Cloud Provisioning System

Edwin Malek (Supervisor)
Dionny Santiago (Leader)
Xiayu Dong (Developer)

Overview of the System

Cloud Provision system:

A graphical Editor, capable of generating a CPS model, and transform its models to different XML types files and well as generating a series of cloud Calls for CPS deployment

Design Methodology

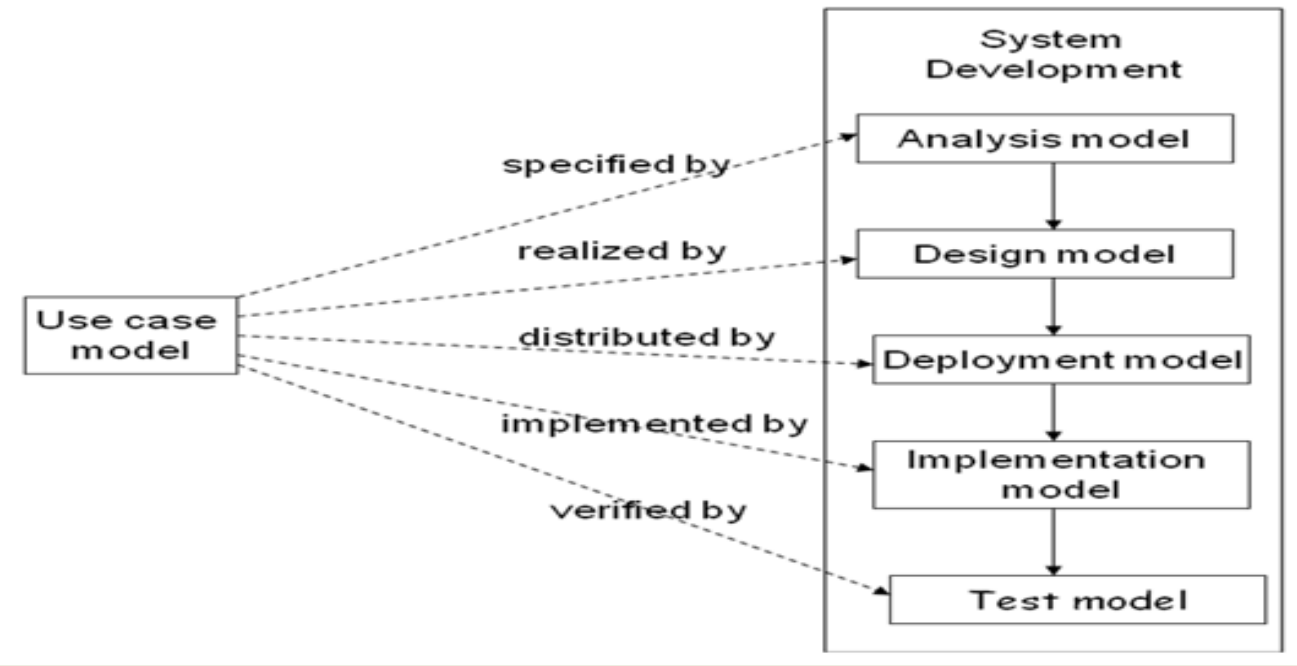


Figure 1-1 The Unified Software Development Process

Design Methodology (con't)

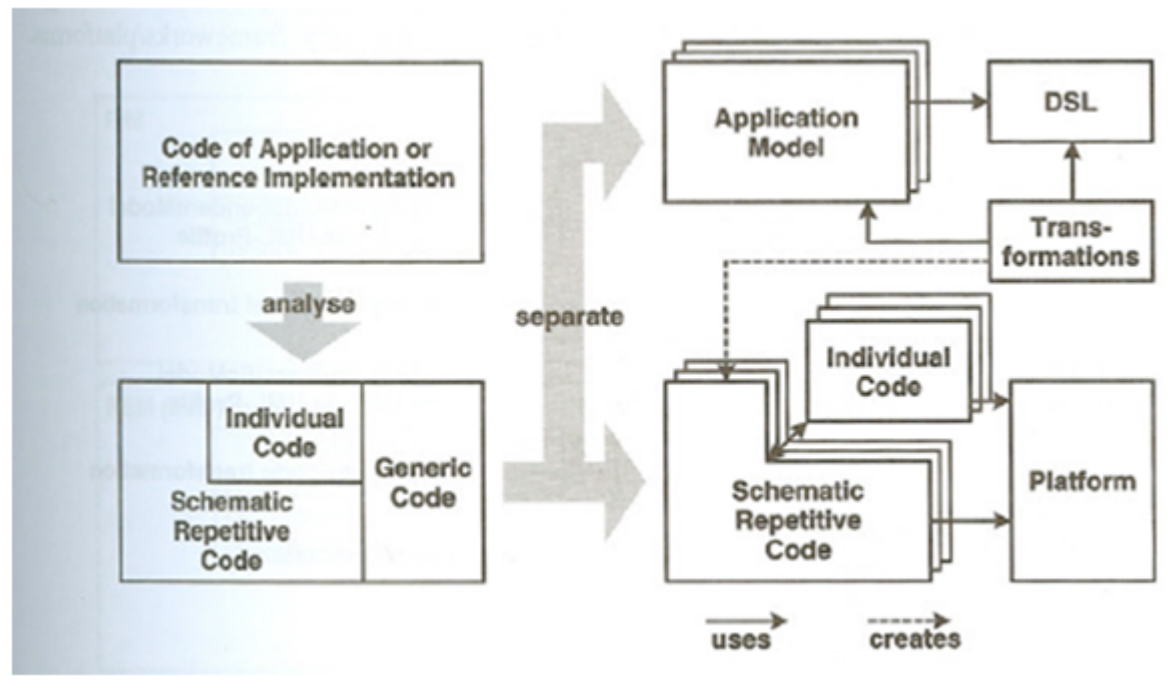


Figure 1-2 The Model Driven Software Development Process

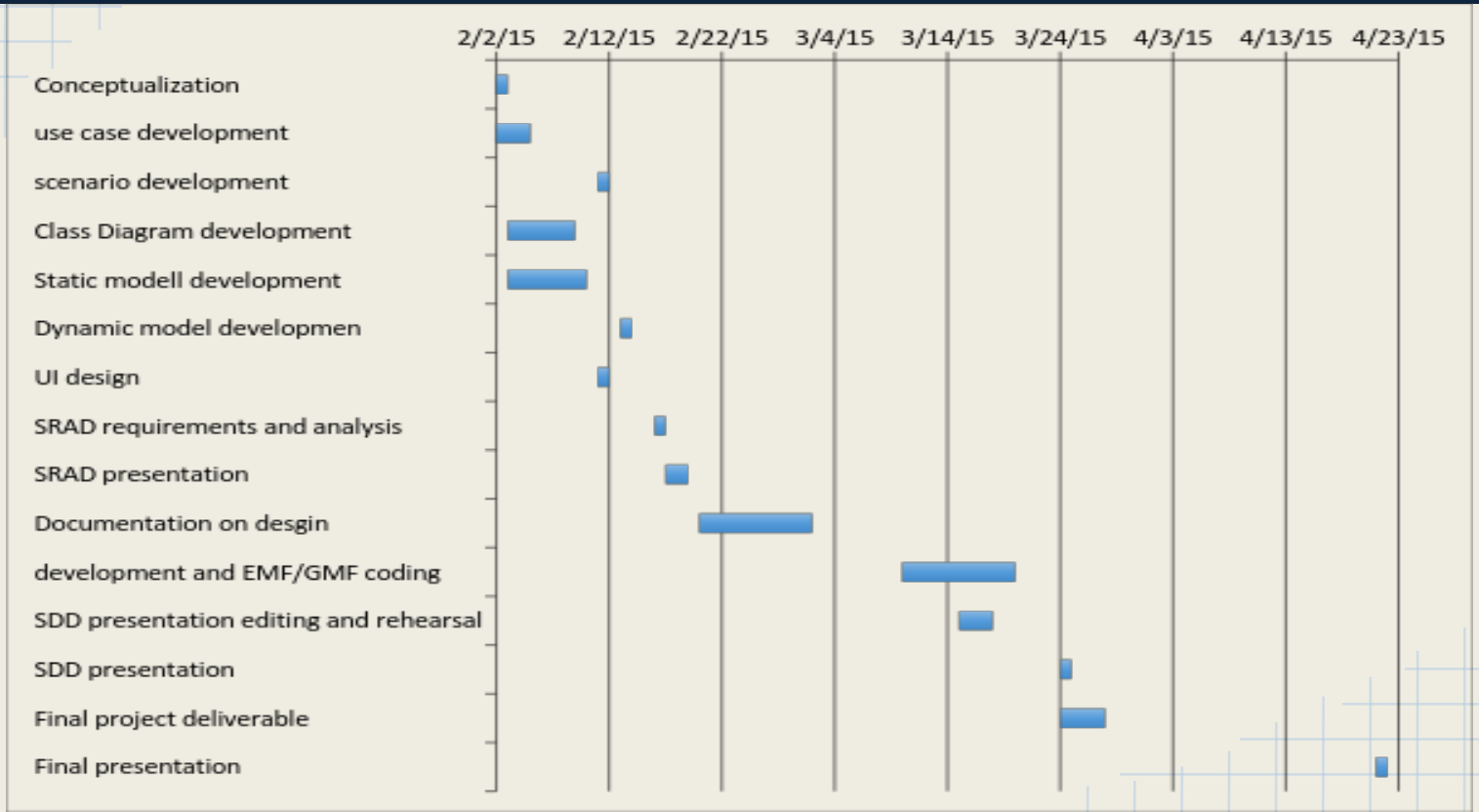
Project Plan

Xiayu Dong: Filtering and Transformation coding.

Dionny Santiago: Leader, GVE constraints creation, Filtering and transformation coding, EE supervisor.

Edwin Malek: Document organiser, GVE/EE constraints supervisor, System testing and Execution supervisor.

Project Plan (con't)



System Requirements

The system shall allow the user to create a model by adding different types of Nodes to a canvas representing the possible deployment options available today among most of the cloud provisioning providers.

System Requirements

The system shall allow the user to load an existing model. As well as viewing the model in a tree representation.

System Requirements (con’t)

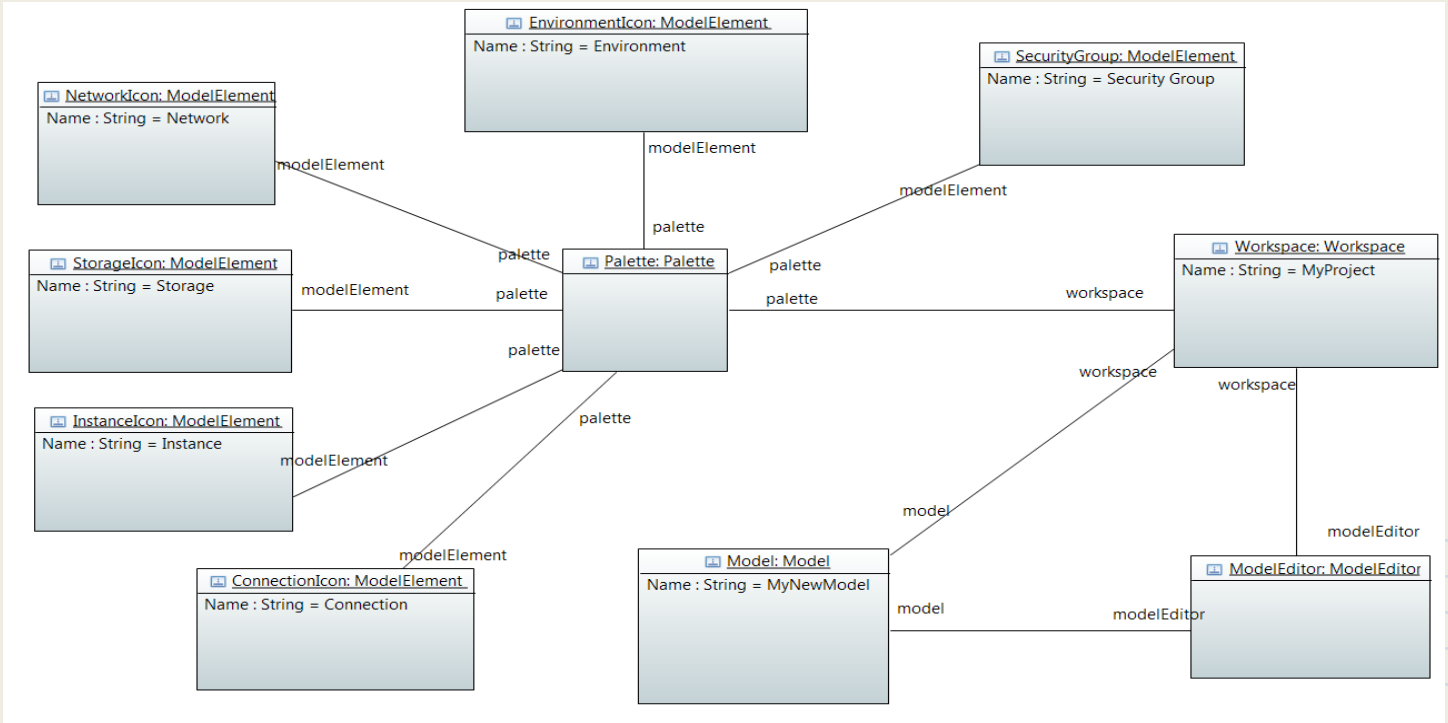
The system shall be able to analyze a user previously created CPS model and transform the model to a different type of XML file.

System Requirements (con’t)

The system shall allow the user to transform the generated XML type file to generate appropriate cloud calls according to the CPS provider (i.e Amazon EC2 API calls).

Validation of Analysis Model

CPROVME001 - Create new Model: Object Diagram



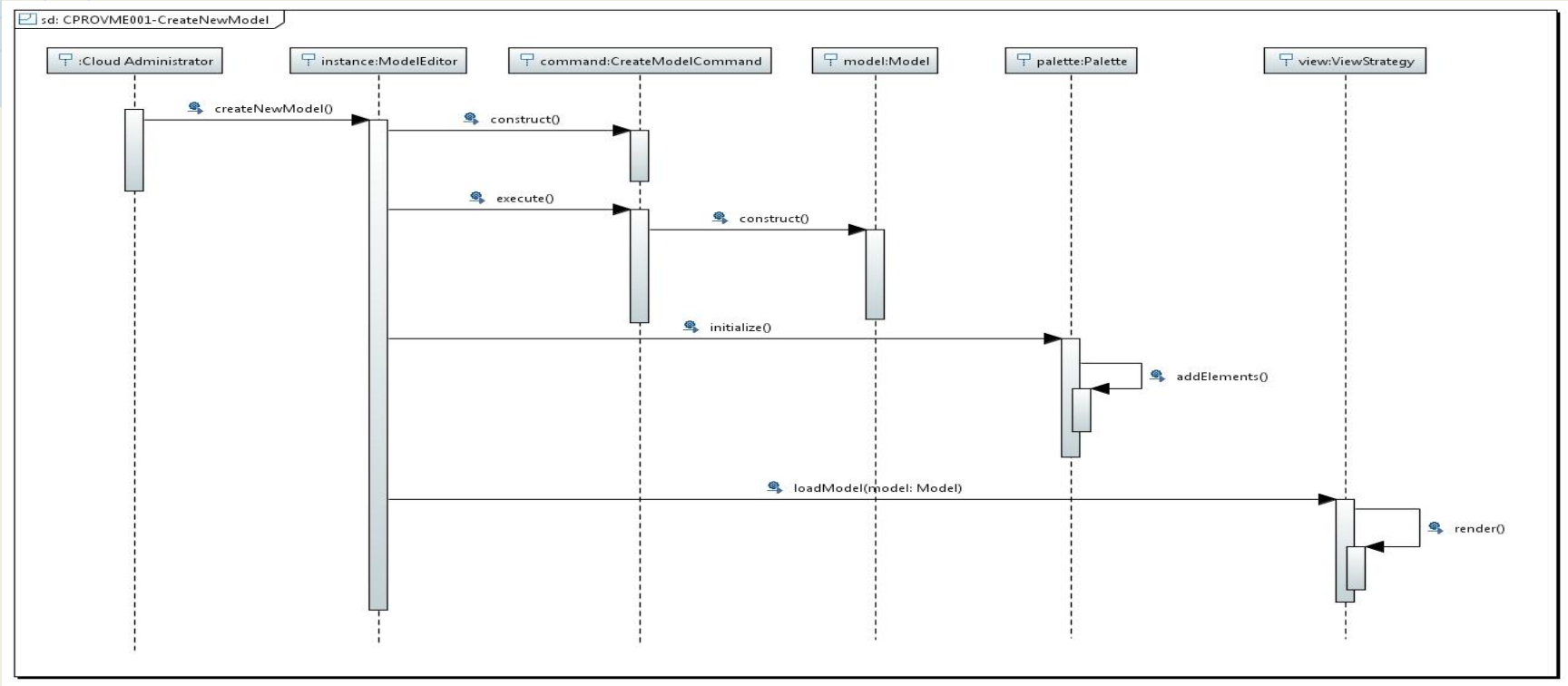
Validation of Analysis Model

CPROVME001 SCE1/SCE2:

Object diagram checklist:

Checklist:	**Yes**	**No**
1.Have all the necessary object correspond to a class in the class diagram ? consistency/completeness	✔	
2. Have all the necessary methods for each class been identified in the diagram? completeness	✔	
3. Have the dependencies/Associations been chosen correctly in the diagram ? correctness/ completeness	✔	
5. Is the diagram readable ? consistency	✔	
6. Are all necessary objects necessary to the execution of the test case in the diagram ? completeness	✔	

Validation of Analysis Model



Validation of Analysis Model

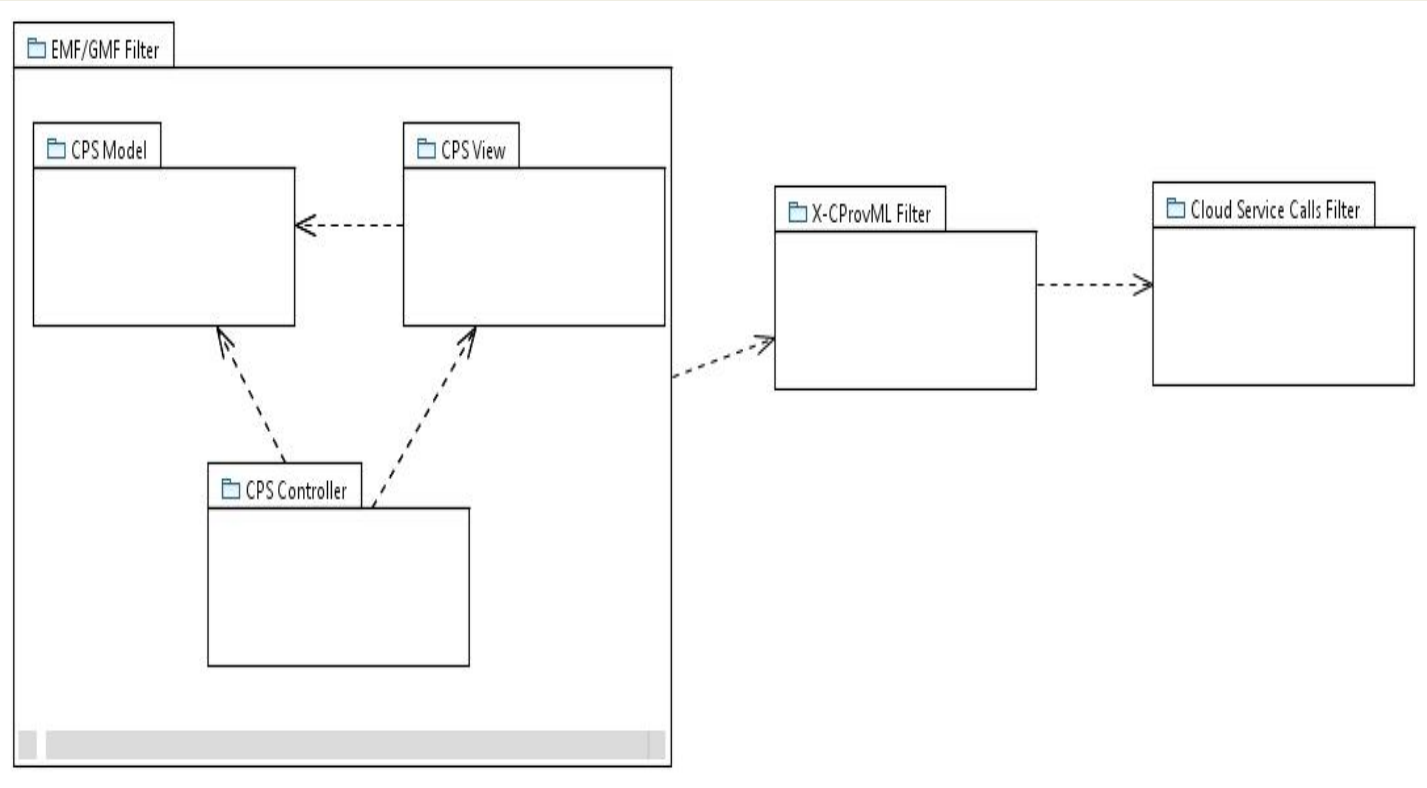
Sequence Diagram:

CPROVME001:

- Create New Model

checklist:	**Yes**	**No**
1.Do all relevant verbs in the use case represent method calls? completeness	✔	
2. Do all user interactions go to boundary objects ? correctness/completeness	✔	
3. Are all constraints represented in the sequence diagram? correctness/completeness	✔	
4. Are all objects created at the correct time ? correctness	✔	
5. Is the diagram readable ?	✔	
6. Are the method calls that require a return message present on the diagram ? correctness	✔	
7. Do all methods calls result in an action ? consistency	✔	

Validation of System Model



Validation of System Model

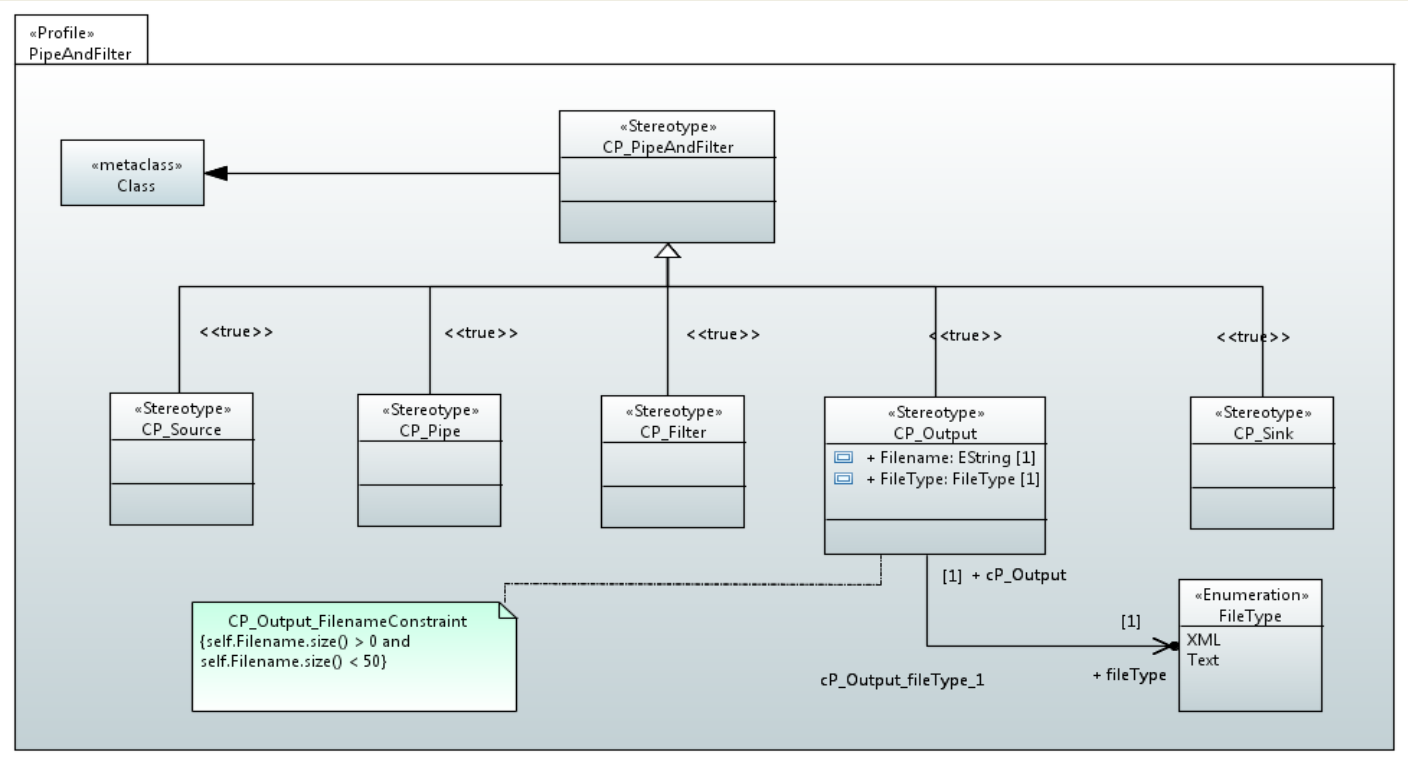
Package Diagram:

Architecture:

(MVC, Pipe And Filter)

checklist:	Yes	No
1.Are all the subsystem represented in the diagram ? completeness	✔	
2. Are all the architectures chosen aiding in solving a potential problem ? completeness/correctness	✔	
3. Have primary and secondary architecture been identified ? completeness	✔	
4. Have all the dependencies between the subsystem been identified in the diagram ? correctness/consistency	✔	
5. Is the diagram readable ?	✔	
6. Does each subsystem clearly identify with a profile? consistency		X
7. Have all the naming conventions been respected in the diagram? consistency	✔	
8. Does the entire system architecture promote low coupling? correctness/consistency	✔	

Validation of System Model



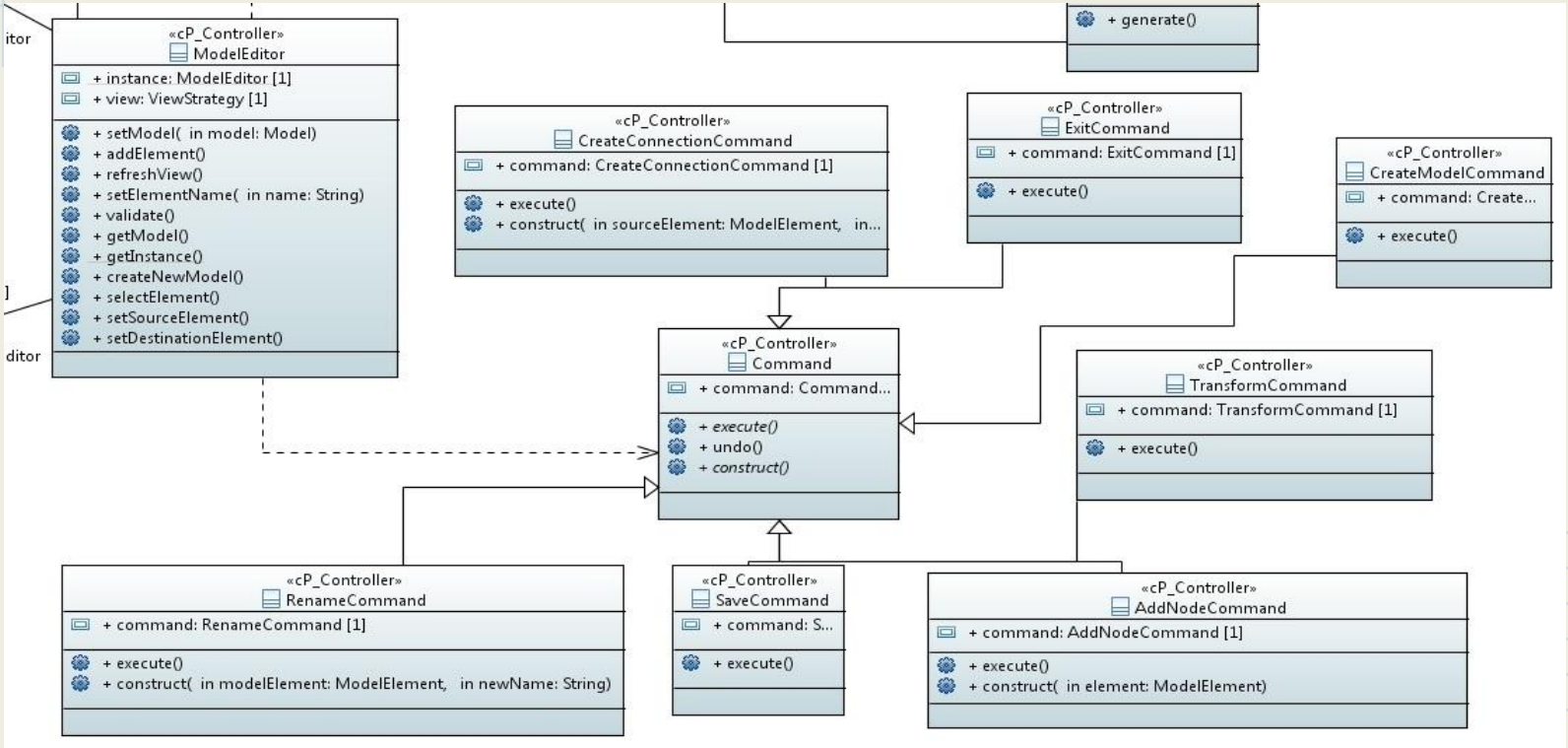
Validation of System Model

Profile:
Pipe & Filter

checklist:	**Yes**	**No**
1.Are all the architectural components represented in the diagram? completeness	✔	
2. Are all the components chosen aiding in solving a potential problem? completeness/correctness	✔	
3. Is the diagram readable?	✔	
4. Are all appropriate constraints defined according to OCL standards? consistency		X
5. Have all the naming conventions been respected in the diagram? consistency	✔	

Validation of Detailed Design

Controller Subsystem:



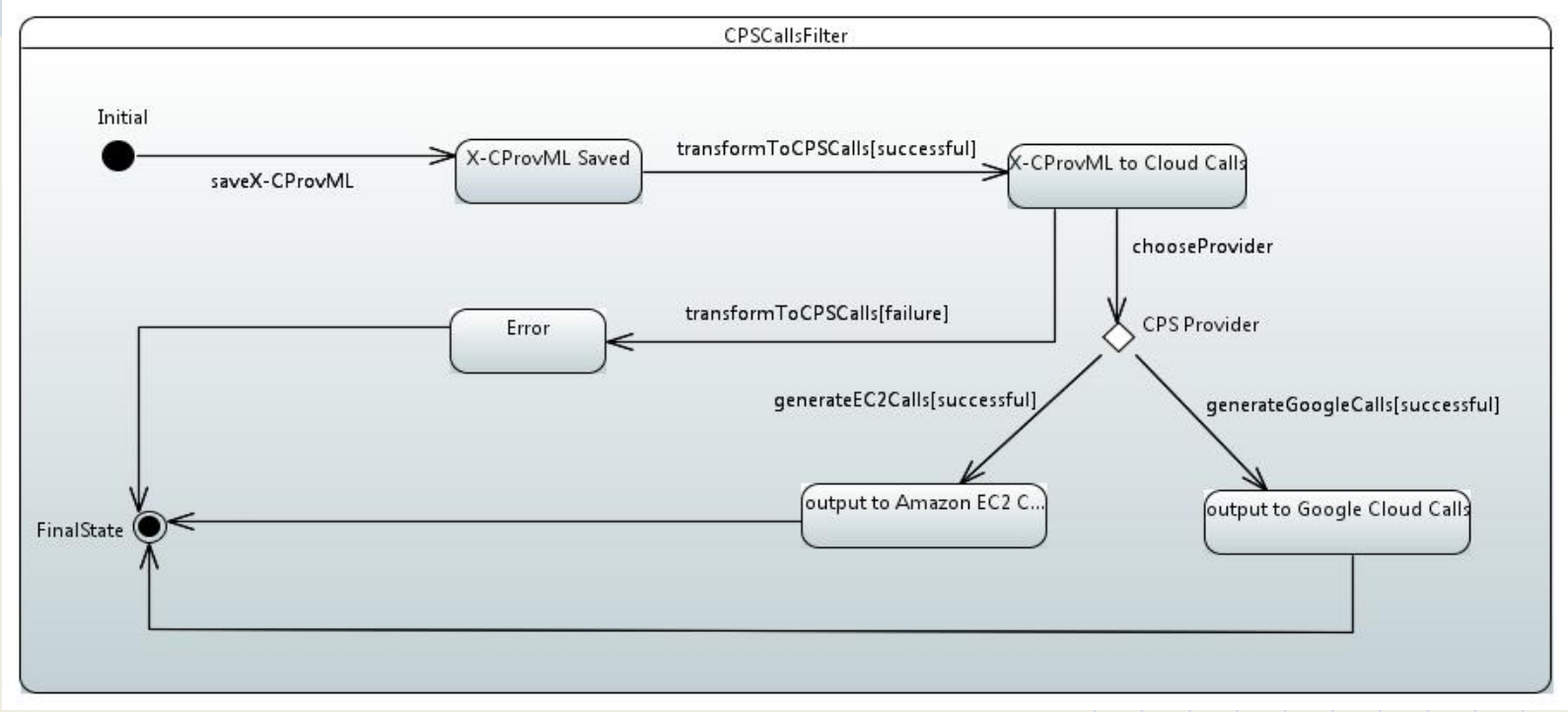
Validation of Detailed Design

Detailed Class Diagram Checklist

checklist:	Yes	No
1.Have all the necessary classes been identified in the diagram ? completeness	✔	
2. Have all the necessary methods for each class been identified in the diagram? completeness	✔	
3. Have the dependencies/Associations been chosen correctly in the diagram ? correctness/ completeness	✔	
4. Are all the necessary inheritance been represented in the diagram ? correctness/completeness	✔	
5. Is the diagram readable ?	✔	
6. Are all subsystem classes depicted in the diagram? completeness	✔	
7. Are all constraints identified according to UML 2.0 OCL representation ? consistency		X
8. Have all the stereotypes created in the profiles been used in the diagram ? consistency/completness	✔	
9. Do the classes identified in the diagram promote high cohesion ? consistency/correctness	✔	

Validation of Detailed Design

State Machine Diagram (CPSCalls Filter)



Validation of Detailed Design

State machine diagram checklist:

CPS calls Filter

checklist:	**Yes**	**No**
1.Does the diagram have a starting and final state? completeness/correctness	✔	
2. Have all the states been identified in the diagram ? completeness	✔	
3. Have all events/actions been used correctly in the diagram? correctness	✔	
4. Have all guards been identified in the diagram ? completeness	✔	
5. Is the diagram readable ?	✔	
6. Have all alternate paths been identified in the diagram ? completeness/consistency	✔	
7. Have all the naming convention been respected in the diagram? consistency	✔	

Validation of Implementation

Test Cases ID	Result
CPROVME001-TC-S001- Create New Model	PASS
CPROVME001-TC-S002- Create New Model	PASS
CPROVME001-TC-R003- Create New Model	PASS
CPROVME005-TC-S004- Add Instance Node	PASS
CPROVME005-TC-S005- Add Instance Node	PASS
CPROVME005-TC-R006- Add Instance Node	PASS
CPROVME014-TC-S007- Add Connection	PASS
CPROVME014-TC-S008- Add Connection	PASS
CPROVME014-TC-R009- Add Connection	PASS

Validation of Implementation

CPROVME009-TC-S010- Edit Instance Properties	PASS
CPROVME009-TC-S011- Edit Instance Properties	PASS
CPROVME009-TC-R012- Edit Instance Properties	PASS
CPROVME013-TC-S013- Delete Node	PASS
CPROVME013-TC-S014- Delete Node	PASS
CPROVME013-TC-R015- Delete Node	PASS
CPROVME018-TC-S016- Save Model	PASS
CPROVME018-TC-S017- Save Model	PASS
CPROVME018-TC-R018- Save Model	PASS
CPROVEE002-TC-S019- Transform to CPROV XML	PASS

Validation of Implementation

CPROVEE002-TC-S020- Transform to CPROV XML	PASS
CPROVEE002-TC-R021- Transform to CPROV XML	PASS
CPROVEE003-TC-S022- Validation Against CPS Schema	PASS
CPROVEE003-TC-S023- Validation Against CPS Schema	FAIL
CPROVEE003-TC-R024- Validation Against CPS Schema	PASS
CPROVEE004-TC-S025- Transformation to Amazon EC2 format	FAIL
CPROVEE004-TC-S026- Transformation to Amazon EC2 format	FAIL
CPROVEE004-TC-R027- Transformation to Amazon EC2 format	FAIL