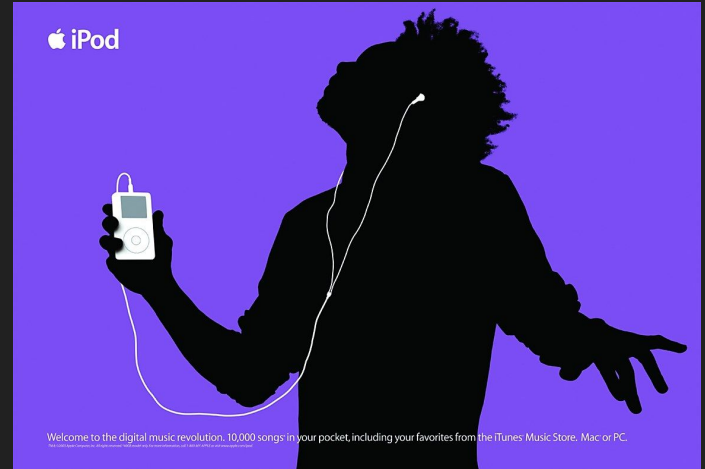


Product Owners Guide

Product Management – Background



ipod Example

Clear vision: "1000 songs in your pocket"

Deep understanding of user pain points

Relentless focus on user experience

Result = Revolutionary product that created new markets +
led to dominance of iPhone

Magic Leap Example

Tech in search of a problem

No clear user need being solved

Impressive tech but no compelling "why"

Result = Expensive failures despite massive resources (\$ multi billions)

The Product Management Spectrum

Broken

Works Well



Features built that nobody uses

Teams work in silos without context

Great tech with 0 market fit

Endless meetings without decisions

Too tech focussed & not connected to actual user needs

Solve real user problems

Teams = aligned and motivated

Ship features users actually want

Business value is clear and measurable

The "why" is always clear to everyone

"If a product manager is making all the decisions, then they are not a good product manager."

Build: An Unorthodox Guide to Making Things Worth Making

Bad

Makes all the decisions & becomes a bottleneck

Good

creates the context for the team to make great decisions



"Use blue for that button, not green"

"Write the error message exactly like this"

"Implement the API this specific way"

"Schedule the deploy for Tuesday at 2pm"

"Use React instead of Vue for this component"

"Users are confused at this step - how might we make it clearer?"

"The goal is sub-200ms response time - what's the best approach?"

"Support is overwhelmed with password reset tickets - how do we solve this?"

"We need to ship by Q3 for the holiday season - what's realistic?"

Why Good PM Matters

The Cost of Poor Product Management

Wasted Resources + Market Failure + Demoralization

Value of Strong Product Management

Build products + features that users want & will actually use + find PMF

TF Build – Product Management Framework

Vision → Data

Start with vision “The Why” then validate with data

The Virus of Doubt

Help users realize their current solution isn't good enough

Make the Intangible Tangible

Prototype the entire customer journey, not just the product

Decision Framework

PM/PO Should Own

WHAT problems to solve
(vision/strategy)

WHY it matters (business case)

WHEN it ships
(timeline/priorities)

Success metrics (how we measure)

Team Should Own

HOW to solve it (technical approach)

Design details (UI/UX specifics)

Implementation choices (tech stack, architecture)

Task breakdown (sprint planning / task allocation)

Core Product Owner/Manager Responsibilities

"The 'why' of your product is the most important thing. If you can't explain why someone needs your product (or feature), then they probably don't."

Build: An Unorthodox Guide to Making Things Worth Making

Responsibilities

Product Management sits at the intersection of

Business : What makes money + aligns with company goals ?

Technology : What is actually possible ?

User Experience : What do users need & want ?

Key Responsibilities - ext

Vision & Strategy

Help define & understand the Why

Create + maintain product roadmap

Align product with company objectives

Customer Advocacy

Understand the problem deeply before building solutions

Deep understanding users pain points (regular feedback / interviews)

Remember: "Users aren't numbers" - data shows what, conversations show why

Delivery & Iteration

Prioritize ruthlessly based on value

Write clear requirements (PRDs/Concept Docs)

Ship early, gather feedback, iterate

Own the outcome, not just the output

Team Collaboration

Help bridge gaps between engineering, bd, marketing etc

Facilitate without dictating

Share & gather context constantly

Product Owner vs Product Manager

Product Manager

Strategic focus (the "what" and "why")

Market and business oriented

Owns the vision and roadmap

Product Owner (Scrum specific)

Tactical focus (the "what" and "when")

Team and delivery oriented

Owns the backlog and sprint goals

Best Practices & Tips

Start with the Problem, Not the Solution

Uber started with "can't find a cab in Paris"

What daily frustration are we solving?

Master the Art of Storytelling

Emotional and rational appeal

Simple explanation of complex concepts

Constant reminder of the problem being solved

Focus on the Entire Customer Journey

Discovery: How do they find us?

Onboarding: First experience

Usage: Daily interaction

Support: When things go wrong

Advocacy: How they share with others

Best Practices (Agile + Remote from Atlassian)

Developer Shared Understanding Over Documentation

Why are we doing X, How does this solve user pain points ?

Ensure all teams aligned on the why

Write the press release before you build (Fadell's trick)

Be Agile About Being Agile

Focus on outcomes, not processes for process sake

Start simple, evolve as needed

Where to start ? (your first week)

Talk to users (don't just read data & make assumptions)

Understand the tech (don't just blindly start building but understand options / limitations)

Build team relationships (particularly with relevant cross team stakeholders)

Map the current user journey

Action Items & Next Steps

This Week

Schedule user interviews - Each P0 should talk to 1 user this week

Create a one-page press release - Why does your product exist?

Map the full customer journey - From discovery to advocacy

Review existing roadmap / epics - Apply new prioritization lens

Ongoing

Weekly check-ins - Share learnings and challenges

Read- "Build" by Tony Fadell

Mentor - Pair with experienced POs ?

Practice User Journey Mapping / Concept Doc / PRD writing

Summary

Find the Balance

Between vision and data, saying yes and no, leading and empowering

Focus on the "why"

If you can't explain why something matters to users, it probably doesn't!

Ship and learn

Perfect is the enemy of good; iteration beats perfection

You're not alone

Product ownership is hard, lean on the team and ask for help