Professional Software Development 3
Group Exercise 1 part 4:
Prototyping in an Initial Sprint

Jeremy Singer

Revision 4065, made 11/11/2013 by jsinger

1 Initial Sprint

The week commencing **Monday 11 November 2013** is the first sprint phase for your PSD3 group exercise 1[1]. There are two main objectives for this initial one-week sprint:

1. to get each team working collaboratively on implementation activities, using appropriate tools and techniques.

2. to explore uncertainty in several user requirements, by means of **spike prototypes**.

We will assess the outcomes of the sprint, with respect to these objectives, in post-sprint events: i.e. the internal team retrospective meeting and the client-focused review meeting.

2 Initial Sprint Backlog

Recall from the lectures on the Scrum development model that the sprint activities are recorded in a Sprint Backlog. Individual tasks are removed from this backlog and acted upon by team members.

Obviously, each team must construct its own unique sprint backlog. There are some general suggestions below:

1. select a language for prototype implementation work. (I would recommend Python, Ruby or Java.)

2. Set up source code repository for prototype code. (I would recommend git.)

3. Mock all components that do not require meaningful implementation in the prototypes. These components might include user authentication, student database, course database, student attendance records.

4. Implement a simple keyboard-driven, command-based, user interface.

5. Implement the two spike prototypes outlined below.

3 Spike Prototypes

In a dialogue with the client, the teams have identified two user stories which are poorly understood in terms of the required functionality. These stories have been selected as candidates for functional spike prototyping.

[1] The Computing Science Undergraduate Teaching Session Management System, see `http://moodle2.gla.ac.uk/mod/page/view.php?id=52475`

3.1 CSV Export Story

> As an admin, I want to export selected course data as a CSV file so that I can upload this data to other University information systems.

Conditions of Satisfaction

1. Exported CSV file can be read correctly by Microsoft Excel.
2. Admin should be able to export all student attendance records for a single course (course selected by admin).
3. Admin should be able to export all recorded information for a single student (like a transcript) (student selected by admin).
4. Exported CSV file conforms to specification given in moodle online documentation[2].

3.2 Attendance Monitoring Story

> As a tutor, I want to be able to record attendance of students at a particular session so that this data can be stored for university records.

Conditions of Satisfaction

1. Tutor selects a particular session, is presented with a list of expected students at that session. Tutor should be able to select which ones are present or absent interactively, via user interface.
2. Alternatively, tutor can upload a barcode scanner CSV file of student barcodes to the system, which selects which students are present or absent. This is displayed to the tutor for checking.
3. Both the above approaches generate the same internal data representation, which is stored persistently in the system and may be viewed/edited later by tutor or course lecturer or admin.

4 General Advice

This is your first sprint. It should be an interesting learning experience for your team. Start by constructing your sprint backlog and using planning poker to estimate task times. Record each individual task as a trac ticket. Tag each task as a `Sprint1Task`. When a team member starts working on a task, assign that task ticket to that member in trac. Then move the ticket through the trac ticket states, i.e. from `new` to `assigned` to `closed`.

Remember that the sprint lasts for **one week** only. You should aim to finish the sprint on Friday (or some time over the weekend.) Try to build up some team momentum. Make sure you have daily standup meetings, ideally just before lectures each morning. Try to spend some time every day on implementation work.

During the week, you should aim to get some client feedback. Each team has an allowance of *three email tokens*, for interacting with me or Sarah. Each time a team member sends an email to Sarah or me, this counts as one used token. Please indicate in your email subject header that this is a PSD3 sprint email from your team, e.g. `Subject: PSD3 client interaction from Team X`. Note that all email tokens expire on Friday 15 November at 1700.

[2] moodle grade import feature `http://docs.moodle.org/23/en/Grade_import`

By end of of the sprint week, you should have a better understanding of the functional requirements. Please document your fresh insights as comments on the appropriate trac tickets, or as separate wiki pages. You should also have generated a git repository with logs of your prototype development work. Both these items will be assessed, and will therefore form part of your final GE1 coursework portfolio.

We will have a sprint retrospective meeting next Monday morning, to assess how the teams' sprints fared. We will also facilitate initial (non-assessed) feedback next week, as part of the quality assurance exercise. This will involve peer review of your sprint outcomes by another team.