Definitions of the Terms

User stories

A lightweight format to express the desired business value and to represent the conversation related to the task.

For example, "As a user, I want to log in to the system".

Definition of done

A checklist of the types of work that the team is expected to successfully complete for the task to be declared as 'done' (shippable). The definition of done is typically applied to multiple tasks in an iteration. For example, "unit test passed, code reviewed, live on the production server".

Acceptance criteria

The desired behavior that used to determine whether a task has been successfully developed and can be accepted by a user or customer. For example, "The function must work on all devices."

Acceptance test or Test cases

The tests to determine whether or not to accept a developing system based on the predefined acceptance criteria or requirements.

Functional requirements

The description of a function or a task to be developed, which enables users to accomplish their tasks when using the software.

Nonfunctional requirements

A description of behavior (requirement) that does not relate to functionality but a property or characteristic that a system should have been met, e.g., reliability, efficiency, usability, maintainability, or portability.

Task dependencies

A reliance that the task has on a factor, event, or the other task outside its scope of work.

Code examples

A chunk of example code (or a link to an example code) to assist the implementation.

UI wireframes (or pre-designed UI)

Graphical design, mockups, or wireframes of user interfaces (or a link to these materials) as a guide for software development.

API reference guides

A document (or a link to a document) that describes the interfaces required to invoke a module or connect to another system.