

make a **180°** life change

University of British Columbia Okanagan

COSC 304 - Project Design

Names Removed

CONTENTS:

1.	Introduction	2
1.1.	Purpose	2
1.2.	Mission Statement	2
1.3.	Executive Summary	2
2.	Domain Assumptions	3
2.1.	Customers	3
2.2.	Degree orders	3
2.3.	Degree Product	3
2.4.	Shipment	3
2.5.	Warehouse	3
3.	Entity Description	4
3.1.	UML Diagram	4
3.2.	UML Diagram brief description	4
3.3.	Entity Description	5
3.4.	Relational Assumptions	7
3.5.	Relational Schema – SQL DDL	8
4.	Interface	
4.1.	Site Map	10
4.2.	List of Features	10
4.3.	Interface Description	10
5.	Going Forward	
5.1.	Limitations	11
5.2.	Projected Problems	11

1 INTRODUCTION

1.1 Purpose of Document

The purpose of this document is to outline the design and underlying structure of the 180° website by providing domain assumptions, an UML diagram and corresponding SQL DDL and a site map.

1.2 Mission Statement

To give everyone the opportunity to change their life by 180 degrees and find their dream job by providing them with the required degree from one of the best institutions in Canada or the institution of their choice.

1.3 Executive Summary

180degrees.ca is an online shop for university degrees where anyone can buy their desired degree. The website offers degrees from the top 20 Canadian institutions according to topuniversity.com with the most prestigious one being the most expensive. Customers can either search for a degree by degree type, specialization or institution name, browse them, or decide to buy a degree by any institution of their choice for a fixed price. In that case, the customer has to provide the required institution information. When buying a degree, a first and last name and the desired year of graduation is required.

Everyone can browse degrees and add them to the shopping cart, but in order to purchase one, a customer has to have an account. To create an account, a user has to provide first and last name, a unique email address and a password. When checking out, the shipping address and name and credit card details are required.

Company personnel can log in as administrator who is able to change prices and add, update or remove degrees.

2 DOMAIN ASSUMPTIONS

2.1 User

- Users are identified by a unique, auto-incremented integer
- User firstName must not be null
- User lastName must not be null
- User emailAddresses must be unique and not null
- User password must not be null
- Non-users may browse the website, but must create an account before making a purchase
- Company personnel can log in as administrator

2.2 Degree order

- Orders are identified by a unique, auto-incremented integer
- odate tracks the date the order was placed
- Customers can only pay via credit card
- The credit card information (cardNumber, cardName, cardExpiration, securityCode) must not be null

2.3 Degree Product

- Degrees are identified by a unique, auto-incremented integer
- There are 20 pre-set degrees to choose from
- A customer may customize the degree if it is not included in the list of 20
- Degree institution must not be null
- Degree dspecialization must not be null
- Degree price must not be null
- Degree dtype must be either 'Bachelor', 'Master', or 'PhD'

2.4 Shipment

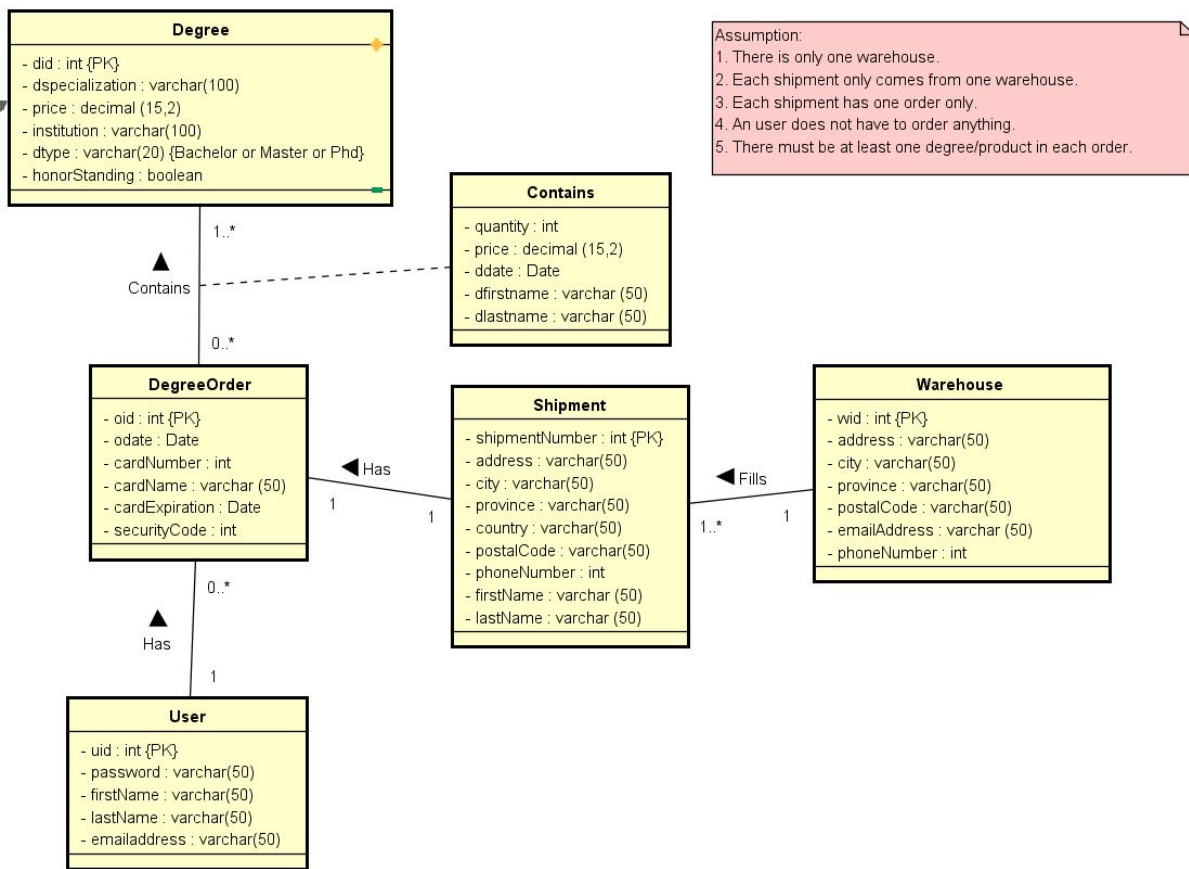
- Shipment address must not be null
- Shipment city must not be null
- Shipment province must not be null
- Shipment Country must not be null
- Shipment postalCode must not be null
- Each shipment comes from one Warehouse

2.5 Warehouse

- There is only one warehouse
- The warehouse's full address must not be null
- The warehouse fills the shipments

3. ENTITY DESCRIPTION

3.1 UML Diagram



3.2 UML Diagram brief description

3.2.1 User

A user is somebody who has successfully created an account. A person does not have to login in order to view products; however, a person must have logged in and placed an order to be a customer. The following information are in the User entity: an unique user id, password, first name, last name, email address.

3.2.2 Degree

Degrees are what our group has selected to sell for this project. In other words, a degree is a product in our definition. It may be used interchangeably. The following information are in the Degree entity: an unique degree id, degree specialization, price of a degree, institution, degree type, and honor standing.

3.2.3 DegreeOrder

When a customer placed an order, the following information will be recorded: order id, order date, credit card number, credit card name, credit card expiration date, and the 3-digit security code on the back of the credit card.

3.2.4 Contains

Apart from the general degree information for each order, the following information are also recorded: quantity, price, degree graduation date, first and last name shown on the degree.

3.2.5 Shipment

A shipment entity contains the following shipping information: a shipment id, address, city, province, country, postal code, phone number, first and last name of receiver.

3.2.6 Warehouse

A warehouse is where a shipment for each order is filled. A warehouse contains the following information: warehouse id, address, city, province, postal code, email address, and phone number.

3.3 Entity Description

3.3.1 Warehouse

Attribute	Description
wid: INT	Warehouse ID. Cannot be null
address: VARCHAR(50)	Address of Warehouse
city: VARCHAR(50)	City Warehouse is located in
province: VARCHAR(50)	Province of Warehouse
emailAddress: VARCHAR(50)	Email Address of Warehouse
phoneNumber: INT	Phone number of Warehouse

3.3.2 Shipment

Attribute	Description
shipmentNumber: INT	Auto-increment integer that identifies the shipment
address: VARCHAR(50)	Address of Shipment
phoneNumber: INT	Phone Number of the customer receiving the shipment
firstName: VARCHAR(50)	First name of the customer receiving the shipment
lastName: VARCHAR(50)	Last name of the customer receiving the shipment
wid: INT	Warehouse ID of the warehouse the shipment is filled by

3.3.3 User

Attribute	Description
uid: INT	Unique id to the user
password: VARCHAR(50)	User's account password
firstName: VARCHAR(50)	User's first name
lastName: VARCHAR(50)	User's last name
emailAddress: VARCHAR(50)	User's email address

3.3.4 Degree

Attribute	Description
did: INT	Unique, Identifiable ID to each degree
dspecialization: VARCHAR(50)	Specialization of the degree
price: DECIMAL(15,2)	Price of the degree
institution: VARCHAR(50)	Institution for the degree
dtype: VARCHAR(20)	Type of Degree. Choose from 'Bachelor', 'Master', or 'PhD'
honorStanding: BOOLEAN	Yes or No for honorStanding for the Degree

3.3.5 Degree Order

Attribute	Description
oid: INT	Unique, Identifiable ID for each order
odate: DATE	Date the order was placed
cardNumber: INT	Customer's credit card number
cardExpiration: DATE	Customer's credit card expiration date
securityCode: INT	Customer's credit card security code
shipmentNumber: INT	Foreign key to Shipment(shipmentNumber)
uid: INT	Foreign key to User(uid). The ID of the user making the order

3.3.6 Contains

Attribute	Description
did: INT	Unique degree ID of the degree being ordered
oid: INT	Unique order ID of the order being placed
quantity: INT	Amount of degrees in the order
price: DECIMAL(15,2)	Price of the order
dfirstName: VARCHAR(50)	First name that is on the degree being ordered
dlastName: VARCHAR(50)	Last name that is on the degree being ordered

3.4 Relational Assumptions

Relationship	Description
User has DegreeOrder	This is a many-to-one relationship that associates a customer/user with their order. A user can place multiple orders. An order can only be associated with one user.
DegreeOrder contains Degree	This is a many-to-many relationship that associates the degree that is in the order. An order may have multiple degrees placed in an order. Each degree may be placed multiple times in different orders.
Warehouse fills Shipment	This is a many-to-one relationship that associates the shipment being filled at a warehouse. There is only one warehouse, but it can fill multiple shipments. A shipment only comes from one warehouse.
Shipment has DegreeOrder	This is a one-to-one relationship that associates the order being shipped. Each shipment only contains one Degree Order, and each order is placed in one shipment at a time.

3.5 Relational Schema - SQL DDL

```

CREATE TABLE Warehouse (
    wid          INTEGER    NOT NULL,
    address      VARCHAR(50) NOT NULL,
    city         VARCHAR(50) NOT NULL,
    province    VARCHAR(50) NOT NULL,
    postalCode   VARCHAR(50) NOT NULL,
    emailAddress VARCHAR(50) NOT NULL,
    phoneNumber  INTEGER    NOT NULL,
    PRIMARY KEY (wid)
);

CREATE TABLE Shipment (
    shipmentNumber INTEGER    AUTO_INCREMENT,
    address         VARCHAR(50) NOT NULL,
    city           VARCHAR(50) NOT NULL,
    province       VARCHAR(50) NOT NULL,
    country        VARCHAR(50) NOT NULL,
    postalCode     VARCHAR(50) NOT NULL,
    phoneNumber    INTEGER    NOT NULL,
    firstName      VARCHAR(50) NOT NULL,
    lastName       VARCHAR(50) NOT NULL,
    wid            INTEGER    NOT NULL,
    PRIMARY KEY (shipmentNumber),
    FOREIGN KEY (wid) REFERENCES Warehouse(wid)
        ON DELETE NO ACTION
        ON UPDATE CASCADE
);

CREATE TABLE User (
    uid          INTEGER    AUTO_INCREMENT,
    password     VARCHAR(50) NOT NULL,
    firstName    VARCHAR(50) NOT NULL,
    lastName     VARCHAR(50) NOT NULL,
    emailAddress VARCHAR(50) NOT NULL,
    PRIMARY KEY (uid)
);

CREATE TABLE Degree (
    did          INTEGER    AUTO_INCREMENT,
    dspecialization VARCHAR(100) NOT NULL,
    price        DECIMAL(15,2) NOT NULL,
    institution  VARCHAR(100) NOT NULL,
    dtype       VARCHAR(20) CHECK(dtype IN ('Bachelor', 'Master', 'PhD')),
    honorStanding BOOLEAN,
    PRIMARY KEY (did)
);

```

```

CREATE TABLE DegreeOrder (
    oid                INTEGER    AUTO_INCREMENT,
    odate              DATE       NOT NULL,
    cardNumber         INTEGER    NOT NULL,
    cardName           VARCHAR(50) NOT NULL,
    cardExpiration     DATE       NOT NULL,
    securityCode       INTEGER    NOT NULL,
    shipmentNumber     INTEGER    NOT NULL,
    uid                INTEGER    NOT NULL,
    PRIMARY KEY (oid),
    FOREIGN KEY (shipmentNumber) REFERENCES Shipment(shipmentNumber)
        ON DELETE NO ACTION
        ON UPDATE CASCADE,
    FOREIGN KEY (uid) REFERENCES User(uid)
        ON DELETE NO ACTION
        ON UPDATE CASCADE
);

```

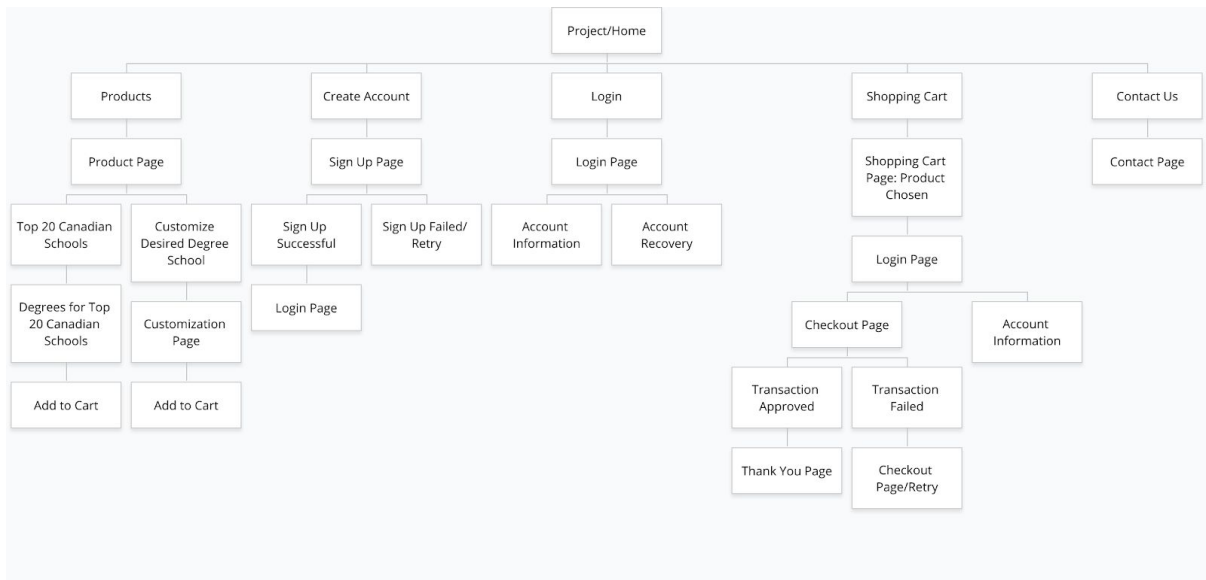
```

CREATE TABLE Contains (
    did                INTEGER    NOT NULL,
    oid                INTEGER    NOT NULL,
    quantity           INTEGER    NOT NULL,
    price              DECIMAL(15,2) NOT NULL,
    ddate              DATE       NOT NULL,
    dfirstName         VARCHAR(50) NOT NULL,
    dlastName          VARCHAR(50) NOT NULL,
    PRIMARY KEY (did, oid),
    FOREIGN KEY (did) REFERENCES Degree(did)
        ON DELETE NO ACTION
        ON UPDATE CASCADE,
    FOREIGN KEY (oid) REFERENCES DegreeOrder(oid)
        ON DELETE NO ACTION
        ON UPDATE CASCADE
);

```

4. SITE MAP

4.1 Site Map



4.2 List of Features

- Search for a degree by degree type, specialization or institution name
- Browse degrees
- Customize a degree
- Create user account
- Add, remove or update products in shopping cart
- List of shipped products, prices and their delivery status
- Company members have access as administrator who can manage the products

4.3 Interface Description

4.3.1 Home

The Home page is the first page visited on the site. It offers a list of options: to go to the products, create an account, login, visit your shopping cart, and a contact page. If the user decides to create an account, they can select the 'create an account' button and input their information. After successfully creating an account, it will direct them to the login page. If the user has a pre-existing account, they may log in using the 'login' button, and enter their account information. After successfully logging in, the login page will direct them to the account information page. There will be a home button direct users to the home page. The 'Contact Us' button will take the user to a page with information to contact the website director.

4.3.2 Product page

The product page will allow the user to browse the top 20 Canadian schools, search a degree by institution, specialization, type, or customize their desired degree. If a user decides to customize their degree, they will be directed to the customization page where all the required degree information can be inputted. Once the user has picked a listed degree, or customized a degree, they can add it to their cart.

4.3.3 Shopping Cart Page

The user can view their order on the shopping cart page. They can proceed to place the order, or make changes to their order. They can add, remove, or update products in the shopping cart. Once the order is placed, a user may view the list of shipped products, prices, and its delivery status.

5 GOING FORWARD

5.1 Limitations

- We don't know what the degrees from specific institutions look like, so we can't show a preview image
- There is not a real warehouse, so degrees could not be physically made
- Orders will not be placed, filled, or shipped
- Showing the list of shipped products is not possible without an order being placed
- The degrees are only for Canadian schools

5.2 Projected Problems

- Organizing the degrees
- Overloading a degree table when someone makes a new degree