



simpli-city

The Road User Information System Of The Future

WP5 – Mobility Services Framework

D5.4: Mobility Service and Application Marketplaces Prototypes

Deliverable Lead: ASC

Contributing Partners: WORLD, TUV

Delivery Date: 22.04.2015

Dissemination Level: Public

Version 1.0

This deliverable describes the work carried out during the development of the prototypes of the SIMPLI-CITY Mobility Service and Application Marketplaces. It specifies the scope and the degree of fulfilment of the requirements to be covered by the marketplaces. It specifies how to install and execute the different subcomponents implemented.



Document Status	
Deliverable Lead	Kristof Kipp, ASC
Internal Reviewer 1	Freddy Lecue, IBM
Internal Reviewer 2	Fredrik Kronlid, TALK
Type	Deliverable
Work Package	WP5 – Mobility Services Framework
ID	D5.4: Mobility Service and Application Marketplaces Prototypes
Due Date	31.03.2015
Delivery Date	22.04.2015
Status	For Approval

Document History	
Contributions	<p>V1.0, Xavier Cases, WORLD, 17.04.2015, Removed all comments and approved all changes.</p> <p>V0.9, Xavier Cases, WORLD, 16.04.2015, Changes on Deployment. Incorporated the third review.</p> <p>V0.8, Xavier Cases, WORLD, 02.04.2015, Incorporated the second review</p> <p>V0.7, Xavier Cases, WORLD, 26.03.2015, Incorporated new screenshot asked in the first review</p> <p>V0.6, Kristof Kipp, ASC, 26.03.2015, Incorporated the first review</p> <p>V0.5, Kristof Kipp, ASC, 23.03.2015, Added Service Marketplace Sections, Cleanup, Formatting.</p> <p>V0.4, Xavier Cases, WORLD, 18.02.2015, Added App Marketplace information, PMA-UI and Web-UI.</p> <p>V0.3, Kristof Kipp, ASC, 09.03.2015, Formatting</p> <p>V0.2, Xavier Cases, WORLD, 18.02.2015, Added Application Marketplace information.</p> <p>V0.1, Kristof Kipp, ASC, 22.01.2015, Added document structure.</p>
Final Version	April 21 th , 2015

Note

This deliverable is subject to final acceptance by the European Commission.

Disclaimer

The views represented in this document only reflect the views of the authors and not the views of the European Union. The European Union is not liable for any use that may be made of the information contained in this document.

Furthermore, the information is provided “as is” and no guarantee or warranty is given that the information is fit for any particular purpose. The user of the information uses it at its sole risk and liability.

Project Partners



TECHNISCHE
UNIVERSITÄT
WIEN
Vienna University of Technology

Vienna University of Technology (Coordinator),
Austria



Ascora GmbH, Germany



TIE Nederland B.V., The Netherlands



Technische Universität Darmstadt, Germany



IBM Research – Ireland
Smarter Cities Technology Centre



Forschungsgesellschaft Mobilität, Austria



Talkamatic AB, Sweden



Atos Worldline, Spain



Centro Ricerche FIAT, Italy



SRM – Reti e Mobilità, Italy

Executive Summary

This deliverable describes the work, which was carried out during the development of the prototype of the Mobility Service and Application Marketplaces Prototypes. For this, this document starts with introducing the Mobility Service and Application Marketplaces and describes the scope of the prototypes.

After this, the degree of fulfilment of each requirement to be covered by the prototypes and specified in the Requirements Analysis Report (D2.3) is described.

The current version of the Mobility Service and Application Marketplaces contains the mobile App Marketplace in form of an app for the Android Platform as well as a web view with the same contents. It also provides the Service Marketplace, which is connected to the Service Registry to view available services as well as create licenses for developers that want to make use of the services.

The current version of the App Marketplace is composed by two components, a mobile app for Android devices to allow users to install/uninstall/comment/search apps on the App Marketplace and a website that provides the full cycle of development/testing/approval/publish of an app.

Table of Contents

1	Introduction	7
1.1	SIMPLI-CITY Project Overview	7
1.2	Deliverable Purpose, Scope and Context	8
1.3	Document Status and Target Audience	8
1.4	Abbreviations and Glossary	8
1.5	Document Structure	8
2	Prototype Scope and Requirements Coverage	10
2.1	Mobility Service and Application Marketplaces – General Information	10
2.2	Scope of the Prototypes	11
2.2.1	App Marketplace PMA-UI	11
2.2.2	App Marketplace Web-UI	12
2.2.3	Service Marketplace UI	12
2.2.4	Payment API	12
2.3	Covered Requirements	13
3	Preparations	17
3.1	App Marketplace Web-UI	17
3.2	Service Marketplace	18
4	Installation (Deployment)	19
4.1	App Marketplace Web-UI	19
4.2	App Marketplace PMA-UI	19
4.3	Service Marketplace	19
5	Execution and Usage of the Software	21
5.1	App Marketplace Web-UI	21
5.2	App Marketplace PMA-UI	25
5.3	Service Marketplace	30
6	Summary	33

1 Introduction

SIMPLI-CITY – The Road User Information System of the Future – is a project funded by the Seventh Framework Programme of the European Commission under Grant Agreement No. 318201. It provides the technological foundation for bringing the “App Revolution” to road users by facilitating data integration, service development, and end user interaction.

Within this document, the prototypes of the SIMPLI-CITY Mobility Service and Application Marketplaces will be presented. The document accompanies the corresponding software prototype, which is the main content of the deliverable.

1.1 SIMPLI-CITY Project Overview

Analogously to the “App Revolution”, SIMPLI-CITY adds a “software layer” to the hardware-driven “product” mobility. SIMPLI-CITY will take advantage of the great success of mobile apps that are currently being provided for systems such as Android, iOS, or Windows Phone. These apps have created new opportunities and even business models by making it possible for developers to produce new apps on top of the mobile device infrastructure. Many of the most advanced and innovative apps have been developed by players formerly not involved in the mobile software market. Hence, SIMPLI-CITY will support third party developers to efficiently realise and sell their mobility-related service and app ideas by a range of methods and tools, including the Mobility Services and App Marketplaces.

In order to foster the wide usage of those services, a holistic framework is needed which structures and bundles potential services that could deliver data from various sources to road user information systems. SIMPLI-CITY will provide such a framework by facilitating the following main project results:

- **Mobility Services Framework:** A next-generation European Wide Service Platform (EWSP) allowing the creation of mobility-related services as well as the creation of corresponding apps. This will enable third party providers to produce a wide range of interoperable, value-added services, and apps for drivers and other road users.
- **Mobility-related Data as a Service:** The integration of various, heterogeneous data sources like sensors, cooperative systems, telematics, open data repositories, people-centric sensing, and media data streams, which can be modelled, accessed, and integrated in a unified way.
- **Personal Mobility Assistant:** An end user assistant that allows road users to make use of the information provided by apps and to interact with them in a non-distracting way – based on a speech recognition approach. New apps can be integrated into the Personal Mobility Assistant in order to extend its functionalities for individual needs.

To achieve its goals, SIMPLI-CITY conducts original research and applies technologies from the fields of Ubiquitous Computing, Big Data, Media Streaming, the Semantic Web, the Internet of Things, the Internet of Services, and Human-Computer Interaction. For more information, please refer to the project website at <http://www.simpli-city.eu>.

1.2 Deliverable Purpose, Scope and Context

The purpose of this document is to provide the means to use the prototypes of the Mobility Service and Application Marketplaces and exploit their functionalities. For this, the scope and requirements of the Mobility Service and Application Marketplaces and these prototypes, the requirements and preparations for users, an installation and usage guide are provided.

The final Mobility Service and Application Marketplaces Prototypes is the outcome of the discussions and implementation work done in project months 9 to 30. It provides the final version of the App Marketplace on two platforms (Android and the web) as well as the final version of the Service Marketplace, which is only available on the web. It also describes the interactions between parts of this component and other components within the SIMPLI-CITY project, e.g., the Service Runtime Environment and the Service Registry.

The web App Marketplace brings to developers the possibility to upload their apps. These apps will be tested, approved and published by dedicated testers to check if submitted apps are ethically and technically acceptable. The web also gives to consumers of apps the possibility to search and find apps and push them to the owned device.

1.3 Document Status and Target Audience

This document is listed in the Description of Work (DoW) as “Public”. It provides the means to exploit the functionalities of the SIMPLI-CITY Mobility Service and Application Marketplaces as defined in deliverable D3.2.2 (Technical Specification).

While the document primarily is aimed at the project partners, this public deliverable can also be useful for the wider scientific and industrial community. This includes other publicly funded projects, which may be interested in collaboration activities.

1.4 Abbreviations and Glossary

A definition of common terms and roles related to the realization of SIMPLI-CITY as well as a list of abbreviations is available in the supplementary document “Supplement: Abbreviations and Glossary”, which is provided in addition to this deliverable.

Further information can be found at <http://www.simpli-city.eu>.

1.5 Document Structure

This deliverable is broken down into the following sections:

Section 1 provides an introduction for this deliverable including a general overview of the project and outlines the purpose, scope, context, status, and target audience of this deliverable.

Section 2 provides an overview of the scope and relationship of the prototype, showing where the Mobility Service and Application Marketplaces fit into the overall SIMPLI-CITY software framework and the outcome of the prototypes. Furthermore, an assessment of the requirements covered by this prototype is given.

D5.4_Mobility_Service_and_Application_Marketplaces_Prototypes_v1.0_For_Approval.docx	Document Version: 1.0	Date: 2015-04-22	Status: For Approval	Page: 8 / 33
http://www.simpli-city.eu/		Copyright © SIMPLI-CITY Project Consortium. All Rights Reserved. Grant Agreement No.: 318201		

Section 3 presents the requirements and preparations to be done by software developers if they want to make use of the final prototypes of the Mobility Service and Application Marketplaces.

Section 4 offers information about the installation and deployment of the provided software package.

Section 5 describes how software developers can use the provided functionalities.

Finally, Section 6 provides a summary of the document.

2 Prototype Scope and Requirements Coverage

In this section the scope of the final prototype and its coverage of the requirements defined in deliverable D2.3 are discussed.

2.1 Mobility Service and Application Marketplaces – General Information

The Mobility Service and Application Marketplaces offer the possibility of adding new functionalities for road users using SIMPLI-CITY. Those new functionalities are provided by installing new apps on the Personal Mobility Assistant (PMA). Each app is wrapped around one or more backend services, which delivers data and perform server-side computations.

Apps are offered in the SIMPLI-CITY App Marketplace using a PMA-based user interface (UI) or a web-based view. This allows users to discover, buy and install new apps within the web view as well as upgrade and uninstall apps via the PMA-based App Marketplace.

Analogously, services are offered in the Service Marketplace. While the app side mainly targets end users, the Service Marketplace targets software developers and is therefore usually not directly seen by end users.

The main functionality of the marketplaces is to provide end users with the functionality to discover, download and install apps on the PMA (App Marketplace) and to provide software developers with the functionality to discover and study services as a base for new developments (Service Marketplace).

2.2 Scope of the Prototypes

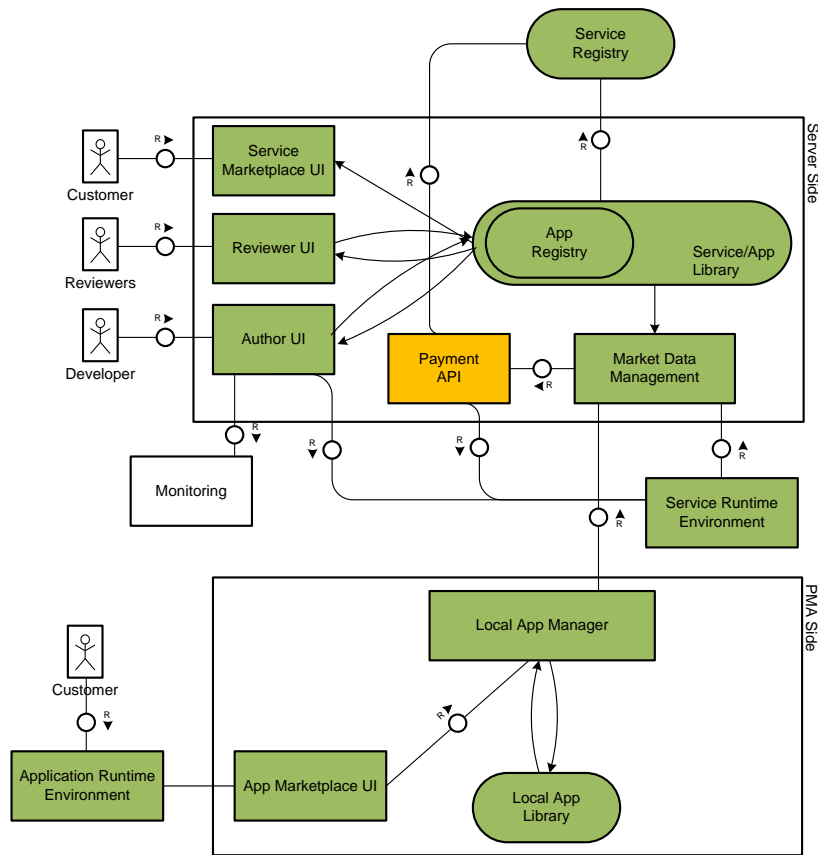


Figure 1: Scope of the Prototypes of the Mobility Service and Application Marketplaces

Figure 1 depicts the status of the development of the Mobility Service and Application Marketplaces Prototypes. For that, the subcomponents that are covered within this final prototype are highlighted. Note, that this figure shows the Mobility Service and Application Marketplaces and also other parts of the SIMPLI-CITY Global Architecture which are not developed within this component.

The status of the implementation is shown using the following colour codes:

- Green: Fully implemented
- Orange: Partially implemented
- White: No implementation, or not part of this deliverable

In the following subsections, the scope and status of the single subcomponents (as depicted in Figure 1) will be discussed in more detail. For the Functional Specification and Technical Specification of these subcomponents, refer to SIMPLI-CITY deliverables D3.2.1 and D3.2.2, respectively.

2.2.1 App Marketplace PMA-UI

The App Marketplace PMA-UI is the main app to find and install new SIMPLI-CITY apps on the PMA. It runs on the PMA which contains the App Marketplace UI as an app named “SIMPLI-CITY Store”.

The App Marketplace PMA-UI allows end users to browse apps published in the marketplace and install them on the device. It provides a search functionality that permits to search for apps matching specific filter parameters, and to read the app description. Moreover, users are able to provide feedback concerning the app, to help other users decide about the quality of apps they may want to install.

2.2.2 App Marketplace Web-UI

The App Marketplace Web-UI is the web app for SIMPLI-CITY and runs independently from the PMA.

The App Marketplace Web-UI allows developers to create, test and publish applications for the PMA. It has a version control for updates and a testing system to verify that the functionalities required work as expected.

The App Marketplace Web-UI, as well as the PMA-UI, allows end users to browse apps published in the marketplace and install them on their devices (PMA). It does the installation through a Push Service. It also provides a search functionality that allows users to search for apps matching specific filter parameters and provides the means to read the app description. The user will be able to also see the comments and feedback provided by other users of the apps to help make the decision.

2.2.3 Service Marketplace UI

The Service Marketplace UI is a web UI aimed at third party developers that permits to search services published in the Service Marketplace in order to (re-)use them within their services and apps.

It provides a search form in order to look for services based on different input parameters like the name of the service, the description, or keywords. The result of the search provides a list of services matching the search parameters. The user is able to visualize the detailed information of a service, and also the information about how to use the service within their applications.

2.2.4 Payment API

The Payment API is not included in a direct way as it was foreseen in the Technical Specification. The App Marketplace (Web-UI and PMA-UI) makes use of an internal method to process payments, while the Service Marketplace makes use of the Service Registry and the Service Runtime Environment to create Licenses. The Service Registry (see deliverable D5.3.2) will then create invoices and handle the payment of the service usage.

2.3 Covered Requirements

This section describes the degree of fulfilment of the requirements to be covered by the Mobility Service and Application Marketplaces as specified in the Requirements Analysis deliverable (D2.3) and the Functional Specification (D3.2.1).

Table 1: Requirements Related the Mobility Service and Application Marketplaces and their Degree of Fulfilment

Requirement	Degree of Fulfilment	Comment
Must Have Requirements		
U62: Versioning of apps (P1)	100%	The App Marketplace provides the means to manage version numbers of apps (similar to U144).
U63: Upgrading of apps (P1)	100%	Users of the App Marketplace are able to upgrade apps in case a new version of an app is available.
U144: App versioning (P1)	100%	The developer is able to commit the app several times increasing the app version.
U64: Quality assurance – Mockup (P1)	100%	Mockup Quality assurance is given as well as exceeded by fulfilling Requirement U65.
U142: Quality/checking certification (P1)	100%	The user role “Tester” exists within the App Marketplace to review and approve apps.
U66: App marketplace easy to use (P1)	100%	The App Marketplace has been done carefully by designers to be friendly and easy to use.
U67: Apps are easy to buy (P1)	100%	The App Marketplace contains an integration of payment services that make it easy to pay for apps.
U68: Apps are easy to search (P1)	100%	The PMA-UI device app has a search box that makes easy to find the desired app. The Web-UI has a page for customers that allow them to find the proper app easily.

Requirement	Degree of Fulfilment	Comment
U69: App download to the device (P1)	100%	The PMA-UI allows direct download of a selected app. The Web-UI provides a way to push the desired app to an owned device (PMA).
U70: App installation (P1)	100%	The PMA-UI device app allows the installation in an easy way.
U71: App uninstallation (P1)	100%	The PMA-UI device app allows the uninstallation in an easy way.
U138: Provision of app statistics (P2)	100%	The App Marketplace provides statistics, e.g. number of downloads/installations and traffic data.
U139: Provision of service statistics (P1)	N/A	This is now part of the Monitoring component of the Service Runtime Environment (see deliverable D5.3.3) and therefore not considered in this prototype.
U182: Provision of statistics, e.g., usage, traffic (P3)	100%	The App Marketplace provides statistics, e.g. number of downloads/installations and traffic data.
U90: Availability	100%	Passwords and sensible data is automatically encrypted and will not be transmitted in a human readable format.
U91: Integrity	100%	The App Marketplace PMA-UI is fully integrated into the PMA, while the Web-UIs for the App and Service Marketplace are both running in an independent environment.
U92: Secure access to system U93: Third party access to the system	100%	Transmission of data between the Marketplaces and the clients are always secured, therefore all sensible data is encrypted.
Should Have Requirements		
U55: Rating of apps (P3)	100%	The App Marketplace provides a way to set comments and evaluate the app by the users.

Requirement	Degree of Fulfilment	Comment
U56: Feedback to developers through the marketplace (P3)	100%	All comments are sent to the developer.
Could Have Requirements		
U57: Social network functionality with the objective of spreading the word (P3)	0%	Not considered in this prototype
U141: Spreading the word (P3)	0%	Not considered in this prototype
U58: User recommendations (P3)	0%	Not considered in this prototype
U60: Payment for services – Mockup (P1)	100%	Makes use of the Service Registry, which handles Licensing of Services.
U128: The system allows app developers to make money (P1)	100%	Licenses can be created within the Service Marketplace and contain different payment information to monetize services upon invocation.
U129: The system allows the consortium to make money from apps (P2)	50%	Regarded for service developers, to be integrated for app developers.
U130: Apps are free but supported, i.e., red hat model (P4)	100%	App developers can directly be contacted by Users within the App Marketplace.
U131: Free and premium versions of apps (P4)	100%	Apps can be marked as free or add download costs for their apps.
U132: Guarantee and minimum service: life time, support time (P5)	100%	In case the life-time or support time ended, developers and maintainers are able to deny further usage of an app or service.
U133: The system allows service developers to make money (P1)	100%	Makes use of the Service Registry, which handles Licensing of Services.
U134: The system allows the consortium to make money from services (P2)	100%	The Service Marketplace makes use of the Service Registry, which handles Licensing of Services.

Requirement	Degree of Fulfilment	Comment
U135: Services are free but supported, i.e. red hat model (P4)	100%	Licenses can be created within the Service Marketplace and contain different payment information to monetize services upon invocation.
U136: Free and premium versions of services (P4)	100%	The Service Marketplace makes use of the Service Registry, which handles Licensing of Services.
U137: Guarantee and minimum service: life time, support time (P5)	100%	The Service Marketplace makes use of the Service Registry, which handles Licensing of Services.
U65: Quality assurance – Fully deployed (P3)	100%	The user role “Tester” exists within the App Marketplace to review and approve apps. Apps need to be approved by Testers before they are available to users.
U140: Promotional aspects (P4)	0%	Not considered in this prototype

3 Preparations

This section provides information about what potential users (both administrators and software developers) need to prepare in order to use the functionalities of the delivered prototype.

3.1 App Marketplace Web-UI

To install the required software to make the App Marketplace Web-UI work, the next steps will have to be followed.

1. Before installing the needed packages it is recommended to update the package index of the Advanced Packing Tool (APT). This is done via the command:

```
sudo apt-get update
```

2. To have a fully working environment available, it is needed to run the following command with a user that is eligible to use “sudo” to install Tomcat server that will be responsible for deploying and presenting the App Marketplace Web-UI:

```
sudo apt-get install tomcat7
```

3. After the installation of tomcat7 a change to port 80 may be needed. (Further configurations in <http://tomcat.apache.org/tomcat-7.0-doc/config/>). The file `/etc/tomcat7/server.xml` has to be edited. Open the file with a file editor (e.g., vi)

```
sudo vi /etc/tomcat7/server.xml
```

4. In lines 72 to 75 of the standard config, the port attribute from the connector tag has to be changed from “8080” to “80”:

```
<Connector port="8080" protocol="HTTP/1.1"
           connectionTimeout="20000"
           URIEncoding="UTF-8"
           redirectPort="8443" />
```

5. Additionally line 47 (the last line) in `/etc/default/tomcat7` has to be changed to match the following:

```
AUTHBIND=yes
```

6. Now the following three commands need to be run:

```
sudo touch /etc/authbind/byport/80
sudo chmod 500 /etc/authbind/byport/80
sudo chown tomcat7 /etc/authbind/byport/80
```

7. Tomcat can now be restarted to make changes take effect:

```
sudo service tomcat7 restart
```

8. A MySQL Server database is needed to provide the persistence, to install it the next command has to be executed:

```
sudo apt-get install mysql-server-5.6
```

9. Once all the steps above are worked through, the preparations for the App Marketplace Web-UI are done.

3.2 Service Marketplace

1. Install Service Runtime Environment (see deliverable D5.3.3 – Service Runtime Environment Prototype II)
2. Before installing the needed packages it is recommended to update the package index of APT. This is done via the command:

```
sudo apt-get update
```

3. To have a fully working environment available, it is only needed to run the following command with a user that is eligible to use “sudo”:

```
sudo apt-get install golang
```

4. Extract the Service Marketplace sources from the *service-marketplace.zip* file.
5. The config file needs to be adapted according to the server the Service Runtime Environment (and therefore the Service Registry the Service Marketplace needs) runs on. The config file can be found in the config folder inside the component package.

```
# Service Registry
URL = "http://128.131.223.138"
Port = "8080"
ServiceName = "/cxf/serviceRegistry"
```

6. The URL section needs to be changed to the IP address the Service Registry is running on. The Port and the ServiceName parts only need to be changed in case the Service Runtime Environment (SRE) configuration is manually changed. This needs to be aligned with the SRE settings.
7. Once all the steps above are worked through, the preparations for the Service Marketplace are done.

4 Installation (Deployment)

This section provides guidelines on how to install and deploy the first prototype of the Media Data Streams on a Debian Linux or a Debian based derivate machine, i.e., Ubuntu or Linux Mint.

4.1 App Marketplace Web-UI

To deploy the App Marketplace Web-UI in the prepared infrastructure, the next steps have to be followed.

1. The following command has to be executed to create the database and populate it with the basic information needed.

```
mysql < create_database_script.sql
```

2. If the database is installed on another server, a new ddbb.properties files must be set. Set the corresponding ddbb.properties, following the one provided and execute the following command.

```
./setdatabase.sh AppMarketplaceWebUI-1.0.war ddbb.properties
```

3. After that, the application file needs to be moved to the deployment folder of Apache Tomcat.

```
sudo mv .AppMarketplaceWebUI-1.0.war /var/lib/tomcat7/webapps
```

4. Once all the steps above are done, the App Marketplace Web-UI is ready to be used.

4.2 App Marketplace PMA-UI

To install the APK of the App Marketplace PMA-UI, the next steps need to be followed:

1. The APK needs to be put into any folder of the device.
2. In the configuration section of the device, the "Unknown Sources" checkbox needs to be in a checked state. This option allows the installation of apps from sources other than the Play Store.
3. The formerly copied APK File needs to be opened from within the Folder Explorer.

4.3 Service Marketplace

After preparing the server to run the Service Marketplace, the archive needs to be extracted. Before the service is started, the \$GOPATH environment variable needs to be set. This includes two steps:

```
cd <projectpath>/src
export GOPATH=`pwd`
```

Afterwards the missing go packages need to be installed. This is automatically done by the environment with the following command:

```
go get
```

After all missing packages are installed the server is ready to be started. It can be started by running the following command:

```
go run main.go
```

Now the demo webserver is running on port 3000 of the local machine, so it is accessible via a web browser. The url for the server is “http://localhost:3000”.

5 Execution and Usage of the Software

This section describes how to use the different subcomponents of the prototype. To access the App Marketplace Web-UI, you can find it in <host>/AppMarketplaceWebUI-1.0

5.1 App Marketplace Web-UI

Depending on the role, a user will be able to do different actions and therefore each role has its own screens. The roles as well as the available screens are listed below.

- Developer
 - Allowed to: Create apps, submit media, submit information, publish the app and/or update it (Last two only after reaching the status “approved for publication”)
 - Screens: My Apps, Add App
- Tester
 - Allowed to: Go through the tests prepared for the app, answer the questions verifying that the app works properly
 - Screens: My Tests
- Approver
 - Allowed to: Create/delete/update tests, approve an app for testing and allow the developer to publish an application or reject it for some reason (the last two only after the testing is done)
 - Screens: Approve for testing, Approve for publication, Tests Management
- Admin
 - Allowed to: Every action
 - Screens: Every screen

Figure 2 shows the My Apps screen that lists all the apps created by a developer. The figure shows different lists depending on the status.

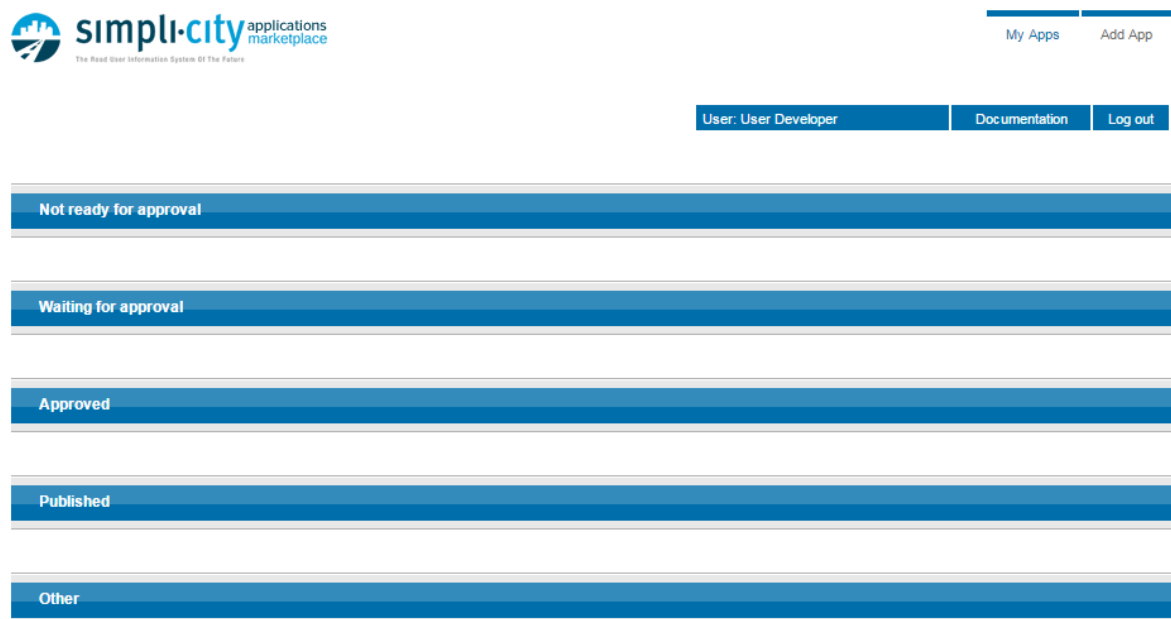


Figure 2: View of My Apps from the Developer Role's Perspective

In Figure 3 the “Add App” screen is shown. This screen allows the developer to create a new app providing the name of the app, a description and other information to enrich it.

Figure 3: View of Add App from the Developer Role's Perspective

Figure 4 shows the “My Tests” screen. In this screen a tester will see the apps that have tests in progress or are already finished. By clicking on the listed apps the tester will get into the detail-view and will be able to see questions and provide an answer to them.

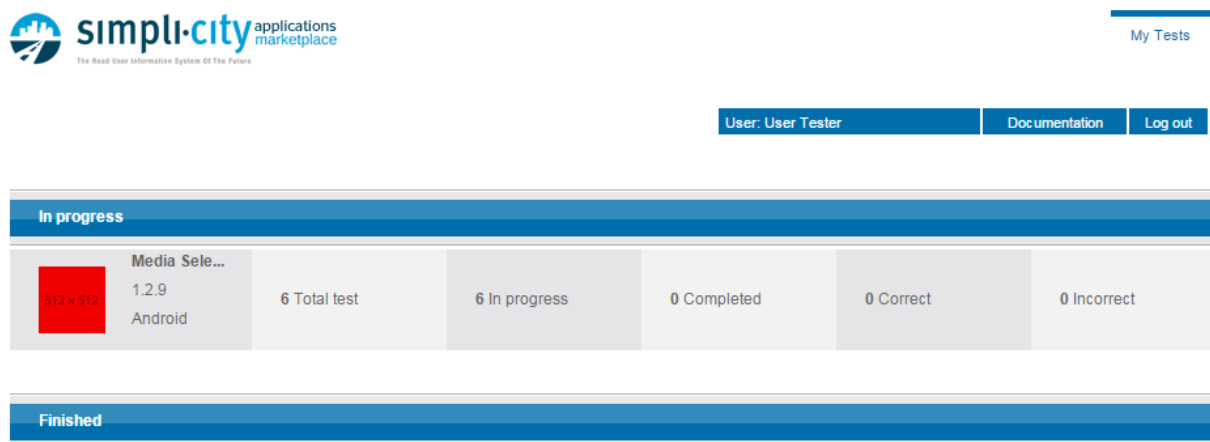


Figure 4: View of My Tests from the Tester Role's Perspective

Figure 5 shows the “Approve for Testing” screen. In this screen an approver can assign a group of tests to a user.

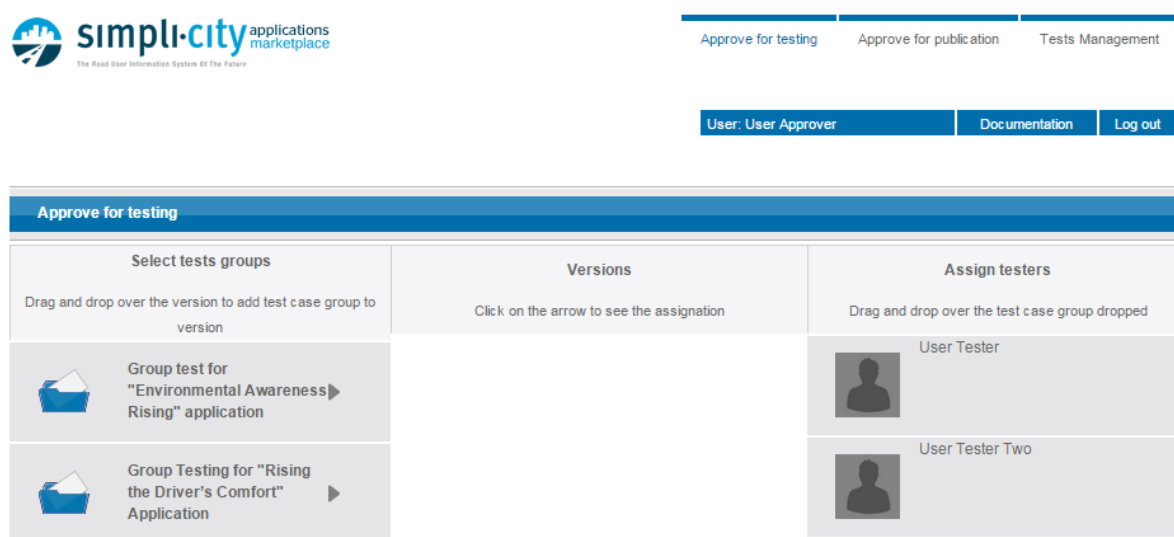


Figure 5: View of Approve for Testing from the Approver Role's Perspective

In Figure 6 the “Approve for publication” screen can be seen. In this screen the approver can see a list of apps and their status. From this screen the approver can “Approve” or “Reject” an app.

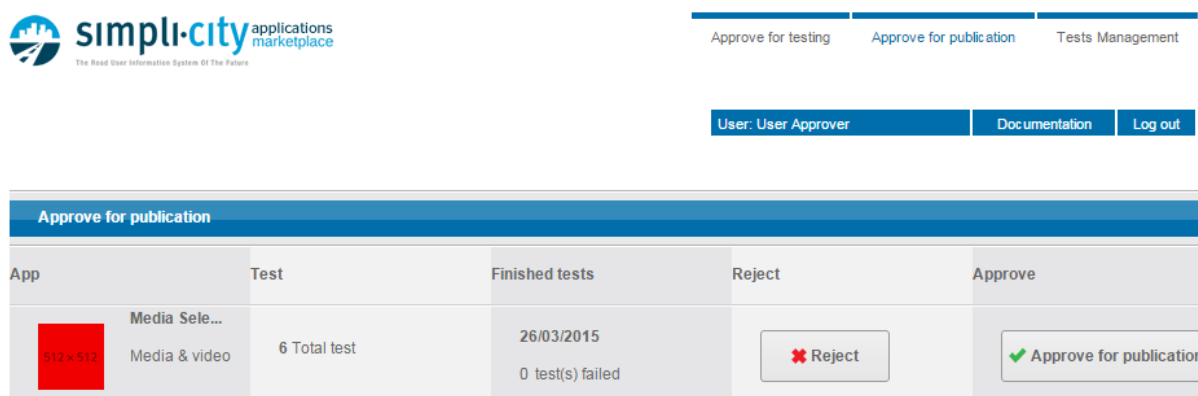


Figure 6: View of Approve for Publication from the Approver Role’s Perspective

Figure 7 depicts the “Test Management” screen. In this screen an approver is able to create new tests, and he/she can group them to categories and assign specific groups of tests to an app.

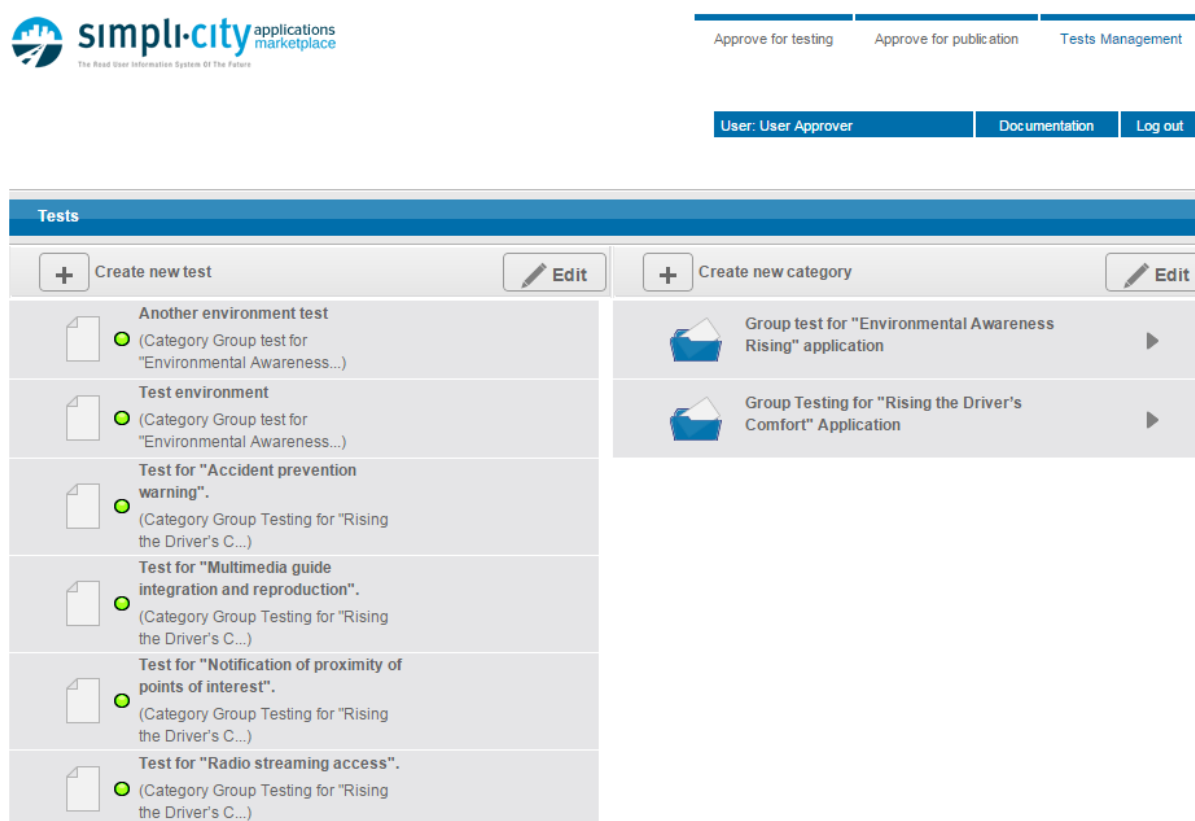


Figure 7: View of Tests Management from the Approver Role’s Perspective

5.2 App Marketplace PMA-UI

The App Marketplace PMA-UI will allow the user to navigate through apps in the App Marketplace in a mobile-friendly way.

In order to navigate there, a default user “admin” with the password “admin” is automatically created.

Figure 8 shows the home screen with the “half menu bar” displayed. In the home screen, apps can be ordered by the ratings of the users or for the time of publication. The menu contains several icons representing specific screens:

- The House represents the home screen
- The Calendar opens the latest changes screen
- The List icon opens the Categories Screen
- The Cloud represents the My Apps Screen
- The Spinning Arrows icon opens the Update Apps Screen

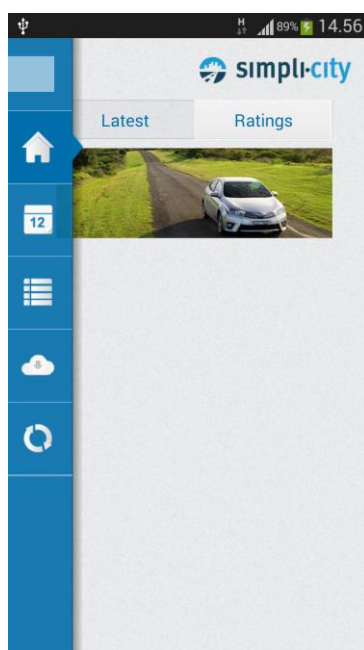


Figure 8: Home Screen with Half Menu Displayed

The description of each icon on the “half menu bar” can be seen when opening the full menu bar as shown in Figure 9. This figure also shows a search box with a magnifying glass that allows searching for apps.

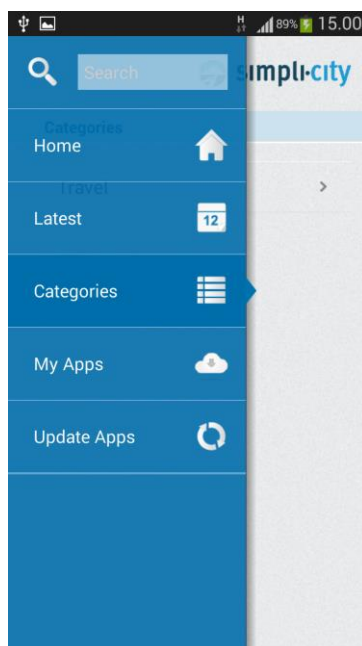


Figure 9: Screen with Full Menu Displayed

Figure 10 shows a list of apps ordered by their publication date. This list shows the name of the app and the valoration made by the customers.

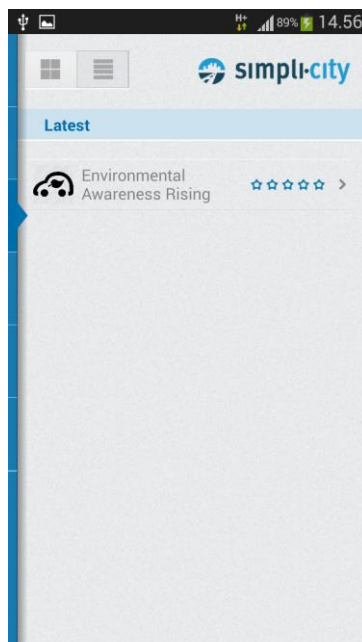


Figure 10: Latest Screen

Figure 11 shows the detail of the app selected. In the upper part details of the app (e.g., the icon, name, average rating, the owner and a button to install it on the device) is shown. The middle part contains a description of the app. In the bottom part of the screenshot shows the media (photos and videos) provided by the owner of the app.

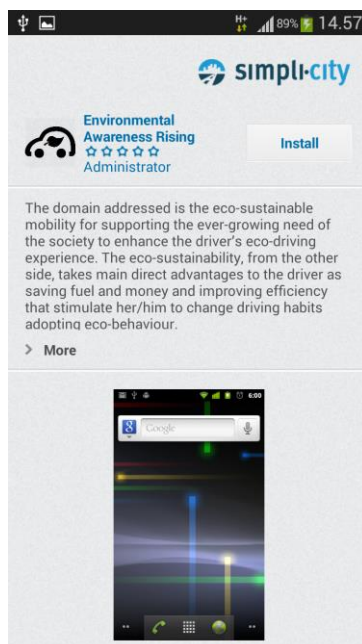


Figure 11: Detail Screen with Description and Media

After scrolling down the app looks like depicted in Figure 12. The Reviews section and a button to report a problem in the app is visible. Additionally some meta-data of the app (e.g., version, size and publication date) is shown.

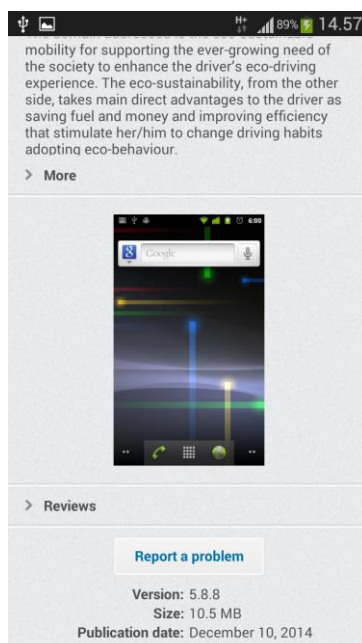


Figure 12: Detail Screen with Media and Extra Information

The Reviews screen shows all the comments and ratings of app users. If not, it is shown empty, as seen in Figure 13.

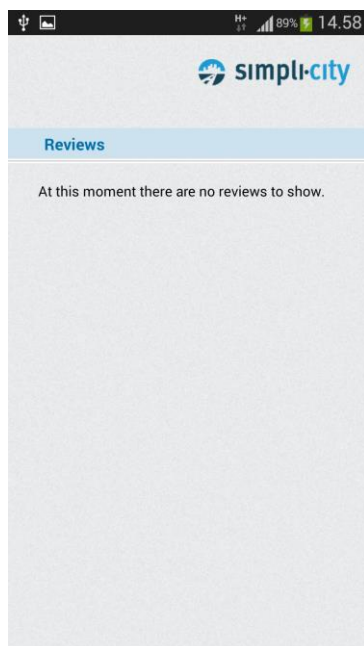


Figure 13: Reviews Screen

Figure 14 shows the applications grouped by category. In the case of the screenshot only Travel apps are published in the App Marketplace. Therefore only one category is shown, but more can be added when using the Marketplace.

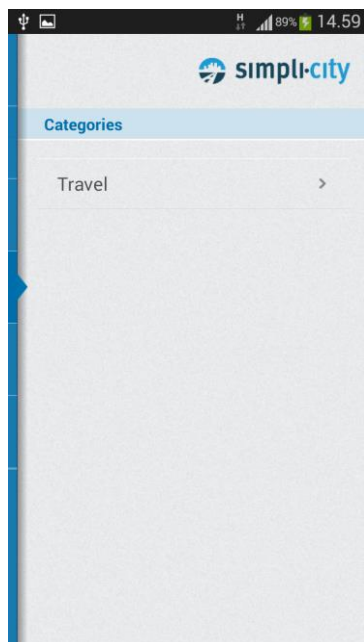


Figure 14: Categories Screen

Figure 15 shows a list or a grid of apps installed on the device, and can be extended to see the details and a button to remove installed apps from the device.

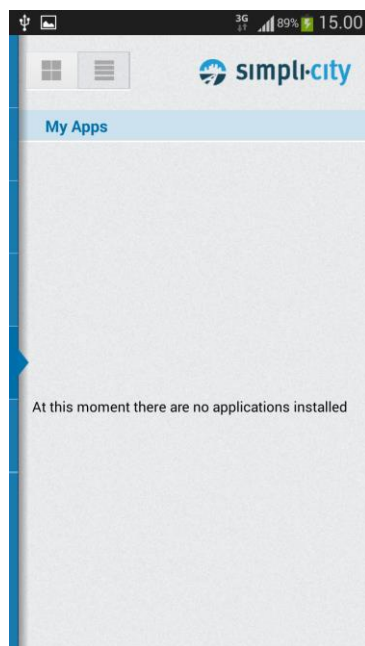


Figure 15: MyApps Screen

Figure 16 shows a list of apps installed in the device that have newer versions than the version installed on the device.

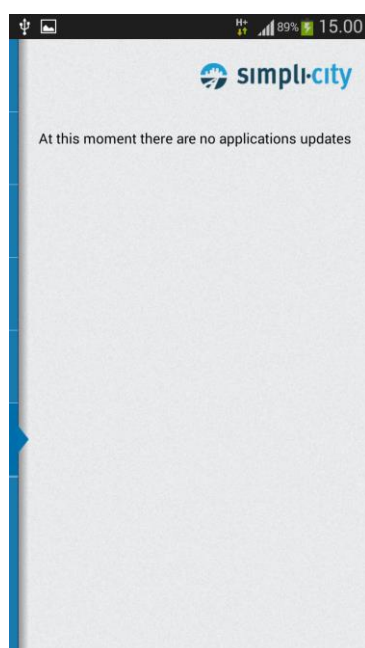


Figure 16: Update Apps Screen

5.3 Service Marketplace

The Service Marketplace needs to be opened inside a web browser. The instructions within Section 4.3 described how to deploy the Service Marketplace. If followed correctly, the Service Marketplace is available by opening the URL <http://localhost:3000>.

Figure 17 shows exactly what is to be seen upon opening the URL. All available services are listed inside a grid. Considering this as the starting point, visitors are able to:

- Login with their Credentials
- Search for Services and Tags
- Look at detailed information about services
- When logged in, the possibility to create new licenses for services
- When logged in and creator of a service, the possibility to change detailed information

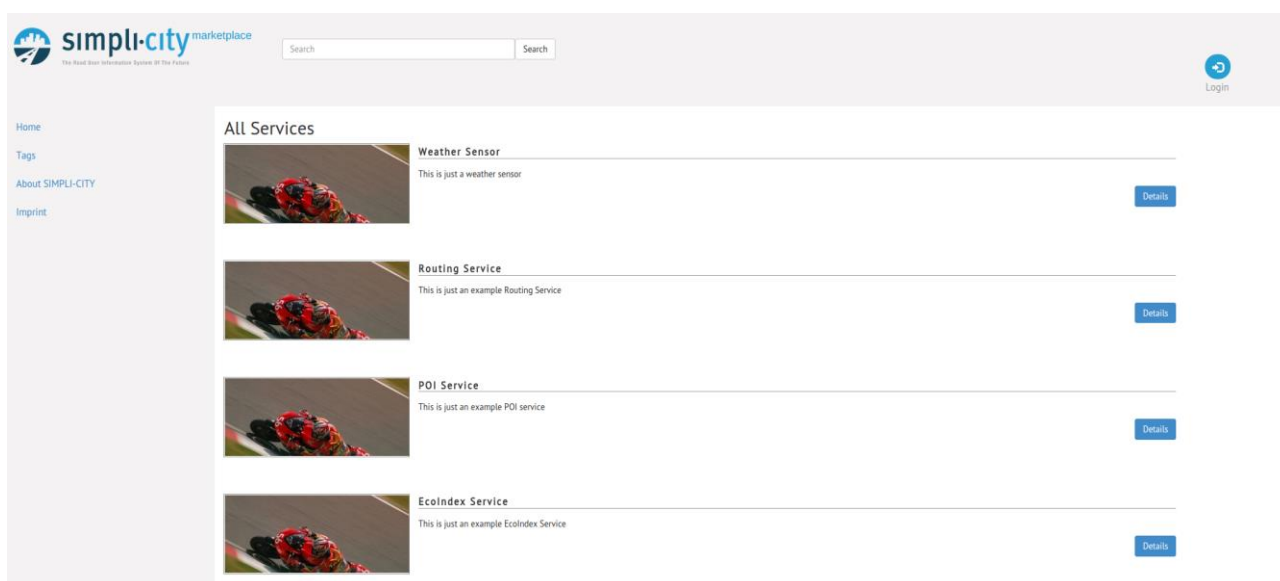


Figure 17: Overview over All SIMPLI-CITY Services

Figure 17 shows the general overview of the SIMPLI-CITY Service Marketplace and is the central starting point for every following action for visitors.

Figure 18: Service Marketplace Login Page

To log into the SIMPLI-CITY Service Marketplace (see Figure 18), a visitor needs to have valid credentials on the Service Registry. How these can be obtained or created is part of D5.3.2, the Service Registry Prototype. Upon signing in, the upper right corner of the view will change into a logout button.

Figure 19: Detailed View of a Service

Figure 19 shows the detail view for a single service, which contains a further description of the service itself and a list of tags given by the service author. If the visitor is registered and logged in, he also may create a new license or view existing licenses for this particular service.

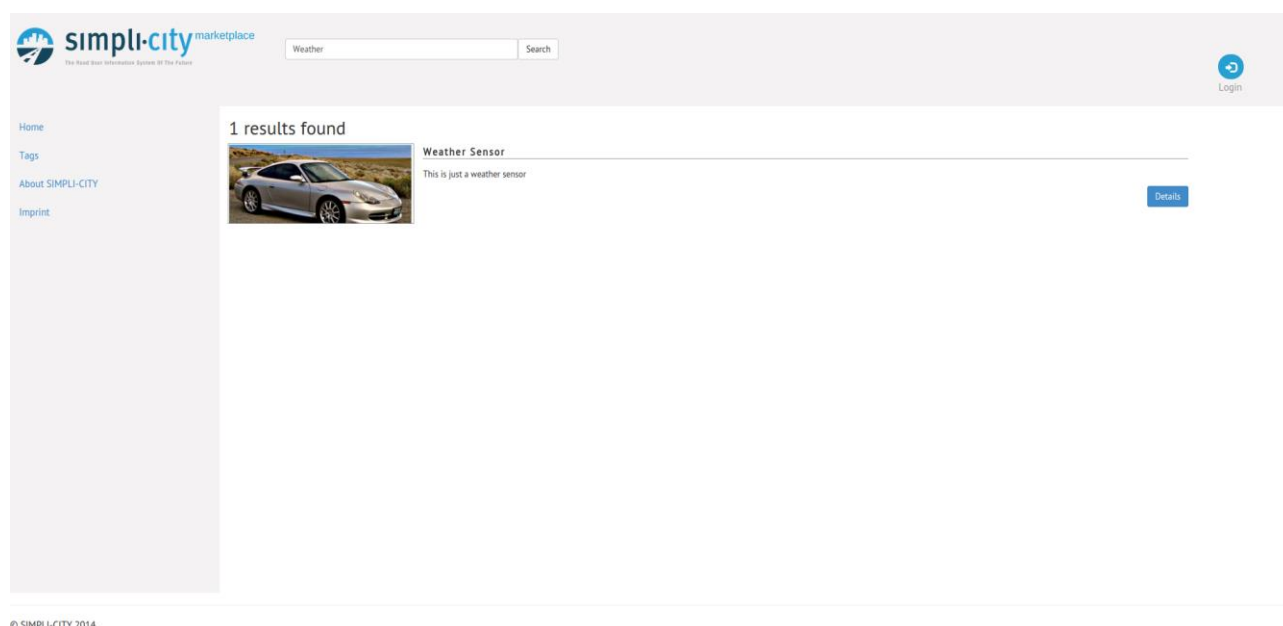


Figure 20: Search Result View

The Service Marketplace is searchable, which makes service discovery easier for app and service developers. After entering a search term in the textbox on top of the page the Service Registry is crawled and services that match the given term are shown. The term can be found in the name, description or as a specific tag.

6 Summary

The Mobility Service and Application Marketplaces Prototypes contain all functionalities that are needed to ease the life of users, app developers as well as service developers. The PMA App and the Web-UI of the App Marketplace contain all means to provide users of SIMPLI-CITY with a good selection of apps and offer a reliable and straight-forward UI to extend the functionality of the PMA.

Service developers benefit from the integration of the Service Registry and the Service Marketplace by having direct access to services and the information stored for these services.