

# Best Practices - Software Development

**Indu R Porwal**

**Senior Manager Solutions Development, IEEE**

**Email: [i.porwal@ieee.org](mailto:i.porwal@ieee.org)**

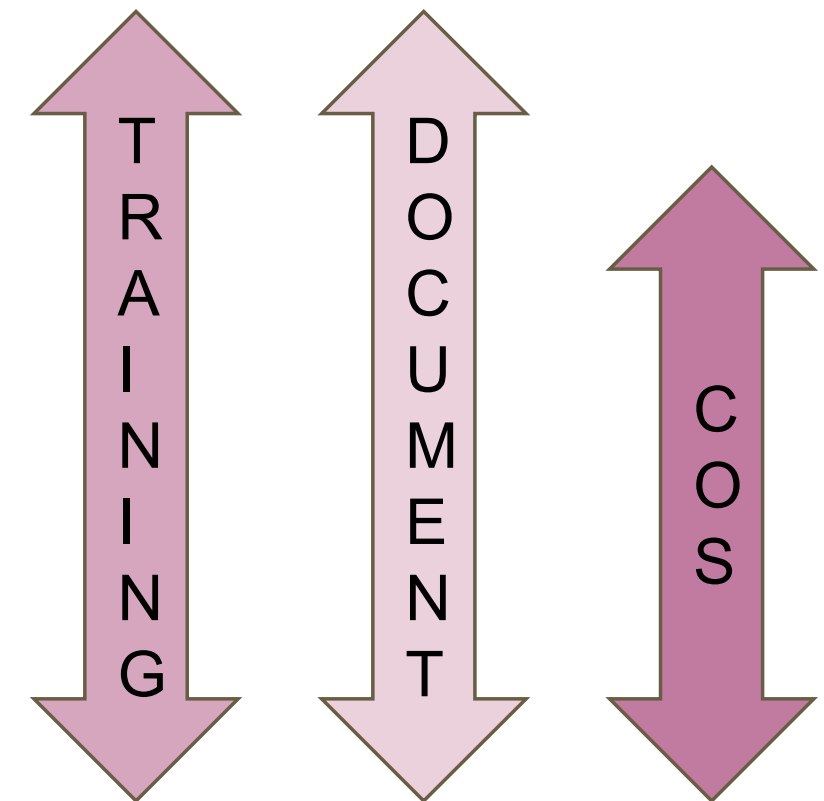
**Linkedin: <https://www.linkedin.com/in/induporwal/>**

# Software Development

1. Project Initiation: Project Development Estimate
2. Team formation: Recruit team members
3. Project Planning : Project Plan , Set Milestones, Targets
4. Requirements Analysis
5. Design
6. Build
7. Testing
8. Implementation
9. Maintenance

AGILE

Retrospect



Own It

Communicate  
More often

Think Out of  
Box

what inspires  
you?

# 1. Project Estimation

1. **What is the purpose?**
  2. **How users intend to use the product?**
  3. Complexity of the product
  4. Highlight Risks/Unknowns
  5. Assumptions
  6. **Probability of Changes**
  7. Availability of client to clarify requirements - Set expectations
  8. Cost estimates - probability of change
-

# Why user wants what's being asked for?

Developer understanding



User ambition



## 2. Development Team formation

1. Define High Performing Team
  2. Define skills of Team
  3. Define Team Training Plan
    - a. Explain Purpose
    - b. Remind Purpose
    - c. Evaluate Understanding
-

# 3. Project Planning

1. Define Project Plan
  2. Define Stakeholders
  3. Defines Roles and Responsibilities
  - 4. Identify Steering Committee**
  - 5. Identify frequency of communication with every stakeholder**
  6. Explain methodology - **AGILE**
  - 7. Set expectations for engagement**
  - 8. Escalation Criteria & levels**
  9. Identify Operational Processes
  - 10. Technical debt Sprints planning**
-

## 4. Requirements Analysis:

1. Requirements mostly capture **what to do, not why**
  2. **Ask, Ask and Ask**
  3. Develop prototype and get early feedback
  4. Discuss alternatives
-



# 5. Design

1. Maintenance - Readability, Modularity, Scalability
  2. Define high level framework
  3. Define **Data Privacy Rules**
  4. Define **permissions** on every action
  5. Define **Coding Best Practices**
  6. Define **Security Considerations**
  7. Set up review process
  8. Set up **Security Scan** process
-



# Reviews & feedback



1. Peer Reviews
2. User feedback
3. Set up a mandatory process for reviews
4. Constructive feedback
5. Is everyone participating?

## 6. Build

1. **Automate** Operational processes
  2. Integrate build with minimum **test** coverage
  3. Define Change Control process
  4. Define build frequency & communication plan
  5. Publish Release notes
-

# 7. Testing

1. Review Purpose
  2. Positive Testing
  3. Negative Testing
  4. **Automated Test Scripts**
  5. Different **browsers**
  6. Different **devices**
  7. **Usability**
  8. Performance - Load Testing
-

# 8. Implementation

1. Define Go-live plan
  2. Explain Roles & Responsibilities
  3. Inform **users** if required
  4. Plan for buffer time
  5. Team on Standby
  6. Define **Disaster Recovery Plan**
  7. Define **back up plans**
-

# 9. Maintenance

1. Define **day to day** operational activities
  2. Roles & Responsibilities
  3. What is **bug**? What is **defect**?
  4. **Service Level Agreements**
  5. Plan regular **Disaster recovery exercises**
-

# Key Principles

1. Brutal Honesty
  2. Communication
  3. Inspired team
  4. Innovation
  5. Retrospect
-

# Brutal Honesty



1. Being Vulnerable
2. Start from I
3. Acceptance of failures
4. Team owns all success and all failures
5. Learn from failures
6. Schedule time to discuss lessons learnt



# Communication



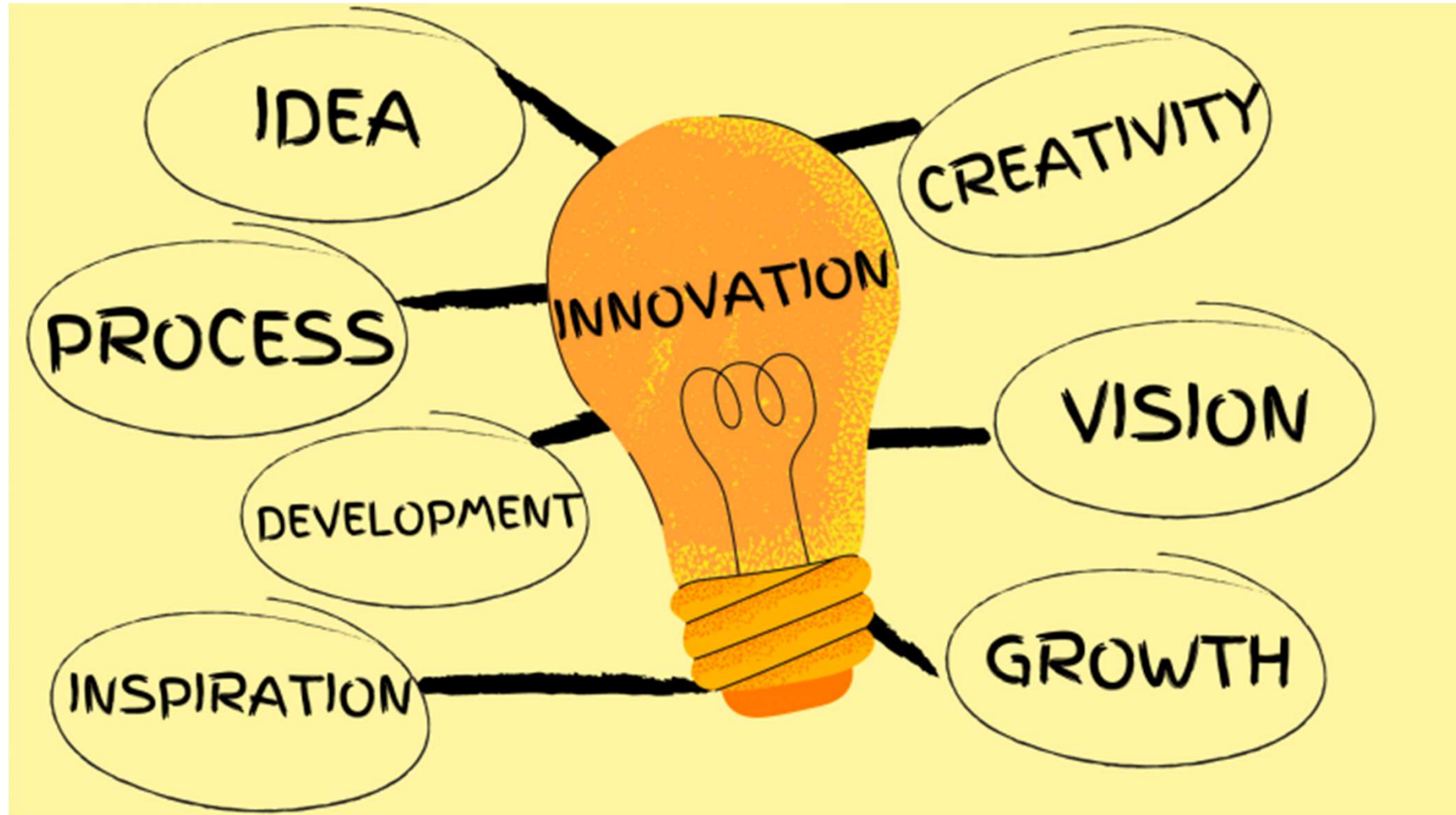
1. Identify stakeholders
2. Identify frequency of updates
3. What updates
4. Escalation Mechanism
5. More is better
6. Upwards & Downwards
7. Steering Committee

# Inspired Team

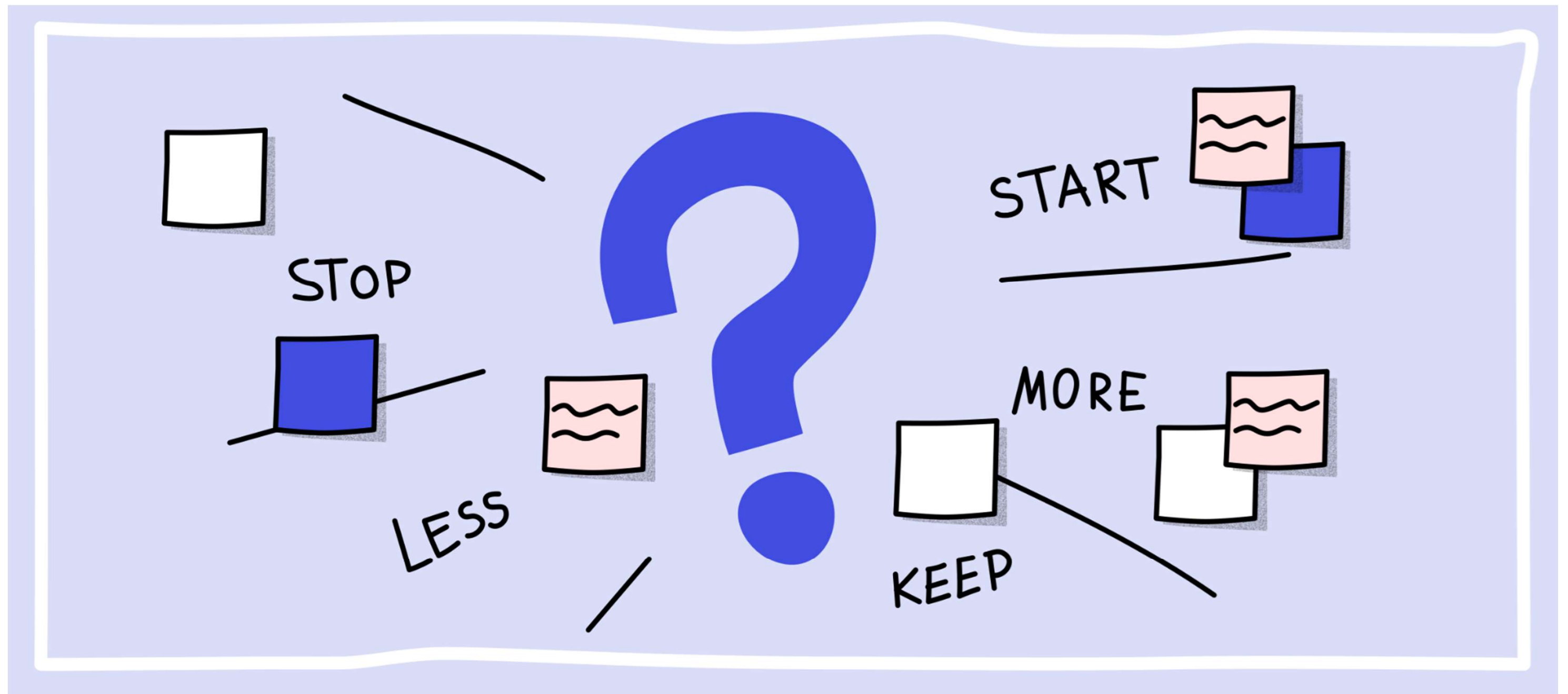


1. Discipline
2. Flexibility
3. Sense of Achievement
4. Each others support

# Innovation



# Retrospect frequently





# Thank you

Indu R Porwal

Email: [i.porwal@ieee.org](mailto:i.porwal@ieee.org)

Linkedin: <https://www.linkedin.com/in/induporwal/>

---