job (FK) * First\_name * Last\_name * Address * About * Profile\_picture |
| Admin | Administrator of the website who managed the website and all the data of the users | * Id (PK) * Email * Password * First\_name * Last\_name |
| Jobs | Information about all of the job’s opportunity in the website | * Id (PK) * Id\_admin (FK) * Id\_user (FK) * Company * Job\_name * Job\_description * Address |
| Messages | Database for user send message to another user | * Id (PK) * Id\_user\_sender (FK) * Id\_user\_receiver (FK) * Message * Created\_at |
| Threads | A running commentary of text messages pertaining to one topic or question | * Id (PK) * Id\_user (FK) * Thread\_name * Thread\_content * Created\_at |
| Comments | Comment is replies from the user for the existing thread or topics | * Id (PK) * Id\_user (FK) * Id\_thread (FK) * Comment * Created\_at |

1. Create ER Diagram for the Community Portal Database



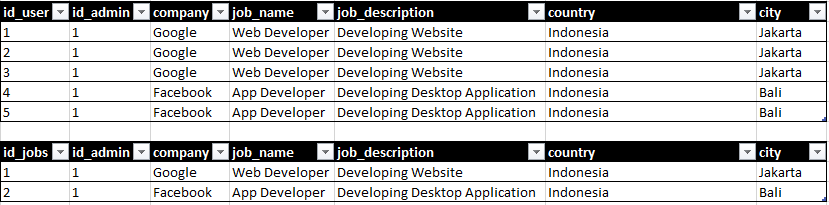


1. Create sample data for all tables in the Community Portal (At least 3 tables)  
     
   **1. Admin**  
     
     
   **2. User Profile**  
     
     
     
     
     
   **3. Jobs**  
   
2. Normalize the complete database to 3rd normal form (Provide screen capture of the normalization done)  
   **Jobs Table**   
   

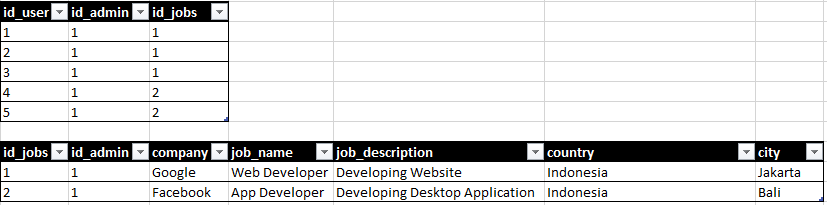
* First Normal Form (1NF)  
  The first normal form is the normal form of database where data must not contain repeating group. The database is in First normal form If,
* It contains only atomic values (single cell only have single value)
* Each Record needs to be unique and there are no repeating groups. Repeating group means a table contains 2 or more values of columns that are closely related.



* Second Normal Form (2NF)  
  The data is said to be in second normalized form If,
* It is in First normal form
* There should not be any partial dependency of any column on primary keys. Means the table have concatenated primary key and each attribute in table depends on that concatenated primary key.
* All Non-key attributes are fully functionally dependent on primary keys. If primary is not composite key then all non-key attributes are fully functionally dependent on primary key.



* Third Normal Form (3NF)  
  The database is in Third normal form if it satisfies following conditions:
* It is in Second normal form
* There is no transitive functional dependency.

****

1. Identify the relationships in tables & document them

* One to one relationship  
  It is used to create a relationship between two tables in which a single row of the first table can only be related to one and only one records of a second table. Similarly, the row of a second table can also be related to anyone row of the first table.

1. Users to user\_profile  
   users can only have one user\_profile and user\_profile can only have one user (account)

* One to many relationship  
  It is used to create a relationship between two tables. Any single rows of the first table can be related to one or more rows of the second tables, but the rows of second tables can only relate to the only row in the first table. It is also known as a **many to one** relationship.

1. Admin to Users  
   Admin can manage many Users data at once
2. Admin to Jobs  
   Admin can post many jobs and jobs can only have 1 admin that posted it
3. User\_profile (user) to jobs  
   User can apply to one job but job can be applied by many user
4. User\_profile (user) to thread  
   User can create many threads and that thread can only have one creator that is user
5. User\_profile (user) to comments  
   User can post as many comments as they want and that comment has only one unique user or creator of that comment
6. Thread to comment  
   thread can consist as many comments on it

* Many to many relationship  
  It is **many to many** relationships that create a relationship between two tables. Each record of the first table can relate to any records (or no records) in the second table. Similarly, each record of the second table can also relate to more than one record of the first table. It is also represented an **N: N** relationship.

1. User to Messages or conversation  
   Two user involved in conversation and every user can involve more than one conversation