
What is a development project?

CSE 403, Spring 2003
Software Engineering

<http://www.cs.washington.edu/education/courses/403/03sp/>

Readings and References

- Readings
 - » *Rapid Development*, Steve McConnell
 - Chapter 2, Rapid-development strategy
 - Chapter 5, Risk management

What is it?

- What is a development project?
 - » *Take a risk and make an investment in order to get a positive payoff*
- Risk is an essential element
 - » if there's no risk at all, then there's no change
- Investment is an essential element
 - » "If wishes were horses, then beggars would ride"
- Positive payoff
 - » Many possible forms

LittleApp investments

- What's the investment?
 - » developer time
 - learning - new domain, new API, new tools
 - doing - requirements, development, test, delivery
 - » time of friends or acquaintances
 - defining the project
 - testing the product
 - » new whiz-bang hardware and software
 - money from parents, department, significant other, ...

LittleApp risks

- What are the risks?
 - » It was a bad idea and nothing was completed
 - Waste of time and money with nothing to show for it
 - » Loss of credibility with your friends or colleagues
 - Will they make the investment next time you ask?
 - » Opportunity cost
 - You didn't work on some other project because you worked on this one. Consequently, you didn't learn about some other domain because you learned about this domain.

LittleApp payoffs

- What are the payoffs?
 - » enjoyable project - creating things is fun!
 - » useful product for you and other users
 - » credibility with friends and colleagues
 - development credibility
 - project completion credibility
 - » increased skills and knowledge
 - » personal confidence that your ideas have value

BigApp investments

- What's the investment?
 - » **Money and time**
 - » Labor hours (expense)
 - project management and support
 - requirements definition, testing, acceptance, training
 - developer learning and doing
 - » Hardware and facilities (capital)
 - development tools, prototypes
 - space for developers and their equipment
 - » Calendar time

BigApp risks

- What are the risks?
 - » Doesn't work, works but not useful, works but value not obvious, works and valuable but not wanted
 - BIG waste of time and money
 - » Loss of credibility inside and outside the company
 - The management that authorized the project loses points
 - The management that ran the project loses points
 - The customer groups that bought the pitch are ticked off
 - Will anyone make the investment next time?
 - » Opportunity cost
 - Something else would have been a better choice and the company missed the chance to do it

BigApp payoffs

- What are the payoffs?
 - » **Money**, directly or indirectly
 - external product - sales, continued business relationship
 - internal product - improved productivity, cost avoidance, faster cycle time, ...
 - » market share
 - they buy your product, they don't buy a competing product
 - your product becomes the standard around which other development takes place - network effect

BigApp payoffs

- What are the payoffs?
 - » credibility
 - success justifies the risk that the customers took selecting the product
 - success justifies the risk that management took authorizing the expenditure to develop the product
 - success may earn the company/group/team the right to do another project, probably with higher risk and bigger investment
 - » capability
 - project management and successful delivery
 - technical knowledge

Will the project idea be approved?

- The money is there to fund *any* size project
 - » There are many more people with money than there are people with good ideas and the ability to bring difficult projects to successful completion
- The trick is to convince yourself and others:
 - » that the risk can be managed
 - » that you will deliver a large positive payoff
- Success is defined differently by all the players
 - » the project must succeed on many levels at once

Would you fund these tasks?

- A task that should be funded because "I think it would be fun to work on."
- A task that the group should be funded to do because "we've always done that kind of project"
 - » "That's my job, not yours. I've got a memo."
- A task that is "clearly better technically than the brain-dead solution proposed by those mush-for-brains marketing people who talked to the idiot managers we have around here."

Risk Management

Rapid Development, McConnell

- The goal
 - » successful project completion
- The job
 - » identify the risks
 - » address the risks with specific actions
 - » avoid or resolve the risks before they become real threats to the project
- Remember this:
 - » Mistakes are made on *every* project. The goal is to get to successful project completion *even though* mistakes were made.

4-April-2003

cse403-03-Context © 2003 University of Washington

13

Levels of risk management

- Crisis management
 - » fire fighting. Address it only after it's a problem.
- Fix on failure
 - » Detect and react. "Exception handler" style.
- Risk mitigation
 - » include slack in the plan for time lost to problems
- Prevention
 - » Execute a plan to identify risks and prevent problems
- Eliminate root causes
 - » Identify and eliminate factors that cause risks

4-April-2003

cse403-03-Context © 2003 University of Washington

14

Four Dimensions

- Projects operate along four dimensions
 - » People
 - development is a social activity, not a machine
 - » Process
 - good processes are enablers for good work
 - » Product
 - what the heck are we building, anyway
 - » Technology
 - good quality development tools appropriate to the job

4-April-2003

cse403-03-Context © 2003 University of Washington

15

Risk identification

- Avoid the classic mistakes
 - » we have good reasons for the decisions we make
 - » we are all led astray by the same bad solutions
- Implement the development basics
 - » Management fundamentals
 - » Technical fundamentals
 - » Quality assurance fundamentals
- Actively manage risks that exist

4-April-2003

cse403-03-Context © 2003 University of Washington

16

Most common schedule risks

- Feature creep
- Requirements or developer gold-plating
- Shortchanged quality assurance
- Overly optimistic schedules
- Inadequate design
- Silver-bullet syndrome
- Research-oriented development
- Weak personnel
- Contractor failure
- Friction between developers and customers