TUM

Project Management and Organization

Lecture 1: Introduction

15 April 2015

Bernd Bruegge

Technische Universität München

Institut für Informatik

Chair for Applied Software Engineering

http://wwwbruegge.in.tum.de

State of the Art: Project Management is still black art

- Complex systems are hard to develop
- Problems with technology, estimates, budgets and deadlines

- Two interesting web sites for failed projects:
- **Why Projects fail:**
  - http://calleam.com/WTPF/?tag=examples-of-failed-projects
- **List of failed and overbudget software projects:**
  - http://en.wikipedia.org/wiki/List_of_failed_and_overbudget_custom_software_projects
  - Airbus 380: http://calleam.com/WTPF/?p=4700
  - JP Morgan: http://calleam.com/WTPF/?p=5517
  - Healthcare.gov: http://calleam.com/WTPF/?p=6061

Assumptions and Requirements for this Class

- **Assumption:**
  - You want to learn more about the ***managerial*** aspects of analysis and design of complex software systems
  - You take this class because you are a bachelor in Wirtschaftsinformatik
- **Requirements:**
  - You must have taken Software Engineering I (SE I, WS 13/14) or an equivalent class (e.g. Advanced Topics in SE, WS 14/15) unless you are a bachelor in Wirtschaftsinformatik
  - **You must have your own Laptop and you are required to bring it to class!**
- **Beneficial:**
  - You understand the issues in the analysis or design of a software system
  - You have practical experience with maintaining or developing a large software system and have experienced major problems
  - You are familiar with UML (Unified Modeling Language).

Objectives of the Class

- At the end of this class…

… you appreciate project organization and management issues in software engineering

  - Manage the construction complex software systems in the context of frequent change

… you understand how to

  - Produce a high quality software system within time while dealing with complexity and change and create an estimate for the resources needed

… you appreciate that managerial knowledge is needed when developing software system.

… you are a technology manager (combining technical and managerial knowledge)

How do we achieve these Objectives?

- **Electronic E-Learning Platform Moodle**
  - Communication platform
  - Lecture slides will be made available on Moodle after each class
  - Exercise slides will be made available on Moodle before each class
- **Lectures and Exercises**:
  - In the lectures, we introduce the concepts
    - You will be doing a couple of *in-class exercises* and *quizzes*
  - In the exercises, you apply the concepts
    - In *individual exercises*, you learn how to use project management workflows and tools
    - In a *team project*, you apply your knowledge and manage the development of a mobile application

Grading

- There is one final exam
- You can obtain a bonus up to 1.4 for the exam grade by
  1) Participating in in-class exercises in the lecture
  2) Answering quizzes correctly
  3) Participating in individual exercises
  4) Creating certain deliverables in the team project

➢ For each exercise and quiz you collect points

➢ For each deliverable, your team (i.e. all members) collect points

- At the end we sum up all points and map it to a bonus

Points	**Bonus**
Less than 20%	-
Between 20% and 40%	0.3
Between 40% and 60%	0.7
Between 60% and 80%	1.0
More than 80%	1.3

Exercise #L1

- **Problem:** How many students are in this lecture hall?
- What's your guess?
- Let's first try to solve this problem without a computer
  - Suggest an algorithm to count the number of all the students in the lecture hall
- Constraint: Get the count as fast as possible.

What is the number of attendees in the lecture hall right now?

Estimate: 220, 280 and 400

Computer-based counting

- OK, now let’s try our question as an in-class exercise
- We have 400 registered students
  - Observation: Not every student brought a laptop today
- So, we have to rephrase the problem:
  - How many attendees have access to Moodle?

Login to Moodle

- Log into your Moodle account and navigate to our class forum
  - URL: www.moodle.tum.de/course/view.php?id=17991
- You should be able to access this Moodle page if you have successfully registered for both courses via TUMOnline.
  1) **German:** Projektorganisation und -management in der Softwaretechnik (IN2083)

     English: Project Organisation and Management in Software Engineering (IN2083)

  2) **German:** Zentralübung zu Projektorganisation und -management in der Softwaretechnik (IN2083)

     English: Exeercise - Project Organisation and Management in Software Engineering (IN2083)

Exercise #L2: Answer this survey in Moodle (3 min)

Technische Universität München

Moodle Startseite | Moodle News | Hilfe & Support | Angebote | Medienzentrum | Stephan Krusche

My home > My courses > SoSe 2015 > Informatik > POM (SS15) > 15 April - 21 April > Continent Survey

Navigation

- My home
- Site home
- My profile
- Current course
  - POM (SS15)
    - Participants
    - 15 April - 21 April
      - **Continent Survey**
- My courses

Continent Survey

View 0 responses

Which continent are you from?

- Australia
- Africa
- South America
- North America
- Asia
- Europe

Save my choice

Exercise #L2: Answer these additional surveys in Moodle

Operating System Survey

Which operating system do you have on your notebook that you bring to class?

- ◯ OS X Yosemite (10.10)
- ◯ OS X Mavericks (10.9) or older
- ◯ Windows 8.x (64bit)
- ◯ Windows 8, 7 or older
- ◯ Linux

Save my choice

Smartphone Survey

Which mobile operating system does your smartphone have?

- ◯ Android
- ◯ iOS
- ◯ Windows Phone

Save my choice

Development Experience Survey

How much experience do you have in developing mobile apps?

- ◯ No experience
- ◯ Less than 2 weeks
- ◯ Less than 3 months
- ◯ Less than 1 year
- ◯ More than 1 year

Save my choice

Development Environment Survey

What is your preferred development environment for a mobile app development project?

- ☐ Android, Android Studio / Eclipse and Java
- ☐ iOS, Xcode and Swift / Objective-C
- ☐ Windows Phone, Visual Studio and C#
- ☐ Other
- ☐ None

Save my choice

And the results are…

- Second Projector

Intended Audience and Lecture Times

- **Audience:**
  - Bachelor students in Wirtschaftsinformatik
  - Students in Informatics (Master, Bachelor with focus on software engineering)
  - Students who anticipate careers that require significant interaction with software engineers
- 2+2 SWS, ECTS Credits: 5
- **Lecture Time:**
  - Every Wednesday 8:15 - 9:45, HS 1
- **Exercise Time:**
  - Every Friday 8:15 - 9:45, HS 1
- You have to participate in one project.

Typical Challenges in Software Project Management

- The requirements are unknown or hard to elicit from the user
- The requirements change frequently (volatile)
- New technology that effects the outcome appears during the project
- The end user cannot be identified
- The mobile end user is not able to evaluate visionary scenarios
- We cannot look the end user "over the shoulder"
- The development is distributed.

Additional Challenges in Global Software Project Management

- Coordination Challenges
  - Reduced communication bandwidth
  - Increased latency
  - Lack of organizational cohesion
  - No informal communication
- Change Request Challenges
  - Change requests mostly through formal channels
  - Changes as ‘late surprises’
  - Rationale for change request often unavailable.

Project Management is a Problem Solving Activity

- **Analysis:** Understand the problem nature and break the problem into pieces
- **Synthesis:** Put the pieces together into a large structure that prepares for the solution
- Problem solving always happens in the context of methodologies

- **George Polya,** 1887–1985, *How to Solve It,* Princeton University Press, 1957
- Summary of the book:
  - www.math.utah.edu/~pa/math/polya.html

Once a problem is solved it is often not hard anymore.

Example: Columbus' egg

A metaphor for a brilliant idea or discovery that seems simple or easy after the fact.

Columbus's Egg Problem

- When Columbus came back from his voyage, there was a celebration dinner
- Many people attending the dinner said, that discovering America was no great accomplishment
- Columbus put an egg on the dinner table and said: "My lords, I will lay a wager with any of you that you are unable to make this egg stand on its end like I will do without any kind of help or aid."

Columbus's Egg Problem

- When Columbus came back from his voyage, there was a celebration dinner
- Many people attending the dinner said, that discovering America was no great accomplishment
- Columbus put an egg on the dinner table and said: "My lords, I will lay a wager with any of you that you are unable to make this egg stand on its end like I will do without any kind of help or aid."|
- They all tried without success
- When the egg was returned to Columbus, he tapped it gently on the table and the egg stood on its end.

Summary of Polya‘s Book

First: *understand the problem.*

Second*: devise a plan.*

Third*: carry out your plan.*

Fourth*: have to examine the solution obtained.*

Software Development and Project Management are both Problem Solving Activities

- **Methodologies:** Collection of techniques, heuristics and tools unified by a philosophical approach
- **Techniques:** Formal procedures for producing results using some well-defined notation
- **Heuristics:** Informal collection of steps
- **Tools**: Instruments or automated systems that help in accomplishing a technique or supporting heuristics

- Specific **project management** problems occur
  - When the available resources time, people and budget are limited
  - When we allow the problem to change (or the problem changes, even if we don’t allow it ☺)

Problem Solving is NOT Everything

It is possible to solve a problem, but make sure you do not apply the solution to the wrong object

Example of a Problem Solving Activity

What was the Problem?

- **User did not focus on the right problem**
- **Result: Incorrect event flow**

➢ **Constrain model instantiation**
➢ **Understand Murphy's law**
➢ **GIGO**

Higher level functionality

Drive to work with the car

2nd level functionality

Get rid off the snow

Example of a Problem Solving Activity

Object-Oriented Analysis:

- **Start with a scenario**
- **Identify the use cases**
- **Identify the participating objects (abstractions)**
- **Identify associations between the abstractions**

- **Constrain Model Instantiation**
  - **Add use case: Select car**

Use Cases:	**Closing_Door, Scraping, Cleaning_Side, Warming_Hands, Slipping, Cleaning_HeadLights**
Objects:	**Person, Driver, Nose, Car, Window, Briefcase, Remote Control**

Object-Oriented Analysis is a Paradigm

- Paradigm:
  - The set of practices that define a scientific discipline *during a particular period of time.*
- Thomas Kuhn, *1922-1996, The Structure of Scientific Revolutions,* 1962
  - History of science & problem solving
  - Kuhn popularized the terms paradigm & paradigm shift

Project Management: Traditional Definition

Project Management is a collection of techniques, methodologies, tools and heuristics that support the development of

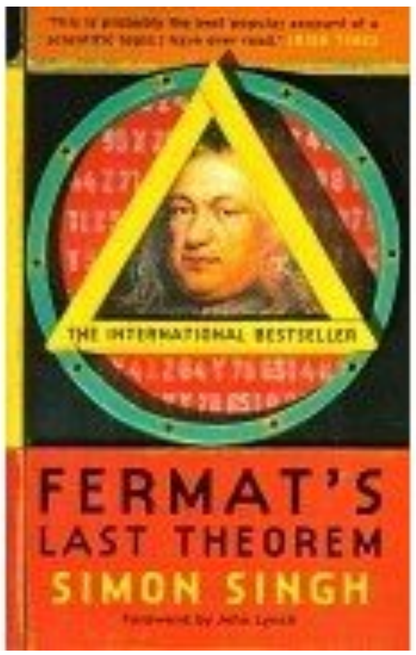
a *high quality software* system

with a given *budget*

before a given *deadline*.

Some Problems Don't have a Deadline

- Fermat's Last Theorem (1621):
  - If $n > 2$, $a^n + b^n = c^n$ has no solutions with non-zero integers a, b, and c.
- Theorem unsolved for 300 years
- Many prizes offered to solve the theorem, most famous the 100.000 Mark prize offered in 1908 by Paul Wolfskehl
- Andrew Wiles, Professor, Princeton University
  - His "project" from 1984-1993: Trying to solve the theorem without telling anybody
  - 1993: Publishes his proof, unfortunately it is wrong
  - 1993-1994: Repair of the proof (with his former student Richard Taylor)
  - 1997: Wiles receives the Wolfskehl Prize



Project Management: Definition used in POM

Project Management is a collection of techniques, methodologies, tools and heuristics that support the development of

a *high quality software* system

with a given *budget*

before a given *deadline*

while ***change occurs***

Example of Change: Running a Rapid

A quiet river

Unplanned event: A hydraulic behind the rock

What is a hydraulic?

What can we do?

Actions to deal with Hydraulics

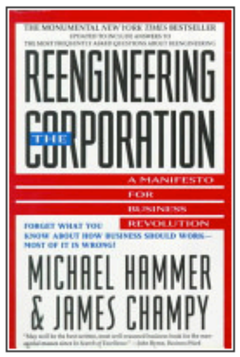
- **Short term actions**
  - Rafter:
    - Use your skills to circumvent the hydraulic. If this not possible, give up and go back
    - Avoid unknown terrain, stick to well known terrain
  - Management:
    - Require the rafters to follow the plan, no matter what the consequences (Defined process)
    - Let the rafters decide what to do, even if the original plan was different (Agile process)
- **Long term actions**
  - Skill improvement: Learn how to deal with a hydraulic
  - Identify Patterns, Heuristics and Lessons learned
  - Process improvement: Explore the route around the corner (rafting: scouting, software development: develop a prototype before the product)
  - Train your team how to react to change.

What is Change?

- Change is when something becomes clear (“it crystalizes”)
- Change is when something new appears (“technology enabler”)
- Change is when something becomes important (“change of requirements”)
- Change can happen fast, changes are often unexpected
  - Manager must anticipate and react to unusual technology
  - One strategy: Look at universities, R&D labs, what is done there will be out on the market soon.

Hammer and Champy (*Reengineering the Corporation, 1993*): “Change is the only thing that is constant”.

Reengineering the corporation: a manifesto for business revolution



Michael Hammer, James Champy

HarperBusiness, 1994 - Business & Economics - 233 pages

★★★½☆

16 Reviews

"Reengineering the Corporation" sets aside much of the received wisdom of the last 200 years of industrial management and in its place presents a new set of organizing principles by which managers can rebuild their businesses.
The book provides numerous examples and in-depth case studies of how leading organizations are achieving significant competitive gains through reengineering: How Ford Motor reduced the size of its North American accounts payable organization by 80% while improving the process; how IBM is
More »

Examples of Change in Software Projects

- **Several jobs are combined**
  - Example: Case worker, a person who deals with the complete workflow of helping another person
- **Processes have multiple versions to support multiple products**
  - Example: Product line requirements engineering
- **Checks and controls are reduced**
  - Example: Developers have the power to make decisions
- **Work is performed where it makes sense**
  - **Outsourcing**: A company subcontracts an activity such as product design or manufacturing to a third-party company
  - **Offshoring**: A company transfers an organizational function to another country (regardless of whether the work is outsourced or stays within the same company).

Methodology Issues

- Methodology: Provides guidance, general principles and strategies for selecting methods and tools in a given project environment in the context of change
- The key questions for which methodologies typically provide guidance include:

Customer: How much should the customer be involved? Should the team interact with the customer?

Planning: How much planning should be done in advance?

SE I

- Reuse: How much of the design should result from reusing past solutions?
- Modeling: How much of the system should be modeled before it is coded?

Process: In how much detail should the process be defined?

Project Monitoring: How often should the work be controlled and monitored? When should the project goals be redefined?

What do you learn in this class?

- Project Organization and Communication
- The difference between process and product
- How to set up a project plan, how to control its execution
- How to deal with change and uncertainty
- Configuration Management
- Release Management
- Rationale Management.

Management vs. Project Management vs. Software Project Management

- **Management**: Getting a specific task done through people
  - usually defined in terms of functions: Planning, organizing, directing, controlling and communicating are typical management functions
- **Project management** is defined in the context of a project
  - tries to accomplish a specific task within a time frame and limited resources (given deadline and given budget)
- **Software project management** is defined in the context of software development
  - Activities to develop a software system within a given time frame and with limited resources while change occurs: Planning, Requirements Elicitation, System Design, Reviews, Release Planning, …

TUM

Project Management and Organization

Lecture 2: Communication

15 April 2015

Bernd Bruegge

Technische Universität München

Institut für Informatik

Chair for Applied Software Engineering

http://wwwbruegge.in.tum.de

Icebreaker

POM
15 April 2015

Icebreaker - Definition

ice•break•er |ˈīsˌbrākər|

noun

a ship designed for breaking a channel through ice.

• a thing that serves to relieve inhibitions or tension between people, or start a conversation.

• a thing that breaks up moving ice so as to lessen its impact, esp. a structure protecting the upstream end of a bridge pier.

Problem Statement

- Functional Requirements:
  - Write your full name on a balloon
  - Inflate the balloon
  - Deposit the balloon at the front of the lecture hall

- Nonfunctional Requirements
  - Don’t blow up the balloon while inflating it
  - Your handwriting must be readable
  - Don’t swallow the balloon.

Project Management Plan

Human Resources: 220-257 participants

Additional Resources: 29 markers, 500 balloons

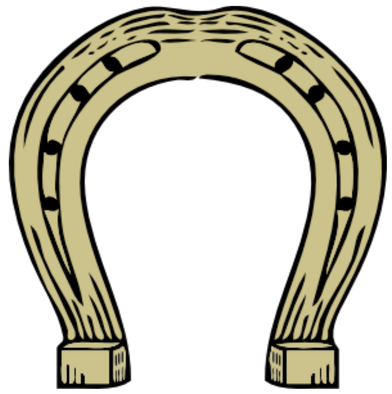
Budget: 10 €

Duration: 15 min

Go for it!

Change!

- Additional functional requirements:
  - Find your balloon
  - Define an order for all the balloons
  - Visualize the order by forming a horseshoe



42

Post Development

- Destroy your balloon: Step on your balloon
- Return your resources: 29 markers
- Return your balloon or put it in the trash

Summary

- Objectives, Assumptions and Requirements of the Class
- Management vs. Project Management vs. Software Project Management
- Revised Definition of Project Management
- Moodle is our communication platform for discussion and in-class exercises.
- We are using commercial tools for communication, issue tracking, release management in the exercises.

Readings

- There is no single textbook
- For each lecture we are using several references
- These references are mentioned at the end of each lecture
- Today:
  - George Polya, How to Solve It, Princeton University Press, 1957
  - Thomas Kuhn, The Structure of Scientific Revolutions, 1962
  - Hammer and Champy, Reengineering the Corporation, 1993, Harper Business