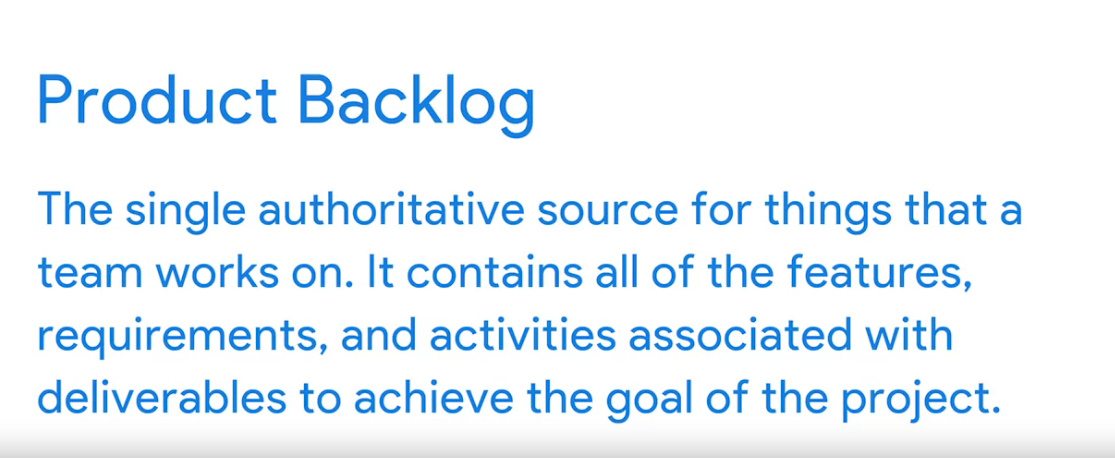
**Building A Product Backlog**



In Product Backlog este scris tot ce trebuie sa facem, ce sa contina, si fiecare item are multa informatie.

* **product backlog** are 3 caracteristici:

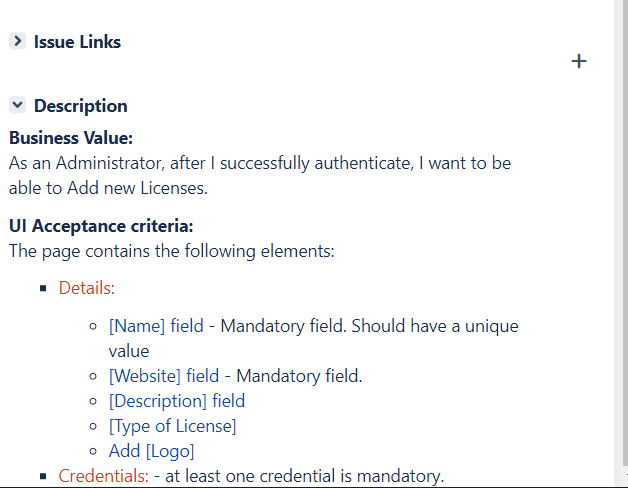
1**. Is a Living artifact** – iteme sunt adaugate in backlog oricand, si pe parcursul proiectului.

**2. Is Owned and adjusted by PO** -

**3. Is a prioritized list of featurs** –in backlog totul e pus in ordinea importantei.

* Important pieces of data and practicese:

1. **The item description** –

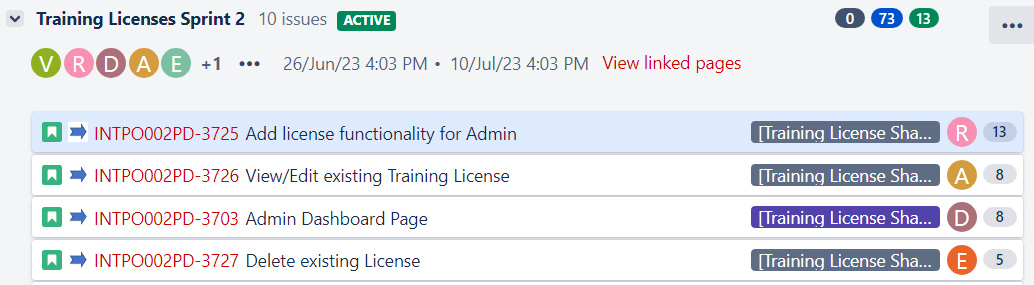


1. **The value** – spune cat de mult bussiness value aduce itemul la client, team etc. Cum sa il reprezinte, e alegerea la team

**3. The order** – ordinea depinde de value si estimate, dar nu numai. Ordinea arata cat de important e itemul.

**4. The estimate** –

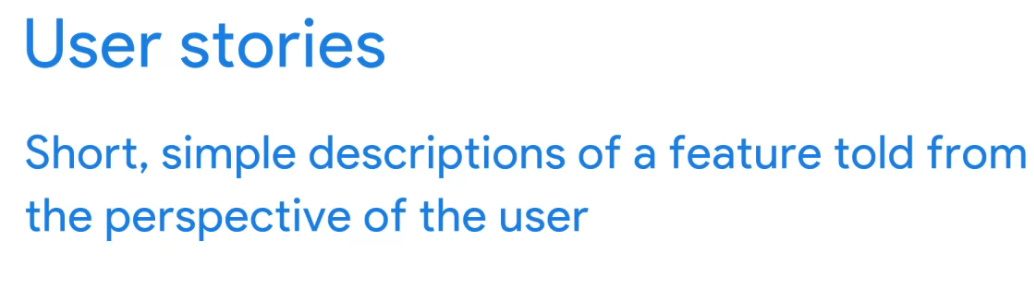
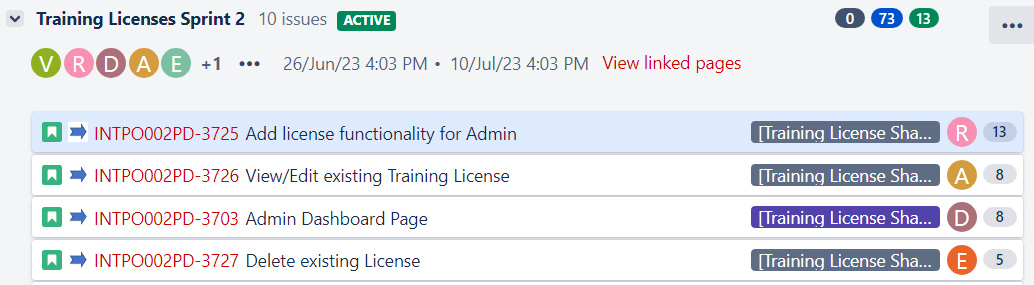
****





* **Product Goal** – obiectiv de lunga durata pe care Scrum team vrea sa il atinga.

**User Story**

* 

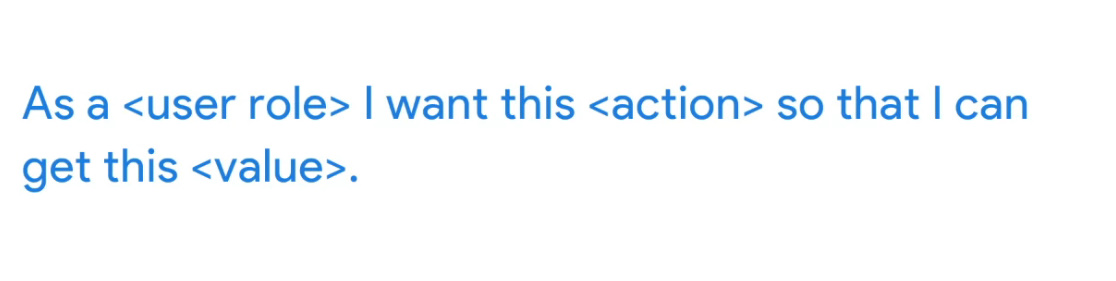


* User Story e creat din 3 elemente:

- **User** – aici scriem cine anume e user, ca User, Admin, Manager etc.

- **Action** -

- **Benefit** – este value sau rezultatul la action



* Fiecare User Story trebuie sa respecte 6 reguli: INVEST

**I**ndependent – story trebuie inceputa si terminata singura, adica nu continua si nu se termina in alte user story si deci nu depinde de ele

**N**egotiable – avem locul unde se discuta, se pun intrebari etc.

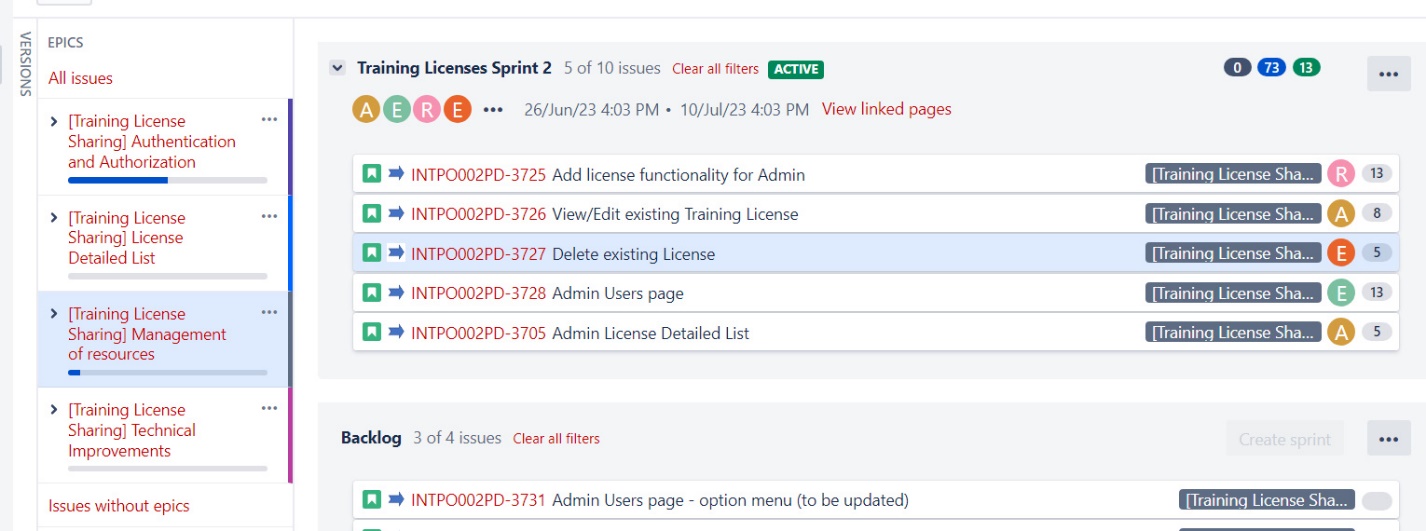
**V**aluable – completarea la un user story trebuie sa aduca o valoare

**E**stimable -

**S**mall – sunt relativ mici

**T**estable – poate fi testat

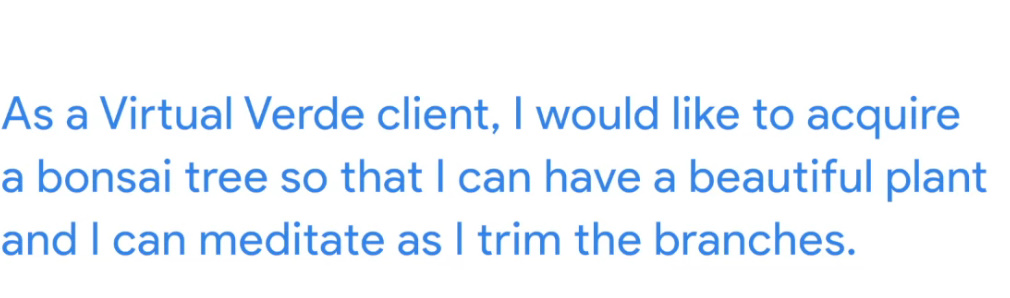
* **Epic** – un grup sau colectie de user stories ce au ceva in comun, de ex se refera la Management, la WebSite Creation etc.



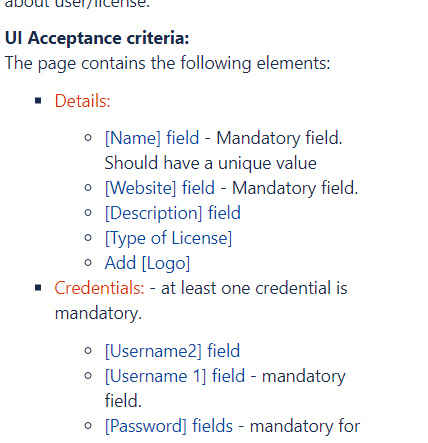


Epic e ca un foarte mare user story, ce nu poate fi livrat intr-o singura iteratie. Epic nu are doar user stories la un sprint, ci la toate.

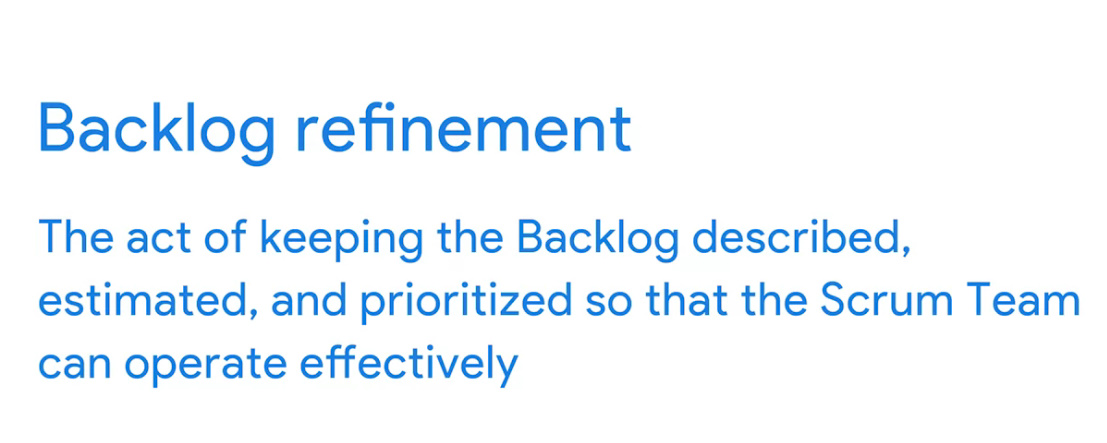
Ex de user story:



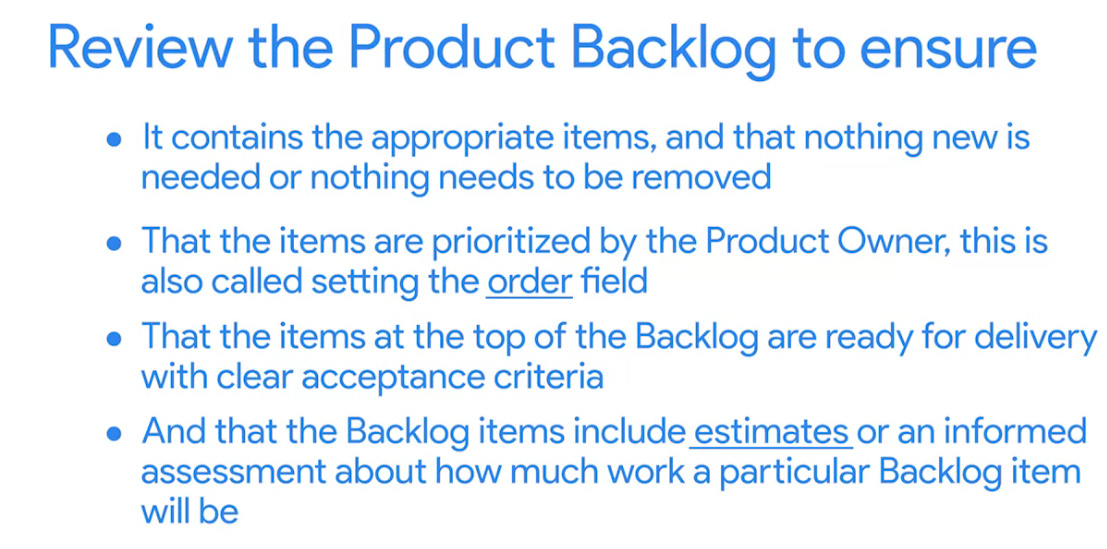
* **Acceptance Criterias** – in user story, PO creaza acceptance criterias, care este o lista de cerinte pe care o folosim ca sa decidem daca user story e gata sau nu

****

**Backlog refinement**

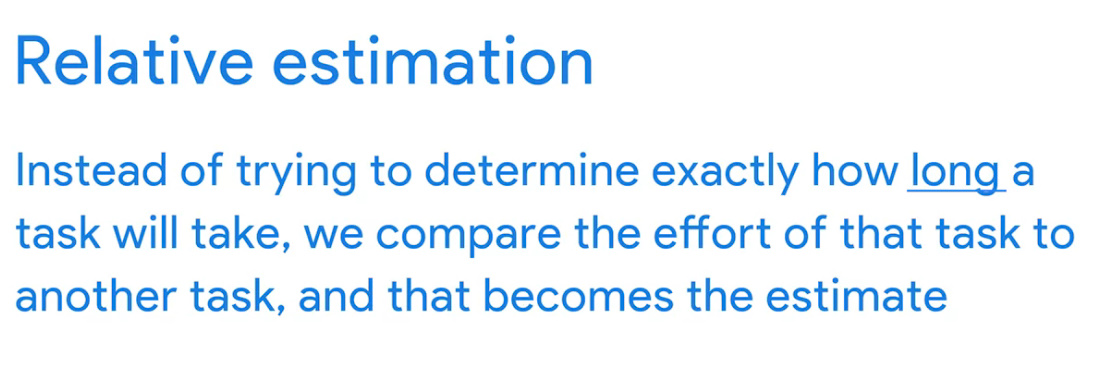


Backlog refinement e cand PO si Scrum team revad backlog pentru a se asigura ca respecta urmatoarele conditii:



**Estimate**

* Scrum se bazeaza pe Relative Estimation



* Unitati de masura la estimation:

- Story points

- T-shirt sizes

* **T-shirt sizes:**

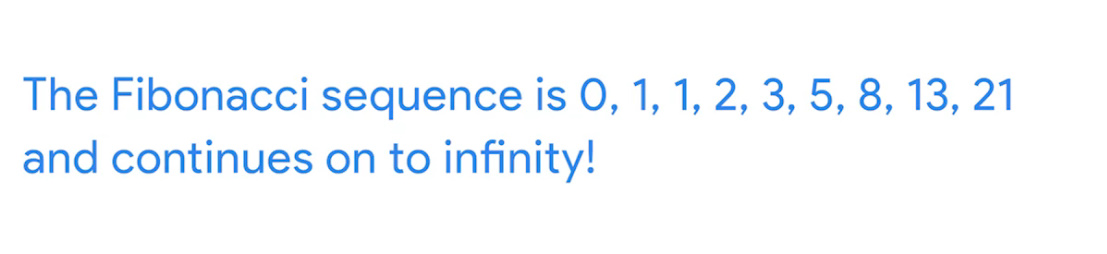
Luam un user story care cam pare ca e prin mijloc, si pe baza lui se fac comparatiile.

Folosim marimile la maici, XS,S,L,M,XL...

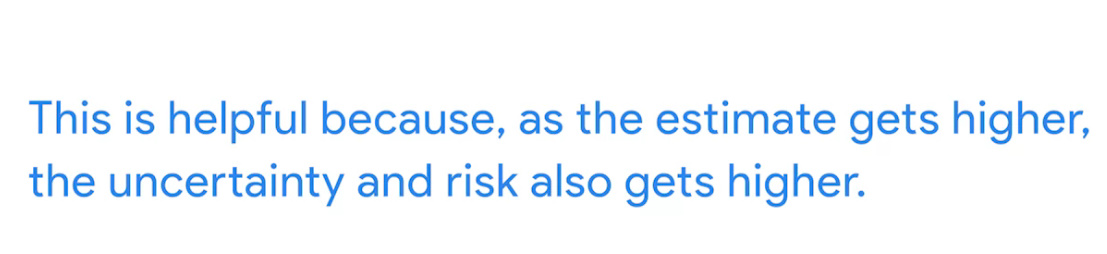
* **Story Points**

Luam un user story care cam pare ca e prin mijloc, si pe baza lui se fac comparatiile. Ele se folosesc in **Planning Poker**

Folosim Fibonnaci numbers:



0 si 1 sunt eliminate!



Ar fi inutil sa ne luptam gen intre 20 sau 21, dar intre 21 si 34 e diferenta chiar, de aia se iau numere ca difera intre ele destul de mult.

**Agile estimation technics**

* **Planning Poker**

Se folosesc carti ce au desenate numerele lui Fibbonaci. Fiecare alege o carte si la final toti le arata. Toti trebuie pana la urma sa fie de acord cu un numar. team discuta estimarile alese, mai ales cand sunt diferente mari de estimari.

* **Dot Voting**

Se fac niste stickere in forma de punct de diferite culori(semnficiatia se stabileste, daca e necesar). Fiecare membru pune la user story punctul colorat ales.

* **Bucket System**

E bun cand sunt foarte multe user stories. Se folosesc carduri, fiecare cu un numar pe el, reprezentand complexitatea.Cardurile sunt puse pe masa, intr-o anumita ordine. Fiecare persoana citeste un user story si apoi il pune pe masa langa cardul ce crede ca corespunde complexitatii

* **Large/Uncertain/Small**

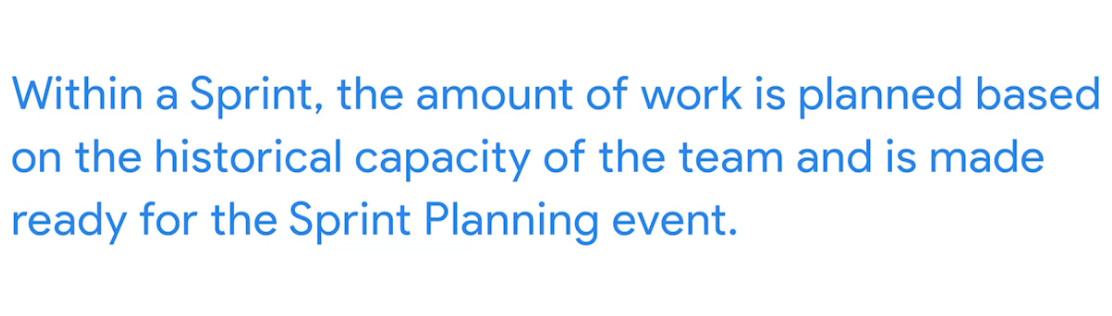
E ca Bucker System, doar ca se folossc carduri ce au pe ele desenat Large, Uncertain si Small

* Ordering method

O scala e preparata si itemele se pun random pe scala aleasa. Apoi, fiecare mmebru misca itemul pe scala unde crede ca ar trebui sa stea, si asa pana toti nu mai vor sa miste nimic.

**Sprint**

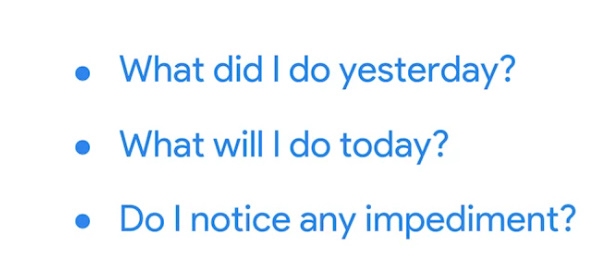
* Sprint mai este numit si Iteration



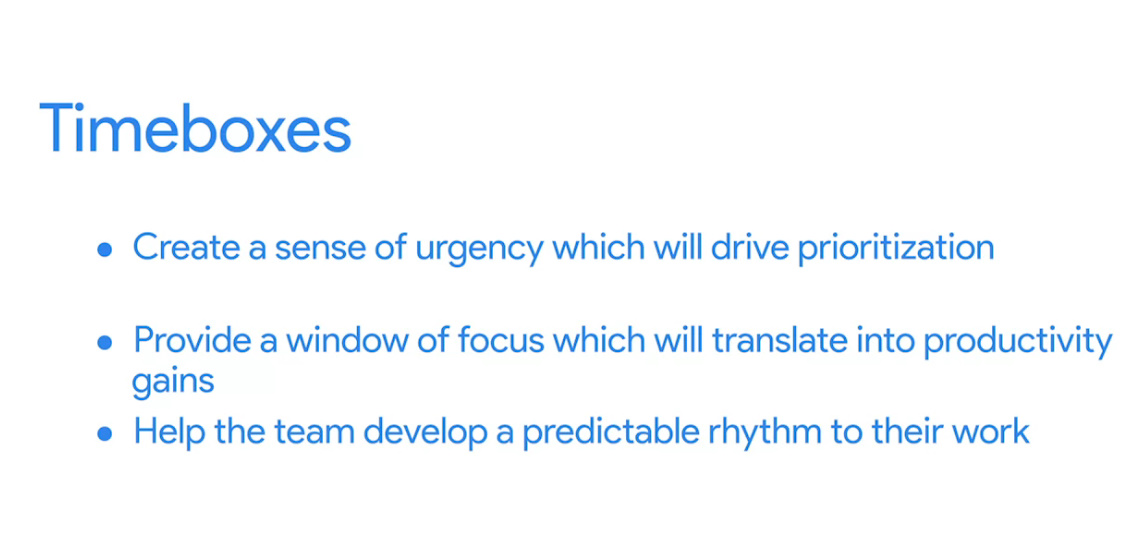
Sprint este ca un subproject.

* Scrum Guide contin urmatoarele evenimente:

1. **Sprint**
2. **Sprint Planning** – selectam ce iteme din Backlog vor fi alese pentru urmatorul sprint si care este spring goal. Alegerea se face de intreaga echipa in dependenta de cate persoane avem, ce capabilitate are team. Aici stabilim sprint backlog. E important sa luam in considerare cati oameni pleaca in vacanta, cati sunt disponibili etc. si vedem cam cate story points am putut face in sprinturile precedente
3. **Daily Scrum(standup)** – discutam despre taskurile de azi, ce probleme sunt, ce nu merge bine si ce merge bine si fiecare raspunde la intrebarile:

****

1. **Sprint Review** – e la final de sprint si demonstram lucrul facut in sprint precedent si vedem ce poate fi marcat ca facut in sprint dat si ce nu. E pasul important pentru Inspection si Adaptation. Facem o prezentare la lucrul facut ca sa vada toti, un demo.
2. **Sprint Retrospective** – tot e la final de sprint si analizam lucrul facut si analizam ce a mers bine si ce ar putea fi fost imbunatatit.

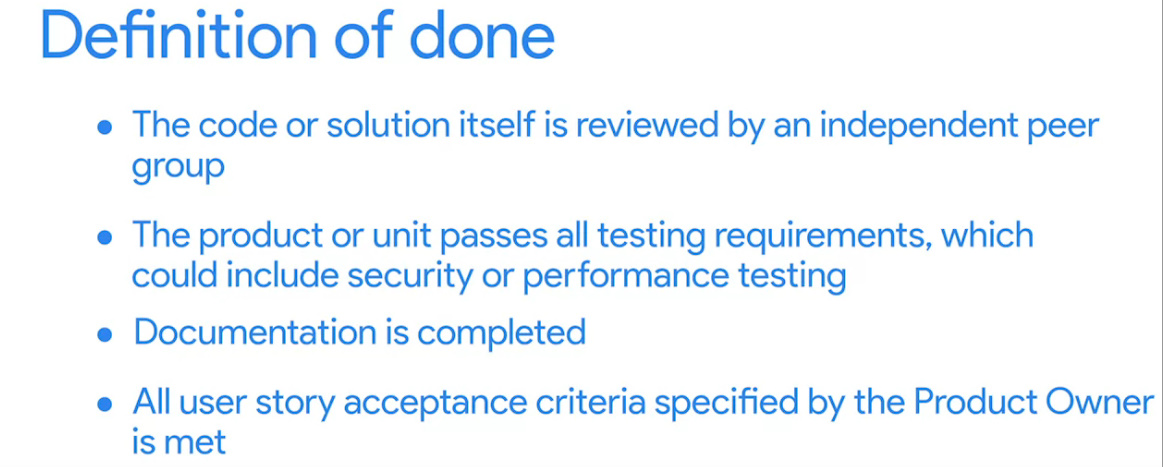
* TimeBoxes – o limita de timp stabilita pentru o anumita activitate

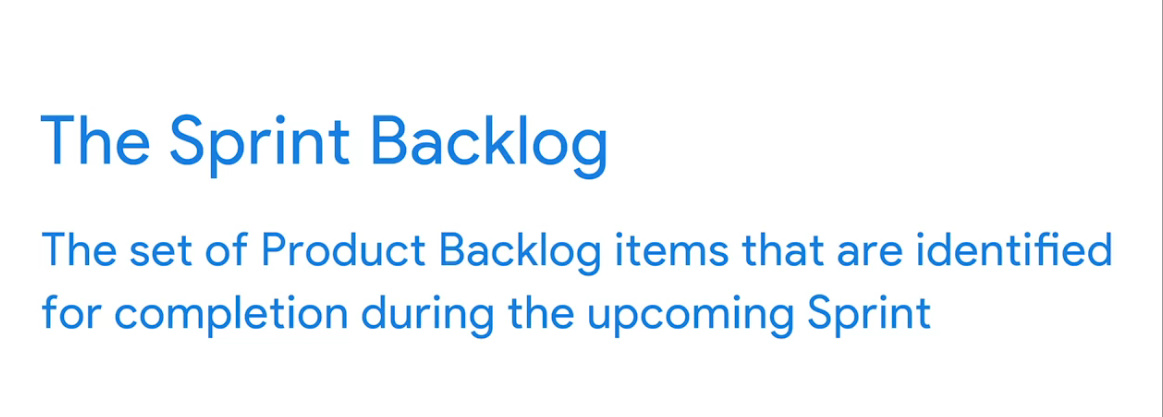
TimeBox este intre 1 – 4 saptamani

* Cum alegem timpul:

1. Daca proiectul va avea schimbari des, ar fi bine sa alegem o perioada cat mai scurta, ca 1-2 saptamani
2. Analizam cat de mult timp le trebuie la Developers pentru a organiza si crea solutia
3. Vedem cat de multe operatii sunt necesare pentru procesul de livrare, ca teste. control al calitatii etc.

* Ce inseamna ca un task e done:



* 

Spring Backlog contine un set de iteme din project backlog ce urmeaza a fi facute in sprint dat

* **Product Increment**: Ceea ce e creat dupa un Sprint
* **Release** – un produs ce contine minimul necesar de requirments indeplinite
* **Minimum viable product(MVP)** – versiune a produsului care contine destule chestii pentru a multumit clientul mai devreme. Nu e inca produsul final, dar deja contine chestii ce le putem arata la client ca sa vada cat de cat cum e aplicatia.