BLEKINGE INSTITUTE OF TECHNOLOGY

BTH

Software Metrics (PA1407)

Lecture 7

Software estimation

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Software Project Planning

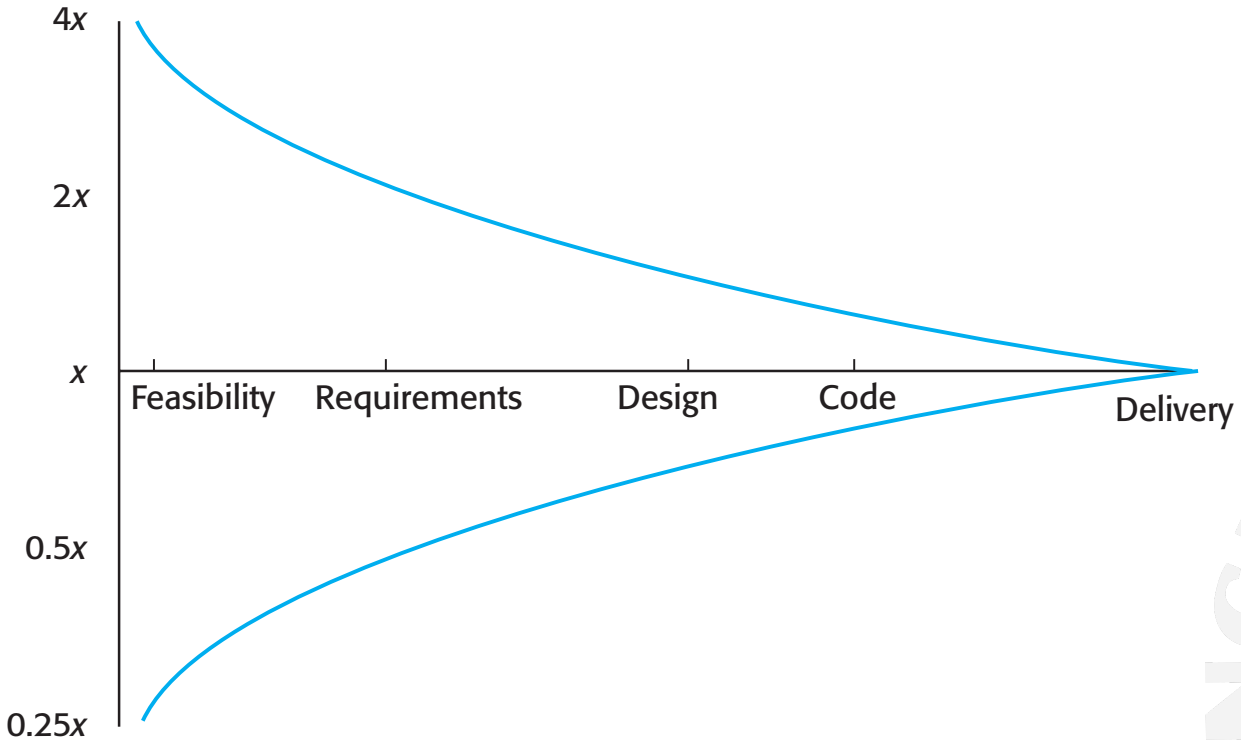
- Software project planning encompasses many activities
  - Estimation, scheduling, risk analysis, quality management planning, and change management planning
- Estimation determines how much money, effort, resources, and time it will take to build a specific system or product
- The software team first estimates
  - The work to be done
  - The resources required
  - The time that will elapse from start to finish
- Then they establish a project schedule that
  - Defines tasks and milestones
  - Identifies who is responsible for conducting each task
  - Specifies the inter-task dependencies

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Software Estimation

- Planning requires managers and the software team to make an initial commitment
- Process and project metrics can provide a historical perspective and valuable input for generation of quantitative estimates.
- Estimation carries inherent risk, and this risk leads to uncertainty.
- The availability of historical information has a strong influence on estimation risk
  - Estimates can be made with greater assurance
  - Overall risk is reduced
- Nevertheless, a project manager should not become obsessive about estimation
  - Plans should be iterative and allow adjustments as time passes and more is made certain

The Cone of Uncertainty



BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Estimation

- Three major categories of software engineering resources
  - People
    - It is the most significant contributor in total project cost. This is why main focus is on effort estimation.
  - Other
    - Development environment, reusable software components etc.
- Estimation approaches
  - Top down Vs Bottom up
- Effort and/or cost estimation techniques/models can be grouped in different categories:
  - Algorithmic
  - Expert based techniques
  - Artificial Intelligence (AI) based techniques

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Algorithmic Techniques/Models

- Cost/Effort is estimated as a mathematical function of product, project and process attributes whose values are estimated by project managers
  - Effort = A x Size^{B} x Cost Drivers
    - A is an organisation-dependent constant, B reflects the disproportionate effort for large projects
- The most commonly used product attribute for cost estimation is code size.
- Most models are similar but they use different values for A, B and M.

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Algorithmic Techniques/Models

- The size of a software system can only be known accurately when it is finished.
- Several factors influence the final size.
  - Reused components, language etc.
- Size estimates become more accurate as the development moves on towards finished product.
- The estimates for other factors (e.g. cost drivers) in the equation are very subjective.
- COCOMO is one example of algorithmic cost estimation

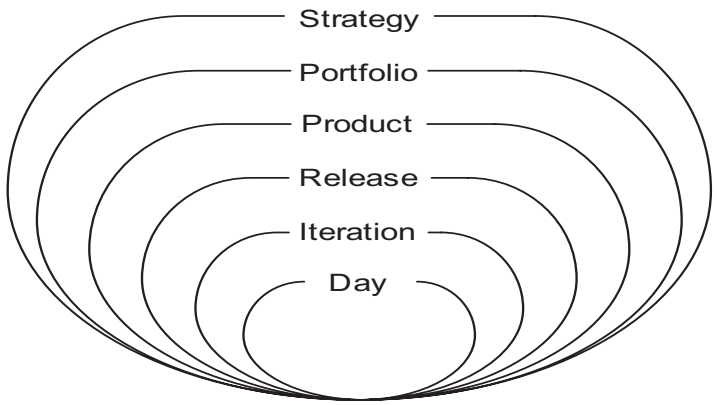
BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

AI Based Techniques

- Case-Based Reasoning (CBR)
  - It assumes that similar problems have similar solution.
  - It provides estimates by comparing the characteristics of the current project with similar past projects.
  - Companies need to build a case base that stores the data of the completed projects.
- Classification and Regression Trees (CART)
  - CART uses independent variables (predictors) to build binary variables
  - Example regression tree of a software project

Agile Planning and Estimation

- In Agile Software Development (ASD) the actual planning is valued more than the resulting documented plans.
- Agile teams plan at release, iteration and day levels
  - Product backlog, release backlog, sprint/iteration backlog
- Agile estimation is also performed at these different levels to support planning



The planning levels in ASD

Planning Poker

- *The best way I've found for agile teams to estimate is by playing planning poker (Grenning 2002)*

- This method tries to make the meetings more short and productive, by making them more fun and dynamic.

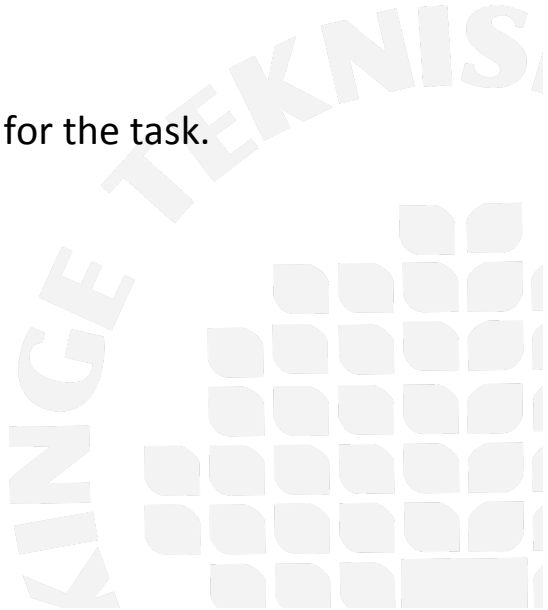
- It is a group estimation technique that allows agile teams to estimate effort in semi-structured sessions during sprint and/or release planning.

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Planning Poker Steps

1. Each estimator is given a deck of cards, each card has a valid estimate written on it.
2. Customer/Product owner reads a story from product backlog and it's discussed briefly.
3. Decompose selected story into tasks.
4. Estimate each task.
   a. Each team member selects a card PRIVATELY that's his or her estimate for the task.
   b. Cards are turned over so all can see them.
   c. Discuss differences (especially outliers).
   d. Re-estimate until estimates converge.
   e. Move on to the next task.

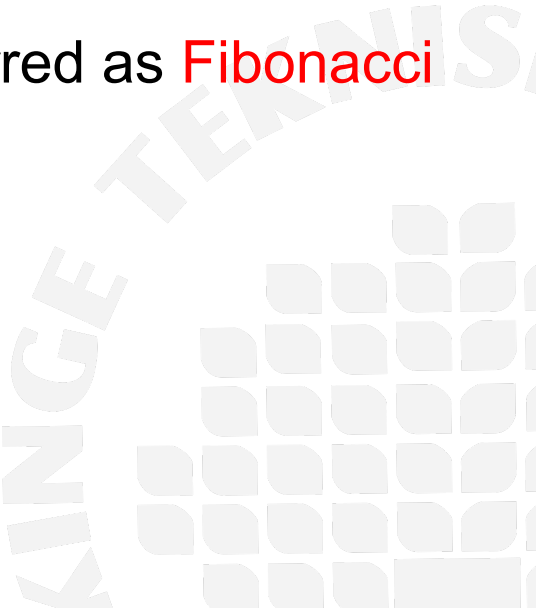
- *Repeat 2-4 if there are more stories in the current iteration*



BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Planning Poker Scales

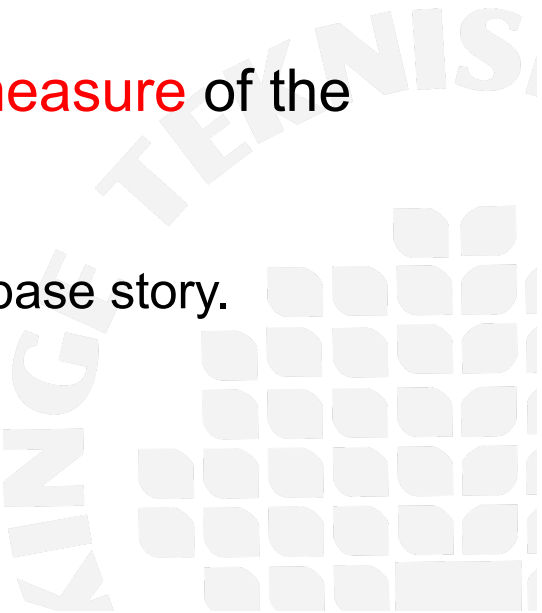
- Examples of estimation values for the cards:
  - 1, 2, 3, 5, 8, 13, 20, 40, 100.
  - ½, 1, 2, 3, 4, 5, 6, 7, ∞
- First one is most frequently used scale, and is referred as Fibonacci scale.



BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Planning Poker: Units

- Hours is an absolute unit and is used when you play planning poker to estimate effort required to implement a task in an iteration.
  - Ideal days/hours is another variation.
- An alternative is story points, which is a relative measure of the size of a user story.
  - Select a base story, and size it as x story points.
  - Other stories are then sized relative to the size of this base story.
- T shirt sizing
  - Small, medium, large, extra large



Planning Poker – an example

- User story 1: *As an administrator I shall be able to …..*
- User story 1 has following tasks (Just an example)
  - Task 1: Code the UI
  - Task 2: Code the ….
  - Task 3: Write test cases for …
  - …
- Task 1 estimates round 1 and round 2 (Example)

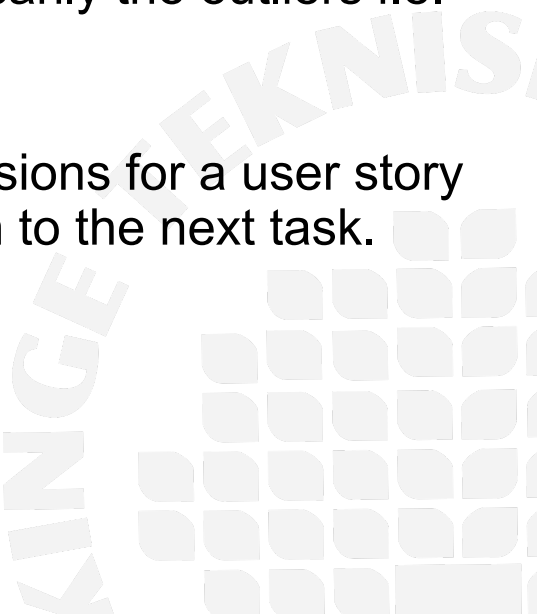
Estimator	Round 1	Round 2
Robert	3	5
Michael	1	3
Anna	5	5
Susan	8	5

- Michael and Susan are asked to explain their estimates at the end of round 1. Session lead decides to have round 2 after discussion.

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Variations

- Yes variations are possible
  - You can ask any member about his estimate, not necessarily the outliers i.e. the highest and lowest.
  - If required, you can use more or less number of rounds.
  - If no consensus is arrived after many rounds and discussions for a user story or task, then pend this task for the moment, and pass on to the next task.



BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

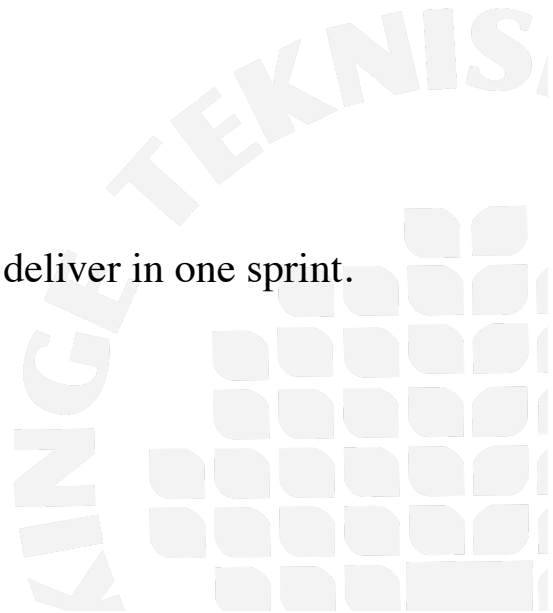
Advantages

- All team members participate.
- It minimizes problems related to anchoring.
- The dialogue between the members result in more accurate estimations.
- Studies have shown that averaging estimations and group discussion lead to better results.

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Velocity-Driven Iteration Planning

- Velocity
  - The rate at which a team can produce working software
    - Measured in story points per iteration/sprint
  - Can be artificially increased by cutting corners on quality
  - Should not be used as a measure of comparison across teams
- In velocity driven iteration planning
  - First, story size (complexity) is estimated.
  - Velocity is measured in terms of number of story points that a team can deliver in one sprint.
  - Based on size and velocity measurement, duration is derived.
- Unreliable in what will be accomplished during an iteration
  - Velocity is mostly useful over a long term



BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

Commitment Driven Iteration Planning

- Discuss the highest priority item on the product backlog
- Decompose it into tasks
- Estimate each task
  - Using planning poker, for example
- Evaluate: “ Can we deliver this item in current iteration”
  - Based on this evaluation, decide if you can pick more items from product backlog or remove this one.

BLEKINGE INSTITUTE OF TECHNOLOGY
BTH

References

- Mike Cohn (2005). Agile estimating and planning. Pearsons Education.
- James Grenning (2002). Planning poker or how to avoid analysis paralysis while release planning. Hawthorn Woods: Renaissance Software Consulting 3.
- Pressman, R.S. (2005). Software engineering: a practitioner's approach. Palgrave Macmillan.