Model Driven Engineering

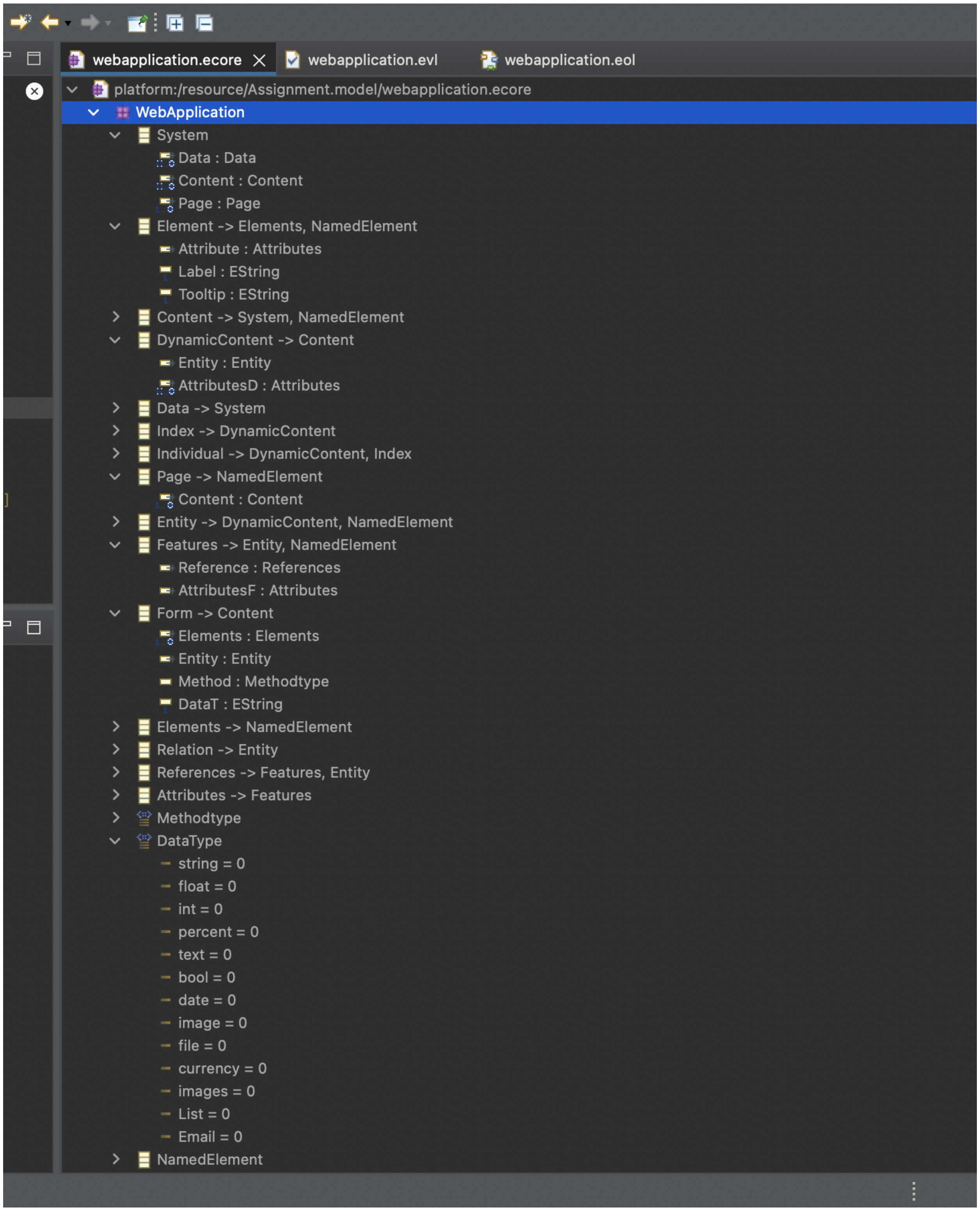
Assignment 4

By

Team Chicken Restaurant:

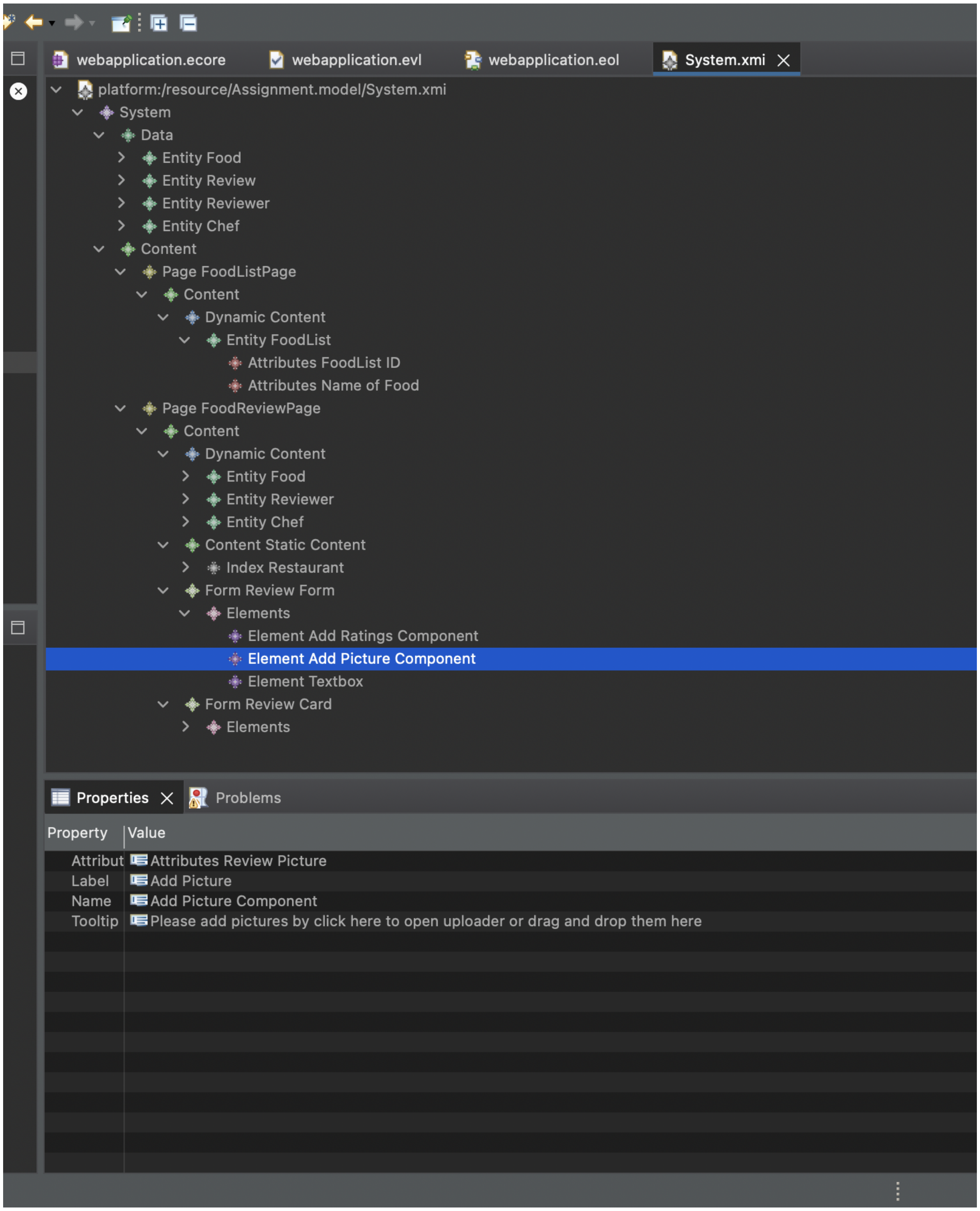
Cindy Aprilia
Korawit Rupanya
Mercy Bamiduro

1)
Based on our previous projects developed on MPS, we defined 16 metaclasses. Each metaclass contains at least one attribute or reference as demonstrated in the screenshot below. Also, the metamodel was properly defined and contains inheritance, containment, enumeration types, attributes, data types, etc.



```
platform:/resource/Assignment.model/webapplication.ecore
  WebApplication
    System
      Data : Data
      Content : Content
      Page : Page
    Element -> Elements, NamedElement
      Attribute : Attributes
      Label : EString
      Tooltip : EString
    Content -> System, NamedElement
    DynamicContent -> Content
      Entity : Entity
      AttributesD : Attributes
    Data -> System
    Index -> DynamicContent
    Individual -> DynamicContent, Index
    Page -> NamedElement
      Content : Content
    Entity -> DynamicContent, NamedElement
    Features -> Entity, NamedElement
      Reference : References
      AttributesF : Attributes
    Form -> Content
      Elements : Elements
      Entity : Entity
      Method : Methodtype
      DataT : EString
    Elements -> NamedElement
    Relation -> Entity
    References -> Features, Entity
    Attributes -> Features
    Methodtype
    DataType
      string = 0
      float = 0
      int = 0
      percent = 0
      text = 0
      bool = 0
      date = 0
      image = 0
      file = 0
      currency = 0
      images = 0
      List = 0
      Email = 0
    NamedElement
```

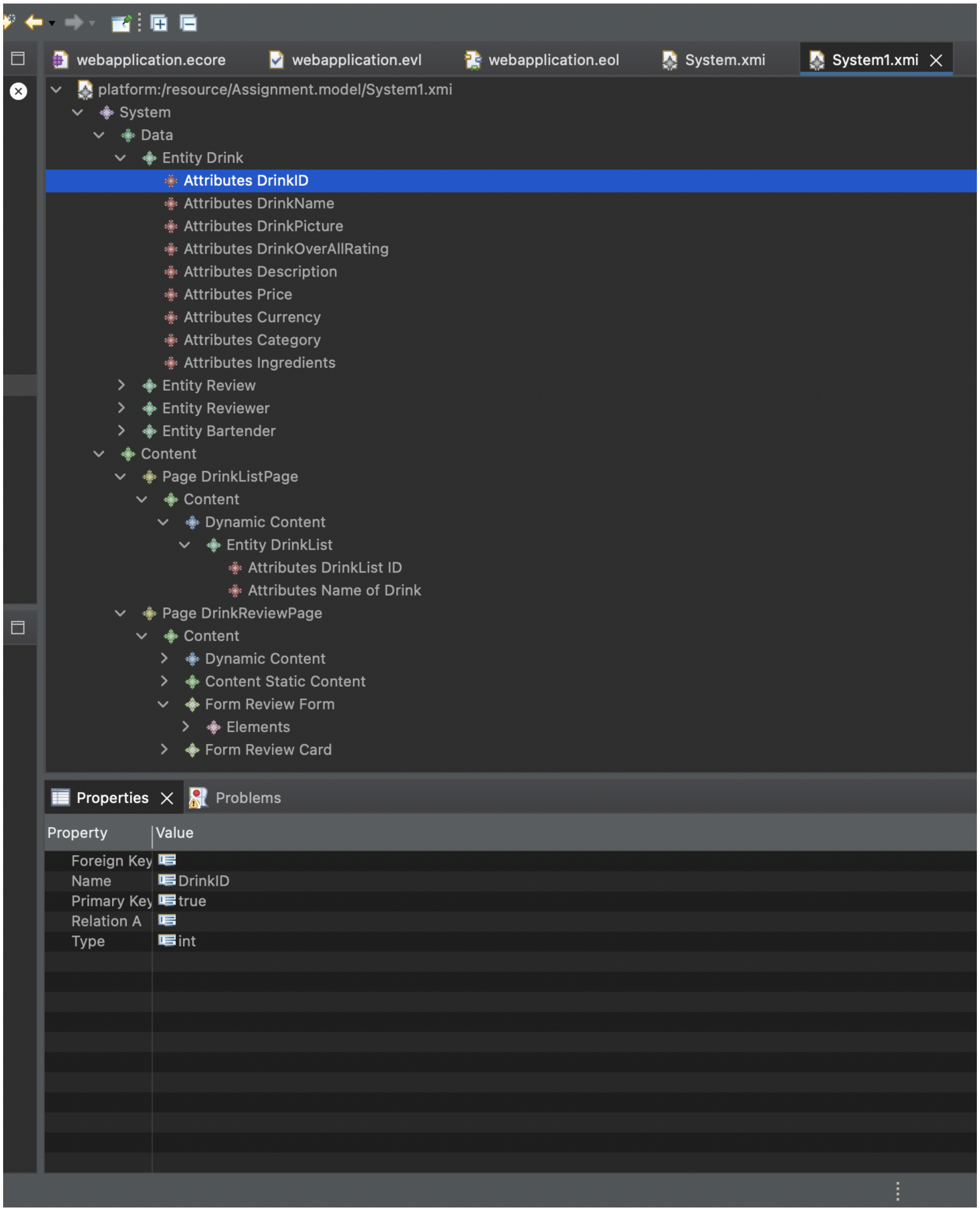
2)
Our metamodel was further instantiated by the first concrete instances named "***System.xmi***" with each concept represented in the above metaclasses, well instantiated in the model as seen in the image below;



```
platform:/resource/Assignment.model/System.xmi
  System
    Data
      Entity Food
      Entity Review
      Entity Reviewer
      Entity Chef
    Content
      Page FoodListPage
        Content
          Dynamic Content
            Entity FoodList
              Attributes FoodList ID
              Attributes Name of Food
      Page FoodReviewPage
        Content
          Dynamic Content
            Entity Food
            Entity Reviewer
            Entity Chef
          Content Static Content
            Index Restaurant
          Form Review Form
            Elements
              Element Add Ratings Component
              Element Add Picture Component
              Element Textbox
          Form Review Card
            Elements
```

Property	Value
Attribut	Attributes Review Picture
Label	Add Picture
Name	Add Picture Component
Tooltip	Please add pictures by click here to open uploader or drag and drop them here

2b)
Additionally, the metamodel was further instantiated by the second concrete instance named "***System1.xmi***" with each concept also represented in the above metaclasses, well instantiated in the model too as seen in the image below;



webapplication.ecore | webapplication.evl | webapplication.eol | System.xmi | System1.xmi

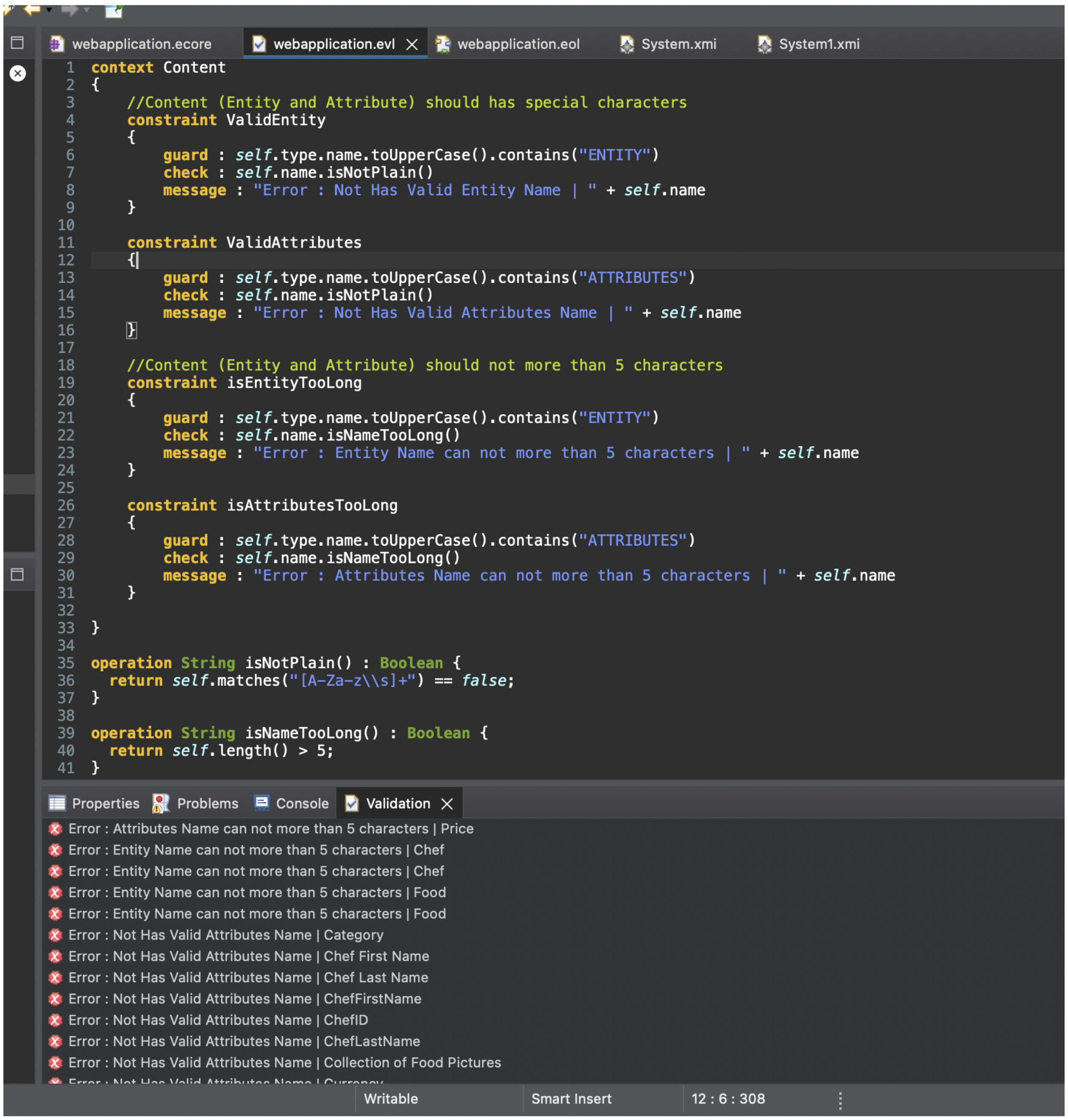
```
platform:/resource/Assignment.model/System1.xmi
  System
    Data
      Entity Drink
        Attributes DrinkID
        Attributes DrinkName
        Attributes DrinkPicture
        Attributes DrinkOverAllRating
        Attributes Description
        Attributes Price
        Attributes Currency
        Attributes Category
        Attributes Ingredients
      Entity Review
      Entity Reviewer
      Entity Bartender
    Content
      Page DrinkListPage
        Content
          Dynamic Content
            Entity DrinkList
              Attributes DrinkList ID
              Attributes Name of Drink
      Page DrinkReviewPage
        Content
          Dynamic Content
          Content Static Content
          Form Review Form
            Elements
          Form Review Card
```

Properties | Problems

Property	Value
Foreign Key	
Name	DrinkID
Primary Key	true
Relation A	
Type	int

3)
We have defined the following constraints for both models as demonstrated in the image below and available in the video recording attached also.

- To check if the Entity has special characters.
- To check if the Attributes have special characters.
- To check if the Entity name has more than 5 characters.
- To check if the Attributes name has more than 5 characters.

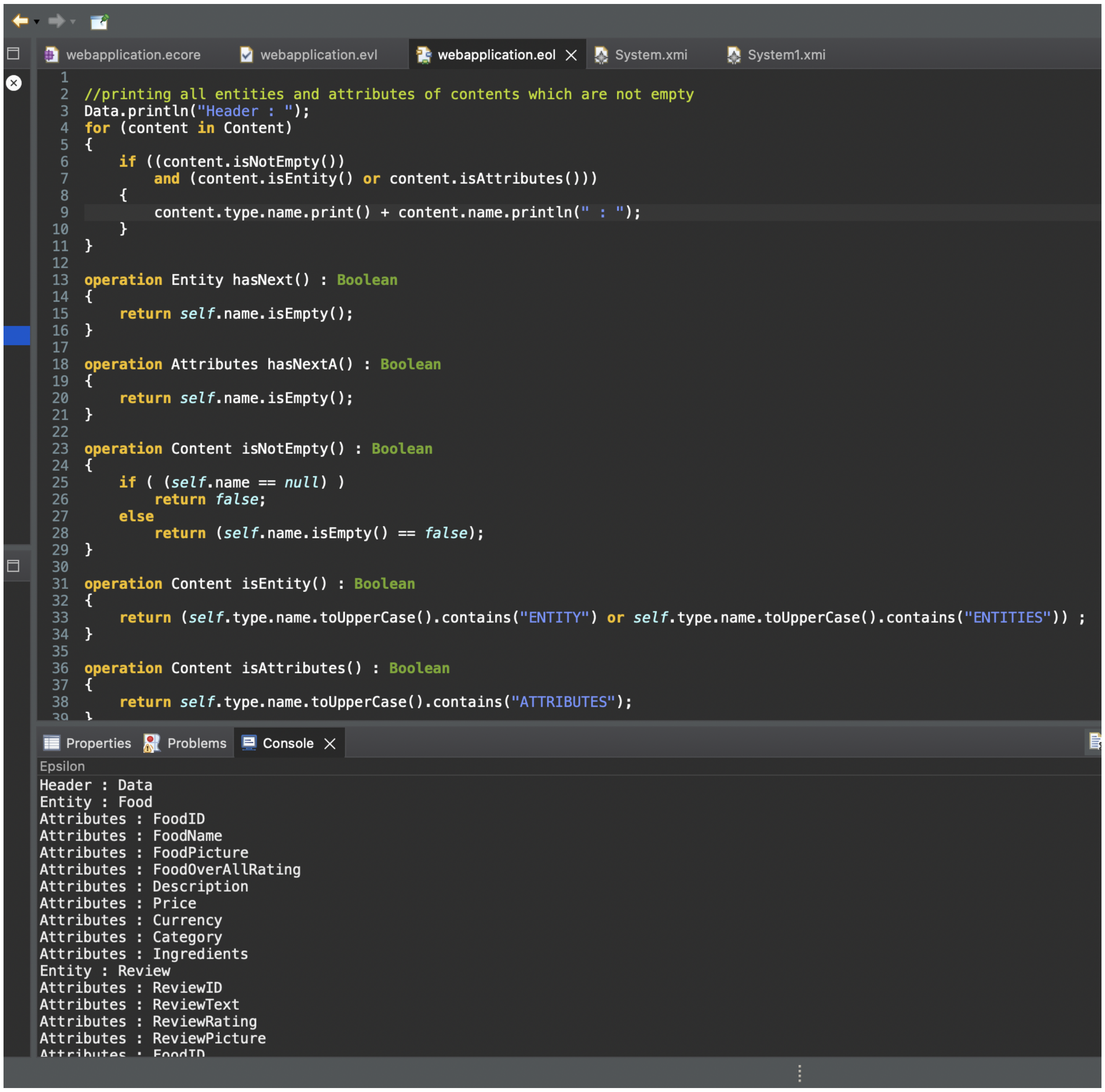


```
context Content
{
    //Content (Entity and Attribute) should has special characters
    constraint ValidEntity
    {
        guard : self.type.name.toUpperCase().contains("ENTITY")
        check : self.name.isNotPlain()
        message : "Error : Not Has Valid Entity Name | " + self.name
    }

    constraint ValidAttributes
    {
        guard : self.type.name.toUpperCase().contains("ATTRIBUTES")
        check : self.name.isNotPlain()
        message : "Error : Not Has Valid Attributes Name | " + self.name
    }

    //Content (Entity and Attribute) should not more than 5 characters
    constraint isEntityTooLong
    {
        guard : self.type.name.toUpperCase().contains("ENTITY")
        check : self.name.isNameTooLong()
        message : "Error : Entity Name can not more than 5 characters | " + self.name
    }

    constraint isAttributesTooLong
    {
        guard : self.type.name.toUpperCase().contains("ATTRIBUTES")
        check : self.name.isNameTooLong()
        message : "Error : Attributes Name can not more than 5 characters | " + self.name
    }

}

operation String isNotPlain() : Boolean {
  return self.matches("[A-Za-z\\s]+") == false;
}

operation String isNameTooLong() : Boolean {
  return self.length() > 5;
}
```

```
Error : Attributes Name can not more than 5 characters | Price
Error : Entity Name can not more than 5 characters | Chef
Error : Entity Name can not more than 5 characters | Chef
Error : Entity Name can not more than 5 characters | Food
Error : Entity Name can not more than 5 characters | Food
Error : Not Has Valid Attributes Name | Category
Error : Not Has Valid Attributes Name | Chef First Name
Error : Not Has Valid Attributes Name | Chef Last Name
Error : Not Has Valid Attributes Name | ChefFirstName
Error : Not Has Valid Attributes Name | ChefID
Error : Not Has Valid Attributes Name | ChefLastName
Error : Not Has Valid Attributes Name | Collection of Food Pictures
Error : Not Has Valid Attributes Name | Currency
```

3b)
We have also defined the following operations as seen in the image below and in the video recording.

- Printing all entities of contents that are not empty.
- Printing all attributes of contents that are not empty.



```
//printing all entities and attributes of contents which are not empty
Data.println("Header : ");
for (content in Content)
{
    if ((content.isNotEmpty())
        and (content.isEntity() or content.isAttributes()))
    {
        content.type.name.print() + content.name.println(" : ");
    }
}

operation Entity hasNext() : Boolean
{
    return self.name.isEmpty();
}

operation Attributes hasNextA() : Boolean
{
    return self.name.isEmpty();
}

operation Content isNotEmpty() : Boolean
{
    if ( (self.name == null) )
        return false;
    else
        return (self.name.isEmpty() == false);
}

operation Content isEntity() : Boolean
{
    return (self.type.name.toUpperCase().contains("ENTITY") or self.type.name.toUpperCase().contains("ENTITIES")) ;
}

operation Content isAttributes() : Boolean
{
    return self.type.name.toUpperCase().contains("ATTRIBUTES");
}
```

```
Header : Data
Entity : Food
Attributes : FoodID
Attributes : FoodName
Attributes : FoodPicture
Attributes : FoodOverAllRating
Attributes : Description
Attributes : Price
Attributes : Currency
Attributes : Category
Attributes : Ingredients
Entity : Review
Attributes : ReviewID
Attributes : ReviewText
Attributes : ReviewRating
Attributes : ReviewPicture
Attributes : FoodID
```

Also, find attached the videos of the outputs of our models.
[EVL:EOL short video clips]