



ENGINEERING & DEVELOPING PRODUCTS

TECHNICAL SOLUTION

SAMPLE SLIDE

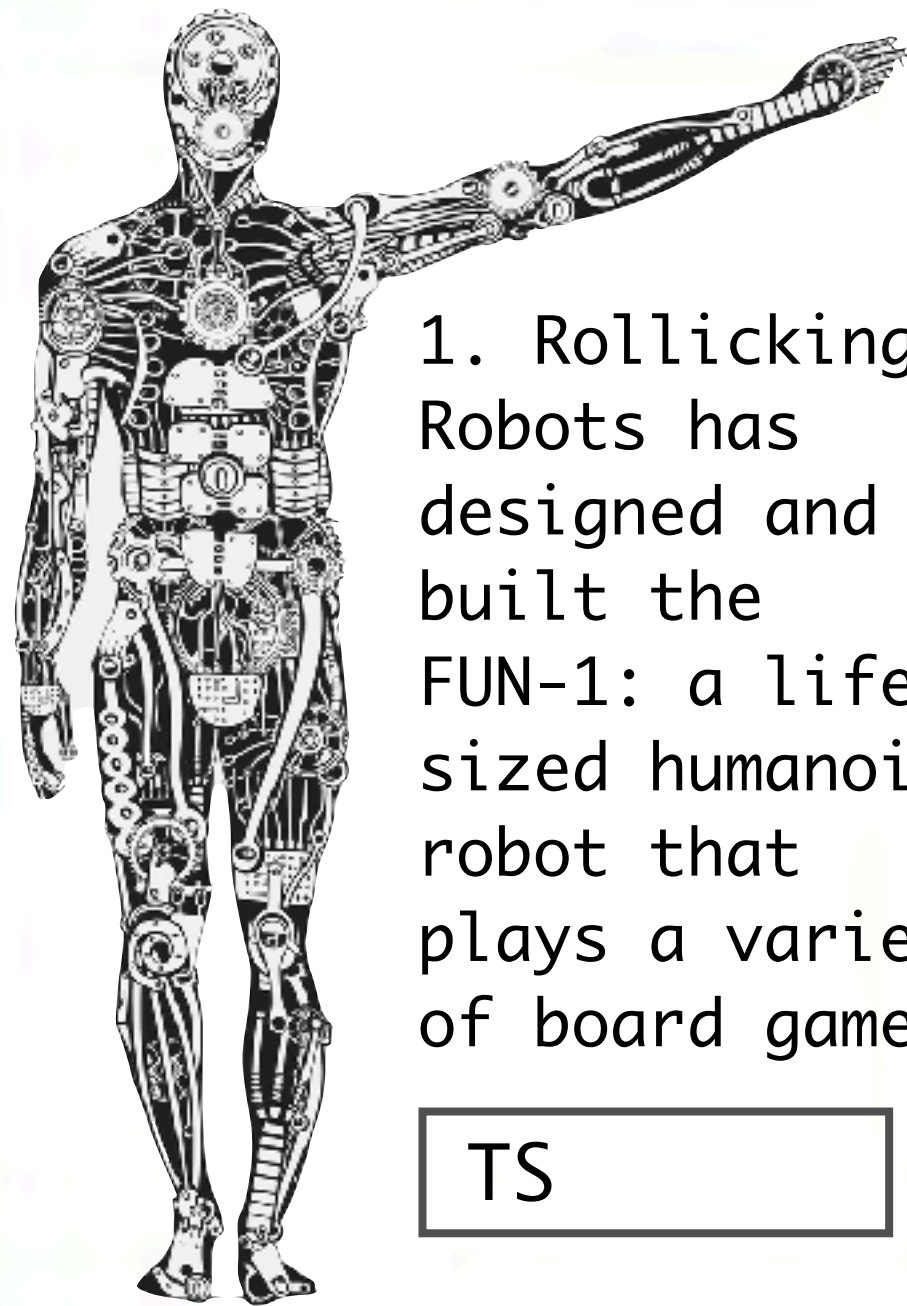


TECHNICAL SOLUTION

EXERCISE SCENARIO: FANTASY

ROLLICKING ROBOTS

Rollicking Robots opened in Tech City two years ago. Its mission: To create entertaining robots that bring joy to the inhabitants of Fantastica.



1. Rollicking Robots has designed and built the FUN-1: a life-sized humanoid robot that plays a variety of board games.

TS

2. Every FUN-1 is shipped with a User Manual providing helpful diagrams and written instructions in languages that include Futhark (Viking), Old English, and Tibetan.



TS

3. An early design review uncovered that the amount of memory targeted for the FUN-1 would likely result in an unacceptably high response time during game play. So, the memory size was quadrupled to increase predicted speed.



TS

4. A technical data package – including performance requirements, engineering drawings, and version control information – was provided to the FUN-1 implementation team.



TS

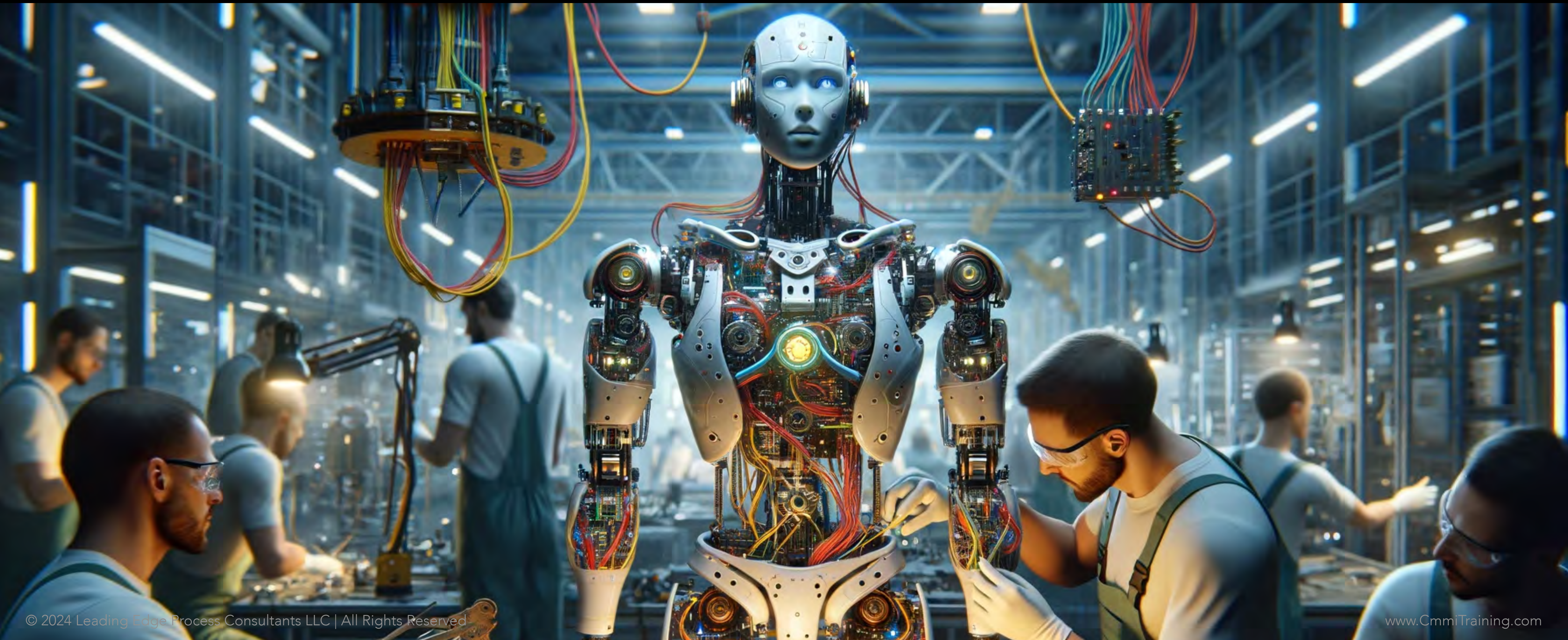


ENGINEERING & DEVELOPING PRODUCTS

PRODUCT INTEGRATION



SAMPLE SLIDE



EXERCISE: SAMPLE EXAM QUESTIONS

1. This requires automated unit and integration tests:

- a. Product Integration (PI)
- b. Scrum
- c. Configuration Management (CM)
- d. Continuous Integration (CI)

2. Component interfaces are:

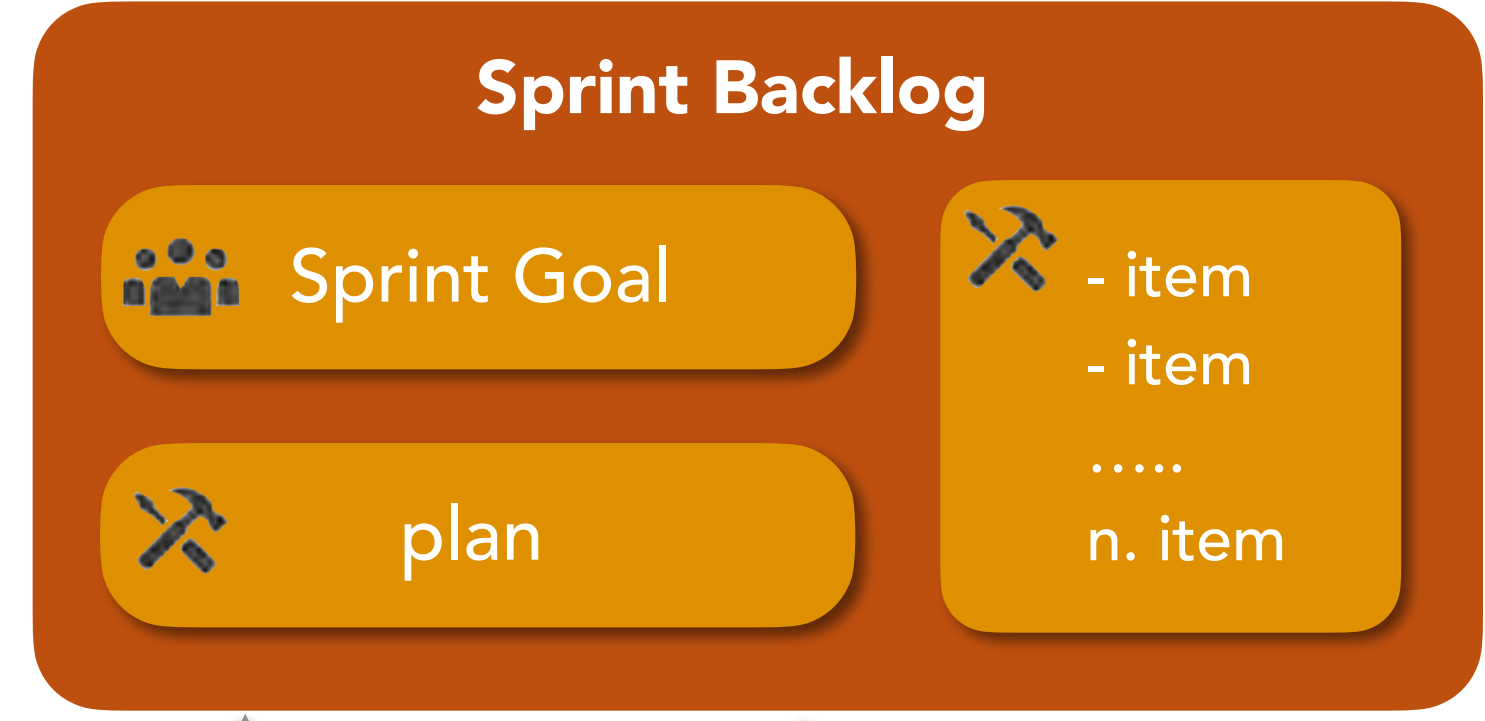
- a. Internal
- b. External
- c. Internal or external
- d. Just too complicated for me

ENGINEERING & DEVELOPING PRODUCTS

SCRUM IN A NUTSHELL

In a nutshell, Scrum requires a **Scrum Master** to foster an environment where:

1. A **Product Owner** orders the work for a complex problem into a **Product Backlog**.



3. The **Scrum Team** and its stakeholders inspect the results and adjust for the next **Sprint**.



Increment(s)

4. Repeat.

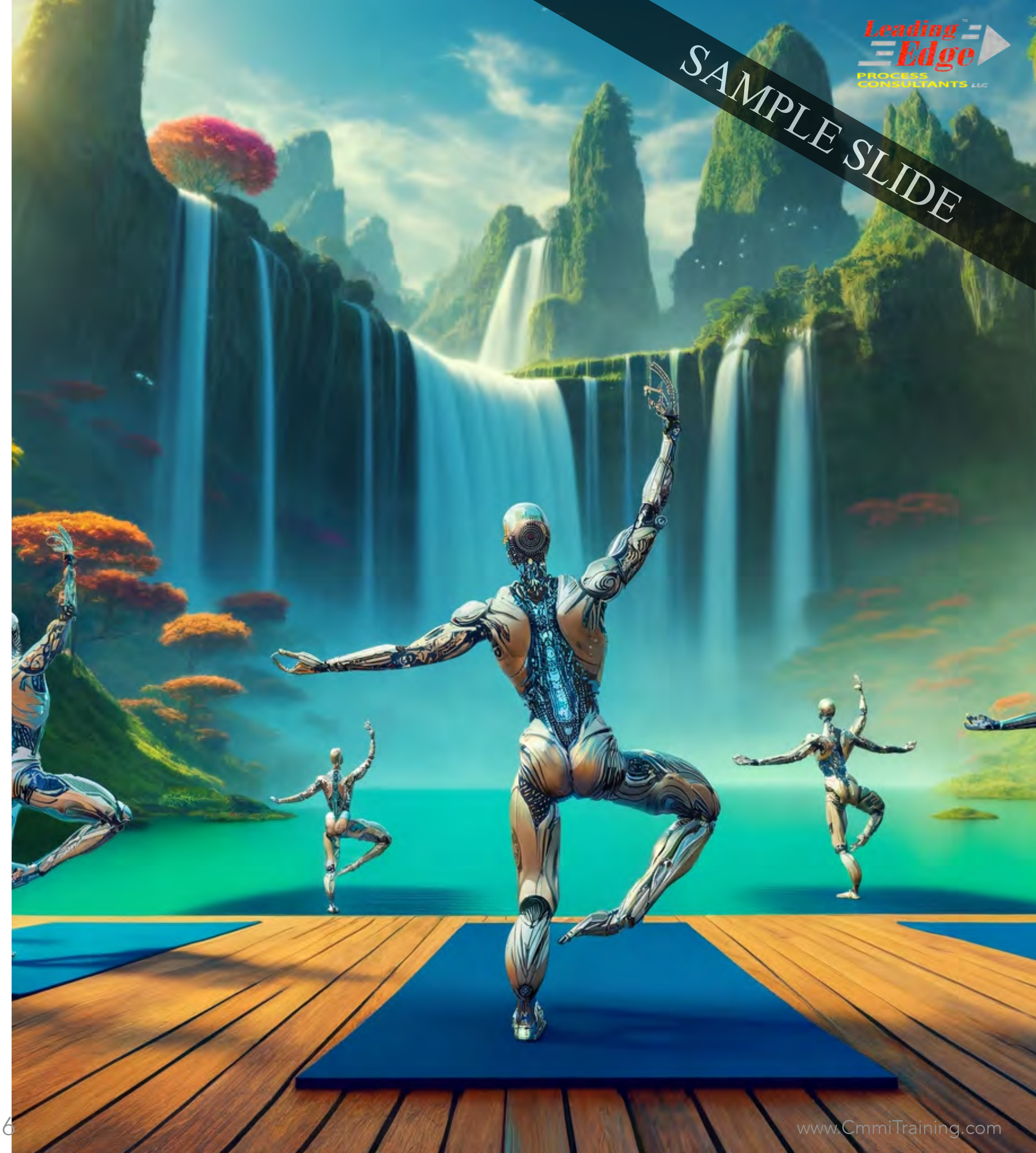
2. The **Scrum Team** turns a selection of the work into an **Increment** of value during a **Sprint**.

The four phases here are quoted verbatim from the Scrum Definition in *The 2020 Scrum Guide* by Ken Schwaber and Jeff Sutherland.

ENGINEERING & DEVELOPING PRODUCTS

EXERCISE: AGILE VS. WATERFALL

1. I'll divide you into teams.
2. As teams, discuss the following:
 - What are key similarities and differences between agile and waterfall lifecycle models?
 - Under what circumstances might you prefer agile methods? How about waterfall?
3. Document your results on the following slide.



EXERCISE: SAMPLE EXAM QUESTIONS

1. In an agile development environment, requirements:
 - a. Are written on-the-fly, by each developer
 - b. Must be written as "shall" statements to be fully compliant
 - c. May be expressed as a backlog of user stories or epics
 - d. Vanish when the cleaning people erase the whiteboard

2. A lifecycle model in which phases progress from requirements to preliminary design to detailed design and so forth is likely:
 - a. Agile
 - b. Waterfall
 - c. Scrum
 - d. Kanban