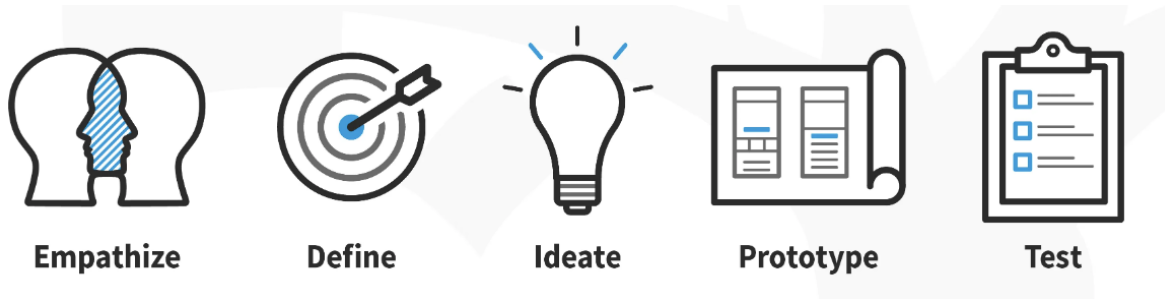


DESIGN THINKING

Es un proceso no lineal, iterativo que se utiliza para la resolución de problemas y tiene 5 fases:



Lo más útil es utilizarlo para solucionar problemas desconocidos.

Ver video Design Thinking

Es crucial desarrollar y perfeccionar las habilidades para comprender y abordar los cambios rápidos en los entornos y comportamientos de los usuarios.

Los equipos tienen la libertad de generar soluciones innovadoras. Al usarlo, se puede obtener información de difícil acceso y aplicar una colección de métodos prácticos para ayudar a encontrar respuestas innovadoras.

Fases

Empatizar: Búsqueda de las necesidades del usuario.

Se debe ser empático para entender el problema que se quiere solucionar. Normalmente haciendo una investigación. Permite dejar de lado las propias ideas y obtener una mirada interna de parte de los usuarios.

Definir: Definir las necesidades y problemas del usuario

Se acumula la información recopilada y se analiza y sintetiza las observaciones para definir los problemas centrales que el equipo identificó. A estas definiciones se le llaman “Problem Statement”.

Idear: Desafiar y crear ideas.

Ahora se cuando se comienza a pensar fuera de la caja, se busca alternativas para visualizar el problema e identificar soluciones innovadoras al problema creado. → Lluvia de ideas es muy útil

Prototipar: Comenzar a crear las soluciones.

Es una fase experimental. El objetivo es identificar la mejor solución para cada problema encontrado. Consiste en una versión muy barata y pequeña del producto. Puede ser prototipado de papel o prototipar partes específicas del producto y no el producto es tu totalidad.

Probar: Probar las soluciones

Se deben evaluar rigurosamente los prototipos. Aunque es la última fase, el design thinking es iterativo, es decir se puede definir uno o más problemas. Se puede volver a fases anteriores, alternar y refinar para obtener alternativas.

<https://xn--designthinkingespaa-d4b.com/que-es-un-brainstorming-y-como-te-ayuda-a-generar-ideas-innovadoras>

BRAINSTORMING

Metodología de generación de ideas cuyo objetivo es encontrar la mayor cantidad de ideas posibles en relación a un reto o problema a solucionar.

Ventajas:

- Potenciar una cultura ágil y de innovación
- Aumentar creatividad de las personas
- Puede servir como punto de partida a la hora de empezar un proyecto

Claves para un buen brainstorming

1. Realizarlo en grupo : 4/6
2. Buscar cantidad: no se juzgan las ideas, dejar que fluyan.
3. Aprovechar las ideas de los demás para generar otras nuevas.
4. Anotar ideas y numerarlas: Establecer número objetivo.
5. Que haya un facilitador: Quien guía el brainstorming, escribe las ideas que surgen y puede dar las suyas.
6. Nombrar persona encargada de anotar ideas: Puede ser el facilitador pero sin decir o escribir ideas que le surjan.
7. Establecer tiempos y cumplirlos: Mucho más eficaz si cuenta con principio y final, los participantes deben conocer el tiempo que cuentan.
8. El espacio es importante: Elegir un lugar cómodo es inspirador. Paredes, pizarrones, asientos cómodos, luz agradable.
9. Lanza tus ideas de forma clara y autoexplicativa. “crear una agencia de viajes para miembros de espacios de coworking.”
10. Incluye estímulos creativos: SCAMPER, Hibridación, Brainwriting.
11. Buscar ideas locas e inesperadas

Paso a paso del brainstorming

→ FASE 1: Diseña la sesión

- Elección del espacio:

El espacio cuenta con mucha luz, idealmente natural.

Las sillas y lugares en los que los participantes pueden sentarse son cómodos, e invitan a sentirse a gusto.

El ambiente que se va a crear será distendido, con espacio para colocar comida y bebida.

Las superficies de la sala tienen mucho espacio para poder pintar en ellas. Como por ejemplo paredes en las que se pueda escribir y dibujar.

El lugar es lo suficientemente grande como para que los participantes sientan que pueden moverse con libertad.

Existen estímulos visuales inspiradores alrededor, tanto por color como por elementos que os hagan sentir en un lugar creativo en el que se respira un espíritu de cambio e ideas.

- Elegir quién será el facilitador: Ideal es que sea externo.

Preparar el espacio, hojas, post its, superficies amplias y marcadores.

Explicar las reglas del juego y comunicar el tiempo con el que cuentan. Dinamizar la sesión, abrir perspectivas y animarlos para dar muchas ideas.

Recoger conclusiones e ideas elegidas por parte del equipo. Así como realizar sugerencias, si procede, sobre los siguientes pasos.

→ FASE 2: Desarrolla la sesión de brainstorming.

- Ejercicio de calentamiento creativo:

- Decirles a los participantes que tienen un minuto para escribir el mayor número de usos posibles de un elemento.
- Pedir que compartan algunos usos que encontraron,
- Recordar que todas las ideas sirven, deben mantener esa actitud de decir todo lo que se les ocurra.

- Inicio de la sesión → Dar pautas fundamentales del proceso
 - Generar mayor número posible de ideas
 - Establecer tiempo → recomendado: 25/35 minutos
 - Recordar las reglas, proceso de innovación.
 - Única idea mala: La que no se dice
- Votación de las mejores ideas:
 - Decir con cuantos votos cuenta cada participante para votar por la idea que más les gusta.

→ FASE 3: Conclusiones

- Si se llevó a cabo en design thinking → Prototipado

Plataformas online para Design Thinking: MIRO y MURAL

Product Use Case Scenarios

Se utilizan los PUC para comunicar las intenciones del producto a los stakeholders (Aquellos que se ven afectados por el proyecto). Es un documento fácil para mostrar la funcionalidad del producto.

Lo mejor es entregar una copia de los requerimientos especificados a los stakeholders, pedirles que lo firmen si están satisfechos.

La diferencia con los BUC es que los BUC contienen todas las funcionalidades que responden a los eventos que disparan al negocio mientras que los PUC contiene solo esa funcionalidad que será implementada en el producto mismo.

1. Identificar los BE
2. Elegir uno y llevarlo a descubrir la funcionalidad que responde a ese evento
3. Escribir escenario para este evento y cuando las partes interesadas estén satisfechas con el escenario, determine cuánto de ese BUC se puede implementar como producto. Lo que resulta ser se convierte en el PUC

Ejemplo:

Business Event Name: *Passenger decides to check in.*

Business Use Case Name: *Check passenger onto flight.*

Trigger: *Passenger's ticket, record locator, or identity and flight.*

Preconditions: *The passenger must have a reservation and a passport.*

Interested Stakeholders: *Check-in agent, marketing, baggage handling, reservations, flight manifest system, workflow, security, destination country's immigration.*

Active Stakeholders: *Passenger (trigger), check-in agent.*

- 1. Locate the passenger's reservation.*
- 2. Ensure the passenger is correctly identified and connected to the right reservation.*
- 3. Check that the passport is valid and belongs to the passenger.*
See procedure guidelines EU-175.
- 4. Attach the passenger's frequent-flyer number to the reservation.*
- 5. Allocate a seat.*
- 6. Get correct responses to security questions.*
- 7. Check the baggage onto the flight.*
- 8. Print and convey to the passenger the boarding pass and bag tags.*
- 9. Wish the passenger a pleasant flight.*

Outcome: *The passenger is recorded as checked onto the flight, the bags are assigned to the flight, a seat is allocated, and the passenger is in possession of a boarding pass and bag claim stubs.*

También necesitamos aportes de las partes interesadas que comprendan las implicaciones técnicas y comerciales y las posibilidades para el límite del producto junto con los objetivos de negocio para el proyecto. Teniendo esa información y con las partes interesadas decididas que la combinación óptima de beneficio, costo y riesgo será logrado mediante la construcción de una máquina que permite a los pasajeros controlarse a sí mismos en sus vuelos.

El escenario PUC para la máquina muestra lo que pretende que haga:

Product Use Case Name: Passenger checks onto flight.

Trigger: Passenger activating the machine.

Preconditions: The passenger must have a reservation.

Interested Stakeholders: Passenger, check-in agent, marketing, baggage handling, reservations, flight manifest system, workflow, security, destination country's immigration.

Actor: Passenger.

- 1. The product asks for the passenger's identity or record locator.*
- 2. The passenger supplies one or the other and the product locates the passenger's reservation.*
- 3. The product asks for a frequent-flyer number if it is not already attached to the reservation.*
- 4. The product asks for and scans the passport if needed.*
- 5. The product shows the allocated seat and accepts the passenger's changes if needed.*
- 6. The product asks for the number of bags and for answers to the security questions.*
- 7. The product checks the baggage onto the flight, and prints the bag tags.*
- 8. The product prints the boarding pass or sends it to the passenger's phone.*
- 9. The product directs the passenger to the bag drop and departure gate.*

Outcome: The passenger is recorded as checked onto the flight, the bags are assigned to the flight, a seat is allocated, and the passenger has a boarding pass and bag claim stubs.

Si los stakeholders hubiesen tomado decisiones más definidas con más restricciones entonces el PUC escenario se vería distinto.

El PUC explica cuál es la intención del producto a un nivel que es aceptable para el negocio. Será necesario hacerle revisiones mientras se presenta, para el momento que finalizaron debe ser muy exacto de lo que se va a construir.

Evitar Ambigüedad

Cuando se escribe requerimientos, debemos pensar en más desafíos que nombres iguales que otra. Si el contexto no es claro, llevará a una ambigüedad.

Ejemplos:

- *El producto debe mostrar el clima por las próximas 24 hs.* → Aca depende del tipo de requerimiento y lo que está cerca de la especificación.

Se recomienda agrupar los requerimientos por PUC y escribirlos uno a la vez, este sistema de organización reduce la ambigüedad.

- *El producto debe comunicar todas las calles predecidas a congelarse.* → “Todas” se refiere a las calles conocidas por el producto o solo las que busca el usuario

Si bien cualquier cosa tiene el potencial de ser ambiguo, el escenario PUC, al establecer un contexto para el requisito, minimiza el riesgo de ambigüedad.

Eliminar todos los pronombres de los requisitos y reemplácese con el sujeto u objeto al que se refieren los pronombres. (Nota la ambigüedad potencial de la oración anterior si hubiéramos dicho "ellos" en lugar de “los pronombres”).

Hacer que el stakeholder lea los requerimientos para asegurarse que estén bien es super importante.

Aca se escribe una descripción de los requerimientos, el verdadero requerimiento es revelado cuando se escribe el criterio de ajuste.

Requerimientos técnicos

Son requerimientos que son funcionalidades necesarias únicamente por la tecnología. No están por razones de negocio sino para que la implementación funcione. Estos deben ser escritos aparte o ser identificados como técnicos y deben ser contados una vez que los requerimientos de negocio hayan sido entendidos en su totalidad.

Agrupando requerimientos

Lo mejor es agrupar los requerimientos funcionales según caso de uso, las ventajas están en que es más fácil descubrir grupos relacionados y testearlos en su totalidad. Pero existen otras formas de agrupar los requerimientos.

Agrupar por feature lo hace más fácil para manipularlos y ajustar la especificación cuando el mercado cambia.

- Ser capaz de involucrar a las partes interesadas con diferentes profundidades, amplitudes y focos de interés
- Para ayudarlo a descubrir los requisitos atómicos en primer lugar
- Ser capaz de manejar el volumen y la complejidad

Alternativas a requisitos funcionales (239-242)

Existen gran cantidad de formas para describir la funcionalidad de un producto.

- Scenarios:

A veces se quiere agregar detalles de implementación a los escenarios y utilizarlos como especificaciones. Ahí se debe necesitar revisar los escenarios. Los devs y testers deben estar seguros de que pueden realizar y probar el producto basado en este nuevo escenario planteado, si el escenario se vuelve muy largo o elaborado entonces se debe revertir y escribir requerimientos funcionales en una manera normal.

No se debe seguir este camino si está subcontractando la construcción a una empresa externa, proveedor u otro departamento de su organización. Para la subcontractación, es mejor evitar el aumento del potencial de mala interpretación escribiendo los requisitos funcionales.

- User stories:

Son otra forma de describir funcionalidades necesarias del producto, se pueden considerar una alternativa para escribir requerimientos funcionales.

As a [role] I want [functionality] so that [reason for or use of the functionality].

Son escritas generalmente por el Product Owner, representante del cliente, que forma parte del equipo ágil. Generalmente estas historias son features que el product owner piensa que deben ser parte del producto.

La intención no es especificar requerimientos, sino actuar como un punto de partida para los requerimientos. Estos se descubren como procesos de desarrollo a partir de conversaciones entre devs y stakeholders.

- Business Process Models

Puede ser usado como una forma de llegar a entender la funcionalidad necesaria. Los stakeholders se relacionan más fácilmente con los diagramas que con los escenarios de texto. La más común son técnicas de UML.

También es posible utilizar BPM como base, y luego escribir los requisitos atómicos de cada una de las actividades que se muestran en el diagrama.

- Requirements for COTS (commercial off-the-shelf)

Se refiere a cualquier producto de software instalable, una alternativa a la creación de software on scratch. Contiene gran parte de la funcionalidad necesaria, pero por lo general requiere cierta cantidad de modificaciones posteriores a la instalación.

En el caso en que la empresa compre un COTS y luego quiera realizar cambios para que se asemejen más a la funcionalidad esperada, el analista del negocio debe indicar a la organización la áreas de deficiencia, las áreas donde COTS producto no machea con el trabajo esperado.

No tiene mucho sentido querer documentar la funcionalidad del COTS pero es recomendado que se comience el análisis del negocio construyendo un diagrama de contexto del área de trabajo que será afectada por los COTS.

No es necesario escribir un completo BUC escenario, pero se debe prestar atención a los datos que llegan y que salen.

Comparar estos datos con las entradas y salidas de datos del producto COTS y escribir los requisitos para cualquier modificación necesaria para traer el COTS datos de acuerdo con la necesidad organizacional de datos.

Este documento describiría las nuevas prácticas de trabajo necesarios para acomodar el producto COTS, análisis de negocio es crucial para asegurar que las funciones del producto COTS realmente matcheen las necesidades de la organización. La tarea acá es descubrir y entender las diferencias entre COTS product y la organización.

Tipos de non-functional requirements (250-251)

Leer extensión de cada tipo de NFR

Los tipos de NFR se pueden ver en el template de Volere Requirements Specification. Es difícil decidir qué tipo de requerimiento es, cuando ocurre esto puede ser que se están poniendo muchos requerimientos juntos.

Se puede poner a un requerimiento muchos tipos, estas categorías no son importantes igual, los requerimientos si.

1. Mirar y sentir: El espíritu de la apariencia del producto.
2. Usabilidad y humanidad: El producto es fácil de usar, o alguna consideración especial para una mejor experiencia de usuario
3. Performance: qué tan rápido, seguro, disponible, y que tan exacto la funcionalidad debe ser.
4. Operacional: El ambiente donde opera el producto y cualquier consideración de esto que deba ser llevada a cabo.
5. Mantenibilidad y soporte: Cambios esperados, tiempo necesario para hacerlo, especificación del soporte brindado al producto.
6. Seguridad: Acceso, confidencialidad, recuperación, auditoría del producto.

7. Cultural y político: requerimientos especiales que vienen porque la cultura y ciertas personas que están en contacto con el producto.
8. Legal: Las leyes y estándares para aplicar al producto.

El método para atacar es inicialmente capturar la intención de los requerimientos para los stakeholders.