# Quality of Software

What can we do about it?

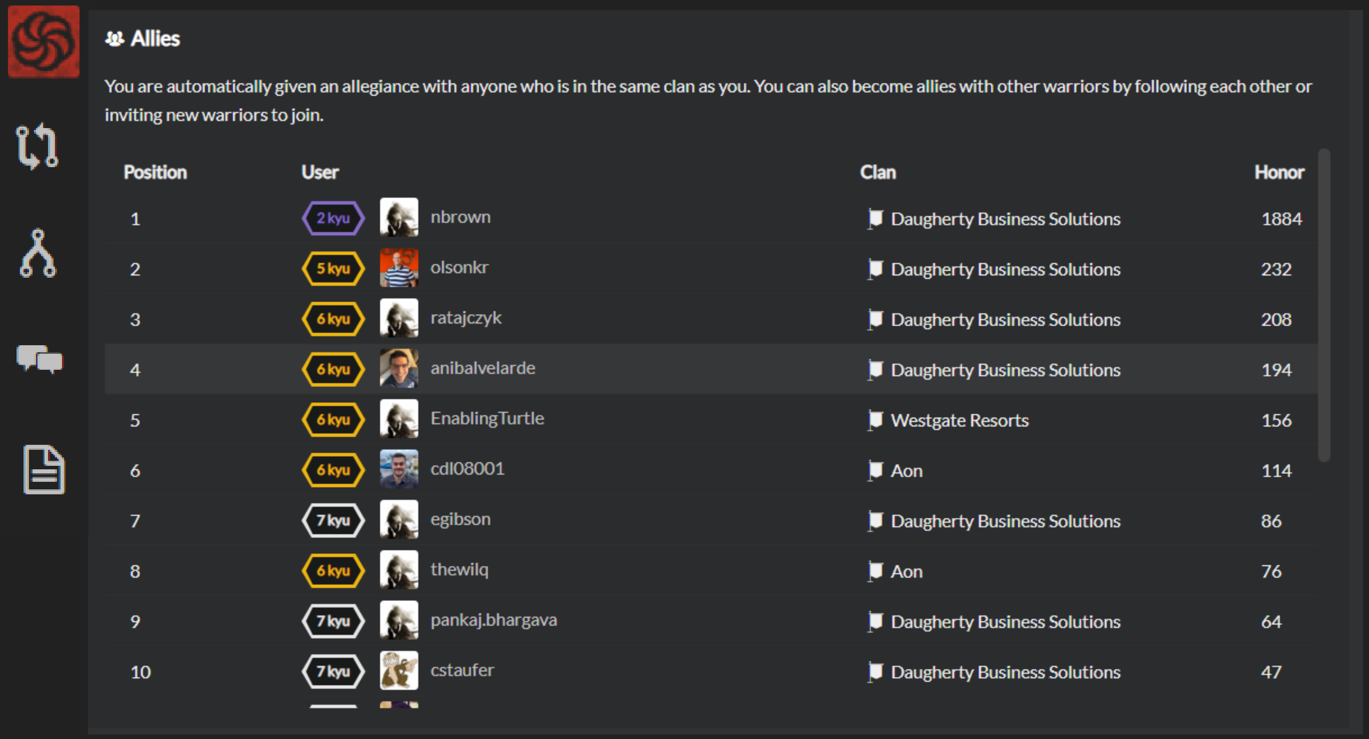
# Who is responsible for the quality of the software you produce?

You are!

# How can you get better at producing quality software?

- Practice
- Practice
- Practice
- Practice some more…

- www.CodeWars.com
- There are other alternatives



**Allies**

You are automatically given an allegiance with anyone who is in the same clan as you. You can also become allies with other warriors by following each other or inviting new warriors to join.

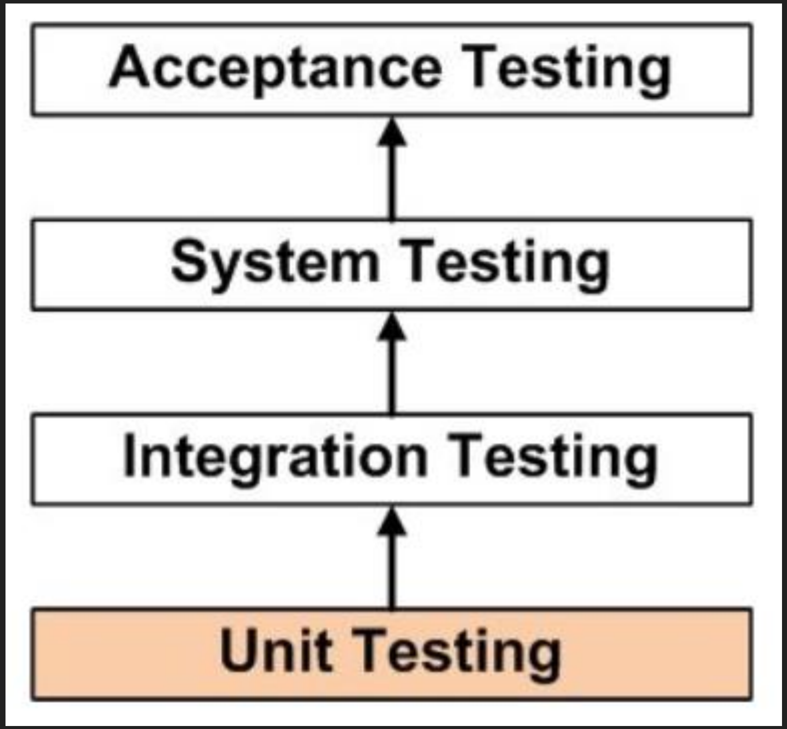
| Position | User | Clan | Honor |
|---|---|---|---|
| 1 | 2 kyu nbrown | Daugherty Business Solutions | 1884 |
| 2 | 5 kyu olsonkr | Daugherty Business Solutions | 232 |
| 3 | 6 kyu ratajczyk | Daugherty Business Solutions | 208 |
| 4 | 6 kyu anibalvelarde | Daugherty Business Solutions | 194 |
| 5 | 6 kyu EnablingTurtle | Westgate Resorts | 156 |
| 6 | 6 kyu cdl08001 | Aon | 114 |
| 7 | 7 kyu egibson | Daugherty Business Solutions | 86 |
| 8 | 6 kyu thewilq | Aon | 76 |
| 9 | 7 kyu pankaj.bhargava | Daugherty Business Solutions | 64 |
| 10 | 7 kyu cstaufer | Daugherty Business Solutions | 47 |

# How to level-up your skills with Software Quality?

- Understand the common sources of software problems
  - Code Smells
- Understand best practices to fix them
  - Refactor
- My tooling of choice for this is
  - Refactoring Guru

# What do you do when you find bug in your code? - a happy path -

- Ideally you:
  - Appropriate testing library
  - Create a test to reproduce/expose the bug
  - Change your code to make your test **pass!**



Acceptance Testing

System Testing

Integration Testing

Unit Testing

# Characteristics of Unit Tests

- Focus on small fragments of logic at a time
- Execute fast ~ average 20 milliseconds per test
- Have clear structure
  - // arrange
  - // act
  - // assert
- Help you determine usability of your code
- Assist with refactoring
- Dev team must be able to trust a Unit Test library's accuracy

# Mocks - Complements to Unit Testing

- What are mocks?
- Why mock?
- What are some common mocking frameworks?
  - JavaScript
  - Java
  - .NET

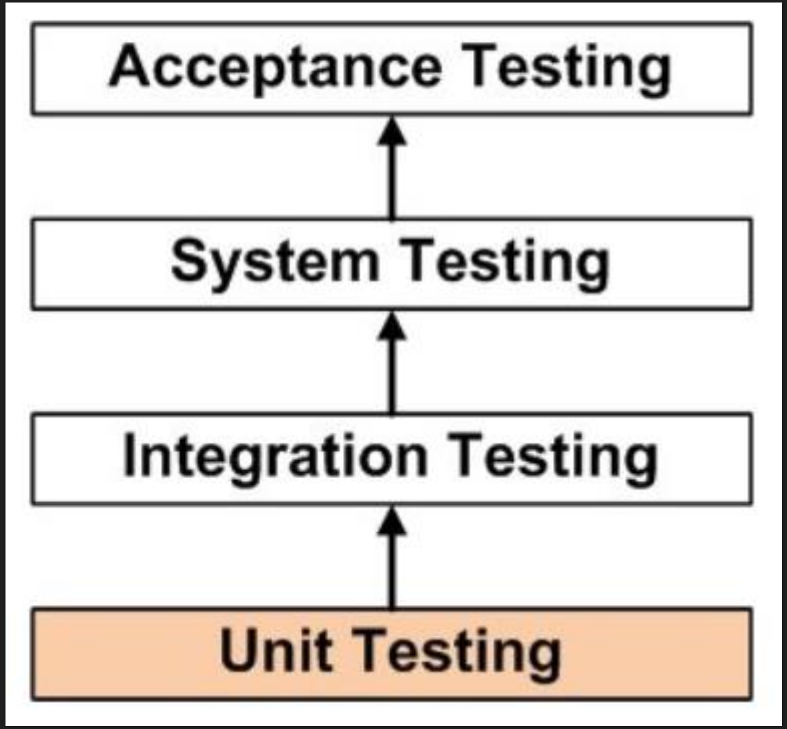
- A lot of the time, it will depend on what our clients work with.

# Demo Time…

- Let’s check out some examples:
  - Microsoft’s VS Code
    - https://github.com/microsoft/vscode/blob/master/src/vs/editor/test/common/services/modelService.test.ts#L39
  - Oracle’s Database Programming Interface for C/C++
    - https://github.com/oracle/odpi/blob/master/test/TestNumbers.c#L270
  - Anibal’s **bitwise-stor** JavaScript Library
    - https://github.com/anibalvelarde/bitwise-stor/blob/master/src/index.test.js#L77

# How about the other types of test?

- A good set of definitions can be found at:
  - www.softwaretestingfundamentals.com



# Difficulties with testing adoption

- Some times those who pay for a Software project have reservations
  - Overheard somewhere, at a software development shop, right at this minute:
    - "You mean you need to create *non-feature* code to finish the *feature* code?"
- The process seems, at first glance, counter intuitive
- It requires a mindset change at various levels
  - Personal
  - Team
  - Org-wide

- Anyone has some stories to share?

# A word about "Education"

- "**Education**" comes from a couple of Latin Root Words
  - ***Educare*** – To train or to mold
  - ***Educere*** – To lead forth

- The second one raises an interesting question:  Which way is *forth*?

- As consultants, we are called to lead forth.  We need to make sure we understand which way is forth.

# Which way is forth?

- Learn to discern Truth
- Learn to discern what is Good

- Where those 2 intersect, there you will find Beauty

- In tech, learning to discern between Principles and Styles is key