Lessons learnt from converting a project from waterfall to Agile

Michael McGuinness[1]

[1]Griffith University, Brisbane, Australia, m.mcguinness@griffith.edu.au

INTRODUCTION

Griffith Universities eResearch Development and Consultancy Team works with researchers to help them achieve their IT goals. Since 2012 eResearch has supported a bespoke Randomisation service (create inhouse by eResearch). The Randomisation service is managed by the AVATAR Group, Menzies Health Institute Queensland, in partnership with the Health Group's Clinical Trials Unit and the Professor of Biostatistics.

In 2018, a replacement for this service was initiated. This would allow: the redevelopment of the solution using up to date technologies; the reuse of existing algorithms to ensure consistency; the ability to meet current development standards and best practices; increase the sustainability of the service by ensuring security and maintenance for next 3 - 5 years; and allow improvements necessary for accessibility on modern devices.

THE PROJECT

The project team lead by Michael McGuinness (Senior Analyst) was tasked with the replacement of this service in a bespoke manner with the reuse of code and functions where possible from the existing service. The first 3 months of the project consisted of a detailed requirement capturing phase. This required extensive time on tasks such as: having the wording just right for new page; and ensuing that every field of the new reports was detailed. This caused discussions lasting weeks and the creation of a very defined plan on exactly when and where everything would go.

Sounds great? Well not really, the project had a very small budget and given the time already spent and all requirements not being documented, something had to change. The above is what is commonly known as the waterfall project management methodology. This requires everything is documented upfront and then the project team goes away and creates a product as documented and returns many months later with the finished product with very little input along the way.

The project was changed to the Agile project management methodology. This allowed us to make changes very quickly and for the direction to be altered based on client feedback (we gave a demo every fortnight). We also used a just in time method where some of the technical details were not defined until we started working on the task.

LESSONS WE LEARNT

Some of the lessons we learnt when converting a project from waterfall to Agile were:

- Do any User Acceptance Testing at the end of each sprint!
    - This reduces the need spend long periods of time at the end doing undertaking all testing in one big bang

- Ensure that the sprint builds something tangible
  - This should be module based, once the core system is setup and running adding other modules should be like adding a brick of Lego (it just connects)
- Call delivery issues early, if we can't achieve a defined timeline, how can we manage it
  - Other than sprint planning meetings, try to have high level direction sessions with the development team to ensure that all of the work can done
  - Use of an Agile management tool (e.g. Jira) and have the development team estimate their tasks accurately
- Understand the rules of engagement with the approval boards (Just because you are using Agile does not change the rules)
  - Don't turn up to this boards underprepared, speak with the members before the meeting to call out any particular feedback that you can resolve before going for approval
- Expect resistance from project team
  - When something is new and unknown people will resist
  - Focus on selling them the benefits of the change
  - Ask them to just allow the process to happen and then happy to discuss if still not working
- Work with other technical teams early on, they are generally happy to help if you give them enough lead to time

IMPLEMENTATION

The replacement Randomisation service is due to go into production on 1ˢᵗ July 2019.