 

**Agile Java Development Code Camp 2012**

Report on the Project Work:

Workload Forecast System

Team 5:

Mihai Iusan

Svetlana Afanasyeva

Ivan Verdezoto

Lappeenranta, 02.11.2012

Table of contents

[1. Aim of the project 3](#_Toc339598285)

[2. Background 3](#_Toc339598286)

[3. User stories 3](#_Toc339598287)

[3.1. User 3](#_Toc339598288)

[3.2. Consultant 4](#_Toc339598289)

[3.3. Manager 4](#_Toc339598290)

[3.4. Director, resourcing and controllers 4](#_Toc339598291)

[4. Sprint Backlogs 4](#_Toc339598292)

[5. Web application architecture 5](#_Toc339598293)

[6. Scrum process 6](#_Toc339598294)

[7. Program documentation 8](#_Toc339598295)

# Aim of the project

The aim of the Code Camp project is to develop web application using Grails. Grails is a model-view-controller (MVC) framework that is built on the Java platform, i.e. it runs on any Java application server. The main advantage is that Grails development environment has pre-integrated components, which help in the development process. Grails itself are based on the Groovy language.

Another goal of the Code Camp was to learn and experience the project development process. It has been proposed for teams to build workload forecast system while following so called Scrum method. Briefly, Scrum is the development process composed of sprints cycles, which are usually subdivided on the daily sprints. Basically, sprint is a team meeting on which the current stage of the project reviewed and decisions on the following development are maid. More information on the development process will be given in the following chapter.

# Background

Capgemini represented a customer for the Code Camp project. Profitability of the company highly depends on the employment rate of its employees. It is a big scale consulting company, which has difficulties of forecasting company’s employees workload hours. At the moment Excel table is used, which is obviously not an optimal solution. Capgemini gave the task to the Code Camp teams to implement workload forecast system. The purpose of the project is to create web application to resolve problem of the forecasting process.

Every team received workload forecast system objective description where functionality and requirements of the project have been described. Usually the requirements which the system should satisfy are developed by the group working on the project. But due to the Code Camp tough schedule they were given by the customer itself.

Several types of the web application should be created according to the system objectives, such as consultants, managers, controlling and directors. Consultants of the company are responsible themselves to input their workload forecast hours. Supervisors should be able to review the forecast and make corrections.

# User stories

First step of the development process is to create the user stories, which are based on the requirements list. Stories, common for every type of web system users, are presented in the beginning and followed by specified stories for consultant, manager, director, etc. Each story was given the priority. It is represented at the sprint backlog.

## User

* As a user, I want to login
* As a user, I want to browse forecast data
* As a user, I want to update forecast data
* As a user, I want to remove activities which is not used anymore in workload estimates
* As a user, I want to receive a message if I book over 100% of total workload
* As a user, I want to update data at the same time with others
* As a user, I want to see ARVE(Assignment Rate vacation excluded)
* As a user, I want to see URVE(Utilization Rate vacation excluded)
* As a user, I want to see COR(Charge-Out-Rate)
* As a user, I want to browse employee forecasts
* As a user, I want to have the workload data save in the Database
* As a user, I want to be able to see the history of all the modifications

## Consultant

* As a consultant, I want to see a list of activities, so that I can input my forecast information.
* As a consultant, I want to input my forecast information in one or many activities at the same time.
* As a consultant, I want to input hours or days per week to different activities.
* As a consultant, I want to input hours or days per month to different activities.

## Manager

* As a manager, I want to input price of engagements of consultants per activity.
* As a manager, I want to input customer information.
* As a manager, I want to input activity types.
* As a manager, I want to input activity information.
* As a manager, I want to view a list of consultants with forecasted workload per activities.
* As a manager, I want to edit forecast information of consultants so that I can adjust workload of people for activities needs and priorities.

## Director, resourcing and controllers

As a director, I want to view forecast workload information of activities, so that I can coordinate resourcing activities for the unit.

# Sprint Backlogs

Team had daily sprints where updates on the project development process were made. From the list of user stories the most important ones has been chosen to implement. Each story was given the priority, which also indicates the aims for weekly sprints. It is represented in the table 1.

Table 1. User stories hierarchy

|  |  |  |
| --- | --- | --- |
| N | User Story | Priority |
| **US001** | As a user, I want to login/ logout | 1 |
| **US002** | As a user I want to register to the system | 1 |
| **US003** | As a user I want to choose my type (manager, consultant, controller, director) | 1 |
| **US004** | As a manager I want to create activity | 1 |
| **US005** | As a manager I want to input customer | 1 |
| **US006** | As a consultant I want to input/update my forecast hours | 2 |
| **US007** | As a manager I want to make changes in the forecast hours of consultants | 2 |
| **US008** | As a controller I want to make instructions | 2 |
| **US009** | As a user, I want to see ARVE(Assignment Rate vacation excluded), URVE(Utilization Rate vacation excluded), COR(Charge-Out-Rate) | 3 |
| **US010** | As a user I want to see modifications history | 3 |
| **US011** | As a user, I want to receive a message if I book over 100% of total workload | 3 |
| **US012** | As a user I want to see/print reports on work load forecast | 3 |

# Web application architecture

Team has created the architecture of the web workload forecast system, which is depicted at the figure below. Forwarding information from one unit of the application to another is shown.

****

Fig. 1. Program architecture

# Scrum process

Our team consists of three members. Roles in the project development have been distributed in the following way:

1. Mihai Iusan: lead developer;
2. Svetlana Afanasyeva: programmer-tester;
3. Ivan Verdezoto: scrum leader, programmer-tester.

Every member of the team has been participating in the creation of the user stories. The aim was to fulfill the requirements of the project. To follow the SCRUM scheme the group had meetings every morning, where updates on the project development were made.

Grails application was a new framework for each member of the team that is why learning of the platform has been put in the schedule.

The main functionality of the web system has been done by Mihai Iusan. For Svetlana Afanasyeva it has been the first introduction to the Java application, buts she also contributed in auditing some functionality. Ivan Verdezoto has been testing the system and playing an active role in the Scrum process. He has prepared the presentation of the team.

Table 2. Status of the Scrum process

****

Burn down chart demonstrates remaining work on the project.



Fig. 2. Burn down chart

# Program documentation

The final program features are described and shown in this chapter. At the figures below all information required is demonstrated. Grin color indicated possibility to access the other web pages, which contain the information of user interest.

At the starting web page it is possible to login to the system. Also it is possible to register new user of the system.



At example shown below user is login into the workload forecasting system. Password is encrypted.



After user enters menu page, he has several options to choose:

1. “Activities”,
2. “Workload”
3. and “Clients”.



Main bar contains:

* Home: which directs user to the starting page
* Logout option
* List Users: where all users of the system are visible and it is possible to access information about them



It is possible to create New User from this page by the way shown below:



* Profile: where all information concerning user is given. From this page it is possible to access information about activities owned by the user.



Let’s look at each option separately:

1. “Activity” contains two sub options: “Activity list” and “Add Activity”.





1. “Workload” also contains two sub options: “Workload list” and “Add Workload”.





1. “Clients” contains two sub options: “Clients list” and “Add Clients”.



