



aleasoluciones

Outside In TDD en diseño aplicando IDD(Interaction Driven Design)

Rubén Eguiluz (ruben) @regiluze

Alberto Pérez (apa) @APA42



aleasoluciones

Muchas
Gracias

Eskerrik
Asko



eternax
Berlin Studio

flywire

autentia

biko²



CODESAI

idealista



aleasoluciones



CodelyTV

StubHub



Disclaimers

- Es nuestra experiencia, no un dogma.
- Es como nosotros lo vemos.
- Os contamos lo que nos ha servido. Experiencia en proyecto XYZ.
- Práctica y escoge lo que te guste o te sirva.



Outside In TDD

- Top Down/London School/Mockist Approach
- Test basados en escenarios requeridos por el cliente
- Recomendado cuando se tiene muy claro la idea general pero los detalles más finos de la implementación no son tan claros.

Outside In TDD

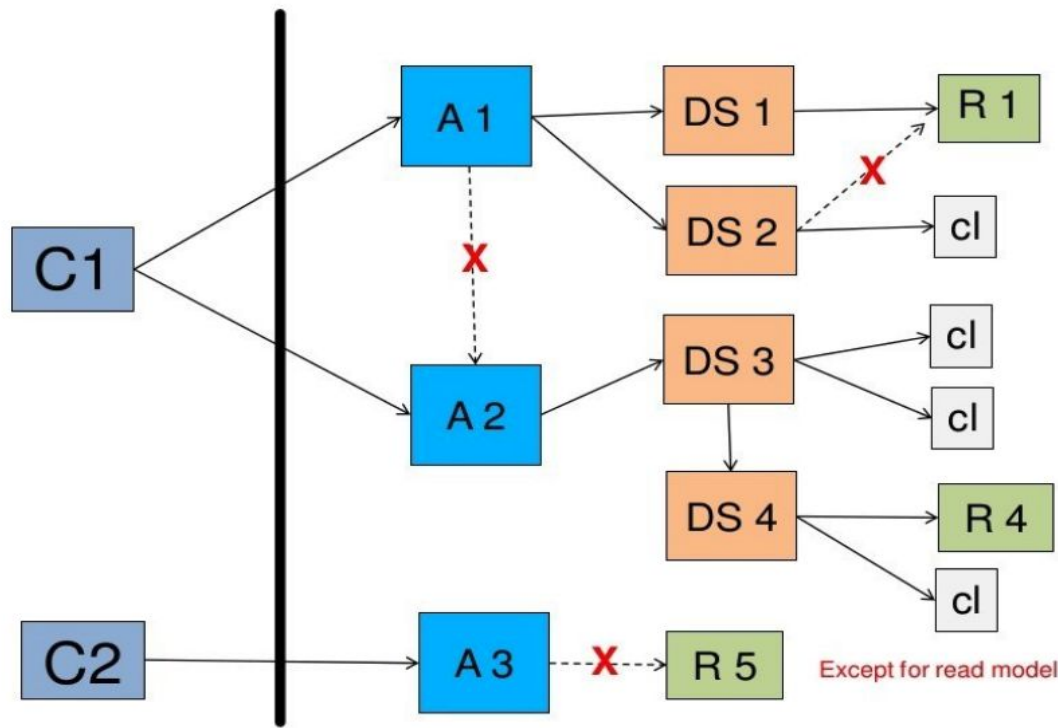
- Nuestra experiencia:
 - Situación (punto de partida) del proyecto XYZ
 - Desconocimiento (el cómo)
 - A nivel de negocio medio claro (puntos gordos) (el qué si, el cómo NPI)
 - A nivel de implementación existe incertidumbre
 - Dependencias con terceros

IDD (Interaction Driven Design)

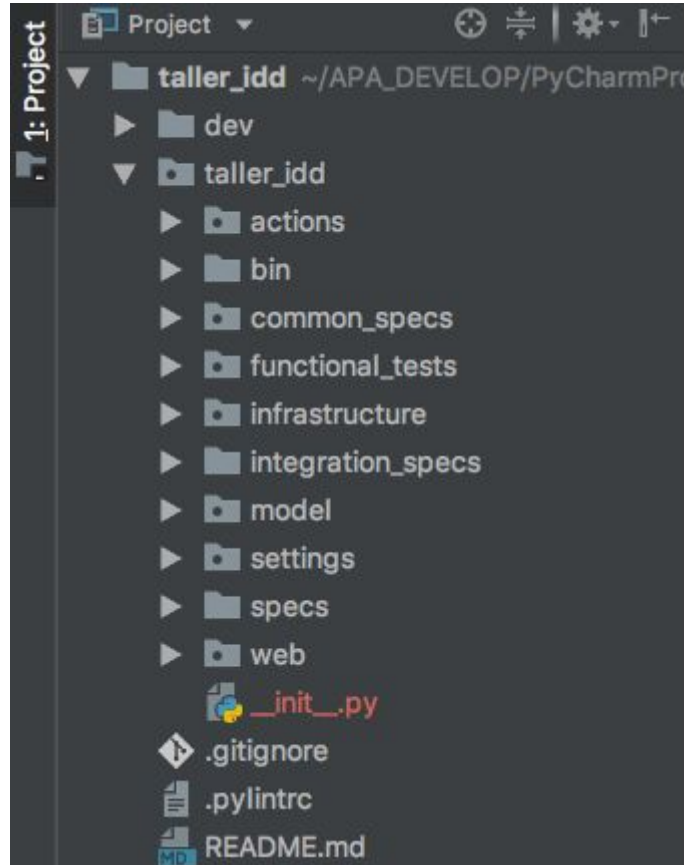
- Propuesta de Sandro Mancuso basada en su experiencia
- Se basa en satisfacer las necesidades de los actores que interactúan con el sistema.
- Facilita:
 - De qué trata el sistema
 - Qué hace el sistema
 - Dónde tengo que hacer un cambio

IDD (Interaction Driven Design)

Domain Model collaborations guideline



IDD (Interaction Driven Design)



IDD (Interaction Driven Design)

- Ventajas
 - ¿Cómo entro al sistema? ¿por dónde empiezo?
 - Organización ficheros, estructura de directorios
 - Pensando en IDD me ha sido más fácil usar OutSideIn, no empezar desde el pozo/zanja

Hemos venido a jugar

- ¿Cómo?
 - En parejas/tríos/etc.
 - 2 Iteraciones (40 minutos + 10 retro)
 - Lenguaje, el que mejor conozcas
 - Framework test, el que mejor conozcas
- **Wifi**
 - Red: **Conferencias**
 - Clave: **ConferenciaS-2011**



El proyecto: CrafterBook

- Una red social para los crafter de Pamplona.
- Puedo darme de alta, seguir a otros crafters y publicar perlas-crafter (mensajes).
- **540Info** quiere leer todos los mensajes publicados.
- **540Info** quiere asegurarse que se cumple el CoC.



El proyecto: CrafterBook

- Consideraciones:
 - Nos olvidamos del delivery-mechanism.
 - Nos centramos en la acciones.
 - Nos centramos en happy-path (tranquilo, no va a fallar).
- Para python 2.x (ejemplo de estructura):

https://github.com/APA42/taller_pamplona_sc_2018



Iteración 1/3

- Para dar de alta un usuario es suficiente su nombre
- Los nombres no se pueden repetir
- Los nombres no pueden ser vacío ni cadena vacía (nil, none, null, "")
- *Opcional*: Puedo seguir a otros usuarios, para hacerlo sólo necesito saber su nombre.
 - No puedo seguir a usuarios que no existan en el sistema

Iteración 1/3 Pistas:

- Las acciones son similares al Patrón Comando (clase con método *execute* o *run*)
- Los test de acciones son de colaboración
- Usa repositorios en memoria o “doblalos”
 - ¿de verdad necesitas repositorios?

Iteración 2/3

- Un usuario puede publicar una perla-crafter, para hacerlo lo único que necesita es su nombre y el mensaje.
 - El crafter debe existir
 - La perla-crafter no puede ser vacío ni cadena vacía (nil, none, null, "")
- Cada vez que se publica una perla-crafter se debe comunicar el crafter y la perla-crafter al **540CNI** (No disponemos de más información pero nos han comentado que puede ser un sms, mms, telegrama, corba, soap, api-rest,)



Iteración 3/3 (bola extra)

- Cuando se publica una perla-crafter, si el contenido del mismo incumple el CoC, se debe notificar a **540PoliceOffice** indicando el crafter y la perla-crafter publicada.
- No conocemos el CoC
- No sabemos cómo se va a notificar a **540PoliceOffice**. Podría ser un sms, mms, telegrama, corba, soap, api-rest, exchange en rabbitmq....)

Conclusiones

- Proporciona de una forma más clara:
 - De qué trata el sistema (qué)
 - Qué hace el sistema (cómo)
 - Por dónde entro al sistema
 - Dónde tengo que tocar para realizar un cambio
- Nos es más fácil realizar OutSide In

Referencias

- Python Valencia - Interaction Driven Design (@elreplicante @python_vlc)
 - <https://www.youtube.com/watch?v=n3hOS0Cj5Mc>
 - <http://slides.com/elreplicante/deck#/>
- <https://codurance.com/2017/12/08/introducing-idd/>
- <https://8thlight.com/blog/georgina-mcfadyen/2016/06/27/inside-out-tdd-vs-outside-in.html>

