# Individual Milestone 5 Report

## Executive Summary

For this milestone, all the MSE students are regrouped together focusing on the plugin task.I was assigned to the decomposing subgroup. Due to it was my first time to get in touch with the plugin related work,I spent the first week of this sprint in understanding the structure of earth plugin and how earth plugin worked on earth. In the second week ,I thought about how to decomposing the existing plugin filemonitor into apis and then I self-assigned myself to the work of security part of plugin,including creating certificate and keys, sign plugin with the key and put everything into database.However,when I was working on it, I found a bug,but I was not sure whether the bug was from, the ruby::openssl or ruby postgres driver.Then Ken decided to solve the problem by using string to represent the signature rather than the binary.I also spent a lot of time on troubleshooting, The main trouble was that my earth running on Aptenna and earth running in console had a conflict.

## Tasks and Activities Assigned

- Create plugin APIs in term of Ken's API standard.
  - Sub-task 1: Have a research on earth plugin.
    - Description:The research includes what is earth plugin , the relationships among filemonitor and other parts of earth, namely ,how they call and work with one another and how plugins work on earth,rather how to create a plugin and how to install a plugin.
    - Affected files:
      `earth/lib/earth_plugins`
      `file_file_monitor.rb`
      `earth/lib/earth_plugin_interface`
      `plugin_manager.rb`
      `earth/script/earthd`
      `earth/script/create_cert`
      `earth/script/sign_plugin`
      `earth/script/install_plugin`
    - Estimated time taken (planned):20 hours
    - Estimated time taken (actual): 21 hours
  - Sub-task 2:Working with group members to decompose the plugin APIs .
    - Description: Discussing with group members and In terms of Ken's API standard, we decomposed the existing plugin file_monitor.rb into APIs.
    - Affected files:
      file_monitor.rb
      api_eta_printer.rb
      api_file_monitor.rb
    - Estimated time taken (planned):10 hours
    - Estimated time taken (actual): 10 hours
- Fixing the conflict between earth running on Aptena and earth running in console.

- Sub-task 1: Find out the reason why I cannot run earth in console by using the command ../script/earthd start
  - Description:I found the socket was occupied by Aptena ,so that earth cannot work on console.
  - Affected files:
  - /tmp/earthd.sock
  - Estimated time taken (planned):6 hours
  - Estimated time taken (actual): 10hours
- Sub-task 2: Uninstall the Aptena Studio plugin on Eclipse
  - Description: At first I did not know how to uninstall it.However, it was done finally.
  - Estimated time taken (planned):5hour
  - Estimated time taken (actual): 7hours

- Work on the security part.
  - Sub-task 1: Learn how to sign and install a plugin.
    - Description: This task included creating certificate,private and public key pair, sign the code of plugin with the private key ,and verify the code and it's signature with the public key in certifucate, when installing a plugin. I spent a lot of time to read the plugin_manager and earthd and learned how to use openssl to do every thing, and felt comfortable to write all the security related functionalities by myself.It indeed took me a lot of time.But When I read Ken's plugin notes,I knew that all I had done were just waste of time.RSP have already finished everything for us.What we need to do was typing create_cert, sign_plugin and install_plugin in command line.
    - Affected files:
      `earth/lib/earth_plugin_interface/plugin_manager.rb`
      `earth/script/earthd`
      `earth/script/create_cert`
      `earth/script/sign_plugin`
      `earth/script/install_plugin`
    - Estimated time taken (planed): 30 hour
    - Estimated time taken (actual): 20 hour
  - Sub-task 2: Put the code,signature, and plugin's name and version into database. And install file monitor as a plugin.
    - Description: Both Keane and I found that if we put the original file_monitor into the database, when we ran ./script/install_plugin the signature can be converted properly which was the same as the code in file_monitor.rb. But once we broke the it into APIs and call the methods that we need from the new created file_monitor file, everything changed. The weired problem is all the '\n' was convered into '\012',so that the code and signature pair could not pass the verification. I spent huge amount of time in testing.I compared the original file_monitor with the new created file_monitor and try to find out the source of the problem. I broke down the original file_monitor little by little,namely the size of file_monitor was becoming smaller and smaller. At first it could work properly, all the '\n' signals were converted in the right way. But when I keeping making its size smaller. The file_monitor was broken by sudden. Therefore, I was sure that this problem depanded on the size of the

file. I considered that it was a bug However,I was not sure where the bug was from, openssl or postgres driver. And finally Ken solved this problem by using string into code and signature in database, instead of binary.

- Estimated time taken (planned):10hour
- Estimated time taken (actual): 25hours

## Resource Contributions

For this milestone, I have estimated that I had spend about 95 hours (including 4 hours meeting time) over the spread of 3 weeks.

## Rooms for Improvement

- We have a big problem in the time and difficulty estimation. And tasks did not assigned to each team member properly.I found that at least half of the team members did not feel comfortable to work in the group.
  - Review more on past milestones, and perform reality checks during planning.
  - Better communication with other members. Team members should trust one another,and be more helpful. Estimation efficiency should be improved.