

Ingeniería de Software II

INGENIERÍA DE SOFTWARE

Créditos académicos: 3

Mgs. David Alberto Angarita García

adavidalberto@eam.edu.co

ACUERDOS



Desconéctate para
Conectarte



JUSTIFICACIÓN

- El estudiante adquiere los conocimientos necesarios para comprender los diversos sistemas de negocios a partir de metodologías dentro del entorno de modelado las cuales permiten la abstracción de diferentes sistemas o entornos de dominio.
- Este espacio académico brinda al estudiante la oportunidad de adquirir competencias relacionadas con la especificación, diseño y la gestión de modelos que soporten la descripción y la implementación de diversos sistemas de información. Todo esto hace parte de la vinculación de nuevos conocimientos adquiridos, la utilización de guías, técnicas y herramientas.
- Considerando las metodologías que el estudiante adquiere por medio del uso de herramientas que apoyan la construcción de modelos y de cómo todo esto hace parte del desarrollo de proyectos reales, a tal punto de crear diseños finales del sistema informático para su posterior implementación. De este modo, las capacidades que los estudiantes adquieren los hacen más competentes en el mundo laboral de tal forma que comprenda los diversos entornos a los que se enfrentan como profesionales.

OBJETIVO

- Brindar al estudiante los conocimientos necesarios para el modelamiento de entornos o escenarios que hacen parte de un dominio real y así llegar al punto de la sistematización y automatización de dichos entornos a través de herramientas y técnicas de apoyo.

UNIDADES TEMÁTICAS

UNIDAD 1: Modelado del dominio

- Concepto General de Clases y el modelado orientado a objetos
- Relaciones entre clases
- Abstracción de un sistema en un diagrama UML.
- Validación del Modelado de Dominio.
- Herencia y Generalización
- Polimorfismo y Encapsulamiento
- Estereotipo e Interfaz

UNIDAD 2: Diseño procedimental

- Conceptos Generales de los Diagramas Procedimentales
- Diagrama de Secuencia
- Diagrama de Colaboración
- Diagrama de Estados.

UNIDADES TEMÁTICAS

- **UNIDAD 3: Diseño De GUI**

- Normas para el Diseño de la GUI
- Clasificación de las Interfaces
- Diseño Centrado en el Usuario

UNIDAD 4: Patrones de diseño

- Conceptos Generales de Patrones de Diseño
- Tipo de Patrones de Diseño
- Principios de Diseño en el POO
- Patrones de Diseño de GoF

UNIDADES TEMÁTICAS

UNIDAD 5: Control Y Administración De Cambios

- Control de Versiones
- Control de Cambios.

EVALUACIÓN

- Corte 1: 30%
 - Participaciones en clase, talleres en clase, extra clase, análisis de documentos y lecturas recomendadas: 40%
 - **Parcial: 60%**
- Corte 2: 30%
 - Participaciones en clase, talleres en clase, extra clase, análisis de documentos y lecturas recomendadas: 40%
 - **Parcial: 60%**
- Corte 3: 40%
 - Participaciones en clase, talleres en clase, extra clase, análisis de documentos y lecturas recomendadas: 40%
 - **Proyecto: 60%**

BIBLIOGRAFÍA

- Calero, C., Moraga, M. A., & Piattini, M. G. (2012). Calidad del producto y proceso software. España: Ra-Ma.
- Pressman, R. S. (2010). Ingenieria del Software Un enfoque practico. Mexico: McGrawHill.
- **WEBGRAFÍA**
 - <https://www.ieee.org/index.html>, visitada el 8 de octubre del 2014.
 - <http://www.sei.cmu.edu/>, visitada el 8 de octubre del 2014.
 - <http://fedesoft.org/>, visitada el 8 de octubre del 2014.
 - <http://www.mintic.gov.co/>, visitada el 8 de octubre del 2014.

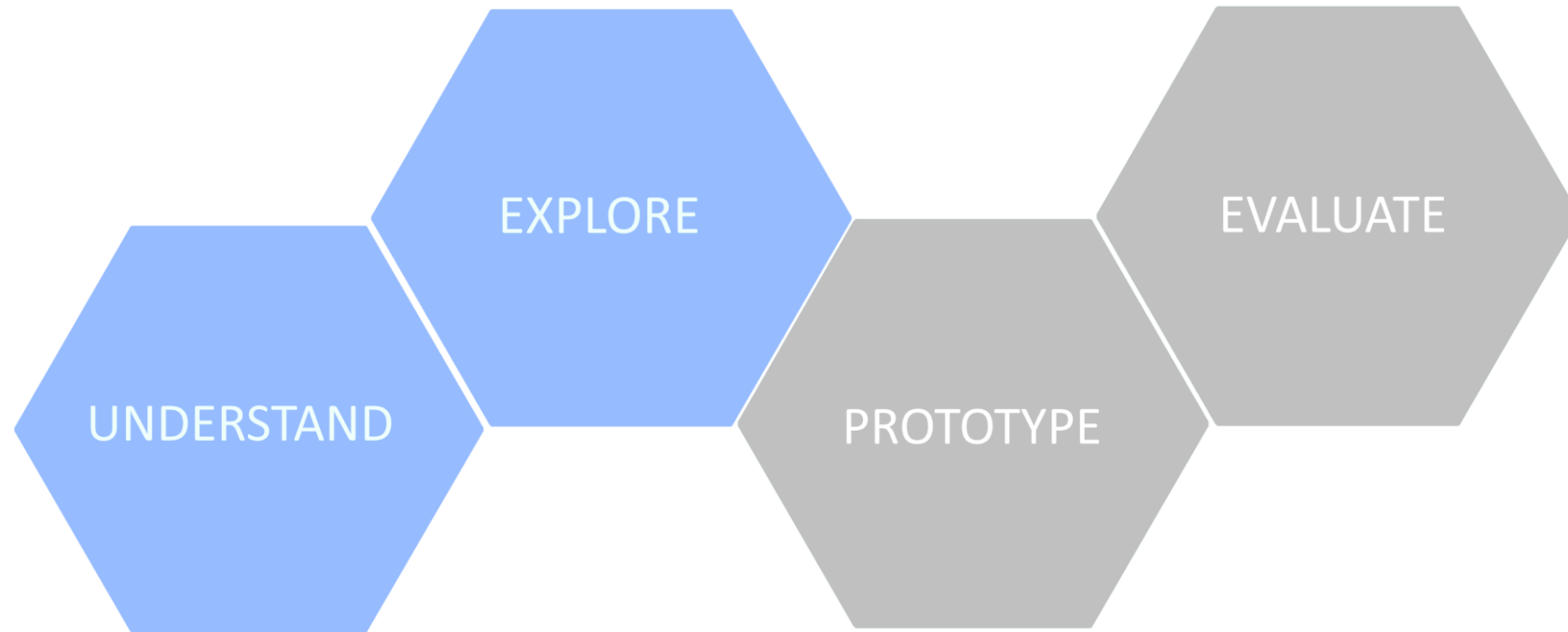
CONCEPTO

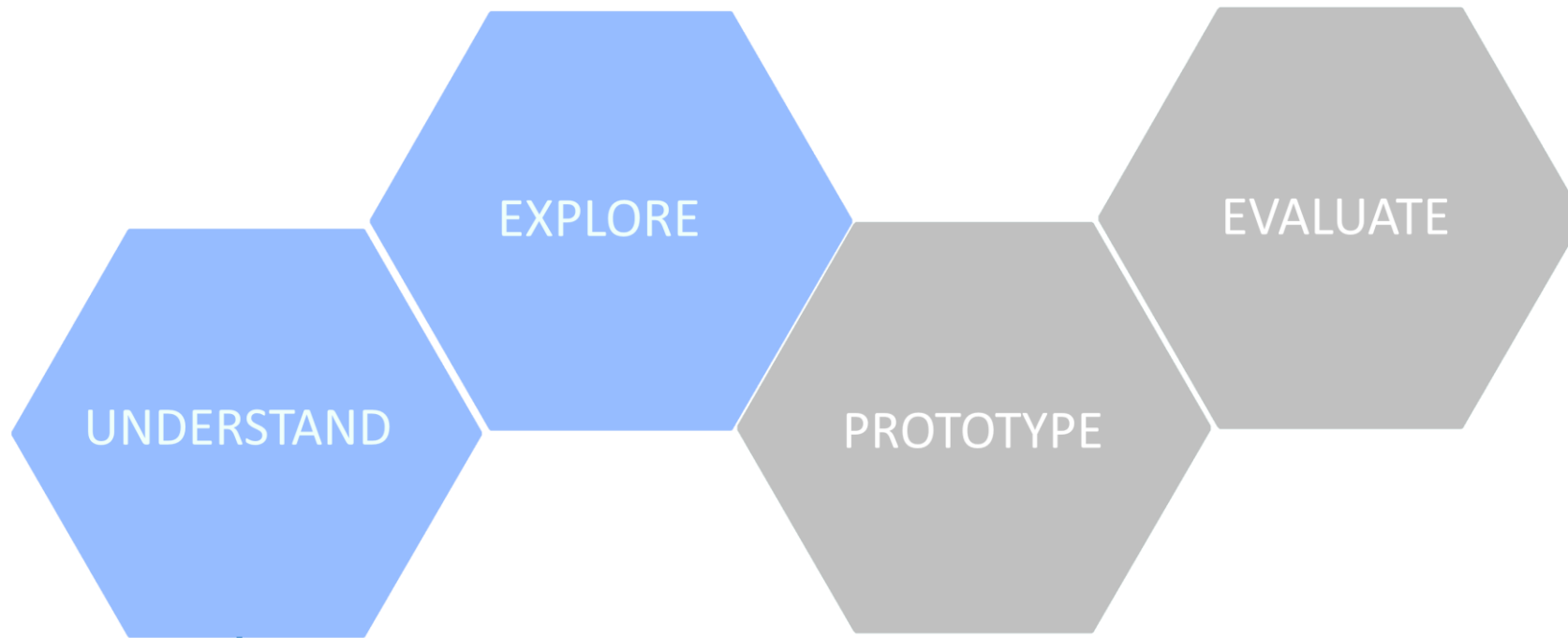
- Su utilidad radica en ser una forma de “inspiración” para el diseño de los objetos *software*
- Es entrada para muchos de los artefactos que se construyen en un proceso *software*
- Un modelo de dominio muestra las **clases conceptuales** significativas en un dominio del problema
- Se centra en las abstracciones relevantes, vocabulario del dominio e información del dominio
- Es el artefacto clave del análisis orientado a objetos
- En UML se utilizan los diagramas de clases para representar los modelos de dominio

“Un diseñador... no busca la solución hasta que identifica el problema real, e incluso entonces, antes de tratar resolverlo, trata de considerar una gran variedad de soluciones posibles. Es entonces cuando finalmente decide qué solución proponer. A este proceso se le conoce con el nombre de Design Thinking”

Don Norman







Empatizamos
No dar una solución inmediata
Entender al cliente o usuario final
Verificar su actualidad
Ser creativos



¿Tasa de fracaso de una startup?



¿Tasa de fracaso de una empresa al lanzar un producto disruptivo?





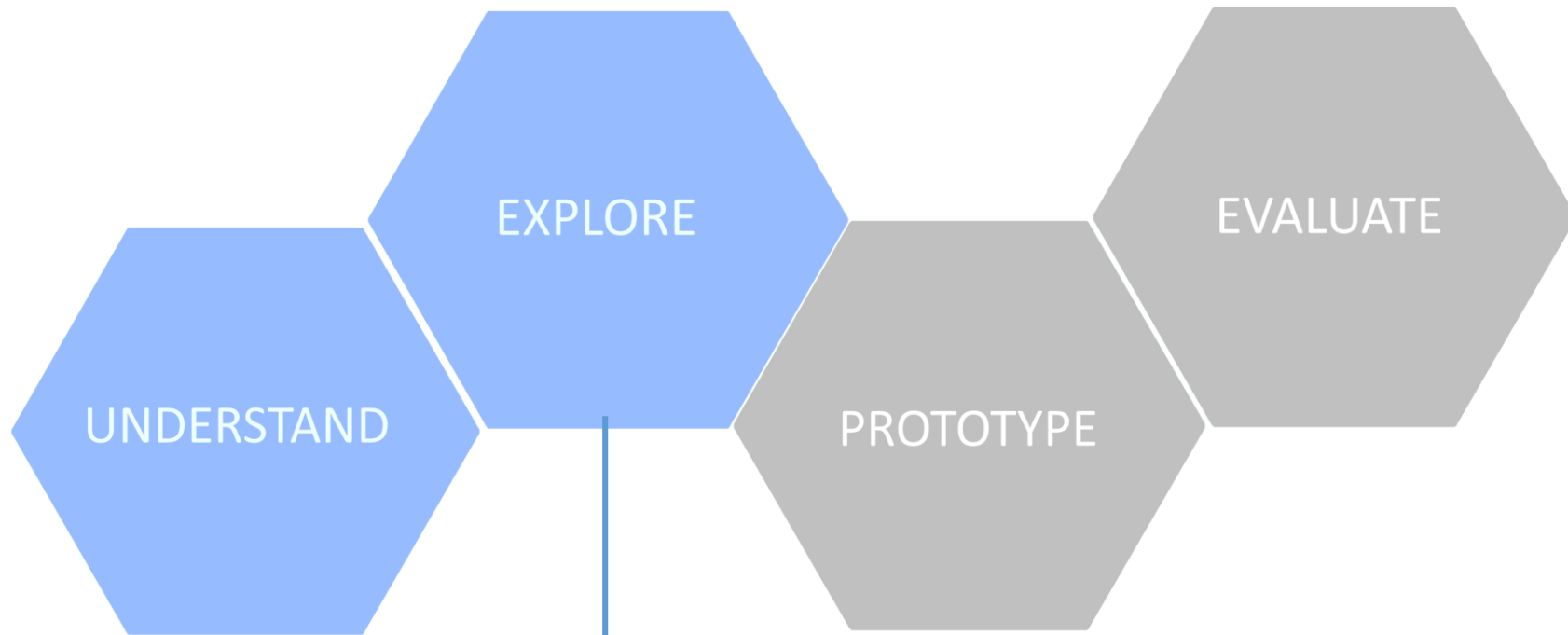
¿Cómo mejorar esta clase?





“Visualice cómo juega la gente. Lo más simple era un ‘tap’. Busque algo fácil de aprender y difícil de dominar como Nolan Bushnell dijo una vez...”

Dong Nguyen



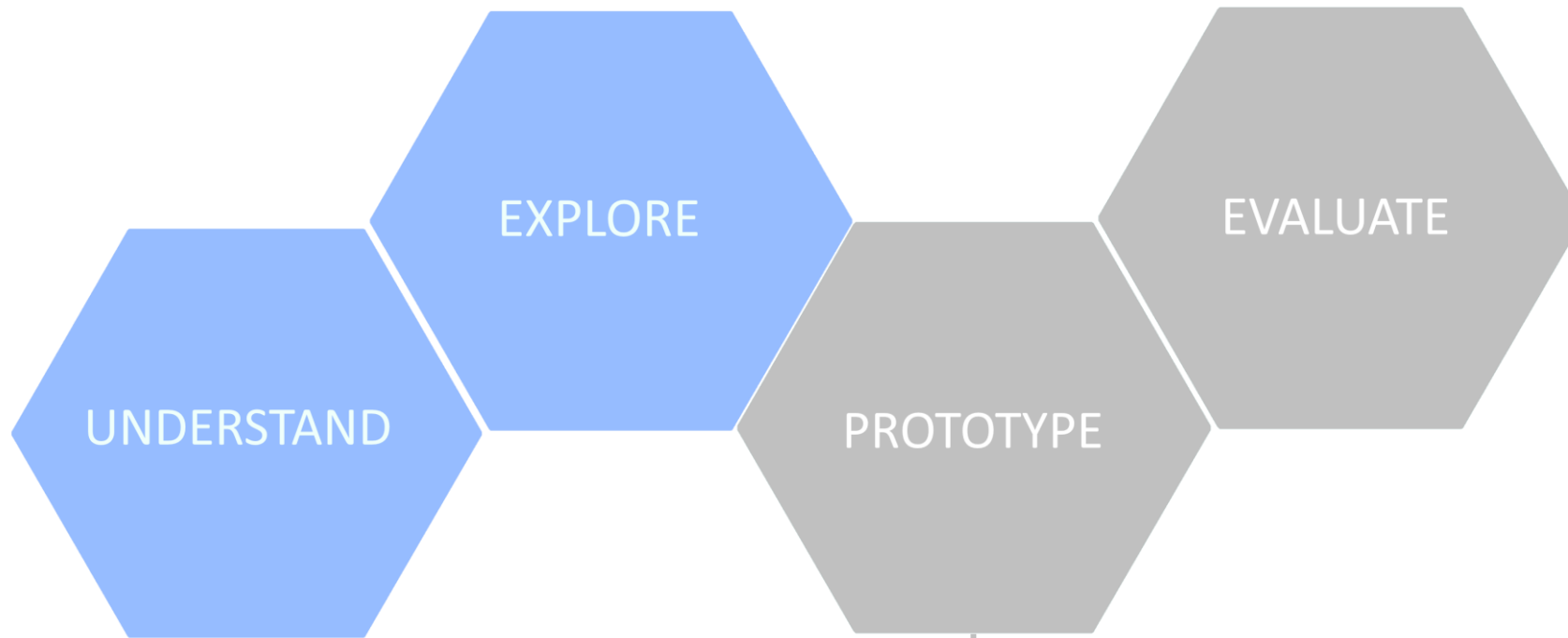
Divergir

Buscar soluciones actuales

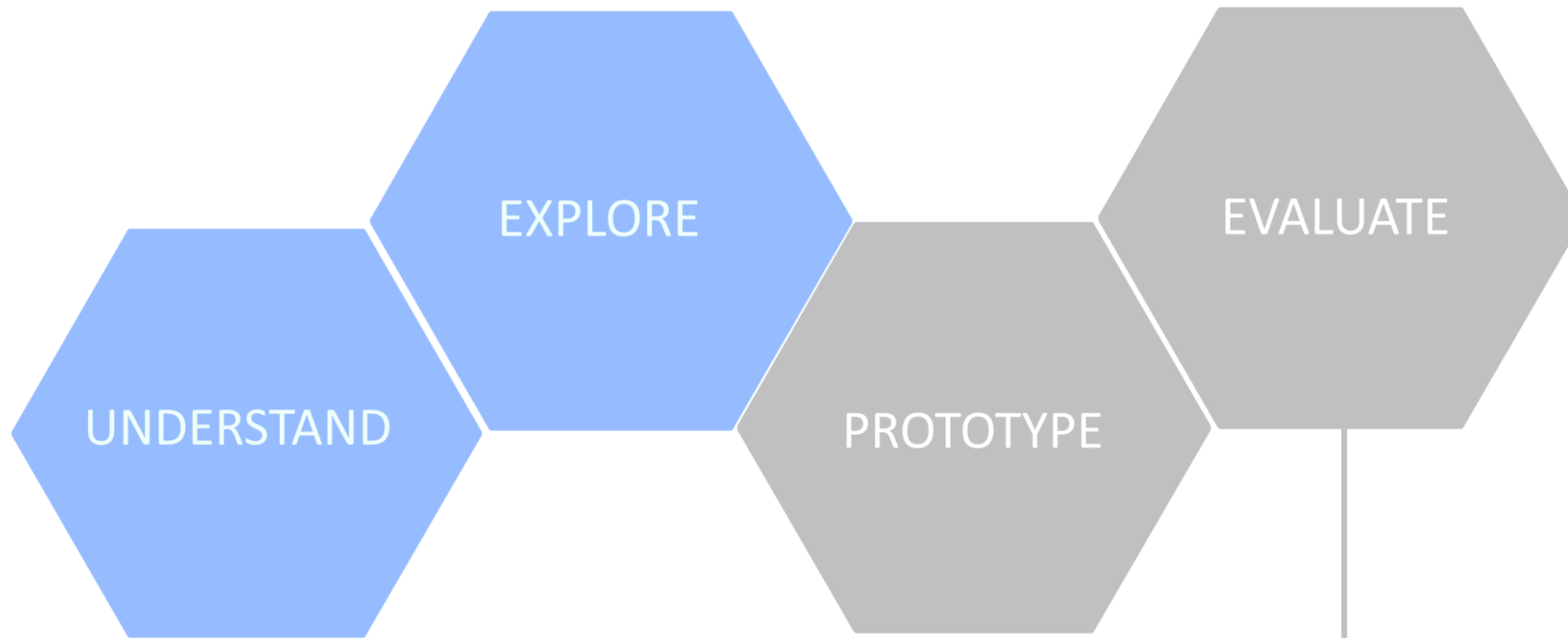
Diferencias entre esas soluciones

Ser creativo

Buscar diferentes soluciones a las actuales



Elegir la mejor solución
Lo que hace distinto a mi propuesta
Reducirlo a una máxima expresión
Explicárselo al cliente de la forma más fácil posible



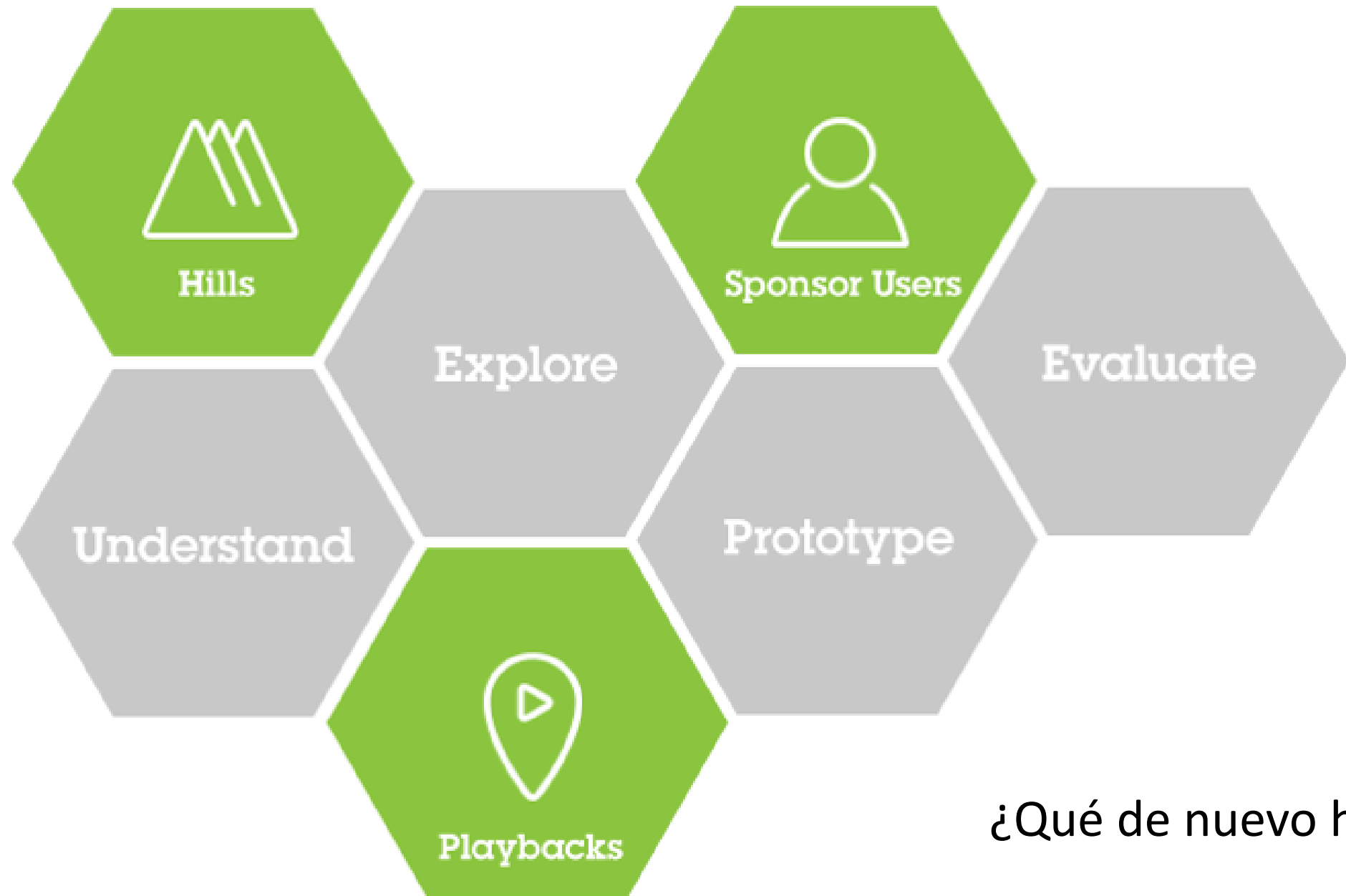
Voy a buscarle la máxima cantidad de problemas
Voy a buscar personas que destruyan mi propuesta
Equipo de trabajo
Clientes
Usuarios finales





“Burbn tenía demasiada funcionalidad y los usuarios no querían un producto complicado. Decidí centrarme en una necesidad concreta, y hacerlo bien”

Kevin Systrom



¿Qué de nuevo hay?

EJERCICIO

- Elija una problemática que haya identificado dentro de la universidad, realice el proceso anteriormente visto y proponga una solución desde uso del software