Software Modelling and Design

Project 3B Reflection

Putting It All Together

The University of Melbourne

Group Details:

Group Number : 18

Adrian Sutanahadi	617462
Denis Thamrin	613726
Leah Park	625302
Md Zahidul Islam	618019

Changes made to the Original Design:

There were not a lot of changes that we made to our original design for the development of the application. However, though our design was good enough to stick with, there were few changes that we made to make it easier to implement and makes the code clearer.

We made some changes in the controller by putting an interface to the controller. Previously the interface is separated from the controller but after some discussion with the tutor we decided to add it to the controller. So in our diagrams we have just added an interface to the controller.

In addition, we have also separated the prediction class for each of the attributes like wind, rainfall, and temperature which ensures more extensibility.

In the UML, we had a lot of method overloading (One function different type signature) but ruby generally does not support it so we need to have two function with different name.

These ideas came out through the discussion with the tutor and from the feedback received for part A. So we agreed on making these small changes which makes more sense for the development of the entire application and ensures more extensibility.

Aspects we found challenging:

The implementation involves a lots of manual coding of ruby on rails which has proved to be pretty time demanding and challenging by itself.

In addition, the implementation part turned out to be more challenging due to the lack of contribution by one of the group members.

Thoughts on the value of spending time and effort developing a thorough design prior to implementation:

Planning is really important for anything to achieve successfully. While developing any application, we do this planning through design. We don't get the solution in one go. We have to make changes, learn and implement. And it's easier to change the design than to change the code since we doesn't have any data sitting in the database when developing them. So we can easily make changes to get the best solution if we design first.

Besides, as we work on the design first, we learned more and get more ideas on how to solve the problem and how we can make changes to get the best ideas. Moreover, designing first give us more maintainability as well.

However, Rails is difficult to be grasped by poor coders, it has too much magic and convention. Not understanding the convention when planning proves us costly.