The Food Safety Market: An SME-powered industrial data platform to boost the competitiveness of European food certification

D4.2 – Annual Report from Software Integration & Testing

DELIVERABLE NUMBER	D4.2
DELIVERABLE TITLE	Annual Report from Software Integration & Testing
RESPONSIBLE AUTHOR	Danai Vergeti (UBITECH)

Co-funded by the Horizon 2020
Framework Programme of the European Union

GRANT AGREEMENT N.	871703
PROJECT ACRONYM	TheFSM
PROJECT FULL NAME	The Food Safety Market: An SME-powered industrial data platform to boost the competitiveness of European food certification
STARTING DATE (DUR.)	01/01/2020 (36 months)
ENDING DATE	31/12/2023
PROJECT WEBSITE	www.foodsafetymarket.eu
COORDINATOR	Nikos Manouselis
ADDRESS	110 Pentelis Str., Marousi, GR15126, Greece
REPLY TO	nikosm@agroknow.com
PHONE	+30 210 6897 905
EU PROJECT OFFICER	Stefano Bertolo
WORKPACKAGE N. | TITLE	WP3 | Platform
WORKPACKAGE LEADER	UBITECH
DELIVERABLE N. | TITLE	D3.3 | Annual Report from TheFSM Software Components Integration and Testing
RESPONSIBLE AUTHOR	Danai Vergeti (UBITECH)
REPLY TO	vergetid@ubitech.eu
DOCUMENT URL	
DATE OF DELIVERY (CONTRACTUAL)	30 April 2021 (M15)
DATE OF DELIVERY (SUBMITTED)	31 May 2021 (M16)
VERSION | STATUS	1.0 | Final
NATURE	Report (R)
DISSEMINATION LEVEL	Public (P)
AUTHORS (PARTNER)	Danai Vergeti (UBITECH), Dimitris Ntalaperas (UBITECH), Iosif Angelidis (UBITECH)
CONTRIBUTORS	Giannis Stoitsis (Agroknow), Nikos Manouselis (Agroknow), Maja Majdak (Agrivi)
REVIEWER	Branimir Rakic (PROSPEH)

VERSION	MODIFICATION(S)	DATE	AUTHOR(S)
0.1	Table of contents	1 April 2021	Danai Vergeti (UBITECH)
0.3	Contribution to section 2	15 April 2021	Danai Vergeti (UBITECH)
0.5	Contribution to section 2	8 May 2021	Danai Vergeti (UBITECH)
0.6	Contribution to section 2	10 May 2021	Giannis Stoitsis (Agroknow), Nikos Manouselis (Agroknow), Maja Majdak (AGRIVI)
0.8	Contribution to sections 3 and 4	17 May 2021	Giannis Stoitsis (Agroknow), Nikos Manouselis (Agroknow), Maja Majdak (AGRIVI), Iosif Angelidis (UBITECH)
0.9	Contribution to sections 1 and 5 and version for review	20 May 2021	Danai Vergeti (UBITECH), Iosif Angelidis (UBITECH), Dimitris Ntalaperas (UBITECH)
0.95	Deliverable review	22 May 2021	Branimir Rakic (PROSPEH)
1.0	Final version	28 May 2021	Danai Vergeti (UBITECH)

PARTNERS		CONTACT
Agroknow IKE (Agroknow, Greece)	Agroknow	Nikos Manouselis (Agroknow) nikosm@agroknow.com
SIRMA AI EAD (SAI, Bulgaria)	ontotext	Svetla Boytcheva (SAI) svetla.boytcheva@ontotext.com
GIOUMPITEK MELETI SCHEDIASMOS YLOPOIISI KAI POLISI ERGON PLIROFORIKIS ETAIREIA PERIORISMENIS EFTHYNIS (UBITECH, Greece)	UBITECH ubiquitous solutions	Danai Vergeti (UBITECH) vergetid@ubitech.eu
AGRIVI DOO ZA PROIZVODNJU, TRGOVINU I USLUGE (Agrivi d.o.o., Croatia)	AGRIVI	Tanja Matosevic (Agrivi d.o.o.) tanja.matosevic@agrivi.com
PROSPEH, POSLOVNE STORITVE IN DIGITALNE RESITVE DOO (PROSPEH DOO, Slovenia)	tracelabs originTrail Core Developers	Ana Bevc (PROSPEH DOO) ana@origin-trail.com
UNIVERSITAT WIEN (UNIVIE, Austria)	universität wien	Tima Anwana (UNIVIE) tima.anwana@univie.ac.at
STICHTING WAGENINGEN RESEARCH (WFSR, Netherlands)	WAGENINGEN UNIVERSITY & RESEARCH	Yamine Bouzembrak (WFSR) yamine.bouzembrak@wur.nl
TUV- AUSTRIA ELLAS MONOPROSOPI ETAIREIA PERIORISMENIS EUTHYNIS (TUV AU HELLAS, Greece)	TÜV AUSTRIA HELLAS	Kostas Mavropoulos (TUV AU HELLAS) konstantinos.mavropoulos@tuv.at
TUV AUSTRIA ROMANIA SRL (TUV AU ROMANIA, Romania)	TÜV AUSTRIA ROMANIA	George Gheorghiou (TUV AU Romania) george.gheorghiu@tuv.at
VALORITALIA SOCIETA PER LA CERTIFICAZIONE DELLE QUALITA'E DELLE PRODUZIONI VITIVINICOLE ITALIANE SRL (VALORITALIA, Italy)	VALORITALIA	Francesca Romero (Valoritalia) francesca.romero@valoritalia.it
TUV AUSTRIA CYPRUS (TUV AU CYPRUS, Cyprus)	TÜV AUSTRIA CYPRUS	Sousanna Charalambidou (TUV AU CYPRUS) sousanna.charalambidou@tuv.at

ACRONYMS LIST

TheFSM	The Food Safety Market
ABAC	Attribute Based Access Control
API	Application Programming Interface
CI/CD	Continuous Integration and Continuous Delivery
REST	Representational state transfer

EXECUTIVE SUMMARY

In the current document we present the integration strategy of TheFSM Platform with the external applications Agrivi 2.0, Food Inspector and FOODAKAI 2.0 which follows the development, testing and deployment strategy for TheFSM technical solution that was presented in D3.3, M12. We focus the on the application testing and integration with TheFSM Platform. The APIs of the applications are presented along with examples of unit tests which test the internal functionality of the applications and their compliance with the specifications. Additionally, integration tests are provisioned in order to assure the interoperability between the platform and external 3rd party applications.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

1.1 Scope

The scope of the current deliverable is to document the actions taken under "Task 4.2 Technical Verification & Integration Testing" and provides an annual report describing the process and outcomes of the integration and testing of TheFSM Applications with TheFSM Platform. The main objectives of D4.2 are: a) to present the release plan of the applications b) to provide the description of the APIs developed for the first version of the applications and the core integration points with TheFSM Platform and c) the testing process and results as well as, the technical evaluation of TheFSM applications.

1.2 Audience

D4.2 targets the consortium members of the TheFSM project and especially the technical partners which participate in WP2, WP3, WP4 with the scope to provide the integration strategy, and release plan of the TheFSM Applications, as well as, the description of the APIs developed for the first version of the applications and the core integration points, and the testing results. Additionally, an audience outside the consortium, with technical background (s/w engineers, developers, architects etc.) can also follow and understand the methodology, the APIs description and integration points and testing results as it is presented in the current document.

1.3 Structure

This deliverable is structured as follows.

- Section 2 documents the approach we follow in order to perform the integration and the testing of the applications including the integration plan and the release plan.
- Section 3 presents the developed APIs of the applications.
- Section 4 is dedicated to the technical verification of the applications. We define how verification is guaranteed via unit testing, thoroughly describing the unit testing process.
- Section 5 concludes this report.

1.4 Relation to other deliverables

The current document is strongly related to D3.1 where the functionalities and the components of the FSM platform are described at a conceptual level. The outcomes of T3.2-T3.5 (until M15) and as they are documented in "D3.3 - Annual Report from TheFSM software components Integration & Testing", as well as, "D4.1 Annual Report from Iterative Application Development" (delivered in M12) provided input for the integration strategy, technical integration and technical tasks.

2. THEFSM APPLICATIONS INTEGRATION

2.1 Methodology

2.1.1 Integration strategy

The integration of TheFSM Applications is a living continuous activity, following the iterative agile process of the applications implementation (D4.1) and the integration strategy of TheFSM Platform (D3.3). We note that, the integration strategy of the applications is part of the holistic integration strategy of TheFSM Platform that was thoroughly presented in D3.3. Thus, the development process, as well as, the integration tools of the applications are the same as they are presented in D3.3 (section 2) and D4.1.

The first iteration of the integration and testing of the applications includes the integration of the first version of the FOODAKAI 2.0 and Agrivi 2.0, as well as, the API Gateway which is the main component provided by TheFSM Platform in order to implement secure and trusted integration with the external applications for data and services sharing through TheFSM Platform. The integration with the Food Inspector 2.0, as well as, the integration with other services of TheFSM Platform, such as, alerting and monitoring will be provided in the next iterations, based on the integration plan and the release plan that are presented in the following sections.

2.1.2 Integration Plan

The first version of TheFSM Applications (v1.0) delivers the prototype version of the applications mentioned in Table 2: TheFSM Applications release plan. The on-time delivery and smooth integration of the features of the applications was driven by the integration plan provided in D3.3, M12 identifying the prioritization and relevant delivery dates of each component, sub-component and application feature. The following table presents the integration plan that guided the integration of the applications with TheFSM Platform and it is part of the main integration plan of TheFSM Platform. In the following table we focus on the related tasks, features, components that are related with the applications integration. These components include a) the applications of TheFSM solution, b) the API Gateway, which is a component provided by TheFSM Platform which enables the secure integration of external applications in order to share data and services, c) the alerting and monitoring components.

Table 1: Integration Plan until M15

Relevant task in DoA	Component	Subcomponent/Action	Output	version	Planned date of delivery	Contractual date of delivery
T4.2.2	FOODAKAI 2.0	FSI Data Platform extensions	API	v1.0	1/2/2021	M12
		UI extensions	UI	v1.0	26/2/2021	M12
		Implementation of the	UI	v1.0	24/9/202	M24

		i18n support and multilingual UI			1	
		Components to integrate FOODAKAI application with TheFSM Platform for exchanging data	UI, API	v1.0	10/9/2021	M24
T4.2.3	Agrivi 2.0	Auditor/certifier/consultant interface	UI	v1.0	31/1/2020	M12
T4.2.1	Food Inspector	Backend api	API	v1.0	10/9/2021	M12
		UI	UI	v1.0	10/9/2021	M12
		Components to integrate Food Inspector application with TheFSM Platform for exchanging data	UI, API	v1.0	4/10/2021	
T4.3.1	API Gateway	Backend API	API	v1.0	M13	M15

2.1.3 Release Plan for TheFSM Applications

The release plan of TheFSM applications provides the delivery status of each application feature for each platform release. Moreover, the components of TheFSM Platform that are related with the integration of the applications are also provided for the sake of completeness. For each component we define the following delivery status: a) Prototype: the prototype of the artefact, including the core functionalities, addressing the core functionalities of the platform, as well, b) Beta version: extended/updated version of the prototype with additional functionalities addressing the core and added-value functionalities of the platform c) Final version: final version of the components, updating and fine tuning the added-value services of the platform.

The release plan of TheFSM Applications is provided in the following table:

Table 2: TheFSM Applications release plan.

Component	v1.0 (M15)	v2.0 (M24)	Final version (M36)
FOODAKAI 2.0	Prototype	Beta version	Final version
AGRIVI 2.0	Prototype	Beta version	Final version
Food Inspector integration	-	Beta version	Final version
Monitoring	-	Prototype	Final version
Alerting	-	Prototype	Final version
API Gateway	Prototype	Beta version	Final version

Additionally, we provide the overall goals/required features per application and component for each milestone.

Table 3: Features per application/component per milestone.

Component	First version (M15)	Second version (M24)	Final version (M36)
FOODAKAI 2.0	• Supplier check service • Suppliers alerts • Supplier risk profile page • Mechanism to Import suppliers • Suppliers risk assessment dashboard • Import data for a parameter of the risk assessment matrix • Mechanism to invite suppliers to create a profile • Risk score for each supplier based on recalls, global incidents and country situation • Mechanism to submit my data using TheFSM platform • Mechanism to retrieve supplier data using TheFSM platform •	• Dashboard highlighting which suppliers are at high risk • Share/exchange Certificate information mechanism • Share/exchange laboratory testing results mechanism • Share/exchange audit reports mechanism • Support of i18N for multilinguality • Multilingual UI/UX and support of new languages •	• Share products/ingredients traceability information • Risk estimation service for the suppliers • Risk prediction services for the supplier
AGRIVI 2.0	• Granulated user permission system • User	· Italian language · Auditor/certifier interface • API extensions	· Other prioritized languages implementation • API extensions

	management interface • Agrivi FMS farm platform with all functionalities		
Food Inspector integration	• Wireframes of the required services	· Check the background of a company to be inspected/certified · Company's risk profile dashboard · Mechanism to share/exchange inspection reports · Submit inspection reports for processing and mining · Set up an initial version of the application based on FOODAKAI 2.0 architecture and functionalities · Develop the mechanism to securely upload the data from company to auditor using TheFSM platform · Develop the mechanism to securely retrieve the data from company to auditor using TheFSM platform • · Alpha version of the Inspector Dashboard with risk monitoring capabilities	• Multilingual UI/UX and support of new languages • Risk estimation service for the companies to be certified/inspected • Risk prediction services for the companies to be certified/inspected • Companies' risk estimation and prediction dashboard
FOODAKAI 2.0 integration	• Integration with the platform through the API Gateway	• Initial integration for alerting and monitoring • Test use cases with interactions between FOODAKAI 2.0 and the platform	• Finalize and refine integration

AGRIVI 2.0 integration	• Integration with the platform through the API Gateway	• Initial integration for alerting and monitoring • Test use cases with interactions between FOODAKAI 2.0 and the platform	• Finalize and refine integration
Food Inspector integration	• Integration with the platform through the API Gateway	• Initial integration for alerting and monitoring • Test use cases with interactions between FOODAKAI 2.0 and the platform	• Finalize and refine integration
API Gateway	• Initial implementation of the API Gateway Service (direct-to-service gateway) • Organization of endpoints per service • Integration with A2C engine	• Implementation of indirect call to service, with intermediate credentials and/or authentication	• Refinement of organizing endpoints per service and addition of more relevant functionalities where applicable based on pilots input and external applications
Monitoring	-	-	• Add monitoring capabilities to the platform
Alerting	-	-	• Add alerting functionalities involving critical actions throughout interactivity with the platform

3. THEFSM APPLICATIONS INTERFACES

In this section, we present with more details the information gathered about the interfaces required for the implementation of the integrated platform by defining the communication between the components.

3.1 Data Exchange Scenarios

The first version of the integration of TheFSM Applications with TheFSM Platform supports two main data exchange scenarios which were selected based on two main criteria: a) the inclusion of the core technical integration points between the external applications and the platform, and b) the added value for the external applications. The integration of TheFSM Applications with TheFSM Platform is a loosely coupled integration process taking place through the API Gateway. More specifically, each application owner as data provider needs to do the following: a) register the API that is planned to be shared through TheFSM Platform and b) define the access policies that are applied on this API, c) follow the steps for authentication/authorization the provided API might require (via API key or JWT provided by the API Gateway side). Additionally, as a data consumer, the following steps are provisioned: a) search the API that he/she is interested to consume, b) select the API that he/she is interested in, c) follow the steps for authentication/authorization. As soon as these steps are performed, TheFSM Platform acts as a secure proxy between the parties and performs authentication, authorization and secure data sharing. The data exchange scenarios are presented below:

a. An external application provides data as a data provider through an API. The data provided by the API have access restrictions. The same application needs to share these data with a restricted list of users through TheFSM Platform.

b. An external application wants to consume data provided through an API from another external application. The shared data are under access restrictions from the data provider.

The next sections document the API interfaces involved in the aforementioned data exchange scenarios, as well as, the relevant integration points as they are supported by the API Gateway. For each API interface we provide the following information:

The following subsections describe these interfaces by detailing the following information:

- **Description:** describes the purpose of the interface.
- **Component providing the interface:** describes the component that is offering the described interface.
- **Consumer components:** describes the components that are using the described interface.
- **Type of interface:** REST, XML-RPC, GUI, Java API etc.
- **Input data:** describes data that is required by the described interface (e.g., Methods or Endpoints, values and parameters of the interface)
- **Output data:** describes the data that is returned by the described interface (e.g., the returned data of methods or REST call)

- **Constraints:** Any other constraints (e.g., specific prerequisites, data-types, encoding, transfer rates) which apply to the interface.
- **Responsibilities:** Partner that is responsible for the implementation and usage of the interface

3.2 API Gateway

Table 4: API Gateway

Name: API Gateway			
Description	Interface of the API Gateway Service		
Component providing the interface	API Gateway		
Consumer components or External Entities	All integrated services into the platform and/or services wishing to have out of the box support of ABAC protection		
Type of Interface	REST		
Input data / Output Data	**Methods or endpoints** of the interface	**Parameters** of the method	**Return Values** of the method
	GET /api/v1/gateway	None	List<ServiceDTO>
	POST /api/v1/gateway	<ServiceDTO>	<ServiceDTO>
	GET /api/v1/gateway/{id}	{id}	<ExceptionMessageDTO>, <ServiceDTO>
	PUT /api/v1/gateway/{id}	{id}, <ServiceDTO>	<ExceptionMessageDTO>, empty body
	DELETE /api/v1/gateway/{id}	{id}	<ExceptionMessageDTO>, empty body
	POST /api/v1/gateway/{id}/endpoint	{id}, <EndpointDTO>	<ExceptionMessageDTO>, empty body
	GET /api/v1/gateway/{serviceId}/endpoint	{serviceId}	<ExceptionMessageDTO>, List<EndpointDTO>
	GET	{serviceId},	<ExceptionMessage

	/api/v1/gateway/{serviceId}/endpoint/{endpointId}	{endpointId}, <EndpointDTO>	DTO>, <EndpointDTO>
	PUT /api/v1/gateway/{serviceId}/endpoint/{endpointId}	{serviceId}, {endpointId}	<ExceptionMessageDTO>, empty body
	DELETE /api/v1/gateway/{serviceId}/endpoint/{endpointId}	{serviceId}, {endpointId}	<ExceptionMessageDTO>, empty body
	PUT /api/v1/gateway/{serviceId}/endpoint/{endpointId}/disable	{serviceId}, {endpointId}	<ExceptionMessageDTO>, empty body
	PUT /api/v1/gateway/{serviceId}/endpoint/{endpointId}/enable	{serviceId}, {endpointId}	<ExceptionMessageDTO>, empty body
	POST /api/v1/gateway/{serviceId}/endpoint/exists	{serviceId}, <endpoint>	<ExceptionMessageDTO>, <EntityExistsDTO>
	POST /api/v1/gateway/call	<RequestDTO>	<ExceptionMessageDTO>, JSON response from called endpoint
	GET /api/v1/gateway/endpoints	None	<ExceptionMessageDTO>, List<EndpointDTO>
	POST /api/v1/gateway/exists	<service>	<ExceptionMessageDTO>, <EntityExistsDTO>
	GET /api/v1/gateway/services	None	<ExceptionMessageDTO>, List<ServiceDTO>
Constraints	N/A		
Responsibilities	UBITECH		

3.3 FOODAKAI 2.0 API

Table 5: FOODAKAI 2.0 API

Name:	
Description	REST API that provides access to all the incidents and supplier data that are used by the FOODAKAI 2.0 application for risk monitoring, risk assessment

	and risk prediction. Documentation available at http://docs.agroknow.com/.		
Component providing the interface	Food Safety Incidents (FSI) data platform of Agroknow		
Consumer components or External Entities	Access to food safety incidents and suppliers data through API		
Type of Interface	REST (JSON)		
Input data / Output Data	**Methods or endpoints** of the interface	**Parameters** of the method	**Return Values** of the method

	https://api.foodakai.com/search-api-1.0/search/	apikey
		detail
		entityType
		existenceQuery
		freetext
		smart
		expand
		highlight
		from
		to
		numericalQueries
		strictQuery
		wildcardQuery
		sourceInclude
		page
		pageSize
			{
			"aggregations": {
			"aggregationName": {
			"attribute": "string",
			"format": "string",
			"interval": "string",
			"size": 0,
			"subAggregation": {
			"attribute": "string",
			"format": "string",
			"interval": "string",
			"size": 0
			}
			}
			},
			"apikey": "string",
			"detail": true,
			"entityType": "string",
			"existenceQuery": [
			"string"
			],
			"expand": true,
			"freetext": "string",
			"from": "yyyy-MM-dd",

		sortOn	"highlight": true,
			"numericalQueries": [
			{
			"attribute": "string",
			"operator": "string",
			"value": 0
			}
			],
			"page": 0,
			"pageSize": 0,
			"smart": true,
			"sortOn": {
			"additionalProp1": "string"
			},
			"sourceInclude": [
			"string"
			],
			"strictQuery": {
			"additionalProp1": "string",
			"additionalProp2": "string",
			"additionalProp3": "string"
			},
			"to": "yyyy-MM-dd",
			"wildcardQuery": {
			"additionalProp1": "string",
			"additionalProp2": "string",
			"additionalProp3": "string"
			}
			}

Constraints	Authentication through API Key		
Responsibilities	Agroknow is responsible for further development, testing, operation and maintenance of the API		

3.4 AGRIVI 2.0 API

Table 6: AGRIVI 2.0 API

Name: AGRIVI API			
Description	AGRIVI 2.0 API		
Component providing the interface	API		
Consumer components or External Entities	Access to farm data through API		
Type of Interface	REST (JSON)		
Input data / Output Data	**Methods or endpoints** of the interface	**Parameters** of the method	**Return Values** of the method
	GET /Companies/{id}	ID	DTO
	POST /Companies/{id}	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /Fields	Company ID	DTO
	POST Fields	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /Fields/{id}	ID	DTO
	POST /Fields/{id}	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR

	GET /Items	Company ID	DTO
	POST /Items	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /Items/{id}	ID	DTO
	POST /Items/{id}	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /ItemCategories	Company ID	DTO
	POST /ItemCategories	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /ItemCategories/{id}	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	POST /ItemCategories/{id}	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /People	Company ID	DTO
	POST /People{id}	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR

	GET /Cultures	Company ID	DTO
	POST /Cultures	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /Countries	Company ID	DTO
	POST /Countries	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /Pests	Company ID	DTO
	POST /Pest	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /pests/{id}	ID	DTO
	POST /pests/{id}	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /Plantations	Company ID	DTO
	POST /Plantations	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /Plantations/{id}	ID	DTO

	POST /Plantations/{id}	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /ProductionTypes	Company ID	DTO
	POST /ProductionTypes	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /ProductonTypes/{id}	ID	DTO
	POST /ProductonTypes/{id}	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /Seasons	Company ID	DTO
	POST /Seasons	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /Seasons/{id}	ID	DTO
	POST /Seasons/{id}	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /LinkedFields	Company ID	DTO
	POST /LinkedFields	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER

			ERROR
	GET /LinkedFields({id}	ID	DTO
	POST /LinkedFields({id}	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /Sorts	Company ID	DTO
	POST /Sorts	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /Sorts/{id}	ID	DTO
	POST /Sorts/{id}	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /TaskCategories	Company ID	DTO
	POST /TaskCategories	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /TaskCategories/{id}	ID	DTO
	POST /TaskCategories/{id}	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR

	GET /Units	Company ID	DTO
	POST /Units	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /Units/{id}	ID	DTO
	POST /Units/{id}	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /ChemicalAnalysisExtent	Company ID	DTO
	POST /ChemicalAnalysisExtent	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /SoilAnalysisDepth	Company ID	DTO
	POST /SoilAnalysisDepth	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /TaskChemicalAnalysis	Company ID	DTO
	POST /TaskChemicalAnalysis	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /TaskChemicalAnalysis/{id}	ID	DTO

	POST /TaskChemicalAnalysis/{id}	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /TaskPlanting	Company ID	DTO
	POST /TaskPlanting	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /TaskPlanting/{id}	ID	DTO
	POST /TaskPlanting/{id}	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /TaskFertilizers	Company ID	DTO
	POST /TaskFertilizers	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /TaskFertilizers/{id}	ID	DTO
	POST /TaskFertilizers/{id}	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /TaskPesticides	Company ID	DTO
	POST /TaskPesticides	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER

			ERROR
	GET /TaskPesticides/{id}	ID	DTO
	POST /TaskPesticides/{id}	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /TaskIrrigation	Company ID	DTO
	POST /TaskIrrigation	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /TaskIrrigation/{id}	ID	DTO
	POST /TaskIrrigation/{id}	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /TaskWorkers	Company ID	DTO
	POST /TaskWorkers	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR
	GET /TaskWorkers/{id}	ID	DTO
	POST /TaskWorkers/{id}	ID, DTO	OK, BAD REQUEST, FORBIDDEN, UNAUTHORIZED, SERVER ERROR

Constraints	N/a
Responsibilities	

All additional documentation can be found here:

https://mobile-2-5.agrivi.com/swagger/index.html

https://swagger.io/specification/

4. TECHNICAL VERIFICATION

This section presents the application testing as part of the overall evaluation strategy in the context of the FSM, that was introduced in D3.3, section 3. Following the general testing strategy of TheFSM, the applications technical testing and evaluation will be based on STEP (Systematic Test and Evaluation Process), a well-established industry methodology for testing and evaluation activities in information technology and software projects. The testing will be performed to verify the proper functioning and performance of the integrated FSM platform. More details on the method can be found in D3.3, section 3.

In summary, in TheFSM, we define the following facets of testing:

- Unit testing that can be performed by the separate development teams when new functionalities are developed.
- Integration testing performed by the development teams in order to test the smooth co-operation between the various layers and components. The integration tests and also any unit tests that will be created for the project validation will be continuously executed based on continuous integration (CI) scheme
- Testing of a set of advanced scenarios based on demonstrators' needs.

These testing facets are presented in the following sections for the applications and the related platform components (API Gateway).

4.1 Unit testing

Unit tests are the tool to test the functional modules of software. In the case of the FSM, development is based on the development of standalone components but also on the adaptation and integration of existing components. The following tables provide indicative unit tests for the API Gateway.

Table 7: Unit Test documentation for API Gateway

Unit Test Case Documentation Form	
Unit Test Reference Code	#UT1
Component	API Gateway
Tester	Junit
Short Description	
Test if an endpoint can be added to a service. The test assumes the service already exists.	
Input Data (Configuration and Policies)	
EndpointDTO, serviceId	

Output Data (test requests)
Status of operation

Unit Test Case Documentation Form	
Unit Test Reference Code	#UT2
Component	API Gateway
Tester	Junit
Short Description	
Test if an endpoint can be disabled. The test assumes the service the endpoint belongs to already exists. The same holds for the endpoint itself.	
Input Data (Configuration and Policies)	
endpointId, serviceId	
Output Data (test requests)	
Status of operation	

Unit Test Case Documentation Form	
Unit Test Reference Code	#UT3
Component	API Gateway
Tester	Junit
Short Description	
Test if a service can be deleted. The test assumes the service already exists and consists of multiple endpoints. Success of the operation occurs when the service and all endpoints attached to it are successfully deleted.	
Input Data (Configuration and Policies)	
serviceId	
Output Data (test requests)	
Status of operation	

Unit Test Case Documentation Form	
Unit Test Reference Code	#UT4

Component	API Gateway
Tester	Junit
Short Description	
Test an API call via the API gateway and obtain the result. We assume the endpoint and the service of the endpoint already exist, as well as the endpoint being enabled.	
Input Data (Configuration and Policies)	
requestDTO	
Output Data (test requests)	
Status of operation in case of failure, or the actual return value of the API called	

Apart from the tests that guarantee the functional correctness of the components, it is important to make tests at the integration level for a complete testing and validation process. This means that integration tests shall be created and used for all identified interfaces and to some major platform functionalities. This can be done using unit testing on the methods that are implementing the integration, in order to make them part of continuous integration and continuous testing process.

4.2 Integration testing

Integration testing is the phase in software testing in which individual software modules are combined and tested as a group. Integration testing in FSM can also be seen as an extension of unit testing. The main idea of integration testing is to start from two components to test the interface between them. More details on the integration testing methods were introduced in D3.3, section 3.

4.2.1 Application integration points and complex flows testing

As it is important for FSM to ensure the proper integration of the applications, tests that are based on functions that cover different integration points will be used.

Table 8: Identified and Planned Integration Tests

Test ID	Test	Interface(s) Tested	Components Used	Short Description
IT1	End to end encryption when data transferring basic test	Security Layer end to end encryption	Security Layer, all other components requiring data transfer from/to the end user	Ensure hybrid encryption works correctly and data are encrypted between transfers.
IT2	ABAC policy editing	ABAC dashboard	ABAC dashboard (UI,	Testing policy CRUD operations work as

		and Security Layer Backend	frontend), Security Layer backend	intended.
IT3	ABAC enforcement test	ABAC dashboard and Security Layer backend	ABAC dashboard (UI, frontend), Security Layer backend, Authentication component	Testing granted status on requests by filtering through authentication and authorization
IT4	Dataset upload	Marketplace interaction with user in order to upload dataset with simple steps	Security Layer, Data Curation Layer	Testing that users with correct attributes can upload datasets. Test each individual step of the dataset upload process (column filtering, sampling, curation, column semantic mapping etc.) and ensure the dataset arrives to the platform encrypted.
IT5	Introduce third-party interface to the platform via the API Gateway and expose it	API Gateway, third-party interface	API Gateway, Security Layer backend	Test if a third-party interface can be successfully introduced into the platform and exposed via the API Gateway.
IT6	API Gateway third-party call and endpoint policy protection	API Gateway, third-party interface	API Gateway, Security Layer backend, Data Encryption Service	• Test if API Gateway can successfully call a third-party interface and return its result to the caller. Test if API Gateway prevents execution of this-party interface call when caller provides malformed data or they are not authorized (integration point with Security Layer backend and ABAC policies).

Next, for the sake of completeness, we provide a few sequence diagrams of the previous integration points.

Expose third-party interface via the API Gateway

This sequence diagram illustrates the steps taken when a user registers and exposes a new third party interface endpoint as part of a new service.

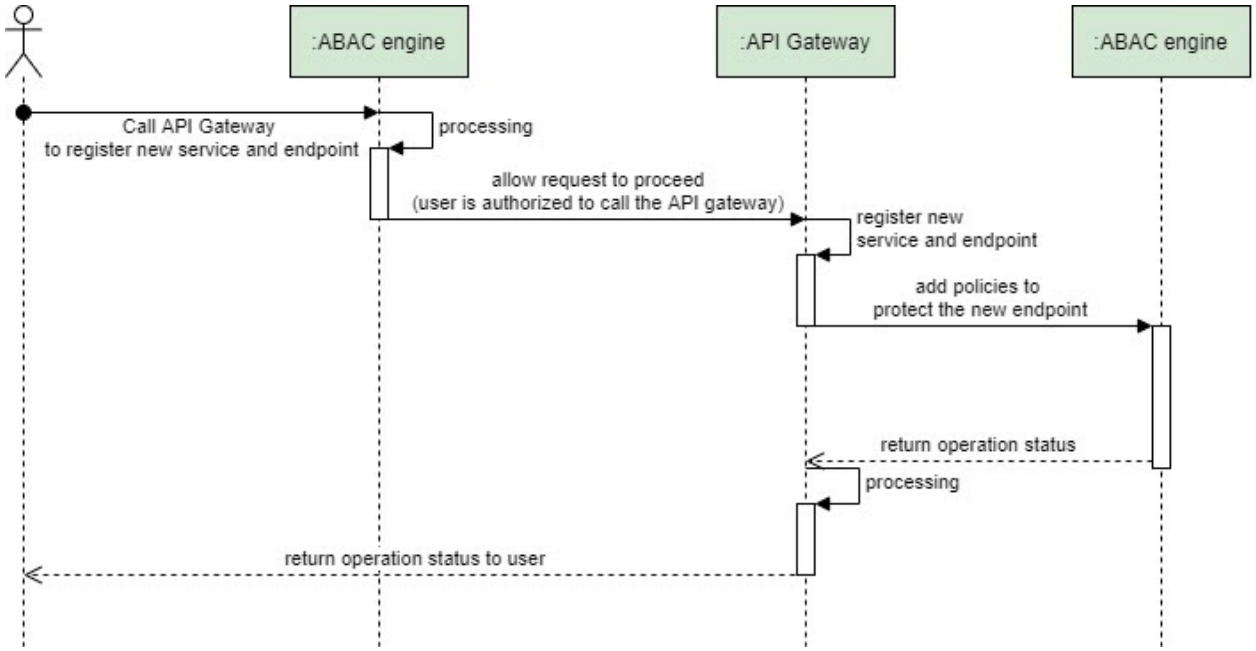

Figure 1: Expose interface

Call third-party interface which is exposed via the API Gateway

This sequence diagram illustrates the steps taken when a user calls the API Gateway so that the API Gateway can call a third-party interface (**Iface** for the sake of the example) on behalf of the caller.

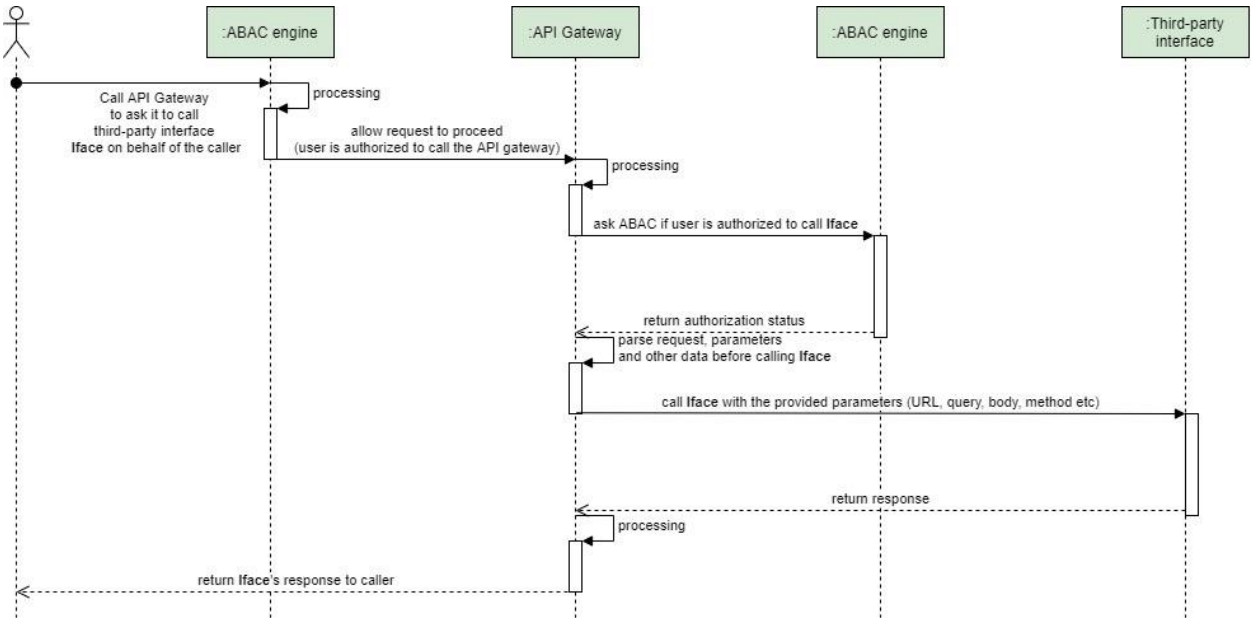

Figure 2: Call interface

5. CONCLUSIONS

In the current document we presented the integration strategy of TheFSM Applications which follows the development, testing and deployment strategy for TheFSM technical solution, as it is defined in D3.3, M12. The iterative process of the project yielded an agile approach which is well adopted by the integration strategy of the applications. The integration points (along with their APIs description) implemented and tested in the current version cover the core and fundamental integration between the applications and the platform. The testing and technical evaluation of the platform also, provided with positive feedback regarding the technical readiness of the first version of TheFSM Applications. As more use cases are covered by the platform and the applications, these tests will evolve, adapt and carefully cover the newly added functionality.

6. REFERENCES

[1] An Overview of the Testing Process | Preface. https://flylib.com/books/en/2.174.1/an_overview_of_the_testing_process.html

[2] http://www.technofunc.com/index.php/erp/178-what-is-integration-testing