

COMPETITIVENESS AND INNOVATION FRAMEWORK PROGRAMME

CIP-ICT-PSP-2013-7 Pilot Type B



FARM-ORIENTED OPEN DATA **IN EUROPE**

WP3 – Service platform integration and deployment in
cloud infrastructure

D3.5.1 Marketplace

Deliverable Lead: PSNC

Deliverable due date: 29/02/2016

Actual submission date: 29/02/2016

Version: 1.1



Document Control Page

Title	D3.5.1 Marketplace
Creator	Maciej Łabędzki (PSNC), Patryk Promiński (PSNC), Adam Rybicki (PSNC)
Description	This document describes the Marketplace application as a component of a bigger ecosystem called "FOODIE Cloud Platform". The document explains architectural as well as implementation issues.
Publisher	FOODIE Consortium
Contributors	Maciej Łabędzki (PSNC), Patryk Promiński (PSNC), Adam Rybicki (PSNC)
Creation date	10/02/2016
Type	Text
Language	en-GB
Rights	copyright "FOODIE Consortium"
Audience	<input type="checkbox"/> internal <input checked="" type="checkbox"/> public <input type="checkbox"/> restricted
Review status	<input type="checkbox"/> Draft <input type="checkbox"/> WP leader accepted <input type="checkbox"/> Technical Manager accepted <input checked="" type="checkbox"/> Coordinator accepted
Action requested	<input type="checkbox"/> to be revised by Partners <input type="checkbox"/> for approval by the WP leader <input type="checkbox"/> for approval by the Technical Committee <input type="checkbox"/> for approval by the Project Coordinator
Requested deadline	

STATEMENT FOR OPEN DOCUMENTS

(c) 2016 FOODIE Consortium

The *FOODIE* Consortium (<http://www.foodie-project.eu>) grants third parties the right to use and distribute all or parts of this document, provided that the *FOODIE* project and the document are properly referenced.

THIS DOCUMENT IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. EXCEPT WHAT SET FORTH BY MANDATORY PROVISIONS OF LAW IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS DOCUMENT, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

About the project

FOODIE project aims at creating a platform hub on the cloud where spatial and non-spatial data related to agricultural sector is available for agri-food stakeholders groups and interoperable. It will offer: an infrastructure for the building of an interacting and collaborative network; the integration of existing open datasets related to agriculture; data publication and data linking of external agriculture data sources, providing specific and high-value applications and services for the support of planning and decision-making processes.

FOODIE project is addressed to four basic groups of users: a) stakeholders from the agriculture sector as end-users of final applications, b) public sector for communication with farmers about taxation, subsidies and regulation, c) researchers for large scale experimentation on real data and d) ICT companies for the development of new applications for agriculture and food sector, mainly using implemented tools

FOODIE specifically works on three pilots:

- Pilot 1: Precision Viticulture (Spain) will focus on the appropriate management of the inherent variability of crops,
- Pilot 2: Open Data for Strategic and Tactical Planning (Czech Republic) will focus on improving future management of agricultural companies (farms) by introducing new tools and management methods,
- Pilot 3: Technology allows integration of logistics via service providers and farm management including traceability (Germany).

Contact information

Miguel Angel Esbri

Project Coordinator

Atos Spain, Madrid, Spain

E-mail: miguel.esbri@atos.net

URL: <http://www.foodie-project.eu>

Twitter: https://twitter.com/FOODIE_Project

Table of Contents

Glossary	6
Executive Summary	7
1 Introduction	8
2 Functionality	9
3 Architecture	10
3.1 Context	10
3.2 Containers.....	11
3.3 Components	12
4 Implementation	14
5 Working instances	16

Index of Figures

Figure 1 FOODIE goals influencing marketplace service	8
Figure 2 Marketplace context architecture	11
Figure 3 Marketplace containers architecture	12
Figure 4 Marketplace components architecture	13

Glossary

The glossary of terms used in this deliverable can be found in the public document “FOODIE_Glossary.pdf” available at: <http://www.foodie-project.eu>

Executive Summary

This document introduces the Marketplace application as a component of the “FOODIE Cloud Platform” ecosystem. Topics covered here include

- Architecture according to Simon’s Brown C4 methodology;
- Used frameworks and implemented solutions;
- Developed end-user’s functionality;
- Deployed resources.

1 Introduction

One of the key objectives of FOODIE project is the development of a marketplace for the agri-food sector, which can provide a virtual space to connect consumers and producers of agricultural data and applications from disparate locations. On the one hand, this online marketplace will enable farmers and other stakeholders to find information available in our platform, such as reports and datasets, and to provide their own data, for instance, by providing an URL to their data services or by uploading data documents (e.g., spread sheets with statistical information). On the other hand, users will be able to easily find relevant services and tools in our platform to address their needs, and they will be able also to submit their own applications. Moreover, it was envisioned that marketplace will provide free access to the users to some of the available resources, e.g., reports, basic tools, but also will enable providers to charge for more detailed information or enhanced options. The final business model for the marketplace is now being agreed as part of the Sustainability tasks in FOODIE (WP6).

To better understand the Marketplace requirements, it may helpful first to understand the role it plays within FOODIE platform. Taking as starting point FOODIE goals and assumptions, we derived general functional and non-functional requirements for Marketplace service. In particular, the expected outcomes from FOODIE project that influence Marketplace include (in bold as depicted in Figure 1):

1. **effective collaboration among groups in agri-food industry**
2. **a network of interacting organizations**
3. **building an open and interoperable agricultural specialized platform hub on the cloud**
4. management of spatial and non-spatial data relevant for farming production;
5. integration of existing and valuable European open datasets related to agriculture;
6. **data publication and data linking**
7. **provide specific and high-value applications and services for the support in the planning and decision-making**

These requirements were also cross-validated with the Pilots requirements document to make sure that they were in line, or didn't misinterpret, the vision of our first end-users.

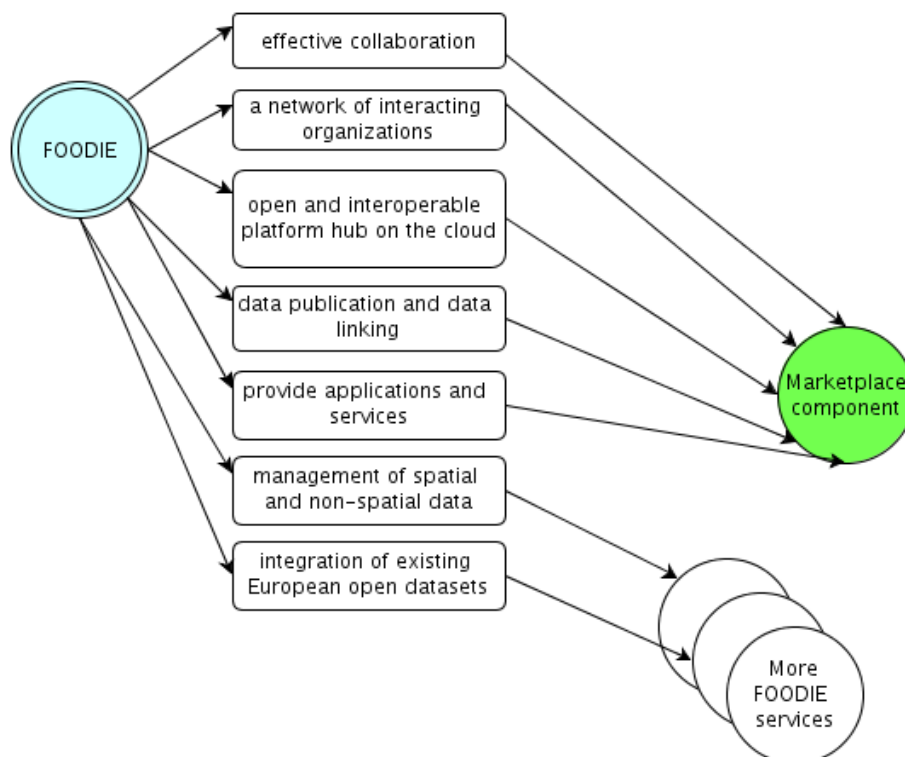


Figure 1 FOODIE goals influencing marketplace service

2 Functionality

The Marketplace is based on e-commerce solution Brodleaf Commerce and thereby it is equipped with all the basic functionality of typical e-commerce platform out-of-the-box. The high-level features, worth to be mentioned, are:

- Search & Browse
- Shopping Cart
- Making Orders
- Administration:
 - Offer Management
 - Order Management
 - Customer Management
- Social features
 - Share products on social media sites like facebook, twitter, google+ and linked-in.

In addition, multiple extensions have been done in order to fit better to FOODIE cloud and agriculture domain. The most relevant that were implemented so far include:

- Support for custom product types related to agriculture industry
- Product description based on Dublin core metadata schema
- Integration with Semantic Annotation Service
- Tag cloud based navigation
- Hierarchical topics for categorizing products and thus facilitating navigation
- Requesting for new products and categories. This also includes the possibility of having discussion about such request (social aspect)
- UI layout which fits to FOODIE template

3 Architecture

In this section we describe the architecture of the FOODIE Marketplace project. For this task we use the C4 approach invented and described by Simon Brown. More details about this approach can be found here: <http://codingthearchitecture.com>.

3.1 Context

From the most high-level perspective Marketplace environment (so-called context) consists of the following elements.

Marketplace

A web application, part of bigger ecosystem - FOODIE cloud platform, allowing to present resources to the community in form of products offer. Marketplace provides tools for offer management, product access and communication between users.

FOODIE Cloud platform

Set of hardware and software components that, either provide functionality directly to end users, or deliver services to other components, for example Marketplace, and thus indirectly provide functionality to end users.

Community

Farmers, IT and agriculture industry entrepreneurs, analysts and consultants. People, who build the community, use Marketplace to exchange resources.

Resources (products)

'Digital' goods published by community to community via Marketplace web platform. For example, applications, services, reports, documents, knowledge and many others.

External applications

Any software that is able to utilize Marketplace machine-to-machine interface. For example, remote systems capable to get and process information about Marketplace resources or systems able to feed Marketplace with resources.

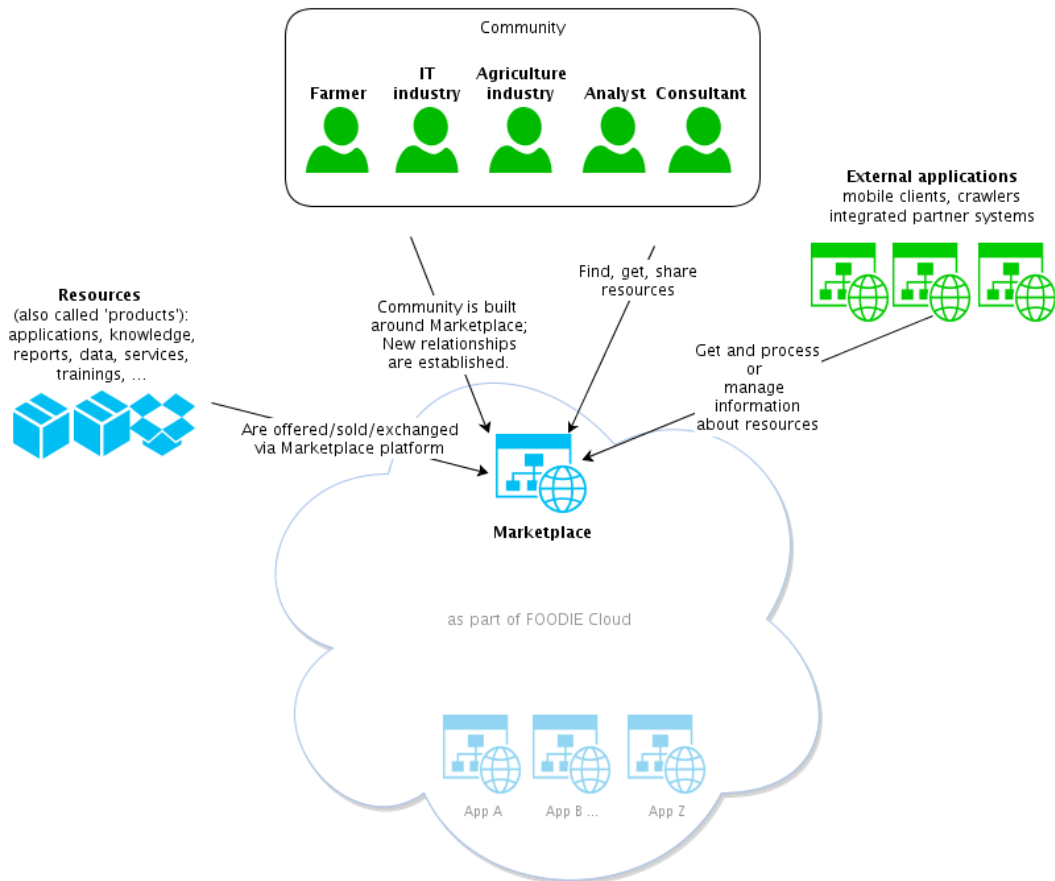


Figure 2 Marketplace context architecture

3.2 Containers

There are two main components in FOODIE Marketplace: User Portal (UP) and Admin Portal (AP). They are Java Web Applications compiled to separate WAR files, and could be deployed on any of Java Web Application container (tested for Apache Tomcat). Both components need a common instance of MySQL Database. In addition, User Portal needs the Apache Solr instance to index the products catalogue. It can be either standalone Apache Solr server (recommended for production infrastructure) or a built-in in User Portal (this setup might be used in test/demo instances). The functionality of both portals is exposed to the users over HTTPS protocol.

User and Admin Portals uses FOODIE Annotation Service (AS) to provide extra features to the users. Although AS is not mandatory for Marketplace to work, it is strongly recommended to ensure connectivity to it from Portals over HTTP protocols.

Deployment strategy

All the four elements of FOODIE Marketplace (User Portal, Admin Portal, Database, Apache Solr) might be installed either on the same physical/virtual machine or on separate ones. The important thing is to ensure JDBC connectivity from User Portal to Database and from Admin Portal to Database. Another restriction is to ensure fast http connection between User Portal and Apache Solr.

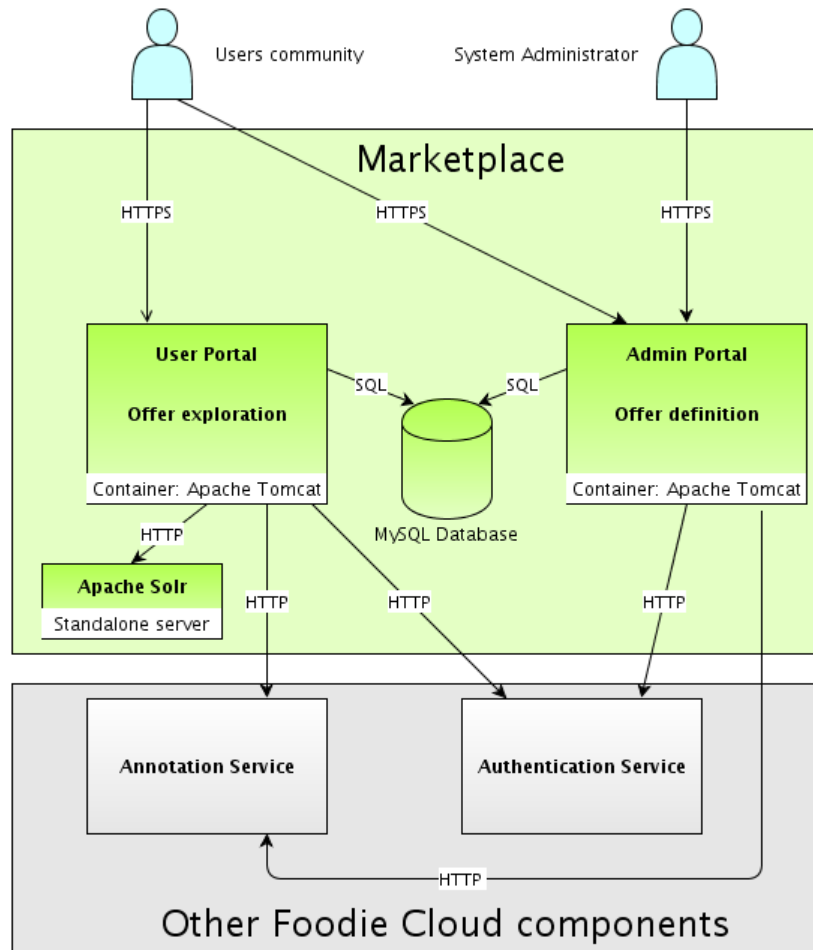


Figure 3 Marketplace containers architecture

3.3 Components

There are 5 functional components that deployed together provides Marketplace functionality. The 'greyed-out' blocks are external systems from Marketplace perspective. Both services (Authentication- and Annotation Services) are developed as independent components of the FOODIE cloud ecosystem.

Basically, **User Portal** provides functionality for users who explore Marketplace to get resources (products). It is a front-end component.

The **Admin Portal** provides all the 'management' functionality, including offer definition and system resources administration. It's a back-end component.

By definition, the **database** persists the offer, not the resources itself. However, storage of relatively small items (e.g. PDF documents) is allowed.

Index engine is responsible for speeding-up the offer exploration and searching. Indexes, such as SOLR, provide optimal query performance when comparing to relational databases.

The **Annotation Service** provides engine for annotating text. Annotations here are understood as semantic-enabled keywords identified for a particular text (e.g. product description). Annotations, as keywords, can be used to search for 'similar' products. Moreover, annotations can have multi-lingual representation (translations) related to a single meaning.

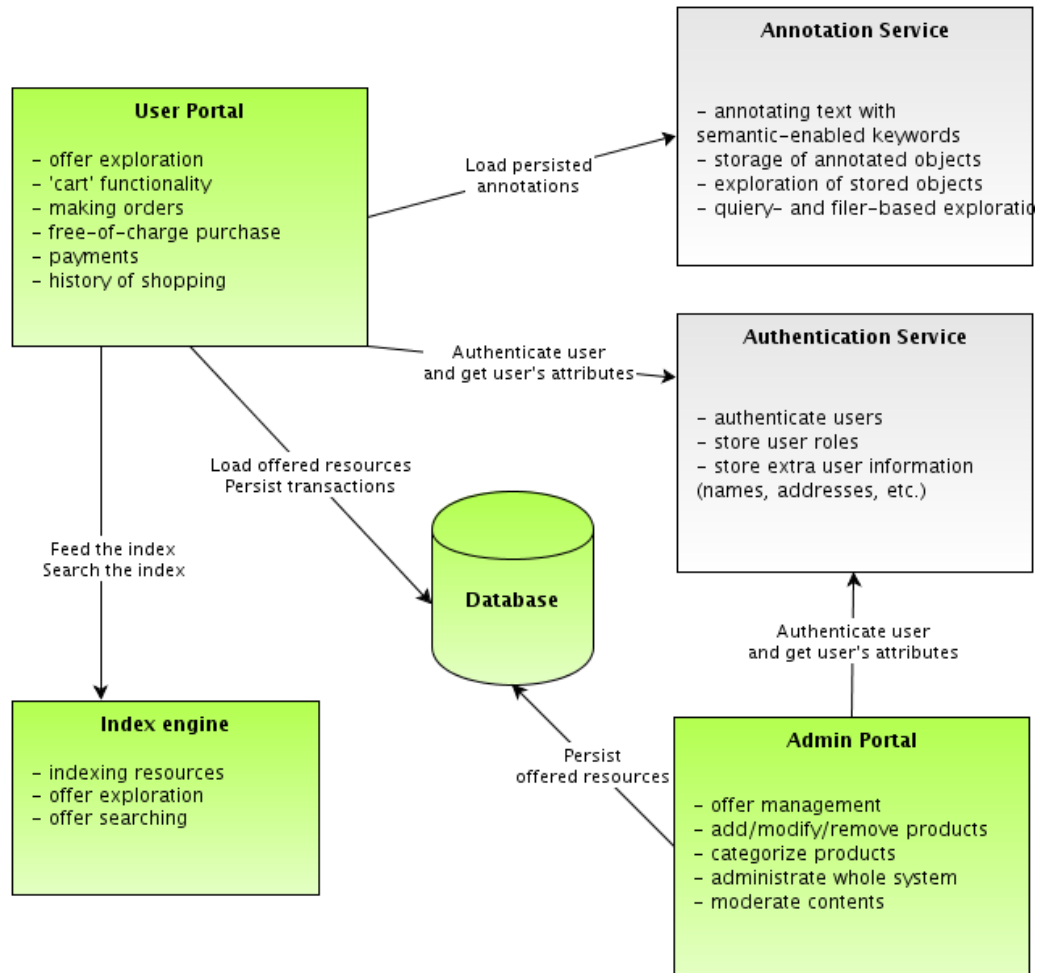


Figure 4 Marketplace components architecture

4 Implementation

Implementation of Marketplace is based on well-known, mature and robust technology stack. Below, there are short descriptions of the most important frameworks which should help in understanding why they are here and why they are a good choice.

Java

(source: <https://www.java.com/en/about/>)

Java is the foundation for virtually every type of networked application and is the global standard for developing and delivering embedded and mobile applications, games, Web-based content, and enterprise software. With more than 9 million developers worldwide, Java enables you to efficiently develop, deploy and use exciting applications and services.

Broadleaf Commerce

Marketplace implementation is strongly based on an ecommerce framework Broadleaf Commerce.

Broadleaf describes itself as follows (source: <http://www.broadleafcommerce.com/features>):

An enterprise solution feature set ideal for Fortune 500 needs, Broadleaf provides the most sought after functionality for supporting B2C, B2B, and B2B2C eCommerce at the best value in the market. Every solution can be customized to ensure your eCommerce site is tailored to your specific requirements. Robust functionality within a lightweight framework lend to some of the characteristics that cause Broadleaf to stand out from the rest. Never feel restricted by a features list again.

Broadleaf Commerce is declared as a scalable solution, free for small and medium projects. However, they offer a paid support and implementation extensions for business that grows-up.

Spring Framework

(source: <https://projects.spring.io/spring-framework/>)

The Spring Framework provides a comprehensive programming and configuration model for modern Java-based enterprise applications - on any kind of deployment platform. A key element of Spring is infrastructural support at the application level: Spring focuses on the "plumbing" of enterprise applications so that teams can focus on application-level business logic, without unnecessary ties to specific deployment environments.

Hibernate

(source: <http://hibernate.org/>)

Hibernate an open source Java persistence framework project. Perform powerful object relational mapping and query databases using HQL and SQL. Hibernate ORM enables developers to more easily write applications whose data outlives the application process. As an Object/Relational Mapping (ORM) framework, Hibernate is concerned with data persistence as it applies to relational databases (via JDBC).

Thymeleaf

(source: <http://www.thymeleaf.org/>)

Thymeleaf is a Java library. It is a template engine capable of processing and generating HTML, XML, JavaScript, CSS and text, and can work both in web and non-web environments. It is better suited for serving the view layer of web applications, but it can process files in many formats, even in offline environments.

It provides an optional module for integration with Spring MVC, so that you can use it as a complete substitute of JSP in your applications made with this technology, even with HTML5.

Solr

(source: <http://lucene.apache.org/solr/>)

Solr is the popular, blazing-fast, open source enterprise search platform built on Apache Lucene™. Solr is highly reliable, scalable and fault tolerant, providing distributed indexing, replication and load-balanced querying, au-

tomated failover and recovery, centralized configuration and more. Solr powers the search and navigation features of many of the world's largest internet sites.

Jersey

(source: <https://jersey.java.net/>)

Jersey RESTful Web Services framework is open source, production quality, framework for developing RESTful Web Services in Java that provides support for JAX-RS APIs and serves as a JAX-RS (JSR 311 & JSR 339) Reference Implementation. Jersey framework is more than the JAX-RS Reference Implementation. Jersey provides its own API that extend the JAX-RS toolkit with additional features and utilities to further simplify RESTful service and client development. Jersey also exposes numerous extension SPIs so that developers may extend Jersey to best suit their needs.

5 Working instances

There are two working instances of Marketplace.

- <https://foodie-cloud.org/marketplace> - the global demo which illustrates how the farmers from different countries can cooperate thanks to FOODIE Marketplace
- <https://foodie-vm3.man.poznan.pl/wodr-marketplace> - the local demo fulfilling specific polish farmers' needs. It has been instantiated in cooperation with WODR. It is an example of how FOODIE Marketplace can be adapted to local market specific conditions on a country level.

These instances are concretely on the virtual machine 6 of FOODIE cloud.

A user guide for marketplace is available as a video at: <https://youtu.be/fpCrtN8QOzw>