

D-TILE Techniques Part III: Concept Development & Reducing Uncertainty

Due: Mon, Oct 6 @ 12pm

Submission: Please submit a PDF version of your assignment to Canvas.

Problem Space

You've already explored needs, assumptions, and directions for a real MIT student problem (Homework 1). You've developed your first prototype, albeit for a separate problem (Homework 2). Now it's time to **develop concepts** and **identify the major uncertainties** that might make or break your project.

This assignment will push you to move from **abstract problem framing** → **tangible design concept** while practicing **risk analysis and reduction**.

Part 1: Concept Development (Steps 6–9 of the 12 Steps)

6. Specify Desired Outcomes

- Define **clear, measurable outcomes** your solution should achieve.
- Examples: "Reduce time spent finding food options by 50%," "Enable students to schedule study groups within 2 minutes," "Provide stress-relief interactions students can access in under 10 seconds."

7. Concept Generation

- Brainstorm at least **5 different concepts** for how your system could solve the problem.
- Push for variety: different platforms (chatbot, app, Raspberry Pi device, hybrid).
- Document ideas quickly... sketches, bullet points, short descriptions.

8. Concept Downselection

- Select **1 concept** to move forward with.
- Justify your decision by comparing feasibility, impact, and alignment with outcomes.

- Briefly note why you eliminated other options.

9. Concept Articulation & Prototyping

- Create a detailed description of your chosen concept (appearance, user experience, system behavior).
 - Provide **visuals**: sketches, diagrams, wireframes, or mock-ups.
 - Build a **low-fidelity prototype** (paper, cardboard, Wizard-of-Oz demo, or digital mock-up).
-

Part 2: Reducing Uncertainty

1. Identify at least **7 key uncertainties** in your project (technical feasibility, user adoption, clarity of interaction, cost, etc.).
 2. For each, rate:
 - **Uncertainty Level** (1–5)
 - **Difficulty to Test** (1–5)
 3. Create a **Risk vs. Cost graph** plotting all uncertainties.
 4. Draw your **Risk Tolerance Threshold** line.
 5. For all items above the threshold, propose **1–2 quick, inexpensive experiments** you can run in the next week (e.g., paper user test, quick API call, cardboard model, peer survey).
-

Why This Matters

- You'll move beyond abstract problem framing into **real prototypes** you can test.
- Early uncertainty reduction prevents wasted effort and helps you adapt quickly in a short 6-week project cycle.
- Together, this sets you up for **usability testing and iteration** in the weeks ahead.