# University of Mosul

## College of Computer Sciences and Mathematics

## Department of Computer Science


# Software Project Management

# Ch11: Agile Project Management



**Asst. Prof. Ashraf Abdulmunim Althanoon**

# ➢Agile Project Management

In any software business, managers need to know what is going on and whether or not a project is likely to meet its objectives and deliver the software on time with the proposed budget. Plan-driven approaches to software development evolved to meet this need.

Managers draw up a plan for the project showing what should be delivered, when it should be delivered, and who will work on the development of the project deliverables. A plan-based approach requires a manager to have a stable view of everything that has to be developed and the development processes.

# ➢Agile Project Management

## ➢What Is Agile Project Management

**“We are uncovering better ways of developing [products] by doing it and helping others do it. Through this work we have come to value:**

***Individuals and interactions over processes and tools***
***Working [products] over comprehensive documentation***
***Customer collaboration over contract negotiations***
***Responding to change over following a plan***

**That is, while there is value in the items on the right, we value the items on the left more.”**

# ➢Agile Project Management

## ➢What Is Agile Project Management

- ➢ Implementing agile projects
- ➢Fully supported production versions of partial solutions are released to the end user quarterly or semi-annually
- ➢ Intermediate versions are released to a focus group every 2-4 weeks
- ➢ Co-located agile project teams
- ➢ Cross project dependencies
- ➢ Project portfolio management

# ➢ Implementing agile projects

- Adding more functions and features to the solution and implementing them at the same time sounds great.
- The customer and end user can benefit from experiencing the solution unfolding over short periods of time, providing valuable feedback to developers on further additions and changes to the solution.
- Getting frequent client feedback is critical to discovering the complete solution and ultimately to project success.

# ➢Fully supported production versions of partial solutions are released to the end user quarterly or semi-annually

- This is commensurate with the organizational practices of the organization to implement the change,. Input from the end user should be well collected.

- There is a benefit in having longer periods to experiment and get comfortable with a new tool. You will gain valuable insight into the intuitive properties of your solution and see what the learning curve looks like.

- Getting frequent client feedback is critical to discovering the complete solution and ultimately to project success.

# ➢ Intermediate Versions Are Released to a Focus Group Every 2–4 Weeks

- You don’t stand idly by and wait for end-user feedback from the quarterly releases. That flies in the face of delivering business value early and often. You should ask them to commit to reviewing and critiquing every version of the solution.

- The focus group should have some of the client members of the project team on it as well as a few other key end users. A focus group of 10 members is a good working group.

# ➢Co-located agile project teams

- Every proponent of APM approaches advises using small co-located teams of highly skilled professionals who are assigned 100 percent to the project and who can work without supervision.
- The success of developing software projects depends on team members knowing what needs to be done and getting it done with little or no supervision. That’s a nice goal to strive for
- The reality is that the skills in most demand are in short supply and so the availability of individuals who possess those skills to work on a project is a problem. Out of necessity these professionals are assigned to multiple projects.

# ➢Cross project dependencies

- Consider this scenario. **Harry** is your only data warehouse design professional. When he finishes the data warehouse design on the Alpha Project, he is scheduled to begin the data warehouse design on the Beta Project. This raises the following management questions:
- **If Project Alpha is delayed, what is the impact on Project Beta?**
- **Who decides the project priority if there is a scheduling conflict with**
- **Harry?**
- **Can Harry's work on Project Alpha be overlapped with his work on**
- **Project Beta?**
- **What if Harry leaves the company?**

# ➢Project portfolio management

- **What should we do?**
- **What can we do?**
- **What will we do?**
- **How will we do it?**

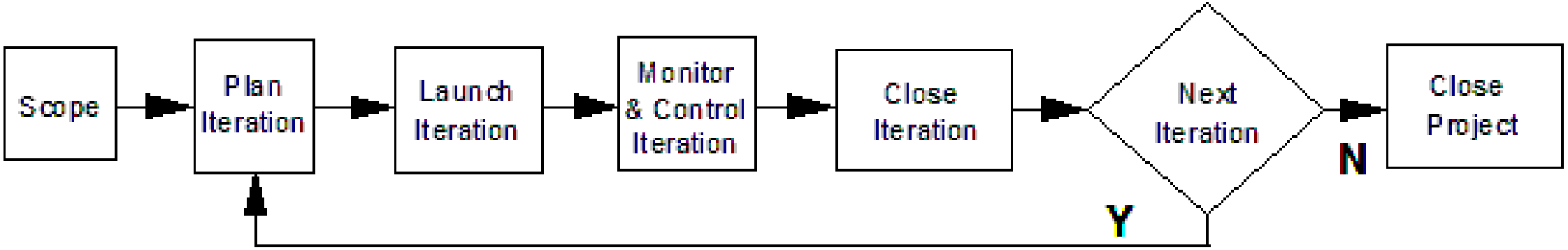
✓ The answer is a list of potential projects prioritized usually by business value. The answers to the next two questions can be based solely on the skills inventory and the availability of those skills over the planning horizon of the portfolio and the scheduling needs of the projects in the portfolio.

✓ The effective management of the contents of the project portfolio depends on access to a solid HRMS. There are commercially available software systems for portfolio management under a variety of resource constraints.

# Iterative Project Management Life Cycle Model

## Definition

- An Iterative Project Management Life Cycle (PMLC) Model consists of a number of phases that are repeated in groups with a feedback loop after each group is completed. At the discretion of the client the last phase in a group may release a partial solution.



# Iterative Project Management Life Cycle Model

- **Iterative approaches** are used when you have an initial version of the solution, but it is known to fall short in terms of features and perhaps functions.

- The iterative cycles are designed to select and integrate the missing pieces of the solution.

- The intermediate versions give the client something to work with as they attempt to learn and discover additional needed features.

# When to use an Iterative PMLC Model

- Most but not all of the solution is clearly known.
- You might otherwise have chosen the Incremental PMLC model but have a strong suspicion that there will be more than a minimum number of scope change requests.
- You might otherwise have chosen an Adaptive PMLC model but are concerned about lack of client involvement.

# Iterative PMLC Model

- **Scope Phase**
  - Implementation of intermediate solutions can be problematic
  - Final solution cannot be defined at the start of the project
- **Plan Phase**
  - The complete plan for building the known solution
  - The partial plan for the high priority functions

# Iterative PMLC Model

## ➢Launch Phase

| Characteristics | TPM project team | APM project team |
|---|---|---|
| Size | Could be very large | Usually less than 15 |
| Skill Level | All levels | Most skilled |
| Location | Co-located or distributed | Co-located |
| Experience level | Junior to senior | Senior |
| Position Responsibility | Requires supervision | In supervised |

# Iterative Project Management Life Cycle Model

❖**Monitoring and Controlling Phase of an Iterative PMLC Model**

- ✓ Care must be taken not to overload engineers and developers with these kinds of added tasks. Let them remain relatively free to continue with the other parts of the project.

❖**Closing Phase of an Iterative PMLC Model**

- ✓ The closing phase of the iterative PMLC model is similar to the closing phase of the TPM PMLC model in that there are customer-defined criteria that must be met in order for the job results to be considered complete.

# Iterative Project Management Life Cycle Model

## Characteristics

- The solution is known but not to the expected depth (i.e., features are  not complete)

- Often uses iconic or simulated prototypes to discover the complete solution

# Iterative Project Management Life Cycle Model

## Strengths

- Client can review current solution for suggested improvements
- Can accommodate scope changes between iterations
- Adapts to changing business conditions

# Iterative Project Management Life Cycle Model

## Weaknesses

- Requires a more actively client than Linear and Incremental PMLC models require
- Requires co-located teams
- Implementation of intermediate solutions can be problematic
- Final solution cannot be specified at the start of the project

# Iterative Project Management Life Cycle Model

Types of Iterative PMLC Models

- Prototyping
- Rational Unified Process (RUP)

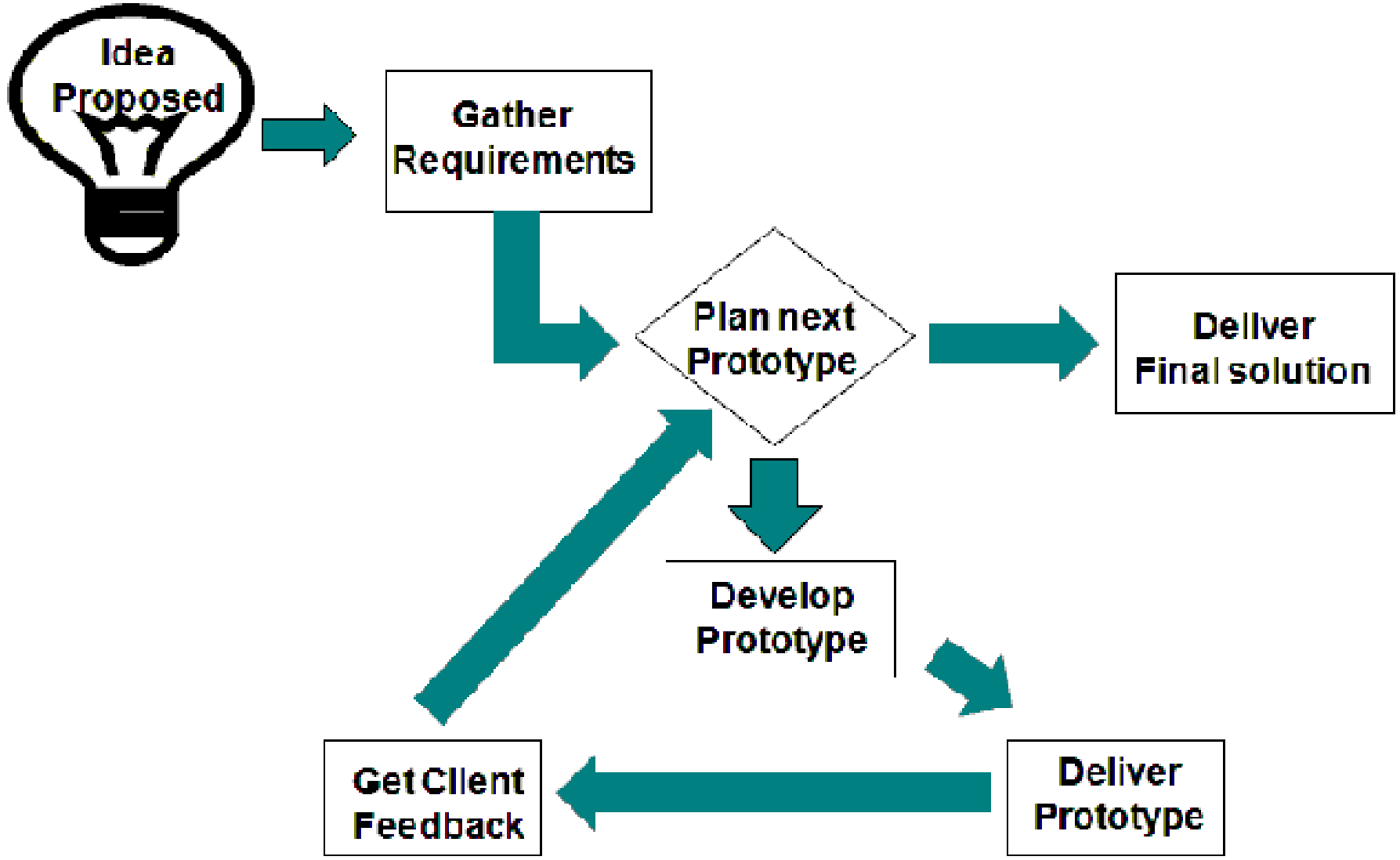
# Iterative Project Management Life Cycle Model

## Prototyping

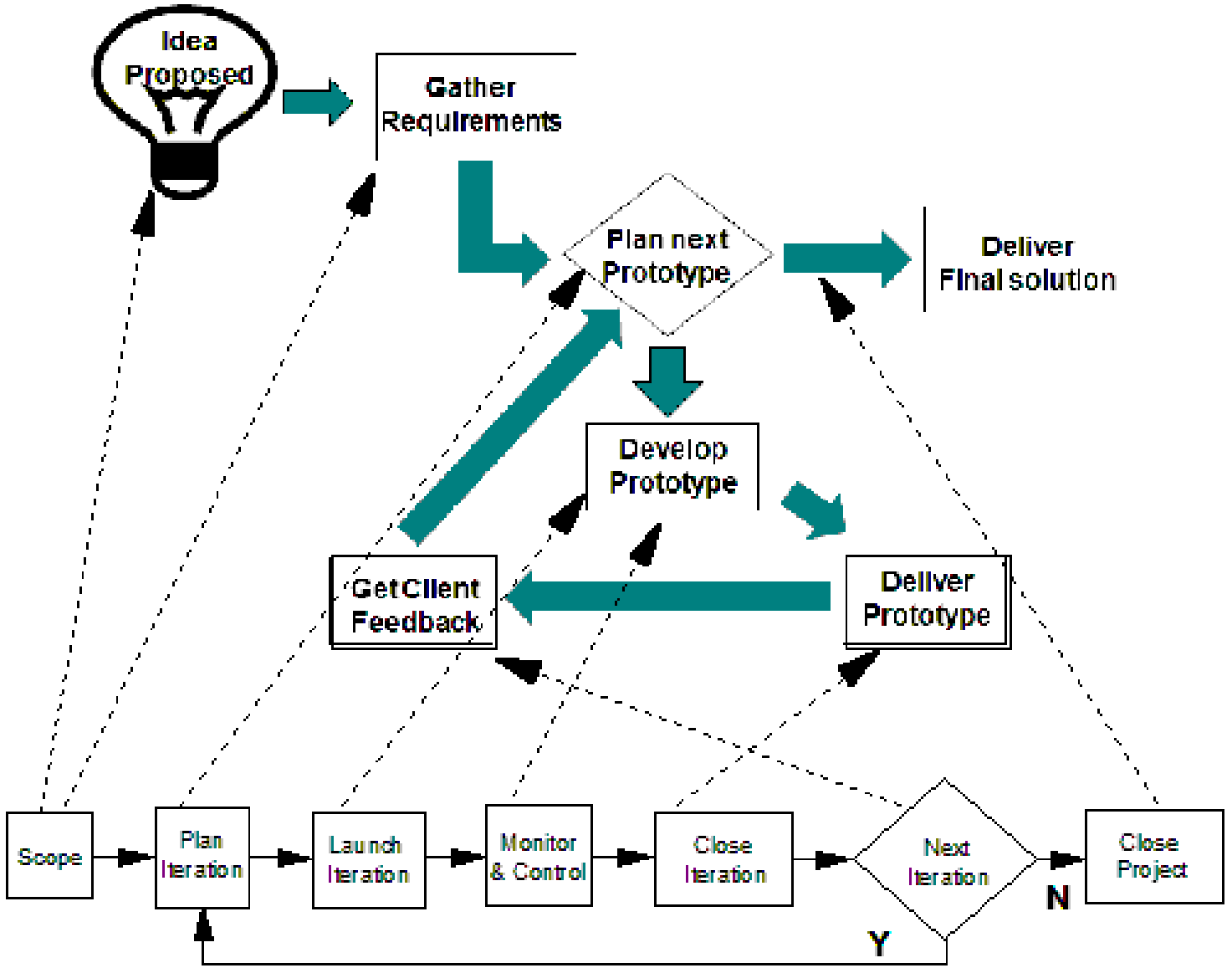
Prototyping Model is a software development model in which prototype is built, tested, and reworked until an acceptable prototype is achieved. It also creates base to produce the final system or software. **It works best in scenarios where the project's requirements are not known in detail**. It is an iterative, trial and error method which takes place between developer and client.

# Iterative Project Management Life Cycle Model

## Prototyping



# Prototyping mapped to Iterative PMLC Model

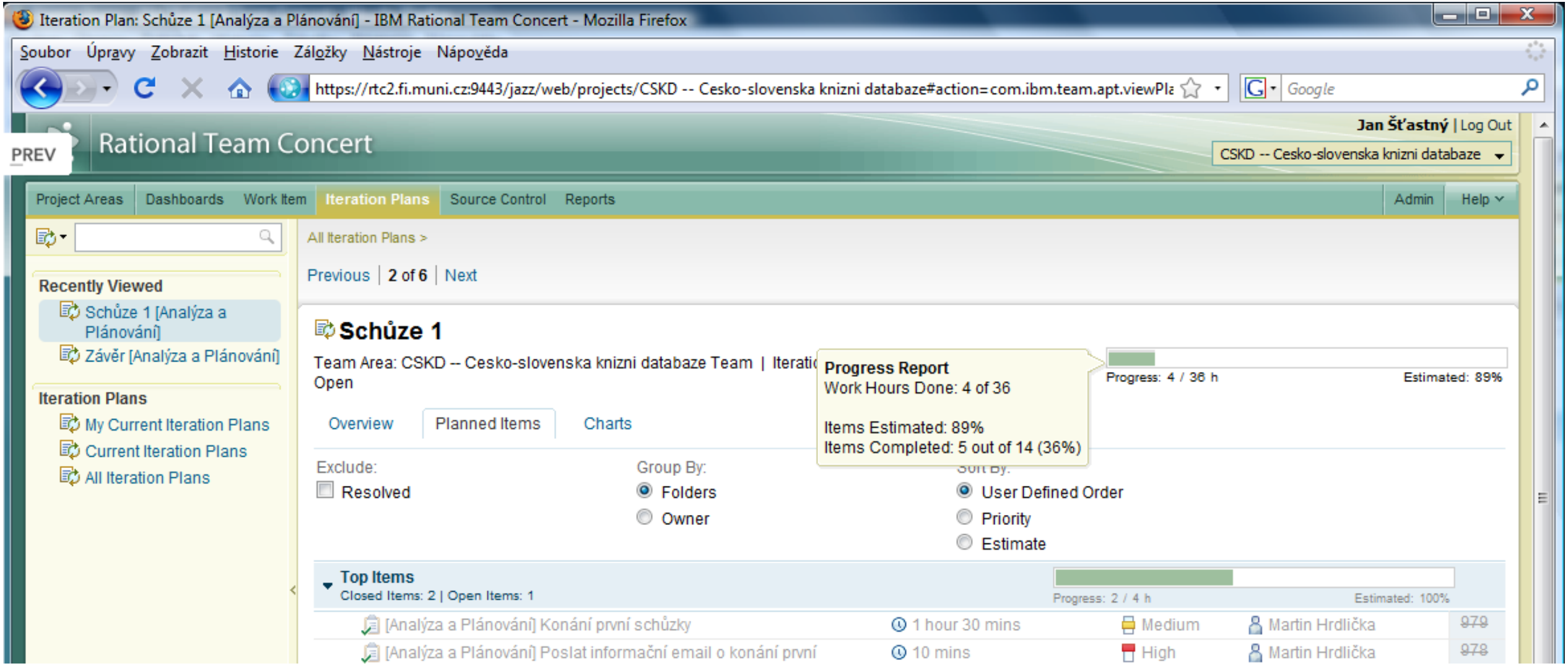


# Rational Unified Process (RUP)

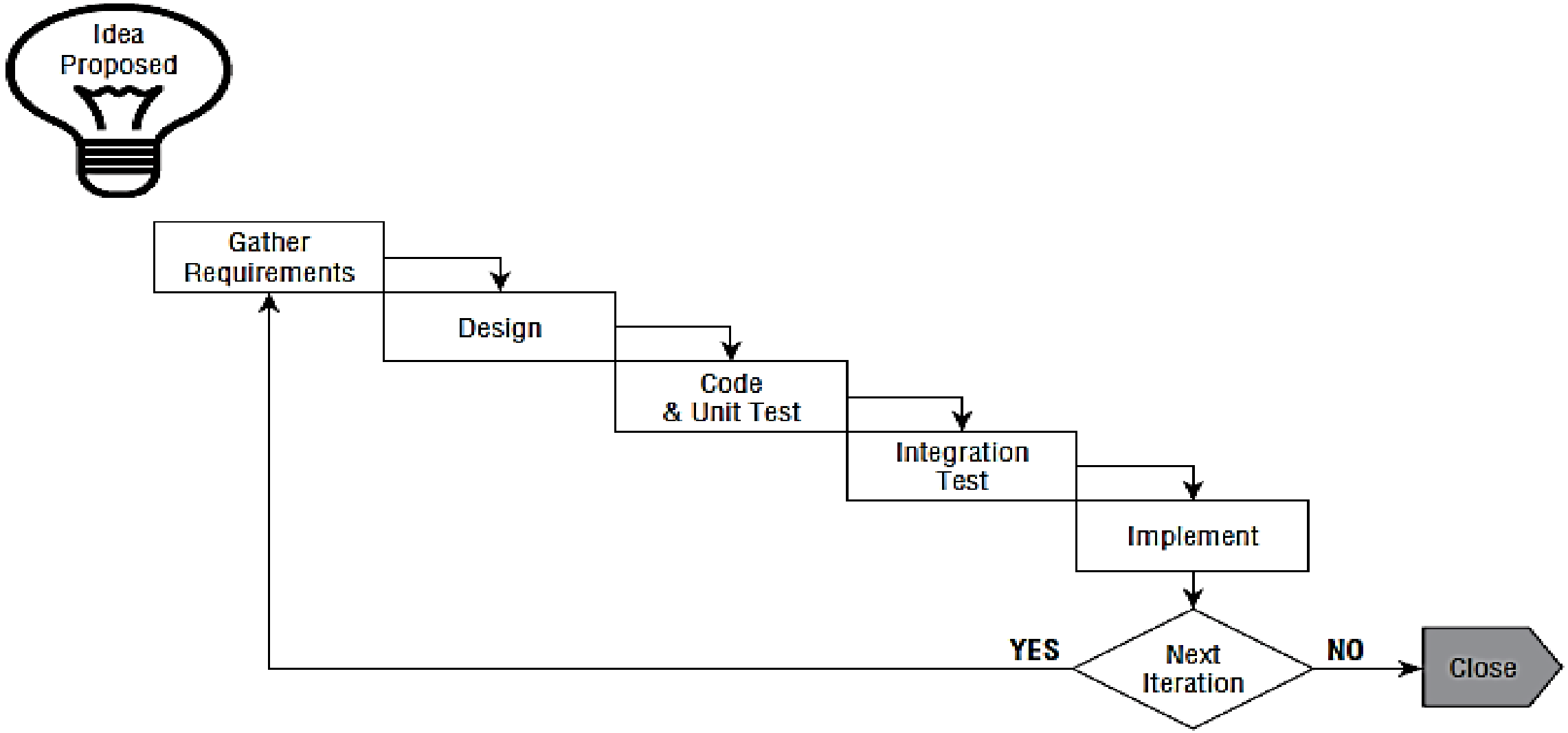
Rational Unified Process (RUP) is a software development process for object-oriented models. It is also known as the Unified Process Model. It is created by Rational corporation and is designed and documented using UML (Unified Modeling Language). This process is included in IBM Rational Method Composer (RMC) product. IBM (International Business Machine Corporation) allows us to customize, design, and personalize the unified process. RUP is proposed by Ivar Jacobson, Grady Bootch, and James Rambaugh. Some characteristics of RUP include use-case driven, Iterative (repetition of the process), and Incremental (increase in value) by nature, delivered online using web technology, can be customized or tailored in modular and electronic form, etc.

**RUP reduces unexpected development costs and prevents wastage of resources**

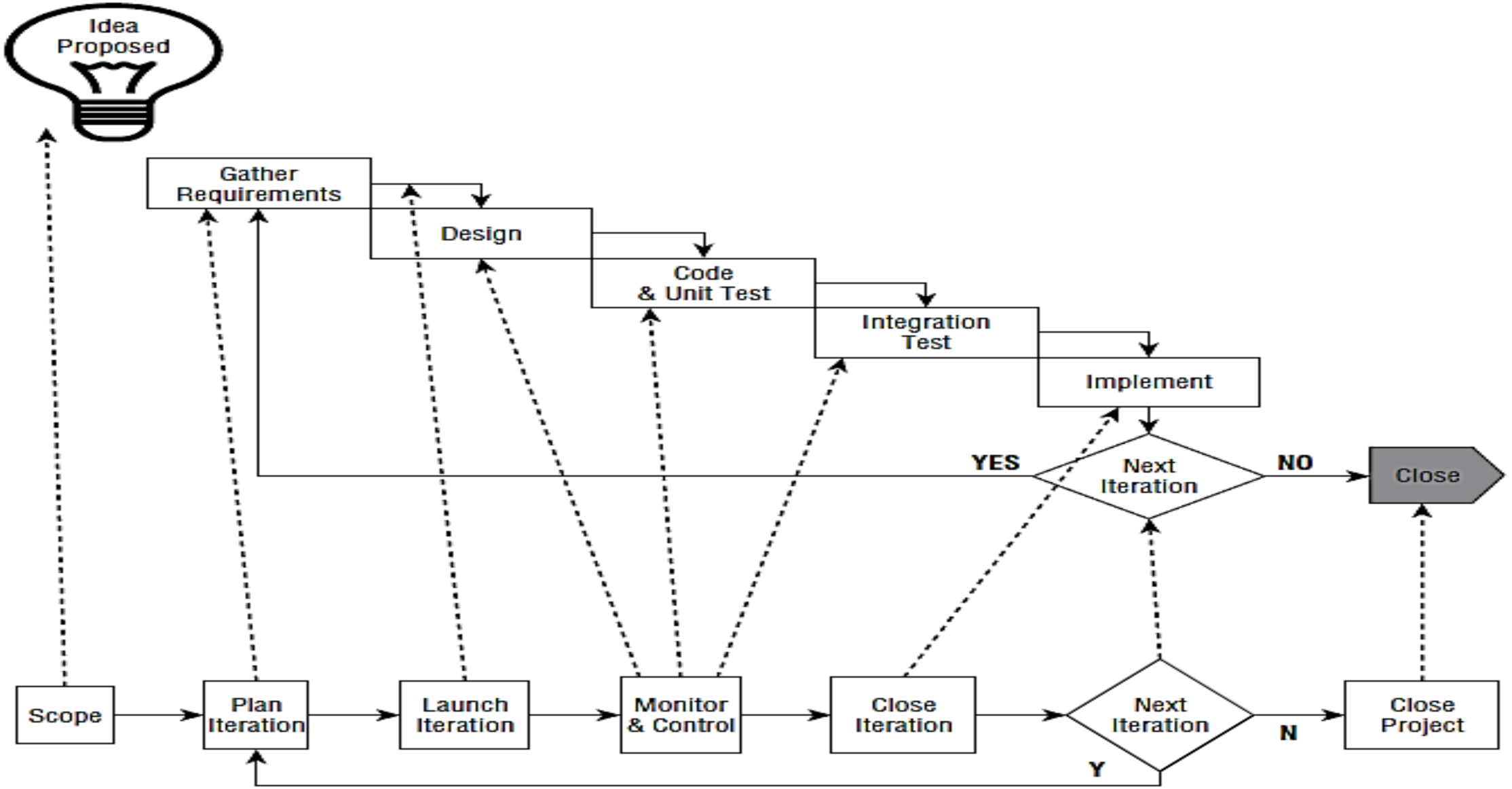
# RUP mapped to Iterative PMLC Model



# RUP mapped to Iterative PMLC Model



# RUP mapped to Iterative PMLC Model



# Rational Unified Process (RUP)

**The main phases in a RUP approach are as follows:**

1. Inception
2. Elaboration
3. Construction
4. Transition

# Rational Unified Process (RUP)

**Inception :**

Inception has as its objective the definition and concurrence of all the stakeholders as to the scope of the software development project. The scope is bounded by a number of use cases that define the functions that the software system must perform. Cost, schedule, and risk are also estimated as a preparation for the Elaboration Phase.

# Rational Unified Process (RUP)

**Elaboration:**

Elaboration is the engineering phase of a RUP project. It is here that the details of the problem and its solution are formed and the architecture is finalized. That permits more refined estimates of time, cost, and risk. Prototypes are often built as an aid to the design considerations, more detailed functionality, and features.

# Rational Unified Process (RUP)

**Construction:**

In the Construction Phase, the current design is turned into a working system. If this is phase has been repeated, then the most recent designs are integrated into the current solution and a more enhanced solution is turned over to the client.

# Rational Unified Process (RUP)

## Transition

Transition turns over a solution that the client and end user can put into production. It need not be a complete solution, but it does need to have sufficient business value to be released to the end user by the client. Later minor enhancements will be made to integrate features defined but not integrated.

# Iterative PMLC Model

## When to Use an Iterative PMLC Model

- Iterative models are intuitive to the client
- Iterative models easily engage the client
- Iterative models give immediate feedback on the effect of solution changes
- Iterative models are most tolerant of assessing and evaluating the impact of alternatives
- There is no fixed deadline for completion

Thank you