A picture containing text, clipart

Description automatically generated

**Formative Assignment 2**

|  |  |
| --- | --- |
| Module Name | WSQ Database Design and Implementation (SF) |
| Course Name | Postgraduate Diploma in Software Engineering |
| Assignment Title | Database Implementation |

|  |  |
| --- | --- |
| Learner Name | Huynh Minh Phu |

|  |
| --- |
| Learner declaration |
| I certify that the work submitted for this assignment is my own and research sources are fully acknowledged.  Learner signature: Phu Date: 15/12/2023 |

## Purpose of this assignment

To demonstrate your capabilities in the following areas:

* Implement a MySQL Database based on design created in Assignment 1
* Write SQL queries for functional requirements
* Set user access and privileges

## Submission Format

The submission is in the form of an individual written report with examples and illustrations. This should be written in a concise, formal academic style using single spacing and font size 12.You are required to make use of headings, paragraphs, and subsections as appropriate, and all work must be supported with research and referenced using the Harvard referencing system. Please also provide a bibliography using the Harvard referencing system.

## Submission file

Submission File Name format: CohortCode\_FullName\_AssignmentNumber

E.g.: PGSE-DDI-0823\_AliceTan\_A2.docx

## Scenario

You currently work as a Data Engineer for Brightica design agency, where you design and implement data models for client-centric products. As part of the role, your manager Mr. Andrew assigned the project to develop an optimal database design to deliver Rich Internet Application for Boutiqa. Boutiqa is a marketplace for sellers to promote their products and for consumers to purchase with ease. The company wants to have a consumer-centric application with an enhanced user experience.

You are required to demonstrate your capabilities in the following areas:

* Planning of database use group
* Conceptual, Logical, and Physical design of the database
* Writing queries and stored procedures to optimize the system performance and management reports.

The scope of the project in this module is to design and develop and implement the database. The overview of the project is as below:

There are 3 types of users:

1. Sellers
2. Consumers
3. Administrator

Sellers should be able to perform following functions in the portal:

1. Register in the portal.
2. Update their Profile after logging in.
3. Maintain the product catalog to promote their products.

Consumers should be able to perform following functions in the portal:

1. Register in the portal.
2. Update their Profile after logging in.
3. Search products.
4. Choose products to view the details.
5. Add, edit, and remove items in the shopping cart.

Administrator should be able to perform following functions in the portal:

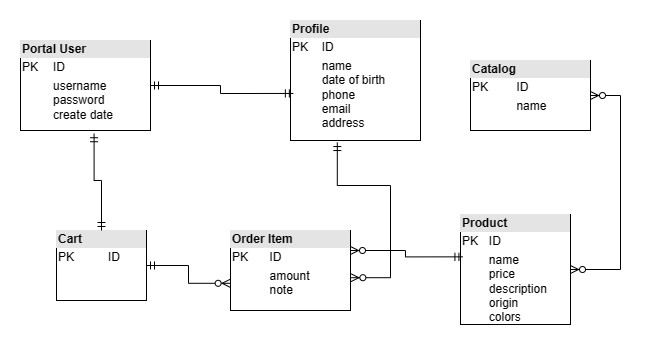
1. Administer user data.
2. Send bulk email invite to potential clients to register.

You will be required to:

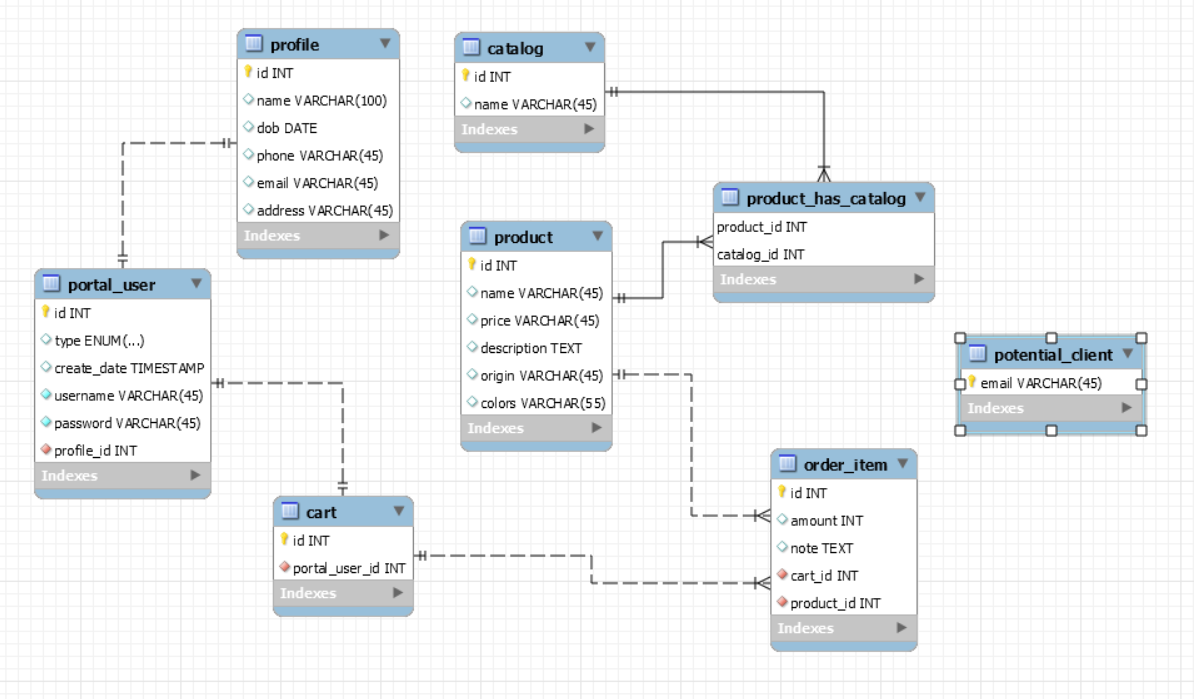
* Create a logical design with appropriate field names, field type & size
* Implement the database in MySQL
* Briefly list the web pages in the scenario and relevant queries which will be used by each of them
* Create a user name in MySQL which can be used by the application, give required access privileges.
* Create indexes and explain the rationale for creating those indexes

## Evidence to exhibit

* Screenshot of Logical Design



* Screenshot of Physical Design (interim)



* Implement the database in MySQL

drop database if exists portal;

CREATE database portal;

use portal;

CREATE TABLE IF NOT EXISTS `portal`.`profile` (

`id` INT NOT NULL AUTO\_INCREMENT,

`name` VARCHAR(100) NULL,

`dob` DATE NULL,

`phone` VARCHAR(45) NULL,

`email` VARCHAR(45) NULL,

`address` VARCHAR(45) NULL,

PRIMARY KEY (`id`))

ENGINE = InnoDB;

CREATE TABLE IF NOT EXISTS `portal`.`portal\_user` (

`id` INT NOT NULL AUTO\_INCREMENT,

`type` ENUM('SELLER', 'CONSUMER', 'ADMIN') NULL,

`create\_date` TIMESTAMP NULL,

`username` VARCHAR(45) NOT NULL,

`password` VARCHAR(45) NOT NULL,

`profile\_id` INT NOT NULL,

PRIMARY KEY (`id`),

UNIQUE INDEX `username\_UNIQUE` (`username` ASC) VISIBLE,

INDEX `fk\_portal\_user\_profile\_idx` (`profile\_id` ASC) VISIBLE,

CONSTRAINT `fk\_portal\_user\_profile`

FOREIGN KEY (`profile\_id`)

REFERENCES `portal`.`profile` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION);

CREATE TABLE IF NOT EXISTS `portal`.`catalog` (

`id` INT NOT NULL AUTO\_INCREMENT,

`name` VARCHAR(45) NULL,

PRIMARY KEY (`id`));

CREATE TABLE IF NOT EXISTS `portal`.`product` (

`id` INT NOT NULL AUTO\_INCREMENT,

`name` VARCHAR(45) NULL,

`price` VARCHAR(45) NULL,

`description` TEXT NULL,

`origin` VARCHAR(45) NULL,

`colors` VARCHAR(55) NULL,

PRIMARY KEY (`id`));

CREATE TABLE IF NOT EXISTS `portal`.`cart` (

`id` INT NOT NULL AUTO\_INCREMENT,

`portal\_user\_id` INT NOT NULL,

PRIMARY KEY (`id`),

INDEX `fk\_cart\_portal\_user1\_idx` (`portal\_user\_id` ASC) VISIBLE,

CONSTRAINT `fk\_cart\_portal\_user1`

FOREIGN KEY (`portal\_user\_id`)

REFERENCES `portal`.`portal\_user` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION);

CREATE TABLE IF NOT EXISTS `portal`.`potential\_client` (

`email` VARCHAR(45) NOT NULL,

PRIMARY KEY (`email`));

CREATE TABLE IF NOT EXISTS `portal`.`potential\_client` (

`email` VARCHAR(45) NOT NULL,

PRIMARY KEY (`email`));

CREATE TABLE IF NOT EXISTS `portal`.`order\_item` (

`id` INT NOT NULL AUTO\_INCREMENT,

`amount` INT NULL,

`note` TEXT NULL,

`cart\_id` INT NOT NULL,

`product\_id` INT NOT NULL,

PRIMARY KEY (`id`),

INDEX `fk\_order\_item\_cart1\_idx` (`cart\_id` ASC) VISIBLE,

INDEX `fk\_order\_item\_product1\_idx` (`product\_id` ASC) VISIBLE,

CONSTRAINT `fk\_order\_item\_cart1`

FOREIGN KEY (`cart\_id`)

REFERENCES `portal`.`cart` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_order\_item\_product1`

FOREIGN KEY (`product\_id`)

REFERENCES `portal`.`product` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION);

CREATE TABLE IF NOT EXISTS `portal`.`product\_catalog` (

`product\_id` INT NOT NULL,

`catalog\_id` INT NOT NULL,

PRIMARY KEY (`product\_id`, `catalog\_id`),

INDEX `fk\_product\_has\_catalog\_catalog1\_idx` (`catalog\_id` ASC) VISIBLE,

INDEX `fk\_product\_has\_catalog\_product1\_idx` (`product\_id` ASC) VISIBLE,

CONSTRAINT `fk\_product\_has\_catalog\_product1`

FOREIGN KEY (`product\_id`)

REFERENCES `portal`.`product` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_product\_has\_catalog\_catalog1`

FOREIGN KEY (`catalog\_id`)

REFERENCES `portal`.`catalog` (`id`)

ON DELETE NO ACTION

ON UPDATE NO ACTION);

* List of pages with query and screenshot of result

|  |  |  |
| --- | --- | --- |
| **Page** | **Purpose of query** | **Query** |
| <list the pages> | <a possible purpose of the query which is used in this page for the function> | <query>  <screenshot> |
| Registration | For account creation of new users | INSERT INTO portal\_user (`type`, username, `password`, profile\_id) VALUES ('ADMIN', 'admin', '123', 1); |
| Update profile info | For update profile info | UPDATE `profile` SET `name` = 'Tran Van A', phone='039193910', email='demo@gmail.com' where id = 1; |
| Add New Product | Add new product | INSERT INTO product(name, price, description, origin, colors)  VALUES('Giay nike air force 1', 300000, 'GIay dep', 'VIETNAM', 'WHITE, BLACK'); |
| Search Product | Search product by name | SELECT \* FROM PRODUCT where name like '%air force 1%'; |
| Get product info | Get product info by id | SELECT \* FROM PRODUCT where id = 1; |
| Update product info | Update product info | UPDATE product set price=90000, origin='CHINA' where id = 1; |
| Add new catalog | Add new catalog | INSERT INTO catalog(name) values ('Thoi trang'), ('Giay dep'); |
| Add product to cart | Add product to cart | INSERT INTO order\_item(amount, note, cart\_id, product\_id) values(3, 'None', 2, 2); |

* Provide SQL of the user with access privileges

CREATE USER 'user1'@'localhost' IDENTIFIED BY '123';

GRANT SELECT ON portal.\* TO 'user1'@'localhost';

CREATE USER 'user2'@'localhost' IDENTIFIED BY '123';

GRANT ALL ON portal.\* TO 'user2'@'localhost';

* Screenshot of index

|  |  |  |
| --- | --- | --- |
| **Table** | **Index name** | **Query** |
| profile | profile\_email\_idx  profile\_phone\_idx  profile\_dob\_idx | CREATE INDEX profile\_email\_idx on profile(email);  CREATE INDEX profile\_phone\_idx on profile(phone);  CREATE INDEX profile\_dob\_idx on profile(dob); |
| Portal\_user | portaluser\_username\_idx | CREATE INDEX portaluser\_username\_idx on portal\_user(username); |
| Product | product\_price\_idx | CREATE INDEX product\_price\_idx on product(price); |

* Rationale for index creation

|  |  |
| --- | --- |
| **Index name** | **Retionale** |
| profile\_email\_idx | Helps search users by email more effectively.  SELECT \* FROM profile where email |
| profile\_phone\_idx | Helps search users by phone more effectively.  SELECT \* FROM profile where phone = ? |
| profile\_dob\_idx | Helps search users by date of birth more effectively.  SELECT \* FROM profile where dob = ? |
| portaluser\_username\_idx | Helps search users by username more effectively when login.  SELECT \* FROM portal\_user where username = ? |
| product\_price\_idx | Helps search users by price in range more effectively.  SELECT \* FROM portal\_user where price >= ? and price <= ? |