Project Proposal
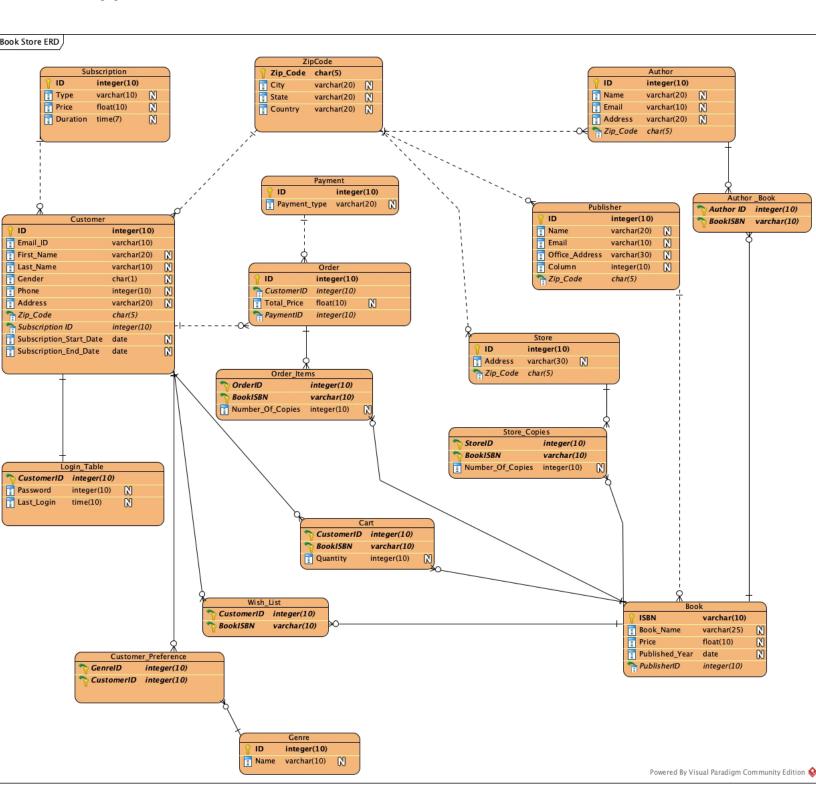
System Description

We propose to develop a Book Store application which lets the customers to search, view and buy different books of different genres with just few clicks from any device which has the browser and the Internet access. The customers of the Book Store can either get the books delivered to their homes or avail of the facility of in-store pickup. This system will help them place an order for their books according to their likings. Each customer would log in/ signup through their email address. Besides being the ecommerce application for the customers, it also enables the admin users of the Book Store to manage the books by adding, updating and deleting information related to the Books. This application would make use of all the CRUD operations, Views, Stored Procedures, Functions and Triggers to serve as one-stop shop for end users and effective management system for business users.

Main Features of the Application

- A new user will be able to sign up / An existing user can log in using his/her email id and password.
- A new user can subscribe to one of the 4 plans (basic, silver, gold, platinum)
- An existing user can upgrade his/her subscription plan by paying the difference.
- Depending on the subscription plan the customer can get discounts and faster delivery.
- Users can either choose the option of delivering books at home or in-store pickup
- Users will get an alert when the books on his/her wish list are back in stock.
- Users can search for books using filters on:
    - Book Title
    - Genre
    - Price Range
- Users can place an order for multiple books and specify the quantity for each book.
- Users can add more than one order.
- Users can view placed orders.
- Users can update his / her address, zip code, and email id.
- Admins can manage books related information by Insert, Modify and Delete.

Initial ERD

Book Store ERD

Subscription

🔑 ID	integer(10)	
Type	varchar(10)	N
Price	float(10)	N
Duration	time(7)	N

ZipCode

🔑 Zip_Code	char(5)	
City	varchar(20)	N
State	varchar(20)	N
Country	varchar(20)	N

Author

🔑 ID	integer(10)	
Name	varchar(20)	N
Email	varchar(10)	N
Address	varchar(20)	N
Zip_Code	char(5)	

Payment

🔑 ID	integer(10)	
Payment_type	varchar(20)	N

Publisher

🔑 ID	integer(10)	
Name	varchar(20)	N
Email	varchar(10)	N
Office_Address	varchar(30)	N
Column	integer(10)	N
Zip_Code	char(5)	

Author_Book

Author ID	integer(10)
BookISBN	varchar(10)

Customer

🔑 ID	integer(10)	
Email_ID	varchar(10)	
First_Name	varchar(20)	N
Last_Name	varchar(10)	N
Gender	char(1)	N
Phone	integer(10)	N
Address	varchar(20)	N
Zip_Code	char(5)	
Subscription ID	integer(10)	
Subscription_Start_Date	date	N
Subscription_End_Date	date	N

Order

🔑 ID	integer(10)	
CustomerID	integer(10)	
Total_Price	float(10)	N
PaymentID	integer(10)	

Store

🔑 ID	integer(10)	
Address	varchar(30)	N
Zip_Code	char(5)	

Order_Items

OrderID	integer(10)	
BookISBN	varchar(10)	
Number_Of_Copies	integer(10)	N

Store_Copies

StoreID	integer(10)	
BookISBN	varchar(10)	
Number_Of_Copies	integer(10)	N

Login_Table

CustomerID	integer(10)	
Password	integer(10)	N
Last_Login	time(10)	N

Cart

CustomerID	integer(10)	
BookISBN	varchar(10)	
Quantity	integer(10)	N

Wish_List

CustomerID	integer(10)
BookISBN	varchar(10)

Book

🔑 ISBN	varchar(10)	
Book_Name	varchar(25)	N
Price	float(10)	N
Published_Year	date	N
PublisherID	integer(10)	

Customer_Preference

GenreID	integer(10)
CustomerID	integer(10)

Genre

🔑 ID	integer(10)	
Name	varchar(10)	N

Data Sources:

- Books data has been pulled from http://www2.informatik.uni-freiburg.de/~cziegler/BX/ (The Institute for Informatik Freiburg). We obtained ISBN, Book Title, Authors, Publisher name, Year of Publication, and book image URLs from this dataset.
- Genre information for each book is extracted from https://www.goodreads.com/ by performing the web scraping using the *Beautiful Soup* package in Python.
- For customer and store information we got the dataset from Kaggle and mockaroo.

After fetching all the raw data from multiple sources, we did the Data Processing part, where we removed the duplicate books information and the books which has incorrect information in authors column. There were some books with more than one author, so we broke it down into individual tuples with different author names and other fields being same to maintain atomic value property. In genres table, each tuple was split into numerous rows to be in one normal form because each book has multiple categories.

Libraries/Platforms:

For our project we are using *Beautiful Soup* to for web scrapping the data from different websites and we are using *pandas* and *numpy* for loading and cleaning/text processing of the data. We would be using mysqlclient to connect MySQL database with python.

Project Participants:

All the project participants would work in all the layers of this full stack application development i.e.., Database design, backend development and frontend design. We separated the whole application into three modules and assigned to each participant to develop database, backend and frontend for each module.

Module	Assigned To
User Module	Ajay Karthick Senthil Kumar
Book Module	Nikhil Narayanarao Kudupudi
Order Module	Shivani Ashish Mundle

Each participant will do the below steps for the assigned module.

- Create DB elements such as Tables, Views, Stored Procedures, Functions, Triggers pertaining to the corresponding module.
- Collect, preprocess data and insert into the corresponding tables.
- Develop Backend endpoints for all the functionalities applicable to the assigned module which uses CRUD operations, stored procedures, functions, and triggers.
- Design UI elements related to the assigned module.

Participants will collaborate with each other if there are any dependencies across the modules.

User Module (Ajay Karthick Senthil Kumar)

Assignee will create Database elements, backend functionalities, and UI elements for the below submodules.

- Two different types of users – Admin and Customer. (Each user will have different levels of access and use cases)
- Addition of new customer through registration.
- Authorization while logging in.
- Admin can insert, update, and delete customer information.
- Customers can update their personal information and delete their account.
- Customers can insert information to Preferences table based on genre.
- Customers can add, update and delete his/her subscription.
- Customers can insert books to their Wishlist.
- Save login records of the customers and admins to track the records.

Book Module (Nikhil Narayanarao Kudupudi)

Assignee will create DB elements, backend functionalities, and UI elements for the below submodules.

- Insert, Update, and Delete the Book information can be performed by admin user.
- Insert / Update the availability of book copies in particular store can be done by admin.
- Availability of books - Number of copies available in each stores needs to be tracked based on the customer's location.
- Fetching all the books information from database and displaying it for the customers.
- Searching and displaying a particular book detail based on book title for the customers.
- Filtering and displaying books based on genre for the customers
- Filtering and displaying books based on price range for the customers.
- Filtering and displaying books based on author for the customers.

Order Module (Shivani Ashish Mundle)

Assignee will create DB elements, backend functionalities, and UI elements for the below submodules.

- Insert, Update and Delete information to cart by customer.
- Quantity of the items to purchase can be modified by the customer.
- Insert information to orders table once customer purchase it.
- Fetch and display the list of items purchased by the customer so far.
- Filtering and displaying the orders information based on the date to the customer.
- Reduce the number of copies available in each store needs to be updated whenever the customer makes the purchases.
- Delivery date needs to be calculated based on the customer's subscription.
- Subscription end date needs to be updated as per their plan.
- Payment information can be updated by the customer.