Model Driven Engineering

Assignment 4

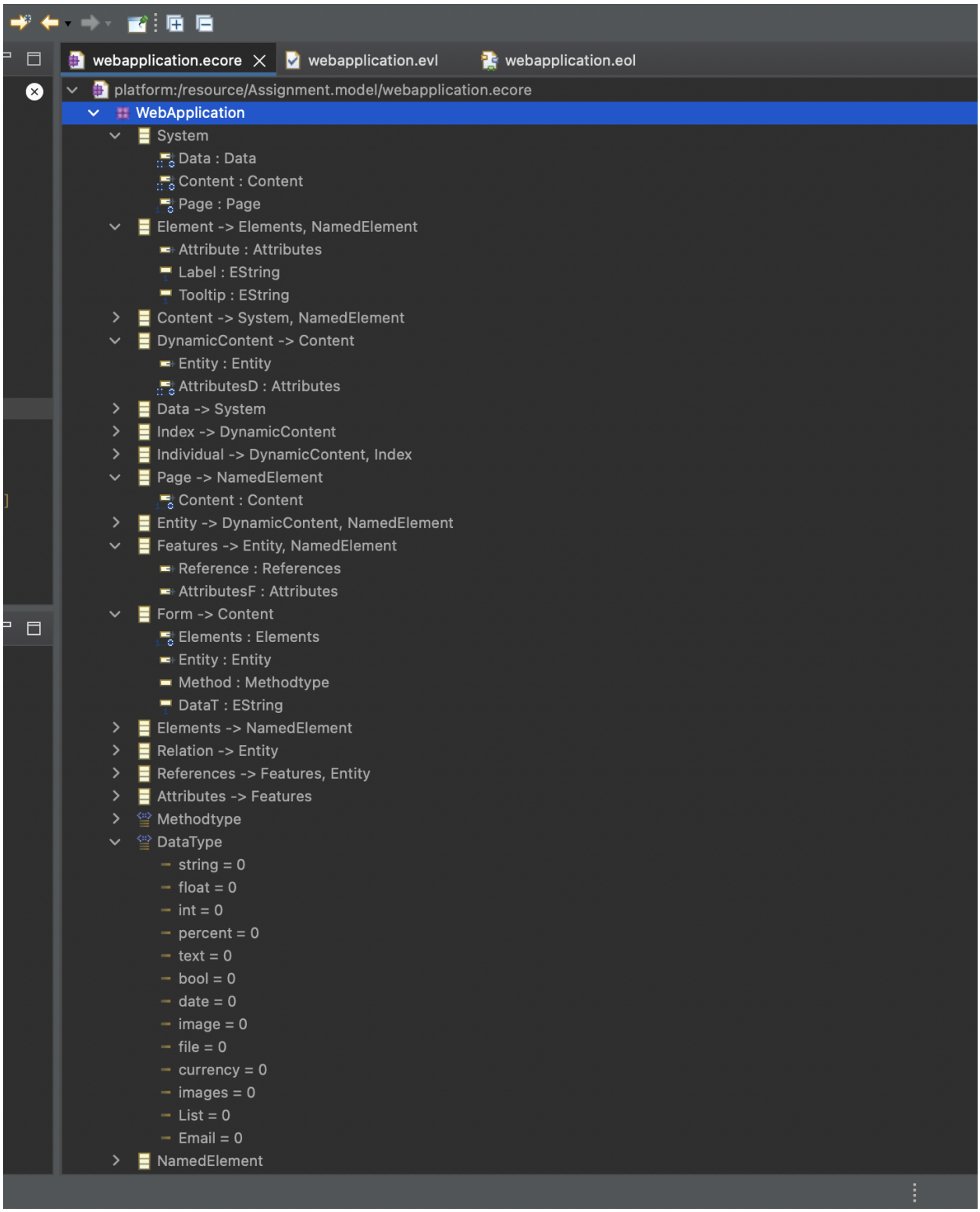
By

Team Chicken Restaurant:

Cindy Aprilia
Korawit Rupanya
Mercy Bamiduro

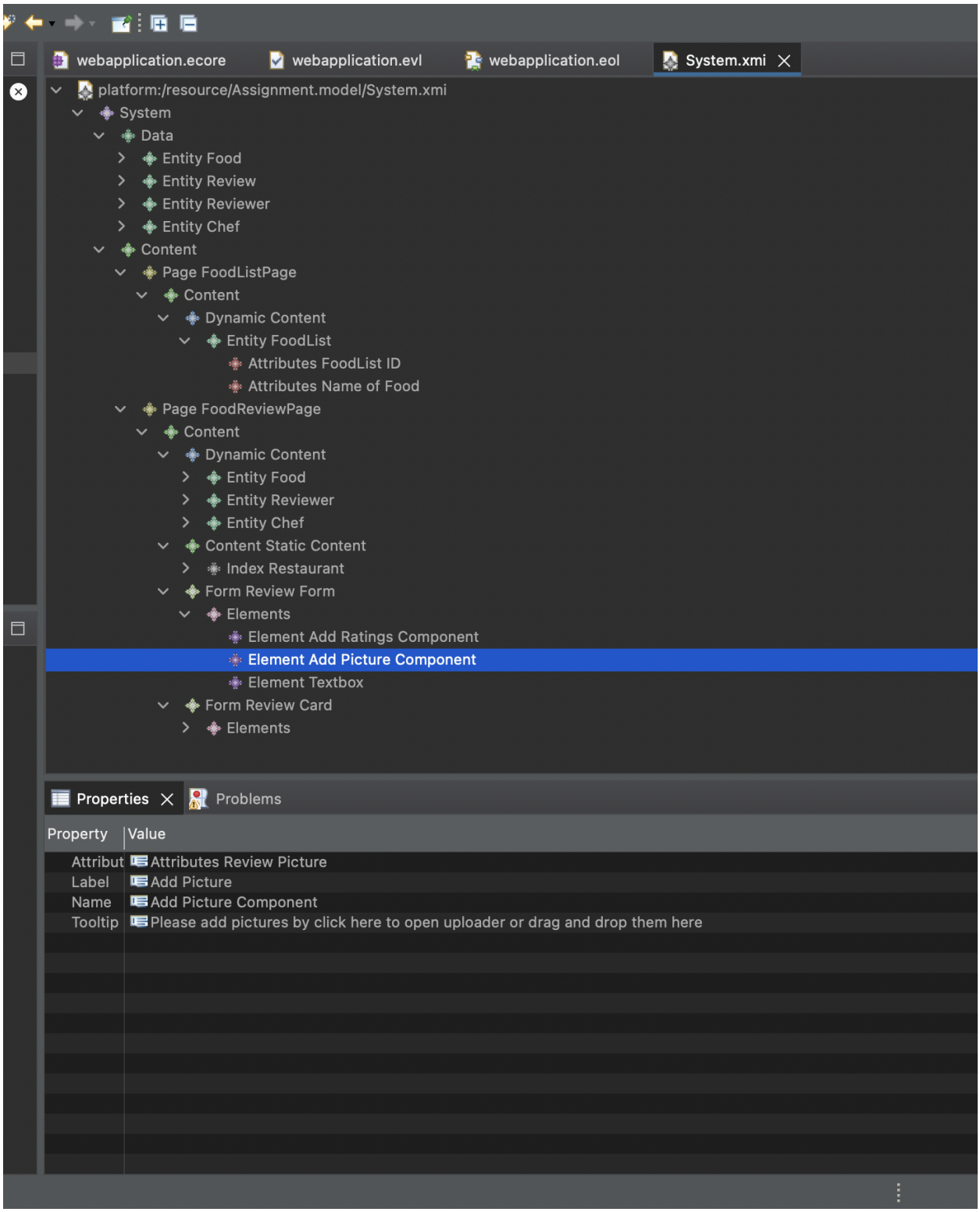
1)

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.

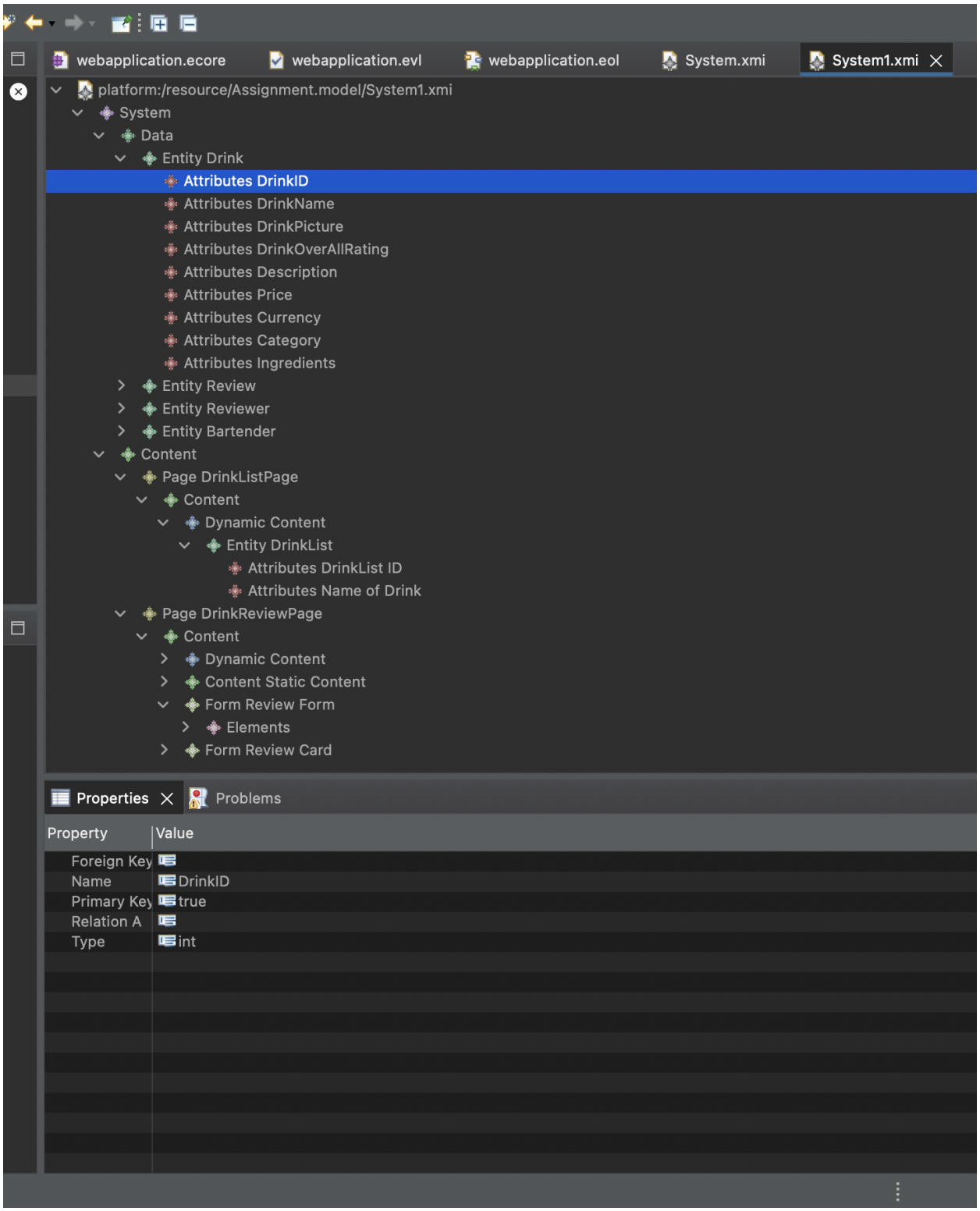


2)

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;



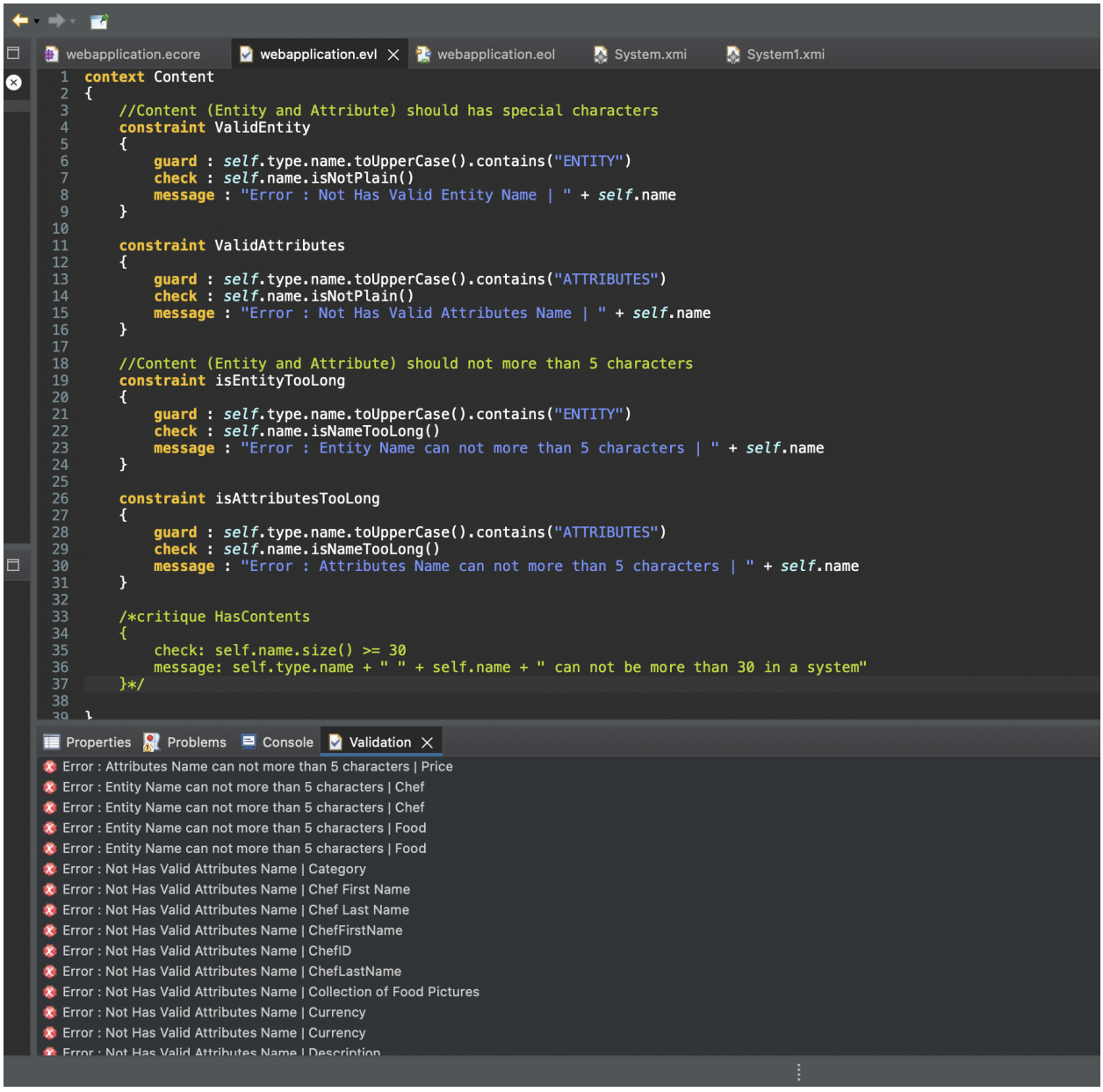
2b)
Additionally, the metamodel was further instantiated by the second concrete instance named "***System1.xmi***" with each concept also represented in the above metaclasses, well instantiated in the model too as seen in the image below;

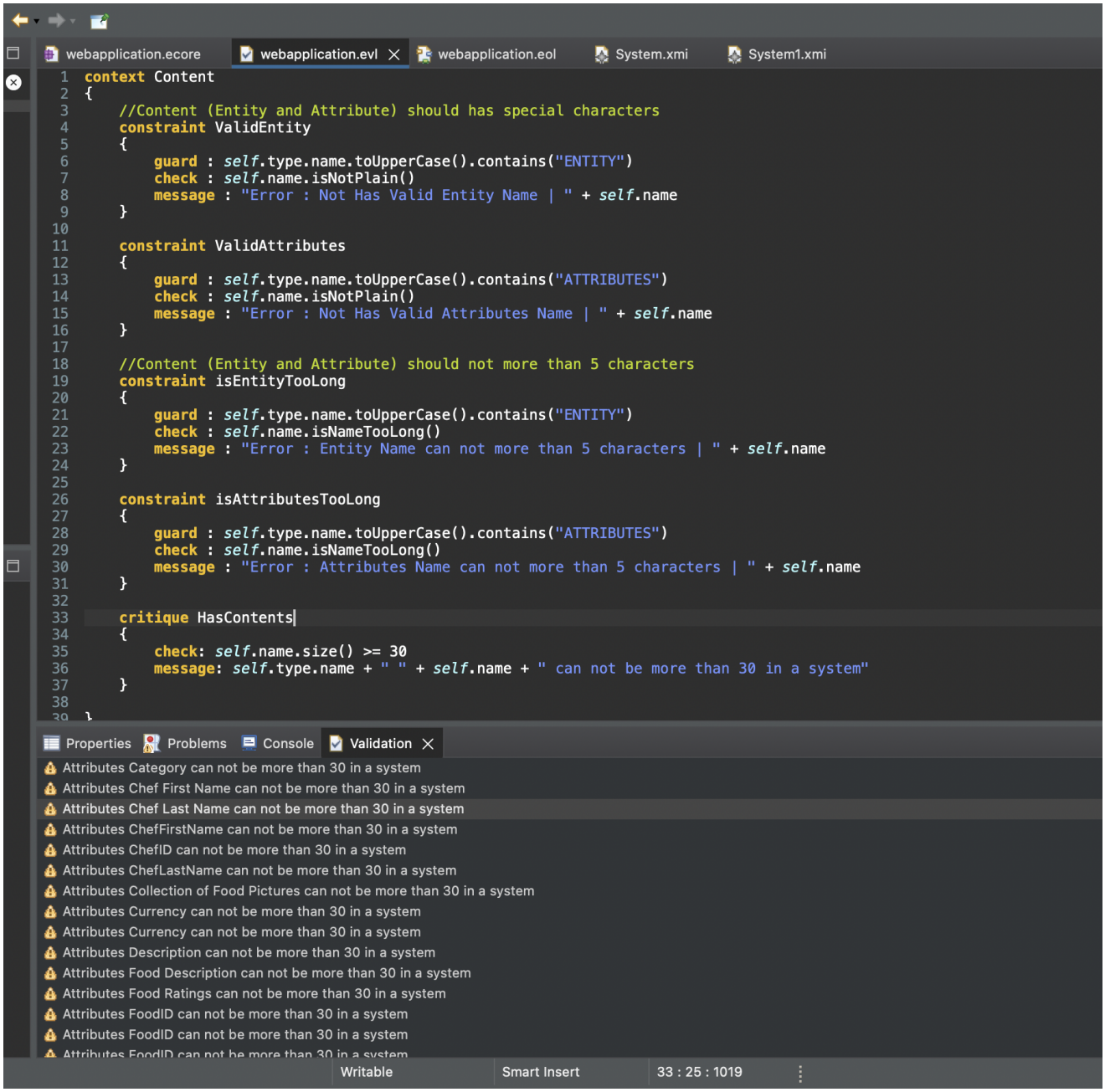


3)

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

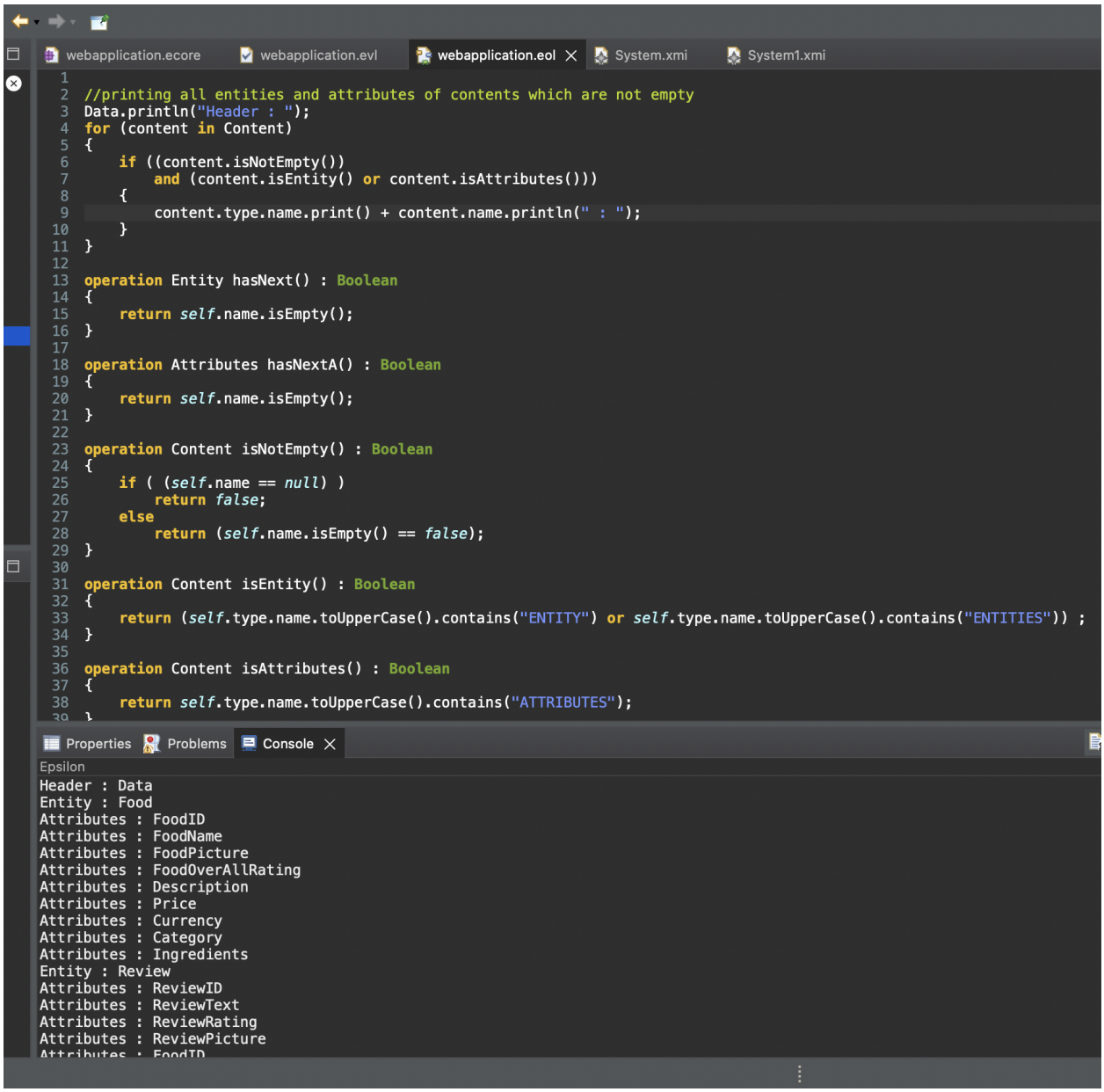
- To check if the Entity has special characters.
- To check if the Attributes have special characters.
- To check if the Entity name has more than 5 characters.
- To check if the Attributes name has more than 5 characters.
- To check if the Content is more than 30 in a system.




webapplication.ecore
webapplication.evl
webapplication.eol
System.xmi
System1.xmi
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
}
critique HasContents
{
check: self.name.size() >= 30
message: self.type.name + " " + self.name + " can not be more than 30 in a system"
}
}
Properties
Problems
Console
Validation
Attributes Category can not be more than 30 in a system
Attributes Chef First Name can not be more than 30 in a system
Attributes Chef Last Name can not be more than 30 in a system
Attributes ChefFirstName can not be more than 30 in a system
Attributes ChefID can not be more than 30 in a system
Attributes ChefLastName can not be more than 30 in a system
Attributes Collection of Food Pictures can not be more than 30 in a system
Attributes Currency can not be more than 30 in a system
Attributes Currency can not be more than 30 in a system
Attributes Description can not be more than 30 in a system
Attributes Food Description can not be more than 30 in a system
Attributes Food Ratings can not be more than 30 in a system
Attributes FoodID can not be more than 30 in a system
Attributes FoodID can not be more than 30 in a system
Writable
Smart Insert
33 : 25 : 1019

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.



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