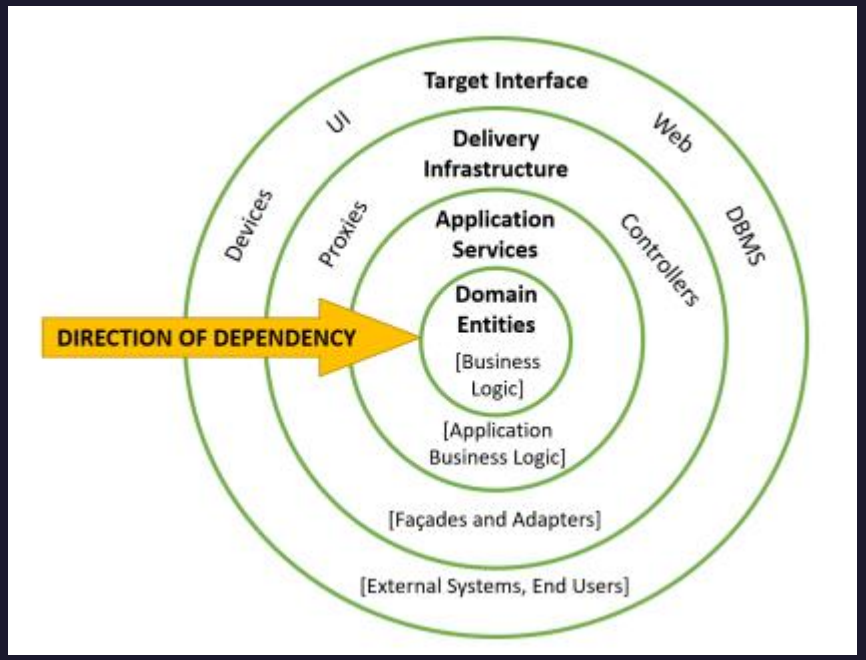
A New Approach

My experience revisiting a coding challenge after learning outside-in TDD and hexagonal architecture

Sean Jennings

Recapping Outside-In TDD

- Also known as acceptance test driven design
- Identify key responsibilities upfront
- Use mocks to sketch design of system
- Takes place during the red phase of TDD
- Less mess than classic TDD
- Favours assertion of mocks over state-based assertions
- Naturally follows directions of dependencies



The challenge

- Create a console application (no UI necessary) for a game in which one player moves a single piece around a board.
- The board is 8 by 8 in size
- A player starts on the bottom-left square
- A player can move their piece up, down, left or right one square by entering U, D, L, or R respectively
- A random number of squares are ‘landmines’
- The aim of the game is to get to the top side of the board without hitting more than 2 landmines
- Following each move the game displays the new location and whether any landmines were hit, plus whether the game has been lost (more than 2 land mines hit) or won (reached the top of the board)

The Original Approach

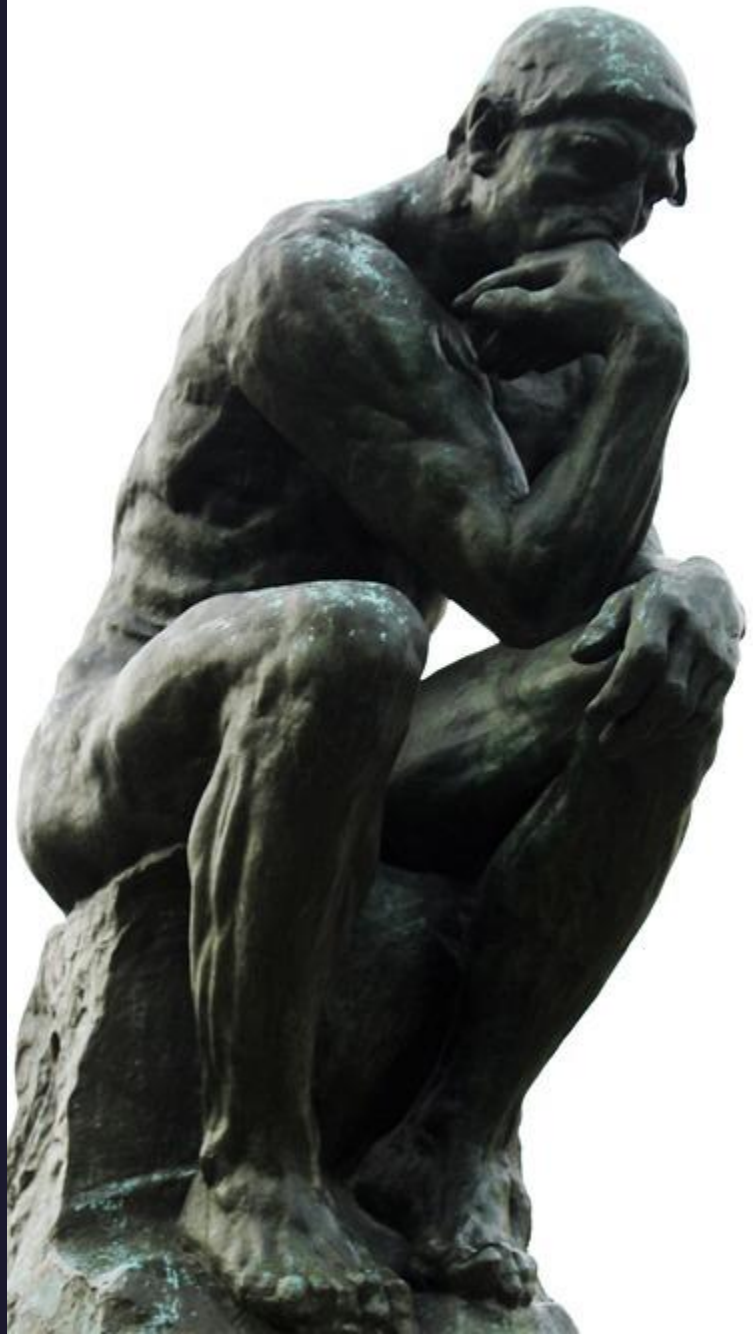
- Not test-driven development
- Tests not focused on behaviour
- Procedural
- No single-responsibility
- No clear architecture

“The way to get started is to quit talking and begin doing.”

Walt Disney

Considerations

- No external dependencies – no integration tests needed
- Requires random generation of mines – not deterministic
- Mocking at every layer results in implementation focused tests that can be fragile
- Challenge does not warrant deep layers of dependencies



The Acceptance Tests

Happy Scenario

Given there are no mines

And a game is in play

When the player moves to the top of the board

Then the player wins the game

Sad Scenario

Given the board is full of mines

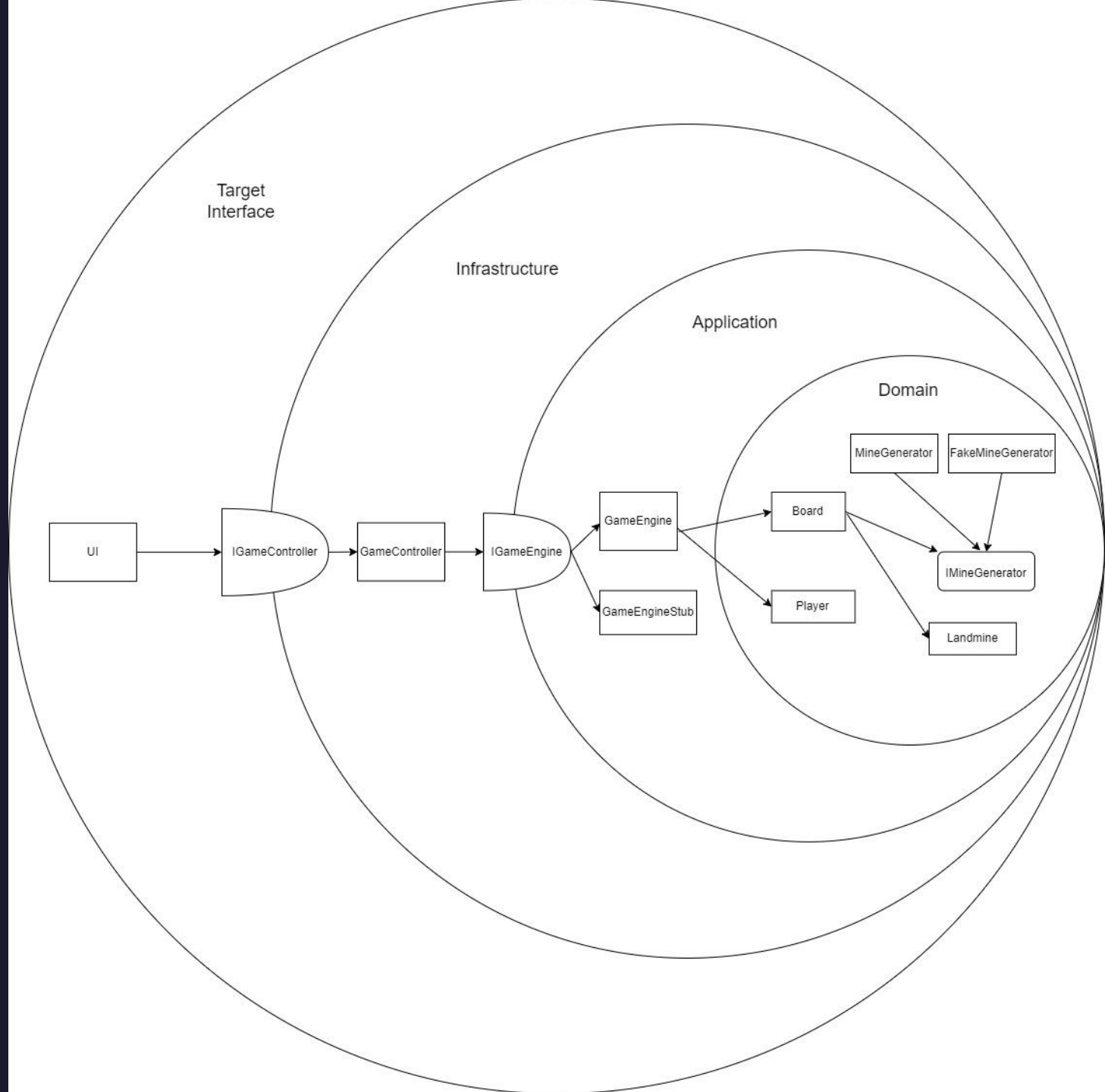
And a game is in play

When the player moves three steps

Then the player loses the game

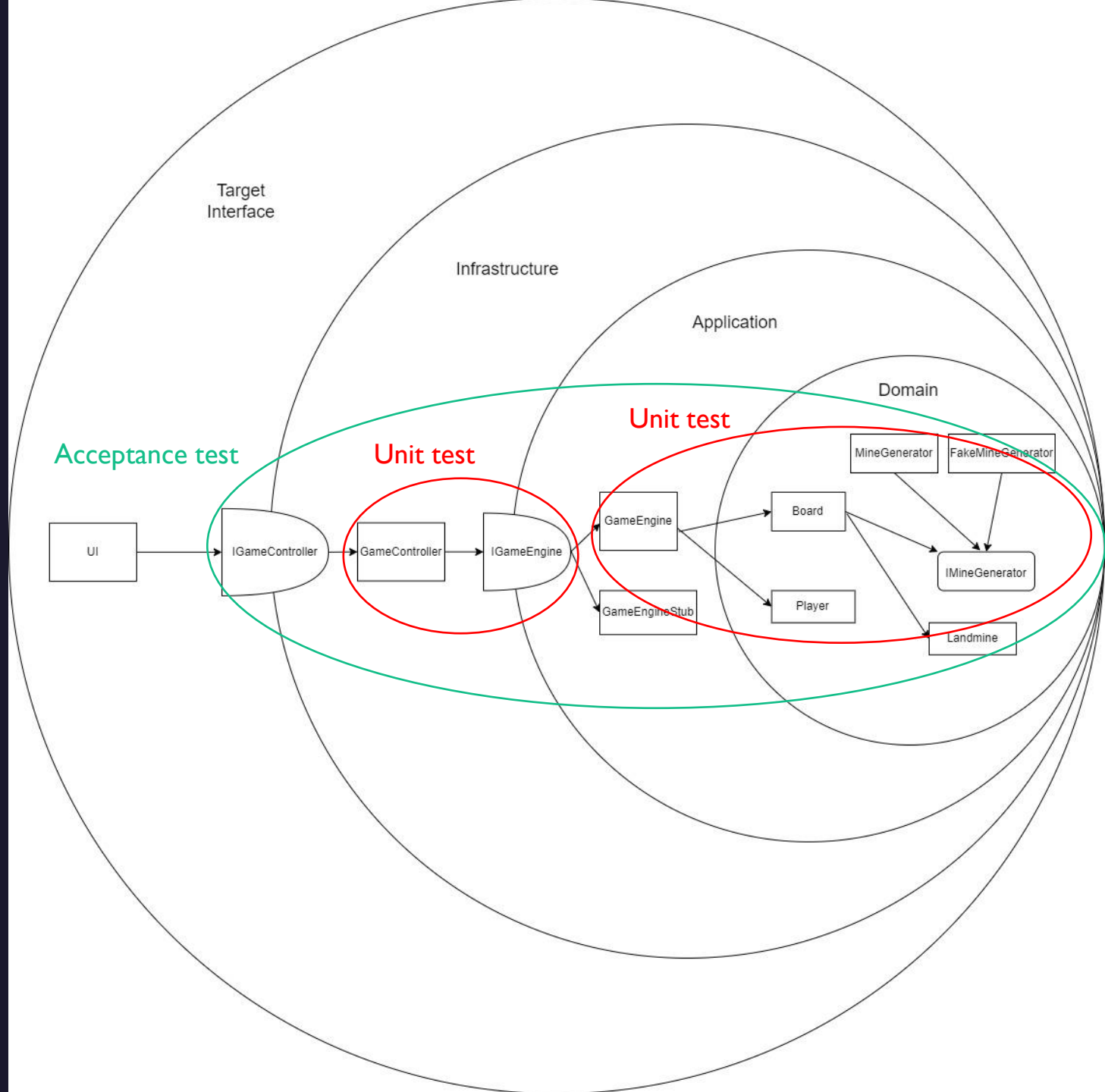
The Design

- Hexagonal architecture with driving-side only
- Ports between layers
- Mine generation taken out of board



The Test Approach

- Decided to take outside-in approach of sketching layers but…
- Chose classic-TDD state-based assertions over mock assertions once inside application layer
- Unit tested application layer and domain layer together without mocking internal dependencies
- …except mine creation which was faked



Summary

- Hexagonal architecture promoted separation of responsibilities and easier testing
- Acceptance test driven design produced a much cleaner and extensible design

Questions?

References

- Alcor Academy - Agile Technical Practices Distilled
- The game - https://github.com/CuriousConsistency/MineGame

Thank You

sean.jennings@fdbhealth.com

linkedin.com/in/sean-martin-somerset