

### **Seminar 3 - FoodCloud**

This seminar was given by Roy Philips who is the chief technology officer at FoodCloud. FoodCloud was set up to solve the problem of food wastage that occurs daily in supermarket chains. It is a system that acts as an intermediary between the supermarket and a charity to reduce the cost of food disposal by creating an incentive for the supermarket to donate food near the expiry to charities. Food cloud is a way of managing the donations process. The incentive that draws supermarkets into donating food is that it both creates positive media profile because they are donating to a good cause, it has a positive effect on the environment because food doesn't need to go to a waste pile and it also alleviates cost of food disposal as it's cheaper for companies to have someone eat surplus food. One of the key aspects of this platform is its fairness algorithm which is hoped tries to ensure the same charities does not always get the best foods or the same foods. This means every charity has equal opportunity to benefit from the donations. Another very clever aspect was to include an impact interface which allows companies to visualize how much food was donated by individual stores and quantify the amount of meals it relates to when the charities utilize their donations

#### **Software Methodology:**

The software methodology that is applied in FoodCloud is based from an agile framework and the tools and methods that are used are very flexible. Scrum Pair Programming and Kanban which is a process management tool are all utilized. Kanban allows programmers to pull work as the programmer's capacity permit rather than when it is requested. Food cloud operate using a small 7-person team each is responsible for their own area, so the android developer has control over the android app and so on. They also work on stories which are like tasks where they only take on at most 4 stories at a time. This platform is in its infancy, so their architecture is very much being developed on the fly with adjustments being made at each iteration. Originally it was built as a dual, module with one module acting as the donations processing centre and the other taking care of the communications gateway between the donator and the charity. However, due to more traffic on the servers and more requests these 2 modules would be later abstracted out to create more modules that adhere better to the single responsibility principles. This will allow the future platform to scale more effectively as it gets more users from home and abroad. This platform is written in Scala and ReactJS makes use of the Heroku platform as a service which helps in the management of databases and it also makes use of the Akka streams which adopts the let it crash at has set instructions for recovering from a crash.

#### **Use of Modelling:**

Roy uses google slides as a mock up tool to make a more informal UML diagram to give a fundamental basic concept of what the team are trying to achieve overall. Google slides is a nice way of Interacting with other members of the team which can be useful when sharing ideas or making comments and suggestions on the overall design of the FoodCloud platform. These informal diagrams provide a basic idea of how the system should work to the other developers,

so they can provide the iOS and android apps in a similar fashion. Their development process tries not to predict the future or make estimates of unknown factors because a lot of what they are achieving is can't be summarized in hard and fast knowledge. It is much more continuous and changing environment.

### **Testing:**

Test coverage was originally very good but as the relational database grew it became very slow to test and subsequently this was ignored for some time. However more recently they have switched to H2 database which supports the original PostgreSQL framework this makes it easier to test because in memory table can be created as well as disk-based tables. Now they have good test coverage again which is kept up to date. They have been quite lucky over their development that they haven't encountered too many bugs. So, it has worked well for them.

### **Software Quality & Refactoring**

Software quality and refactoring is less of an issue in a platform like this due to the size of the team and the fact that each person is fundamentally responsible for an aspect of the platform they are usually writing code for each part on their own or maybe some pair programming. The refactoring process is done but especially now where they are abstracting out the modules to adhere to the single responsibility principle. However, they really work on a make it work make it elegant then forget about it. Where possible the rule is they try and make the methods as meaningful as possible as they go and correct/refactor as they go which is just part of the continuous develop and deploy cycle

### **General Comments**

Roy enjoys working in a small team as its quite focused on what you are responsible for and there is a lot of creative control when there are less parties involved. FoodCloud development process is way more unstructured than that of large companies this is probably due to mainly one to two people working on a module. This means that these developers have massive knowledge on the internal working of the system and are highly skilled in their area. They collaborate but from a perspective of the general direction and flow of the project and can assist each other in coding aspects where needed but at the end of the day for most of the time they are delivering massive chunks of the platform by using small amounts of pair programming but mostly solo programming.