

# Notes on Software Product Management Week 7 Jan 5

## Key Topics Covered

### 1. Prototyping:

- **Four Dimensions:** Visual, Breadth, Depth, Interaction.
- Tools like Balsamiq for wireframing were introduced.
- Importance of usability testing and design principles.
- Assignment: Create wireframes for a parking app (Search and Results screens).

### 2. Agile Methodology:

- **Essence of Agility:** Fast response and flexibility to change.
- Agile is about effectively responding to changes, not just following processes like Scrum.
- **Key Characteristics:**
  - Speed and flexibility.
  - Ability to change direction effectively.
  - Collaboration between teams and stakeholders.

### 3. Agile Manifesto:

- **Four Core Values:**
  - Individuals and interactions over processes and tools.
  - Working software over comprehensive documentation.
  - Customer collaboration over contract negotiation.
  - Responding to change over following a plan.
- **Twelve Principles:**
  - Deliver early and continuously.
  - Welcome late-stage changes.
  - Frequent delivery of working software.
  - Daily collaboration between business and developers.
  - Build projects around motivated individuals.
  - Use face-to-face communication for efficiency.
  - Maintain sustainable development pace.
  - Continuous attention to technical excellence and good design.
  - Simplicity in maximising work not done.
  - Allow self-organising teams.
  - Reflect and adapt regularly.

### 4. Agile Planning:

- **Sprint Calendar:** Typically one or two weeks.
  - Backlog refinement (Wednesday/Thursday before Sprint).
  - Sprint planning (Monday).
  - Daily standups (Tue-Wed-Thu).
  - Sprint review (Friday) — Demo of completed work.
  - Retrospective (End of Sprint or beginning of next Sprint).
- **Backlog Grooming:**
  - Ensure readiness of stories.
  - Prioritise based on importance.
  - Factor in holidays and dependencies.
- **Daily Standups:**
  - Focus on: What was done, what will be done, and blockers.
  - Keep it brief (15 minutes).

## 5. Effectiveness vs. Efficiency:

- Effectiveness: Doing the right thing (most important).
- Efficiency: Doing things right with fewer resources.
- Key questions to assess effectiveness:
  - Are users finding and using the feature?
  - Are they getting value?
  - Are there drop-offs or support issues?
- Decide whether to improve a feature, leave it as is, or discard it.

## 6. Estimations in Agile:

- **Types:** Budgetary and working estimates.
- Budgetary: High-level and approximate (e.g., 2 months of effort).
- Working: Detailed planning done by the team doing the work.
- Relative Sizing:
  - Use T-shirt sizes (XS, S, M, L, XL, XXL) or Fibonacci sequence (1, 2, 3, 5, 8, 13).
  - Helps prioritise work and determine team velocity.
- Spend ~20% of budgetary estimate time to create working estimates.

## 7. Persistent Teams:

- Keep team structure constant for efficiency.
- Avoid frequent changes to minimise "Form-Storm-Norm-Perform" cycles.
- Storm phases can arise from changes in team members, work profiles, or priorities.

## 8. Sprint Retrospectives:

- Focus on: What went well, what could be better, and future improvements.
- Ensure psychological safety to encourage open discussion.

## Suggested 5-Mark Questions with Humanized Answers

### Q1: Can you explain the main values behind the Agile Manifesto?

**Answer:** The Agile Manifesto provides four key values that help teams work effectively:

1. **People and Communication Over Tools:** Agile prioritizes collaboration and open communication over strict reliance on tools or rigid processes.
2. **Functional Software Over Documents:** Instead of focusing on lengthy documents, Agile ensures the product itself works well and meets user needs.
3. **Working Closely With Customers:** Teams partner with customers to create solutions that adapt to their requirements instead of strictly adhering to pre-set contracts.
4. **Flexibility Over Fixed Plans:** Agile embraces changes to meet new challenges rather than sticking to outdated plans.

These principles ensure products deliver real value and adapt to changing situations effectively.

---

## Q2: Why are Sprint Retrospectives so important in Agile projects?

**Answer:** Sprint Retrospectives allow teams to continuously improve their work by asking three key questions:

1. "What did we do well?" — Reflecting on successes helps teams identify effective practices.
2. "What could be better?" — Teams discuss challenges and areas for improvement.
3. "What actions should we take next?" — Identifying concrete changes ensures improvement.

These meetings also promote open discussions and create a safe space for sharing feedback, which strengthens collaboration and productivity in the long run.

---

## Q3: How does the Fibonacci sequence improve task estimations in Agile?

**Answer:** The Fibonacci sequence (1, 2, 3, 5, 8, 13, etc.) helps teams estimate tasks by grouping them into manageable sizes. Here's how:

1. **Addresses Uncertainty:** Larger numbers reflect increasing difficulty or uncertainty for bigger tasks.
2. **Makes Planning Simpler:** Teams assign story points based on task sizes, making it easier to prioritize work.
3. **Encourages Productivity:** Teams focus on completing smaller tasks first, improving overall efficiency.
4. **Tracks Progress:** Velocity (completed story points per Sprint) can be used for better planning.
5. **Facilitates Team Agreement:** The sequence encourages discussions that lead to consensus on task complexities.

Using Fibonacci numbers aligns well with Agile's goal of iterative improvement and adaptability.

# Software Product Management 8 Week

## Difference Between UI and UX

UI (User Interface): Focuses on the look and layout of the product.

UX (User Experience): Encompasses the overall feel of the product, including usability, efficiency, and satisfaction.

Example: Airbnb's UI is visually appealing, but a bad UX example was shared where a booking was cancelled, and the refund took 5-7 days, causing inconvenience.

## Product Success Rates and MVPs (Minimum Viable Products)

Product Success Rate: Less than 10% of products succeed.

MVP Strategy: Instead of investing all resources into one MVP, iterate and test multiple versions to increase chances of success.

Build-Measure-Learn Loop: Emphasizes iterative development to refine the product based on user feedback.

## Prioritization Techniques

RICE Method: Reach, Impact, Confidence, Effort.

Reach: Number of people affected.

Impact: Benefit to users.

Confidence: Certainty of success.

Effort: Resources required.

Moscow Method: Must do, Should do, Could do, Won't do.

Helps in categorizing tasks based on their importance and urgency.

## **Recap of Weeks 1-7**

Week 1: Differences between project and product businesses, types of products, and product management basics.

Week 2: Reasons for product failure, product life cycle, and technology adoption curve.

Week 3: Understanding customer needs, design thinking, and product ecosystems.

Week 4: Product risks, lean mindset, and MVPs.

Week 5: Customer journey maps, story maps, and user stories.

Week 6: Prototyping, design principles, and usability testing.

Week 7: Agile methodology, estimation, planning, and people aspects in agile teams.

Important Questions (5 Marks Each)

### **Explain the difference between UI and UX with an example.**

Answer: UI refers to the visual elements of a product, such as colors, layout, and typography. UX encompasses the overall experience of using the product, including ease of use, efficiency, and satisfaction. For example, Airbnb's UI is visually appealing, but a bad UX example was shared where a booking was canceled, and the refund took 5-7 days, causing inconvenience.

### **What is the significance of the MVP strategy in product development?**

Answer: The MVP strategy involves creating a minimum viable product to test and validate ideas with minimal resources. It allows for iterative development, where feedback is used to refine the product. This approach increases the chances of success by avoiding large investments in unproven ideas and focusing on continuous improvement.

### **Describe the RICE method of prioritization and its components.**

Answer: The RICE method is a prioritization framework that considers Reach, Impact, Confidence, and Effort.

Reach: The number of people affected by the feature.

Impact: The benefit to users.

Confidence: The certainty of success.

Effort: The resources required to implement the feature.

Features with higher RICE scores are prioritized as they promise higher value with lower effort.

**What are the key stages of the product life cycle, and what activities are involved in each stage?**

Answer: The product life cycle includes Introduction, Growth, Maturity, and Decline stages.

Introduction: Launching the product and achieving product-market fit.

Growth: Increasing user base and usage.

Maturity: Maximizing profitability and reducing costs.

Decline: Managing the product's end-of-life and transitioning users to new products.

**How does the Agile methodology help in responding to changes effectively?**

Answer: Agile methodology emphasizes iterative development and flexibility. It involves working in small increments, allowing teams to respond quickly to changes in the environment or user needs.

Agile practices like daily stand-ups, sprint planning, and retrospectives ensure continuous improvement and adaptability.