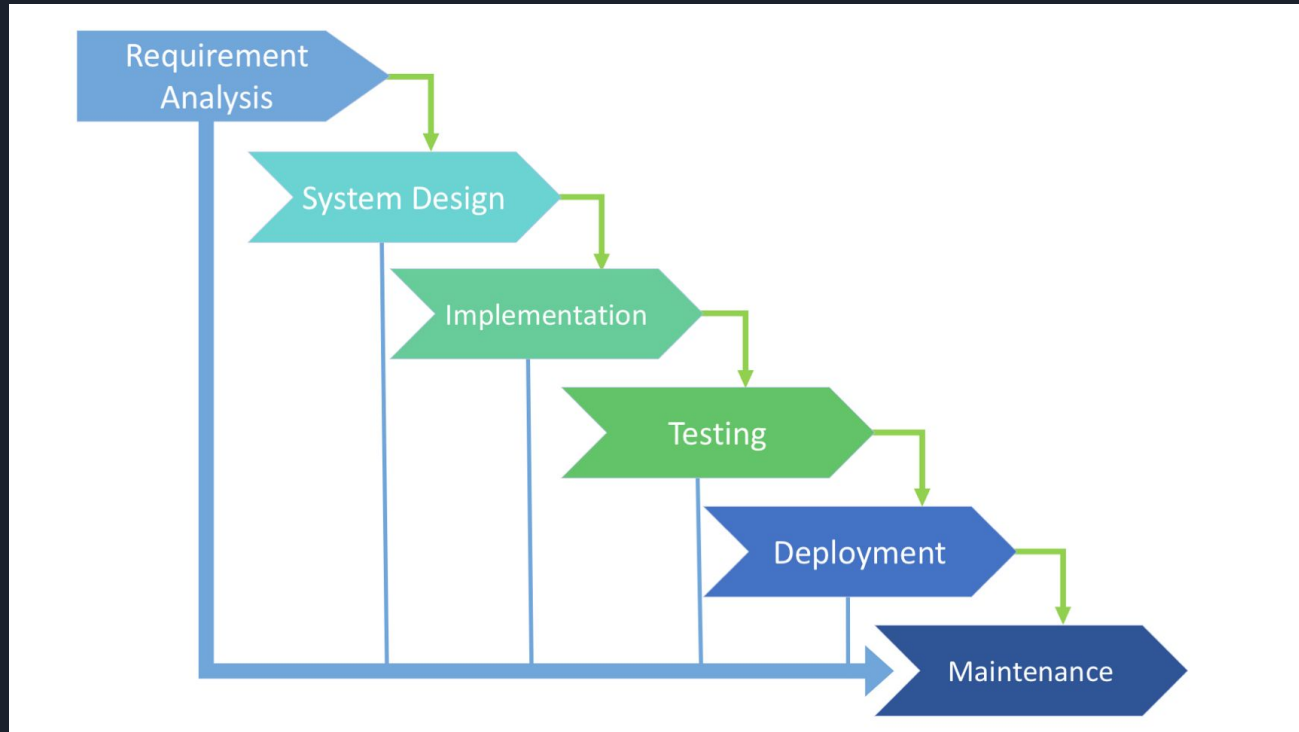
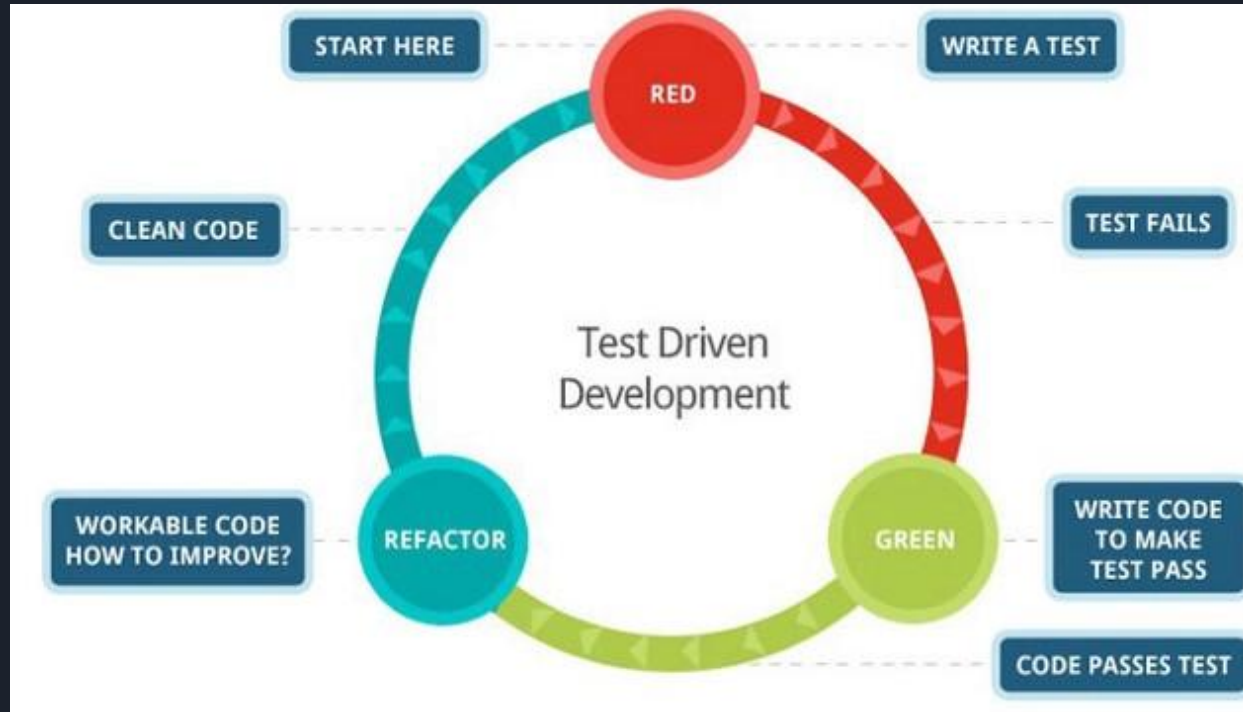


TDD: Uso luego implemento

Cascade development



TDD: Test Driven Development





Conclusión?





TDD: Ventajas





TDD: Ventajas



- Focalizado en la funcionalidad, abstrayéndose del todo



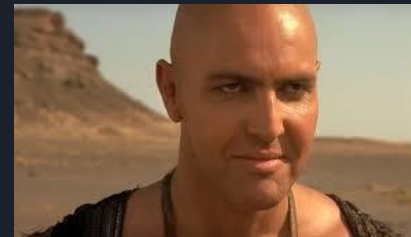
TDD: Ventajas



- Focalizado en la funcionalidad, abstrayéndose del todo
- Reduce los refactors



TDD: Ventajas



- Focalizado en la funcionalidad, abstrayéndose del todo
- Reduce los refactors
- Agiliza el desarrollo



TDD: Ventajas



- Focalizado en la funcionalidad, abstrayéndose del todo
- Reduce los refactors
- Agiliza el desarrollo
- Simplifica el diseño



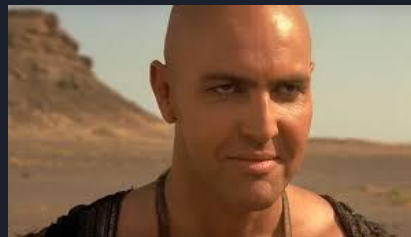
TDD: Ventajas



- Focalizado en la funcionalidad, abstrayéndose del todo
- Reduce los refactors
- Agiliza el desarrollo
- Simplifica el diseño
- Contribuye a no tener componentes innecesarios



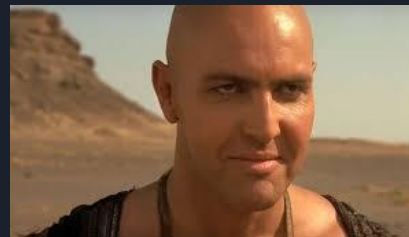
TDD: Ventajas



- Focalizado en la funcionalidad, abstrayéndose del todo
- Reduce los refactors
- Agiliza el desarrollo
- Simplifica el diseño
- Contribuye a no tener componentes innecesarios
- Ayuda a resolver el problema de la hoja en blanco



TDD: Ventajas



- Focalizado en la funcionalidad, abstrayéndose del todo
- Reduce los refactors
- Agiliza el desarrollo
- Simplifica el diseño
- Contribuye a no tener componentes innecesarios
- Ayuda a resolver el problema de la hoja en blanco
- Mejora la **DX (Developer Experience)**, da visibilidad al progreso



TDD

Entonces todo OK?



TDD: Entonces todo OK?

Casi

TDD: Desventajas





TDD: Desventajas



- Riesgo de perder la visión global del sistema



TDD: Desventajas



- Riesgo de perder la visión global del sistema
- Problema de la hoja en blanco



TDD: Desventajas



- Riesgo de perder la visión global del sistema
- Problema de la hoja en blanco
- Al principio **todo falla**



TDD: Desventajas



- Riesgo de perder la visión global del sistema
- Problema de la hoja en blanco
- Al principio **todo falla**
- Curva de aprendizaje “rústica”



TDD: Desventajas



- Riesgo de perder la visión global del sistema
- Problema de la hoja en blanco
- Al principio **todo falla**
- Curva de aprendizaje “rústica”
- Los test tienen benzoato de potasio



TDD

Como se usa?



TDD: Como se usa?

Agile

TDD con Agile o Agile con TDD?





Agile

User Story



Agile

User Story?



User Story

- Requisito mínimo y concreto



User Story

- Requisito mínimo y concreto
- Estimable (números primos, romanos, cartas)



User Story

- Requisito mínimo y concreto
- Estimable (números primos, romanos, cartas)
- Independiente del resto



User Story

Como se usa?



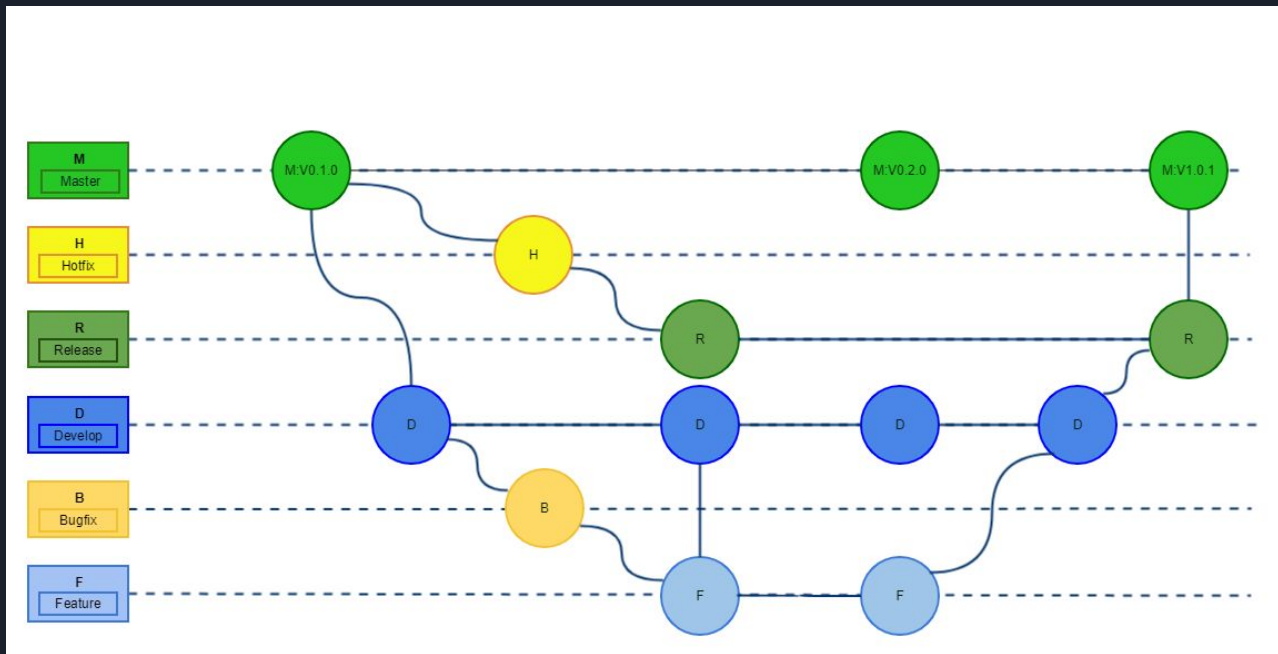
User Story: Como se usa?

Feature Branch




Ufff

Feature branch





User Story: Como se usa?

User Story  Branch

Y entonces como uso TDD???



TDD: Como lo uso?

1. Agarras un user story



TDD: Como lo uso?

1. Agarras un user story
2. Creas un branch



TDD: Como lo uso?

1. Agarras un user story
2. Creas un branch
3. Haces un test para el caso más genérico del US



TDD: Como lo uso?

1. Agarras un user story
2. Creas un branch
3. Haces un test para el caso más genérico del US
4. Lo corres



TDD: Como lo uso?

1. Agarras un user story
2. Creas un branch
3. Haces un test para el caso más genérico del US
4. Lo corres
5. Falla

TDD: Como lo uso?





TDD: Como lo uso?

1. Agarras un user story
2. Creas un branch
3. Haces un test para el caso más genérico del US
4. Lo corres
5. Falla
6. Implementas (si, es lo último)



TDD: Como lo uso?

1. Agarras un user story
2. Creas un branch
3. Haces un test para el caso más genérico del US
4. Lo corres
5. Falla
6. Implementas (si, es lo último)
7. Lo corres



TDD: Como lo uso?

1. Agarras un user story
2. Creas un branch
3. Haces un test para el caso más genérico del US
4. Lo corres
5. Falla
6. Implementas (si, es lo último)
7. Lo corres
8. Error? Paso 6, OK? Paso 9



TDD: Como lo uso?

1. Agarras un user story
2. Creas un branch
3. Haces un test para el caso más genérico del US
4. Lo corres
5. Falla
6. Implementas (si, es lo último)
7. Lo corres
8. Error? Paso 6, OK? Paso 9
9. `git add / commit / pull / push origin feature_branch`



TDD: Como lo uso?

1. Agarras un user story
2. Creas un branch
3. Haces un test para el caso más genérico del US
4. Lo corres
5. Falla
6. Implementas (si, es lo último)
7. Lo corres
8. Error? Paso 6, OK? Paso 9
9. `git add / commit / pull / push origin feature_branch`
10. Te tomas un cafe



Dudas???

Spotipy





Spotipy

Usemos las US que alguien en algún lugar armo

Spotipy: US

- usuari@ sin cuenta se tiene que poder registrar con un email no usado
- usuari@ con cuenta se loguea
- usuario@ debe registrarse con clave segura
- usuari@ con cuenta sube un tema
- usuari@ con elimina uno de sus temas
- usuari@ obtiene informacion de un tema
- usuari@ califica un tema
- usuari@ crea un playlist
- usuari@ elimina un playlist
- usuari@ agrega tema a un playlist
- usuari@ elimina tema a un playlist
- usuari@ obtiene los temas de un playlist

Spotipy: US

- usuari@ sin cuenta se tiene que poder registrar con un email no usado

Spotipy: US

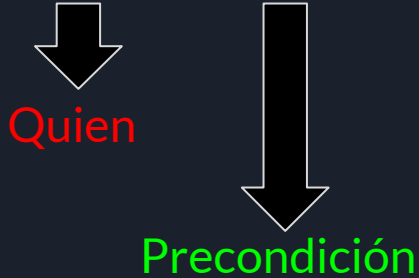
- **usuari@** sin cuenta se tiene que poder registrar con un email no usado



Quien

Spotipy: US


- `usuari@sin cuenta` se tiene que poder registrar con un email no usado




Spotipy: US

- **usuari@sin cuenta** se tiene que poder registrar con un email no usado


Quien



Precondición



Acción / resultado

Spotipy: US

- **usuari@sin cuenta** se tiene que poder registrar **con un email no usado**


Quien


Precondición


Acción / resultado


Precondición

Vamos a codear no?

