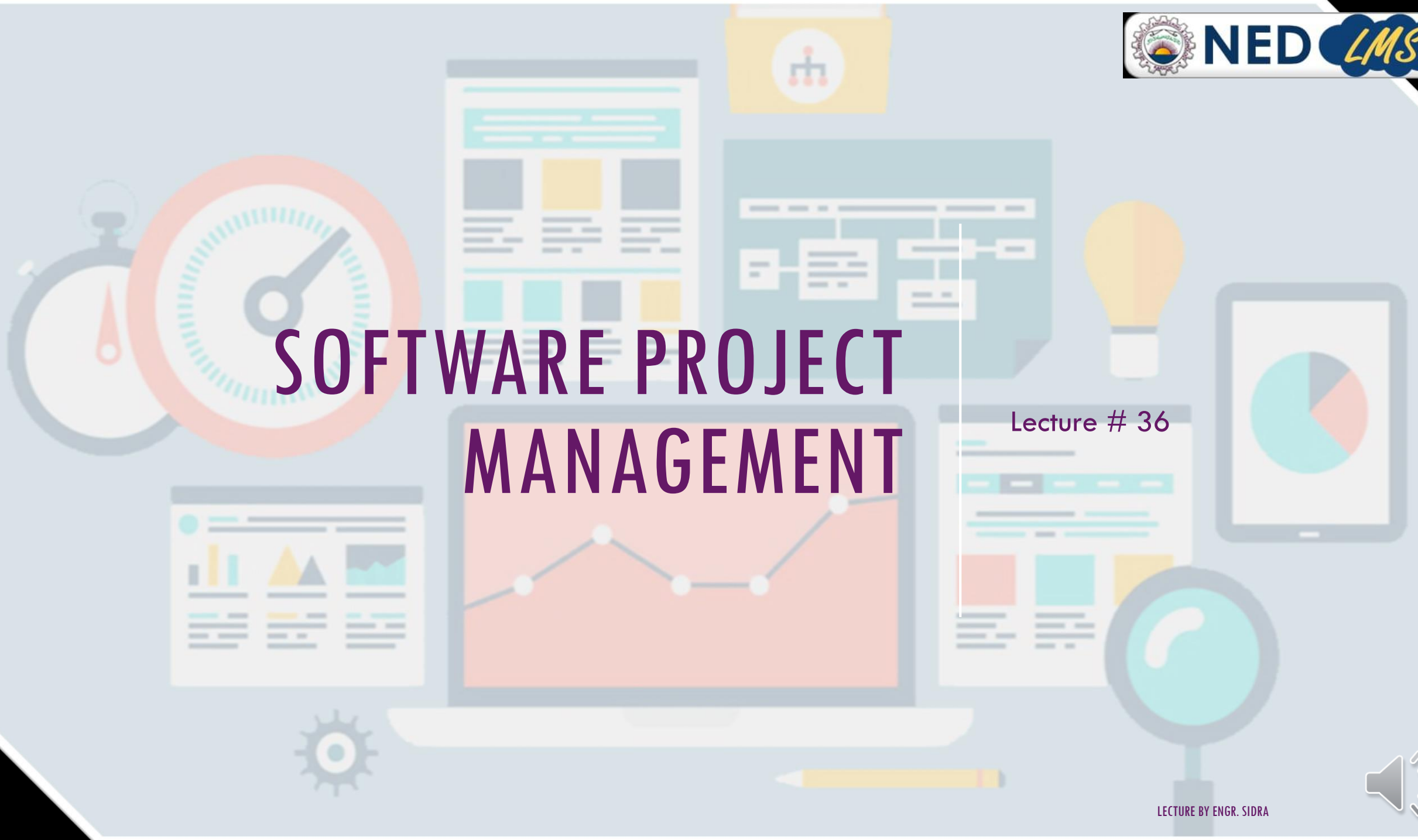
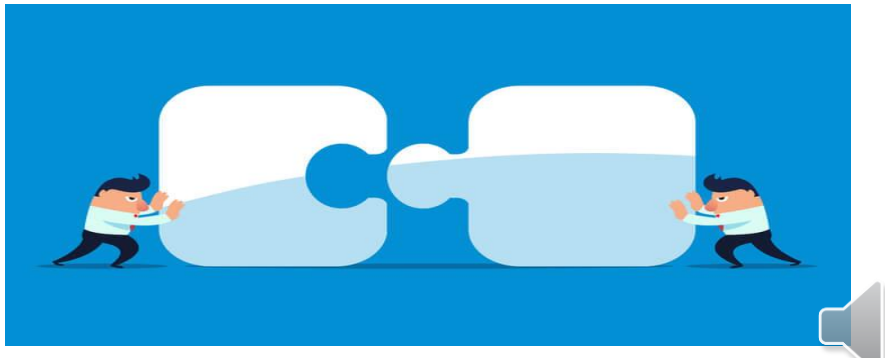
SOFTWARE PROJECT MANAGEMENT

Lecture # 36



THE PRODUCT

- Scope
  - Context. How does the software to be built fit into a larger system, product, or business context and what constraints are imposed as a result of the context?
  - Information objectives. What customer-visible data objects are produced as output from the software? What data objects are required for input?
  - Function and performance. What function does the software perform to transform input data into output? Are any special performance characteristics to be addressed?
- Software project scope must be unambiguous and understandable at the management and technical levels



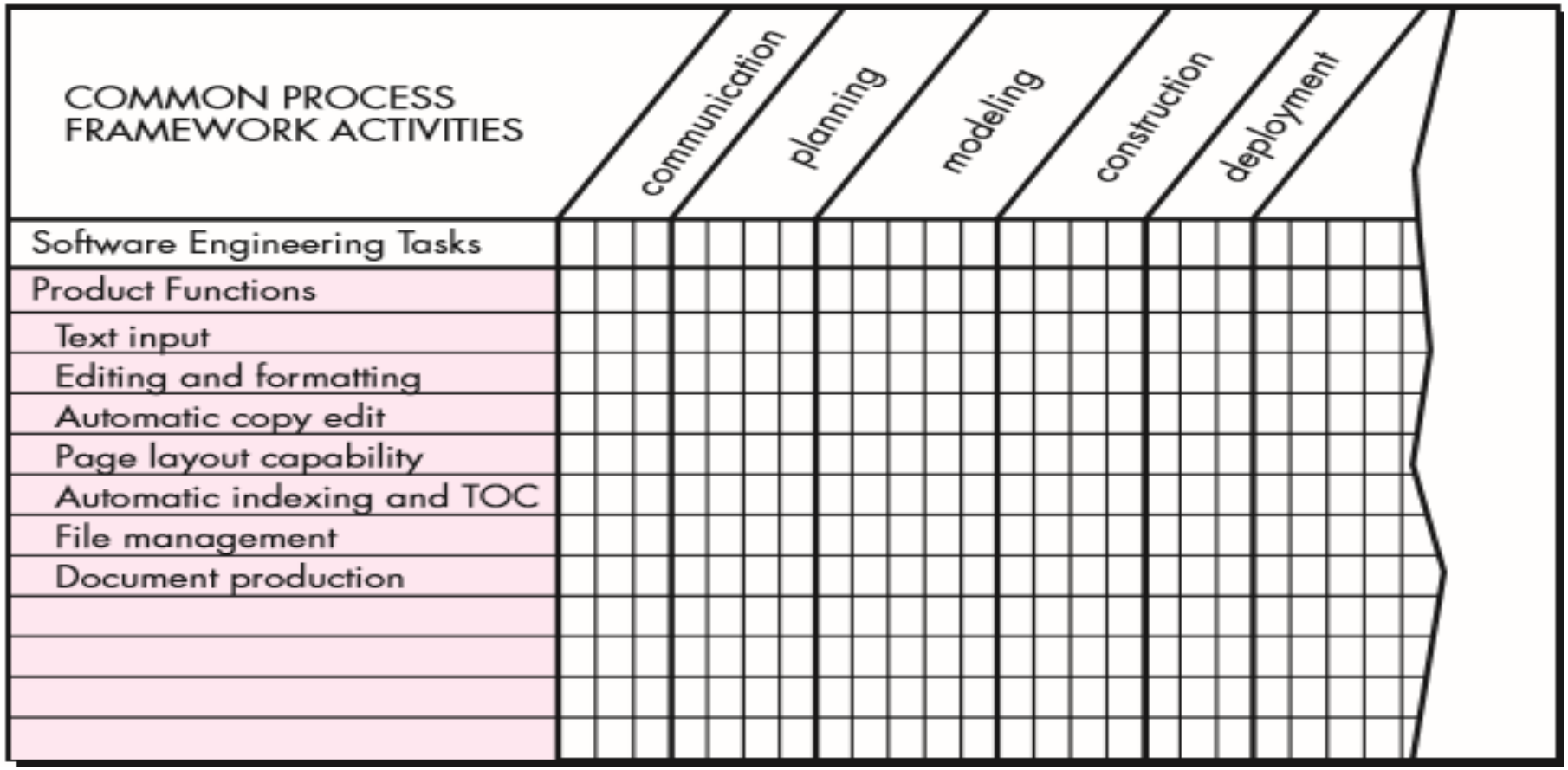
PROBLEM DECOMPOSITION

- Sometimes called partitioning or problem elaboration
- Once scope is defined …
  - It is decomposed into constituent functions
  - It is decomposed into user-visible data objects
    - or
  - It is decomposed into a set of problem classes
- Decomposition process continues until all functions or problem classes have been defined

THE PROCESS

- Once a process framework has been established
  - Consider project characteristics
  - Determine the degree of rigor required
  - Define a task set for each software engineering activity
    - Task set =
      - Software engineering tasks
      - Work products
      - Quality assurance points
      - Milestones

MELDING THE PROBLEM AND THE PROCESS



PROCESS DECOMPOSITION

- A software team should have a significant degree of flexibility in choosing the software process model that is best for the project and the software engineering tasks that populate the process model once it is chosen.
- The process framework is invariant and serves as the basis for all work performed by a software organization.
- But actual work tasks do vary. Process decomposition commences when the project manager asks, “How do we accomplish this framework activity?”

EXAMPLE: COMMUNICATION ACTIVITY TASKS

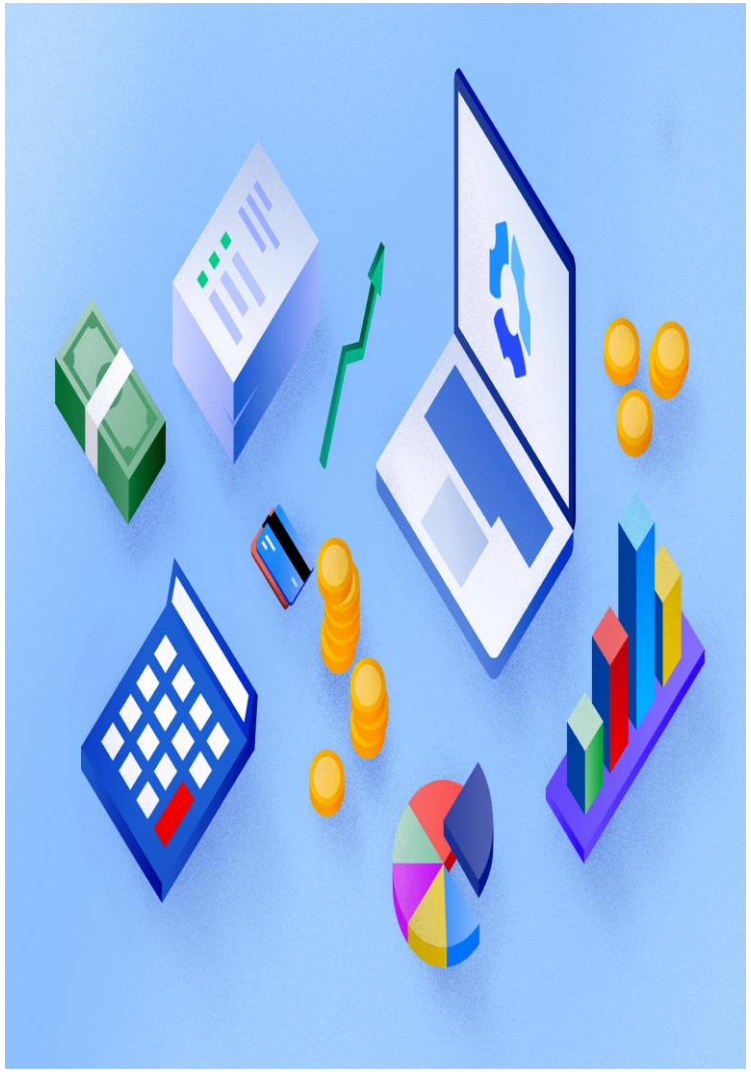
- For small Simple project:
  1. Develop list of clarification issues.
  2. Meet with stakeholders to address clarification issues.
  3. Jointly develop a statement of scope.
  4. Review the statement of scope with all concerned.
  5. Modify the statement of scope as required

EXAMPLE: COMMUNICATION ACTIVITY TASKS

- For Complex Project:
  1. Review the customer request.
  2. Plan and schedule a formal, facilitated meeting with all stakeholders.
  3. Conduct research to specify the proposed solution and existing approaches.
  4. Prepare a “working document” and an agenda for the formal meeting.
  5. Conduct the meeting.
  6. Jointly develop mini-specs that reflect data, functional, and behavioral features of the software. Alternatively, develop use cases that describe the software from the user’s point of view.
  7. Review each mini-spec or use case for correctness, consistency, and lack of ambiguity.
  8. Assemble the mini-specs into a scoping document.
  9. Review the scoping document or collection of use cases with all concerned.
  10. Modify the scoping document or use cases as required

THE PROJECT

- Projects get into trouble when …
  - Software people don't understand their customer's needs.
  - The product scope is poorly defined.
  - Changes are managed poorly.
  - The chosen technology changes.
  - Business needs change [or are ill-defined].
  - Deadlines are unrealistic.
  - Users are resistant.
  - Sponsorship is lost [or was never properly obtained].
  - The project team lacks people with appropriate skills.
  - Managers [and practitioners] avoid best practices and lessons learned



COMMON-SENSE APPROACH TO PROJECTS

- Start on the right foot.
  - This is accomplished by working hard (very hard) to understand the problem that is to be solved and then setting realistic objectives and expectations.
- Maintain momentum.
  - The project manager must provide incentives to keep turnover of personnel to an absolute minimum, the team should emphasize quality in every task it performs, and senior management should do everything possible to stay out of the team' s way.

COMMON-SENSE APPROACH TO PROJECTS

- **Track progress.**
  - For a software project, progress is tracked as work products (e.g., models, source code, sets of test cases) are produced and approved (using formal technical reviews) as part of a quality assurance activity.
- **Make smart decisions.**
  - In essence, the decisions of the project manager and the software team should be to "keep it simple."
- **Conduct a postmortem analysis.**
  - Establish a consistent mechanism for extracting lessons learned for each project.

THE W^{5}HH PRINCIPLE

- Why is the system being developed?
- What will be done?
- When will it be accomplished?
- Who is responsible?
- Where are they organizationally located?
- How will the job be done technically and managerially?
- How much of each resource (e.g., people, software, tools, database) will be needed?

Barry Boehm [Boe96]