

Reality Sensing, Mining and Augmentation   
for Mobile Citizen–Government Dialogue

FP7-288815

**Test scenario’s &** **results**

**Server Side Mining Component (C9)  
Service Line Detection**

|  |  |
| --- | --- |
| fp7_logo | eu-flag |

co-funded by the European Union

**1. Template instructions**

This template is used for documenting test scenarios and test results. ‘D4.4 – Technical verification and testing strategies’ describes per phase which tests need to be performed and which work package/partner is responsible for setting up and performing these tests.

Along with the software development the test scenarios are constructed based on the requirement as described in ‘D4.1 – System Architecture and Design’ and ‘D5.1 – Detailed Use Case Descriptions’.

These test scenarios are described and agreed upon before starting the actual tests. This means that all blue sections need to be pre-filled before starting the actual test. The red sections need to be completed during/after the test.

**2. Test configuration**

|  |  |
| --- | --- |
| Software identification | |
| Name | Server Side Mining Component (C9) – Service Line Detection |
| Versions | Release from 1. June 2014 |

|  |  |
| --- | --- |
| Test period | |
| Test phase | Service Level Testing |
| Test Types | Functional |
| Test Status | Test plan concept |
| Planned test start date | 01.06.2014 |
| Actual test start date | 23.06.2014 |
| Test completion date | 01.07.2014 |
| Partners(s) | UKOB |
| Tester(s) | Christoph Schaefer |

|  |  |
| --- | --- |
| Test environment | |
| Test environment | Development |
| Test devices |  |
| Test pc’s | MacBook Air Mid 2012   * OS X Version 10.9.3 * Chrome Browser Version 35.0.1916.153 |

|  |  |
| --- | --- |
| References | |
| Reference | Service URL:  http://liveandgov.uni-koblenz.de/SLD/api  Test URL:  <http://liveandgov.uni-koblenz.de/SLD/test>  Technical documentation and evaluation of the classification performance is contained in D1.2. |

## Approach

The service line detection service provides a REST API for classifying service lines based on the time tables and Live GPS data. As the system-level component testing takes place in Koblenz, we cannot use the service lines and see if the detection results are correct. Instead we use a specialized web tool (cf. Figure [1]) that allows the simulation of GPS tracks from the individual vehicles.

Remark: Based on the feedback and gathered data from the first trial, we are currently developing a new service line classifier (using Hidden Markov Models). This classifier is not ready for use yet. But it is likely that we will be to use it during the field trial. Therefore, the component testing in this document is using a SLD classifier that received only little improvements from the one deployed in the first trial.

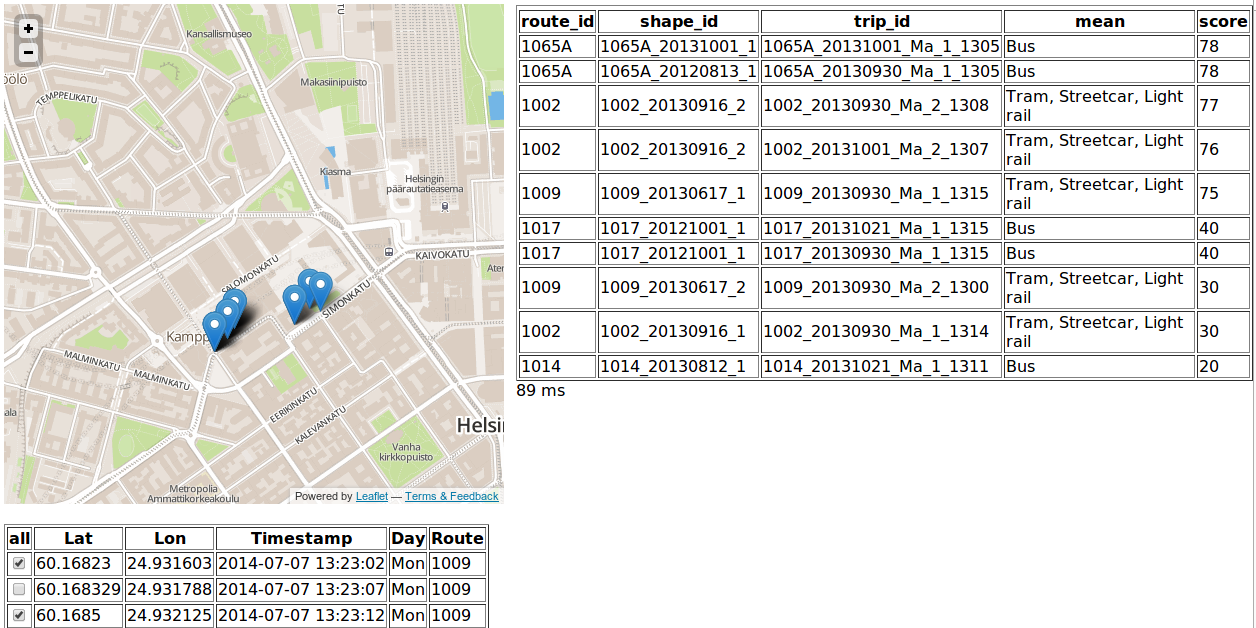


Figure 1: Service Line Detection Test Tool

## Scenarios

The table below should describe the test scenarios executed by the testers to make sure the software meet its requirements and is ready for deployment.

General guidelines for describing scenario’s:

* Tests should be described is such a way that somebody with only minor project knowledge should be able to perform them, so be specific.
* Concentrate on real life scenarios. What are the users, and what should they be able to with the application.
* Try to make separate test scenarios for individual function points.
* While writing test cases keep in mind all your test cases should be simple and easy to understand. Don’t write explanations like essays. Be to the point.
* Keep in mind input data for test cases is very important part in testing, your test cases should validate range of input data. Also check how system behaves in the normal & abnormal conditions, e.g. purposely provide invalid input.
* Make sure test scenarios are added that cover all test types (Functional / User Acceptance / Security / Interoperability), however it is not required to make separate sections for each test type.
* Make sure the test scenarios covers all the required functionality. Assume that all functionality that is not covered by the test scenarios does not work.
* Avoid repetition of test cases

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No. | Requirements | Expected behaviour | Results round 1 | Results round 2 | Results round 3 |
| 1 | R-SC.8. The service shall accept HTTP requests and send back a valid HTTP reply. | A number of preselected sensor samples shall be submitted to the service using a test script. The return code of the response shall be checked to be equal to 200.  The response time for one query should be below 100 milliseconds. | OK | OK |  |
| 2 | RB.8 Detect if a user is riding the bus | Using the test web application create test samples for 5 different bus routes.  Check that at least 4 of 5 bus routes were correctly classified by the SLD module. | NOK | NOK |  |
| 3 | RB.9 Detect if a user is using the train | Using the test web application create test samples for 5 different train routes.  Check that at least 4 of 5 routes were correctly classified by the SLD module. | NOK | NOK |  |
| 4 | RB.10 Detect if a user is using the tram | Using the test web application create test samples for 5 different tram routes.  Check that at least 4 of 5 routes were correctly classified by the SLD module. | NOK | OK |  |
| 5 | RB.13 Detect if a user is using the ferry | Using the test web application create test samples for 5 different ferry routes.  Check that at least 4 of 5 routes were correctly classified by the SLD module. | NOK | NOK |  |
| 6 | Service Center Integration: Healthchecks | The component shall sent health-check signals in regular intervals to the Live+Gov Service Center.  The Service Center Web Application shall show “Staus: OK” for the HAR service. | OK | OK |  |
| 8 | Service Center Integration: Log Files | The component shall upload log files in regular intervals to the Live+Gov service center.  The Service Center Web Application shall show the received log files. | OK | OK |  |

**4. Issues raised**

|  |  |
| --- | --- |
| Issue No. | 1 |
| Scenario ID | 2 |
| Severity | High |
| Type | Bug |
| Summary | Map does not load |
| Description | Just visit the test site at <http://liveandgov.uni-koblenz.de/SLD/test> . There seems to be an error with the map API. All the calls return a 403 Forbidden error. |
| Workaround | - |
| Recommendations | Looks like the map API has changed. Update the Java Script. |

|  |  |
| --- | --- |
| Issue No. | 2 |
| Severity | High |
| Type | Bug |
| Summary | Testing website includes only trams |
| Description | The testing website includes only vehicles with route id 1001 to 1010, which are trams/streetcars |
| Workaround | - |
| Recommendations | Leave the Bus/Ferry/Train testing for integration level testing with real users. |

**5. Issue screenshots**

|  |  |
| --- | --- |
| Issue No. | 1 |
| Macintosh HD:Users:cehlen:Desktop:Screen Shot 2014-06-20 at 17.46.59.png | |

|  |  |
| --- | --- |
|  |  |