

Acceptance tests

Integration tests

Unit tests

The whole team

Retros Against Humanity | Improvement cards

Retros Against Humanity | Improvement cards

Retros Against Humanity | Improvement cards

Retros Against Humanity | Improvement cards

The business analyst

The test analyst

A developer

Retros Against Humanity | Improvement cards

Retros Against Humanity | Improvement cards

Retros Against Humanity | Improvement cards

_____ should write the acceptance criteria.

Retros Against Humanity | Good cards

_____ should write the unit tests.

Retros Against Humanity | Good cards

_____ should write the integration tests.

Retros Against Humanity | Good cards

_____ should write the acceptance tests.

Retros Against Humanity | Good cards

You can use _____ tests to test detailed logic.

Retros Against Humanity | Good cards

You can use _____ tests to test edge cases.

Retros Against Humanity | Good cards

Your _____ tests will become brittle if you focus too much on UI specifics.

Retros Against Humanity | Good cards

There should be more _____ than any other tests.

Retros Against Humanity | Good cards

_____ should decide what to test.

Retros Against Humanity | Good cards

_____ are testing at the highest level.

Retros Against Humanity | Good cards

You can use _____ tests to test the inputs and outputs of a particular service.

Retros Against Humanity | Good cards

You can use _____ tests to test the correct inter-operation of subsystems.

Retros Against Humanity | Good cards

_____ are the most important tests.

Retros Against Humanity | Good cards

_____ are testing at the lowest level.

Retros Against Humanity | Good cards

We should run the _____ tests every time we build our code.

Retros Against Humanity | Good cards

The slowest tests are likely to be _____ tests.

Retros Against Humanity | Good cards