(2)

- debe incluir los criterios de aceptación.
- Pruebas de aceptación - hay positivas (pasan) y negativas (fallan)
  - acuerdo, si funciona lo acordado se acepta la user.
- Definition of Ready DoR
- definición de listo - control de calidad
  - cuando está lista para ser incluida en el sprint backlog.
  - aplicado a la user. Apto para ser integrado en una iteración — Do Ready
  - definido por el equipo
- ~~Definition of Done~~ → una buena user stories
  - determina si la user está terminada. completa y correcta. El PO define si pasa o producción.
  - Definition of Done
  - INVEST (INVEST)
    - Independent - puede implementarse en cualquier orden
    - negotiable - historia definida en "qué" no el "cómo". conversación.
    - valuable - debe tener valor para el negocio (no al desarrollo, tecnologías)
    - estimable - cuando puede asignarle un número (costos) Tamaño, complejidad y esfuerzo.
    - small - consumible en una iteración, comienza y termina en la iteración (sprint) → tamaño relativo al equipo y el tiempo.
    - testable - demostrar que fueron implementadas
- Algo más
  - NO ERS
    - no son especificaciones detalladas de requerimientos
  - niveles de granularidad (abstracción) — mastering agile software - Mike cohn
    - epic - user story grande, hay descomponerla en user más pequeñas. (dejamos para después, prioridad baja).
    - theme - colección de user stories relacionadas
    - user story: descripción de una funcionalidad desde la perspectiva del cliente o usuario

Clase 24/08

Requerimientos ágiles

- no establecidos desde un comienzo
- FOCO: valor del negocio. Software como medio de entrega de valor
- partimos desde una "visión" de producto → define la primera versión del producto.
  - product backlog: nunca está completo
    - necesitamos una colección de historias de comienzo
    - apunta a recibir cambios hasta en etapas finales.
    - no podemos tener una visión final del producto al comienzo. No definimos de antemano
    - pila o cola PRIORIZADA → fn. del product owner (que forma parte del negocio)
  - JUST IN TIME: diferir decisiones
    - detallar solo lo que hace falta en el momento

- comunicación cara a cara
- mejores requerimientos
- equipos auto organizados

User stories - usada en [XP / Scrum (adoptada, no nativa)]

- gestión binaria = está terminada o no: No hay porcentaje de avance completitud → TABLEROS (visible para todo el equipo)
  - visualización del trabajo
  - 3 columnas
    - to do
    - doing
    - done
- vs gestión tradicional - porcentaje (gantt) de avance: realmente no se puede medir
- por esto hacemos user stories, para poder entregar valor rápidamente. → SPRINT BACKLOG

- el producto tiene calidad cuando cumple con los RQ.

- definition of done - define cuando historia se puede mostrar al product owner.
  - definido por el equipo de desarrollo.
- expectativas y necesidades.

o las user stories épicas → las ponemos al fondo para recordar

spike – nivel de incertidumbre en el que no podemos estimar una user story (siempre la hay, pero esto es cuando excede el límite)
- desinformación
- técnica – como, tecnologías. necesidad de comprensión.
- funcionalidad – del lado del negocio del product owner. (incertidumbre de como el usuario interactua con el sist.) → prototipos.
  - no mezclar la funcionalidad con la investigación de la spike
  - cuando la investigación termina, queda la user para estar en el sprint backlog. (termina la spike).

no es recomendable dejar us y spike en la misma iteración

cuando surgen errores o defectos entran en el product backlog (con una prioridad dependiendo del impacto)

no tiene formato de user necesariamente. Puede afectar a una° más user stories.

cuántas user podemos realizar en un sprint. (trabajo, recursos y tiempo)

Estimaciones ágiles – tiene asociado una probabilidad, hay incertidumbre. Tenemos que hacer una predicción.
- no funcionan bien en el software (aunque pasan en muchos ámbitos). No son precisas.
- ① son relativas, no absolutas. somos mejores comparando que determinando un valor.
  - Estimaciones por comparación.
  - necesitamos algo con lo que comparar.
- ② se hace foco en la certeza y no en la precisión (es cara). "dentro de dos meses" → más estimación no da más certeza.
  - generalmente no se cumplen cuando hay precisión excesiva
- ③ diferir las decisiones hasta el último tiempo determinado
  - determinamos cuando es necesario (no todo)
  - al principio determinamos tamaño
  - luego, en el sprint planning, se reune el equipo y estimamos las user priorizadas el PO
    - estimación de historias
    - " de implementación
    - Poker planning
- ④ Estima el equipo, el que hace el trabajo. NO QUIEN DIRIGE
  - equipos autogestionados.

velocity: suma del nº de story points que el equipo completa en una iteración. → TERMINADAS, NO EN PROCESO

$$\text{duración} = \frac{\Sigma \text{ story points}}{\text{velocidad}}$$

unidad de estimación de user stories → story point (punto de historia)
- valor cuantificado: representa el tamaño de la user history.
- usa la serie de Fibonacci. Tiene crecimiento exponencial como el software.
  - 0, 1, 1, 2, 3, 5, 8, 13, 21, ...
- reconocimiento del equipo, se estima en forma colectiva.
- user story canónica: elemento para realizar la comparación. la primera se caracteriza y luego se comparan todas con esta.
  - comparado con el definition of done, no solo el tiempo de programación.
- Poker planning (comparando con la canónica.)
  - se estima cada uno sin mostrar su carta (para que no haya influencias)
  - se hace visible las estimaciones, explica cada uno, comenzando por los extremos (más alto y más bajo)
  - van comparando hasta llegar a una estimación. (consenso) si hay muchas diferencias comenzamos de nuevo.
- como elegimos la canónica
  - la más baja (1 punto) – con menos complicaciones → la más usada
    - pueden ocurrir cosas más fáciles
  - de 2
    - más difícil de determinar.
  - de 3
    - da márgen por arriba y abajo

HAY QUE JUSTIFICAR tiene que ir el proyecto

Planificación
Release -
Iteración

③ Ingeniería y Calidad de Software.

3 dimensiones del story point en user story
- complejidad: dificultad innata (partes y relaciones)
- esfuerzo: trabajo, horas necesarias ideales (sin distracciones) lineales → depende de quien lo haga.
- incertidumbre: nivel de desinformación asociado a la historia → sirve cuando hay mucha duda. (duda)

valor de homogeneización (capaz el mismo peso se distribuye distinto)
a medida que avanza el proyecto se van haciendo más certeros los estimadores.

MVP → último tema del parcial

Clase 31/08

PARCIAL: user stories + estimación
PRÁCTICO - mínimo producto variable

Gestión de Productos

Proceso → Proyecto → Producto

Pasamos de un "sistemita" a un proyecto. Proyecto de software.
Proyecto ≠ vs Producto
Project manager → administra recursos y personal
gestión de proyecto (unidad de gestión)

para entender que hacemos productos con servicios - product manager

Las organizaciones cuyo core no era el desarrollo de software subestiman el SOFTWARE. Ahora es más tenido en cuenta, TERCERIZAN. Hablamos de proyecto.

a veces coinciden con el product owner

cuando el negocio es el SOFTWARE: hablamos de PRODUCTO

¿Por qué productos?
- satisfacer a los clientes
- obtener mucho dinero
- muchos usuarios (apreciados) (medida del uso del producto)
- realizar una gran visión, cambiar el mundo

características de un producto
- Time to market: tiempo de salir al mercado
  - que sea lo más rápido posible
  - decidir que tiene que tener el producto desde la primera iteración (muchas de las func. no son usadas).

Evolución de productos de software
- focalizar el desarrollo en la experiencia de usuario

Antes - funcionalidad - que haga lo que tiene que hacer que ande y sin errores = UTILIDAD
- ahora no basta. Alguno otro puede hacerlo más usable
  - confiable - seguridad relativa
  - usabilidad - relacionado con UX.
    - conveniente
    - placentero
    - significativo - le CAMBIA la vida a la gente. Es Whatsapp

MVP, MVP, MMF dinosaurio agile

comprender un producto nuevo tiene una hipótesis de valor único: producto/servicio único

Inventar algo "nuevo" es difícil:

UVP: oferta de valor: descripción esencial del producto (servicios en términos de beneficio para tu cliente.