TECHNOLOGY

Caltech | Center for Technology & Management Education

Live Session 03: End User Web Pages DB Structure for End User

TECHNOLOGY

Caltech | Center for Technology & Management Education

Develop the Web Pages in Angular for End User Web App

You Already Know

Before we begin, lets recall what we have covered till now:

- Angular

- Cucumber

Caltech | Center for Technology & Management Education

Project Planning with Agile

- Developed User Stories under the Epic Admin and End User

- Managed project using JIRA by planning the Sprints

Develop Angular Project Structures

- Created project structure for Admin and end users

- Explored Angular CLI commands

- Generated components and services for projects

Sync the Projects with Git on Github

- Pushed the Angular Projects on github

A Day in the Life of a Full Stack Developer

As a full stack web developer, our key role is to develop both client and server software.

Angular and Node can be used to build front end of the web page.

Spring Boot, Java, and MySQL/MongoDB can be used to build at the back end.

A Day in the Life of a Full Stack Developer

- Now, Bob needs to create a web page and DB for the end-user web wpp to further proceed with his project.

Let me use previously used technologies to created web page and DB for End user Web App.

In this lesson, we will learn the skills of HTML and CSS to design and develop web pages in previously created Angular Project and User SQL Commands to create tables in DB for End-User Web App and help Bob complete his task effectively and quickly.

Learning Objectives

By the end of this lesson, you will be able to:

- Develop the web page templates for end-user web app

- Develop the CSS for styling the web pages and configure routing

- Use SQL commands create tables for end-user web app

- Create primary and foreign key relations within tables

Caltech | Center for Technology & Management Education

Angular Component Templates for End User

Caltech | Center for Technology & Management Education

Web App for End User

In this project, various modules will be managed, such as:

Listing the products for end-user

Managing cart

Managing wishlist and orders
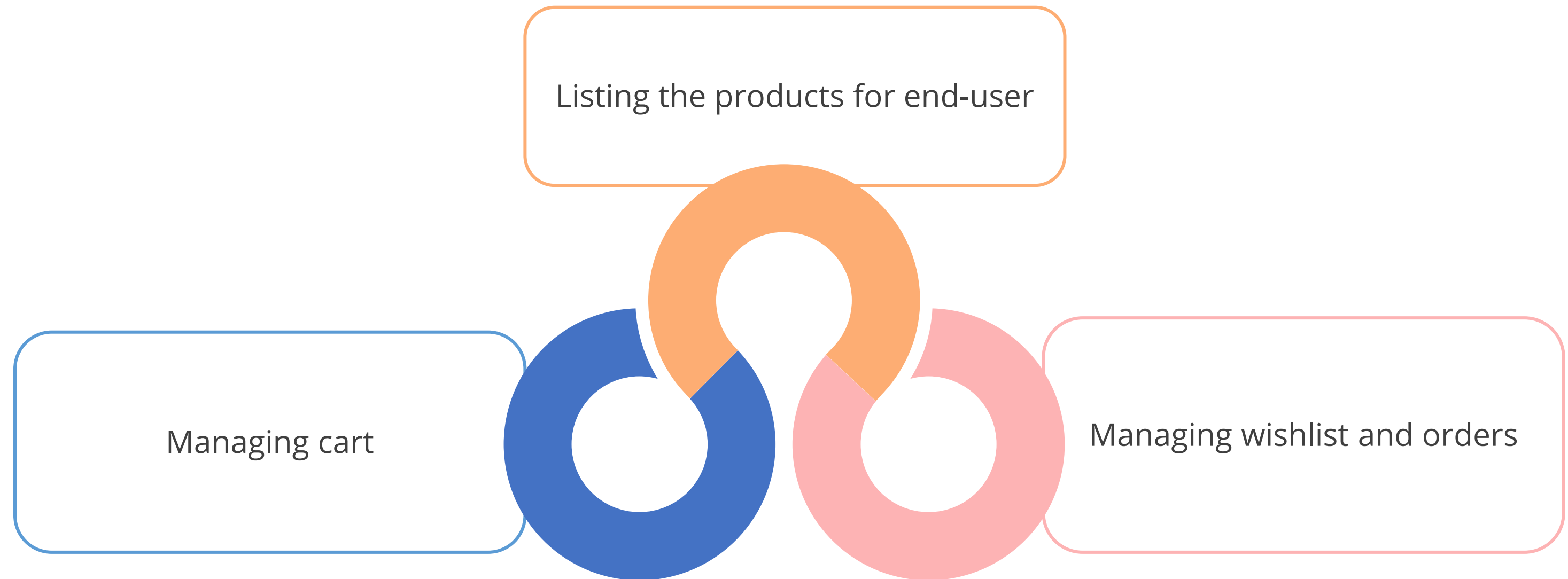
Web App for End User

For basic operations, the web page templates are developed using:

Web Page for Authentication Login Component

Develop a login page for end user to sign in and enter into the home page.

E-COMMERCE STORE

Login

Email

josh@example.com

Password

••••••••••
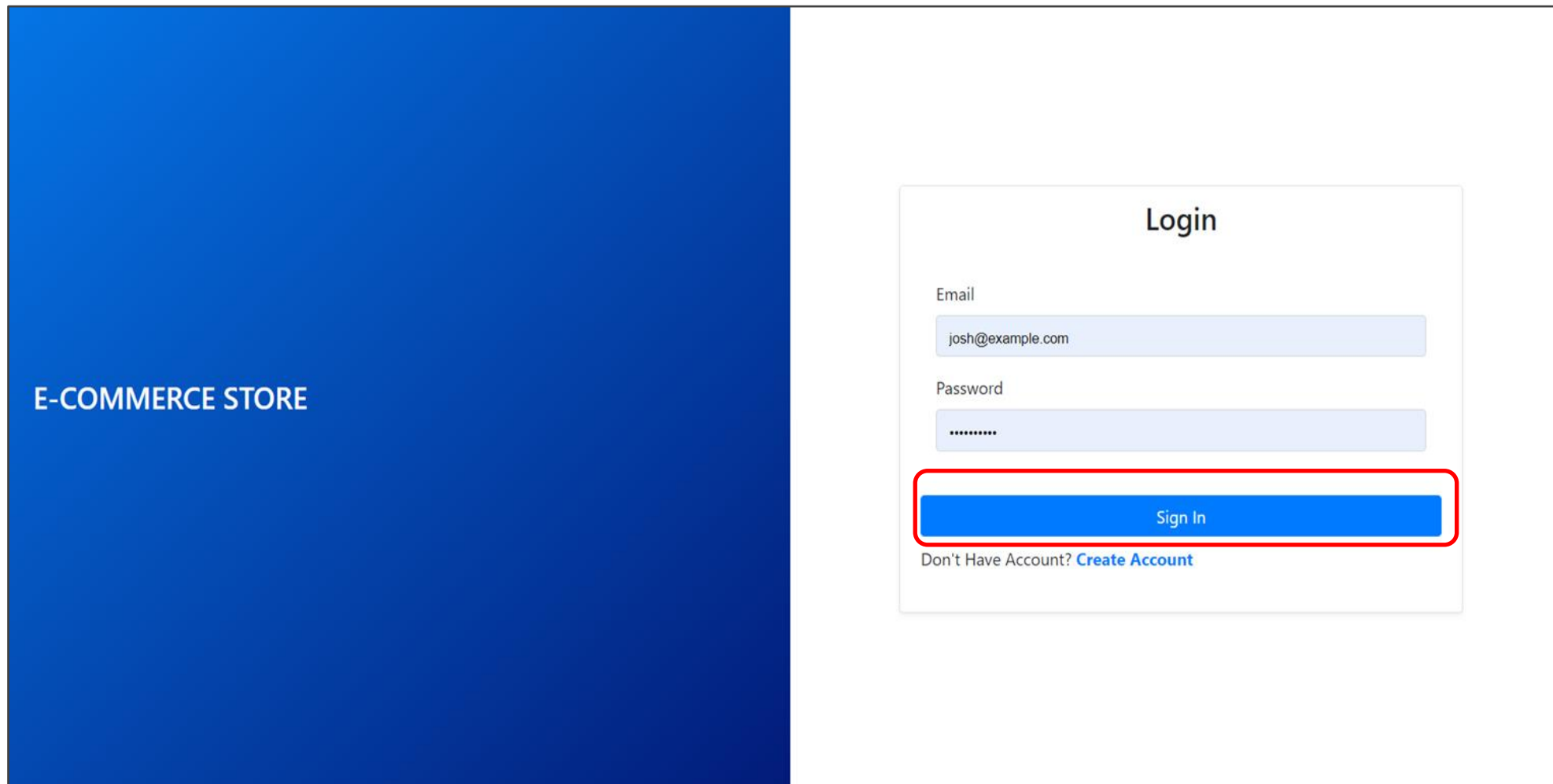
Sign In

Don't Have Account? **Create Account**

Login page

Web Page for Authentication Login Component

Design the login page with an email, a password, and a login button.

Caltech | Center for Technology & Management Education

Web Page for Authentication Login Component

Click on the login button, validate the user, and navigate to the home page which will have a navigation bar with various options.

Web Page for Authentication Login Component

Various files from the given directory: src/app/pages/login/

login.component.css

CSS goes here to design the page and forms.

login.component.html

HTML code is written in this template file.

login.component.ts

Logic will be written in typescript file.

Caltech | Center for Technology & Management Education

Web Page for User Registration Component

A new user needs to register.

E-COMMERCE STORE

Register

Name

eg: John Peterson

Email

eg: abc@example.com

Password

****

Sign Up

Already Have Account? **Sign In**

Caltech | Center for Technology & Management Education

Web Page for User Registration Component

Design a register component that can take Name, Email, and Password as input to register the user on the estore web app.

Web Page for User Registration Component

After registration, navigate to the home page, which will have a navigation bar with various options.

Web Page for User Registration Component

In the directory: src/app/pages/register/

register.component.css

CSS goes here to design the page and forms.

register.component.html

HTML code is written in this template file.

register.component.ts

Logic will be written in typescript file.
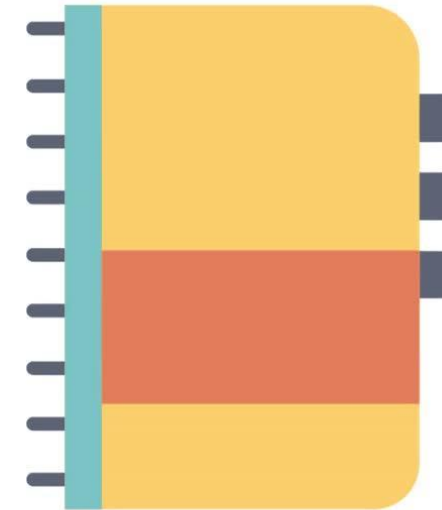
Web Page for Profile Component

Users can manage their profile by navigating to the profile component on the navigation bar.

Web Page for Profile Component

The web page will have details of the users who can manage their contact information, that is, the address for delivering the order.

Web Page for Profile Component

In the directory: src/app/pages/user-profile/

user-profile.component.css

CSS goes here to design the page and forms.

user-profile.component.html

HTML code is written in this template file.

user-profile.component.ts

Logic will be written in typescript file.

Web Page for Products Component

UI is developed to show the various products added by the admin.

Web Page for Products Component

All the products are listed along with the options to buy the product, add to cart, or add to wishlist.

Web Page for Products Component

Search, sort, and filter operations can also be implemented on the same web page.
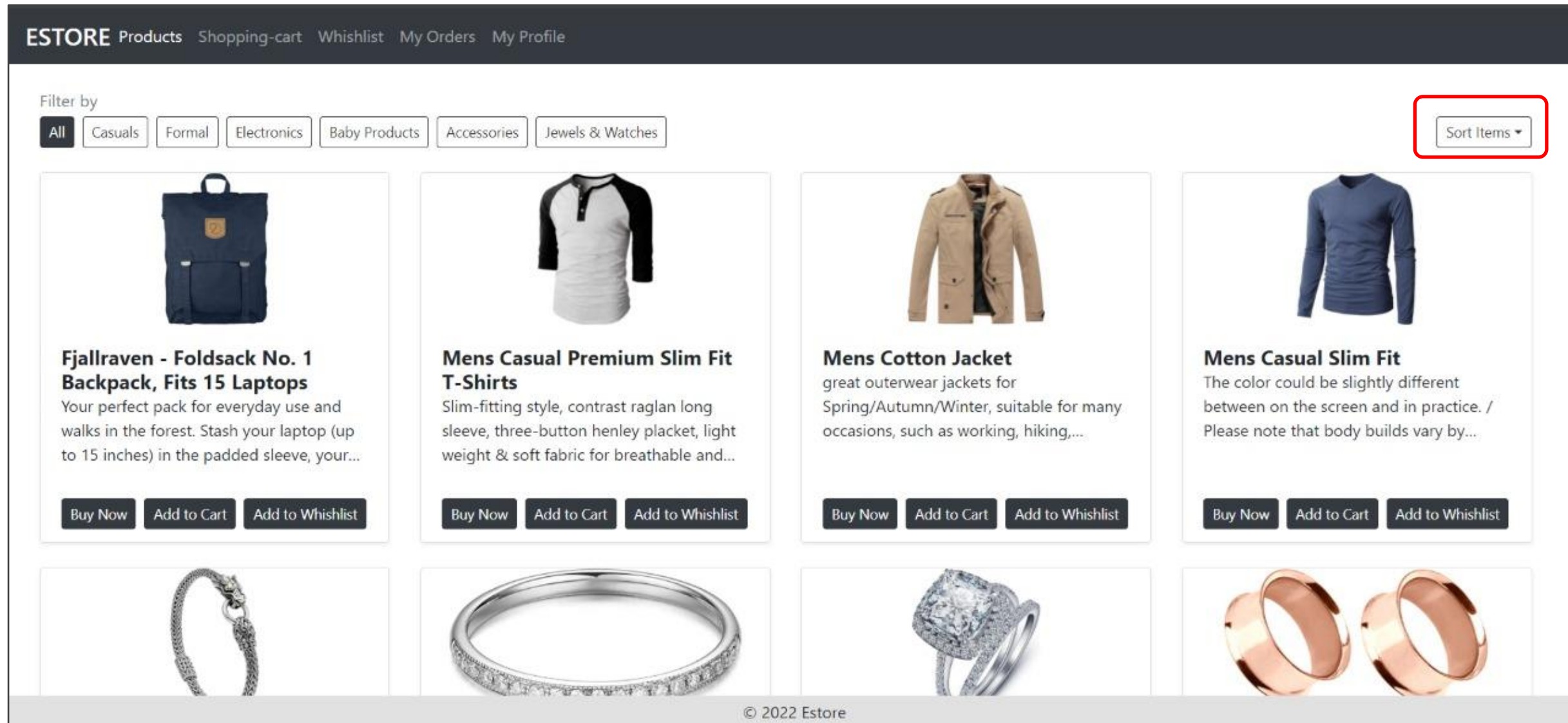
Web Page for Products Component

In the directory: src/app/pages/products/

products.component.css

CSS goes here to design the page and forms.

products.component.html

HTML code is written in this template file.

products.component.ts

Logic will be written in typescript file.

Web Page for Wishlist Component

Users can mark any product as the favourite by moving it to the wishlist.

Web Page for Wishlist Component

The Wishlist component shows the list of products that a user may buy in future.

Web Page for Wishlist Component

There is a button to move the product to the cart by clicking on Add to Cart near the product.

Web Page for Wishlist Component

In the directory: src/app/pages/wishlist/

wishlist.component.css

CSS goes here to design the page and forms.

wishlist.component.html

HTML code is written in this template file.

wishlist.component.ts

Logic will be written in typescript file.

Web Page for Cart Component

The navigation bar is used to navigate to the cart tab where a user can see a list of the products along with quantity and price for the product.

Web Page for Cart Component

Summary of the products in order to be placed shall be shown along with the option to buy i.e. to place an order.

Web Page for Cart Component

Payment's method page can also be implemented.

Web Page for Cart Component

In the directory: src/app/pages/shopping-cart/

shopping-cart.component.css

CSS goes here to design the page and forms.

shopping-cart.component.html

HTML code is written in this template file.

Shopping-cart.component.ts

Logic will be written in typescript file.

Caltech | Center for Technology & Management Education

Web Page for Checkout Component

From the cart, users will be navigated to the final checkout page.

Caltech | Center for Technology & Management Education

Web Page for Checkout Component

Users must choose the payment method and address for confirmation.

Web Page for Checkout Component

Finally, users must click on the place order button, and a successful order creation message will be shown.

Web Page for Checkout Component

In the directory: src/app/pages/checkout/

checkout.component.css

CSS goes here to design the page and forms.

checkout.component.html

HTML code is written in this template file.

checkout.component.ts

Logic will be written in typescript file.

Caltech | Center for Technology & Management Education

Web Page for Orders Component : View Orders

As the User will place an order from the account, the list of orders, along with the order status, will be displayed to the users for reference.

ESTORE Products Shopping-cart Whishlist **My Orders** My Profile

Order Placed	Total	Ship To	
03 December 2021	**₹839.60**	**John Watson**	

Order ID: **#3456789345**
View Order Details

John Hardy Women's Legends Naga Gold & Silver Dragon Station Chain Bracelet
From our Legends Collection, the Naga was inspired by the mythical water dragon that protects the ocean's pearl. Wear facing inward to be bestowed with love and abundance, or outward for protection.
₹695.00

Mens Casual Premium Slim Fit T-Shirts
Slim-fitting style, contrast raglan long sleeve, three-button henley placket, light weight & soft fabric for breathable and comfortable wearing. And Solid stitched shirts with round neck made for durability and a great fit for casual fashion wear and diehard baseball fans. The Henley style round neckline includes a three-button placket.
₹22.30

© 2022 Estore

Caltech | Center for Technology & Management Education

Web Page for Orders Component : View Orders

The detailed model can be opened on click of view order details in future as in when connected to the backend.

Web Page for Orders Component : View Orders

In the directory: src/app/pages/orders/

order.component.css

CSS goes here to design the page and forms.

order.component.html

HTML code is written in this template file.

order.component.ts

Logic will be written in typescript file.

BUY

TECHNOLOGY

Caltech | Center for Technology & Management Education

DB Structure for End User

Creating Database

Creating DataBase in MySQL

MySQL CLI usage to create and work with the database.

Use database ecommerce
Selects current working database

02

Create database ecommerce
Creates the database for ecommerce

01

Show tables
Lists all the tables in the database

03
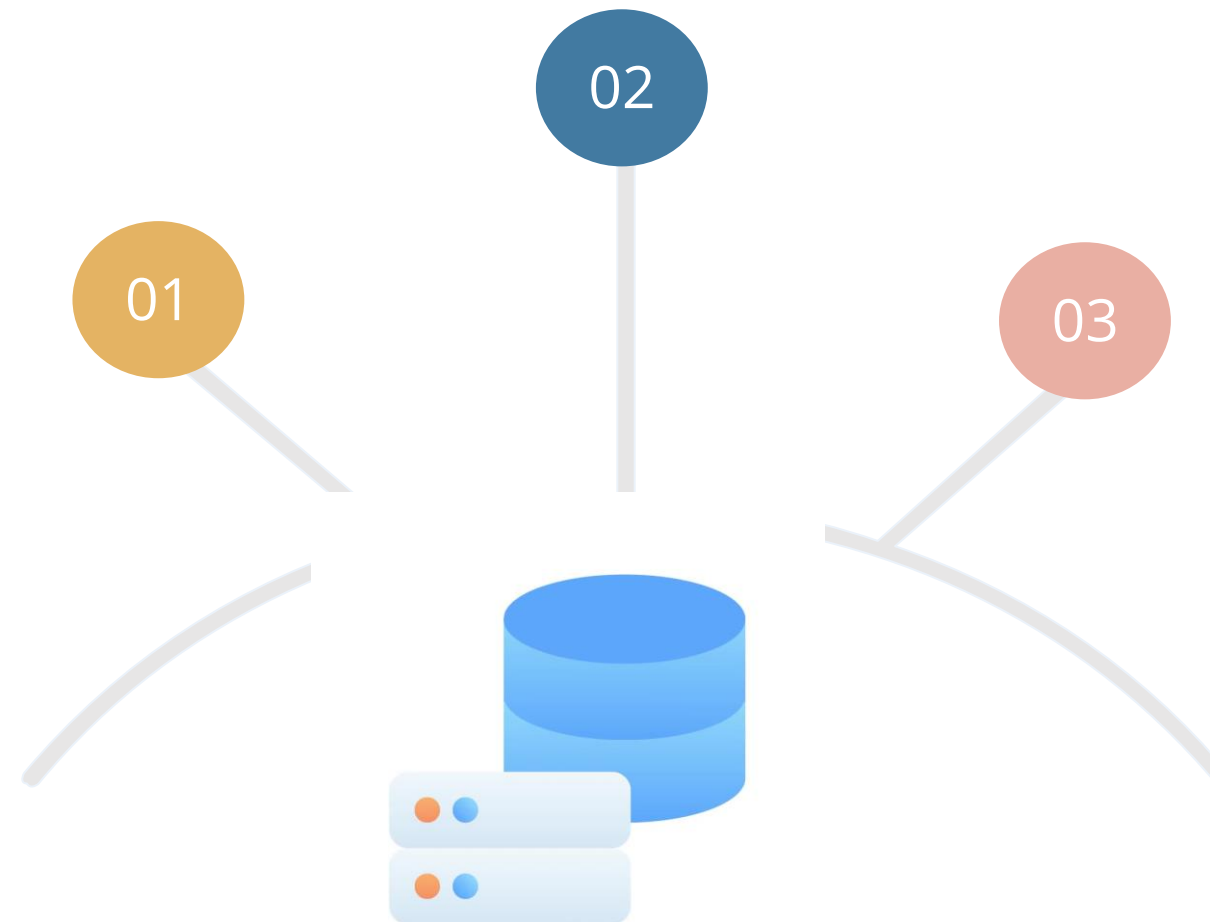
Creating Tables

Creating Table

Many tables are common for usage. Example: the Order Table in Admin will be used by the Web App for the End User as well.

So, tables required for the cart and the wishlist will only be created.

Rest all tables are already created during the Admin Table Structure.

Generating User Table for Login

Make sure the Database is selected as e-commerce or execute the command to make the DB selection for e-commerce as:

Use database ecommerce:

```sql
CREATE TABLE USERS(

    userId              INTEGER NOT NULL PRIMARY KEY AUTO_INCREMENT,

    email               VARCHAR(50) NOT NULL,

    password            VARCHAR(50) NOT NULL,

    fullName            VARCHAR(255) NOT NULL,

    street              VARCHAR(50) DEFAULT NULL,

    city                VARCHAR(50) DEFAULT NULL,

    state               VARCHAR(50) DEFAULT NULL,

    country             VARCHAR(50) DEFAULT NULL,

    pincode             INTEGER,

    image               VARCHAR(1000),

    contact             BIGINT,

    addedOn             DATETIME DEFAULT CURRENT_TIMESTAMP
);

) NOT NULL,
```

Generating User Table for Login

For simplicity, one address per user is considered.

In case a user has many addresses, an address table is created separately with a primary and foreign key relationship.
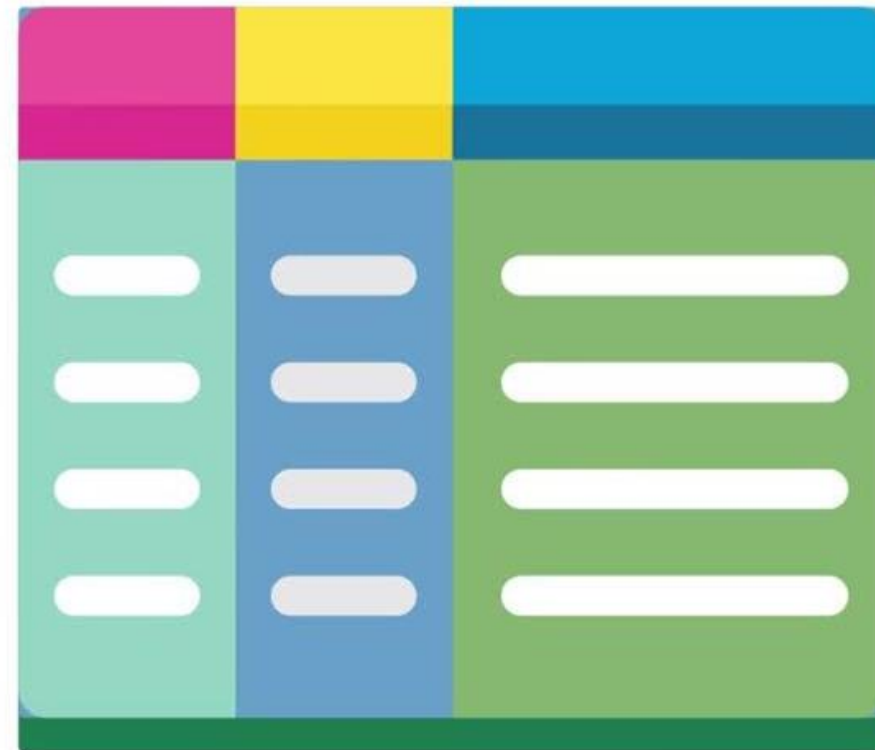
Generating Wishlist Table

Make sure the database is selected as ecommerce, or you can execute the command to make the db selection for ecommerce as :

Use database ecommerce:

```
CREATE TABLE WISHLIST {

  wishListId          INTEGER NOT NULL PRIMARY KEY AUTO_INCREMENT,

  productId           INTEGER NOT NULL,

  userId              INTEGER NOT NULL,

  FOREIGN KEY (productId) REFERENCES PRODUCTS(productId),

  FOREIGN KEY (userId) REFERENCES USERS(userId)

}
```

Generating Cart Table

Make sure the database is selected as eCommerce, or you can execute the command to make the db selection for eCommerce as:

Use database ecommerce:

```
CREATE TABLE CART {

  cartId            INTEGER NOT NULL PRIMARY KEY AUTO_INCREMENT,

  productId         INTEGER NOT NULL,

  qunatity          INTEGER DEFAULT 1,

  userId            INTEGER NOT NULL,

  FOREIGN KEY (productId) REFERENCES PRODUCTS(productId),

  FOREIGN KEY (userId) REFERENCES USERS(userId)

}
```

Key Takeaways

○ For basic operations, the web page templates are developed using HTML and CSS.

○ Users can manage their profile by navigating to the profile component on navigation bar.

○ UI is developed to show the various products added by the admin.

○ The Wishlist component shows the list of products which a user may buy in the future.

Key Takeaways

◉ The navigation bar is used to navigate to the cart tab where users can see a list of the products along with quantity and price for the product

◉ MySQL CLI is used to create and work with a database.

◉ Address Tables are created separately with the primary and foreign key relationship.

Before the Next Class

Since you have successfully completed this session. Before next discussion you should go through:

- Maven

- JSP

What's Next?

Now, we have finished our Web Pages for the Admin Dashboard. In our next live discussion, we will:

- Create dynamic Web Project using Enterprise Edition of Java in Eclipse

- Create the Model Structure i.e. POJO Classes for the Project

- Implement DAO and Service Design Patterns

- Create Servlets and JDBC Connections

Caltech | Center for Technology & Management Education