UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

AGILE DEVELOPMENT METHODOLOGY - SCRUM
iCrawler
Emilio Mumba

The 5 Concurrent Nodes

Khathutshelo Shaun Matidza
Sylvester Sandile Mpangane
Thabang Michael Letageng
Matthew Nel

Department of Computer Science, University of Pretoria

Oct 2015

Contents

1 Overview

This document is intended to serve as a reference for the development methodology used for the iCrawler mobile monitoring application, which was proposed by Mr. Emilio Mumba for Software Engineering (COS301) main project.

The application is primarily intended for parents who need to monitor the activities of their children on their mobile devices. Moreover it can be used to promote readiness in digital forensics, protect users from malicious entities and activities, and provide proactive measures that are undertaken by the mobile device user/owner. The application achieves this by monitoring the user's activities and collecting data logs from the device. The data is then persisted onto a database which can be viewed by logging into the dasboard component of the application which generates reports that give the user/owner a good understanding of what their mobile device(s) have been used for.

2 Roles

There are three core roles with a range of ancillary roles. These core roles are committed to the project in the scrum process.

2.1 Product Owner

Name: Emilio Raymond Mumba
Responsibilities: He is responsible for product vision; he accepts or rejects each product increment. He constantly re-prioritizes the Product Backlog, adjusting any longterm expectations such as release plans. He is the final arbiter of requirements questions.

2.2 Development Team

Name: The 5 Concurrent Nodes
Responsibilities: Intensely collaborative and cross-functional. They are responsible for delivering potentially shippable increments (PSIs) of product at the end of each sprint (the sprint goal). The team was initially made up of five individuals, however one member had to discontinue the course leaving four individuals to do the actual work.

2.3 Scrum Master

Name: Khathutshelo Shaun Matidza
Responsibilities: He is responsible for ensuring that the team follows the agreed scrum processes, facilitating key sessions, and encourages the team to improve. He enforces timeboxes. He is part of the development team

3 Events

3.1 Sprint

A sprint is the basic unit of development in scrum. It is restricted to a specific duration. The duration is fixed in advance for each sprint and is normally between one week and one month, with two weeks being the most common.

At the beginning of a sprint we hold a *sprint planning event*. This event takes place after every contact session with the client or module coordinators, which is usually every two weeks. During this planning we decide on what work needs to be done during the sprint duration. Our client (stakeholder) has access to our Git Hub repository, which at the end of each sprint he gets to see the current progress of the app.
When the sprint comes to an end we hold a *sprint review*. Here we review the work that was completed and the planned work that was not completed during the past sprint. We also present the completed work to the client/stakeholders (a.k.a *demo*).
The review is thus followed by a *sprint retrospective*. On this event we reflect on the past sprint; we identify and agree on continuous process improvement actions.

3.2 Daily Scrum

We hold a *daily scrum* (or stand-up) each day during a sprint to discuss what an individual did the day before, what they plan on doing on the present day and additionally if they see any impediments that might prevent them from reaching the sprint goal. The best time we opted for is after a lecture that we all share; this is to try and have all members to attend the daily scrum (although attendances of all members is not compulsory). The length of the daily scrum is constrained to 15 minutes at maximum., which explains why we stand.

4 Artifacts

4.1 Product backlog

The *product backlog* comprises pf an ordered list of *requirements* that a scrum team maintains for a product. It consists of features, bug fixes, non-functional requirements, etc.-whatever needs doing in order to successfully deliver a viable product.

Item	**Est. time**	**Priority**
User login	5hrs	High
User register	5hrs	High
Retrieve device info	5hrs	High
Database	20hrs	High
Retrieve installed apps info	5hrs	Low
Run in background	5hrs	High
Sms monitor	5hrs	High
Browser monitor	5hrs	High
Dashboard GUI	15hrs	Medium
User manual	30hrs	Medium
App GUI	5hrs	Medium
Call monitor	5hrs	High
Functional requirements	5hrs	High
Architectural requirements	5hrs	High
Password recovery	5hrs	Medium
Data encryption	5hrs	High
Location monitor	5hrs	low
Splash screen	10hrs	Medium
Data summary	20hrs	High
Retrieve wifi activities	6hrs	Low
App tour	6hrs	Medium
Development methodology	5hrs	High
Data usage	6hrs	Low

4.2 Sprint backlog

The *sprint backlog* is the list of work the development must address during the next sprint.

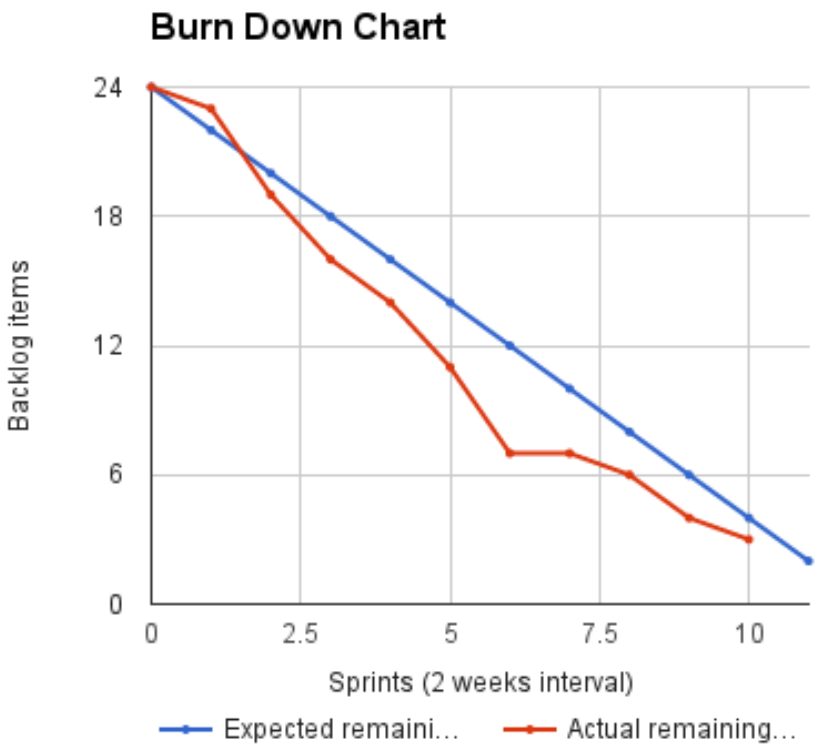
Item	**Duration**	**Feedback**
App GUI	3rd May - 15th May	Completed
User registration Run in background Functional requirements doc Database	17th May - 29th May	Completed
User login Dashboard GUI Retrieve device info Sms monitor	24th June - 24th July	Completed
Browser monitor Calls monitor	27th July - 7th Aug	Completed
Retrieve installed apps Data summary	9th Aug - 21 Aug	Completed
REST api password recovery	24th Aug - 4th Sept	Completed
Monitor device location User manual app tour guide	7th Sept - 2nd Oct	Completed
Retrieve wireless activity Data encryption Data usage	6th Oct - 16th Oct	In progress
Splash screen Development methodology doc Architectural requirement doc	19th Oct - 23th Oct	Incomplete

4.3 Product increment

The *increment* (or *potentially shippable increment*, PSI) is the sum of all the product backlog items completed during a sprint and all previous sprints. At the end of a sprint, the increment must be completed, according to the team's Definition of Done (DoD), and in a usable condition regardless of whether the product owner decides to actually release it.
Click here to go to gitHub repository

4.4 Sprint burn-down chart

The *sprint burndown chart* is a public displayed chart showing remaining work in the sprint backlog. It gives a simple view of the sprint progress. During the sprint planning the ideal burndown chart is plotted. During the sprint, each member picks up tasks from the sprint backlog and works on them. The burndown chart is updated day by day.



Click here for online burn-down chart

5 Terminology

The following explains some of the terms used in the scrum process.

Scrum team
Product owner, scrum master and development team

Product owner
The person responsible for maintaining the product backlog by representing the interests of the stakeholders, and ensuring the value of the work the development team does.

Scrum master
The person responsible for the scrum process, making sure it is used correctly and maximizing its benefits.

Development team
A cross-functional group of people responsible for delivering potentially shippable increments of product at the end of every sprint.

Sprint burn-down chart
Daily progress for a sprint over the sprint's length.

Product backlog
A prioritized list of high-level requirements.

Sprint backlog
A prioritized list of tasks to be completed during the sprint.

Sprint
A time period (typically 1-4 weeks) in which development occurs on a set of backlog items that the team has committed to.