## Estimaciones:

* No ser dogmático sobre nada.
* Ser pragmático es la clave del éxito en casi todo.

**Tips:**

* Si las estimaciones se utilizan como compromisos son muy peligrosas y perjudiciales para cualquier organización.
* Lo más beneficioso en las estimaciones es el “proceso de hacerlas”.
* La estimación podría servir como una gran respuesta temprana sobre si el trabajo planificado es factible o no.
* La estimación puede servir como una gran protección para el equipo.

### Estimaciones agiles:

* Las features/stories son estimadas usando una medida de tamaño relativo conocida como story points (SP).
* Las medidas relativas no son absolutas.
* Story points no es una medida basada en tiempo.

#### Estimación relativa:

* Las personas no saben estimar en términos absolutos.
* Somos buenos comparando cosas.
* Comparar es generalmente más rápido.
* Se obtiene una mejor dinámica grupal y pensamiento de equipo más que individual.
* Se emplea mejor el tiempo de análisis de las storys.

**Tamaño:**

* El tamaño es una medida de la cantidad de trabajo necesaria para producir una feature/story.
* El tamaño indica:
  + Cuán compleja es una feature/story.
  + Cuánto trabajo es requerido para hacer o completar una feature/story.
  + Cuán grande es una feature/story.

### Tamaño vs esfuerzo:

Las estimaciones basadas en tiempo son más propensas a errores debido a varias razones:

* Habilidades.
* Conocimiento.
* Asunciones.
* Experiencia.
* Familiaridad con los dominios de aplicación/negocio.

Tamaño no es esfuerzo.

Para el tamaño se debe elegir una escala. Una vez elegida la escala no se cambia, si se cambia cambiamos el metro patrón.

### Story Point:

* Es una unidad de medida específica (del equipo) de complejidad, riesgo y esfuerzo.
* Story point da idea del “peso” de cada story y decide cuan grande (compleja) es.
* La complejidad de una feature/story tiende a incrementarse exponencialmente.
* Se miden a partir de los datos de la serie de Fibonacci.

### Velocidad (velocity):

* Velocidad/velocity: es una medida (métrica) del progreso de un equipo. Se calcula sumando el número de story points (asignados a cada user story) que el equipo completa durante la iteración.
* Se cuentan los story points de las user story que están completas, no parcialmente completas.
* La velocidad corrige los errores de estimación.

### Propuesta de método de estimación:

**Poker estimation:**

* Popular entre los Agile practicioners.
* Combina opinión de experto, analogía y desegregación.
* Participantes en “planning poker” son desarrolladores.
  + Las personas más competentes en resolver una tarea deben ser quienes las estiman.

#### ¿Cómo decodificar las estimaciones?

Imagen en blanco y negro

Descripción generada automáticamente con confianza media

* 0: quizás usted no tenga idea de su producto o funcionalidad en este punto.
* ½, 1: funcionalidad pequeña (usualmente cosmética).
* 2-3: funcionalidad pequeña a mediana. Es lo que queremos.
* 5: funcionalidad media. Es lo que queremos.
* 8: funcionalidad grande, de todas formas, lo podemos hacer, pero hay que preguntarse sino se puede partir o dividir en algo más pequeño. No es lo mejor.
* 13: ¿alguien puede explicar porque no lo podemos dividir?
* 20: ¿cuál es la razón de negocio que justifica semejante story y más fuerte aún, porque no se puede dividir?
* 40: no hay forma de hacer esto en un sprint.
* 100: confirmación de que está algo muy mal. Mejor ni arrancar.

#### Poker planning:

**Prerrequisitos:**

* Lista de features/stories a ser estimadas.
* Cada estimador tiene un mazo de cartas.

**Pasos:**

1. Determinar la base story (la canónica) que será usada para comparar con las otras stories.
   1. La story a ser estimada se lee a todo el equipo.
   2. Los estimadores discuten la story, haciendo preguntas al producto owner (las que se necesiten).
   3. Cada estimador selecciona una carta y pone la carta boca debajo de en la mesa.
   4. Cuando todos pusieron las cartas, las mismas se exponen al mismo tiempo.
   5. Si todos los estimadores selecciona el mismo valor, ese es el valor. Si no, los estimadores discuten sus resultados, poniendo especial atención en los más altos y en los más bajos. Después de discutir ir a 1.c.
2. Se toma la próxima story, se discute con el producto owner.
3. Cada estimador asigna a la story un valor por comparación contra la base story. “Cuán grande/pequeña, compleja, riesgosa es esta story comparada con la base story?”- Volver a 1.c.

## Gestión de productos:

### ¿Por qué creamos productos?

* Para satisfacer a los clientes.
* Para tener muchos usuarios logueados.
* Para obtener mucho dinero.
* Realizar una gran visión, cambiar el mundo.

### ¿Qué características realmente utilizamos de un producto de software?

Diagrama

Descripción generada automáticamente con confianza media

### Evolución de los productos de software:

Diagrama

Descripción generada automáticamente

### Productos mínimos para la gestión de productos:

* Minimal viable product (producto mínimo viable).
* Minimal viable feature (característica mínima viable).
* Minimal release feature (características mínimas del release).
* Minimal marketable feature (característica mínima comercializable).

#### Relación:

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

### MVP:

* Versión de un nuevo producto creado con el menor esfuerzo posible.
* Dirigido a un subconjunto de clientes potenciales.
* Utilizado para obtener aprendizaje validado.
* Más cercano a los prototipos que a una versión real funcionando de un producto.
* Es un concepto de Lean Startup que enfatiza el impacto del aprendizaje en el desarrollo de nuevos productos.
* Una premisa clave detrás de la idea de MVP es que usted produce un producto real que puede ofrecer a los clientes y observar su comportamiento real con el producto o servicio.
* “Versión de un nuevo producto que permite a un equipo que permite a un equipo recopilar la cantidad máxima de aprendizaje validado sobre clientes con el menor esfuerzo”. Este aprendizaje validado viene en forma de si sus clientes realmente comprarán su producto.
* Ver lo que la gente realmente hace con respecto a un producto es mucho más confiable que preguntarle a la gente qué harían.
* Tiene el valor suficiente para que las personas estén dispuestas a usarlo o comprarlo inicialmente.
* Demuestra suficiente beneficio futuro para retener a los primeros usuarios.
* Proporciona un ciclo de retroalimentación para guiar el desarrollo futuro.

### MMF:

* Es la pieza más pequeña de funcionalidad que puede ser liberada.
* Tiene valor tanto para la organización como para los usuarios.
* Es parte de un MMR o MMP.

### MMP:

* Primer release de un MMR dirigido a primeros usuarios (early adopters).
* Focalizado en características clave que satisfarán a este grupo clave.

### MMR:

* Release de un producto que tiene el conjunto de características más pequeño posible.
* El incremento más pequeño que ofrece un valor nuevo a los usuarios y satisface sus necesidades actuales.
* MMP = MMR1.

### MVF:

* Es una mini versión del MVP.
* Es una característica a pequeña escala que se puede construir e implementar rápidamente, utilizando recursos mínimos, para una población objetivo para probar la utilidad y adopción de la característica.
* Un MVF debe proporcionar un valor claro a los usuarios y ser fácil de usar.
* MVF requiere recursos mínimos, los estándares de calidad de la industria y la producción deben guiar el diseño y la confiabilidad.
* El grupo de usuarios para un MVF, son los primeros en adoptar los clientes leales que han compartido conocimientos anteriormente o los miembros de una junta asesora de clientes. Usurarios flexibles y tolerantes.
* Los resultados le ayudarán a tomar decisiones estratégicas sobre productos.

### Errores comunes:

* Confundir a un MVP, que se enfoca en el aprendizaje, con característica comercializable mínima (MMF) o con producto comercializable mínimo (MMP), ambos se enfocan en “ganar”.
* El riesgo de esto es entregar algo sin considerar si es lo correcto que satisface las necesidades del cliente.
* Enfatizar la parte mínima del MVP con exclusión de la parte viable. El producto entregado no es de calidad suficiente para proporcionar una evaluación precisa de si los clientes utilizarán el producto.
* Entregar lo que consideran un MVP, y luego no hacer más cambios a ese producto, independientemente de los comentarios que reciban al respecto.

### Valor vs desperdicio:

* Lean Thinking define la creación de valor como proveer beneficio a los clientes, cualquier otra cosa es desperdicio.
* La productividad de un Startup no puede medirse en términos de cuánto se construye cada día, por el contrario, se debe medir en términos de averiguar la cosa correcta a construir cada día.

#### Build-experiment-learn feedback loop:

Diagrama

Descripción generada automáticamente

#### Hacia la creación de valor:

* El éxito no es entregar un producto, el éxito se trata de entregar un producto (o característica de producto) que el cliente usará.
* La forma de hacerlo es alinear los esfuerzos continuamente hacia las necesidades reales de los clientes.
* The build-experiment-learn feedback loop permite descubrir las necesidades del cliente y alinearlas metodológicamente.

### La fase construir: MVP:

* Ingresar lo más rápido posible con un MVP.
* Un MVP varía en complejidad desde pruebas de humo extremadamente simples hasta prototipos tempranos.
  + Prueba de humo (smoke tests): no hay producto construido aún.
  + Prototipos: un producto con características faltantes o problemas.
* Decidir qué tan complejo no se puede hacer un MPV de manera formulada:
  + Requiere juicio.
  + En caso de duda, simplifique.
  + Evite la construcción excesiva y la promesa excesiva.
  + Cualquier trabajo adicional más allá de lo que necesita para comenzar el ciclo podría ser un desperdicio.
* Un MVP no sólo habla sobre el diseño del producto y las preguntas técnicas, sino que también sirve para probar hipótesis comerciales fundamentales.
* Por lo tanto, sirve para proporcionar una dosis necesaria de realidad.

### Dilema: la audacia de cero:

* A menudo es más fácil recaudar dinero cuando tiene:
  + Cero ingresos.
  + Cero clientes.
  + Tracción cero.

Que cuando tienes una pequeña cantidad de cada uno.

* Cero invita a la imaginación, pero los números pequeños hacen preguntas sobre si los números grandes alguna vez se materializarán.
* Este fenómeno (llamado la audacia de cero) crea un incentivo brutal:
  + Aplazar el lanzamiento de cualquier versión de un producto hasta que esté seguro del éxito.
* Si pospone experimentar con su MVP, surgirán algunos resultados desafortunados como:
  + La cantidad de trabajo desperdiciado puede aumentar.
  + Se perderán los comentarios esenciales.
  + El riesgo de que su startup construya algo que nadie quiere puede aumentar.
* Use un MVP para experimentar (inicialmente, en silencio) con los primeros usuarios en el mercado.
* Verifique su concepto probando TODOS sus elementos, comenzando por los más riesgosos.

### Supuestos de “Saltos de Fe”:

* Los elementos más riesgosos del plan/concepto de una startup (es decir, las partes de las que todo depende) se denominan supuestos de salto de fe.
* La mayoría de las personas no conocen una determinada solución (o incluso un problema); pero una vez que experimentan la solución, no pueden imaginar cómo vivirían sin ella.
* Hipótesis del valor:
  + Prueba si el producto realmente está entregando valor a los clientes después de que comienzan a usarlo.
  + Una métrica de prueba: tasa de retención.
* Hipótesis de crecimiento:
  + Prueba cómo nuevos clientes descubrirán el producto.
  + Una métrica de prueba: tasa de referencia o Net Promoter Score (NPS).

### Preparar un MVP:

1. Encontrar un nicho.
2. Crear un roadmap realista.
3. Investigar la competencia.
4. Pre-vender el MVP.
5. Testear las suposiciones.
6. Asegurarse que el MVP resuelve el problema correcto.
7. Focalizar en las funcionalidades principales.

### Características de un MVP:

* Diseño:
  + Diseño adecuado.
  + Consigue una UX que deleita.
  + Logra satisfacer el aspecto visual y de interacción.
* Usabilidad:
  + Tiene suficiente valor para que la gente esté dispuesta a usarlo/comprarlo.
  + Resulta útil para su público objetivo.
* Confiabilidad:
  + Involucra a los early adopters para que puedan confiar en la solución plenamente.
  + Incluso cuando tenga poco tiempo en el mercado.
* Funcionalidad:
  + Tiene las funciones necesarias para solucionar un problema específico.
  + Satisface las demandas y permite evaluar las funciones a implementar más adelante.

Diagrama

Descripción generada automáticamenteUn conjunto de letras negras en un fondo blanco

Descripción generada automáticamente con confianza media

### Matriz de priorización para el MVP:

Gráfico, Gráfico de rectángulos

Descripción generada automáticamente