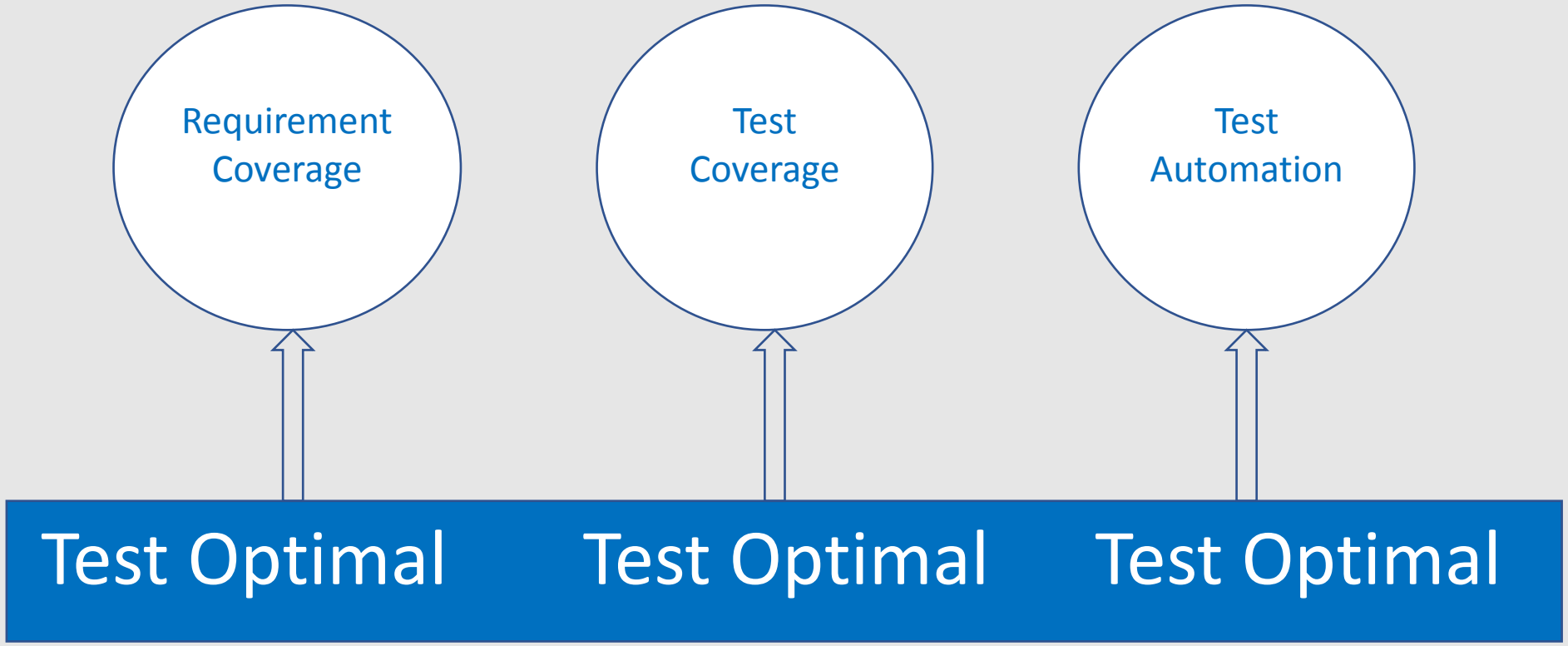
Test embedded system developed in C or other program languages

TestOptimal Introduction

Contact Amy Xia: xia.amy@testoptimal.com

Why choose TestOptimal for embedded system

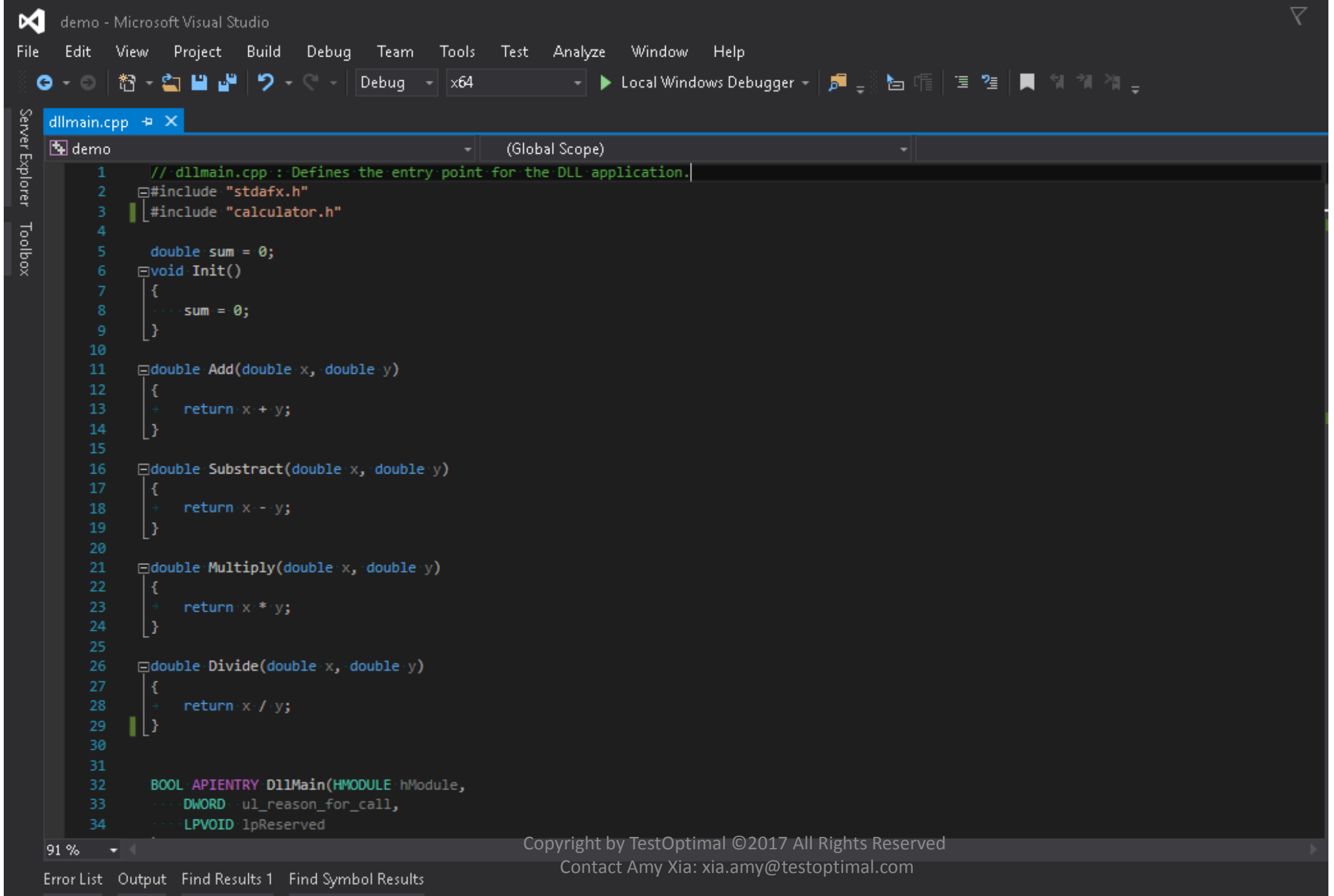


Contact Amy Xia: xia.amy@testoptimal.com

Sample embedded system developed in C

To make it simple, suppose the embedded system compose of four functions:

Contact Amy Xia: xia.amy@testoptimal.com



TestOptimal MScript

MScript is an XML based scripting to either directly drive your application under test (AUT) or generate testing script to be executed or read later for offline testing in any programming language syntax.

- Call custom plugin
- Call remote agent

Contact Amy Xia: xia.amy@testoptimal.com

Embedded Systems

Bridge???

testoptimal
Model-Based Testing

Contact Amy Xia: xia.amy@testoptimal.com

Two Options of the ‘Bridge’

1

Custom Plugin

Custom Plugins is used to extend *TestOptimal* functionality by implementing a set of functions to be used in MScript. Your plugin can then be enabled in Model Property to call its MScript methods in the model

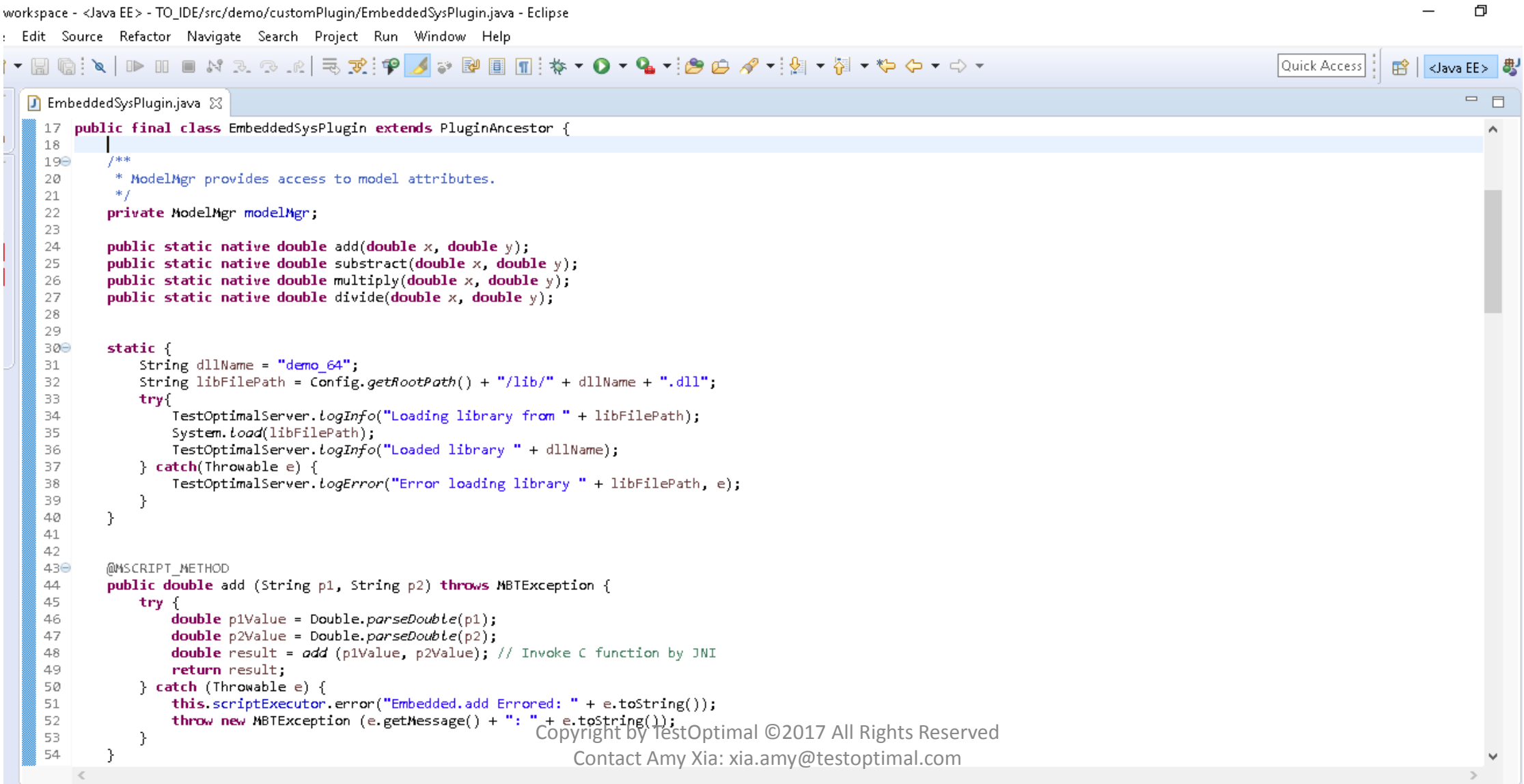
2

Remote Agent

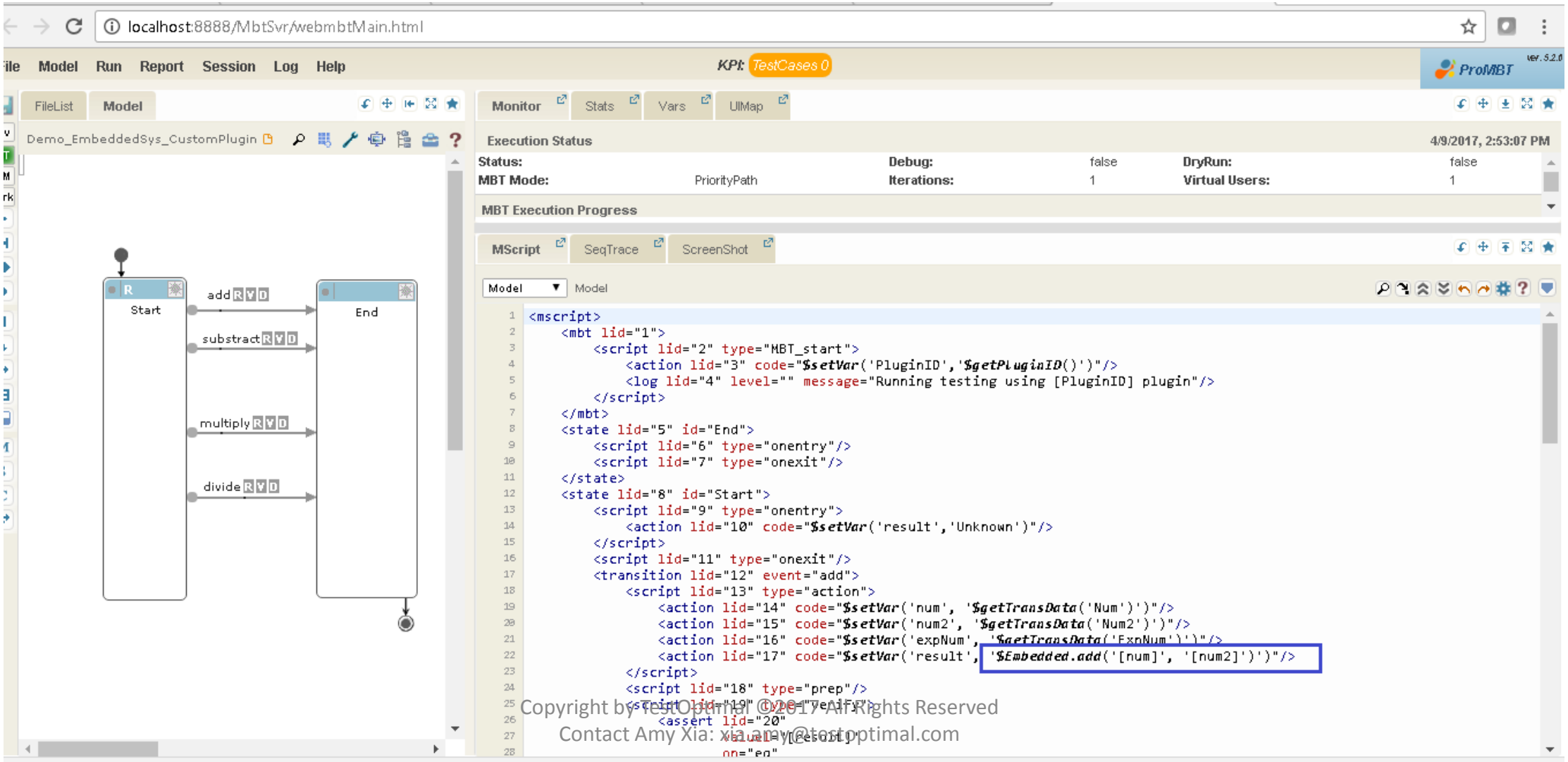
Remote Agent is used to integrate *TestOptimal* with your embedded system. It operates on the plain http, as long as the external tool can send and receive http requests, you can integrate it with *TestOptimal*. This covers pretty much any tools on the market.

Contact Amy Xia: xia.amy@testoptimal.com

Custom Plugin – 1st Bridge



TestOptimal invokes functions defined in Custom Plugin



Embedded Systems

Developed in C or other program languages

Custom Plugin

Developed in Java to expose necessary functions for mscripts

testoptimal
Model-Based Testing

Mscripts invokes functions defined in custom plugin and drives the model running

Contact Amy Xia: xia.amy@testoptimal.com

Remote Agent – 2nd Bridge

/orkspace - <Java EE> - TOAgent/src/demo/DemoEmbeddedSysAgent.java - Eclipse

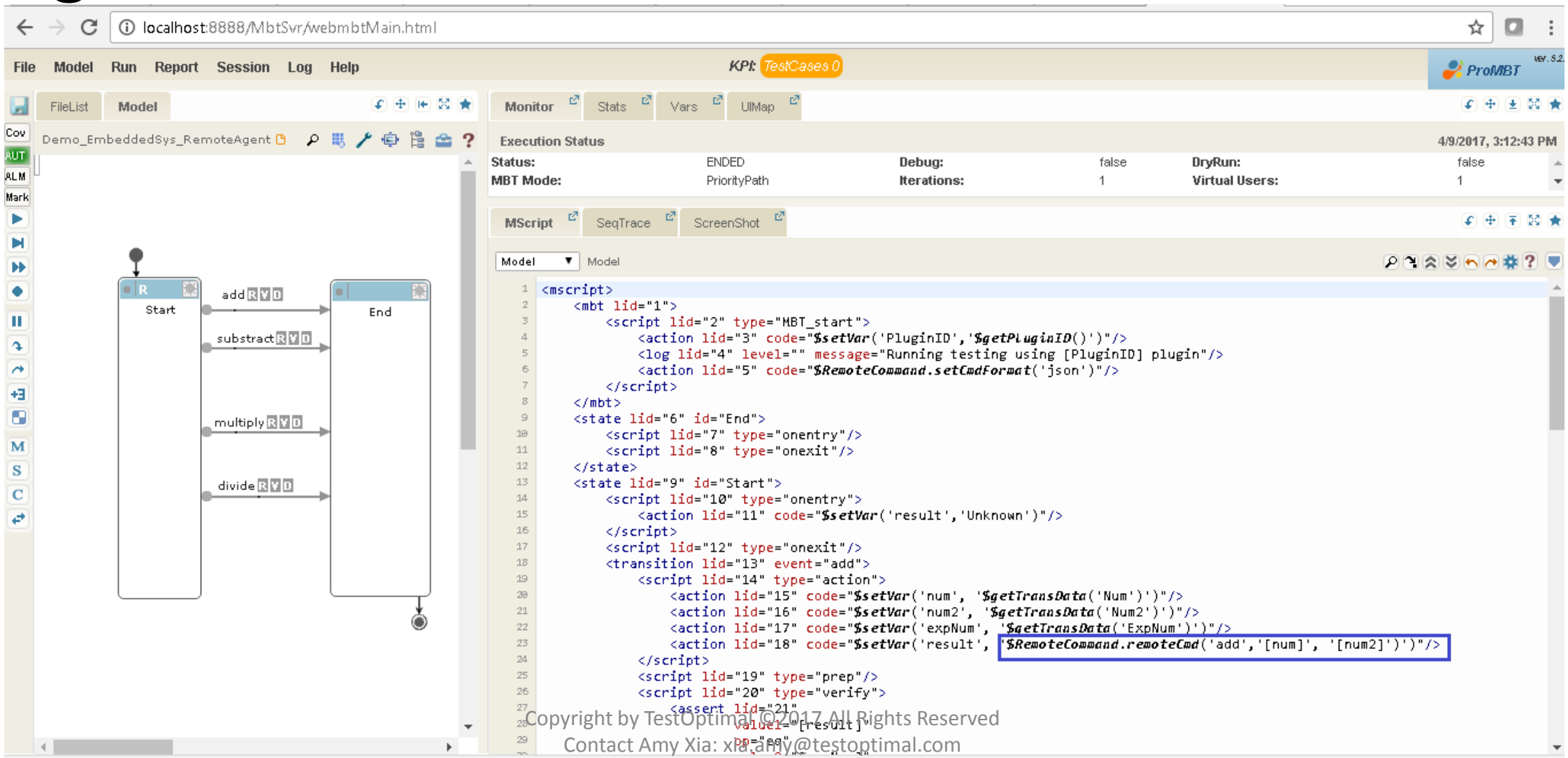
Edit Source Refactor Navigate Search Project Run Window Help

Quick Access <Java EE>

DemoEmbeddedSysAgent.java

```
63    @Override
64    public String execute(RemoteCmd remoteCmd_p) throws Exception {
65        String cmdSyntax = remoteCmd_p.getCmdActionSyntax();
66        this.info("Remote cmd: " + cmdSyntax);
67        this.cmdJSON = new JSONObject(cmdSyntax);
68
69        String cmdAction = this.cmdJSON.getString("ACTION");
70        this.info("Action: " + cmdAction);
71
72        if (cmdAction.equalsIgnoreCase("MBT_start")) {
73            this.info("Starting VendingMachine");
74            return "OK";
75        }
76        else if (cmdAction.equalsIgnoreCase("launchAUT")) {
77            this.info("Running the test");
78            return "OK";
79        }
80        else if (cmdAction.equalsIgnoreCase("resetAUT")) {
81            this.info("Resetting the test");
82            return "OK";
83        }
84        else if (cmdAction.equalsIgnoreCase("add")) {
85            String p1 = this.cmdJSON.getString("P1");
86            String p2 = this.cmdJSON.getString("P2");
87            this.info("add " + p1 + ", " + p2);
88            try {
89                double p1Value = Double.parseDouble(p1);
90                double p2Value = Double.parseDouble(p2);
91                double d = add(p1Value, p2Value);
92                return new Double(d).toString();
93                //return "add result: " + d;
94            }
95            catch (Throwable e) {
96                this.error("Failed to call add function: " + e.toString());
97                throw new Exception (e.toString());
98            }
99        }
100
```

TestOptimal sends http requests to Remote Agent



Embedded Systems

Developed in C or other program languages

Remote Agent

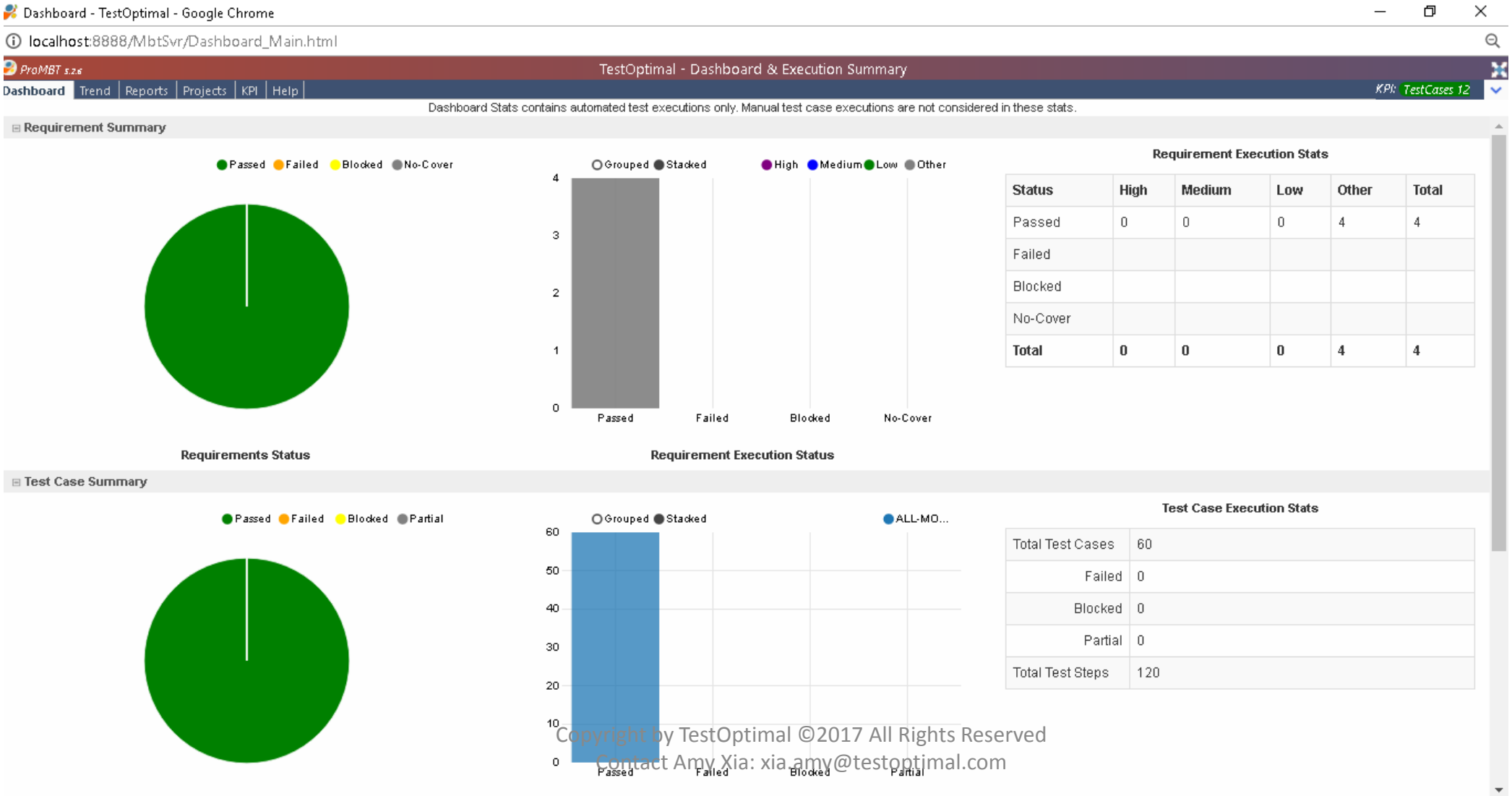
Developed in Java or other program language to handle the http requests between Embedded System and TestOptimal

testoptimal
Model-Based Testing

Mscripts sends to and receives http requests from Remote Agent and drives the model running

Contact Amy Xia: xia.amy@testoptimal.com

TestOptimal Dashboard



Dashboard - TestOptimal - Google Chrome

localhost:8888/MbtSvr/Dashboard_Main.html

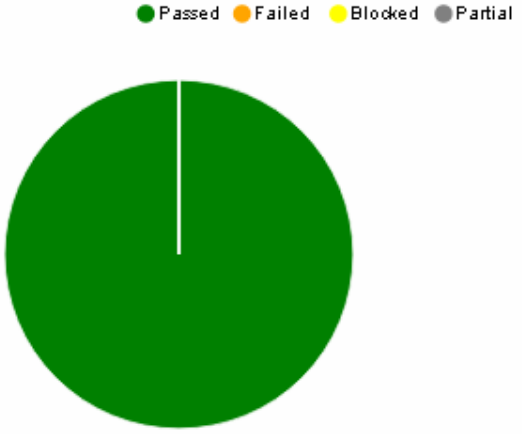
ProMBT 5.2.6

TestOptimal - Dashboard & Execution Summary

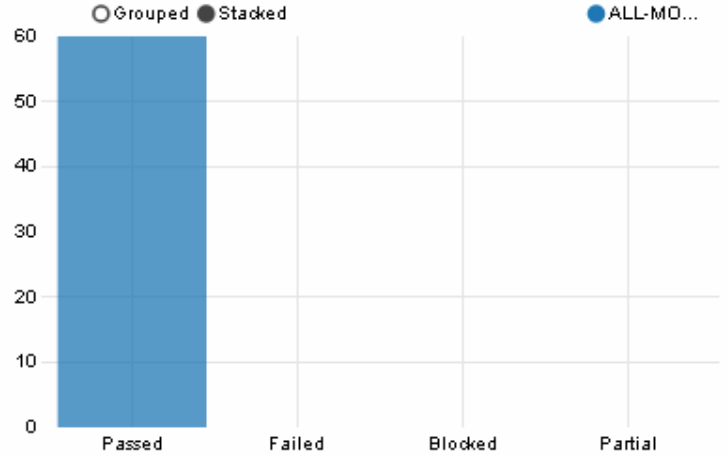
Dashboard | Trend | Reports | Projects | KPI | Help

KPI: TestCases 12

Dashboard Stats contains automated test executions only. Manual test case executions are not considered in these stats.



Test Case Execution Status

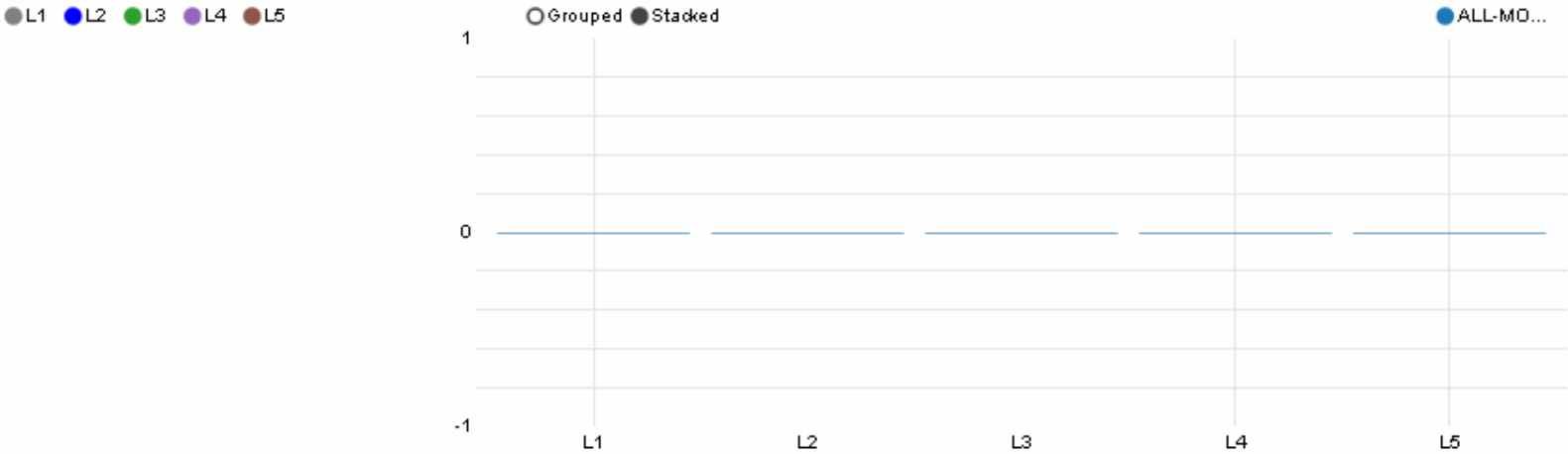


Test Case By Project

Test Case Execution Stats

Total Test Cases	60
Failed	0
Blocked	0
Partial	0
Total Test Steps	120

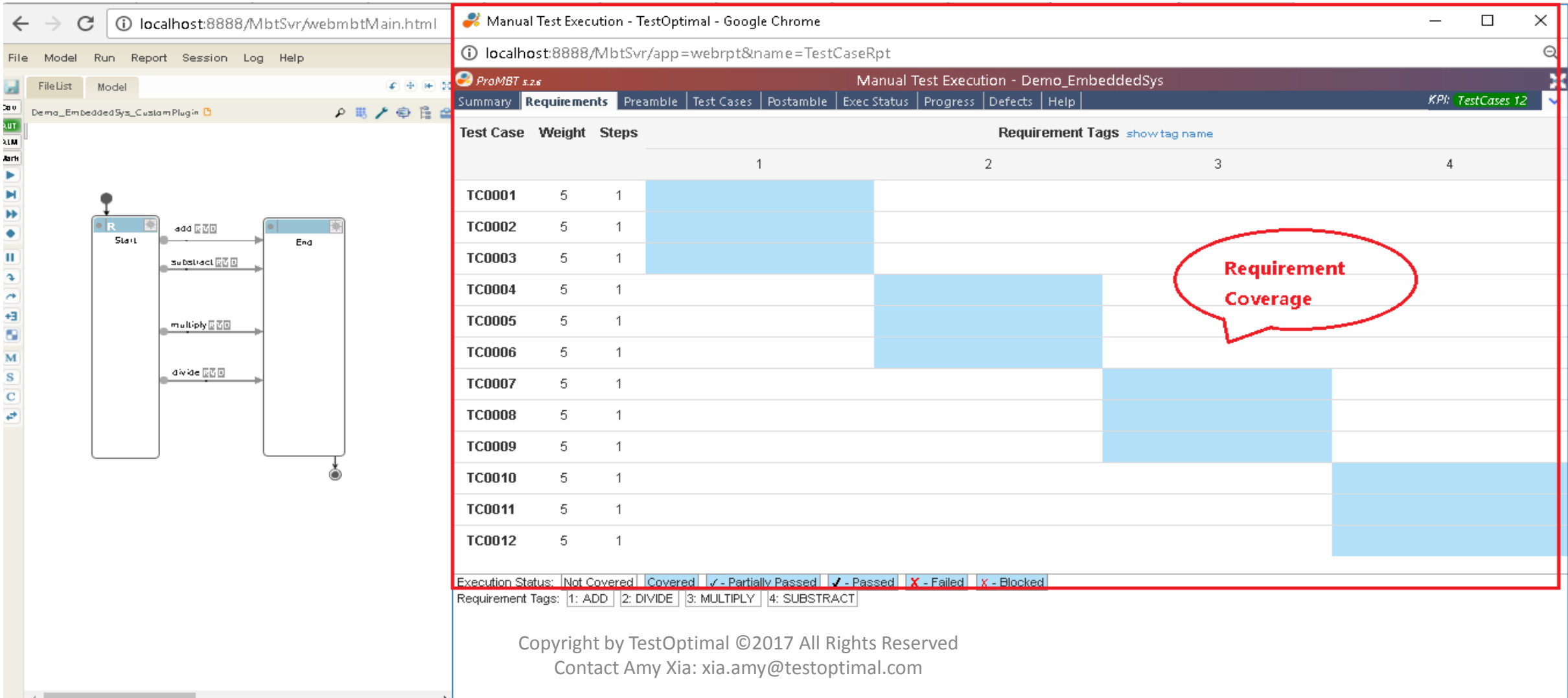
Defects Summary



Defects By Severity

Defects By Project

TestOptimal provides requirement coverage status



Test Cases generated after run the model