



WRITING HIGH QUALITY MENDIX APPS

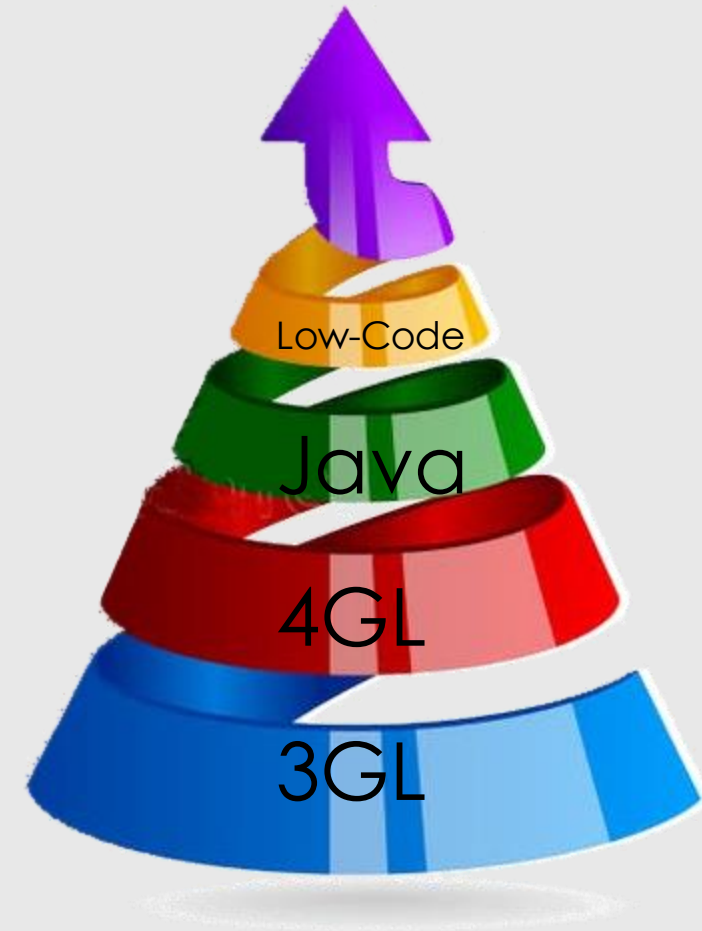
And proving it with SAT!

Contents

- Introduction
- Stoneworx Assessment Tool
- What is good quality
- Why another tool?
- SAT
- Demo
- Further development

Introduction

- In IT since 1994
- 4 years of which in Mendix roles
- 1.5 year for Stoneworx
 - First months: no paid project +
 - Upskill trainee code assessments +
 - Wanted to learn about MX SDK's



Stoneworx Assessment Tool

- First take: Quality Assessment Tool
- Evolution:
 - Insight in quality issues
 - Security
 - Microflow structures

What is good quality code?

- Above all: Subjective as can be
- What you and your peers agree upon
 - [Mendix community](#)
 - Your company / the company you work for
 - Your project team

So we need a flexible set of quality controls that reflect the common ground of the project environment

Why another tool

- This one is free / [open source](#)
- it is adaptable
- It runs local: direct feedback
- It can scan both local mpr files or on team server
- It installs easy: docker or nodejs
- It will get better and better



Stoneworx Assessment Tool: Quality

Improve quality:

- Reliability
- Security
- Maintainability
- Project Hygiene

Hygiene	NC1	format: [PRE]_[Entity(s)]_description
Hygiene	NC2	Prefix must be allowed
Hygiene	NC3	Entity must exist
Hygiene	NC4	Entity must exist in same module
Maintainability	CM1	Commit not on correct hierarchy level(ACT or one level down)
Maintainability	IP1	Show Page action outside of ACT
Maintainability	IP2	Close Page action outside of ACT
Maintainability	ND1	Nesting of subs too deep
Maintainability	CX1	Too many actions in a single microflow
Maintainability	CX2	Too complex microflow
Maintainability	CX3	Too complex expression in Create/ Change Object
Maintainability	CX4	Too complex expression in Create / Change Variable
Reliability	EH1	Java Action without custom error handling
Security	DU1	Demo users not allowed in production app
Security	PM1	Microflow of this type should contain permissions

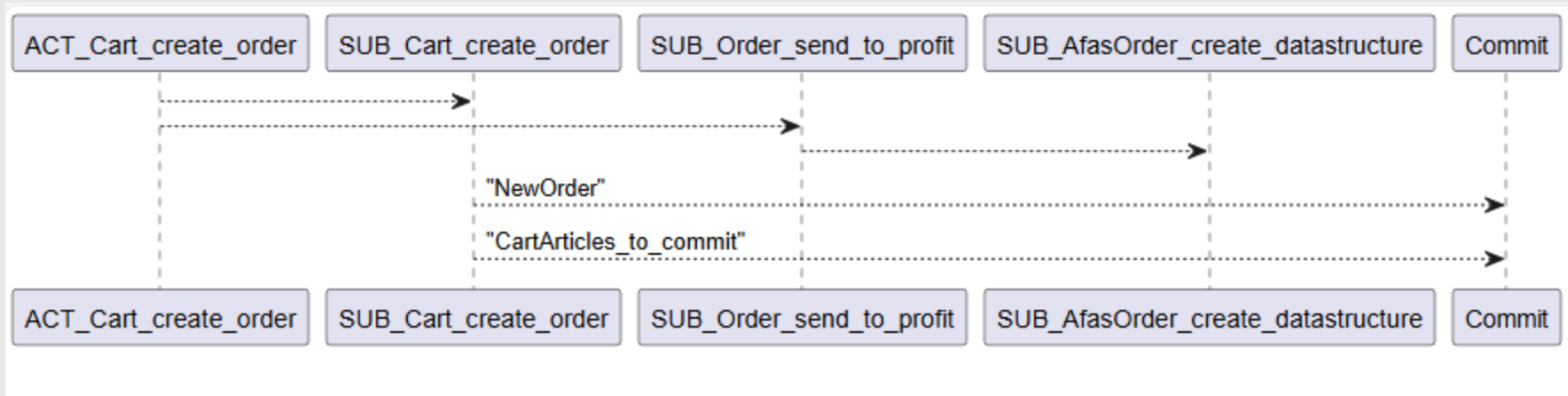
Stoneworx Assessment Tool: Security

Export a csv with a access rules per Entity/Attribute. Create your own reports from there...

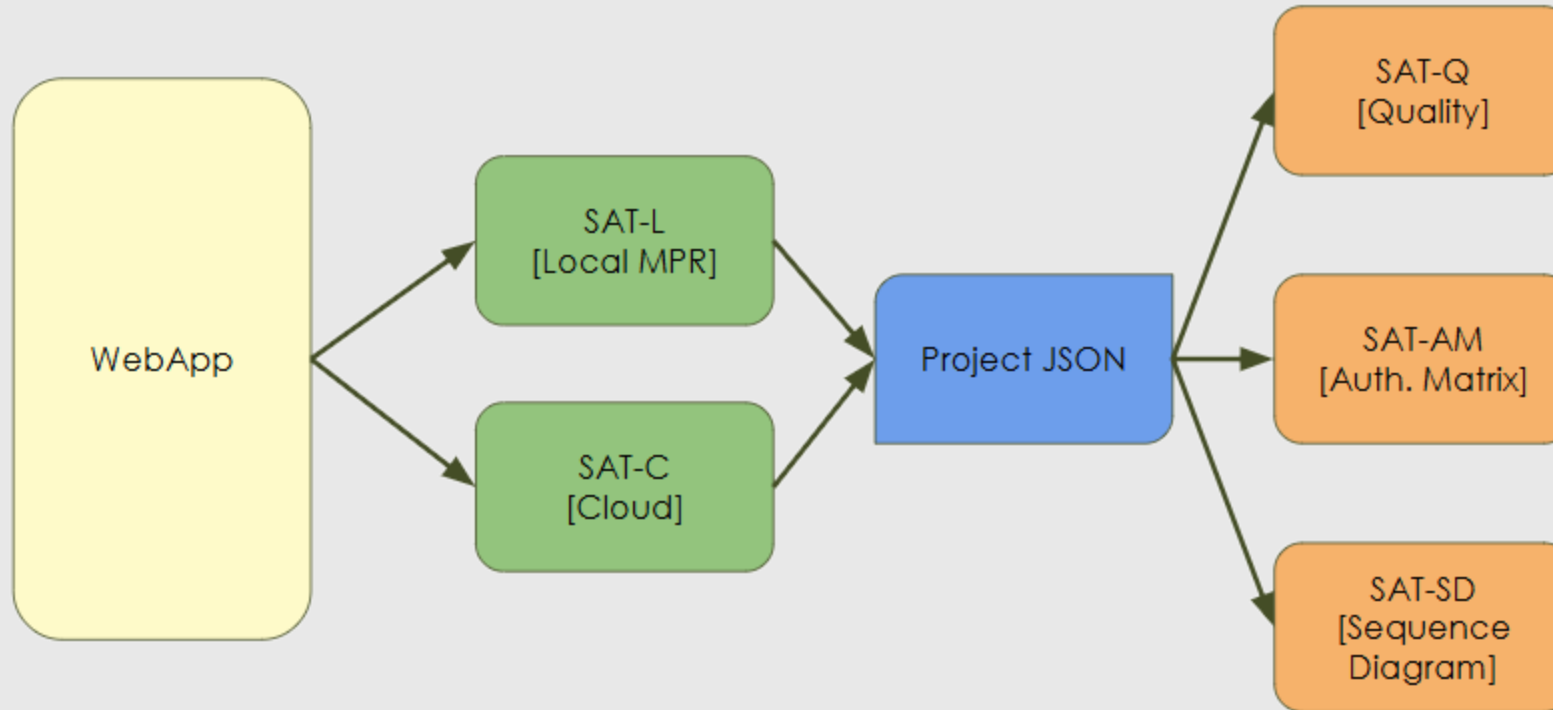
module	fromApp	entity	persistent	attribute	type	app rol	module	rights	default	create	delete	xpath
ModuleName	FALSE	EntityA	FALSE	ATTR a	attr	Developer	Developer	ReadWrite	None	TRUE	TRUE	
ModuleName	FALSE	EntityA	FALSE	ATTR b	attr	Developer	Developer	ReadWrite	None	TRUE	TRUE	
ModuleName	FALSE	EntityA	FALSE	ATTR c	attr	Developer	Developer	ReadWrite	None	TRUE	TRUE	
ModuleName	FALSE	EntityA	FALSE	ATTR d	attr	Developer	Developer	ReadWrite	None	TRUE	TRUE	
ModuleName	FALSE	EntityB	FALSE	ATTR d	attr	Developer	Developer	ReadWrite	None	TRUE	TRUE	
ModuleName	FALSE	EntityB	FALSE	ATTR e	attr	Developer	Developer	ReadWrite	None	TRUE	TRUE	
ModuleName	FALSE	EntityB	TRUE	ATTR f	attr	Developer	Developer	ReadWrite	None	TRUE	TRUE	
ModuleName	FALSE	EntityB	TRUE	ATTR g	attr	Developer	Developer	ReadWrite	None	TRUE	TRUE	
ModuleName	FALSE	EntityB	TRUE	ATTR h	attr	Developer	Developer	ReadWrite	None	TRUE	TRUE	
ModuleName	FALSE	EntityB	TRUE	ATTR i	attr	Developer	Developer	ReadWrite	None	TRUE	TRUE	[ATTR I = 1]

Stoneworx Assessment Tool: Microflows

Generate a text file that can be rendered into a Sequence Diagram



Stoneworx Assessment Tool: Quick Overview



SAT-L

Parses the MPR which is actually a SQLite DB with a lot of bson info.

UnitID	ContainerID	ContainmentName	TreeConflict	ContentsHash	C
È z¬x C ¿x ; i [16]	È z¬x C ¿x ; i [16]		0	/A6Vlgo8LT2zcg+Aq4HE4WwPLXA13llqUY	
= pNáSzG« kv30Â [16]	È z¬x C ¿x ; i [16]	ProjectDocuments	0	fwwaF4Asw7HqhkWJMo7uyCkyQgJpmx5/	
«)ΠÉ \ 3)ŷÎ [16]	È z¬x C ¿x ; i [16]	ProjectDocuments	0	XARgfT9uRY0C50iPJ1ly4DviwJstNvdm8KC	
μiÒW/lúO ³£z@Ú:ß [16]	È z¬x C ¿x ; i [16]	ProjectDocuments	0	ZxnvhQgL3N2RITxBi54GnJCHd5UvkUpzBb	
½eo½ # S F ¥M F [16]	È z¬x C ¿x ; i [16]	ProjectDocuments	0	OAbtbS/MjFobMiYflWJes/FOVF46NBjfJ06,	
üHÔøÁ F¯ oÕÑ È [16]	È z¬x C ¿x ; i [16]	Modules	0	RFahm9usi4EiPJ0/1XRr7MAhYqrc04WEGR	
á ì+Ô J§ §àþ ë [16]	üHÔøÁ F¯ oÕÑ È	DomainModel	0	PSVrjHSq4XiHqgjflenxUdfb4ZzajBdRu8wx	

SAT-C

Connects to working copy in cloud with Mendix SDK's.

```
workingCopyFile = branch + '.workingcopy';
return new Promise((resolve, reject) => {
  const client = new mendixplatformsdk_1.MendixPlatformClient();
  if (clear) {
    wcFile = "";
    wcID = "";
  } else readWorkingCopyFile(appID, workingCopyFile, branch);
  console.log(`GET APP: ${appID}-${branch}`);
  app = client.getApp(appID);
  console.log(`LOADING: ${appID}-${branch}`);
});
```

SAT-Q: Configuration & Extensions

- Config in a json file: [default.json](#)
- Modules in nodejs modules in [SAT-Q](#):

```
56     {
57         "fnc": "ErrorHandling",
58         "options": {
59             "allowedJava": [
60                 "CommunityCommons.EndTransaction",
61                 "CommunityCommons.getGUID",
62                 "OIDC.DecodeJWTPlainText"
63             ]
64         }
65     },
```

```
1  const CheckModule = require("../CheckModule");
2
3  module.exports = class ErrorHandling extends CheckModule {
4  constructor(options) {
5      super(options);
6      this.errorCodes = {
7          "EH1": "Java Action without custom error handling"
8      };
9  }
10
11  check = function (model, microflow) {
12      let allowedJava = this.options.allowedJava;
13      let ignoreRuleAnnotations = microflow.getIgnoreRuleAnnotations(microflow);
14      this.setup(model, microflow);
15      let mfActions = microflow.actions;
16      let javaActions = mfActions.filter((action) => {
17          return action.type == 'Microflows$JavaActionCallAction'
18      })
19      if (javaActions.length > 0) {
20          javaActions.forEach((javaAction) => {
21              let isAllowed = allowedJava.find((allowedJavaName) => allowedJavaName === javaAction.javaActionName);
22              if (!isAllowed) {
23                  let errorHandling = javaAction.errorHandling||'';
24                  if (!(errorHandling.startsWith('Custom'))){
25                      this.addErrors("EH1", ignoreRuleAnnotations);
26                  }
27              }
28          })
29      }
30      return this.errors;
31  }
32  }
```

Demo!

Stoneworx Assessment Tools

Use this tool to execute the SAT program on your Mendix project.

Select SAT Program:

☐ Use local mpr

☐ Use working copy

Output File Name

Exclude Modules:

☐ Quality Assessment

☐ Execute diff with previous version

☐ Authorisation Matrix

☐ Sequence Diagram

Microflow Name:

Aggregate Prefixes:

Execute

Further development

- Add more checks (get inspired by the commercial tools)
- Add more diagrams (like Class Diagram)
- Suggestions? -> Mail me!



Slide deck available in Github Repo:
<https://github.com/StoneworxNL/SAT>