HW03: Evaluations

Team 5

Lecture 17

Organization

lecture 17 UML submission and readme is well organized as requested.

UML Eval

Documentation: Poor 60%

Major sections of UML diagram (design parts that pertain to main functionality and to main specifications) are either missing or lacking any meaningful content. There is barely sufficient documentation. I can see team 5 is making their project in an MC-V pattern. But the structure of it is very confusing. Such that their Model has three methods inside with three instance names, which is incorrect. In MVC, the Controller should point at Model and View. Controller handles instance(parameter), Model handles the performance with the instance from controller, last View displays the result from using the instance in controller with the method from model. What is more, View and Controller are not connected causing more confusion.

Additionally, Interface RMI shouldn't be at the middle, since it shows Controller, view, model is going to implement RMI, but it's not. Then there are lack of interface clarification, such as class remote, interface for model, class unicastRemoteObject for model, class observable, interface observer and so on.

Last, Team 5 didn't put any instance, and methods in each class that they clarity in the UML.

Specifications: Poor 60%

Significant details of the design specification are violated, and/or significant design instructions are either ignored or entirely misunderstood. There is a confusion with so many arrows pointing from controller to the method in model, it's like class Model is never been used. Then the arrow from server to controller, it's like the server is using class controller, which is wrong, since server is not a class. Same as the arrow between RMI Client and View, RMI Client is not a class. Method Label in View has no arrow points from or points to. However, The UML has a bad formatting according to the UML class diagram tuition, which classes and methods are not showing as they should be.

0.6 * 0.5 + 0.6 * 0.5 = 60 pts

Lecture 18

Organization

Lecture 18 UML submission and readme is well organized as requested.

Documentation: Poor 60%

Major sections of UML diagram (design parts that pertain to main functionality and to main specifications) are either missing or lacking any meaningful content.
There is barely sufficient documentation. Such as that Controller and Model are not connected, class View is using interface Observer, and class Observable is not being used, which is wrong, since interface Observer should be implemented by Controller, and Model extends class Observable. Also, class unicastRemoteObject is not yet shown in the UML, class unicastRemoteObject should extend by Model, and that is very important. Therefore, this team didn't detail and clearly show everything they use in their project to the Class diagram, and that caused a larger confusion to people who read their UML.

Specification: Adequate 80%
Minor details of the design specification are violated or misunderstood. Two design patterns they choose are MVC and Observer, there is a problem that they did not clearly indicate which side is the server. What is more, the class GameOfLifeView uses interface Observer is wrong, since GameOfLifeView is only for display grid. There is also a syntax error in the controller, since one class can't implement remote and Observer at the same time, plus remote is a class not an interface, so it can't be implemented by a class. At last, it's better to make some of the classes private if they are not used by other classes, so that is more safe.

0.6 * 0.5 + 0.8 * 0.5 = 70 pts

Lab 10

Organization
Lab10 is UML submission is here, but the readme file is not found. **- 8%**

<u>UML Eval(This UML is exactly the same as lecture 18 UML)</u>
Documentation: Adequate 80%
Major sections of UML diagram (design parts that pertain to main functionality and to main specifications) are either missing or lacking any meaningful content.
There is barely sufficient documentation. Such as that Controller and Model are not connected, class View is using interface Observer, and class Observable is not being used, which is wrong, since interface Observer should be implemented by Controller, and Model extends class Observable. Also, class unicastRemoteObject is not yet shown in the UML, class unicastRemoteObject should extend by Model, and that is very important. Therefore, this team didn't detail and clearly show everything they use in their project to the Class diagram, and that caused a larger confusion to people who read their UML.

Specifications: Adequate 80%
Minor details of the design specification are violated or misunderstood. Two design patterns they choose are MVC and Observer, there is a problem that they did not clearly indicate which side is the server. What is more, the class GameOfLifeView uses interface Observer is wrong, since GameOfLifeView is only for display grid. There is also a syntax error in the controller, since one class can't implement remote and Observer at the same time, plus remote is a class

not an interface, so it can't be implemented by a class. At last, it's better to make some of the classes private if they are not used by other classes, so that is more safe.

0.8 * 0.5 + 0.8 * 0.5 - 0.08 = 72 points

Team 6

Lecture 17

Organization
lecture 17 is well organized as requested.

UML Eval
Documentation: Poor 60%
Based on the diagram I am not sure what the client views are connected to and how the clientview connects to the server. Not sure how the interface can get updates when it's only connected to the server. None of the variables are marked if they are public or private. Poor documentation.

Specifications: Insufficient 20%
It is not clear at all how the server and client link conceptually. Furthermore, there is no model and controller as asked for in the documentation.

0.6 * 0.5 + 0.2 * 0.5 = 40 points

Lecture 18

Organization
Lecture18 is UML submission is here, but the readme file is not found. **- 4%**

UML Eval
Documentation: Adequate 80%
One or two design aspects are missing, and there is sufficient documentation. For instance they have everything listed out as their instance for their project, also they have listed every method that they used in the project. It is looking good that they have a RMI stand in the middle of the UML, and have arrows pointing to each other, such as the arrow between the interface and controller. But there is a problem with class Observable, they wrote it as an interface.

Specifications: Adequate 80%

For the specifications, there are some minor things worth mentioning, which is the confusion with how they initial a 2d Array with next shape and current shape. Since they are not the right syntax with Java.
There is also an error with an interface ServerIntf implementing another interface remote. Since Interface can extend but implement, however, remote is class, can't be implemented.
For the methods inside, some of those shouldn't be public since they are used only with class.
Additionally, there are some minor issues with the arrows too, which should use a dashed line with an empty triangle for implementation, and use a straight line with an empty triangle for inheritance.
Next, the arrow between controller and model, and controller and view are wrong, since the arrow should only point from controller to model, controller to view instead of pointing to each other.
Last, the model cannot extend the unicastRemoteObject and Observable at the same time.

0.8 * 0.5 + 0.8 * 0.5 - 0.04 = 76 points

Lab 10

Organization
Lab10 is UML submission is here, but the readme file is not found. - 8%

UML Eval(This UML is exactly the same as lecture 18 UML)

Documentation: Adequate 80%
Significant details of the design specification are violated, and/or significant design instructions are either ignored or entirely misunderstood. They clearly indicate which side is the server. However, there are some mistakes. First, the observable is written as an interface. In fact, observable is a class. Second, the model cannot extend the unicastRemoteObject and Observable at the same time.

Specifications: Adequate 80%
Minor details of the design specification are violated or misunderstood. They follow mostly the requests of lab10, and two design patterns they choose are MVC and Observer. Furthermore, they also clearly indicate that which side is the server and which side is the client, and we can see that the remote object pattern they choose is M-VC. Just have some errors with the arrows, and some syntax problems with the instance they create, such as interface ServerIntf implementing another interface remote. Since Interface can extend but implement, however, remote is class, can't be implemented. So use a straight line with an empty triangle for inheritance. For more detail, you can see from the comment from lecture 18.

0.8 * 0.5 + 0.8 * 0.5 - 0.08 = 72 points

Team 7

Lecture 17

Organization

Lecture17 is well organized as requested, but the readme file is not found. **- 2%**

UML Eval

Documentation: Adequate 80%

One or two design aspects are missing, and there is sufficient documentation. As I see in the UML, the only problem is they have not yet clarified class Observable and interface Observer. Anything else is understandable with all the instances and methods showing. Also, there is one thing worth mentioning, they have private methods and public methods separated in each class. It makes their UML shows more details and makes their project more clearly to everyone.

Specifications: Excellent 100%

No design specification parts are missing except they have not yet clarified class Observable and interface Observer. Other than that they did a good job with every single arrow between class and class or class and interface. It's simple and clean.

0.8 * 0.5 + 1.0 * 0.5 - 0.02= 88 points

Lecture 18

Organization

Lecture18 is well organized as requested, but the readme file is not found. **- 4%**

UML Eval

Documentation: Excellent 100%

UML diagram is well-prepared, and clearly understandable. As I see in the UML, there is no major part missing and everything is understandable with all the instances and methods showing. Also, there is one thing worth mentioning, they have private methods and public methods separated in each class. It makes their UML shows more details and makes their project more clearly to everyone.

Specifications: Poor 60%

Minor details of the design specification are violated or misunderstood in this case. Since I saw there are many syntax errors in the uml, such as the arrow between GameOfLifeModel and Node, and Node and CellObsever. I don't understand what is the relationship between those two from that arrow, since there is no header to point on both sides.
Furthermore, class Node can't extends interface Observer and extends class Observable at the same time. However, interfaces can't be extended by a class. Else things are clear and detailed.

1.0 * 0.5 + 0.6 * 0.5 - 0.04 = 76 points

Lab 10

Organization

Lab10 is well organized as requested, but the readme file is not found. **- 8%**

UML Eval

Documentation: **Adequate 80%**

One or two design aspects are missing, and there is sufficient documentation. They have not yet clarified class Observable and interface Observer. Other than that, this UML diagram has sufficient details and is really easy to understand. In this diagram, their instance variables and methods are set as public and private as well. Moreover, the UML arrows are also used correctly. For example, they use an implementation arrow to connect the GameOfLifeController and ActionListener.

Specifications: **Excellent 100%**

No design specification parts are missing except they have not yet clarified class Observable and interface Observer. Other than that they do not miss any request of lab10, and they also clearly point out which side is the server. Additionally, to contrast with other teams, they choose two different design patterns to do this lab, and those two design patterns and Decorator and MVC, respectively. Furthermore, the Remote Object Pattern they choose is M-VC.

0.8 * 0.5 + 1.0 * 0.5 - 0.08 = 82 points

Team 8

Lecture 17

Organization

Organization is correct for the lecture 17 UML submission and readme.

UML Eval

Documentation: Adequate

Based on your UML it seems like there can only be two clients at the same time. There should be multiplicities to show this. Not sure why everything is public and also what the remote does because there's no variables or methods.

Specifications: Adequate

Everything was met except for the fact that you can't tell which side is the server and client side.

0.8 * 0.5 + 0.8 * 0.5 = 80 points

Lecture 18

Organization

Lecture17 UML submission is here, but the readme file is not found. **- 4%**

UML Eval

Documentation: Adequate 80%

One or two design aspects are missing, and there is sufficient documentation. As I see in the UML. In their case, they use an M-VC pattern, so that they have a model in their RMI. But there are few major parts missing. Such are the interface Observer, class Observable, and class UnicastRemoteObject. It will be more detailed and clear if they mention those all the classes that they use in their project.

Specifications: Poor 60%

Significant details of the design specification are violated, and/or significant design instructions are either ignored or entirely misunderstood in this case. Since I saw there are many syntax errors in the uml, such as the arrow between GameOfLifeModel interface and Remote class, it should be a straight line with an empty triangle shows that GameOfLifeModel interface extends class Remote. Also with class DeadCell and class Cell, and class AliveCell and class Cell. The relationship between class AliveCell and class DeadCell with class Cell can't be implemented, since neither three of them are interfaces. Therefore, the arrows in here should be a straight line with an empty triangle.

What is more, it will be more clear if they put a line RMI in the middle like the instruction said, and put GameOfLifeModel with every class and interface related to it on the left.

At last, there is an issue with Model, Controller, and View. Just remember that Controller updates view, and Controller notify Model. Model and View don't do anything. So there should not be arrows pointing back from either Model or View to Controller.

0.8 * 0.5 + 0.6 * 0.5 - 0.2 = 68 points

Lab 10

Organization

Lab10 UML submission is here, but the readme file is not found. **- 8%**

UML Eval(This UML is exactly the same as lecture 18 UML)

Documentation: **Adequate 80%**

It is easy to understand, but there are some small mistakes. First, In this UML diagram, they did not indicate which class extends unicastRemoteObject. In the RMI pattern, unicastRemoteObject is a very important component. Thus, I think they have to clearly indicate that.

Specifications: **Poor 60%**

Significant details of the design specification are violated, and/or significant design instructions are either ignored or entirely misunderstood in this case. Two design patterns they choose are Factory and MVC. Nonetheless, I just mentioned that they incorrectly use the Factory and MVC design patterns(M - V). Furthermore, In MVC, Model cannot have a relationship with view. However, their model class is connected with a cell class. In the cell class, there is a method called paintComponent. Thus, their design pattern is incorrectly implemented. Finally, they also made all instance variables and methods public.
Next, there are many syntax errors in the uml, such as the arrow between GameOfLifeModel interface and Remote class, it should be a straight line with an empty triangle shows that GameOfLifeModel interface extends class Remote as the comment in lecture 18.

0.6 * 0.5 + 0.8 * 0.5 - 0.08 = 62 points