How Specification by Example and Test-Driven-Development Help to Avoid Technical Debt

Wolfgang Trumler,
Frances Paulisch

Corporate Technology,
Germany
Can we avoid Technical Debt?
NO!
Do we want to avoid Technical Debt?
It depends!
Quadrants of Technical Debt by Martin Fowler

"We don't have time to discuss the requirements in such a detail."

"What's that feature all about?"

"We don’t have time for design”

“What’s Layering?”

“Now we know how we should have done it”

“We must ship now and deal with consequences”
How technical debt is introduced

- Add additional cases
- System Test
- Apply quick fix
- Change behavior
How to address these issues

Specification by Example

Acceptance Tests

Unit Tests
Specification by Example

Key Concepts

Examples → Clarify → Requirements

Become → Tests

Validate
Encoding Directions in Patient Space

A coronal S.G. (in the same view orientation, unrotated)

Shows readout in patient's right-left, phase in patient's feet-head.

The other orientations change accordingly; see table below.

<table>
<thead>
<tr>
<th>S.G. Orientation</th>
<th>Readout Encoding Direction</th>
<th>Phase Encoding Direction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coronal</td>
<td>Right → Left</td>
<td>Feet → Head</td>
</tr>
<tr>
<td>Sagittal</td>
<td>Anterior → Posterior</td>
<td>Feet → Head</td>
</tr>
<tr>
<td>Transverse</td>
<td>Right → Left</td>
<td>Post → Ant</td>
</tr>
</tbody>
</table>

Coronal

Sagittal

Transverse
Acceptance Testing with Cucumber

**Feature:** AddIn wants to load images to the planning and stamp segments

During a scan workflow on AddIn may want to load multiple images to planning or stamp segments. The AddIn can specify a list of images - segment combinations to be loaded into the GSP.

In contrast to the regular image loading functionality, where images are shown as soon as they are loaded, the images requested by the AddIn should all be shown at the same time after all images have been loaded.

To load an image to a planning segment (left, middle, right, all) the AddIn has to specify the sopInstanceUid and the frame number of the image. When multiple images are to be loaded in the same planning segment the last one defined is actually shown.

To load an image to a stamp segment the seriesUid and the position in the stamp segment has to be defined by the AddIn. The position starts with 0 to load the series as the top most. Position 1 is after the first series. Position -1 appends the series at the end.

**Remark:**
The default image loading handling into empty segments is specified else where.

**Scenario:** AddIn wants to load images in different segments

<table>
<thead>
<tr>
<th>frameset</th>
<th>image index</th>
<th>planning segment</th>
</tr>
</thead>
<tbody>
<tr>
<td>first</td>
<td>1</td>
<td>left</td>
</tr>
<tr>
<td>second</td>
<td>2</td>
<td>middle</td>
</tr>
<tr>
<td>third</td>
<td>1</td>
<td>right</td>
</tr>
</tbody>
</table>

Then the left planning segment shows image 1 of the first frameset

And the middle planning segment shows image 2 of the second frameset

And the right planning segment shows image 1 of the third frameset
Unit Testing

[Test]
public void Handle_ShouldCallApplyChanges()  
{
    // Arrange

    // Act
    m_ParameterCardSync.Handle(new ApplyParameterCardEvent());

    // Assert
    m_ApplyCheck.Verify(m => m.ApplyChanges());
}

[Test]
public void Handle_ShouldNotCallApplyChanges_WhenApplyFuncIsNull()  
{
    // Arrange
    m_ParameterCardSync = new ParameterCardSync(m_EventAggrator.Object, () => null);

    // Act
    m_ParameterCardSync.Handle(new ApplyParameterCardEvent());

    // Assert
    m_ApplyCheck.Verify(m => m.ApplyChanges(), Times.Never);

Addressing the Quadrants of Technical Debt

Reckless

"We don’t have time for design"

Prudent

"We must ship now and deal with consequences"

Deliberate

Inadvertent

"What’s Layering?"

“Now we know how we should have done it"
## Defect Rate

### Defects / 1k Lines of Code

<table>
<thead>
<tr>
<th></th>
<th>In-house Testing</th>
<th>Released Product</th>
</tr>
</thead>
<tbody>
<tr>
<td>Industry Average</td>
<td>15 - 50</td>
<td></td>
</tr>
<tr>
<td>Microsoft Applications</td>
<td>10 – 20</td>
<td>0.5</td>
</tr>
<tr>
<td>Cleanroom development</td>
<td>3</td>
<td>0.1</td>
</tr>
</tbody>
</table>


### In-house testing (not completed yet)

- All reported: 7.42
- Related to our component: 4.64
The Role of the Architect

Up-front Design → Design → Refactor
Summary

We applied

- Specification by Example
- Acceptance and extensive Unit Testing
to avoid technical debt in late phases of the project

Used Defect Rate as performance indicator

Methodology mix worked well for us (very low Defect Rate)

Planning to roll-out on broader scale
Thank you!

Wolfgang Trumler
Frances Paulish
Corporate Technology
Germany