
Table of Contents

Introduction	1.1
--------------	-----

1. Lectures

01 Project Organisation	2.1
02 Software Lifecycle Models I	2.2
03 Scrum	2.3
04 Software Lifecycle Models II	2.4
05 Usability Management I	2.5
06 Usability Management II	2.6
07 Software Configuration Management I	2.7
08 Developing Winning Proposals	2.8
09 Kanban	2.9
10 Software Configuration Management II	2.10
11 Contracting	2.11
12 Estimation and Scheduling	2.12
13 Build Management	2.13
14 Global Project Management	2.14
15 Agile Architecture	2.15
17 Release Management	2.16
18 Risk and Demo Management	2.17

2. Exercises

3. Questions

Quiz Questions	4.1
----------------	-----

POM Lecture Summary

Course Summary, Summer Term 2017

tl;dr: This purpose of this document to collaboratively create a both concise and detailed course summary of the *POM* Lecture from 2017 Summer Term at TUM.

To learn as effective as possible, I would like to encourage everyone to engage in the discussion evolving around the content of this document. If you have questions or challenge what someone else wrote please do so in a **constructive way**. We are all new to the subject of Protein Prediction and mistakes happen. Let's learn from them together!

Official Lecture Resources

Lecture Homepage: [Web](#)

Lecture Resources: [Moodle](#)

Getting Started

This document is set up a **Gitbook** and hosted on **Github**. When you read this, you were already granted access to the repository so the first step is done.

The easiest way to start contributing is to download **Gitbook Editor** (available for Mac, Linux, Windows) from [here](#).

Before you add / change anything, please read through the Contribution Guide.

Contribution Guide

Tell others what you work on | Write meaningful commit messages | Push often | Use American English

Why is there a contribution guide? I think it is in everyone's best interest to keep this summary as easy to understand as possible for everyone. This guideline should help to maintain consistency across the entire document.

Each section may contain a short additional information on how to format things specific to that section. Please have a look there as well.

1. Adding new content

1.1 Adding minor updates

If you add minor updates, like the answer to a single question, you can do this on the `develop` branch directly. Make sure your commit has a meaningful message.

1.2 Adding major updates

If you add major updates, like several related changes (e.g. an entire lecture summary), go along as follows:

1. Add a new **issue** on Github, describing what you are working on
2. Create a `feature/<issue-name>` branch and add your changes
3. Open a pull-request to merge back into `develop` and add the other contributors as reviewers
4. Once the pull request is merged, delete your feature branch and close the issue by referencing the merge commit

Why so complicated? This way the issues reflect new changes and are transparent for all contributors.

2. Challenging existing content

If you find obvious mistakes (typos, clearly wrong statements) just change them directly.

If you are challenging statements, answers to questions etc. which might not be trivial to understand go along as follows:

1. Open a new **issue** on github.
2. Reference the the statement in question you consider to be wrong
3. Provide an explanation why you think it is wrong
4. Provide your correct solution.

3. Adding new contributors

The purpose of this document is to foster collaborative learning - hence to make this as inclusive as possible. This being said, too many collaborators would probably lead to chaos . If you know other students personally, you want to add to the project shoot me a message and we will figure it out.

01 Project Organisation

3 May - 9 May

Slides: [Introduction](#)

Slides: [Team Building](#)

Slides: [Project Organization](#)

02 Software Lifecycle Models I

10 May - 16 May

Slides: [Moodle](#)

03 Scrum

17 May - 23 May

Slides: [Moodle](#)

04 Software Lifecycle Models II

17 May - 23 May

Slides: [Moodle](#)

05 Usability Management I

31 May - 6 June

Slides: [Moodle](#)

06 Usability Management II

31 May - 6 June

Slides: [Moodle](#)

007 Software Configuration Management I

7 June - 13 June

Slides: [Moodle](#)

08 Developing Winning Proposals

7 June - 13 June

Slides: [Moodle](#)

09 Kanban

14 June - 20 June

Slides: [Moodle](#)

10 Software Configuration Management II

21 June - 27 June

Slides: [Moodle](#)

11 Contracting

21 June - 27 June

Slides: [Moodle](#)

12 Estimation and Scheduling

28 June - 4 July

Slides: [Moodle](#)

13 Build Management

5 July - 11 July

Slides: [Moodle](#)

14 Global Project Management

5 July - 11 July

Slides: [Moodle](#)

15 Agile Architecture

12 July - 18 July

Slides: [Moodle](#)

17 Release Management

12 July - 18 July

Slides: [Moodle](#)

18 Risk and Demo Management

19 July - 25 July

Slides: [Moodle](#)

POM - Question Catalogue

TUM #POM #tbd

1. Quiz Questions

Quiz 1: Introduction

Question: How did the Columbus egg problem end?

- a. Columbus destroyed the egg
- b. The egg stood on its end
- c. The problem is not yet solved
- d. The other people found a solution to the problem

Question: Which goals of a software project define its success in modern project management?

- a. Extend the deadline
- b. Develop High Quality
- c. Adapt to changes
- d. Exceed the budget

Quiz 2: Team Building

Question: The roles Scrum Master, Product Owner and Developer describe what kind of project?

- a. Agile Project
- b. Heavyweight Project

Question: Which of the following stages are used to describe team development?

- a. Hunting
- b. Storming
- c. Fighting
- d. Forming

Question: Let's recap the differences between groups and teams.

- a. A [Team] has a common goal.
- b. Participants of a [Group] are loosely connected.
- c. Each [Team] is a [Group], but not each [Group] is also a [Team].
- d. [Team] members need to work together.

Quiz 3: Project Organization

Question: What is the defining characteristic of a project function in comparison to other activities?

- a. Project functions contain other low-level activities
- b. A project function spans the entire duration of the project

- c. Project functions are activities that require an experienced developer
- d. A project function culminates in a milestone

Question: What are main responsibilities of the project manager?

- a. Maintaining accountability
- b. Analyze the application domain
- c. Design and implement tests
- d. Administering the resources

Question: Which of the following is NOT a unit of work?

- a. A work package
- b. A deliverable
- c. A task
- d. An activity

Question: What are typical challenges in a matrix organization?

- a. Conflicting interests between department and project
- b. Projects cannot be tailored around the problem
- c. Competing demands for team members in multiple projects
- d. Teams cannot be assembled rapidly

Question: What are typical organization forms?

- a. Delegation Organization
- b. Matrix Organization
- c. Line Organization
- d. Project-based Organization

Question: What are characteristics of information flow in non-hierarchical project organizations?

- a. It is easier to manage because of the simplified communication structure
- b. It increases bureaucracy due to the complicated decision flow
- c. It expects decisions to be made at each level
- d. It reduces development time because the communication is more efficient

Question: What are typical roles in meeting management?

- a. Primary Facilitator
- b. Scrum Master
- c. Minute Taker
- d. Product

Question: What are characteristics of European and Polynesian Navigation?

- a. Agile Planning as in the Polynesian Navigation uses context dependent actions
- b. Traditional Planning as in European Navigation corrects the course to follow the initial plan when a deviation is observed
- c. Traditional Planning as in the European Navigation can easily react to changes

Question: Which of the following is an example for an asynchronous communication mechanism?

- a. Stand-up Meeting
- b. Slack
- c. Informal Meeting
- d. Confluence

Quiz 4: Software Lifecycle Models 1

Question: Which managerial challenges come with modeling?

- a. Complexity
- b. Improved communication
- c. Redundancy
- d. The Analysis Paralysis Antipattern

Question: What are possible changes when tailoring a software lifecycle model to specific needs?

- a. Leave out the activity "Maintenance"
- b. Leave out all activities
- c. "Test" before "Implementing" the system
- d. Rename "Detailed Design" to "Object Design"

Question: Which of the following statements about modeling are correct?

- a. Conceptual models represent a person's perception of an idea
- b. Specification models communicate the idea to a tool
- c. A mockup drawn on a napkin is a specification model
- d. The best way to support communication is formal modeling

Question: Which of the following statements about software lifecycle models are correct?

- a. XP, Kanban, Scrum and Unified Process are agile models Incorrect
- b. Extrem Programming is based on feedback loops
- c. V-Model and V-Model XT are both sequential models
- d. The Waterfall Model is the current state-of-the-art practice

Question: Which of the following software lifecycle models are iterative models?

- a. Spiral Model
- b. V-Model
- c. Waterfall
- d. Unified Process

Question: Which process steps are part of the Waterfall Model?

- a. Design Correct
- b. System Testing
- c. Installation
- d. Unit Testing

Quiz 5: Scrum

Question: Which statements about Scrum teams and roles are correct?

- a. The development team works on items in the Sprint Backlog
- b. The Product Owner is responsible for the process
- c. The Scrum Master prioritizes the backlog items
- d. The development team is self-organizing

Question: Which statements about Scrum teams and roles are correct?

- a. As a student I want to access my course schedule so that I am on time in class
- b. As a kid I want to have a slide so that I can use it.
- c. As a cook I want a system that is connected to my fridge.

d. I want a possibility to determine the scale of discount of a customer.

Question: Which of the following are Scrum artifacts?

- a. Potentially Shippable Product Increment
- b. Daily Scrum Meeting
- c. Product Kickoff Meeting
- d. Product Backlog

Quiz 6: Software Lifecycle Models 2

Question: What is correct about iterative and incremental development?

- a. Incremental development only changes existing functionality
- b. Refactorings can cause product iterations
- c. Iterative development aims to improve existing system parts

Question: What is correct about prototypes?

- a. Prototypes are used in the Waterfall model
- b. Vertical prototypes only cover one layer of the system
- c. Rapid prototyping is a misnomer
- d. Prototypes are used in the Spiral model

Question: What is correct about verification and validation?

- a. Both techniques are internal processes
- b. Verification evaluates whether you are building the system in the right way
- c. Both techniques are used in the V-Model
- d. Validation assures that you are building the right system

Question: Which of the following workflows are supporting workflows in the Unified Process?

- a. Configuration and Change Management
- b. Analysis and Design
- c. Deployment
- d. Project Management

Question: Which of the following statements about agile software lifecycle models are correct?

- a. It is difficult to introduce agile methods in a company that is only used to sequential lifecycle models
- b. Agile methods are not recommended for small projects with frequent changes
- c. In Test Driven Development (TDD), developer first implement a functionality, then refactor it, and finally test it
- d. The deliverable in Extreme Programming (XP) is comparable to a Product Increment in Scrum

Question: Which practices are used in Extreme Programming (XP)?

- a. Write production code with two people
- b. Improve understandability of source code while leaving its behavior unchanged
- c. Divide a project into the engineering and production stage
- d. Write a test before implementing a feature, then refactor

Quiz 7: Usability Management 1

Question: Which terms are correlated to Usability Management?

- a. Error handling
- b. Portability
- c. Memorability
- d. Learnability

Question: Which statements about user interfaces are correct?

- a. The effort for the development of good user interfaces is high
- b. The more controls a user interface includes, the better is its learnability
- c. User interface development requires interaction with users
- d. All error messages in the user interface should be intended for the developers

Question: Which term does the following sentence describe? "A measurement of how easily a user can use an application"

- a. Usability
- b. Accessibility
- c. Functionality

Question: What are characteristics of a good heuristic evaluation?

- a. All problems are justified properly
- b. In general, there is no need to consider any other heuristics other than Nielsen's 10
- c. The interface is evaluated multiple times

Question: Which heuristics need to be considered regarding consistency and standards?

- a. Similar buttons should do similar actions
- b. All options must always be visible to the user
- c. Actions should be performed in a similar way as the target platform's actions

Question: Which heuristics have to be taken into consideration when reporting errors to users?

- a. Show a detailed report with all technical details of what went wrong
- b. Offer constructive help with the error and explain why it occurred
- c. Be as precise as possible and include the user input
- d. Blame the user

Quiz 8: Usability Management 2

Question: Which statements about human capabilities in UI development are correct?

- a. Long term human memory has a large capacity and little decay
- b. As a consequence of Fitt's law, it is recommended to use hierarchical menus in UI design
- c. Fitts's law describes how fast a user can move his hand to a target of a certain size at a certain distance

Question: What are challenges in prototyping?

- a. Prototyping does not allow iterations
- b. A prototype with many problems can lead to negative feelings of the software
- c. Prototyping always increases the costs Incorrect
- d. Customers might have wrong expectations after they have seen a prototype

Quiz 9: Software Configuration Management 1

Question: Promotion vs. Release vs. Revision. What is correct?

- a. A potentially shippable product increment is a promotion
- b. A release is a version made available to users or clients
- c. Revisions change major functionality of the developed system
- d. Promotions are distributed only team internally

Question: What is correct about the object model for configuration management?

- a. Controlled items are modeled with the composite pattern
- b. The master directory contains all the promotions
- c. A configuration item can have multiple versions
- d. A configuration item can be a version

Question: Which of the following are Software Configuration Management activities?

- a. Promotion Management
- b. Risk Management
- c. Branch management
- d. Configuration item identification

Question: What are bad practices in distributed version control?

- a. Keep the working copy up to date
- b. Commit changes for two different bugs at once
- c. Promote changes often to avoid merge conflicts
- d. Change promoted versions in the remote repository

Question: What is correct about Git commands?

- a. To promote changes you use **git commit**
- b. **git pull** is a compound command composed of fetch & merge
- c. **git clone** creates a local working copy of an existing git repository
- d. **git add** adds file changes in your existing directory to your index (staging area)

Question: What are bad practices in distributed version control?

- a. Administrators have total control over who can do what
- b. The master directory can be restored from a programmer's directory if the server crashes
- c. Branches are lightweight
- d. It has a low learning curve

Question: What are characteristics of change requests?

- a. Each change request includes a prototypical implementation of the solution to support the assessment of the request
- b. Change requests are only approved if they are easy to implement
- c. Each change request triggers a change request process

Question: What is NOT a typical activity in a change control process?

- a. Create a new baseline
- b. Evaluate the change request
- c. Ask a developer for permission
- d. Implement the change

Quiz 10: Developing Winning Proposals

Question: What is true about the content and effort of proposal types?

- a. Co-sourcing may include resource options in a controlled manner
- b. Outsourcing may include resource options in a controlled manner
- c. Venture has defined deliverables and outcomes
- d. Consulting has an indefinite duration

Question: What is correct about proposals?

- a. The proposal is part of the final contract
- b. In real life, IT proposals are mostly limited to design and implementation projects
- c. A proposal is about a specific problem, but typically does not include a description of the solution
- d. A proposal does not include legal statements

Question: Which are key attributes of a proposal?

- a. Flexible
- b. Tailored
- c. Formal

Question: What makes a winning proposal?

- a. Focus the proposal on the procurement department
- b. Offer a good price
- c. Gain the trust of the client
- d. Promise the client to deliver the best possible solution

Question: The following text is a part of an already existing proposal:

"...At the end of the project we are providing the best software running on a new, faster and fully functional server. In addition, all of the existing data, in an optimal way, have been moved from the old server to the new. All of the current functionality still exist. Access to the system...."

Is this proposal acceptable?

- a. Yes
- b. No

Question: What are good reasons to decline bidding for a proposal?

- a. Previous similar proposals made to this client were all rejected
- b. You currently do not have the required skills or capacity
- c. You already have a lot of business with this client
- d. A lot of interaction with the client is required to create the proposal

Quiz 11: Kanban

Question: A Kanban system is an example of a ...

- a. Drum-Buffer-Rope system
- b. Pull system
- c. Push system

Question: Which statements about "Kaizen" and waste are correct?

- a. Consistent and need-based capacity utilization is the goal when reducing waste from inconsistency and inhomogeneity.
- b. Reducing wait time is an example for improvement through reduction of activities that don't add value.

c. Kaizen aims to revolutionize the old process by replacing it with a new one.

Question: What should the Kanban board reflect?

- a. What the team actually does
- b. The process of the team as it is specified in official project documents
- c. It depends
- d. The desirable process of the team

Question: What does Kanban focus on when introduced to teams with an existing development process?

- a. Changing roles and responsibilities
- b. Evolutionary change
- c. Leadership
- d. None of the mentioned

Question: In a Kanban project, the development team is blocked, because the deployment team has hit the deploy WIP limit. If they would start to work on an additional task, they would break the development WIP limit. What should the development team do according to Kanban?

- a. Increase the WIP limit of the development team
- b. Increase the WIP limit of the deployment team
- c. Help the deployment team to solve the problem
- d. Break the development WIP limit and start another task

Question: How does a WIP reduction affect the lead time?

- a. Lengthens lead time
- b. Shortens lead time
- c. Has no effect

Quiz 12: Software Configuration Management 1

Question: Which statements about pull requests are correct?

- a. When the quality of the changes in a pull request is not ok, the reviewer must implement the requested improvements
- b. In the code review workflow, the reviewer approves the pull request and then merges the changes of the feature branch to the development branch.
- c. When a developer creates a pull request, and then commits further changes to the source branch of the pull request, the pull request is automatically updated

Question: Which statements about pull requests are correct?

- a. Poor code in the development branch instills a sense of abandonment and spreads out to the rest of the source code
- b. It recommends that developers should use test driven development
- c. It describes how to avoid the phenomenon of software entropy
- d. Poor but working code should stay unrepaired, otherwise you loose too much time

Question: What are best practices for branch and merge management?

- a. A good practice to avoid merge conflicts is to pull often from the development branch into the feature branch.

- b. Avoid having feature branches that span through a long period of time.
- c. Make use of a specific branching model.
- d. All requirements should be mapped to one feature branch, regardless of their size.

Quiz 13: Contracting

Question: Select all acceptance-relevant aspects to be considered in contracts.

- a. Validation of the project results
- b. Verification of the project results
- c. Check for completeness

Question: For contracting parties, it is important to know ...

- a. Once the contract is concluded, contractors have to implement quality management procedures to meet all project goals defined in the iteration plan
- b. Once the contract is concluded, clients just receive deliverables and confirm timely shipment.
- c. There are always two roles (client and contractor) that have individual goals and approaches.

Question: Contracts, in general, define ...

- a. The responsibilities of the respective contracting parties
- b. The basic project parameters in terms of deliverables, schedule, and budget.
- c. How quality requirements regarding real-time are implemented.

Question: Select an interaction strategy that should guide contracting procedures as well as project operation in general to establish a fair set up.

- a. Competitive (my win is your loss)
- b. Indifferent (I don't care - you lose)
- c. Cooperative (win-win)

Question: For the different contract types, select the statements that apply:

- a. Fixed-price contracts do not allow for defining any incentives
- b. Fixed-price contracts are the option preferred by clients to minimize their project risk
- c. Using a time & material contract, clients face the risk to never get a working system

Question: Regarding the practices to embody contracts, the following statements apply:

- a. Money for nothing is an incentive model awarding good performance
- b. Maximum price is a means to better control the project budget by setting limits
- c. The agile fixed price combines many proven practices thus fully replacing the classic fixed price

Quiz 14: Estimation and Scheduling

Question: What should be considered when estimating?

- a. Complexity of the project
- b. Project duration
- c. Project budget
- d. Infrastructure

Question: What is correct about methods for estimation?

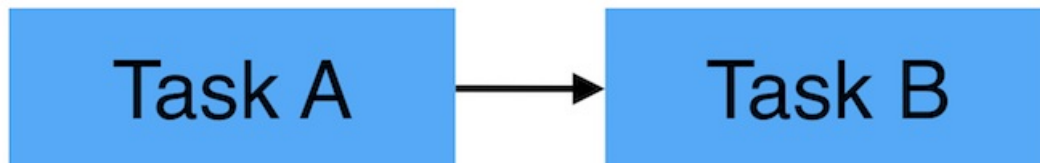
- a. COCOMO is an expert estimation technique with a top-down approach
- b. Planning poker is a team expert estimation technique
- c. Expert estimations are based on formulas

d. The lines of code estimation is considered as malpractice

Question: What are challenges of planning poker?

- a. Different teams estimate differently
- b. It helps to discover important project issues
- c. Estimates can vary based on the abilities of the people who are estimating

Question: Which type of dependency is shown in the following picture?



- a. Finish-to-Finish
- b. Start-to-Finish
- c. Start-to-Start
- d. Finish-to-Start

Question: What are good strategies to shorten the project duration?

- a. Consider to use existing components
- b. Find activities that can be parallelized on the critical path
- c. Add manpower to the project

Question: What does a good project manager do?

- a. Establish good and clear communication only with the best employees
- b. Stimulate independent thinking
- c. Avoid assumptions

Quiz 15: Build Management

Question: What is correct about continuous integration?

- a. The build process usually includes multiple steps
- b. High complexity of software systems makes it harder to integrate the software components
- c. The later integration occurs, the less likely it is that unexpected failures occur
- d. The continuous integration server notifies developer about the build status

Question: What are typical fault handling techniques?

- a. Fault detection: testing
- b. Fault ignorance: ignore all faults
- c. Fault tolerance: software configuration management
- d. Fault avoidance: software configuration management

Question: Choose the correct answers about testing.

- a. Integration testing tests the interfaces between subsystems

- b. System testing confirms that classes and methods are correct
- c. During a beta test, software is tested as realistic as possible in the target environment
- d. Acceptance tests are performed by the client, not the developer

Quiz 16: Global Project Management

Question: Estimate how many large IT-projects (those with initial price tags exceeding \$15 million) go so badly that they can threaten the very existence of the company?

- No correct answer given. I think it was 6%. **Question:** Communication is crucial. How can it be leveraged to reduce challenges specific for global projects? a. Hold isolated team meetings only b. Usage of appropriate communication tools c. Plan for both asynchronous and synchronous communication d. Enforce formal and trackable communication **Question:** Which of the following mitigations solve problems in global setups? a. Improvement epics in the backlog b. Additional time for innovation between sprints c. The introduction of collaboration forums to improve the autonomy of teams in later phases of the product life cycle d. Teams should present their delivery for other teams to prevent isolated teams **Question:** What are key success factors in global projects? a. Time zone utilization b. Non-negotiable requirements c. Static teams d. Meeting notes and documentation are easy accessible

Quiz 17: Agile Architecture

Question: How are estimations done based on story mapping? a. The first estimate of the potential development team will be tracked against reality and every deviation will be managed individually b. After the first few sprints, the backlog is re-estimated and the assumed velocity is checked against the real velocity c. The potential development team quickly estimates all known epics/tasks relatively, and then has a closer look at 2-5 user stories d. The most senior developer is doing the first estimate and discusses everything with the project manager

Question: What is correct about user story mapping? a. Tasks on the right side have to be done before tasks on the left side b. Tasks on top are more important than tasks on the bottom c. To define tasks for releases, the tasks are split horizontally into groups d. We do not focus on users, but only on tasks and activities they perform **Question:** What are typical guidelines about architectural decisions? a. In an agile project, software architecture should not be documented, because changes occur too often b. Political and organizational constraints influence the software architecture c. Decisions should be delayed until the last responsible moment (and that might be the start of the project/product development) d. Software architecture must always be done in detail upfront **Question:** What can be said in general about software architecture? a. Arc42 is a standard for architecture documentation that focuses on different views on the system b. A good architecture is something that only software architects and software developers understand c. Technical layering is more important than domain layering in the long run d. Software architecture is something that evolves over time and has to be managed carefully **Question:** Why are quality attributes so important? a. Quality attributes are usually distributed over the system, so they have to be taken into account on most changes of the system b. Quality attributes, also known as quality goals or non-functional requirements, often conflict with each

other and therefore have to be balanced with knowledge of the domain of the system c. All quality goals are equally important and therefore are the same for all systems d. Quality attributes define the detailed functional aspects of a system and therefore define the business logic

Quiz 18: Release Management

Question: What are best practices for release management?

- a. The release process needs to be reliable and repeatable
- b. Release externally first and then promote the release to internal test users
- c. Automate every step of the build process
- d. Release early and often to obtain user feedback

Question: Select the correct terminology

- a. Continuous integration: Technique where members of a team integrate their work frequently.
- b. Continuous delivery: Approach in which teams keep producing valuable software in short cycles and ensure that the software can be reliably released at any time.
- c. Continuous deployment: Every change that passes automated tests is automatically deployed.
- d. Continuous Software Engineering: Organizational capability to develop, release and learn from software in short cycles.

Question: What is meant by DevOps?

- a. An approach that can only be used by small companies
- b. An acronym for Development + Operations
- c. An emphasis on development, quality and operation departments to work closely together to release software faster and more reliable
- d. A new software process model