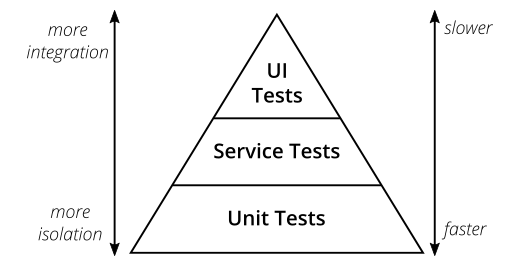
**Testing**

Testing is vital within software engineering for ensuring a program fulfils its’ intended purpose. Testing enables the programmer to have confidence that his future changes have not caused issues with work prior.

Some general guidelines for ensuring code quality through testing are as follows:

* Test one thing at a time. Tests should be as granular as possible. This ensures that what you are testing will not be disrupted by other methods which are the cause of error. Testing should begin with the most granular section, with the least dependencies. 
* Be diligent towards testing edge cases: This is done because edge cases are the most common culprit resulting in errors. An example of an edge case is having an exam result where if <= 40 would result in a fail and the provided result was 40, declaring the grade as a pass.
* Ensure that all developers involved in the feature being tested are aware of the full scope of how this feature needs to work and how it will integrate with the wider system. This provides greater frame of references for people to debug future issues arising as a result of any unsuspected errors resulting from this feature.
* Make sure to do quick sanity checks before kicking off a long-winded regression test. These should also be done at night so they don’t eat into your working hours.

Testing granularity: <https://martinfowler.com/articles/practical-test-pyramid.html>

Edge Cases: <https://www.mindfulqa.com/edge-cases/>

Team Cohesion: <https://www.indeed.com/career-advice/career-development/team-cohesiveness>

Sanity testing vs Regression Testing:

<https://www.testgrid.io/blog/sanity-testing-vs-regression-testing/>

<https://www.geeksforgeeks.org/difference-between-sanity-testing-and-regression-testing/>