# Meeting transcript

Here is a sample meeting transcript between client team (Side A) and product software team (Side B) discussing requirements for a new software:

Client (Side A): We would like your team to propose a plan to improve productivity of tester and testing process with the help of AI. The scope of the project should focus on driving quick win and prove that AI can help testers improve quality of work and speed to delivery.

Project manager (Side B): Sure thing, here’s the project proposal we have prepared:

To enhance the clarity and effectiveness of your project proposal for TestVista, consider the following revisions and additions:

## Title: Project Proposal for TestVista: Enhancing Software Testing Efficiency through Automation

**Executive Summary:**

TestVista is a cutting-edge Language Model AI tool designed to revolutionize the productivity of software testers. By automating the **generation**, **prioritization**, and planning of **test cases** based on software requirement specifications (**SRS**), TestVista enables testers to focus on critical testing tasks, ensuring higher quality and efficiency in software development.

TestVista represents a transformative step forward in the field of software testing. By automating critical aspects of the test case generation and planning process, it not only improves the efficiency and accuracy of testing but also **empowers testers to dedicate more time to addressing complex testing challenges**. With TestVista, software companies can achieve faster deployment times and higher software quality, enhancing overall competitive edge.

### Business Values:

- **Increased Productivity**: Reduces manual test case generation and planning time.

- **Enhanced Accuracy**: Minimizes human errors by leveraging AI to suggest test cases based on detailed analysis of SRS and historical data.

- **Improved Test Coverage**: Ensures all critical test scenarios are covered through systematic test plan generation.

### Detailed Description of Use Cases:

1. **Data Input and Integration**:

- Inputs: Testers upload key documents: requirement docs (SRS), past test cases, and release notes.

- Integration: Compatibility with existing Fsoft testing systems like TestMan or AkAT.

2. **Automated Test Case Generation**:

- Functionality: Based on the acceptance criteria in the SRS and historical data, TestVista suggests relevant and prioritized test cases.

- Tester Interaction: Testers have the final say, with options to approve, reject, or edit the generated test cases.

3. **Release-Based Test Planning:**

- Trigger: Activated upon deployment.

- Outcome: Suggests a comprehensive test plan aligned with the latest release notes.

4. **Test Plan Export:**

- Formats: Allows testers to export finalized test cases into Excel or Docs formats.

- Utility: Facilitates easy sharing and documentation of test plans.

5. **Test Data Generation:**

- Alignment: Generates test data that is consistent with each test case, enhancing test accuracy and relevance.

### Front-End Requirements:

- Web Application: Ensures a user-friendly interface for interacting with TestVista.

- Data Upload Capability: Users must be able to easily upload relevant documents for processing.

- Interactive Feedback System: Users should be able to view, interact with, and modify the steps and outputs generated by TestVista.

### Non-Functional Requirements:

- Authentication: Integration with Fsoft's SAM/SSO for secure user authentication.

- Data Analytics: Collection and storage of user interaction data (clicks, edits, approvals/rejections, test cases generated) for performance analytics and system improvement.