**Beer Explorer**

Beer Explorer is a Ecommerce web site where we order beer in very efficient manner.

We create all the component in Lightning Web Component.

**Concept Used In LWC:**

* Lightning Input Field
* Lightning Search input
* Decorator(@api,@track,@wire)
* imperative method & imperative method(parameters)
* Event (System event & Custom Event)
* Lightning-record-form
* Lightning design system
* Modal popup
* Toast message
* lightning/uiRecordApi
* NavigationMixin

Concept Used in Apex:

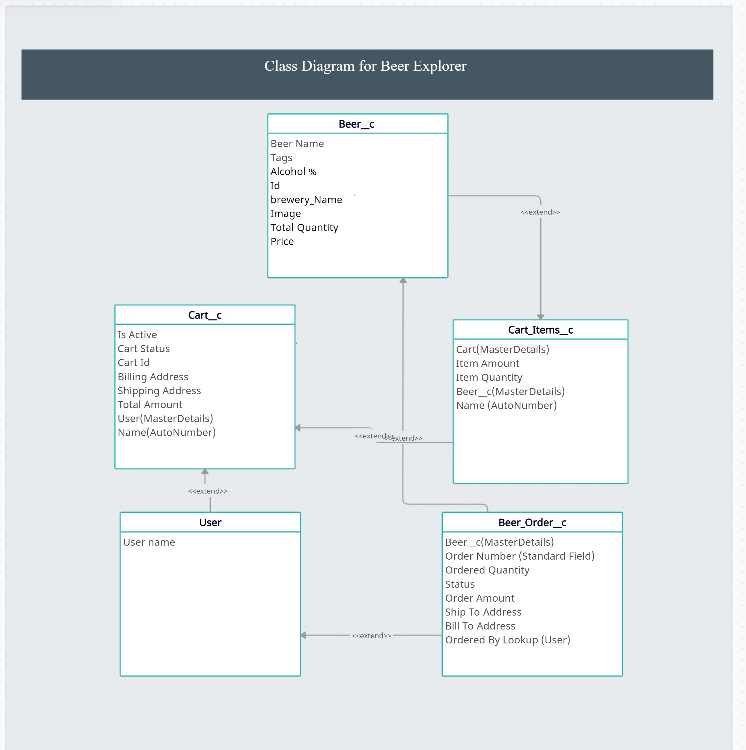
* SOQl
* AggregateFunction

Admin:

* Custom Object
* Custom Field
* Email Template
* Email Alert
* Flow

Data Model: -

1. Create a product list with an image.
2. Add multiple filters to the product list.
3. Add the product to Cart.
4. Can create Order.
5. I create 3 objects for this task (Beer\_\_c,Beer\_order\_\_c,cart\_\_c)

****

**Beer\_\_c**

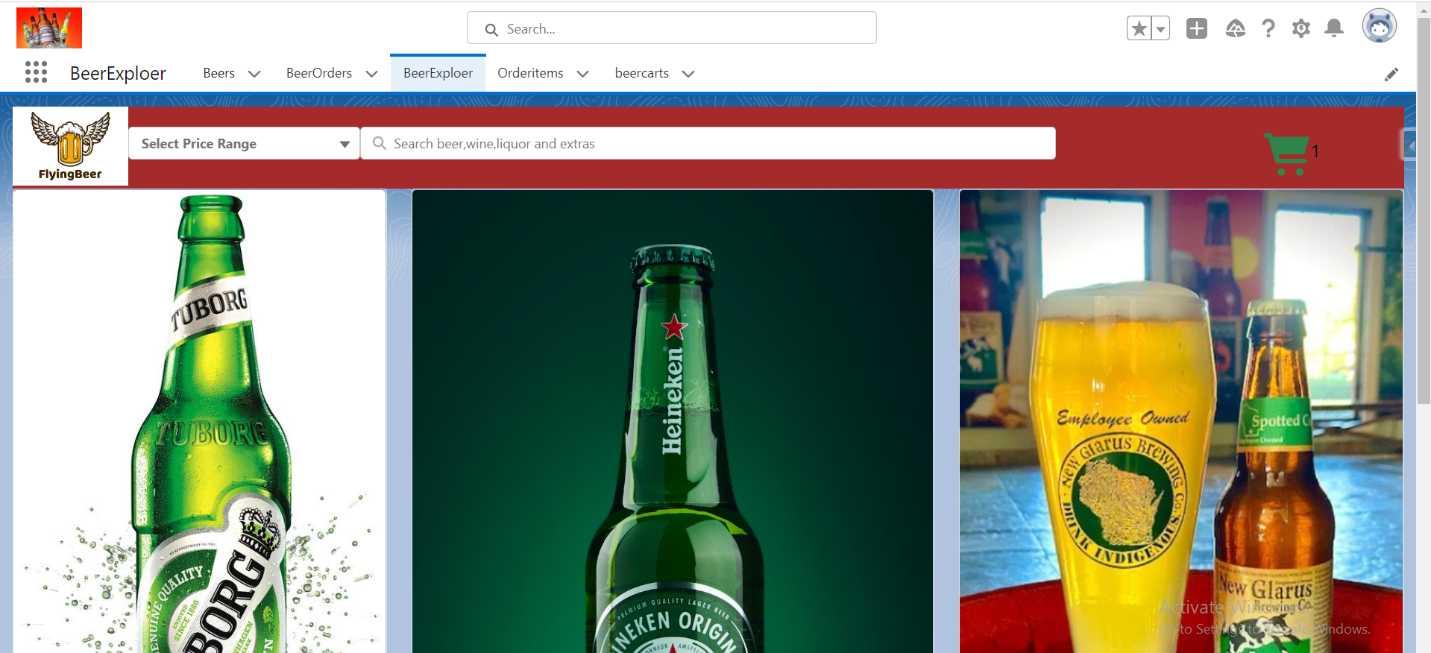
|  |  |  |
| --- | --- | --- |
| **Sr. No** | **Field Label** | **Data Type** |
| **1** | Beer Name | Text (80) |
| **2** | Tags | Text (80) |
| **3** | Alcohol % | Percentage (3,2) |
| **4** | Id | Text (10) |
| **5** | brewery\_id | Text (10) |
| **6** | brewery\_Name | Text (255) |
| **7** | Image | URL |
| **8** | Total Quantity | Number (5,0) |
| **9** | Remaining Quantity | Number (5,0) |
| **10** | Price | Currency (5,0) |

**Beer\_Order\_\_c**

|  |  |  |
| --- | --- | --- |
| **Sr. No** | **Field Label** | **Data Type** |
| **1** | Order Number (Standard Field) | AutoNumber {ORD - 0000} |
| **2** | Beer\_\_c | Lookup (Beer\_\_c) |
| **3** | Ordered Quantity | Number (3, 0) |
| **5** | Order Amount | Currency (5,2) |
| **6** | [Billing Street](https://stronginfo-dev-ed.lightning.force.com/lightning/setup/ObjectManager/01I5j000000Gkt8/FieldsAndRelationships/00N5j00000J2Bcm/view), [Billing City](https://stronginfo-dev-ed.lightning.force.com/lightning/setup/ObjectManager/01I5j000000Gkt8/FieldsAndRelationships/00N5j00000J2BkM/view), [Billing Country](https://stronginfo-dev-ed.lightning.force.com/lightning/setup/ObjectManager/01I5j000000Gkt8/FieldsAndRelationships/00N5j00000J2DP6/view), [Billing Postal Code](https://stronginfo-dev-ed.lightning.force.com/lightning/setup/ObjectManager/01I5j000000Gkt8/FieldsAndRelationships/00N5j00000J2DOw/view), [Billing State](https://stronginfo-dev-ed.lightning.force.com/lightning/setup/ObjectManager/01I5j000000Gkt8/FieldsAndRelationships/00N5j00000J2BkW/view) | Text(200) |
| **7** | [Shipping City](https://stronginfo-dev-ed.lightning.force.com/lightning/setup/ObjectManager/01I5j000000Gkt8/FieldsAndRelationships/00N5j00000J2BkR/view), [Shipping Country](https://stronginfo-dev-ed.lightning.force.com/lightning/setup/ObjectManager/01I5j000000Gkt8/FieldsAndRelationships/00N5j00000J2DPB/view), [Shipping Postal Code](https://stronginfo-dev-ed.lightning.force.com/lightning/setup/ObjectManager/01I5j000000Gkt8/FieldsAndRelationships/00N5j00000J2DP1/view), [Shipping State](https://stronginfo-dev-ed.lightning.force.com/lightning/setup/ObjectManager/01I5j000000Gkt8/FieldsAndRelationships/00N5j00000J2Bkb/view), [Shipping Street](https://stronginfo-dev-ed.lightning.force.com/lightning/setup/ObjectManager/01I5j000000Gkt8/FieldsAndRelationships/00N5j00000J2Bhr/view) | Text(200) |
| **8** | [Customer Name](https://stronginfo-dev-ed.lightning.force.com/lightning/setup/ObjectManager/01I5j000000Gkt8/FieldsAndRelationships/00N5j00000J2huw/view) | Text(200) |
| **9** | Email | Email |
| **10** | [Phone](https://stronginfo-dev-ed.lightning.force.com/lightning/setup/ObjectManager/01I5j000000Gkt8/FieldsAndRelationships/00N5j00000J2hvp/view) | Text(10) |

**Beer\_Cart\_\_c**

|  |  |  |
| --- | --- | --- |
| **Sr. No** | **Field Label** | **Data Type** |
| **1** | Beer Name | Text (80) |
| **2** | Tags | Text (80) |
| **3** | Alcohol % | Percentage (3,2) |
| **4** | Id | Text (10) |
| **5** | brewery\_Name | Text (255) |
| **6** | Image | URL |
| **7** | Total Quantity | Number (5,0) |
| **8** | Price | Currency (5,0) |



**Apex Class:**

public class Beersearchhelper {

//Get all the beer details

    @AuraEnabled(cacheable=true)

    public static list<beer\_\_c> getbeer(string searchbeername){

        return [select id,Name,Image\_\_c,Price\_\_c,Alcohol\_\_c,Tags\_\_c,brewery\_Name\_\_c,Id\_\_c,  Total\_Quantity\_\_c from beer\_\_c where name like:searchbeername+'%' limit 20];

    }

//getbeerbyusingfilter

     @AuraEnabled(cacheable=true)

    public static list<beer\_\_c> getbeerbyprice(integer pricerange){

        system.debug(pricerange);

        return [select id,Name,Image\_\_c,Price\_\_c,Alcohol\_\_c,Tags\_\_c,brewery\_Name\_\_c,Id\_\_c,Total\_Quantity\_\_c from beer\_\_c where  Price\_\_c <=: pricerange ];

    }

//create beer record in beercart object

    @AuraEnabled

    public static list<beercart\_\_c> createbeerrecord(string name,string beerimage,decimal beerPrice,decimal beerAlcohol,string beerbrewery\_Name){

        list<beercart\_\_c>beerlist=new list<beercart\_\_c>();

        beercart\_\_c beer =new beercart\_\_c ();

        beer.Name=name;

        beer.Image\_\_c=beerimage;

        beer.Price\_\_c=beerPrice;

        beer.Alcohol\_\_c=beerAlcohol;

         beer.brewery\_Name\_\_c=beerbrewery\_Name;

            beerlist.add(beer);

        if(!beerlist.isempty()){

            insert beerlist;

            return beerlist;

        }

        return null;

    }

//fetch all the cart record

     @AuraEnabled(cacheable=true)

    public static list<beercart\_\_c> getbeercart(){

        return [select id,Name,Image\_\_c,Price\_\_c,Alcohol\_\_c,Tags\_\_c,brewery\_Name\_\_c from beercart\_\_c];

    }

    //fetch all cart size record

     @AuraEnabled(cacheable=true)

    public static List<AggregateResult>  getsizeofcart(){

        return [Select COUNT(Id) from beercart\_\_c];

    }

    //calculate total price

     @AuraEnabled(cacheable=true)

    public static List<AggregateResult>  getsumofammuont(){

        return [select sum(price\_\_c) from beercart\_\_c];

    }

//create multiple beer order

     @AuraEnabled

     public static list<BeerOrder\_\_c> createmultiplebeerorder(string bstreet,string bcity,string bcountry, string bpostal,string bstate,

      string sstreet,string scity,string scountry,string spostal,string sstate,Decimal totalamount,Integer nooforder,

      string cname,string cphone,string cemail

     ){

        list<BeerOrder\_\_c>listbeerorder=new list<BeerOrder\_\_c>();

           BeerOrder\_\_c beerorder=new BeerOrder\_\_c();

           beerorder.Billing\_City\_\_c= bcity;

           beerorder.Billing\_Country\_\_c= bcountry;

           beerorder.Billing\_Postal\_Code\_\_c=bpostal;

           beerorder.Billing\_State\_\_c=bstate;

           beerorder.Billing\_Street\_\_c=bstreet;

           beerorder.Shipping\_City\_\_c=scity;

           beerorder.Shipping\_Country\_\_c=scountry;

           beerorder.Shipping\_Postal\_Code\_\_c=spostal;

           beerorder.Shipping\_\_c=sstate;

           beerorder.Shipping\_Street\_\_c=sstreet;

           beerorder.Ordered\_Quantity\_\_c=nooforder;

           beerorder.Order\_Amount\_\_c=totalamount;

           beerorder.Customer\_Name\_\_c=cname;

           beerorder.Phone\_\_c=cphone;

           beerorder.email\_\_c=cemail;

              listbeerorder.add(beerorder);

        if(!listbeerorder.isempty()){

            insert listbeerorder;

            return listbeerorder;

        }

     return null;

     }

    //fetch all the Order details record

     @AuraEnabled(cacheable=true)

    public static list<BeerOrder\_\_c> getbeerorder(){

        return [select id,Name,email\_\_c,Billing\_City\_\_c,Billing\_Country\_\_c,Billing\_Postal\_Code\_\_c,Billing\_State\_\_c,

        Billing\_Street\_\_c,Order\_Amount\_\_c,Ordered\_Quantity\_\_c,

        Shipping\_City\_\_c,Shipping\_Country\_\_c,Shipping\_Postal\_Code\_\_c,Shipping\_\_c,Shipping\_Street\_\_c,Customer\_Name\_\_c

         from BeerOrder\_\_c order by createdDate DESC limit 1];

    }

}