CS 405G Introduction to Database Systems

Fall 2013

Project Report

By

Feiyu Shi & Libby Ferland

12/1/2013

1. Database Design
2. ER Diagram

See Appendix.

1. Relational Schema Design, Functional Dependencies, and Normalization

**Entities:**

**Customer**(string:Email, string:Fname, string:LName, string:Password)

Email -> Fname, LName, Password

3NF

**Basket**(string:CEmail, int:BasketId, date: ShopDate)

CEmail, BasketId -> ShopDate

3NF

**Orders**(int:POrderID, string:Status)

POrderID -> Status

3NF

**Employee**(int:EId, string:EPassword, string:LastName, string:FirstName, Boolean:IsManager)

EId -> EPasword, LastName, FirstName, IsManager

3NF

**Item**(int:IId, string:IName, string:Category, string:Description, float:IPrice, int:Quantity, float:PromoPrice)

IId -> IName, Category, Description, IPrice, Quantity, PromoPrice

3NF

**ShipPrice**(string:ShipMethod, float:ShipRate)

ShipMethod -> ShipRate

3NF

**AddressBook**(int: AddrIndex, string: CEmail, string:AddrLine1, string:AddrLine2, string:City, string:State, int:Zip, Boolean: IsVisible)

AddrIndex, CEmail -> AddrLine1, AddrLine2, City, State, Zip, IsVisible

3NF

**PaymentMethods**(string:CEmail,int:CardNo,string:CHolderLastName,string:CHolderFirstName,date:CExpirDate, Boolean:IsVisible)

CEmail, CardNo -> CHolderLastName, CHolderFirstName, CExpirDate, IsVisible

3NF

**Relationships:**

**Purchase**(string:CEmail, int:InvoiceNo, int:PurchaseRating, string:Review, date:PurchaseDate)

CEmail, InvoiceNo -> PurchaseRating, Review, PurchaseDate

3NF

**Ship**(int:EId, int: OrderID, date:ShipDate)

EId, OrderID -> ShipDate

3NF

**Promote**(int:EId, int:IId, float:PromoRate, string:PStartDate, string:PEndDate)

EId, IId -> PromoRate, PStartDate, PEndDate

3NF

**BasketContains**(string:CEmail,int:BasketId, int:IId, int:BQuantity)

CEmail, BasktId, IId -> BQuantity

3NF

**OrderContains**(int: COrderID, int:IId, int:OQuantity)

COrderID, IId -> OQuantity

3NF

**BillingAddress**(string:CEmail, int:CardNo, int: AddrIndex, Boolean: IsVisible)

CEmail, CardNo, AddrIndex -> Isvisible

3NF

**ShippedTo**(int:OderID, string:CEmail, int:AddrIndex)

All attributes are primary keys, and they don’t depend on each other.

**PaidWith**(string:CEmail,int:CardNo, int:OrderID)

All attributes are primary keys, and they don’t depend on each other.

**ShippedBy**(int:OderID,string:ShipMethod)

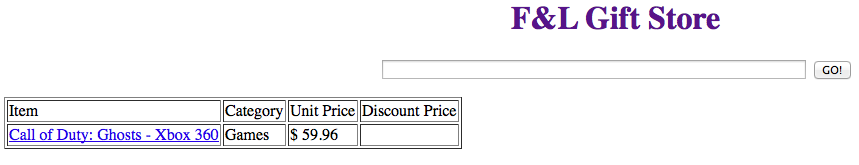
All attributes are primary keys, and they don’t depend on each other.

1. Description of Programs
2. Program Functions
3. Search Engine

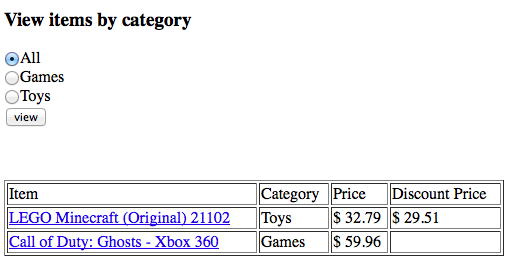
Input form:



Sample output:

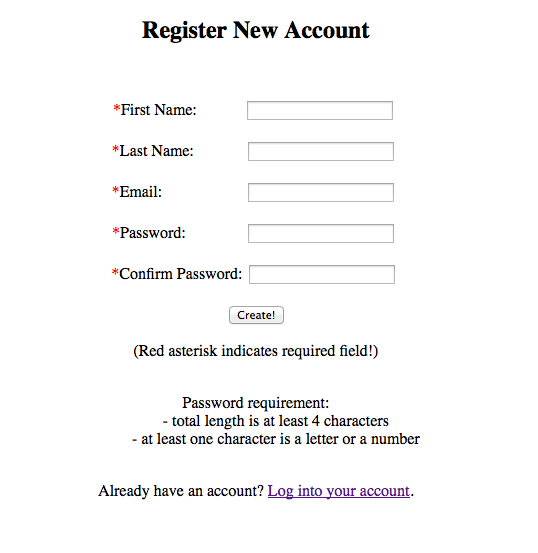


1. View by Category

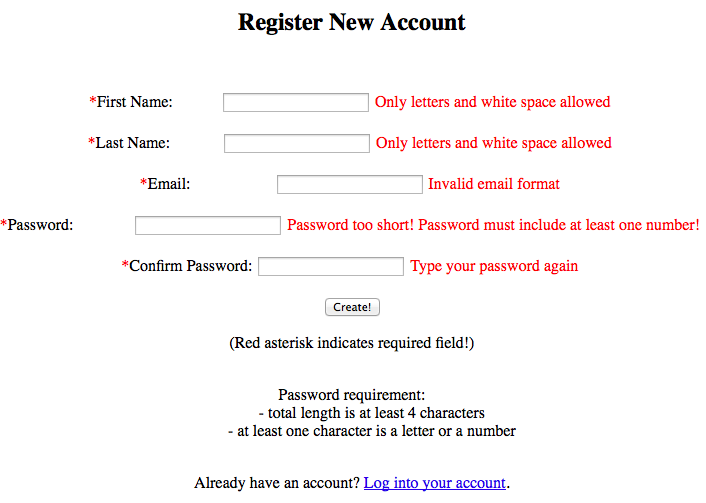


1. Customer Registration

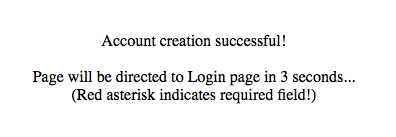
Input form:



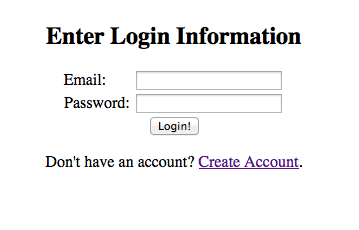
Output form 1 (errors):



Output form 2:

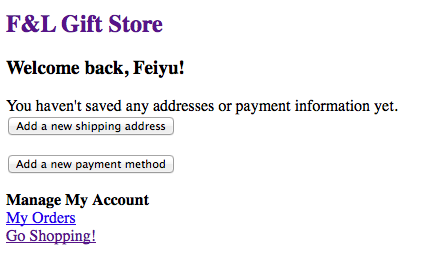


1. Customer Login

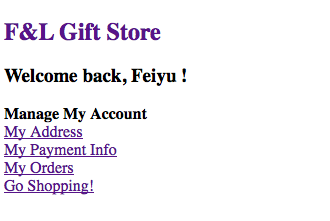


1. Customer Account

Page of new customer:

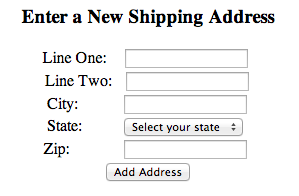


Page of old customer:



1. Address Book

Input form 1:



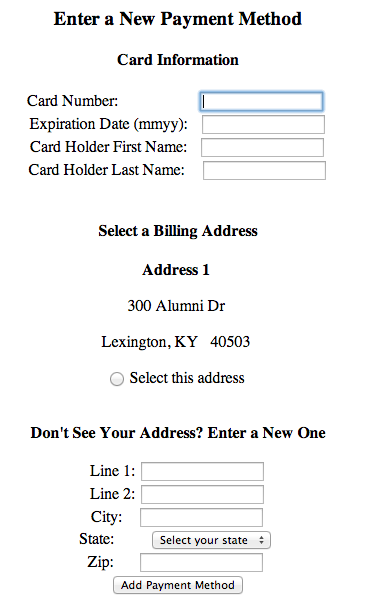
Input form 2:



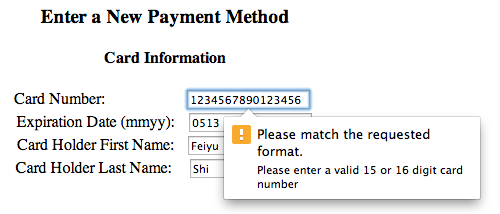


1. Payment Method

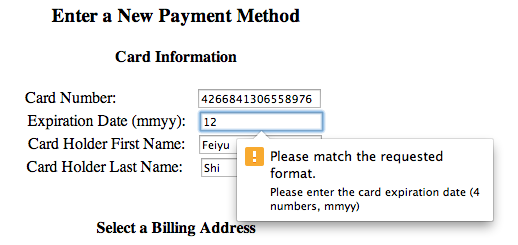
Input form 1:



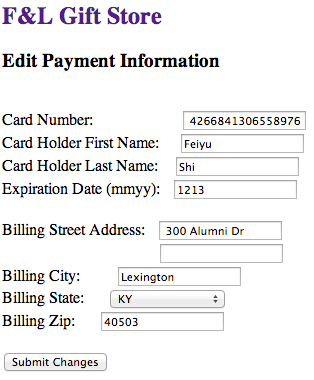
Input form 2 (error):



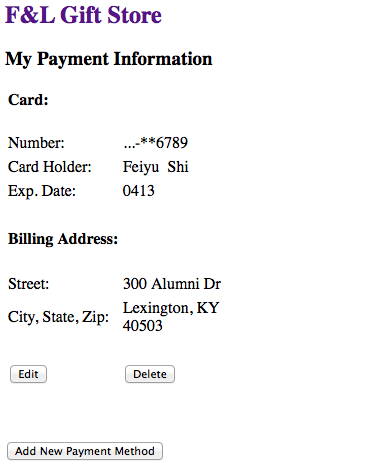
Input form 3 (error):



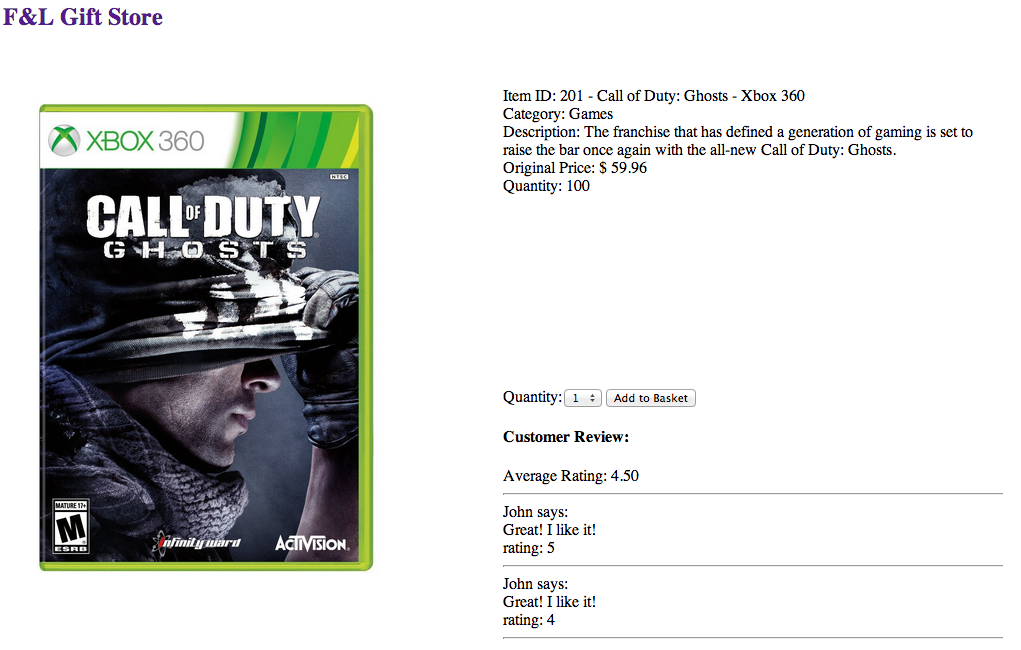
Input form 4:



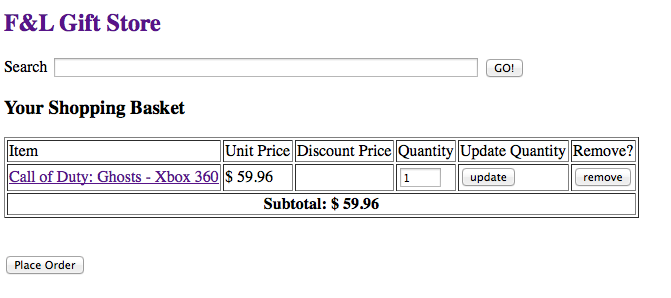
Output form:



1. Item Page

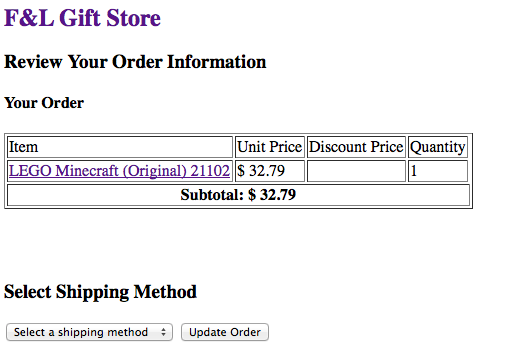


1. Customer Basket

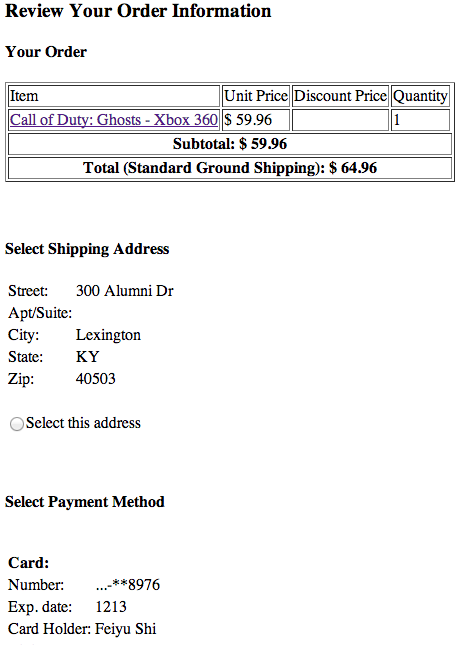


1. Order Placement

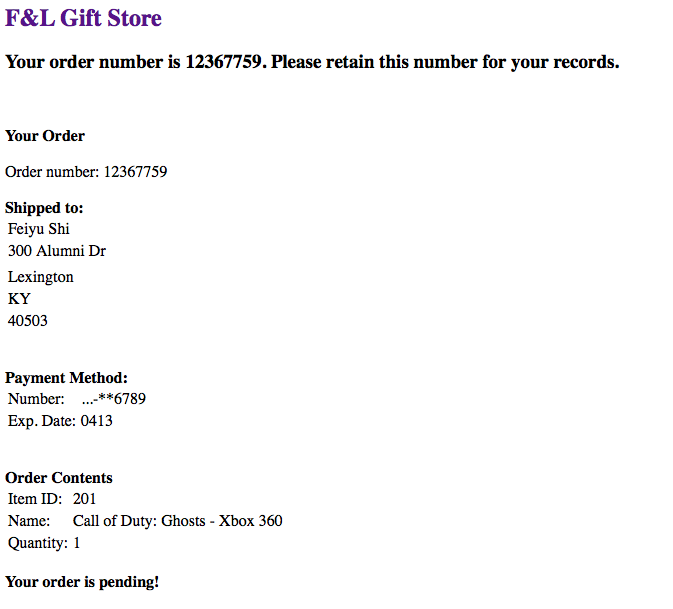
Output form 1:



Output form 2:

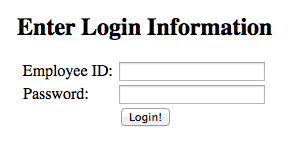


Output form 3:



1. Employee Login

Input form:

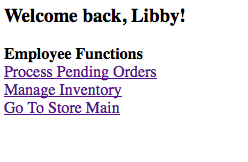


1. Employee Home

Output form 1:



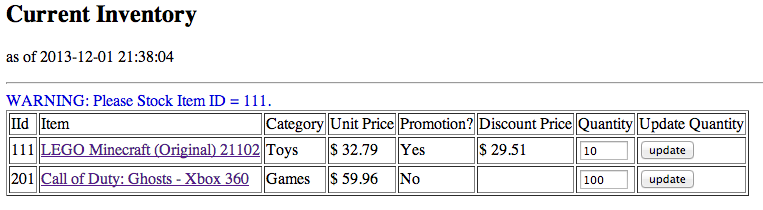
Output form2:



1. Pending Orders

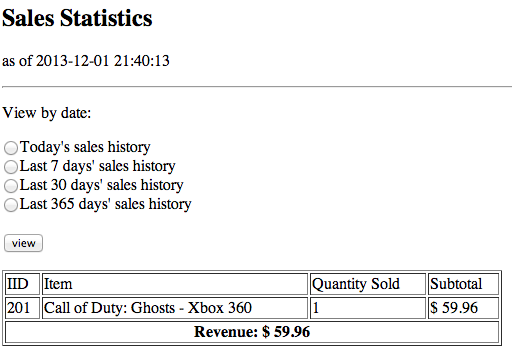


1. Inventory Management

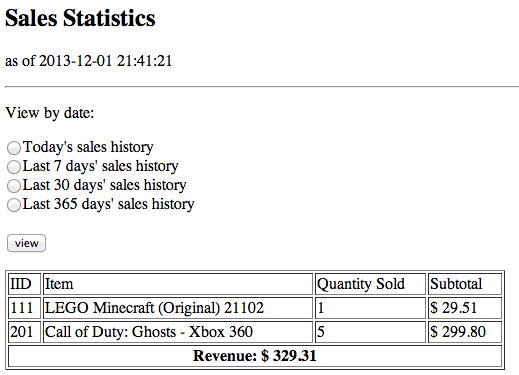


1. Manager Statistics

Output form 1 (sales, today):

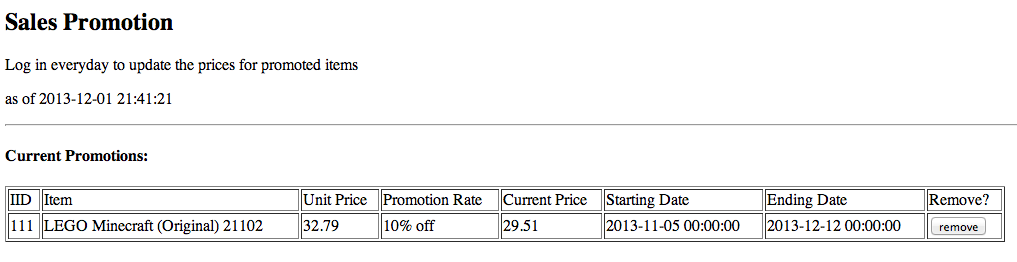


Output form 2 (sales, this year):

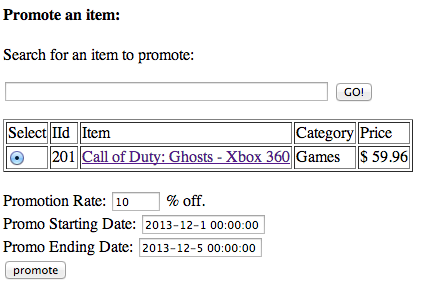


1. Manager Promotion

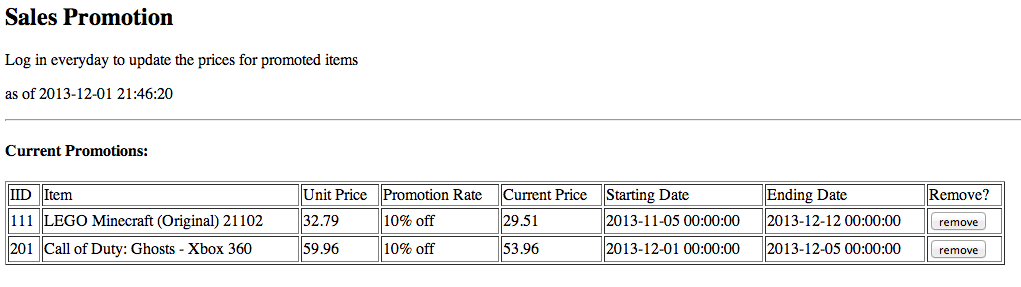
Output form 1:



Output form 2:



Output form 3:



1. Testing
2. Testing Process
3. Customer Account Registration

We test it by feeding in incomplete contents or contents with wrong format, and then all registration are blocked. If successful, we can observe the customer information in the database.

1. Customer Address Book and Payment Method

We test it by adding address and payment method and check them in the database. We can also edit that information and “delete” it. We actually can’t delete address and payment method from the database, because of the foreign key constraints, but we can prevent customer from seeing it in their account page.

1. Customer Basket

We add items from item page, and then check the basket. If it’s all right, it will display the item with correct quantity. We edit the quantity, if it’s negative or over the amount of stock, it will show error information. We can also remove the item. All corresponding information can be check in the database.

1. Customer Placing Order and Order History

When click place order button, we can first update shipping method and then select shipping address and payment method from address book and payment method storage. After all that, we can actually place the order and review the order information on the confirmation page. Order data could be seen in the database, customer order history page and the pending order page in the employee panel.

1. Employee Pending Orders

Employee should see all the pending orders. When click the ship button, corresponding inventory is updated and the order status is changed in customer order history.

1. Project Experience