Required Try-It Activity 6.1: Applying Experimentation to Your Organization

Formulate an experiment to help you address a business decision.

Task

In this assignment, you will have the opportunity to devise an experiment to help you address a real-world business decision. This decision could be about a new business venture, innovation, or untested strategy. It could be specific (like pricing) or broad (like product design).

Answer the following questions:

1. What is the decision you need to make?

Should Concentrix Enterprise Technology develop and launch a "Cloud-Native Modernization Accelerator" platform that uses AI-driven code analysis and automated refactoring tools to help enterprise clients migrate legacy applications to cloud-native architectures?
This platform would combine automated assessment, migration planning, and continuous optimization capabilities

2. How will your business benefit from making the right decision? What is the cost if you make the wrong decision? Please provide relevant examples.

Benefits of the right decision:

- *Potential $50-75M new revenue stream targeting the $1.3+ trillion global digital transformation market*
- *Accelerate delivery timelines for Application Services Engagement (30-50% faster based on industry benchmarks)*
- *Enhanced competitive positioning against Accenture, Deloitte, and IBM Consulting*
- *Improved margins through automated tooling reducing manual effort*
- *Cross-selling opportunities across existing Technology Transformation Clients*

Costs of the wrong decision:

- *$3-5M platform development investment with no ROI (industry average for failed digital transformation projects is $4.12M)*
- *18-24 months of engineering resources diverted from billable client work*
- *Missed market opportunity as enterprises increasingly demand automated modernization*
- *Potential client defection to competitors offering similar capabilities*
- *Internal team frustration from building unused technology*

3. What information are you missing that would help you make the right decision? Provide examples for your response.

- ***Market demand validation:*** *How many enterprise clients would pay premium rates for automated modernization vs. traditional consulting approaches?*
- ***Technical accuracy:*** *Can AI-driven code analysis reliably identify modernization patterns across diverse legacy technologies (.NET, Java, COBOL)?*
- ***ROI demonstration:*** *What measurable time / cost savings can we guarantee compared to manual modernization approaches?*
- ***Integration Complexity:*** *How seamlessly does the platform integrate with existing CI/CD pipelines and cloud platforms (AWS, Azure, GCP)?*
- ***Competitive differentiation:*** *What unique capabilities would distinguish our platform from existing tools AWS Application Discovery Service or Azure Migrate?*

4. Describe an experiment that you could conduct to help you learn the answer.

   - If your question is specific (e.g., choose A vs. B, or set the optimal price), describe a convergent experiment that could help you find the answers.
   - If your question is open-ended (e.g., “What does the customer want most in this new product?"), then describe a plan for a divergent experiment. What would a prototype or minimum viable product (MVP) look like? (i.e., how could you maximize learning while minimizing cost?

Prototype / MVP Approach: *Create three specialized assessment modules targeting different modernization scenarios:*

1. *Legacy monolithic decomposition with microservices recommendations*
2. *Database modernization with cloud-native database suggestions*
3. *Infrastructure-as-Code generation from existing configurations*

Experiment Structure:

- *Partner with 4 existing enterprise clients across different industries (manufacturing, financial services, retail, healthcare)*
- *Deploy prototype modules against their actual legacy applications (non-production environments)*
- *Run 6-week assessment pilots per client using hybrid approach (automated analysis + expert validation)*
- *Focus on one application per client to limit scope and risk*

Maximize Learning / Minimize Cost:

- *Leverage existing client relationships and ongoing engagements*
- *Use open-source code analysis tools as foundation rather than building from scratch*
- *Start with read-only assessments before attempting automated refactoring*
- *Collect detailed feedback on accuracy, usefulness, and time savings compared to manual analysis*

Success Metrics:

- *Assessment accuracy rates validated by senior architects*
- *Client willingness to expand scope based on initial results*
- *Time savings compared to traditional discovery phases*
- *Technical debt identification effectiveness*

Classification: *This is* ***divergent experiment*** *because we're exploring open-ended questions about platform capabilities, market fit, and client value perception rather than testing specific features against each other.*