

Levels of Testing

Example: Onlinedegree dashboard

Initial requirements gathering

- Who are the stakeholders?
 - Students - log in and see courses
 - Admins - manage students
 - Teachers - update / manage course material ?
- Functionality
 - Each group has different needs
- Non-functional requirements
 - Page colour, font, logo



MY DASHBOARD

- Latest Updates
- Exam Cities and Hall Tickets
- My Current Courses
- **Completed & Pending Courses**
- Academic Calendar
- Documents for Download
- Submitted Documents

Reporting harassment: IITM BSc Degree
Team is committed to ensuring that
everyone is equally valued.... [Read
More](#)

Foundational Level Courses

COMPLETED COURSES

OTHER COURSES

Example: Student page

Functional

- Latest updates
- Register exam hall preferences
- Download hall ticket
- Update course registration
- View completed courses
- ...

Example: Student page

Functional

- Latest updates
- Register exam hall preferences
- Download hall ticket
- Update course registration
- View completed courses
- ...

Non-functional

- Header / Footer colours
- Copyright notice and extra information
- Logo
- Fonts
- ...

Requirements gathering

- Extensive discussions with end-users required
- Avoid language ambiguity
- Capture use cases and *examples*
- Start thinking about test cases and how the requirements will be validated

Units of Implementation

- Break functional requirements down to small, implementable *units*
- Examples:
 - view course list
 - edit course status
 - edit exam preferences
 - download completion certificate
- Each one may become a single controller
 - May also combine multiple into a single controller

Unit Testing

- Test each individual unit of implementation
- May be single controllers
 - May even be part of a controller
- Clearly define inputs and expected outputs
- Testable in isolation?
 - Can each unit be tested without the entire system?
 - Create artificial data set to check whether a single update works

Example: Unit tests

Student registers for a course

- Create dummy DB:
 - One student
 - One course
- Test
 - Controller to add course for student
 - Form to be displayed
 - Invalid student ID, course ID - error codes?
 - Add student more than once?

Integration

- Application consists of multiple modules:
 - Student management
 - Course management
 - Payment interfaces
 - Admin interface

Integration Testing

- Example of integration:
 - Student + Payment gateway
 - Student + Course + Admin
 - All of the above . . .
- Potential problems:
 - Individual units work - combined system does not
 - Dependencies violations in server - redesign?
- Continuous integration
 - Combined with version control systems: CI
 - Each commit to main branch triggers a re-evaluation of integration tests
 - Multiple times a day possible

System-level Testing

- One step beyond integration
- Includes server, environment
- Mainly black-box: should validate final usage

Example: onlinedegree

- Deploy on final environment: Google app-engine
- Test domains used
- Confirm all aspects of behaviour
- Non-functional tests:
 - Performance under load
 - Number of instances, scaling
 - Cost!

System testing Automation

- Has to simulate actual user interaction
- Browser automation frameworks
 - Selenium (example)
- Includes database, persistent connections etc.
- Typically a complete secondary system

User Acceptance Testing

- Deploy final system
- Tested by restricted set of users - pilot
- “Beta” testing
 - Beta- software: pre-production