

**Project Design**

University of British Columbia – Okanagan Campus

COSC 304 – Introduction to Databases – Prof. Dr. Ramon Lawrence – 2018



Daniella Davis

Igat Matthew

Jonathan Gresl

Wiliam Setiawan

**Table of Contents**

[0 Introduction 3](#_Toc528538503)

[0.1 Purpose of Document (In Progress) 3](#_Toc528538504)

[0.2 Mission Statement 3](#_Toc528538505)

[0.3 Executive Summary (In Progress) 3](#_Toc528538506)

[1 Domain Assumptions 5](#_Toc528538507)

[1.1 Customer 5](#_Toc528538508)

[1.2 Products (Animals) 5](#_Toc528538509)

[1.3 Storage / Warehouses 5](#_Toc528538510)

[1.4 Shipping 5](#_Toc528538511)

[1.5 Payment 5](#_Toc528538512)

[1.6 Legal 5](#_Toc528538513)

[2 Data Model 6](#_Toc528538514)

[2.1 UML Diagram 6](#_Toc528538515)

[2.2 Entity Descriptions 7](#_Toc528538516)

[2.3 Relational Assumptions 9](#_Toc528538517)

[2.4 Relational Schema – DDL 10](#_Toc528538518)

[3 Web Interface 11](#_Toc528538519)

[3.1 Site Description 11](#_Toc528538520)

[3.2 Planned Features 11](#_Toc528538521)

[3.3 Site Map 13](#_Toc528538522)

1. Introduction
   1. **Purpose of Document** (In Progress)

The purpose of this document……

* 1. Mission Statement

Our vision is to provide people all over the world with animals from areas all over the world; to build a web-based store where people can discover any animal that they might want to buy. Users should be able to create and manage their accounts. The system should also provide administrative functions for managing warehouses, products, customers, and orders.

* 1. Executive Summary (In Progress)

ZOO BC has seen an explosion of growth over the couple of years. The website is designed specifically

for people who want to really make an impact in their pet caring skills.

// ZOOBC's website is designed to provide customers/users a way to purchase animals for their own use.

Customers are allowed to search and browse through our catalogue of animals.

A user has to login/register before adding anything to their shopping cart and proceeding to the checkout page

to confirm their order, shipping address, billing address and payment info.

The system won't save any information for guests.

When a user wants to register an account, they provide their name, email address, set up a password (etc.)

While logged in, the customer will have the option to reset/change password, change email address, see their order status

update/edit their shipping address (etc.)

The website also has its own section of logging in Admin users to update and manage/maintain the website?

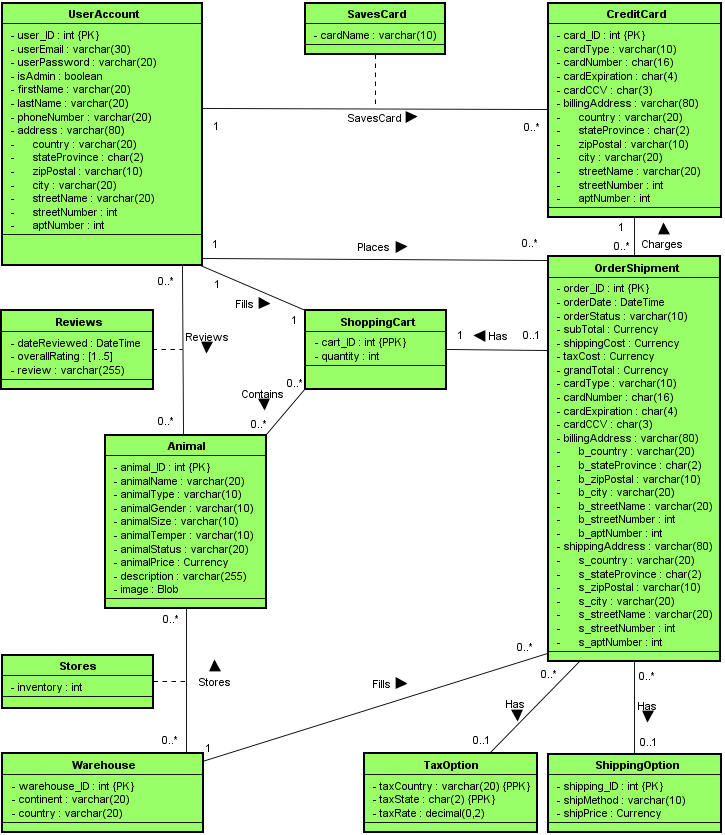
Or can users be administrators when manually set up to be one?

Choosing a product will allow users to pick quantity and animals will have a description, image and price.

1. Domain Assumptions
   1. Customer

* The customer’s intentions with the animals are not considered during sale.
* A customer will only have one shopping cart at a given time.
* Users must be logged in to an account before they are able to add animals to cart.
  1. Products (Animals)
* Animals will be the only product ever sold.
* The animals for sale are mass-produced and not unique to the buyer.
  1. Storage / Warehouses
* There will be one warehouse per continent.
* Warehouses are stocked by a 3rd party vendor unrelated to this domain.
* Each warehouse maintains a large supply of every animal offered.
  1. Shipping
* Animal pickups at warehouses is not permitted. They can be delivered only.
* The user profile address will be used as the default shipping address.
* Storage of multiple shipping addresses to user profile is not supported.
* Shipments are handled by one 3rd party shipping company for all countries.
* The health of the animals is maintained by the shipping provider.
* Ground and air transportation options are available for all animals.
* Flat shipping rate apply for all animals, regardless of size and weight.
* The animals are delivered directly to the customer’s shipping address.
  1. Payment
* All transactions will be processed using a 16-digit credit card.
* All transactions are advertised and processed using CAD currency.
* There will only be one tax rate per country and state combination.
* Returns or refunds will not be accepted.
  1. Legal
* Verification of government permits for importing or exporting animals is not required.

1. Data Model
   1. UML Diagram



* 1. Entity Descriptions

|  |  |
| --- | --- |
| **UserAccount** | |
| **Attribute** | **Description** |
| user\_ID (Primary Key) | Uniquely identifies a user. Auto-increments during account creation. |
| userEmail | Used for account access. Should match [a@a.com](mailto:a@a.com) format. |
| userPassword | Used for account access. Should be at least 5 characters long. |
| isAdmin | Boolean flag to identify administrator account type. Triggers administrator portal. |
| firstName | First name of the user. First letter should be stored as upper case. |
| lastName | Last name of the user. First letter should be stored as upper case. |
| phoneNumber | Phone number of the user. Stored as a string. Should match # (###) ### - #### format. |
| address | The address of the user. Composite key containing the sub-attributes below. |
| country | The country that the user lives in. Used as default for shipping. |
| stateProvince | The state or province that the user lives in. Used as default for shipping. |
| zipPostal | The zip or postal code for the user’s address. Used as default for shipping. |
| city | The city, town, or village that the user lives in. Used as default for shipping. |
| streetName | The name of the street that the user lives on. Used as default for shipping. |
| streetNumber | The house or building number of the user’s address. Used as default for shipping. |
| aptNumber | The suite, condo, or apartment number for the user’s address. Optional. |

|  |  |
| --- | --- |
| **Animal** | |
| **Attribute** | **Description** |
| animal\_ID (Primary Key) | Uniquely identifies an animal name (model). Auto-increments when adding to Animal. |
| animalName | The name of the animal (model). [Frog, Eagle, Piranha, Starfish, Mammoth, Chameleon] |
| animalType | Defines the type of animal. [Amphibian, Bird, Fish, Invertebrate, Mammal, Reptile] |
| animalGender | Defines the gender of the animal. [Female, Male] |
| animalSize | Defines the size of the animal. [Tiny, Small, Medium, Large, Giant] |
| animalTemper | Defines the temperament of the animal. [Domestic, Farm, Wild, Ferocious] |
| animalStatus | Defines the conservation status of the animal. [Common, Threatened, Endangered] |
| animalPrice | The pre-tax price of the animal for the customer. |
| description | A lengthy description (up to 255 characters) of the animal. |
| image | An image of the animal. Stored in database as binary large object. |

|  |  |
| --- | --- |
| **CreditCard** | |
| **Attribute** | **Description** |
| card\_ID {PK} | Uniquely identifies a credit card stored by a user. |
| cardType | Defines the type of credit card. [VISA, Mastercard, AMEX] |
| cardNumber | The credit card account number. Entered as #### #### #### #### but trimmed after. |
| cardExpiration | The expiration year and month of the credit card. Stored in MMYY format. |
| cardCCV | The **C**ard **V**erification **V**alue number found on the back of the credit card in ### format. |
| billingAddress | The credit card’s billing address. Composite key containing the sub-attributes below. |
| country | The country for the credit card’s address. Used to help determine the tax rate. |
| stateProvince | The state or province for the credit card’s address. Used to help determine the tax rate. |
| zipPostal | The zip or postal code for the credit card’s address. |
| city | The city, town, or village for the credit card’s address. |
| streetName | The name of the street for the credit card’s address. |
| streetNumber | The house or building number for the credit card’s address. |
| aptNumber | The suite, condo, or apartment number for the credit card’s address. Optional. |

|  |  |
| --- | --- |
| **Warehouse** | |
| **Attribute** | **Description** |
| warehouse\_ID (Primary Key) | Uniquely identifies a warehouse. Auto-increments when adding to Warehouse. |
| continent | The continent in which the warehouse is located. [Africa, Antarctica, Asia, Europe, North America, Central America, South America, Australia] |
| country | The country in which the warehouse is located. |

|  |  |
| --- | --- |
| **OrderShipment** | |
| **Attribute** | **Description** |
| order\_ID (Primary Key) | Uniquely identifies an order. Auto-incremented when adding to Order. |
| orderDate | The date and time that the order was placed. Determined by server at time of purchase. |
| orderStatus | The status of the order. Administrators can modify. [In Process, Shipped, Delivered] |
| subTotal | The total cost of the order before taxes and shipping costs. |
| shippingCost | The shipping cost determined by the shipping method selected by the customer. |
| taxCost | The cost of taxes for the order. Calculated by multiplying the tax rate with the sub-total. |
| grandTotal | The overall cost of the product (animal) sub-total, shipping cost, and taxes. |
| cardType | Defines the type of credit card. [VISA, Mastercard, AMEX] |
| cardNumber | The credit card account number. Entered as #### #### #### #### but trimmed after. |
| cardExpiration | The expiration year and month of the credit card. Stored in MMYY format. |
| cardCCV | The **C**ard **V**erification **V**alue number found on the back of the credit card in ### format. |
| billingAddress | The billing address for the order. **Stored at the time of purchase**. |
| b\_country | The country for the charged credit card’s address. Used to help determine the tax rate. |
| b\_stateProvince | The state of province for the charged credit card’s address. |
| b\_zipPostal | The zip or postal code for the charged credit card’s address. |
| b\_City | The city, town, or village for the charged credit card’s address. |
| b\_streetName | The name of the street for the charged credit card’s address. |
| b\_streetNumber | The house or building number for the charged credit card’s address. |
| b\_aptNumber | The suite, condo, or apartment number for the charged credit card’s address. Optional. |
| shippingAddress | The shipping address for the order. **Stored at the time of purchase**. |
| s\_country | The country for the shipping address. Used to help determine the tax rate. |
| s\_stateProvince | The state of province for the shipping address. Used to help determine the tax rate. |
| s\_zipPostal | The zip or postal code for the shipping address. |
| s\_city | The city, town, or village for the shipping address. |
| s\_streetName | The name of the street for the shipping address. |
| s\_streetNumber | The house or building number for the shipping address. |
| s\_aptNumber | The suite, condo, or apartment number for the shipping address. Optional. |

|  |  |
| --- | --- |
| **ShippingOption** | |
| **Attribute** | **Description** |
| shipping\_ID (Primary Key) | Uniquely identifies a shipping option. Auto-increments when adding to Shipping Option. |
| shipMethod | Defines the shipping option associated with the shipping price. [Ground, Air] |
| shipPrice | The price associated with the shipping method. |

|  |  |
| --- | --- |
| **TaxOption** | |
| **Attribute** | **Description** |
| taxCountry (Partial Primary Key) | Matches with the country from the order’s billing address to determine the tax rate. |
| taxState (Partial Primary Key) | Matches with the state or prov. from the order’s billing address to determine tax rate. |
| taxRate | The tax rate specified by the country and state or prov. Applied to the order’s sub-total. |

|  |  |
| --- | --- |
| **ShoppingCart** | |
| **Attribute** | **Description** |
| cart\_ID (Partial Primary Key) | Unsure about this one. |
| quantity | The quantity of a specific animal (model) added to the user’s shopping cart. |

|  |  |
| --- | --- |
| **Reviews** | |
| **Attribute** | **Description** |
| dateReviewed | The date and time that the user left a review on the animal (model) received. |
| overallRating | A rating between 1 and 5, lowest to highest of the animal (model) received. [1..5] |
| review | A descriptive review of the animal. One review per an animal, per a user - after purchase. |

|  |  |
| --- | --- |
| **Stores** | |
| **Attribute** | **Description** |
| inventory | The quantity of a specified animal (model) at a specified warehouse. |

* 1. Relational Assumptions
  2. Relational Schema – DDL

|  |  |
| --- | --- |
| CREATE TABLE UserAccount (  User\_ID INTEGER,  userEmail VARCHAR(30) NOT NULL,  userPassword VARCHAR(20) NOT NULL,  isAdmin BOOLEAN,  firstName VARCHAR(20) NOT NULL,  lastName VARCHAR(20) NOT NULL,  phoneNumber INTEGER NOT NULL,  address VARCHAR(80) NOT NULL,  country VARCHAR(20) NOT NULL,  stateProvince CHAR(2) NOT NULL,  zipPostal VARCHAR(10) NOT NULL,  city VARCHAR(20) NOT NULL,  streetName VARCHAR(20) NOT NULL,  streetNumber INTEGER NOT NULL,  aptNumber INTEGER,  PRIMARY KEY (User\_ID),  FOREIGN KEY (cart\_ID) REFERENCES ShoppingCart(cart\_ID),  UNIQUE (userEmail),  );  CREATE TABLE Animal (  animal\_ID INTEGER,  animalName VARCHAR(20) NOT NULL,  animalType VARCHAR(10) NOT NULL,  animalGender VARCHAR(10) NOT NULL,  animalSize VARCHAR(10) NOT NULL,  animalTemper VARCHAR(10) NOT NULL,  animalStatus VARCHAR(20) NOT NULL,  animalPrice CURRENCY NOT NULL,  description VARCHAR(255) NOT NULL,  image BLOB,  PRIMARY KEY (animal\_ID)  );  CREATE TABLE CreditCard (  card\_ID INTEGER,  cardType VARCHAR(10) NOT NULL,  cardNumber CHAR(16) NOT NULL,  cardExpiration CHAR(4) NOT NULL,  cardCCV CHAR(3) NOT NULL,  billingAddress VARCHAR(80) NOT NULL,  country VARCHAR(20) NOT NULL,  stateProvince CHAR(2) NOT NULL,  zipPostal VARCHAR(10) NOT NULL,  city VARCHAR(20) NOT NULL,  streetName VARCHAR(20) NOT NULL,  streetNumber INTEGER NOT NULL,  aptNumber INTEGER,  PRIMARY KEY (card\_ID)  );  CREATE TABLE Warehouse (  warehouse\_ID INTEGER,  continent VARCHAR(20),  country VARCHAR(20),  PRIMARY KEY (warehouse\_ID)  ); | CREATE TABLE OrderShipment (  order\_ID INTEGER,  orderDate DATETIME NOT NULL,  orderStatus VARCHAR(10) NOT NULL,  subTotal CURRENCY,  shippingCost CURRENCY,  taxCost CURRENCY,  grandTotal CURRENCY,  cardType VARCHAR(10) NOT NULL,  cardNumber CHAR(16) NOT NULL,  cardExpiration CHAR(4) NOT NULL,  cardCCV CHAR(3) NOT NULL,  billingAddress VARCHAR(80) NOT NULL,  b\_country VARCHAR(20) NOT NULL,  b\_stateProvince CHAR(2) NOT NULL,  b\_zipPostal VARCHAR(10) NOT NULL,  b\_city VARCHAR(20) NOT NULL,  b\_streetName VARCHAR(20) NOT NULL,  b\_streetNumber INTEGER NOT NULL,  b\_aptNumber INTEGER,  shippingAddress VARCHAR(80) NOT NULL,  s\_country VARCHAR(20) NOT NULL,  s\_stateProvince CHAR(2) NOT NULL,  s\_zipPostal VARCHAR(10) NOT NULL,  s\_city VARCHAR(20) NOT NULL,  s\_streetName VARCHAR(20) NOT NULL,  s\_streetNumber INTEGER NOT NULL,  s\_aptNumber INTEGER,  PRIMARY KEY (order\_ID)  );  CREATE TABLE ShippingOption (  shipping\_ID INTEGER,  shipMethod VARCHAR(10) NOT NULL,  shipPrice CURRENCY NOT NULL,  PRIMARY KEY (shipping\_ID)  );  CREATE TABLE TaxOption (  taxCountry VARCHAR(20),  taxState CHAR(2),  taxRate DECIMAL(0,2),  PRIMARY KEY (taxCountry, taxState)  );  CREATE TABLE ShoppingCart (  cart\_ID INTEGER,  quantity INTEGER,  PRIMARY KEY (cart\_ID)  );  CREATE TABLE Reviews (  dateReviewed DATETIME,  overallRating INTEGER [1..5],  review VARCHAR(255)  PRIMARY KEY ( , )  );  CREATE TABLE Stores (  inventory INTEGER,  PRIMARY KEY ( , )  ); |

1. Web Interface
   1. Site Description
   2. Planned Features

|  |  |  |  |
| --- | --- | --- | --- |
| **Final Overall System Features** | **Points** | **% Complete** | **Code Files** |
| **Hosting** |  |  |  |
| Running on cosc304 server | 2 |  |  |
|  |  |  |  |
| **Main/Home Page** |  |  |  |
| Search for a product by name | 1 |  |  |
| Browse products by category | 1 |  |  |
| List products (by search/browse) | 1 |  |  |
| List products with image | 2 |  |  |
| Page header with menu | 1 |  |  |
| Page header shows current logged in user | 1 |  |  |
|  |  |  |  |
| **Shopping Cart** |  |  |  |
| Add to shopping cart | 1 |  |  |
| View shopping cart | 1 |  |  |
| Update item quantity in shopping cart | 2 |  |  |
| Remove item from shopping cart | 1 |  |  |
| Improved formatting/UI (e.g. in header bar) | 1 |  |  |
| Cart stored in database between sessions | 3 |  |  |
| **Final Overall System Features** | **Points** | **% Complete** | **Code Files** |
| **Checkout** |  |  |  |
| Checkout with customer id | 1 |  |  |
| Checkout with payment/shipment info | 2 |  |  |
| Checkout with data validation | 3 |  |  |
| Inventory tracking per item | 2 |  |  |
| Inventory tracking by store/warehouse | 3 |  |  |
| Calculate taxes and shipping cost | 2 |  |  |
|  |  |  |  |
| **Product Detail Page** |  |  |  |
| Product detail page and item description | 3 |  |  |
| Product detail has an image | 1 |  |  |
| Product detail has an image from database | 4 |  |  |
|  |  |  |  |
| **User Accounts and Login** |  |  |  |
| Create user account page | 3 |  |  |
| Create user account with data validation | 3 |  |  |
| Login/logout | 2 |  |  |
| Page listing all orders for user | 1 |  |  |
|  |  |  |  |
| **Product Reviews** |  |  |  |
| Ability to enter a review on a product | 2 |  |  |
| Display product review on product detail | 1 |  |  |
| Restrict to one review per user on item purchased | 2 |  |  |
|  |  |  |  |
| **Administrator Portal** |  |  |  |
| Secured by login | 3 |  |  |
| List all customers | 1 |  |  |
| List report showing total sales/orders | 1 |  |  |
| Report with a graph | 3 |  |  |
| Add new product | 2 |  |  |
| Update/delete product | 2 |  |  |
| Change order status/ship order | 1 |  |  |
| Upload a photo to database for product | 4 |  |  |
| Database restore with SQL script | 2 |  |  |
| Add/update warehouse, customer | 2 |  |  |
|  |  |  |  |
| **Database System/General** |  |  |  |
| Implement some validation using triggers | 2 |  |  |
| Use AJAX for some pages | 4 |  |  |
| Bonus – User interface and navigation/usability |  |  |  |
| Bonus – User can save multiple credit cards |  |  |  |

* 1. Site Map

