Deliverable D200.7

Security, Privacy & Trust (SPT) component

WP 200

Project Acronym & Number:	FIspace – 604 123
Project Title:	FIspace: Future Internet Business Collaboration Networks in Agri-Food, Transport and Logistics
Funding Scheme:	Collaborative Project - Large-scale Integrated Project (IP)
Latest version of Annex 1:	2013-10-03
Start date of the project:	01.04.2013
Duration:	24
Status:	Draft
Editor:	Said Rahma (ATOS)
Contributors (to the "R" part of the deliverable[1]; ordered by project partner)	**ATOS**: Said Rahma Rodriguez **KOC**: Serdar Arslan, Engin Dağdeviren, Seyhun Mehmet Futacı
Document Identifier:	FIspace-D200.7-FIspace_Integrated_Release_V3-SPT-v0.4.docx
Date:	27.02.2015
Revision:	004
Project website address:	http://www.FIspace.eu

[1] Contributors to FIspace code ("P") include ATB, UDE, IBM, ATOS, KOC, TOG, AST, NKUA, UPM and LimeTri; contributing persons are listed at https://bitbucket.org/fispace/profile/members

The FIspace Project

Leveraging on outcomes of two complementary Phase 1 use case projects (FInest & SmartAgriFood), aim of FIspace is to pioneer towards fundamental changes on how collaborative business networks will work in future. FIspace will develop a multi-domain Business Collaboration Space (short: FIspace) that employs FI technologies for enabling seamless collaboration in open, cross-organizational business networks, establish eight working Experimentation Sites in Europe where Pilot Applications are tested in Early Trials for Agri-Food, Transport & Logistics and prepare for industrial uptake by engaging with players & associations from relevant industry sectors and IT industry.

Project Summary

As a use case project in Phase 2 of the FI PPP, FIspace aims at developing and validating novel Future-Internet-enabled solutions to address the pressing challenges arising in collaborative business networks, focussing on use cases from the Agri-Food, Transport and Logistics industries. FIspace will focus on exploiting, incorporating and validating the Generic Enablers provided by the FI PPP Core Platform with the aim of realising an extensible collaboration service for business networks together with a set of innovative test applications that allow for radical improvements in how networked businesses can work in the future. Those solutions will be demonstrated and tested through early trials on experimentation sites across Europe. The project results will be open to the FI PPP program and the general public, and the pro-active engagement of larger user communities and external solution providers will foster innovation and industrial uptake planned for Phase 3 of the FI PPP.

Project Consortium

- DLO; Netherlands
- ATB Bremen; Germany
- IBM; Israel
- KocSistem; Turkey
- Aston University; United Kingdom
- ENoLL; Belgium
- KTBL; Germany
- NKUA; Greece
- Wageningen University; Netherlands
- PlusFresc; Spain
- FloriCode; Netherlands
- Kverneland; Netherlands
- North Sea Container Line; Norway
- LimeTri; Netherlands
- BO-MO; Slovenia
- MOBICS; Greece
- Fraunhofer IML; Germany
- Q-ray; Netherlands
- FINCONS; Italy
- Kühne + Nagel; Switzerland
- University Duisburg Essen; Germany
- ATOS; Spain
- The Open Group; United Kingdom
- CentMa; Germany
- iMinds; Belgium
- Marintek; Norway
- University Politecnica Madrid; Spain
- Arcelik; Turkey
- EuroPoolSystem; Germany
- GS1 Germany; Germany
- Mieloo & Alexander; Netherlands
- OPEKEPE; Greece
- Innovators; Greece
- CIT; Spain
- SDZ; Germany
- Snoopmedia; Germany
- EECC; Germany
- CBT; Spain

More Information

Harald Sundmaeker (coordinator) e-mail: sundmaeker@atb-bremen.de
Bert Vermeer (deputy coordinator) e-mail: bert.vermeer@wur.nl
Project Website Web link: http://www.fispace.eu/

Dissemination Level

PU	Public	
PP	Restricted to other programme participants (including the Commission Services)	**X**
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Change History

Version	Notes	Date
001	Creation of the document SPT component	09.12.2014
002	Update of the contents, update the overview section to focus on the new approach of SPT related to use Keycloak technology Update the internal SPT REST API, Information model and Interaction model sections Minor changes related to rephrasing and typo error Update of the references table	23.01.2015
003	Internal review process, checking URL links to the FIspace Web online documentation Update of the abbreviation table, update of the references table Final version ready for submission of approved document	06.02.2015
004	Update of the coordinator information in the section "*More Information*" Added FIspace development repository and documentation references, formatting improvement Final version ready for submission to EC	27.02.2015
005		
006		

Abbreviations

AAA	Authentication, Authorisation, and Accounting
ACSI	Artifact-Centric Service Interoperation
AdvB	Advisory Board
AJAX	Asynchronous JavaScript + XML
API	Application Programming Interface
App	Software Application
B2B	Business-to-business
B2C	Business-to-Consumer
BCM	Business Collaboration Module in FIspace
BCO	Business Collaboration Objects in FIspace
BE	Business Entities
BPPC	Business Process Participant Configuration
BSS	Business Support Systems
CDR	Charging Detailed Records
CEP	Complex Event Processing
CSB	Cloud Service Bus
CSS	Cascading Style Sheets
CSV	Comma-Separated Values
D	Deliverable
DAO	Data Access Object
DB	Database
DoW	Description of Work
EC	European Commission
EDI	Electronic Data Interchange
EE	Experimentation Environment
e.g.	Exempli gratia = for example
EPA	Event Processing Agent
EPM	Event Processing Module in FIspace
ESB	Enterprise Service Bus
EU	European Union
FIA	Future Internet Assembly
FI-PPP	Future Internet Public Private Partnership
FP7	Framework Programme 7
GA	Grant Agreement
GE	Generic Enabler
GUI	Graphical User Interface
HTML	HyperText Markup Language
IaaS	Infrastructure as a Service
ICT	Information and Communication Technology
IDE	Integrated Development Environment
IDM	Identity Management
i.e.	id est = that is to say
IE	Integration Environment
IEC	International Electrotechnical Commission
IETF	Internet Engineering Task Force
I/O	Input / Output
IoT	Internet of Things
IP	Intellectual Property
IP (protocol)	Internet Protocol
IPR	Intellectual Property Rights
IPsec	Internet Protocol Security
IT	Information Technology
ITU	International Telecommunication Union
ISO	International Standardization Organisation
J2SE	Java 2 Platform, Standard Edition
JDK	Java Development Kit
JDT	Related to Eclipse Java Development Tools
JMX	Java Management Extensions
JRE	Java Runtime Environment
JS	JavaScript
JSON	JavaScript Object Notation
JSP	Java Server Page
JVM	Java Virtual Machine
KPI	Key Performance Indicator
LPA	Logistics Planning Application
M	Month
MTBF	Mean Time Between Failures
MVC	Model–View–Controller
OASIS	Organization for the Advancement of Structured Information Standards
OAuth	Open standard Authentication protocol
OMG	Object Management Group
OSS	Operational Support Systems
P2P	Peer-to-peer
PaaS	Platform as a Service
PDE	Related to Eclipse Java Development Tools
PE	Production Environment
PIA	Product Information App

PIE	Preliminary Integration Environment
PKI	Public Key Infrastructure
PM	Person Month
POM	Project Object Model (used by maven tools)
Proton	IBM Proactive Technology Online
QoS	Quality of Service
RBAC	Role-Based Access Control
RCP	Rich Client Platform
REST	Representational State Transfer
RFC	Request for Comments
RSS	Revenue Sharing System
RTD	Research and Technological Development
SaaS	Software as a Service
SDI	System and Data Integration layer in FIspace
SDK	Software Development Kit
SME	Small and Medium Sized Enterprise
SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
SOA-RM	(OASIS) Reference Model for Service Oriented Architecture
SPT	Security, Privacy and Trust Framework
SSH	Secure Shell
SSL	Secure Sockets Layer
SSO	Single Sign On
ST	Sub-Task
SWT	Standard Widget Toolkit
T	Task
TCP	Transmission Control Protocol
TIC	Tailored Information for Consumers
TLS	Transport Layer Security
TPM	Transport Planning Module
UAA	User Management, Authentication and Authorisation
UI	User Interface
UML	Unified Modeling Language
URI	Universal Resource Identifier
URL	Universal Resource Locator
USDL	Unified Service Description Language
VM	Virtual Machine
VPN	Virtual Private Network
W3C	World Wide Web Consortium
WADL	Web Application Description Language
WLAN	Wireless Local Area Network
WP	Work Package
WS	Web Service
WSDL	Web Services Description Language
XLS/XLSX	Microsoft Excel file Format
XML	eXtensible Markup Language
XSD	XML Schema Definition

Table of Contents

List of Figures

List of Tables

1 Introduction

This document aims at describing the third release (V3) of the FIspace, encompassing the implementations along with usage guidance and technical documentation of each FIspace component.

It reports on the description concerning the **Security, Privacy & Trust (SPT) core component**, the description of the development and implementation of the SPT core components that is part of the FIspace platform.

The aim of the Security, Privacy & Trust framework of the FIspace platform is to provide secure and reliable access and, where needed, exchange of confidential business information and transactions using secure authentication and authorisation methods that meet required levels of security assurance. Authentication, authorisation and accounting technologies will provide user management & access control features.

The main features of the SPT framework have been driven by an initial analysis of the SPT functionalities that will be required by industrial actors that will be users of the FIspace platform, and industrial technology suppliers who will exploit the FIspace platform to provide Apps and associated services to the industrial actors. The main feature categories that have been considered in the design of the SPT framework for FIspace are:

- **Identity and Trust**: Current situation is that often two business actors establish identity and trust to ex-change information based on some previous knowledge of one another, having been in physical communication. In more advanced and eventually more common scenarios, actors will not be able to rely on having physical contact with other FIspace actors, and strategies such as exploiting online profiles, reputation (ranking), certification or registration data bases, etc. will be supported.
- **Access Control**: This will include features in order to validate a user's identify and thus only allow individuals and organizations that are authorized to connect and that they can only access the information and data they are allowed to access.
- **Authentication**: This will include facilities for authenticating individual users, third-party systems, networked resources, and it will need to go down to fine-grained events, and data objects to ensure that only authentic entities are allowed to connect and communicate with the FIspace platform.
- **Data Security**: Those mechanisms will ensure that data is being encrypted and does not leave the FIspace premises unencrypted, as well as that data can only be accessed by users with the respective credentials.
- **Security Assurance**: FIspace will provide strong security assurance that commercial information and transactions are secure, can be trusted and are not vulnerable to malicious actions. FIspace will use a compositional security assurance and accounting process, separating concerns where possible. In a component based design process, independently developed components are assessed and matched to specific system security requirements to determine if they meet the system security objectives. For independently developed components such as Apps it is possible to provide assurance provided we can verify an App adheres to a set of system-wide and App-specific security policies. As the cost of full verification of independent Apps is costly and time consuming, FIspace comple-

ments the verification of security policy adherence by Apps with monitoring mechanisms to detect and prevent unacceptable or unexpected App behaviour.

- **Developer support** to ensure correct usage of necessary security mechanisms in FIspace: SPT patterns and guidelines underlie the Development Toolkit (see SDK document) to ensure that SPT issues are considered by App developers.

Concerning privacy and data ownership, one important design consideration that should be mentioned is that operational and business data per se is typically not stored persistently in the FIspace platform (i.e., in the Cloud). Rather data resides with the data owner (and on its premises) but FIspace will provide access to this data (programmatic and access rights) to the entities that require to get access to this data. Typically, only "meta-data" such as events about actual data objects that have changed (change event) will be stored and managed by the platform, as well as user registration information.

Online documentation for SPT Framework: http://dev.fispace.eu/doc/wiki/spt

1.1 Scope

The aim of this document is mainly to describe and detail the **FIspace SPT core component** at development and implementation level, giving detailed and technical information related to the design and the implementation as well as information about the related technologies and standard taken as a .reference to build each component.

Along this development activities and tasks, there is a set of resources, online documentation, tutorial and other external resource that refer to the Generic Enablers that can provide more technical information and user guides for the community and people who want to use the FIspace platform for Business collaboration or developers who want to create and develop business application (Apps developer) for a specific domain of application.

Table 1 shows the links to other online resources related to FIspace project and FIWARE.

Description	Link
FIspace Business collaboration web site	http://www.fispace.eu/
FIspace Developer Documentation web site	http://dev.fispace.eu/doc/wiki/Home
FIspace Deliverables web site	http://www.fispace.eu/deliverable.html
FIspace Tutorial web site	http://www.fispace.eu/tutorials.html
FIWARE web site	http://www.fi-ppp.eu/projects/fi-ware/
FIWARE Catalogue of the Generic Enablers (GEs)	http://catalogue.fi-ware.org/
FIWARE community web site	http://www.fi-ware.org/community/

Table 1: Other FIspace and FIWARE resources

Table 2 shows the links to the Wirecloud online documentation.

Description	Link
FIWARE - Catalogue - Application Mashup - Wirecloud	http://catalogue.fi-ware.org/enablers/application-mashup-wirecloud
FIWARE - Catalogue - Application Mashup - Wirecloud Documentation	http://catalogue.fi-ware.org/enablers/application-mashup-wirecloud/documentation
FIWARE - Application Mashup - Wirecloud - User and Programmer Guide	https://forge.fi-ware.org/plugins/mediawiki/wiki/fiware/index.php/Application_Mashup_-_Wirecloud_-_User_and_Programmer_Guide
Dashboard - Wirecloud home page	http://conwet.fi.upm.es/wirecloud/
Dashboard - The WireCloud Mashup Platform	http://conwet.fi.upm.es/docs/display/wirecloud/The+WireCloud+Mashup+Platform
Dashboard - Welcome to CoNWeT-Wirecloud Confluence	http://conwet.fi.upm.es/docs/dashboard.action
Dashboard - User Guide	http://conwet.fi.upm.es/docs/display/wirecloud/WireCloud+User%27s+Guide
Dashboard - WireCloud Installation and Administration Guide	http://conwet.fi.upm.es/docs/display/wirecloud/Wire-Cloud+Installation+and+Administration+Guide

Table 2: Wirecloud online documentation

Table 3 shows the links to the WStore online documentation.

Description	Link
FIWARE - Catalogue - Store - WStore	http://catalogue.fi-ware.org/enablers/store-wstore
FIWARE - Catalogue - Store - WStore Documentation	http://catalogue.fi-ware.org/enablers/store-wstore/documentation
FIWARE - Store - W-Store - User and Programmer Guide	https://forge.fi-ware.org/plugins/mediawiki/wiki/fiware/index.php/Store_-_W-Store_-_User_and_Programmer_Guide
FIWARE - Store - W-Store - Store - W-Store - Installation and Admin-istration Guide	https://forge.fi-ware.org/plugins/mediawiki/wiki/fiware/index.php/Store_-_W-Store_-_Installation_and_Administration_Guide

Table 3: Store online documentation

Table 4 shows the external development tools references.

Description	Link
Java Environment, JVM, JRE, JDK (Oracle)	http://www.oracle.com/technetwork/java/javase/downloads/index.html
Eclipse IDE (Integrated Development Environment)	https://www.eclipse.org/ , https://www.eclipse.org/downloads/
Maven	http://maven.apache.org/ , http://maven.apache.org/download.cgi

Table 4: External development tools references

Table 5 shows the FIspace development repository and documentation references based on the bitbucket tools for collaborative development.

Bitbucket is a hosting site for the distributed version control systems (DVCS) Git (http://git-scm.com/) and Mercurial (http://mercurial.selenic.com/). The service offering includes an issue tracker and wiki, as well as integration with a number of popular services such as Basecamp, Flowdock, and Twitter.

Description	Link
Bitbucket FIspace repository home page	https://bitbucket.org/fispace
Bitbucket FIspace core component home page	https://bitbucket.org/fispace/core/wiki/Home
Bitbucket FIspace Roadmap page	https://bitbucket.org/fispace/core/wiki/roadmap

Table 5: Bitbucket collaborative environment for FIspace development

1.2 Intended audience

The main interest groups of this deliverable are the participating teams and the responsible partners of FIspace project involved in the development activities, setup and preparation of the development phase. This document is relevant to the software engineer, programmers and developers who are the persons directly involved in the development, participating effectively on the design and implementation of the FIspace platform and the underlying components and sub-systems who want to know more about some technical information intrinsic to the FIspace platform.

At the technical level this document is relevant to: system architects; information systems designers; system developers and application developers; software engineers; other audiences who provide design services and applications using relevant standards and the recommendations of standards bodies like IETF, ITU, ISO, W3C, etc.

Partners involved in the integration tasks include: system integrators; people to test, validate and evaluate the FIspace platform and associated systems; can be also interested.

1.3 General remark

This document follows the ISO/IEC Directives, Part 2: Rules for the structure and drafting of International Standards w.r.t. the usage of the word “shall”. The word “shall” (not “must”) is the verb form used to indicate a requirement to be strictly followed to conform to this specification.

This document describes the corresponding core components involved in the FIspace core platform. It presents the development currently done and the corresponding implementation, the main features developed, as well as the related technologies and environment requirements.

In most of the following sections the structure is organized as:

- **Overview**: provides an overall introduction to the component, a description, of the internal architecture and features among other.
- **Interfaces or Application programming interface (API)**: describes the API accessible for the users or entities of the component (typically applications, but a component may also be used by other components).
- **Information model**: describes or specifies the component from an information perspective describing information objects of the component domain.
- **Interaction model**: describes or specifies main usage component “scenarios” associated with the component/GEs, sequence diagrams.
- **High level composite architecture**: describes or shows the main components constituting the set of components (this perspective is optional, since some component consists of only one main component).

Notice that some components only need to describe some of the item above described.

2 Security, Privacy & Trust (SPT)

2.1 Overview

The aim of the Security, Privacy & Trust (SPT) framework of the FIspace platform is to provide secure and reliable access and, where needed, exchange of confidential business information and transactions using secure authentication and authorisation methods that meet required levels of security assurance. Authentication, authorisation and accounting technologies will provide user management & access control features.

Initially; Digital Self GE and Access Control GE were planned to be used for authentication and authorisation. Authentication, user management and authorisation functionalities were tested successfully using these GEs. However when security related GEs became obsolete then security team initiated the mitigation plan to provide the Security, Privacy and Trust component.

Mitigation plan consist of open-source solution Keycloak [32] (http://keycloak.jboss.org/). It basically provides Integrated Single Sign-On (SSO) and Identity Management (IDM) for browser apps and RESTful web services. Built on top of the OAuth 2.0 [33], Open ID Connect, JSON Web Token (JWT) and SAML 2.0 specifications. Options are to deploy it with an existing app server, as a black-box appliance, or as an Openshift cloud service and/or cartridge.

Role Based Access Control functionalities are provided under the “realm” - concept of a Keycloak. A realm secures and manages security metadata for a set of users, applications, and registered OAuth clients. Users can be created within a specific realm within the Administration console. Roles (permission types) can be defined at the realm level and you can also set up user role mappings to assign these permissions to specific users.

SPT package is being used by FIspace Front-End, and Single Sign-On functionality provides seamless access between the Front-End, WireCloud and WStore components. User management is being done federatively by SPT package.

In WP300, the task T350 “*Experimentation Environment’s web application*” was fully integrated with this new SPT package for the user authentication. Associated roles defined for the environment and UI is customized based on the role of the authenticated user.

Integration of this new SPT package to Logistics Planning Application is also successfully completed. Concerning the RBAC system, security policies have been defined based on the use case scenarios and these roles are currently used for in Logistics Planning Application (LPA).

SPT package currently provides major security functionalities including; user authentication, SSO and Single Log Out for browser applications, user registration, forgot password support (user can have an email sent to them to reset password), user session management. Admin can view user sessions and what applications/clients have an access token. Sessions can be invalidated per realm or per user.

2.2 Interfaces / API

Currently RESTful APIs are directly provided by SPT package. These APIs can be used fully by the service applications and only internal applications can use these APIs.

Table 6 presents the list of API related to the user, authentication and role management which is based on the Keycloak API and underlying functionalities and capabilities.

Method	URL	Description
GET	/admin/realms/fispace/applications	List of applications belonging to this realm.
POST	/admin/realms/fispace/applications	Create a new application.
PUT	/admin/realms/fispace/applications/{app-name}	Update the application.
GET	/admin/realms/fispace/applications/{app-name}	Get representation of the application.
DELETE	/admin/realms/fispace/applications/{app-name}	Delete this application.
GET	/admin/realms/fispace/applications/{app-name}/allowed-origins	Returns set of allowed origin.
PUT	/admin/realms/fispace/applications/{app-name}/allowed-origins	Change the set of allowed origins.
DELETE	/admin/realms/fispace/applications/{app-name}/allowed-origins	Remove set of allowed origins from current allowed origins list.
POST	/admin/realms/fispace/applications/{app-name}/client-secret	Generates a new secret for this application
GET	/admin/realms/fispace/applications/{app-name}/client-secret	Get the secret of this application
GET	/admin/realms/fispace/applications/{app-name}/installation/json	Return keycloak.json file for this application to be used to configure the adapter of that application.
GET	/admin/realms/fispace/applications/{app-name}/roles	List all roles for this realm or application
POST	/admin/realms/fispace/applications/{app-name}/roles	Create a new role for this realm or application
GET	/admin/realms/fispace/applications/{app-name}/roles/{role-name}	Get a role by name
DELETE	/admin/realms/fispace/applications/{app-name}/roles/{role-name}	Delete a role by name
PUT	/admin/realms/fispace/applications/{app-name}/roles/{role-name}	Update a role by name
GET	/admin/realms/fispace/oauth-clients	Get a list of oauth clients in this realm.
POST	/admin/realms/fispace/oauth-clients	Create an oauth client
PUT	/admin/realms/fispace/oauth-clients/{clientId}	Update the oauth client

Method	URL	Description
GET	/admin/realms/fispace/oauth-clients/{clientId}	Get a representation of the oauth client
DELETE	/admin/realms/fispace/oauth-clients/{clientId}	Remove the OAuth Client
GET	/admin/realms/fispace/oauth-clients/{clientId}/claims	Get the claims a client is allowed to ask for
PUT	/admin/realms/fispace/oauth-clients/{clientId}/claims	Set the cliams a client is allowed to ask for.
POST	/admin/realms/fispace/oauth-clients/{clientId}/client-secret	Generate a new client secret for the oauth client
GET	/admin/realms/fispace/oauth-clients/{clientId}/client-secret	Get the secret of the oauth client
GET	/admin/realms/fispace/roles	List all roles for this realm or application
POST	/admin/realms/fispace/roles	Create a new role for this realm or application
GET	/admin/realms/fispace/roles/{role-name}	Get a role by name
DELETE	/admin/realms/fispace/roles/{role-name}	Delete a role by name
PUT	/admin/realms/fispace/roles/{role-name}	Update a role by name
GET	/admin/realms/fispace/users/{username}/role-mappings	Get role mappings for this user
GET	/admin/realms/fispace/users/{username}/role-mappings/applications/{app}	Get application-level role mappings for this user for a specific app
POST	/admin/realms/fispace/users/{username}/role-mappings/applications/{app}	Add application-level roles to the user role mapping.
DELETE	/admin/realms/fispace/users/{username}/role-mappings/applications/{app}	Delete application-level roles from user role mapping.
GET	/admin/realms/fispace/users/{username}/role-mappings/applications/{app}/available	Get available application-level roles that can be mapped to the user
GET	/admin/realms/fispace/users/{username}/role-mappings/applications/{app}/composite	Get effective application-level role mappings.
GET	/admin/realms/fispace/users/{username}/role-mappings/realm	Get realm-level role mappings for this user
POST	/admin/realms/fispace/users/{username}/role-mappings/realm	Add realm-level role mappings
DELETE	/admin/realms/fispace/users/{username}/role-mappings/realm	Delete realm-level role mappings
GET	/admin/realms/fispace/users/{username}/role-mappings/realm/available	Realm-level roles that can be mapped to this user
GET	/admin/realms/fispace/users/{username}/role-mappings/realm/composite	Effective realm-level role mappings for this user.

Table 6: SPT – API user, authentication and role management

2.3 Information model

This section presents the basic information about the main class and object model managed and used by the SPT component that is essentially based on the Keycloak open source software and the corresponding model that has been integrated in the SPT component. These classes represent the main entities involved to securing the FIspace platform, at user level and communications with external resources and applications, among others, by the way of the following entities:

- Applications, services and API allowed to interact securely with the FIspace platform and underlying components.
- OAuth Clients which provide the security information and credential to authenticate several kind of entities and resources against the FIspace platform.
- Roles which provide the security information allowing to manage the different roles assigned to each entities or resources.
- Users which provide the security information allowing to manage the users information, user data, assigned credentials and roles.

`public class` **`ApplicationRepresentation`**

Modifier and Type	Field and Description
protected String	adminUrl
protected String	baseUrl
protected Boolean	bearerOnly
protected ClaimRepresentation	claims
protected String[]	defaultRoles
protected Boolean	enabled
protected Boolean	fullScopeAllowed
protected String	id
protected String	name
protected Integer	notBefore
protected Boolean	publicClient
protected List<String>	redirectUris
protected String	secret
protected Boolean	surrogateAuthRequired
protected List<String>	webOrigins

public class **OAuthClientRepresentation**

Modifier and Type	Field and Description
protected ClaimRepresentation	claims
protected Boolean	directGrantsOnly
protected Boolean	enabled
protected Boolean	fullScopeAllowed
protected String	id
protected String	name
protected Integer	notBefore
protected Boolean	publicClient
protected List<String>	redirectUris
protected String	secret
protected List<String>	webOrigins

public class **RoleRepresentation**

Modifier and Type	Field and Description
protected boolean	composite
protected RoleRepresentation.Composites	composites
protected String	description
protected String	id
protected String	name

public class **UserRepresentation**

Modifier and Type	Field and Description
protected Map<String,List<String>>	applicationRoles
protected Map<String,String>	attributes
protected List<CredentialRepresentation>	credentials
protected String	email

protected boolean	emailVerified
protected boolean	enabled
protected String	federationLink
protected String	firstName
protected String	id
protected String	lastName
protected List<String>	realmRoles
protected List<String>	requiredActions
protected String	self
protected List<SocialLinkRepresentation>	socialLinks
protected boolean	totp
protected String	username

2.4 Interaction model

SPT uses access tokens to secure web invocations. Access tokens contain security metadata specifying the identity of the user as well as the role mappings for that user. The format of these tokens is a Keycloak extension to the JSON Web Token specification. Each realm has a private and public key pair which it uses to digitally sign the access token using the JSON Web Signature specification. Applications can verify the integrity of the digitally signed access token using the public key of the realm. The protocols used to obtain this token are defined by the OAuth 2.0 specification.

Signed access tokens can also be propagated by REST client requests within an Authorisation header. So, you have a distributed security model that is centrally managed, yet does not require an IDM Server hit per request, only for the initial login.

Each application still ought to have an internal security mechanism to follow up on user sessions etc.

2.5 High level composite architecture

Keycloak provides Single Sign-on functionalities along with the APIs required for the management. Both FIspace platform and external applications running on FIspace platform use the SSO layer to provide secure authentication and retrieve user information. However management APIs are only to be used by the FIspace platform, more specifically FIspace Front-End.

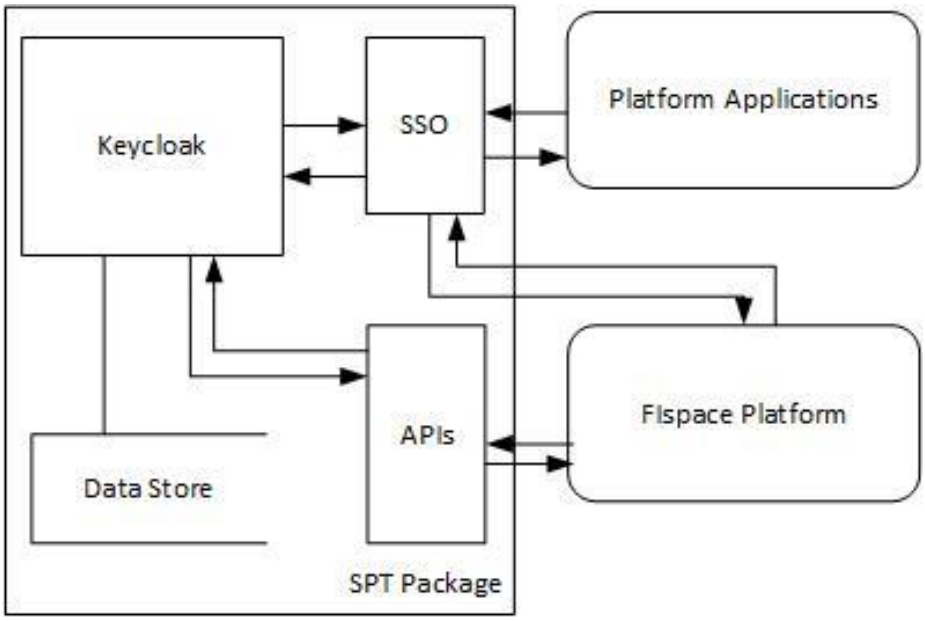

Figure 1: SPT high-level architecture

2.6 Integration Guide for Applications

User resources are secured by Keycloak. As explained in the high-level architecture; any application programming language supporting OAuth 2.0 protocols can be used for the authentication. Although FIspace Front-End is the only application using APIs at the moment; any development language supporting REST APIs can be adequate to develop an application using these provided APIs. However these management APIs are only to be used by the FIspace platform.

Information on OAuth implementation and related documents for Security, Privacy and Trust can be found at https://bitbucket.org/fispace/core/wiki/Home under SPT HOWTO, among other resources:

- OAuth Implementation using Keycloak:

  https://bitbucket.org/fispace/core/wiki/OAuth%20Implementation%20using%20Keycloak
- Widget Security & SSO:

  https://bitbucket.org/fispace/core/wiki/Widget%20Security%20&%20SSO
- JBoss documentation – “Pure Client Javascript Adapter”

  http://docs.jboss.org/keycloak/docs/1.0.4.Final/userguide/html/ch07.html#javascript-adapter
- JBoss Keycloak web site, documentation and tutorial:

  http://keycloak.jboss.org/

  http://keycloak.jboss.org/docs
- JBoss Keycloak tutorial – OAuth Clients:

  https://www.youtube.com/watch?v=DwSmDFTDP1I

3 Glossary

The glossary provides the coherent terminological framework used in this document.

3.1 Terms and definitions

This section provides definitions of any terms that may be needed in order for the reader to understand the terminology used in the document. The author should define any definition/acronym or technical term used in the document that may be unfamiliar to the reader, and it is best to err on the side of too many rather than too few definitions. This also allows the author to frame a word within a specific context, which provides the reader with a common understanding of the author's definition.

Access control

Authorisation (or delegation) for performing a certain action (based on privileges management). The access control is carried out once the Identification and Authentication procedures have been performed.

Accounting

Process of gathering information about the usage of resources by subjects.

Acceptance and trust

Acceptability indicates the degree of approval of a technology by the users. It depends on whether the technology can satisfy the needs and expectations of its users and potential stakeholders. Within the framework of introducing new technologies, acceptability relates to social and individual aspects as well.

Application

Use of capabilities, including hardware, software and data, provided by an information system specific to the satisfaction of a set of user requirements in a given application domain.

Application Domain

Integrated set of problems, terms, information and tasks of a specific thematic domain that an application (e.g. an information system or a set of information systems) has to cope with.

Application Schema [ISO/FDIS 19109:2003]

Conceptual schema for data required by one or more applications.

Architecture (of a system) [ISO/IEC 10746-2:1996]

Set of rules to define the structure of a system and the interrelationships between its parts.

Architecture (of a system) [ISO/IEC 10746-2:1996]

Set of rules to define the structure of a system and the interrelationships between its parts.

Authentication

Process of verifying the identity of a certain subject. In other words authentication indicates whether a subject is who/what it seems to be.

Generally speaking, this proof can depend on a secret that can be, e.g. what somebody has (key, smart card, …), what somebody knows (password, …), what somebody is (biometrical data, …)

Authorisation

Process of determining whether a subject is allowed to have the specified types of access to a particular resource. This is done by evaluating applicable access control information contained in a so called authorisation context. Usually, authorisation is carried out after the identification and authentication. Once a subject is identified and authenticated, it may be authorized (or not) to perform different types of access.

Availability

Availability refers to the degree to which a system, subsystem, or equipment is in a specified operable and committable state at the start of a mission, when the mission is called for at an unknown, i.e., a random time. So, availability is the proportion of time that a system is in operating condition.

Capability

Capabilities are a set of functionalities, through a combination of software and hardware, used to provide services and data. They can reside in a system or for example in a terminal itself as embedded capabilities or they can be available through the network services and infrastructure and others communication technologies as external capabilities.

Catalogue [derived from http://www.opengeospatial.org/resources/?page=glossary]

Collection of entries, each of which describes and points to a feature collection. Catalogues include indexed listings of feature collections, their contents, their coverages, and of meta-information. A catalogue registers the existence, location, and description of feature collections held by an Information Community. Catalogues provide the capability to add and delete entries. A minimum Catalogue will include the name for the feature collection and the locational handle that specifies where these data may be found. Each catalogue is unique to its Information Community.

Certificate Authority

A Trusted Third Party, responsible for ensuring the binding between the public keys and the personal data of their respective owners.

Component

Hardware component (device) or Software Component.

Conceptual model [ISO/FDIS 19109:2003(E); ISO 19101]

Model that defines concepts of a universe of discourse.

Conceptual schema [ISO/FDIS 19109:2003(E); ISO 19101]

Formal description of a conceptual model.

Coverage [ISO 19123]

Function from a spatial, temporal or spatiotemporal domain to an attribute range. A coverage associates a position within its domain to a record of values of defined data types. Thus, a coverage is a feature with multiple values for each attribute type, where each direct position within the geometric representation of the feature has a single value for each attribute type.

Data acquisition

Methods of data acquisition include methods to collect background data, digitally acquire data from sensors, and subjective data (such as data acquired from questionnaires). In addition, data in the form of manually or automatically transcribed data and reductions of collected data is also considered sensor acquired data (but with a manual sensor – the analyst).

Description Logics

Family of logic based knowledge representation languages that are a decidable subset of first order logic with well-defined semantics and inferencing (problem decision procedures). In Description Logics, a distinction is made between the terminological knowledge and the assertional knowledge. This distinction is useful for knowledge base modelling and engineering: for modelling it is just natural to distinguish between concepts and individuals; for engineering it helps by separating key inference problems.

Digital Certificate

A kind of digital document that contains structured information about the identity of its owner along with her/his public key, signed all together with a Certificate Authority's private key.

Digital Signature

The encrypted form of a message with the private key of the owner, indicating in a secure way the creator of the message, as well as the identity of a signed data.

Encryption

The act of modifying the contents of a message in an algorithmic and secure way, so that it can not be observed or altered in while in transit.

End-User

All users that are involved in an application domain and that use the applications, the services built by the system users according to the system and service Architecture.

Feature [derived from ISO 19101]

Abstraction of a real world phenomenon [ISO 19101] perceived in the context of an Application. In this general sense, a feature corresponds to an "object" in analysis and design models.

Framework [http://www.opengeospatial.org/resources/?page=glossary]

An information architecture that comprises, in terms of software design, a reusable software template, or skeleton, from which key enabling and supporting services can be selected, configured and integrated with application code.

Generic

A service is generic, if it is independent of the application domain. A service infrastructure is generic, if it is independent of the application domain and if it can adapt to different organisational structures at different sites, without programming (ideally).

Identification

The identification process allows relating a person/device with the service environment. The “electronic identity” is something like a credential or a “business card”, suitable to be verified throughout the authentication process.

Implementation [http://www.opengeospatial.org/resources/?page=glossary]

Software package that conforms to a standard or specification. A specific instance of a more generally defined system.

Info-structure Service

Service that is required to operate a system oriented service in the sense that it plays an indispensable role in the operation of an architecture or system oriented service.

Interface [ISO 19119:2005; http://www.opengis.org/docs/02-112.pdf]

Named set of operations that characterize the behaviour of an entity.

The aggregation of operations in an interface, and the definition of interface, shall be for the purpose of software reusability. The specification of an interface shall include a static portion that includes definition of the operations. The specification of an interface shall include a dynamic portion that includes any restrictions on the order of invoking the operations.

Interoperability [ISO 19119:2005 or OGC; http://www.opengeospatial.org/resources/?page=glossary]

Capability to communicate, execute programs, or transfer data among various functional units in a manner that require the user to have little or no knowledge of the unique characteristics of those units [ISO 2382-1]. (http://www.opengeospatial.org/ogc/glossary/i)

Loose coupling [W3C; http://www.w3.org/TR/2004/NOTE-ws-gloss-20040211/#loosecoupling]

Coupling is the dependency between interacting systems. This dependency can be decomposed into real dependency and artificial dependency: Real dependency is the set of features or services that a system consumes from other systems. The real dependency always exists and cannot be reduced. Artificial dependency is the set of factors that a system has to comply with in order to consume the features or services provided by other systems. Typical artificial dependency factors are language dependency, platform dependency, API dependency, etc. Artificial dependency always exists, but it or its cost can be reduced. Loose coupling describes the configuration in which artificial dependency has been reduced to the minimum.

Middleware [http://www.opengeospatial.org/resources/?page=glossary]

Software in a distributed computing environment that mediates between clients and servers.

Open Architecture [based on (Powell 1991)] [34]

Architecture whose specifications are published and made freely available to interested vendors and users with a view of widespread adoption of the architecture. An open ar-

chitecture makes use of existing standards where appropriate and possible and otherwise contributes to the evolution of relevant new standards.

Operation [ISO 19119:2005; http://www.opengis.org/docs/02-112.pdf]

Specification of a transformation or query that an object may be called to execute. An operation has a name and a list of parameters.

Performance indicators definition (PI)

PIs are quantitative or qualitative measurements, agreed on beforehand, expressed as a percentage, index, rate or other value, which is monitored at regular or irregular intervals and can be compared with one or more criteria.

Platform (Service)

Set of infrastructural means and rules that describe how to specify service interfaces and related information and how to invoke services in a distributed system.

Reference Model [ISO Archiving Standards; http://ssdoo.gsfc.nasa.gov/nost/isoas/us04/defn.html]

A reference model is a framework for understanding significant relationships among the entities of some environment, and for the development of consistent standards or specifications supporting that environment. A reference model is based on a small number of unifying concepts and may be used as a basis for education and explaining standards to a non-specialist.

Reliability

Reliability is the ability of a system or component to perform its required functions in routine circumstances, as well as hostile or unexpected circumstances, under stated conditions for a specified period of time.

Resource

Functions (possibly provided through services) or data objects.

Service [ISO 19119:2005; ISO/IEC TR 14252; http://www.opengis.org/docs/02-112.pdf]

Distinct part of the functionality that is provided by an entity through interfaces.

REST

Representational state transfer (REST) is an abstraction of the architecture of the World Wide Web; more precisely, REST is an architectural style consisting of a coordinated set of architectural constraints applied to components, connectors, and data elements, within a distributed hypermedia system. REST ignores the details of component implementation and protocol syntax in order to focus on the roles of components, the constraints upon their interaction with other components, and their interpretation of significant data elements.

Service [ISO 19119:2005; ISO/IEC TR 14252; http://www.opengis.org/docs/02-112.pdf]

Distinct part of the functionality that is provided by an entity through interfaces.

Session

Temporary association between a subject and a principal as a result of an authentication process initiated by the subject. Information about a session is stored in authentication session information.

SOAP

Simple Object Access protocol is a protocol specification for exchanging structured information in the implementation of web services in computer networks. It uses XML Information Set for its message format, and relies on other application layer protocols, most notably Hypertext Transfer Protocol (HTTP) or Simple Mail Transfer Protocol (SMTP), for message negotiation and transmission.

Software Component [derived from component definition of http://www.opengeospatial.org/resources/?page=glossary]

Software program unit that performs one or more functions and that communicates and interoperates with other components through common interfaces.

Source System

Container of unstructured, semi-structured or structured data and/or a provider of functions in terms of services. The source systems are of very heterogeneous nature and contain information in a variety of types and formats.

Support Service

Service that facilitates the operation of an architecture or system oriented service, e.g. providing an added value by combining the usage of Info-Structure Services.

System [ISO/IEC 10746-2:1996]

Something of interest as a whole or as comprised of parts. Therefore a system may be referred to as an entity. A component of a system may itself be a system, in which case it may be called a sub-system.

Note: For modelling purposes, the concept of system is understood in its general, system theoretic sense. The term "system" can refer to an information processing system but can also be applied more generally.

System User

Provider of services that are used for an application domain as well as IT architects, system developers, integrators and administrators that conceive, develop, deploy and run applications for an application domain.

Terminal

Terminals are a mobile device that is capable of running mobile services and/or mobile applications.

Use case

A common definition of use cases is the one described by Jacobson (Jacobson et al (1995) [35]): "*When a user uses the system, she or he will perform a behaviourally related sequence of transactions in a dialogue with the system. We call such a special sequence a use case*". In Other words, a use case is a textual presentation or a story about the usage of the system told from an end user's perspective.

The use cases provide some tools for people, with different skills (e.g. software developers and non-technology oriented people), to communicate with each other. The use

cases are general descriptions of needs or situations that often are related to basic scenarios and that are independent of the technologies and implementations of the underlying system.

User

Human acting in the role of a system user or end user of the service and system.

WADL

The Web Application Description Language is a machine-readable XML description of HTTP-based web applications (typically REST web services) WADL models the resources provided by a service and the relationships between them. WADL is intended to simplify the reuse of web services that are based on the existing HTTP architecture of the Web. It is platform and language independent and aims to promote reuse of applications beyond the basic use in a web browser.

Web Service

Self-contained, self-describing, modular service that can be published, located, and invoked across the Web. A Web service performs functions, which can be anything from simple requests to complicated business processes. Once a Web service is deployed, other applications (and other Web services) can discover and invoke the deployed service.

W3C Web Service [W3C, http://www.w3.org/TR/2004/NOTE-ws-gloss-20040211/#webservice]

Software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically WSDL). Other systems interact with the Web service in a manner prescribed by its description using SOAP-messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards.

4 References

The following references are used as background documents for the preparation of this document. References are categorized standards (i.e. standards and specifications from the consortium working groups or alliances and specifications or drafts standardization bodies) and other documents, publications and technical or scientific books.

[1]	FIspace project. FIspace: Future Internet Business Collaboration Networks in Agri-Food, Transport and Logistics. Deliverable D200.1 *"FIspace Design and Release Plan"*, 2014.
[2]	FIspace project. FIspace: Future Internet Business Collaboration Networks in Agri-Food, Transport and Logistics. Deliverable D200.2 *"FIspace Technical Architecture and Specification"*, 2014
[3]	FIspace project. FIspace: Future Internet Business Collaboration Networks in Agri-Food, Transport and Logistics. Deliverable D200.3 *"FIspace Integrated Release V1"*, 2014
[4]	FIspace project. FIspace: Future Internet Business Collaboration Networks in Agri-Food, Transport and Logistics. Deliverable D200.4 *"FIspace Development Progress Report and V1 Updates"*, *2014*
[5]	FIspace project. FIspace: Future Internet Business Collaboration Networks in Agri-Food, Transport and Logistics. Deliverable D200.5 *"FIspace Integrated Release V2"*, 2014.
[6]	FIspace project. FIspace: Future Internet Business Collaboration Networks in Agri-Food, Transport and Logistics. Deliverable D200.6 *"FIspace Development Progress Report and V2 Updates"*, *2014*
[7]	FIspace project. FIspace: Future Internet Business Collaboration Networks in Agri-Food, Transport and Logistics. D200.5 Annex *"FIspace Front-End User Guide"*
[8]	FIspace project. FIspace: Future Internet Business Collaboration Networks in Agri-Food, Transport and Logistics. D200.5 Annex *"FIspace SDK User and Developer Guide"*
[9]	FIspace project. FIspace: Future Internet Business Collaboration Networks in Agri-Food, Transport and Logistics. Deliverable D200.7 *"FIspace Integrated Release V3"*, 2014.
[10]	FIspace project. FIspace: Future Internet Business Collaboration Networks in Agri-Food, Transport and Logistics. D200.7 Annex *"FIspace Front-End User Guide"*
[11]	FIspace project. FIspace: Future Internet Business Collaboration Networks in Agri-Food, Transport and Logistics. D200.7 Annex *"FIspace SDK User and Developer Guide"*

[12]	FIspace Business collaboration web site. http://www.fispace.eu/
[13]	FIspace Developer Documentation web site. http://dev.fispace.eu/doc/wiki/Home
[14]	FIspace Deliverables web site. http://www.fispace.eu/deliverable.html
[15]	FIspace Tutorial web site. http://www.fispace.eu/tutorials.html
[16]	FIspace Front-End Users Information web site. http://dev.fispace.eu/doc/wiki/gui
[17]	FIspace Front-End User Guide web site. http://dev.fispace.eu/doc/wiki/gui/gui-guide
[18]	FIspace App Developer Intro web site. http://dev.fispace.eu/doc/wiki/App%20Developer%20Intro
[19]	FIspace SDK Guide web site. http://dev.fispace.eu/doc/wiki/sdk
[20]	FIspace apps for newbies web site. https://bitbucket.org/fispace/apps/wiki/FIspace%20apps%20for%20newbies
[21]	FIWARE web site. http://www.fi-ppp.eu/projects/fi-ware/
[22]	FIWARE Catalogue of the Generic Enablers (GEs). http://catalogue.fi-ware.org/
[23]	FIWARE community web site. http://www.fi-ware.org/community/
[24]	FIWARE - Catalogue - Application Mashup – Wirecloud web site. http://catalogue.fi-ware.org/enablers/application-mashup-wirecloud
[25]	FIWARE - Catalogue - Store – Wstore web site. http://catalogue.fi-ware.org/enablers/store-wstore
[26]	Eclipse web site. https://www.eclipse.org/ , https://www.eclipse.org/downloads/
[27]	Maven web site. http://maven.apache.org/ , http://maven.apache.org/download.cgi
[28]	RabbitMQ web site. http://www.rabbitmq.com/
[29]	Express web site. http://expressjs.com/
[30]	ACSI EU-project web site. http://www.acsi-project.eu/
[31]	Proton - CEP recorded webinar and tutorial. http://edu.fi-ware.eu/course/view.php?id=58 Proton - CEP user and programmers guide. http://forge.fi-ware.org/plugins/mediawiki/wiki/fiware/index.php/CEP_-

	_User_and_Programmer_Guide
[32]	Keycloak JBoss project. http://keycloak.jboss.org/ Keycloak JBoss documentation. http://keycloak.jboss.org/docs
[33]	OAuth 2.0. Open standard Authentication protocol specification. http://tools.ietf.org/html/rfc6749
[34]	Powell, D. (Ed.) (1991). Delta-4: A Generic Architecture for Dependable Distributed Computing. Re-search Reports ESPRIT. Project 818/2252 Delta-4 Vol.1. ISBN 3-540-54985-4 Springer-Verlag 1991.
[35]	Jacobson, I., Bylund, S., Jonsson, P., and Ehneboom, S. (1995), “Modeling with Use Cases: Using contracts and use cases to build plugable architectures”. Journal of Object Oriented Programming, Vol. 8, No. 2, pp. 18-24.


FIspace
Business Collaboration