

# 101 Solutions

## Unit Testing Documentation

101 Solutions

September 10, 2013

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>What we are using</b>	<b>1</b>
<b>3</b>	<b>Purposes of unit testing</b>	<b>1</b>
<b>4</b>	<b>Contract testing</b>	<b>1</b>
<b>5</b>	<b>Testing the JSON class</b>	<b>1</b>
5.1	Why test it . . . . .	1
5.2	Methodology . . . . .	2
<b>6</b>	<b>Unit Testing AppMan</b>	<b>2</b>
6.1	Communication . . . . .	2
6.2	Management Class . . . . .	2
<b>7</b>	<b>Unit Testing AppManClient</b>	<b>2</b>
7.1	Communication . . . . .	2
7.2	Management Class . . . . .	2

# **1 Introduction**

This document provides an outline of the approach followed when doing unit testing. Different units will be tested and mock testing will be used in order to display that the classes behave as it should.

## **2 What we are using**

We are making use of QT unit tests which allows for a cross platform unit testing. This allows us to test the applicaiton for both windows and linux.

## **3 Purposes of unit testing**

Unit testing is used in order to test across levels of granularity. This is done in order to test whether the individual modules or units behave as expected.

## **4 Contract testing**

Within unit testing that are done there are testing done to determine if contracts are met.

- If preconditions are met
  - Service is provided
  - All post conditions hold true after service was provided
- Any other units not directly tested will be mocked in order to simulate the unit requesting the service

## **5 Testing the JSON class**

### **5.1 Why test it**

Within both the AppMan and AppManClient applications there is a class JSON which converts any JSON value into QVariantMap and QVariant maps in order to make use of json for communication.

## 5.2 Methodology

There are two areas with regards to JSON that are tested. Firstly the JSON class which parses the JSON string, and secondly the functions generating the JSON strings in order to communicate. Both of these classes and functions are mirrored on both the AppMan and AppManClient applications.

- JSON
  - AppMan - JSON class is tested in order to see if it generates the required classes from a JSON string
  - AppManClient - JSON class is tested in order to see if it generates the required classes from a JSON string
- JSON communication
  - AppMan - JSON generating functions which will generate the JSON strings and prepend and append the required strings
  - AppManClient - JSON generating functions which will generate the JSON strings and prepend and append the required strings

## 6 Integration Testing AppMan

### 6.1 Communication

### 6.2 Management Class

- Signal Testing : Signals are tested in order to determine whether they exhibit their required behavior when the management class should emit them.
  - A signal emitted when a new machine connects successfully
  - A signal emitted if a machine updates its md5sum value in order to determine which builds are not synched yet

## 7 Integration Testing AppManClient

### 7.1 Communication

### 7.2 Management Class

- Signal Testing : Signals are tested in order to determine whether they exhibit their required behavior when the management class should emit

them.