

03-09

Estimación

Dogmático vs. Pragmático

↳ Ir ajustando estimaciones según la situación

Estimaciones Ágiles

↳ Estimar en colaboración en equipo.

↳ Estimaciones relativas: (en las tradicionales son absolutas) en las ágiles uno hace comparaciones para estimar.

- Estimación absoluta: ¿cuánto mide esto?

- Estimación relativa: ¿esto es más alto que esto?

Las estimaciones relativas se aplican a la

certeza (no a la precisión) y requieren menos recursos.

La medida que utiliza el agilismo para estimar se llama **Story Points**.

Las estimaciones de tiempo son las más difíciles porque juegan factores como la experiencia, el conocimiento, etc.

El tamaño se debe estimar teniendo en cuenta que la complejidad del software crece exponencialmente.

Story Point

Para homogeneizar las estimaciones.

↳ Se utilizan en la planning.

↳ utiliza 3 dimensiones para estimar.
(complejidad, esfuerzo, duda).

↓
exponencial

↘ Lineal

↳ probabilidad

horas de c/persona

cuando hay mucha incertidumbre, surge una spike.

FECHA

HOJA No.

Estimamos cada dimensión por separado y unimos estas 3 estimaciones para asignar un número de la serie de Fib.

Velocidad → mide cantidad de puntos de historia que el equipo completa durante una iteración. (y que el PO haya aceptado)

→ No mide la velocidad del equipo, sino la del producto.

Poker estimation → técnica de SCRUM.

→ Utiliza todos estos conceptos de estimaciones ágiles → base story

→ **Story canónica**: "ejemplo" de story sencilla que se usa como punto de comparación (1)
Si surge una más sencilla se usa el $\frac{1}{2}$.
Para un spike → ?

→ Se estiman las US que se quieren implementar en ese sprint. A partir de 13 es mucho.

→ No es obligatorio que este el PO.

→ Todos los miembros del equipo eligen un puntaje, y después lo justifican.

Si hay números muy dispares se vuelve a votar.

→ Al partir una US hay que tener en cuenta que aporten valor de negocio. También, al partir una US, se agrega complejidad para unir estas US (interfaces)

→ Una US se descompone en tareas en los equipos novatos y las tareas se dividen en horas. No todas las US con los mismos puntos se descomponen igual.

→ Una US muy grande es una Epica, que no se puede implementar en una iteración. Las US están encima en el Backlog.

FECHA

HOJA Nº

Productos Mínimos para la Gestión de Productos

MVP Minimal Viable Product

Mínimo: mínimas características necesarias (rápido y barato)

Viable: me permite validar una hipótesis
¿el mercado quiere mi producto?

MMP → producto listo para producción.
(MARKETABLE)

↳ Minimal comercializable product.

MVF Minimal Viable Feature

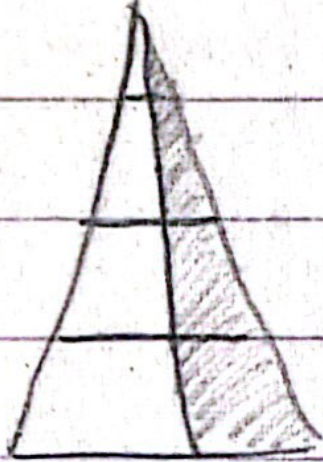
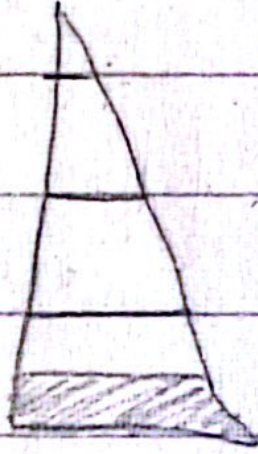
Si quiero validar una sola característica.

MMR Minimal Marketable Release

Versiones del producto ↙

Preparar MVP

- ① Encontrar un Nicho
- ② Crear un roadmap realista
- ③ Investigar la competencia
- ④ Pre-vender el MVP (buscar inversores)
- ⑤ Testear las suposiciones
- ⑥ Asegurar que el MVP resuelve el problema (validar la hipótesis)
- ⑦ Focalizar las funcionalidades principales.

	cómo hacer un MVP	cómo no hacer un MVP
Diseño		
Usabilidad		
Confiablez		
Funcionalidad		