
Successful RAD and Prototyping on a Capstone Project

Dr Christo Potgieter

WINTEC
Tristram Street,
Hamilton New Zealand
christo.potgieter@wintec.ac.nz

Michael Stone (Student)

WINTEC
Tristram Street,
Hamilton New Zealand
linuxfornerds@hotmail.com

Abstract

This poster reports on experiences with Rapid Application Development (RAD) & Prototyping during a capstone project, considering CITRENZ (NACCQ) literature.

Keywords

Capstone project, RAD, Prototyping

Introduction

During 1998-2006, the annual conference contained presentations about capstone projects (Bridgeman, 2007). Problems with the development approach for capstone projects are apparent in a several aspects (Mann & Smith, 2006). Examples of approaches include the traditional SDLC

This poster paper appeared at the 2nd annual conference of Computing and Information Technology Research and Education New Zealand (CITRENZ2011) incorporating the 24th Annual Conference of the National Advisory Committee on Computing Qualifications, Rotorua, New Zealand, July 6-8. Samuel Mann and Michael Verhaart (Eds).

(ie. waterfall), incremental, prototyping, Rapid Application Development (RAD) and various agile methods (Strode & Clark, 2007).

Advice re capstone projects in CITRENZ

Mann & Smith (2004) proposed a framework from exemplar projects, that combines a software development life cycle approach with prototyping and early development:

- "SDLC" identified methodology but with prototyping, yet not replacing 'normal' design
- Functional requirements, tested with interactivity prototypes for early deliverable to client
- Prototypes used in communication with client and are part of integrated testing
- Maturation by revolutionary (cf evolutionary). Staged replacement for hardware
- Stable platform for development created early on
- In an educational setting the approach proposed has two groups of advantages: (a) Communication tool, client ownership, robust product through testing. (b) Overcome the disadvantage of inexperienced developers. It was observed that there is a high correlation between the process score and the grade, ie. a clear positive association between organization and outcomes.

Balance is required to avoid doing prototypes instead of normal design or alternatively maintaining "naïve" view of

SDLC which results in poorly tested designs. Work by weaker student groups tend to exhibit two characteristics:

- Developed a prototype and initially tested it, but this became the project with subsequent poor development
- Were poor at incorporating early development and testing into their SDLC approach

These are principles to be considered.

Our plan for success

The capstone project developed a practice self-testing tool for tutors, to run on a variety of mobile phones used by students. It was a first for everybody involved, includes low-level operations of equipment, targets a range of equipment and must be cheap to implement. However, it was mostly a "Proof of Concept", providing evidence of ability and baseline for enhancement.

RAD with Prototyping was chosen because it enables selective exploration to test operability during design and development. We followed the following principles:

- Planned the organization considering the above, creating "paper trace" and to publish as research
- Defined a classical SDLC with parallel component development and loops for exploration
- Do "Problem-Solving" for each technology challenge: Separate folder, standard template, report write-up
- Maintained a tight register of risks and problems, linked to loops and "Problem-Solving"
- Logged experiences, progress and client communication daily and event-driven

- Strong scope management to have a list of new and outstanding matters remaining at end of time
- Refined and tested components substantially so that they are reliable building blocks
- Student and supervisor met weekly to discuss status (no specifics of technical matters, client is expert)
- Smooth user interface had lower priority than proven technical ability

Did it work?

Project delivered satisfactory "Proof of Concept", overcame the critical technical challenges, remained on track re planned scope and time, produced useful records during and after project, delivered a product with documentation that is suitable for basic use and further refinement. Less was delivered than planned but it is tested and with documentation delivered can be completed with more time. Final marks not yet known at time of submission.

References

- Bridgeman, N. (2008). Capstone Projects – An NACCQ Retrospective.
- Mann, S. & Smith, L.G. (2004). Role of the development methodology and prototyping within capstone projects.
- Mann, S. & Smith, L. (2006). A value proposition model for capstone projects.
- Stone, M. (2011). Documentation and software re project "Tutorial creation tool for mobile devices"
- Strode, D & Clark, J. (2007). Methodology in Software Development Capstone Projects.