IF-938

Week 3 - Defining The Software Problem

Firhat Hidayat, S.Kom., M.T.

INSTITUT
TEKNOLOGI
HARAPAN
BANGSA

Veritas vos liberabit

Referensi

Utama

- Tim Brown (2019), Change by Design
- Michael Lewrick, Patrick Link, and Larry Leifer (2018), The Design Thinking Playbook: Mindful Digital Transformation of Teams, Products, Services, Businesses and Ecosystems
- Jake Knapp, John Zeratsky, and Braden Kowitz (2016), Sprint: How to Solve Big Problems and Test New Ideas in Just Five Days
- IDEO.org (2015), The Field Guide to Human-Centered Design
- Tom Kelley and David Kelley (2013), Creative Confidence
- Jeanne Liedtka, Andrew King, and Kevin Bennett (2013), Solving Problems with Design Thinking: Ten Stories of What Works
- Nigel Cross (2011), Design Thinking: Understanding How Designers Think and Work
- Jeanne Liedtka and Tim Ogilvie (2011), Designing for Growth: A Design Thinking Toolkit for Managers
- Marc Stickdorn and Jakob Schneider(2010) , This is Service Design Thinking: Basics, Tools, Cases
- Tom Kelley (2001), The Art of Innovation: Lessons in Creativity from IDEO, America's Leading Design Firm
- Don Norman (1988), The Design of everyday things

Agenda

Week 2

1. What is Problem Definition?
2. Why is Defining the Problem Crucial in Software Development?
3. Characteristics of a Good Problem Statement
4. Techniques for Crafting Problem Statements

Assignment

1. Hands-on Activity: Crafting a Problem Statement
2. Assignment: Write a Problem Statement

What is Problem Definition?

Problem Statement

- **Definition:** Problem definition is the **process** of **understanding** and **articulating** the **core challenge** you're trying **to solve**.
- **Importance in Software Development:** A clear problem statement guides the entire design and development process, ensuring the team focuses on solving the right issue.

Why Defining the Problem is Crucial in Software Development

Problem Statement

- **Prevents Wasted Effort:** Helps avoid building features or products that don’t solve the user’s actual problems.
- **Focuses the Team:** Creates alignment across the team around what needs to be solved.
- **Guides the Design Process:** Sets the foundation for ideation and prototyping.

Case Study

Real-World Example: Misaligned Problem Definition
Nokia’s Failure in the Smartphone Market

Background

Case Study

- Nokia pernah menjadi pemimpin global dalam industri ponsel, terkenal karena daya tahan dan desain inovatifnya.

- Namun, ketika pasar ponsel pintar mulai beralih ke layar sentuh dan sistem operasi yang lebih canggih, Nokia kesulitan untuk beradaptasi dan kehilangan posisinya yang dominan, akhirnya tertinggal di belakang para pesaing seperti Apple dan Google.

The Callenge

Case Study

- Masalah utama Nokia bukan hanya kegagalan untuk berinovasi, tetapi juga kegagalan untuk memahami kebutuhan penggunanya yang berubah dengan cepat. Ketika Apple memperkenalkan iPhone pada tahun 2007 dengan antarmuka layar sentuh yang intuitif dan ekosistem aplikasinya, Nokia tetap fokus pada perangkat keras dan peningkatan bertahap pada sistem operasi Symbian yang sudah ada.

- The core issue was that Nokia defined the problem as: **“We need to improve our phone hardware and maintain the Symbian OS to stay competitive.”**

- In contrast, the real problem was: **“Users want a seamless, modern smartphone experience, with a focus on apps, an easy-to-use interface, and ecosystem connectivity.”**

Reality behind captain America's shield

The Result

Case Study

- Nokia’s misaligned problem definition led to several missteps:
    - **Symbian OS:** Nokia terus berinvestasi pada OS Symbian, yang sudah ketinggalan zaman dan tidak efisien dibandingkan dengan pengalaman pengguna yang mulus yang ditawarkan oleh iOS dan Android.
    - **Late Shift to Touchscreens:** Nokia terlambat mengadopsi ponsel berbasis layar sentuh sepenuhnya, dan malah memilih merilis model hybrid yang memiliki keypad fisik di samping layar sentuh, yang tidak sesuai dengan keinginan pengguna untuk pengalaman yang lebih modern dengan layar sentuh penuh.
    - **App Ecosystem:** Nokia meremehkan pentingnya membangun ekosistem aplikasi yang kuat, sesuatu yang telah dijadikan landasan strategi oleh Apple dan Google.

Lesson and Takeway

Case Study

- Empathy and User-Centered Design
    - Nokia gagal berempati dengan penggunanya, yang semakin mencari lebih dari sekadar perangkat keras yang andal. Pengguna menginginkan pengalaman menyeluruh yang terintegrasi secara mulus ke dalam kehidupan digital mereka —sesuatu yang sudah dipahami Apple dan Android sejak awal. Mereka menyadari bahwa ponsel pintar berkembang menjadi perangkat multifungsi untuk bekerja, bermain, dan tugas sehari-hari.
- Key Lessons
    - **Misunderstanding the Core Problem:** Nokia terlalu fokus pada peningkatan teknologi yang sudah ada (perangkat keras dan OS Symbian) dan gagal menyadari bahwa kebutuhan pengguna sebenarnya telah bergeser ke arah pengalaman perangkat lunak yang terintegrasi dan didorong oleh aplikasi.
    - **Failure to Empathize with Users:** Alih-alih memahami secara mendalam apa yang diinginkan pengguna di era baru ponsel pintar (kemudahan penggunaan, ekosistem aplikasi, dan antarmuka yang elegan), Nokia tetap terfokus pada kekuatan mereka yang ada, yang semakin menjadi tidak relevan.
- The Takeaway
    - Kasus ini menunjukkan bahwa mendefinisikan masalah yang salah dapat menyesatkan bahkan perusahaan yang paling sukses sekalipun. Kegagalan Nokia dalam mendefinisikan masalah dari perspektif yang berpusat pada pengguna menyebabkan penurunan mereka di pasar ponsel pintar, yang pada akhirnya dikuasai oleh perusahaan-perusahaan yang fokus pada pemahaman dan pemenuhan kebutuhan pengguna yang terus berkembang.
    - Studi kasus ini menggambarkan pentingnya mendefinisikan masalah dengan benar berdasarkan riset pengguna dan empati. Ini menyoroti bagaimana bahkan pemimpin pasar bisa kehilangan posisinya dengan berfokus pada masalah yang salah dan gagal memahami apa yang sebenarnya diinginkan oleh pengguna.

Good Problem Statement

Characteristics of a Good Problem Statement

Good Problem Statement

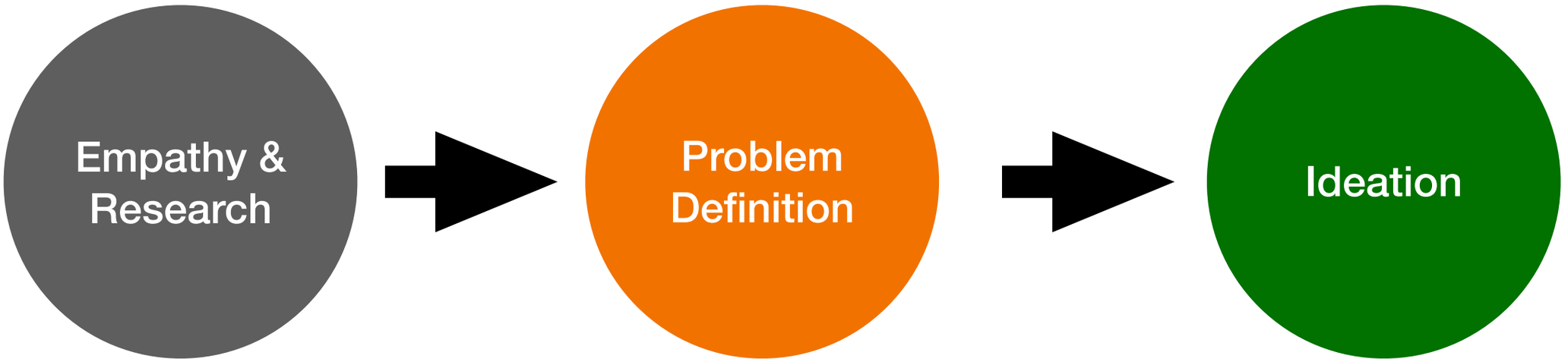
- **User-Centered:** Focuses on the user and their needs.
- **Specific and Clear:** Clearly defines what the problem is and avoids ambiguity.
- **Actionable:** Provides a direction for solution development.
- **Open-Ended:** Leaves room for creative solutions, not prescriptive.

- **Example (Good):** “Users of our travel booking app are frustrated by the lack of personalized recommendations, leading to longer search times.”

- **Example (Bad):** “Our app’s UI is not great.”

Problem Definition in Design Thinking

Good Problem Statement



- A strong problem definition bridges the empathy stage (user research) and ideation (solution development).

Techniques for Crafting Problem Statements

Point of View (POV) Framework

Techniques

- **Who** is the user?
- **What** are their needs?
- **Why** is this important?

- Template:
  - **[User]** needs a way to **[user's need]** because **[insight]**.
- Example:
  - [Frequent travelers] need a way to
  - [find personalized recommendations quickly]
  - because [they often feel overwhelmed by the vast amount of information available when booking trips.]"

5 Whys Method

Techniques

- Ask “Why?” five times to get to the root of the problem.
    - Example: “Why do users abandon the checkout process?” → “Because the process is too long and complicated.”

Example:

1. **Why** do users abandon the shopping cart? Because the checkout process is too long.
2. **Why** is it too long? Because there are too many required fields.
3. **Why** are there so many fields? Because we ask for unnecessary details.
4. **Why** do we ask for unnecessary details? Because our system requires them for every user.
5. **Why** does our system require them? Because it wasn’t designed with the user’s needs in mind.

Resulting Problem Statement:

- “Users abandon the checkout process because there are too many required fields, which are unnecessary for most users. How might we simplify the process to reduce the number of fields?”

HMW (How Might We) Questions

Techniques

- Transform the problem into a question that invites creative solutions.
    - Example: “How might we simplify the checkout process for users?”

Template:

- **How might we [action] for [user] in order to [desired outcome]?**

Example:

- “How might we simplify the checkout process for online shoppers in order to reduce cart abandonment?”

Problem-Solution Fit Template

Techniques

- This template frames the problem in a structured manner that helps articulate the challenge and leaves room for potential solutions.

Template:

- **The problem is that [problem] affects [user] and results in [impact]. A successful solution would [solution criteria].**

Example:

- “The problem is that users of our fitness app struggle to maintain long-term engagement, which results in high churn rates. A successful solution would allow users to set personalized goals and track their progress more easily.”

Job-to-be-Done (JTBD) Statement

Techniques

- The JTBD framework focuses on the outcome users are trying to achieve rather than just their immediate needs. This shifts the problem focus toward the user's underlying goals.

Template:

- **When [situation], [user] wants to [motivation] so that [expected outcome].**

Example:

- "When using public transportation, commuters want to receive real-time updates so that they can plan their routes and avoid delays."

Problem Definition Framework (for Software)

Techniques

- This pattern is often used to define technical or software-related problems, where it's important to explain the root cause and its impact on the user experience.

Template:

- **We have observed that [user group] experiences [problem] when [situation], which causes [impact]. How might we improve [process or product] so that [desired outcome]?**

Example:

- "We have observed that new users experience confusion when setting up their account for the first time, which causes delays in using the platform. How might we improve the onboarding process so that users can start quickly and without frustration?"

Recap and Key Takeaways

Summary

- Definisi masalah yang **jelas** sangat penting untuk membangun solusi software yang sukses.
- **Good problem statement** berpusat pada pengguna, spesifik, dan dapat ditindaklanjuti.
- Teknik seperti POV (Point of View), 5 Whys, dan HMW (How Might We) dapat membantu menyusun pernyataan masalah.
- Tahap **problem definition** menjembatani kesenjangan antara empati dan ideasi.

Assignment

Team Assignment

- Based on the user research you conducted in Meeting 2, craft a problem statement for a software feature or application using **All Technique**.
- S**ubmission:** Due next class. Prepare to present your problem statement in the next session.

Thanks

Do you have any question?

firhat@ithb.ac.id

Resource

- **Reference**
- Garcia-Molina, H., Ullman, J.D., and Widom, J. Database Systems The Complete Book, 2nd Edition, Pearson Education, Inc. 2009.
- Silberschatz, A., Korth, H. F., Sudarshan, S., Database System Concepts, 6th Edition, McGraw-Hill, 2011.

- **Photo and Image**
- Freepik Stories