

Project management tips for the software developer

Ross Cooney

Codeworks Digital Bootcamp

11th November 2009

Introduction

- Ross Cooney
- Technical Director of Rozmic
 - Email and spam filtering applications
 - OS management
 - Datacenter management
 - External project management consultant

Examples of some projects

Software:

- Our emailcloud service is being used by 1200 SME's in the UK
- Netcleanse/Gwavix sold over \$1m in 2004/2005

Hardware:

- The firewall for American Express Ireland
- All web hardware for a UK police force
- Many servers for various hosting companies

Who are you?



Topics

- What is project management ?
- Traits of a great project manager
- Project management techniques
 - Waterfall
 - Agile

What is project management?

Project management is the discipline of planning, organizing, and managing resources to bring about the successful completion of specific project goals and objectives.

What is project management?

- Project management is not:
 - A black art
 - Overly complex
 - A jargon wonderland

What is project management?

Its about understanding the need of the client, the resources of the stakeholders and then creating work streams and making people responsible and accountable for deliverables.

What is project management?

- 1) understanding the need of the client
- 2) the resources of the stakeholders
- 3) creating work streams
- 4) making people responsible and accountable for deliverables.



Christopher Columbus



Christopher Columbus

- No idea where he was going
- No way to measure his success during his voyage
- Arrived somewhere without knowing
- Came back not knowing where he had been

What is project management?

- 1) understanding the need of the client
- 2) the resources of the stakeholders
- 3) creating work streams
- 4) making people responsible and accountable for deliverables.

Project management methodology

- Waterfall
- Agile
- Hybrid

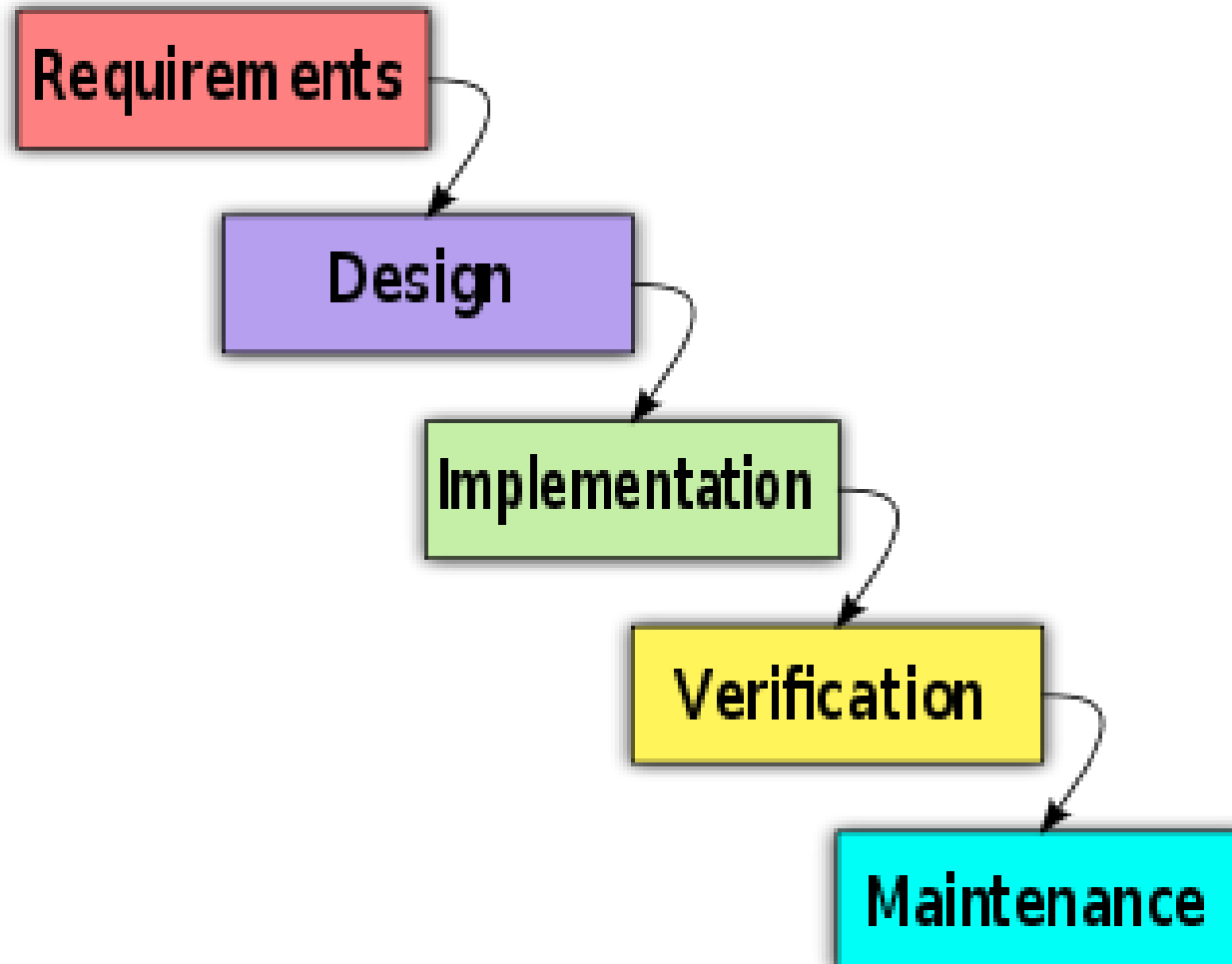
Traits of a great project manager

- The capacity to understand:
 - The clients expectations
 - The teams capabilities
 - Third party delivery schedules
- The capacity to manage:
 - Client expectations
 - The team
 - Supplier / stakeholder relationships
 - Cost

Waterfall

- Winston W. Royce
- Lockheed Software Technology
- Managing the Development of Large Software Systems, 1970
 - An example of a flawed, non working model

Waterfall



Waterfall

Supporting arguments

- Time spent in the planning stages can lead to greater savings in later stages
 - Bugs found earlier are cheaper to remove
- Huge emphasis spent on documentation
- Very simple approach, leads to discipline
- Powerful if used in 'stable' or 'shrink wrapped' projects

Waterfall

Criticism

Waterfall

Criticism

- For complex projects it is often impossible to fully complete a step before moving on
 - requirements capture might not be complete
 - Designers might not be aware of future implementation problems

Winston W. Royce :

“risky and invites failure”

Agile

- Term coined in 2001 at the Agile Manifesto

Agile

- Term coined in 2001 at the Agile Manifesto

Snowbird, Utah.



Agile

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

- **Individuals and interactions** over processes and tools
- **Working software** over comprehensive documentation
- **Customer collaboration** over contract negotiation
- **Responding to change** over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

Agile

- Individuals and interactions
- Working software
- Customer collaboration
- Responding to change

Agile

An attempt to reduce the cost of change

- Multiple short development cycles
- Changes are embraced, indeed desirable and necessary

Agile

- Extreme Programming (XP)
- Scrum
- Adaptive software Development
- Feature Driven Development
- Pragmatic programming

XP



XP

4 basic activities:

- **Coding**
 - Without code there is no project
- **Testing**
 - You cant prove anything without a test
- **Listening**
 - Coders must understand the requirements
- **Designing**
 - Design is used to lower the number of dependencies

XP

4 values + 1:

- **Communication**
 - Simple, regular, jargon free, inclusive
- **Simplicity**
 - You're not going to need it
- **Feedback**
 - Tests, customer, team
- **Courage**
 - Courage to be simple, courage to refactor
- **Respect**
 - Don't commit changes that will break a unit test

XP

29 Rules

- Planning
- Managing
- Designing
- Coding
- Testing

XP

- Collect user stories
- Create a few spike solutions
- Schedule release planning meeting
- Iteration planning meeting
- Stories and test are broken down into tasks
 - Write unit tests first
 - Don't add features unless needed in your task
- Daily stand up meetings
 - Yesterday, today and causes of delay

XP

Criticism

- Any solutions is only as good as its people
- Project creep
- Lack of structure and documentation
- Only works with senior developers
- Requires huge cultural change
- Can be inefficient