

Ingegneria del Software

User Story

Prof. Paolo Giorgini

A.A. 2024/2025

Che cos'è una User Story

Smallest piece of work to deliver value to the customer

- Title

As a <role> I want <objective> so that <motivation>

Es. Come passeggero di Uber, voglio vedere diversi autisti disponibili nella mia zona, così posso scegliere quello più vicino a me

- Descrizione

- Da del contest alla user story
- User journey, Use cases, Pictures, Design page, ...

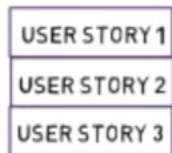
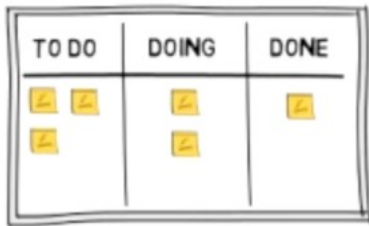
- Acceptance Criteria

- Condizioni per validare l'implementazione della user story

Es. Autisti entro un'area di 4 km, massimo 10 autisti, visualizzare il profilo di questi autisti

Quando si usano le user stories

SCRUM

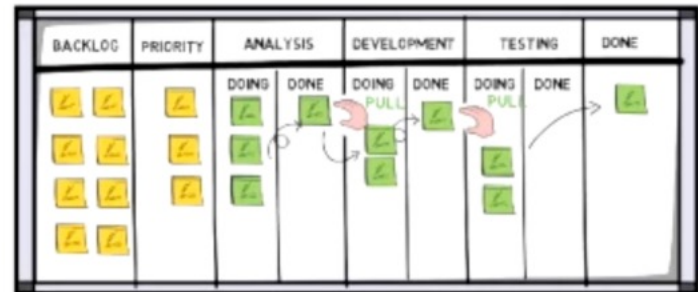


SPRINT BACKLOG



USED TO ESTIMATE, PRIORITIZE AND PLAN SPRINTS

KANBAN



USED TO MANAGE WORK IN PROGRESS LIMIT

Come scrivere una user story

INVEST

- **I**ndependent : can be developed in any order
- **N**egotiable : open the conversation
- **V**aluable : value added to the product
- **E**stimable : relative sizing
- **S**mall : to be completed fasted
- **T**estable : must be validated

Chi deve scriverle?

PRODUCT OWNER



USER STORY 1
USER STORY 2
USER STORY 3
USER STORY 4
USER STORY ...

ENSURE THE PRODUCT BACKLOG EXISTS

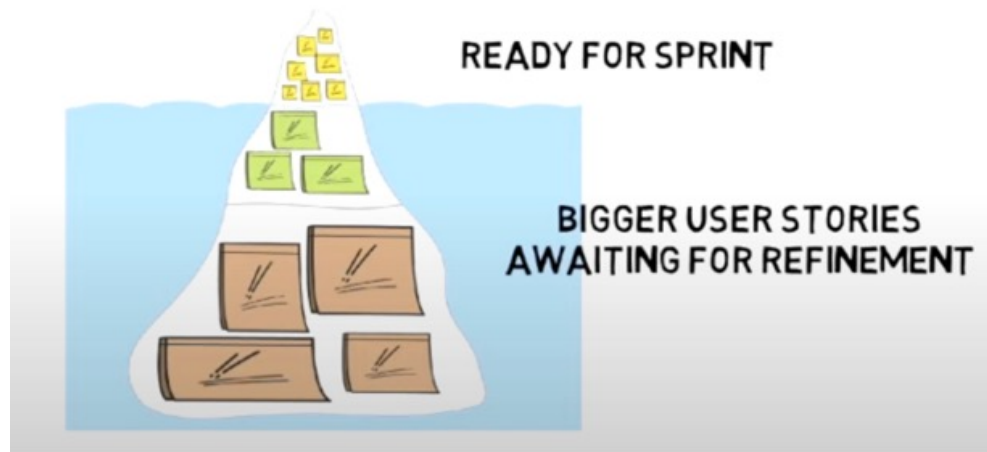
PRIORITIZE THE USER STORIES



**ANYONE CAN CONTRIBUTE
AND WRITE USER STORIES**

Quando scriverle

- Durante l'intero ciclo di sviluppo del prodotto
- Story writing workshop per creare il product backlog



- Nuove user stories possono essere create in qualsiasi momento

User stories (items in the product backlog)

Not a test
case!

Log in		
User story		Importance
<i>As a customer I want to log in to the internet bank so that I can do my banking stuff.</i>		1
how to demo		Estimate
<i>Given that I am at the start page for the internet bank, when I fill in my user name, password and press the Log in button, then I should be logged in to my account at the bank.</i>		8

User stories: examples (I) (functional)

- As a *driver*, I want to *block badly behaved passengers* so *they are never shown me again*.
- As a *passenger*, I want to *link the credit card to my profile* so that *I can pay for a ride faster, easier and without cash*.
- As a *driver*, I want to *add photos of my car in my profile* so that *I can attract more users*.
- As a *passenger*, I want *several available drivers to be displayed* so that *I can choose the most suitable option for me*.

User stories: examples (II)

(non functional)

- As a user, I want to have the application responds quickly to my actions so that I don't get bored
- As a user, I want to have nice error pages when something goes wrong so that I can trust the system
- As a user, I want to have an easy-to-use application so that I can easily use it even though I'm not an IT expert

User stories (I)

Avoid technicalities

- Too technical? Ask why!
 - **User story (v1)**: products are selected from a drop down list
 - Why?
 - To be sure they are selected from a controlled set of items
 - **User story (updated)**: products are selected from a set of predefined options

User stories (II)

Granularity and sub-stories

- Granularity
 - User story: Payment
 - less specific
 - User story: Payment via credit card
 - more specific
- A user story can be split into a set of more concrete/specific user stories
 - User story A \rightarrow User stories A1, A2, A3

User stories (III)

Testing a user story

- **User story:** "A user can pay products in his/her shopping cart by using a credit card".
- Test examples:
 - Test using cardA, cardB and cardC (pass)
 - Test using cardD (fail)
 - Test using a cardA debit card (pass)
 - Test using right/wrong card numbers, security code, name, expiration date (pass/fail)
 - Test using expired cards (fail)
- Such tests capture users' expectations about the final system to support some cards rather than others and provide some real scenarios useful for the devel team

User stories (IV)

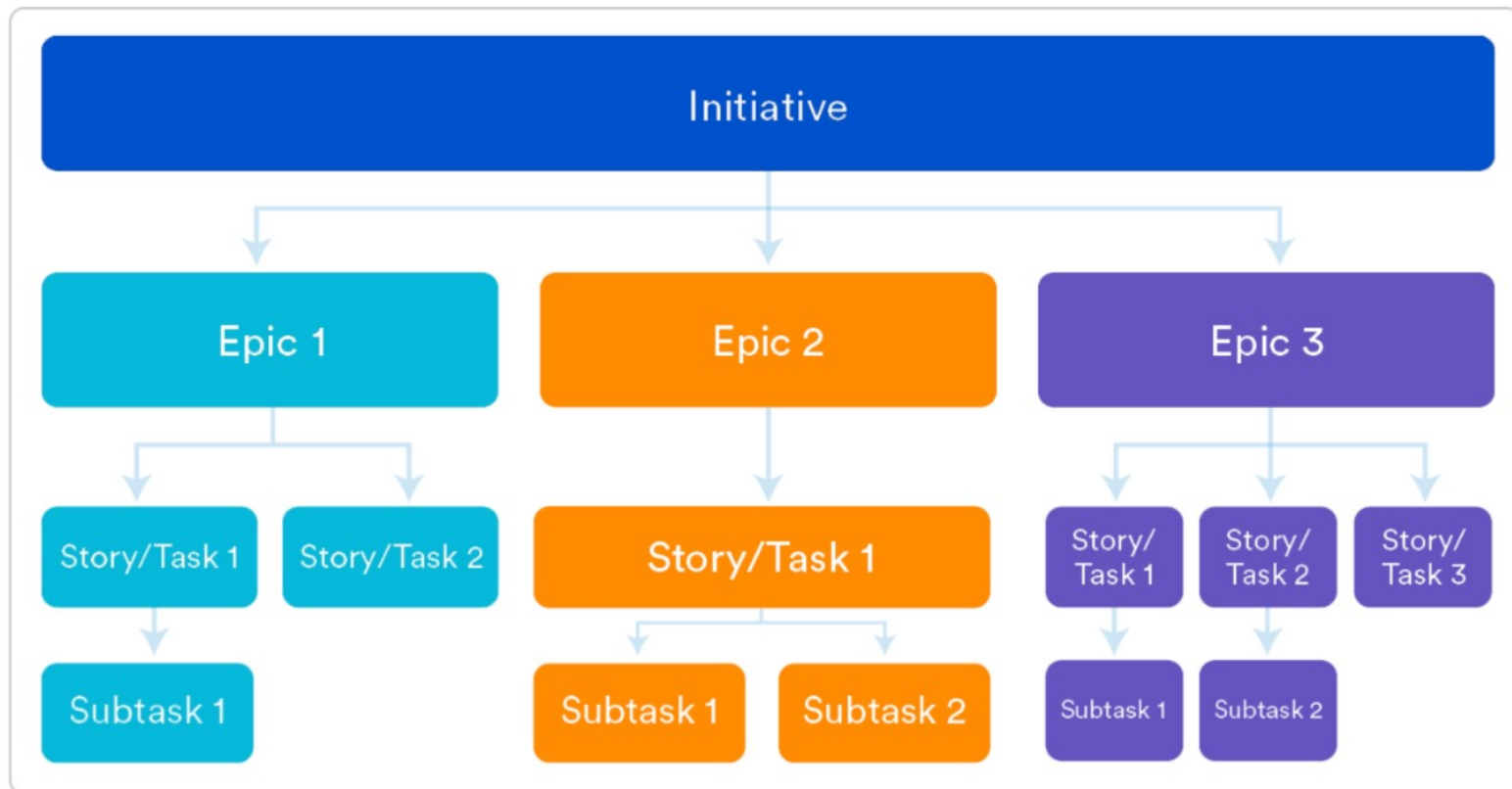
Size of a user story

- There is **no right size** of a User Story
 - 1-3 days to complete could be a good size
 - Several tasks
 - Having 1 big user story (half of a sprint) and the rest smaller can be risky
 - Avoid having a set of large user stories
 - Pay attention to stories like “what & what”. They are 2.
- As a general rule, smaller is better
 - Small enough to be completed soon
 - big enough to be deliverable in its own right

User stories (V)

Epics & User stories

- **Epic:** is a high-level body of work that bands together with a group of related Stories



A Story 📄

An Epic 📖

The smallest unit of work; can't be split

Can be broken down into more specific and smaller elements - User Stories

Fit a shippable product increment that should be delivered during 1 sprint

May be implemented during a few sprints

Represents some value that user will get after implementation

Indicates a more general task (for example, implementation of a whole user-flow)

Quite easy to estimate

Harder to estimate since the scope is flexible

Imagine that you're building a dating app. In this case, good Epic and User Story examples (but don't take them too seriously) will be:

An Epic: Managing profiles

A Story: As an app user, I want to add profile photos so that more people write to me about how awesome I am

A Story: As an admin, I want to delete/block photos from users' profiles so that they don't scare off other people with their nude pics (or violate community rules)

A Story: As an app user, I want to have a separate field where I can tell more about myself so that people fall in love with my personality and not with my penthouse in the center of New York

Estimating effort

Planning Poker

Product owner describes a user story

Team: does questions to understand the story until everything is clear;

Scrum Master: indicates when do the estimation;

Team: members provides an estimate

Team: extreme estimations can be motivated;

ALL: The process is repeated until consensus is reached

1. Customer reads story.



2. Team estimates.

This includes testing effort.



3. Team discusses.



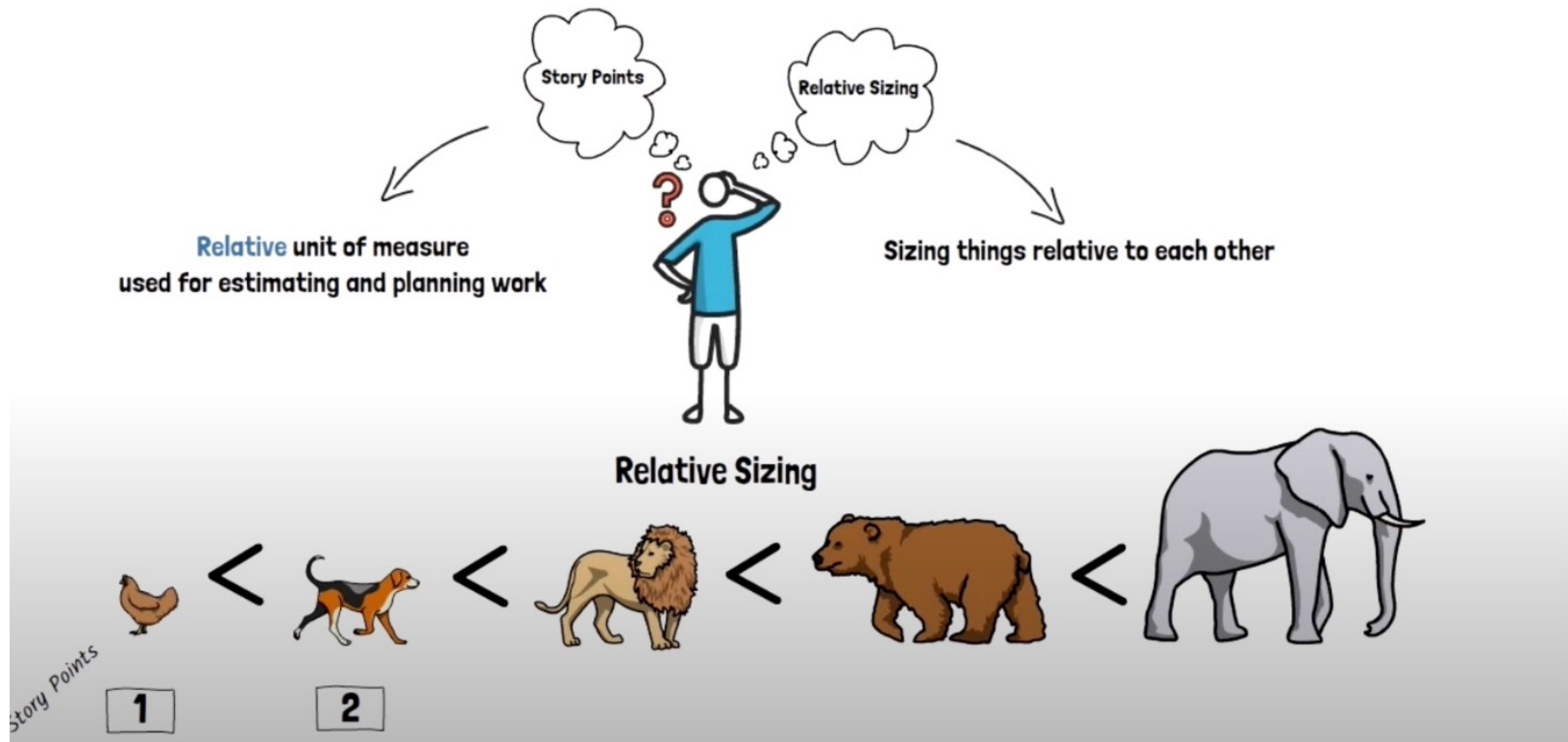
4. Team estimates again.

Repeat until consensus reached.



Estimating effort

What are story points in Agile?



RELATIVE ESTIMATION

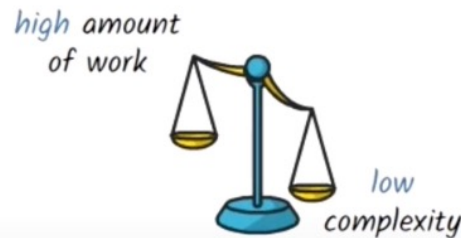
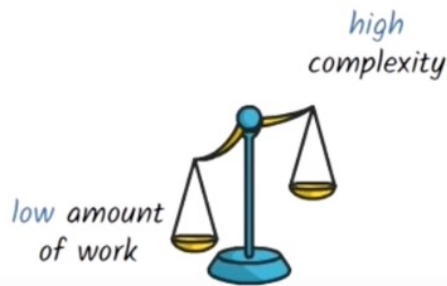


STORY POINT

<https://www.youtube.com/watch?v=LEPLaYcdgeg>

Story points are used for estimating the overall effort

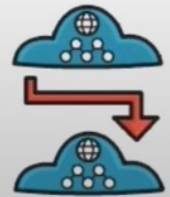
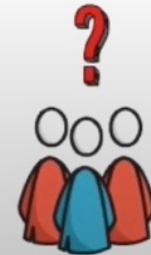
- The amount of work
 - Complexity
 - Risk and Uncertainty
- } One relative size measure



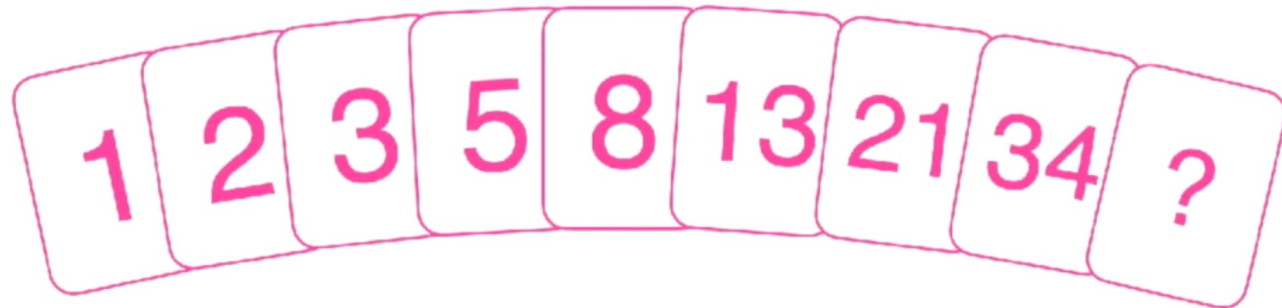
business algorithm



Risk and Uncertainty



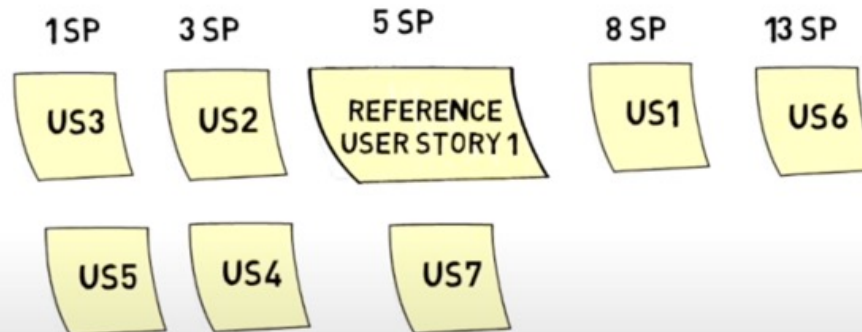
FIBONNACCI SEQUENCE



T-SHIRT SIZING



ESTIMATION WORKSHOP



EVERYONE FROM THE DEVELOPMENT TEAM SHOULD BE INVOLVED WHILE ESTIMATING USER STORIES

From Requirements to User stories (EasyLib)

RF1: Login. Il sistema deve consentire sia agli utenti biblioteca che agli operatori biblioteca di effettuare il login tramite due modalità: utilizzando le credenziali locali (nome utente e password) oppure tramite autenticazione con credenziali Google. In fase di accesso, l'utente o l'operatore deve poter scegliere tra l'inserimento di nome utente e password, già registrati nel sistema, o utilizzare il pulsante "Accedi con Google", che reindirizzerà alla schermata di autenticazione di Google. Una volta completato l'accesso con successo, il sistema reindirizzerà l'utente alla rispettiva dashboard in base al ruolo.

User Story 1: Scelta tra credenziali locali o autenticazione Google per utenti/operatori

- **Come** utente o operatore della biblioteca, **voglio** scegliere tra il login con credenziali locali o tramite Google **in modo** da poter selezionare il metodo di accesso che preferisco
- **Criteri di accettazione:**
 - Entrambi, utenti e operatori, possono scegliere tra l'inserimento di nome utente e password o il pulsante "Accedi con Google".
 - Dopo un login riuscito, il sistema reindirizza l'utente alla dashboard corrispondente al suo ruolo
 - Se il login fallisce (con credenziali locali o Google), il sistema fornisce messaggi di errore adeguati.

TASKS – User Story 1

- 1. Integrare entrambe le opzioni di login (locali e Google) nella stessa pagina**
 - Posizionare correttamente sia il form di login con credenziali locali che il pulsante di login con Google.
- 2. Configurare il flusso decisionale per il login**
 - Implementare la logica che distingue quale metodo di autenticazione (locale o Google) è stato scelto e avvia il processo corretto.
- 3. Gestire il routing dopo il login per entrambi i ruoli (utente e operatore)**
 - Assicurarsi che dopo l'autenticazione l'utente venga reindirizzato alla dashboard corretta in base al suo ruolo (utente o operatore).
- 4. Testare entrambe le modalità di accesso**
 - Testare l'autenticazione con credenziali locali per utenti e operatori.
 - Testare l'autenticazione Google per utenti e operatori.
 - Verificare che entrambe le opzioni funzionino correttamente e conducano alla dashboard giusta.
- 5. Implementare messaggi d'errore distinti per ciascun metodo**
 - Creare messaggi di errore specifici per problemi relativi al login con credenziali locali o tramite Google.

From Requirements to User stories (EasyLib)

RF3: Ricerca. Il sistema deve consentire alle due tipologie di utente di cercare i libri presenti nel catalogo bibliografico del comune di Trento utilizzando diversi criteri di ricerca. Gli utenti devono poter inserire uno o più parametri tra titolo, autore, casa editrice e/o ISBN in un campo di ricerca. Il sistema restituirà una lista di risultati che corrispondono ai criteri inseriti, mostrando dettagli come titolo del libro, autore, anno di pubblicazione, disponibilità attuale e posizione in biblioteca. La ricerca deve permettere la combinazione di più filtri per affinare i risultati.

- **User Story 1:** Ricerca di libri per titolo, autore, casa editrice o ISBN
- **Come** utente della biblioteca, **voglio** poter cercare i libri presenti nel catalogo utilizzando uno o più criteri di ricerca (titolo, autore, casa editrice o ISBN), **in modo** da trovare facilmente i libri che mi interessano
- **Criteri di accettazione:**
 - L'utente può inserire uno o più criteri di ricerca nel campo apposito (titolo, autore, casa editrice o ISBN).
 - Il sistema restituisce una lista di risultati che corrispondono ai criteri inseriti.
 - Ogni risultato deve mostrare dettagli come titolo, autore, anno di pubblicazione, disponibilità e posizione in biblioteca.
 - Se non vengono trovati risultati, viene mostrato un messaggio informativo.

TAKS:

1. Creare il campo di ricerca nella UI

- Progettare e sviluppare il campo di ricerca dove gli utenti possono inserire uno o più criteri (titolo, autore, casa editrice, ISBN).
- Aggiungere placeholder e istruzioni per aiutare gli utenti a capire come utilizzare il campo.

2. Implementare la logica di ricerca per singoli criteri

- Scrivere la logica backend per gestire la ricerca per ciascun criterio (titolo, autore, casa editrice, ISBN) separatamente.
- Collegare la ricerca con il database del catalogo bibliografico.

3. Recuperare e visualizzare i risultati della ricerca

- Implementare il recupero dei risultati dal database in base ai criteri inseriti dall'utente.
- Mostrare i dettagli per ogni risultato: titolo del libro, autore, anno di pubblicazione, disponibilità e posizione in biblioteca.

4. Gestire i casi senza risultati

- Implementare la logica per mostrare un messaggio informativo quando non vengono trovati risultati in base ai criteri di ricerca.

5. Testare la funzionalità di ricerca per singolo criterio

- Testare la ricerca per ciascun criterio (titolo, autore, casa editrice, ISBN) per assicurarsi che funzioni correttamente.
- Verificare che la visualizzazione dei risultati sia corretta e coerente con i dettagli richiesti.

User Story 2: Ricerca avanzata con combinazione di filtri

- **Come** utente della biblioteca, **voglio** poter combinare più filtri di ricerca (titolo, autore, casa editrice e ISBN), **in modo da** affinare i risultati e trovare in modo più preciso il libro che sto cercando
- **Criteri di accettazione:**
 - L'utente può inserire più parametri contemporaneamente (ad esempio, titolo e autore) per restringere i risultati della ricerca.
 - Il sistema restituisce una lista filtrata di risultati basati su tutti i parametri inseriti.
 - I dettagli del risultato devono includere titolo, autore, anno di pubblicazione, disponibilità e posizione in biblioteca.
 - La ricerca deve funzionare anche se alcuni parametri non sono inseriti (ad esempio, solo titolo e autore, senza ISBN).

TASKS

1. **Modificare il campo di ricerca per supportare la combinazione di criteri**

- Estendere la funzionalità del campo di ricerca per consentire l'inserimento di più parametri contemporaneamente (ad esempio, titolo e autore).
- Aggiungere un'interfaccia che permetta agli utenti di combinare filtri in modo semplice e intuitivo.

2. **Implementare la logica di ricerca con più criteri combinati**

- Sviluppare la logica backend che permetta di combinare più criteri di ricerca (titolo, autore, casa editrice, ISBN) per restringere i risultati.
- Assicurarsi che i criteri possano essere utilizzati anche se parzialmente inseriti (ad esempio, solo titolo e autore).

3. **Recuperare e visualizzare i risultati della ricerca combinate**

- Implementare la logica per filtrare i risultati in base a più criteri.
- Mostrare i risultati correttamente con tutti i dettagli richiesti (titolo, autore, anno di pubblicazione, disponibilità e posizione).

4. **Testare la combinazione di criteri per la ricerca avanzata**

- Testare la ricerca combinata con più criteri per garantire che i risultati siano corretti.
- Verificare che la ricerca funzioni correttamente con combinazioni diverse di filtri (ad esempio, titolo + autore, autore + casa editrice, ecc.).

5. **Gestire i casi senza risultati per la ricerca combinate**

- Implementare un messaggio di errore o informativo quando non ci sono risultati basati sui filtri combinati.
- Testare la corretta gestione dei casi senza risultati.

TO DO

- Selezionare i requisiti che ritenete più importanti da sviluppare
 - Anche in ottica di demo per il Comune
- Creare le user stories per i requisiti selezionati
 - Assegnate un valore di importanza
 - Identificate i tasks e stimate gli story points
- Ragionate anche in termini di epiche