**User Stories:**

* A user story is the smallest unit of work in an agile framework.
* A user story is an informal, general explanation of a software feature written from the perspective of the end user or customer.
* They are short, simple descriptions of a feature or requirement told from the perspective of the end-user.
* Describes the user or person who will be interacting with the feature. This could be a specific user type or a broader category of users.
* user stories are a versatile tool that helps Agile teams effectively capture, communicate, and deliver value to users throughout the software development lifecycle.

how user stories apply to real-time applications:

* **Identifying User Needs**: User stories help teams understand the needs and expectations of end-users when interacting with real-time applications. For example, a user story for a **live chat** application might focus on enabling users to send and receive messages in real-time.
* **Communicating Requirements**: User stories provide a simple and concise way to communicate requirements between different stakeholders, including product owners, developers, testers, and designers. They serve as a common language that everyone involved in the project can understand.
* **Guiding Development**: User stories provide a clear understanding of what needs to be built and why. They serve as a **guide for development teams** during implementation, ensuring that the final product meets the intended user needs and expectations.
* **Project Initiation**: User stories can be used during project initiation to gather initial requirements and define the scope of the project.
* **Development**: User stories guide the development process by providing clear requirements and acceptance criteria for implementing features.

**EXAMPLE:**

**User Story Format: User Story Example:**

As a <user role>,(who) As a LinkedIn User

I want to <action>,(what) I want to search for jobs that are remote only,

So that <value>, (why) So that I can apply to jobs that allows me to

work from anywhere.

**Given/When/Then Format: Acceptance Criteria Example:**

**Given** - context or Pre-condition **Given** I am under the job tab in

**When** – some action is carried out LinkedIn,

**Then** – observable outcome & expected results **When** I search for jobs

**And** I filter by remote only

**Then** remote only job are

Displayed

**And** I can apply for them

* The user expresses a need for remote jobs on LinkedIn.
* LinkedIn filters job listings based on the user's preference for remote jobs.
* The user views the filtered job listings tailored to their preference for remote work opportunities.

**User Story**: As a teacher, I want to be able to upload and share educational materials with my students so that they can access resources outside of the classroom.

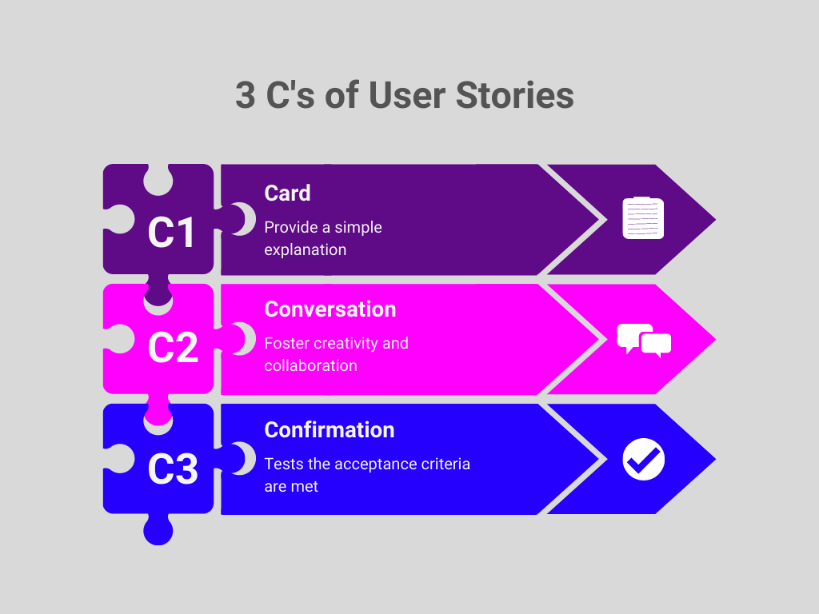
**Acceptance Criteria:**

* Given that I am logged into the teacher's portal,
* When I navigate to the "Materials" section,
* Then I should see an option to upload new educational materials.
* Given that I am uploading educational materials,
* When I select files from my computer and click on the "Upload" button,
* Then the files should be successfully uploaded to the platform

**User Story:** As a user of Zoomato, I want to be able to place a food order from nearby restaurants so that I can receive my food quickly.

**Acceptance Criteria:**

* Given that I am browsing nearby restaurants,
* When I select a restaurant from the list,
* Then the app should display the restaurant's distance from my location, ensuring that it is within a reasonable proximity for quick delivery or pickup.



**INVEST :**

The INVEST criteria is a set of characteristics used to evaluate user stories in Agile development.

Each letter stands for a different attribute:

**Independent:** The user story should be self-contained and not dependent on other stories. For example, a LinkedIn user story about updating profile information should not rely on another user story about adding new profile sections.

**Negotiable:** The details of the user story should be open to discussion and negotiation between the development team and stakeholders. For instance, the specifics of how profile information is displayed on LinkedIn can be negotiated based on user feedback.

**Valuable:** The user story should deliver value to the end user. For example, a LinkedIn user story about receiving personalized job recommendations based on profile information adds value by helping users discover relevant job opportunities.

**Estimable:** The user story should be clear and understandable enough for the development team to estimate the effort required to implement it. For instance, a LinkedIn user story about implementing a feature to endorse skills on profiles should have clear acceptance criteria for developers to estimate the work involved.

**Small:** The user story should be small enough to be completed within a single iteration or sprint. For example, a LinkedIn user story about improving the search functionality within job listings should be small enough to be implemented and tested within a sprint timeframe.

**Testable:** The user story should have clear acceptance criteria that define when the story is considered complete and working as expected. For example, a LinkedIn user story about enabling users to upload a resume when applying for jobs should have testable criteria related to successful resume uploads and error handling.

**EPICS:** an Epic is a large body of work that can be broken down into smaller, more manageable user stories.

* It is a piece of work that’s broken down into specific tasks based on the needs of customers or end-users.

