D4.3: Implementation and development of systems; SSS, data processing and geo-tagged photos framework alpha versions (alpha version)

April/2021

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 870378.

Author(s)/Organisation(s)	Orestis Sampson, Nikos Iliakis, Valantis Tsiakos (ICCS)
Contributor(s)	Nikolaos Tsakiridis, Konstantinos Karyotis (i-BEC)
Work Package	WP4
Delivery Date (DoA)	30/04/2021
Actual Delivery Date	30/04/2021
Abstract:	The deliverable constitutes a report describing the technical details from the implementation and development of the alpha versions of the DIONE in-situ (WP4) components aiming to establish a complete ecosystem of low-cost and easy to use smart tools and techniques for the accumulation of the in-situ information. In particular the report focuses on (i) the microelectromechanical systems (MEMS) based sensor, the measurement protocol, the smartphone application, and the security protocols, (ii) the geo-tagged photos framework iii) the data pre-processing system, including the utilized DBMS, backend processes for data integration and validity, and techniques used for outlier detection.

Document Revision History			
Date	**Version**	**Author/Contributor/ Reviewer**	**Summary of main changes**
15/03/2021	V0.1	ICCS	Initial ToC
05/04/2020	V0.3	ICCS	Geotagged photos framework part
08/04/2021	V0.5	i-BEC	Soil scanning system, backend component
28/04/2021	V0.9	ICCS	Small modifications according to reviewers’ feedback
30/04/2021	V1.0	ICCS	Approved, final version submitted

Dissemination Level		
PU	Public	x
CO	Confidential, only for members of the consortium (including the EC)	

DIONE Consortium			
Participant Number	Participant organisation name	Short name	Country
1	INSTITUTE OF COMMUNICATION AND COMPUTER SYSTEMS	ICCS	EL
2	DIABALKANIKO KENTRO PERIBALLONTOS	i-BEC	EL
3	SINERGISE LABORATORIJ ZA GEOGRAFSKE INFORMACIJSKE SISTEME DOO	SINERGISE	SI
4	CORE INNOVATION AND TECHNOLOGY OE	CI	EL
5	NATIONAL PAYING AGENCY	NMA	LT
6	INOSENS DOO NOVI SAD	INO	RS
7	GILAB doo Beograd-Palilula	GILAB	RS
8	Cyprus Agricultural Payments Organisation (CAPO)	CAPO	EL

LEGAL NOTICE

The information and views set out in this application form are those of the author(s) and do not necessarily reflect the official opinion of the European Union. Neither the European Union institutions and bodies nor any person acting on their behalf may be held responsible for the use which may be made of the information contained therein.

Funding Scheme: Innovation Action (IA) ● Theme: DT-SPACE-01-EO-2018-2020
Start date of project: 01 January, 2020 ● Duration: 30 months

Table of Contents

1 Introduction 7
1.1 Context and background 7
1.2 Connection with DIONE toolbox 7
2 In situ soil scanning system 9
2.1 Component structure and internal architecture 9
2.2 Features deployed for soil spectrometer 10
2.3 Features deployed for android application 12
2.3.1 Installation and first launch 12
2.3.2 Data capture 13
2.3.3 Data transmission 14
3 Farmer's geo-tagged photos framework 17
3.1 Geotagged photos mobile application 17
3.1.1 Component structure and internal architecture 17
3.1.2 Mobile application design and alpha version features 20
3.1.3 Toolbox services 29
3.2 Data integrity, validation and anonymization 30
3.2.1 Outline of integrity framework 30
3.2.2 Integrity algorithms implemented 30
3.2.3 Anonymization component 32
3.3 Data transmission 34
3.4 Compliance of alpha version with technical specifications 34
4 Data processing and storage system 36
4.1 Database management system 36
4.1.1 Deployment and hosting 37
4.1.2 Security overview of RESTHeart 37
4.1.3 RESTHeart extensions 38
4.2 Data preprocessing, outlier and novelty detection techniques 40
4.2.1 Data integrity 41
4.2.2 Sensor output to standardized reflectance 42
4.2.3 Identification of outliers 42
4.2.4 Novelty detection 42
4.3 Data processing 43
4.3.1 Software interface and containerization 43

4.3.2 Model for predicting soil properties ... 44
4.3.3 Processing of geo-tagged photos ... 44
4.4 System API ... 45
4.5 Compliance of alpha version with technical specifications ... 47
5 Next steps and future outlook ... 48
5.1 In-situ soil scanning system ... 48
5.2 Geotagged photos ... 48
5.3 Data preprocessing, outlier and novelty detection techniques ... 48
Appendix A ... 50
Appendix B ... 51
Appendix C ... 54
References ... 55

List of tables

Table 1: Compliance of SSS with technical specifications ... 9
Table 2: Toolbox endpoints used by the geotagged mobile application ... 29
Table 3: Compliance of alpha version of the geotagged photos framework with technical specifications ... 34
Table 4: API endpoints for the database ... 46
Table 5: Compliance with technical specifications ... 47

List of figures

Figure 1: Spectral Engines Nirone S2.2 soil spectrometer ... 10
Figure 2: Soil spectrometer evolution through the three different versions that were developed ... 11
Figure 3: Soil Spectrometer packaging with its accessories ... 12
Figure 4: First usage pathway ... 12
Figure 5: Main interface top ribbon ... 13
Figure 6: White reference calibration ... 13
Figure 7: Soil Spectral Measurement steps ... 14
Figure 8: Geotagged mobile app connections with other components ... 19
Figure 9: a) Geotagged photos mobile application landing page, b) user login page, c) user registration page ... 20
Figure 10: My Feed page (news feed, pending Tasks) ... 21
Figure 11: My parcels page (declared parcels list) ... 21
Figure 12: Parcel Tasks ... 22
Figure 13: Notifications list ... 22
Figure 14: Mapbox navigation to the Parcel ... 23
Figure 15: Mapbox text directions to the Parcel ... 23
Figure 16: a) Restrictions and guidance to allowed photo collection spot b) Parcel borders and allowed photo collection spots, c) Photo taken preview (without AR content) ... 24
Figure 17: List of photos taken for a specific Task ... 26
Figure 18: Existing photo preview ... 26

Figure 19: Dual frequency GNSS signals concept 27
Figure 20: EDAS high level architecture 28
Figure 21: Example of anonymized photo 33
Figure 22: Schematic overview of the mobile and backend process of the geotagged photos integrity, validation and anonymization framework 33
Figure 23: A simplified view of the data processing and storage system 36
Figure 24: Requests in RESTHeart should be both authenticated and authorized. 38
Figure 25: The Google sign-in flow for obtaining an access token for the server-side application. Image taken from https://developers.google.com/identity/sign-in/web/server-side-flow 39
Figure 26: Authentication scheme for integrity requests 40
Figure 27: Geotagged photos backend component connections 45
Figure 28: DIONE toolbox high-level architecture 50
Figure 29: Spatula flattened soil surface 52
Figure 30: Undisturbed sampling tube placement with the aid of hammer 52
Figure 31: Undisturbed sampling tube marginally concealed to the ground 52

List of Abbreviations and Acronyms	
API	Application Programming Interface
CAP	Common Agricultural Policy
EGNOS	European Geostationary Navigation Overlay Service
EO	Earth Observation
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
LC/LU	Land Change / Land Use
MEMS	Microelectromechanical systems
ML	Machine Learning
OTA	Over the Air
SSL	Soil Spectral Library
SSS	Soil Scanning System
TLS	Transport Layer Security
WP	Work Package

1 Introduction

1.1 Context and background

The main objective of the deliverable is to provide an overview of the alpha version of the DIONE in-situ components for complementing EO-data which are developed within the frame of the DIONE project (WP4). These tools aim to establish a complete ecosystem of low-cost and easy to use smart tools allowing for the accumulation of the in-situ information, thus providing near real-time ground truth data. The implementation of these tools was based after considering the user requirements and specifications which were defined in previous work.

The first tool involves the use of point in-situ spectroscopy that, aided by the recent innovations in miniaturizing and reducing the cost of spectrometers, provides ground truth soil observations of key parameters. To do so, the recorded in-situ spectra are transferred to a central data processing and validation system which will translate the spectra to key properties via machine learning methodologies.

The second component pertains to the use of geo-tagged photos which are captured using smartphones and allow for an improved method for the provision of additional evidence regarding the Common Agricultural Policy (CAP) compliance monitoring while simultaneously ensuring that the process is untampered. The added benefit is the quality and trust of the transmitted data as well as the application characteristics with respect to location accuracy and data collection process through the use of augmented reality features.

Finally, a central data processing and storage system concentrates the data from these two tools and ensures their quality and provides user-friendly APIs for the curated data contained. The system ensures that only authorized parties are used as sources of the received data, that the integrity of the data and metadata is not compromised, that no data with highly variable/outlier nature are going to be used and that appropriate forms of data storage will be used to ensure easy retrieval.

It should be noted that data coming from the first tool (miniaturised spectrometers) shall be further exploited and will be translated to key soil properties that will be combined also with open EO data to develop soil maps for the project pilot areas. The latter will be analysed in the next version of this report.

1.2 Connection with DIONE toolbox

The DIONE toolbox constitutes an integrated EO-based toolbox exploiting automated technologies to ensure more frequent, accurate and inexpensive CAP area-based compliance checks while also enabling the assessment of the respective environmental impact. DIONE toolbox consists of three overarching components:

i. DIONE Earth Observation component, that aims to harness DIAS/Sentinel Hub to support data management and processing of EO data, area monitoring markers, drones, data fusion, and super resolution techniques towards the provision of enhance resolution maps of permanent pastures, crop-types, non-productive EFA types and farmers' activities (i.e. grassland mowing/ploughing, etc);
ii. DIONE In-situ component, whose components are analysed in the context of this report
iii. DIONE Green Accountability component that includes a compliance monitoring tool, which decides on beneficiaries' compliance and is integrated with the existing tools of paying

agencies and an AI-enabled environmental performance tool, accompanied with a visualization engine.

A schematic overview of the high-level DIONE toolbox architecture as elaborated in deliverable D2.2 is presented in Appendix A.

The outputs from the tools discussed in this report are further utilized by other DIONE toolbox components in order to facilitate monitoring and evaluation of the compliance with respect to the CAP provisions (i.e. use of geotagged photos as additional evidence when compliance assessment can be achieved through the EO-based products) as well as the assessment of the environmental performance (i.e. use of key soil properties for the computation of relevant metrics for the environmental performance tool, etc).

Last but not least, integration between the various components is achieved through the DIONE toolbox API, supporting data retrieval and storage, integration with external infrastructure (Paying Agencies systems) and identity and access management whilst interoperability considerations for the final data provision is supported by a Spatial Data Infrastructure.

2 In situ soil scanning system

The estimation of soil properties with the use of reflectance spectroscopy has met significant advances during the last years. Due to the destructive nature of traditional laboratory techniques, the need for development of alternative methods lead to the further research of different technologies and instrumentations. This effort is actively supported through the development of DIONE's Soil Scanning System (SSS), part of DIONE's toolbox, which is based on a suitably selected low-cost spectral sensor employing microelectromechanical systems (MEMS) technology. The SSS comprises two key components: the soil spectrometer and the Android application. This chapter contains technical and implementation details for the development of the alpha versions of the soil spectrometer component and the Android application as well.

2.1 Component structure and internal architecture

The system is designated to perform the following actions, as presented in "D2.2: DIONE toolbox specifications and software architecture" and further analysed in "D4.1: Technical specifications of the in-situ soil scanning system (SSS), data processing system and farmer's geo-tagged photos framework":

- The soil spectrometer will acquire the spectrum and transmit it to the Android device via a Bluetooth connection.
- The Android application will be able to trigger the soil spectrometer and initiate the measurement capture service and store it alongside the relative metadata to the Android device's memory.

In addition, the measurements that are captured from the soil spectrometer and stored to the Android device internal memory need to be transmitted to a Central Database, enabling the interconnection of SSS with the rest of DIONE's components.

Furthermore, the development of each system component aims to address the user requirements and technical specifications analysed deliverables D2.2 and D4.1 respectively, through the integration of functionalities covering the recognized needs. The state of each functionality is listed at Table 1.

Table 1: Compliance of SSS with technical specifications

Component	Requirement ID	Short description	Included in alpha version
SSS	SYS_F_SPEC_1	Bluetooth transmission of collected data and associated metadata from soil spectrometer to Android device	YES
SSS	SYS_F_SPEC_2	The smartphone application must include a component to capture a photo of the soil sample using the camera integrated on the smartphone and provide a visual	YES

		update of soil sample and facilitate improper usage identification	
SSS	SYS_NF_USE_1	The smartphone application must support recent Android versions (e.g. >= 5.0.0, Lollipop) and be easily installable through the standard package manager i.e., without requiring root permissions.	YES
SSS	SYS_NF_USE_2	The smartphone application should support multiple languages	PARTIALLY – Supports but the translations will be added in the final version
SSS	SYS_F_SPEC_4	The smartphone application should be able to visualize the acquired spectrum	YES

2.2 Features deployed for soil spectrometer

The soil spectrometer lies to the core of the SSS since all soil related data inflow for this component source from it. To this end, the soil spectrometer's internal characteristics and specifications predefine the range of applications that can be developed and are based on accumulated soil spectra. After a thorough market research, Nirone S2.2 sensor from Spectral Engines Company was selected (detailed description regarding the sensor selection can be found at "D.4.1: Technical specifications of the in-situ soil scanning system (SSS), data processing system and farmer's geo-tagged photos framework"). The S2.2 sensor adopts a patented Fabry–Pérot interferometer covering the spectral range from 1750 to 2150 nm, with a step of 5 nm. Through the analysis of this spectral region of the electromagnetic spectrum, many key soil properties can be identified, providing helpful information for the soil condition. It has been proven that the presence of Soil Organic Carbon (SOC), Calcium carbonate (CaCO3) or soil texture particles (sand, silt and clay) lead to variations on electromagnetic reflectance at specific bands of Visible, Near InfraRed and ShortWave InfraRed (VNIR– SWIR) range, [1] indicating high concentration of the above mentioned soil components, chemical compounds or particles.

The soil spectrometer comes with a built-in illumination that consists of two tungsten vacuum lamps (estimated life expectancy > 40.000 hours). The soil spectrometer's dimensions are small (25x25x17.5mm3), weighing 150g and in combination with the 5V re-chargeable battery that comes with it, all portability characteristics are met. It is moreover able to operate in situ and withstand ambient temperatures between 10° and 50°C. The spectrometer further features a Bluetooth low energy module that is used for the communication with smartphone applications.

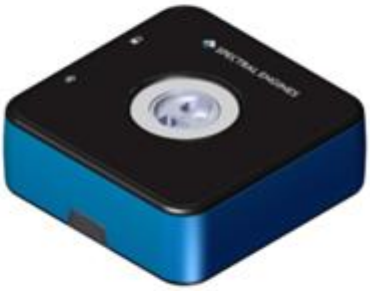
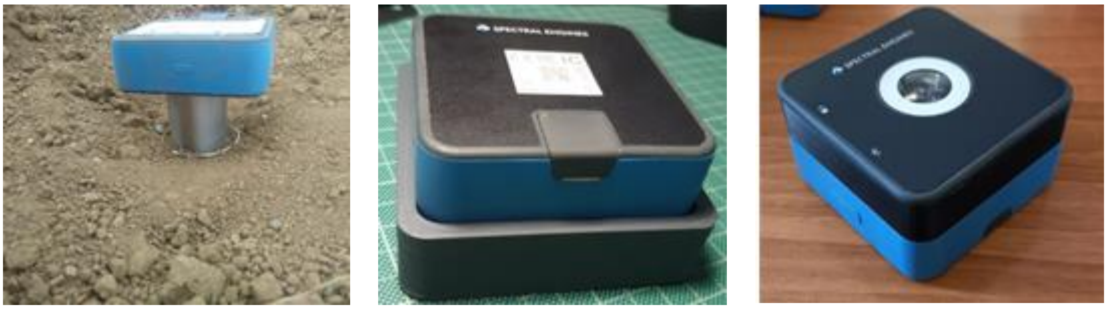


Figure 1: Spectral Engines Nirone S2.2 soil spectrometer

The version that is commercially available from Spectral Engines (Figure 1) covers a wide variety of applications ranging from food quality to powder analysis or plastics. In order to acquire high quality soil spectral measurements, a significant modification related to the sensor's spectral scanning area was identified as crucial. The factory-ready version comes with a limited scanning field, unable to cover the minimum soil particle's size limit, which is 2mm [2]. To overcome this obstacle, the increase of distance between the soil sample and the sensor was proposed. As described in detail at D.4.1, the spectrometer was redesigned and customized to meet the above soil spectroscopy principle. For the production of the new version, a synthetic bracket was 3D printed and mounted to the sensor, increasing the distance from the sensor's aperture to the sensed surface by 20 mm and eventually increasing the sensing area to 3.3mm diameter, meeting the initial objective. The evolution of the development of the bracket can be shown in Figure 2.



(a) *(b)* *(c)*

Figure 2: Soil spectrometer evolution through the three different versions that were developed

The first version contained an optic cylinder Figure 2 (a) and was used to test the suitability of the above-described modification for the acquisition of measurements that will be consistent under various repetitions, in terms of distance metrics, and similar to spectral signatures of the same soil samples taken with the use of a full VNIR–SWIR laboratory spectrometer. The obtained results indicated that this modification would enhance the spectrometer's suitability for soil scanning, thus two upgrades were proposed. The next version comprises a detachable bracket Figure 2 (b) that acts as a spacer between the sensor and the soil, leaving the spectrometer untouched. Since the spectrometer is developed over the scope of user-friendliness, the bracket mounting to the spectrometer would entail an additional step in soil scanning procedure, affecting the simplicity. To this end, a newer version, which is the soil spectrometer alpha version, was developed, having the bracket integrated as depicted in Figure 2 (c). This modification enables the users to acquire representative soil spectral signatures without increasing the operational complexity.

It is noted that at each successive iteration of the soil spectrometer detailed spectra were recorded from a small open spectral library of the eastern Mediterranean region. These spectra were compared with a higher precision spectroradiometer to analyze the degree of compatibility between the two measurements and ensure the validity of the results.

The version to be distributed to end users will contain the following units (Figure 3):

- The soil spectrometer device

- The white calibration panel, which is a spectrally featureless material with high reflectance used as a reference panel (for further information please refer to paragraph 2.3.2)
- USB 2.0 male to USB-C male cable for charging
- Detailed user's guide in English, Greek and Lithuanian and an English narrated video tutorial for proper usage of the spectrometer

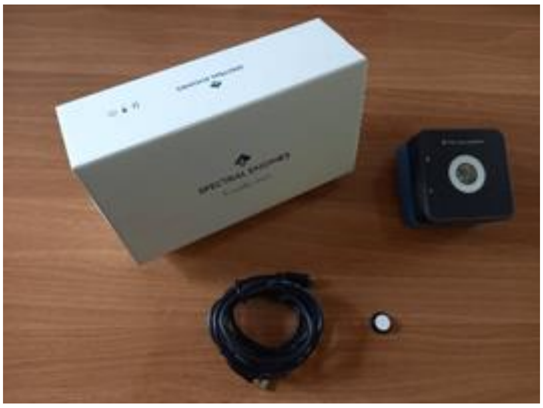


Figure 3: Soil Spectrometer packaging with its accessories

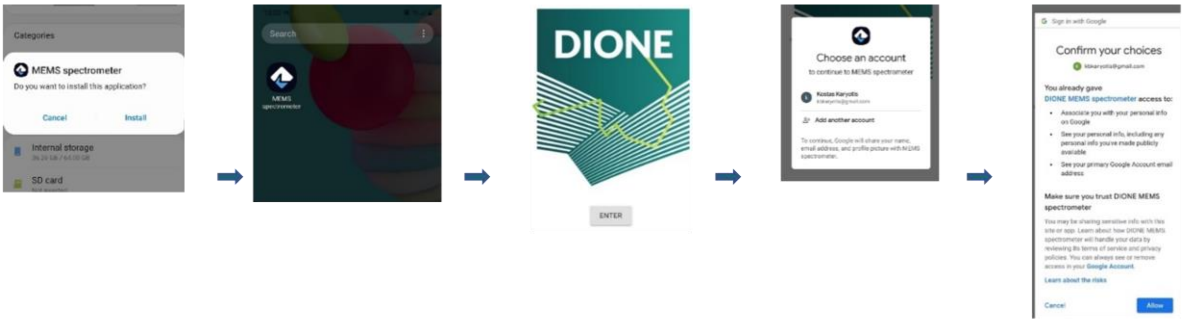
2.3 Features deployed for android application

The second key component of the SSS is the Android application. This application is comprised of a Graphical User Interface that facilitates the interconnection of the soil spectrometer with the Android device. As described before, the connection is performed over Bluetooth. The application was developed with the help of the Capacitor Ionic Framework and is distributed through an apk file. The user is prompted to provide access to the Android device's location, storage space, and enable the Bluetooth connection in order for the application to be fully functioning.

2.3.1 Installation and first launch

The installation apk is delivered to end users along with the soil scanner. It is easily installable to every Android device with Android version 5.0 or newer by enabling the installation from unknown sources feature. The authentication is performed through Google Sign-in at this version (which will be substituted from DIONE's Single Sign On (SSO); more information about this may be found in Section 4.1). The current procedure is semi-automated, since the user's Google-account associated email address should be registered to the central database prior to using the system. Since the device is distributed to a restricted number of authorized users, this adds an extra layer of security, and further enables the monitoring of the data inflow to the central database.

The pathway of the first use is depicted below.



Do you want to install this application?

to continue to MEMS spectrometer

To continue, Google will share your name, email address, and profile picture with MEMS spectrometer.

You already gave DIONE MEMS spectrometer access to:

- Associate you with your personal info on Google
- See your personal info, including any personal info you've made publicly available
- See your primary Google Account email address

Make sure you trust DIONE MEMS spectrometer

You may be sharing sensitive info with this site or app. Learn about how DIONE MEMS spectrometer will handle your data by reviewing its terms of service and privacy policies. You can always see or remove access in your Google Account.

Learn about the risks

Figure 4: First usage pathway

The main interface is characterized from simplicity, aiming to guide the user to the final measurement step-by-step. The top of the screen contains the main ribbon where the user's unique ID is located, and the user can also access the main menu and the help section. The main menu is under the "sandwich menu" icon containing information related to the application release. The help menu will host in future versions text and graphical guides for the SSS usage.

Help
KOSTAS
DIONE MEMS
Spectrometer device
Help
KOSTAS
About
E MEMS

Figure 5: Main interface top ribbon

2.3.2 Data capture

The procedure for spectral measurement is straightforward and consists of three key steps[1].

1) The initial step that is mandatory for the first measurement consists of the acquisition of white reference measurement, which is essential in calculating the reflectance of the samples that will be taken through the comparison to a baseline measurement of a featureless material with minimum absorbance. This material is the white calibration panel that comes with the soil spectrometer and is a synthetic white disk which is characterized by extremely high diffuse reflectance (>99%). It is Lambertian, chemically inert and thermally and environmentally stable. Before taking the measurement, the user must have the device with the sensors' aperture facing up, and place the white reference aligned to the center of the sensor with the white surface touching the sensor's protective glass. The user after launching the application is automatically guided to the above-described step, which is performed by tapping the "WHITE REFERENCE" button. Then they are prompted to select the available spectrometer and connect via Bluetooth. For this reason, the Android device's Bluetooth module must be enabled with pairing mode enabled also. After the pairing is completed, the measuring is initiated, taking about 5 seconds to complete.

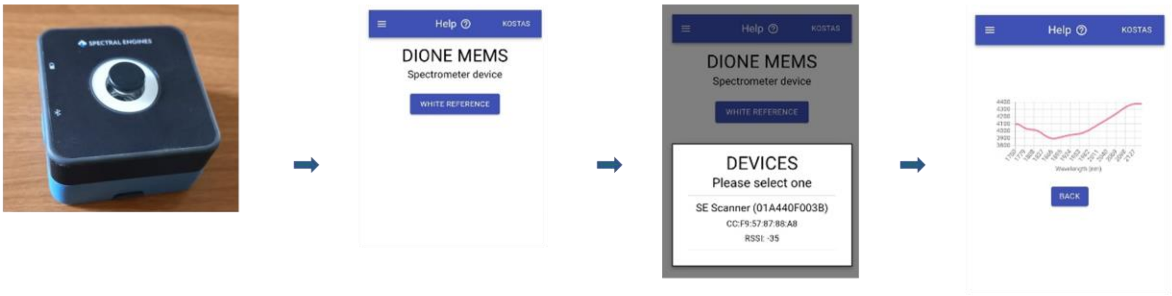
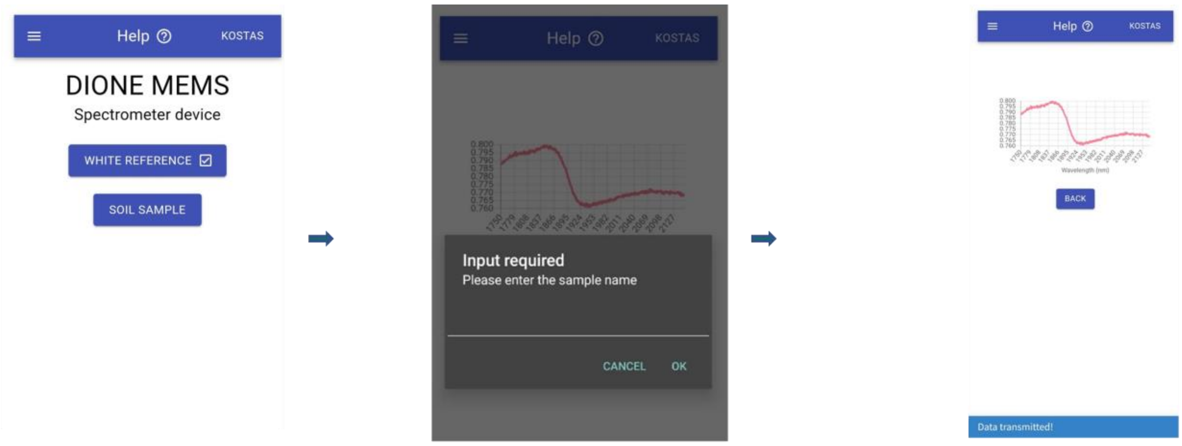


Figure 6: White reference calibration

2) With the completion of the white calibration, the button "SOIL MEASUREMENT" is enabled, and placed below the "WHITE REFERENCE" button. The user needs to place the soil spectrometer on top of the soil sample, with the sensor facing the soil and then taps on "SOIL MEASUREMENT" button. The procedure is exactly the same as the white reference measurement, meaning that the user will be asked to select with which device will be paired and will take about the same amount of time.

[1] A walk-through video containing all needed steps for the operation of soil measurement can be found through the following url: https://www.youtube.com/watch?v=CWP1ODlx0AY
It is noted that the device used in this video is the version before alpha version needing the detachable mounting bracket to operate. A video with the final version will be produced when all components will be finalized



Input required
Please enter the sample name

Data transmitted!

Figure 7: Soil Spectral Measurement steps

3) When the measurement is taken, the camera is enabled, and the user is asked to capture a photo of the exact part of the sample that was measured. This step entails a high-level part of novelty and abnormality detection, aiming to identify measurements that have not been taken properly. Then, the user is prompt to associate the instance with a measurement name. The procedure is finalized when the data are transmitted to the Central Database. This is performed either instantly if a stable internet connection is found, or at a later stage when the user establishes internet connection and taps the sync button. The procedure is completed when the success message appears.

The SSS is instantly ready for a new measurement, for which the same white reference measurement can be used or a new white reference measurement. For further details regarding the measuring protocol, please refer to Appendix B.

2.3.3 Data transmission

The collected data are stored to the Android device in a .json file including the sensed irradiance, its conversion to reflectance and all the relevant metadata (e.g. latitude and longitude, timestamp etc.). In absence of stable internet connection, the data are transmitted at a later stage as mentioned before. This means that the application can be used at fields where no mobile network coverage exists.

A sample json document is as follows:

```
{
  "user": {
    "givenName":
    "familyName":
    "email":
  },
  "device": {
    "address":
    "name":
    "rssi":
    "advertisement":
    "status": "scanResult"
  },
  "sample": {
    "cancelled":
    "value":
  },
  "date": {
    "ISO":
    "date":
    "time":
    "epoch":
  },
  "location": {
    "coords": {
      "latitude":
      "longitude":
      "accuracy":
      "altitude":
      "altitudeAccuracy":
      "speed":
      "heading":
    },
    "timestamp":
  },
"whitereference": [
Irradiance per sensed wavelength of white reference panel],
"whiteparams": {
    "MeasSettings": {
      "Wavelengths": {
        "Mode":,
        "MinWL":,
        "MaxWL":,
        "Step":
      },
      "ScanAverage":,
      "WavelengthAverage":,
      "AutomaticDark":,
      "SubtractDark":,
      "LightSourceControlMode":,
      "LightSourcePowerLevel":,
      "IsWhite":
    }
  },
"irradiance": [
Irradiance per wavelength of white sample],
  "irradianceparams": {
    "MeasSettings": {
      "Wavelengths": {
        "Mode": 1,
        "MinWL": 1750,
        "MaxWL": 2150,
        "Step": 1
      },
      "ScanAverage": 1,
      "WavelengthAverage": 100,
      "AutomaticDark": true,
      "SubtractDark": true,
      "LightSourceControlMode": true,
      "LightSourcePowerLevel": 90,
      "IsWhite": false
    }
  },
  "reflectance": [
Irradiance converted to reflectance per wavelength of soil sample
]
  "photo": {
    "format": "",
    "exif": {},
    "base64String": ""
  }
}
```

It is noted that the photo captured by the cameras is stored as a base64 encoded string, while all the relevant EXIF metadata (including e.g., aperture value, exposure time, focal length, etc.) are also recorded.

3 Farmer’s geo-tagged photos framework

DIONE farmers’ geotagged photos framework aims to complement the Earth Observation data sources with reliable ground-based information about agricultural parcels and thus facilitating CAP compliance monitoring. In this context, the framework comprises different components and technical innovations towards assisting and guiding users to capture efficiently representative photos of their parcels while adhering to current technical recommendations and ensuring the security, validity and reliability of the collected photos.

3.1 Geotagged photos mobile application

The data collection process is supported by a mobile application (frontend) that exposes to the user all the related content about their parcels while enabling the conclusion of the process and the provision of the final photos to the Paying Agencies. In this context, various processes are employed so as to enable among other things the provision of the necessary instructions for farmers to reach a given parcel, the reception of notifications about tasks they need to undertake as well as directions regarding the process of capturing appropriately a photo of a given parcel.

3.1.1 Component structure and internal architecture

The application is developed in the Unity game engine[2] to benefit from the integrated Augmented Reality solution (ARFoundation[3]) that ships with the engine.

Unity is a cross-platform game engine developed initially as a Mac OS X-exclusive game engine. As of 2018, the engine had been extended to support more than 25 platforms. The engine can be used to create multi-dimensional, virtual reality, augmented reality applications.

AR Foundation is a cross-platform framework built for the Unity engine that allows to build augmented reality experiences once, then build for either Android or iOS devices. The package presents an interface for Unity developers to use, but doesn't implement any AR features itself. To use AR Foundation on a target device, separate packages are also needed to target platforms officially supported by Unity:

- ARCore XR Plugin on Android
- ARKit XR Plugin on iOS

Along with Unity, some native android plugins are developed, mainly to handle the low-level operations required for the raw measurements handling and integrity aspects. The augmented reality component aims to provide directions to farmers in order to enable retrieval of representative photos of a given parcel.

Further to this, a trip planning /navigation part is available in the mobile application, being responsible for providing the map tiles needed for the correct representation of the map while also providing the required information for creating a route between the user location and the location of the parcel for which geotagged photos shall be collected.

During the data collection process, a time and location integrity module ensures that the application user cannot tamper with the device location and time by manually setting it. The DIONE platform has strict limitations since the photos taken must represent the real state of a parcel at a very specific time and day. Additionally, some preprocessing steps are taking place in the mobile application in order to

[2] https://unity.com/

[3] https://unity.com/unity/features/arfoundation

enable at a later stage the server-side validation and integrity check of the collected photos. These include cryptographic techniques (signing the photo files with keys that are created for the encoding and decoding) as well as the extraction of relevant data (i.e. Toolbox session, public key, device camera focal length) required by the integrity checks algorithms (see section 3.2).

To communicate downstream to the geotagged mobile application, a service to facilitate the provision of push notifications is in place (OneSignal[4]). It provides an API to deliver messages across various platforms, abstracting details such as the platform the device is running on.

The mobile application also requires some functionalities from the DIONE Toolbox API. A farmer can register to use the DIONE services through the application by providing a unique identification originally provided by their Paying Agency. By registering, the user is provided with credentials with which they can authenticate themselves and access all the relevant data stored in the DIONE Spatial Data Infrastructure that is dynamically connected with the Paying Agencies' systems.

The photo taken, along with the required metadata, is uploaded to be verified by the data integrity, validation and anonymization component of the geotagged photos framework (section 3.2) and to be stored in the Central Database.

[4] https://onesignal.com/

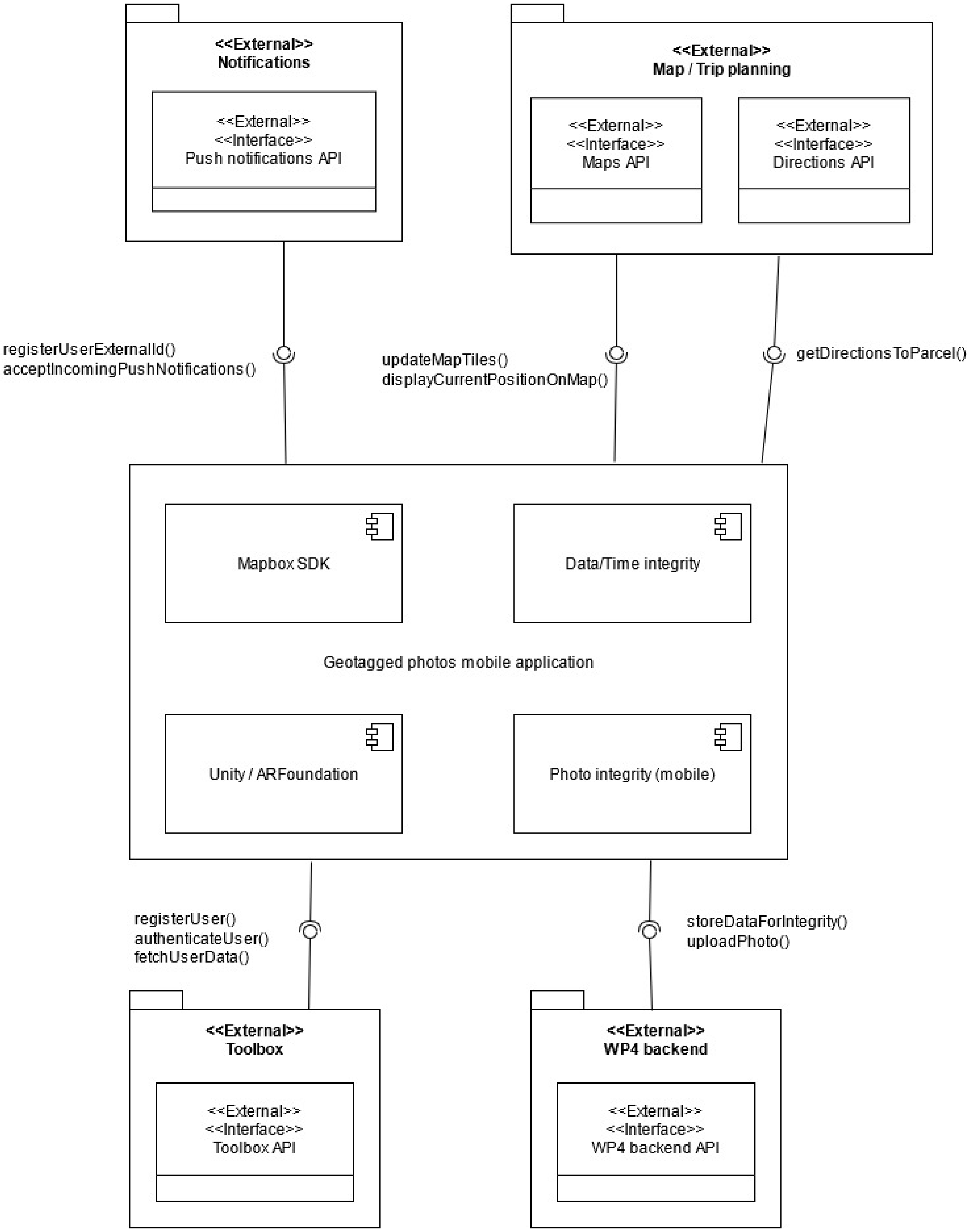


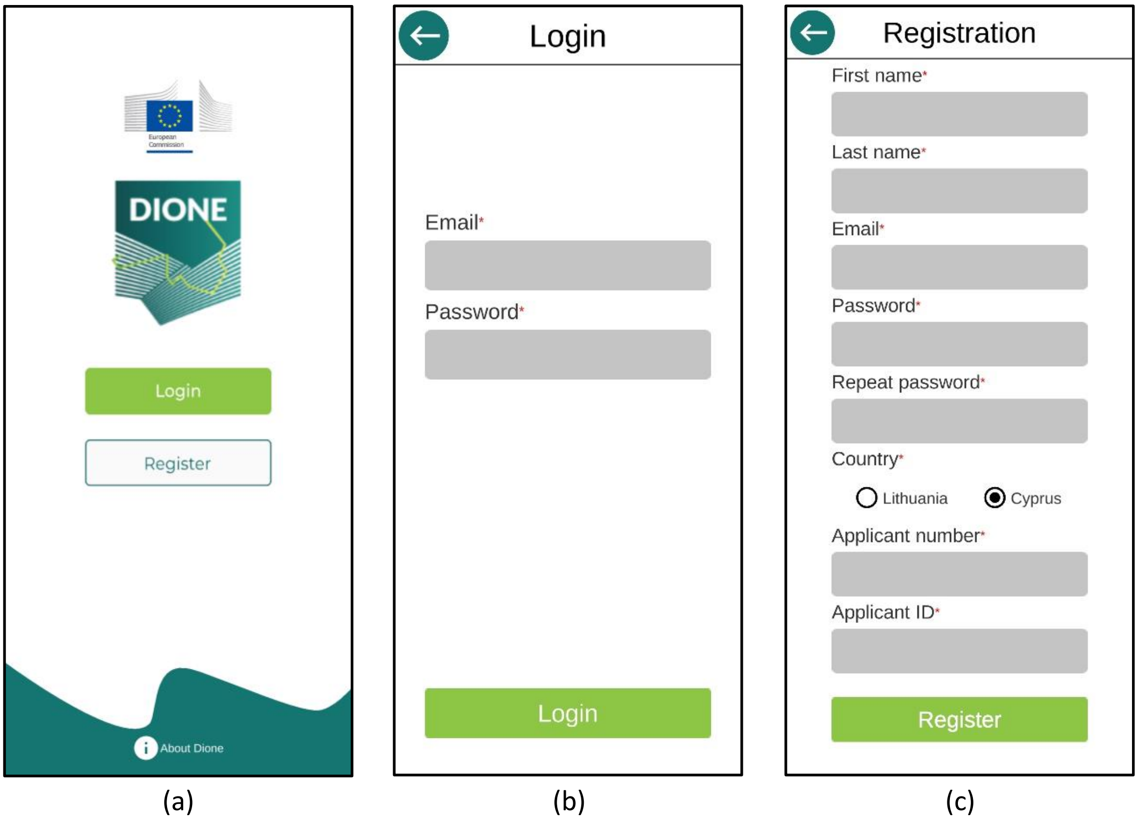
Figure 8: Geotagged mobile app connections with other components

3.1.2 Mobile application design and alpha version features

The alpha version of the geotagged photos mobile application covers all the essential steps of the data acquisition process. This ranges from the reception of notifications from other toolbox components (DIONE compliance monitoring tool) about the provision of additional evidence (parcels photos) to the visualization of related content and the provision of guidance towards the collection of suitable photos in the context of CAP monitoring. The stepping stone for the realization of the mobile application were the system specifications defined in deliverable D4.1 that were closely monitored during the development process in order to ensure that all technical aspects and user needs are met. Additionally, the initially defined wireframes have been evolved and a re-design of the application user interfaces took place during this period following interaction and assessment from the Paying Agencies (NPA, CAPO). The currently designed features and application functionalities are presented below.

Home screen – Authentication/Authorization

The geotagged mobile application will not be open for general use, it will be available only for registered users (farmers). In order to achieve this, the user must have registered themselves in the DIONE platform. The registration process requires several personal information to be stored for user security (password) and authentication (name, applicant identification). The process of verifying the user happens in two stages. The first step is the DIONE platform itself where verification happens by the user receiving an email in the declared address and clicking a verification link. The second step, is performed by the DIONE platform by checking if this user corresponds to some entry in the selected Paying Agency database. This happens by providing some form of applicant identification required by each Paying Agency that ties the specific DIONE user to their data in the Paying Agency's backend infrastructure.



(a) (b) (c)

Figure 9: a) Geotagged photos mobile application landing page, b) user login page, c) user registration page

Upon successful registration, the user may enter their credentials to login. By doing this, the app fetches all the relevant user data (parcels, locations, tasks etc), by querying the respective DIONE Toolbox API endpoints, that are crucial to populate all the different pages, menus and support all the functionalities.

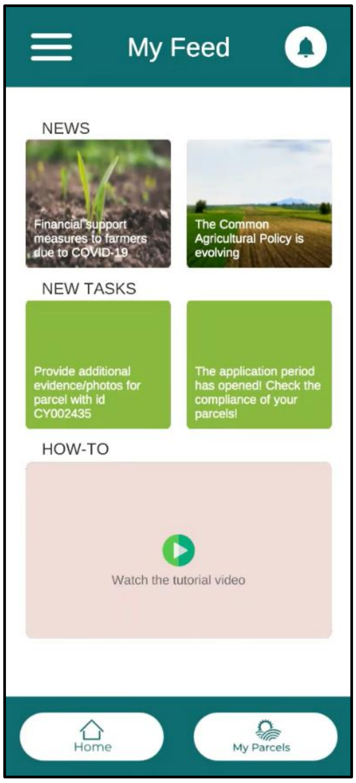
Content visualization

Following user authentication, the application allows the visualization of relevant content associated with the user. The main pages of reference in the mobile app are the “My feed/Home” page and the “My parcels” one. The former displays recent news from the respective Paying Agency (i.e. aspects of importance with respect to the application period, news on CAP implementation in their country, etc) along with the latest pending Tasks for the user, while the latter displays all the user’s declared parcels.

A Task is an action required from the farmer by the Paying Agency Inspector. It is related to a specific parcel and its location can be specified by the Inspector. Each parcel has its own unique page in the mobile app to host the various Tasks related to it.

However, a user can also act proactively and facilitate the compliance assessment process for their parcel, without receiving notice from the PA Inspector. For that cause, a Task is added by default in every Parcel that can be used by the farmer to take photos and upload them towards the DIONE platform before any action may be required by the Agency.

Easy access and navigation between these two pages (“My feed”, “My parcels”) is supported through dedicated buttons in the bottom of the screen. Additionally, the parcel list and content provided (i.e. crop type) is dynamically extracted from the data transmitted via the DIONE toolbox API.



Financial support measures to farmers due to COVID-19

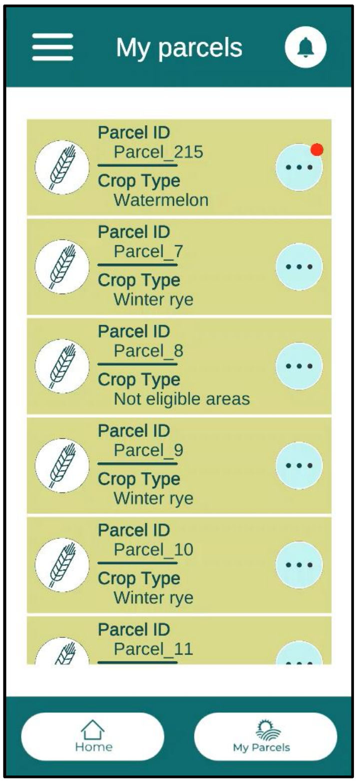
The Common Agricultural Policy is evolving

Provide additional evidence/photos for parcel with id CY002435

The application period has opened! Check the compliance of your parcels!

Watch the tutorial video

Figure 10: My Feed page (news feed, pending Tasks)



Parcel ID Parcel_215
Crop Type Watermelon

Parcel ID Parcel_7
Crop Type Winter rye

Parcel ID Parcel_8
Crop Type Not eligible areas

Parcel ID Parcel_9
Crop Type Winter rye

Parcel ID Parcel_10
Crop Type Winter rye

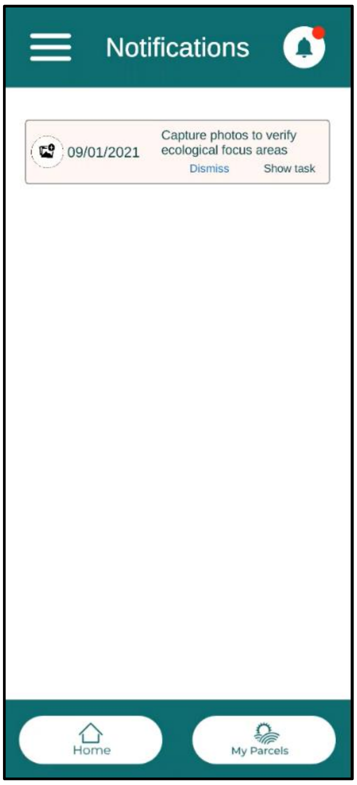
Parcel ID Parcel_11

Figure 11: My parcels page (declared parcels list)

Push notifications

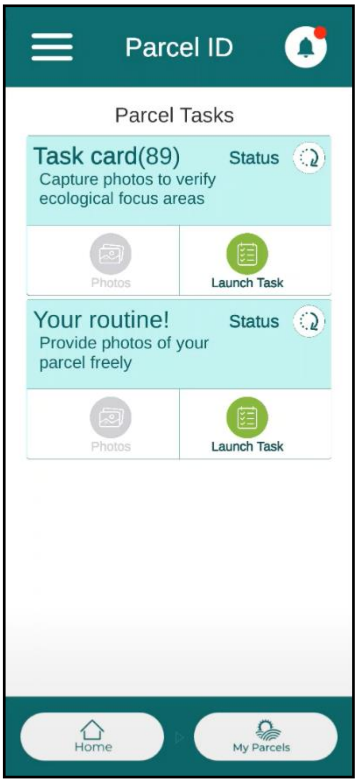
A Paying Agency Inspector may create a new Task for the user at any time through the Compliance monitoring tool. In order to notify the user efficiently, a new Task in the DIONE Toolbox API, triggers a push notification to be sent towards the mobile device. The notification includes a brief description and if selected, opens the geotagged mobile application. Even if the user misses the push notification, they can find them later in the notifications page of the application. Also, while in the application, a red round icon on the notification icon and on the icons in the Parcels list also lets the user know that there are unread notifications or pending Tasks respectively. Each entry in the notifications page, can be dismissed and deleted or selected to open the related Parcel Tasks page where all the relevant Tasks are listed. The Tasks have an associated status based on their progress (photos uploaded, failed integrity check, completed etc).

The push notification system for the mobile application, relies fully on the OneSignal service. This service, primarily works by assigning player IDs to each application user. While this is the common way to go with, the Toolbox API authorization mechanism provides also unique user ids and thus it was utilised in the context of this application case. The OneSignal API, allows the developer to manually assign IDs that are created by an external entity (i.e. Toolbox API), hence managing to avoid multiple ids for different services.



09/01/2021 Capture photos to verify ecological focus areas

Figure 13: Notifications list



Task card(89)
Capture photos to verify ecological focus areas

Your routine!
Provide photos of your parcel freely

Figure 12: Parcel Tasks

Navigation to parcel and defined spots

After pushing the button to launch a specific Task, the user is presented with a map view provided by Mapbox[5]. Its purpose is to guide the user to the Parcel this Task is associated with. In this screen several points can be identified:

- the user position is marked on the map
- a route is rendered between the user position and Parcel position
- the parcel itself is drawn
- a button to change the map layers
- a button to focus the map view to the user position
- a button to display a page with text directions to the Parcel
- a button to switch to the AR session

The mapping platform of choice is the open source Mapbox which along with its very useful APIs, provides the building blocks for a complete position solution. Mapbox is a provider of custom online maps for websites and applications whose data is taken from open data sources, such as OpenStreetMap and NASA, and from purchased proprietary data sources. The Mapbox SDK that is used in the context of the mobile application, constitutes an open-source toolset for building mapping applications for Android devices. An essential part of the SDK if the Native Location Provider that allows the application to make use of the native Android positioning module. That way, the Unity mechanism for position can be overridden and along with it, the low precision it offers.

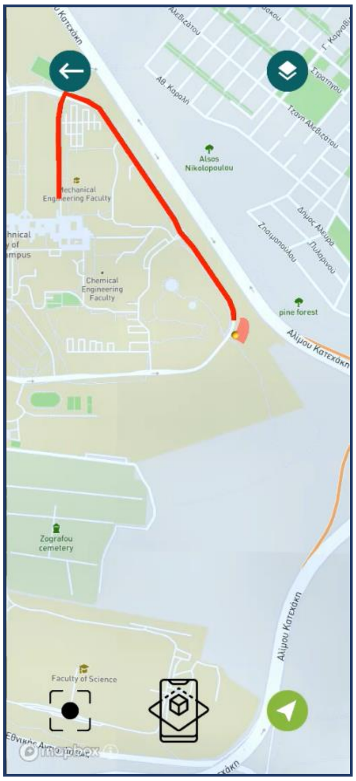
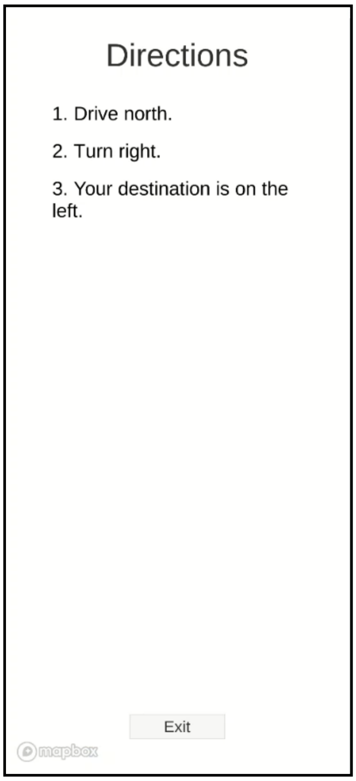


Figure 14: Mapbox navigation to the Parcel



Directions

1. Drive north.
2. Turn right.
3. Your destination is on the left.

Figure 15: Mapbox text directions to the Parcel

[5] https://www.mapbox.com/

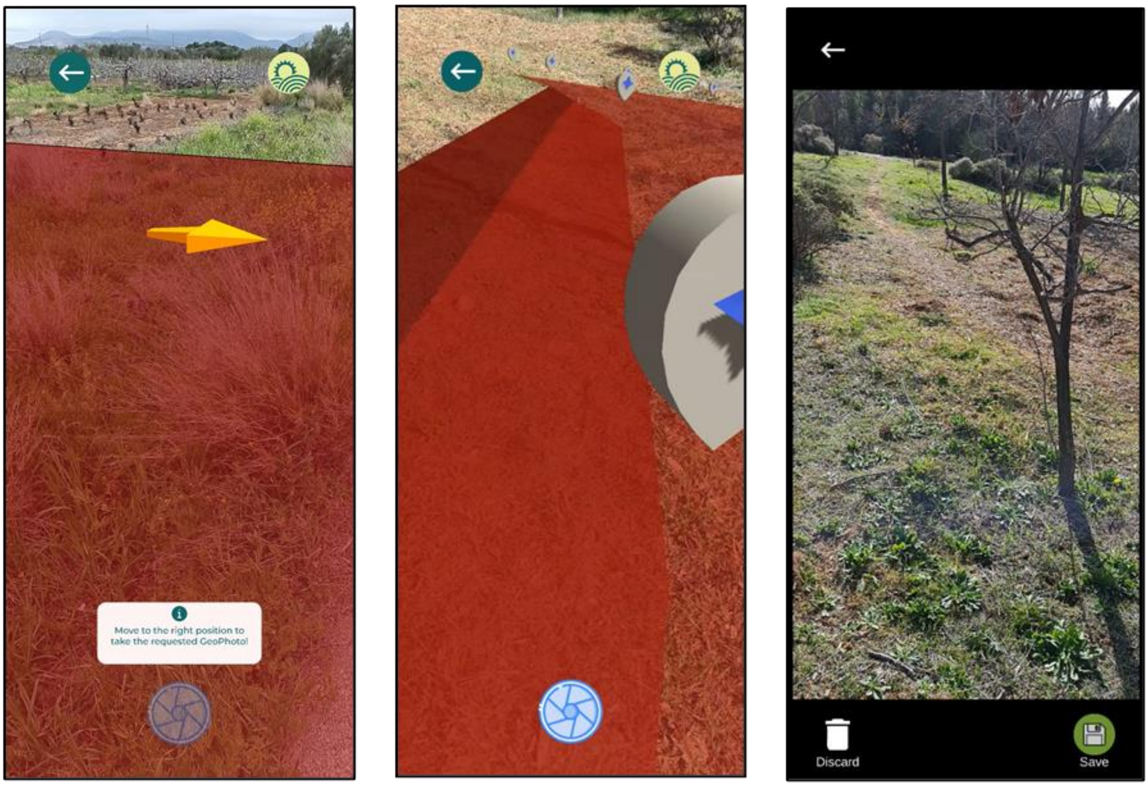
Users can identify their position on the Mapbox map by a marker and the related parcel is drawn as a different layer. Also available, is the routing information from the user's position to the parcel location.

This functionality is provided by the Mapbox Directions API external service. The routing information is denoted on the map by connecting the user's position and destination, along with turn-by-turn text instructions.

Augmented reality photo capture

When requesting geotagged photos from the farmer, the intension is to obtain sufficient information in order to avoid any physical field visits (on-the-spot-checks) by the Paying Agencies' Inspectors. Therefore, the collected images should provide an overview of the parcel, but not necessarily cover its entirety and all the details. In order to assure a comprehensive view on the element and to limit the possibility of image manipulation, it is recommended to provide at least 2 photos of the element captured from different viewpoints or camera heading.

It is advised to capture photos in landscape format (horizontally) and point the camera so that the element to evidence is depicted in the image centre. Considering the image content, photos may be divided into two main framing categories of "overview" and "macro" photos. An overview photo should depict a larger part of the field and include landscape elements other than the main object (crop, activity etc.), if possible. This type of photo aims at reducing the uncertainty linked with the limited accuracy of the geotag and at providing an overview of the field condition. A macro photo must serve to enable the robust identification of the element to evidence. This subject could be a mixture of crop as Ecological Focus Area (EFA) cover, presence of rare crops that cannot be reliably discriminated in the Sentinel data etc.



Move to the right position to take the requested GeoPhoto!

(a) (b) (c)

Figure 16: a) Restrictions and guidance to allowed photo collection spot b) Parcel borders and allowed photo collection spots, c) Photo taken preview (without AR content)

Using ARFoundation, the user is instructed on what is required of them, like where to take the photo from, camera orientation etc. Since a user can take geotagged photos either proactively or in the context of a Task, the application restricts the location of a photo accordingly. If the farmer takes initiative without having a request, he/she is only allowed to take a photo from within a radius from each one of the parcel's corners (vertices). On the other hand, if there is a Task, the farmer is only allowed to take a photo within a radius from the location that the Inspector has selected when creating the specific Task.

Due to the fact that the smartphone providers implement sensor fusion and smoothing algorithms in the determination of the geolocation and the compass readings, it is required to wait at the spot where the photo will be taken for several seconds before releasing the shutter.

The camera orientation is constrained by the tilt angles of the devices to only accept photos taken within a threshold. While in the AR session, the application continuously looks for active applications running in the background, that may tamper with the GNSS signal by mocking the actual location.

ARFoundation is used to handle all the AR session content. AR Foundation can support the following concepts among many:

- Device tracking: track the device's position and orientation in physical space.
- Plane detection: detect horizontal and vertical surfaces.
- Anchor: an arbitrary position and orientation that the device tracks.
- Light estimation: estimates for average color temperature and brightness in physical space.
- 2D image tracking: detect and track 2D images.
- 3D object tracking: detect 3D objects.
- Meshing: generate triangle meshes that correspond to the physical space.
- Raycast: queries physical surroundings for detected planes and feature points.
- Pass-through video: optimized rendering of mobile camera image onto touch screen as the background for AR content.
- Session management: manipulation of the platform-level configuration automatically when AR Features are enable or disabled.

In conjunction with this, the "AR + GPS Location" Unity asset is used to position 3D objects in real-world geographical locations via their GPS coordinates. This asset helps place all points of interest in the AR session so that they correspond to their real-world positions. The points of interests are considered to be the parcel shape vertices and conditionally the location picked by the Inspector for the Task request. These points are used to place AR anchors which are positions within the 3D world that are tracked for stability. A polygon mesh is drawn based on the parcel shape vertices to mark the actual parcel for visibility reasons.

To enhance the user experience, an AR session based on real coordinates should be as stable and accurate as possible. Various techniques have been employed towards this cause:

- The platform of choice for the development of the mobile application is the Unity game engine. Unity provides a mechanism to access location data, however this data is of low precision. This, in turn, leads to a lower position accuracy and a lower fidelity for the AR session in general. To overcome this, a method has been implemented to get the native location information directly from the Android system.

- Since the native location data can be accessed, the accuracy metric provided to calibrate the whole solution is utilised. The positional accuracy is being tracked and the digital content is drawn only when the accuracy is below a specific ceiling. This way, the spawned content is placed accurately enough to begin with. On top of this, all the AR content is re-drawn whenever the accuracy is improved, so that the overall experience is improved as well.
- The geotagged mobile application does not use the traditional camera application to take the photos and this is because of the AR session occupying the camera hardware. Hence there is no mechanism to automatically embed the EXIF metadata to the file as it usually happens. Thus, a dedicated mechanism is employed, that takes the background of the image (ie no AR content) and in the resulting PNG file a custom method is applied to decode and embed the required metadata.

Photos' gallery and data upload

The user can view the photos taken, grouped by Task to know what they already have available and also delete or upload to the backend for integrity checking the desired ones.

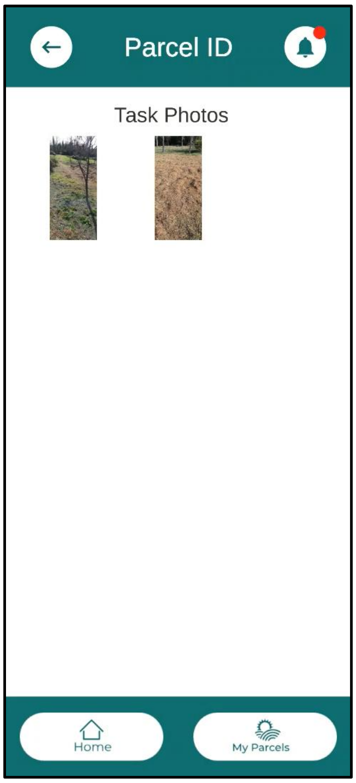


Figure 17: List of photos taken for a specific Task

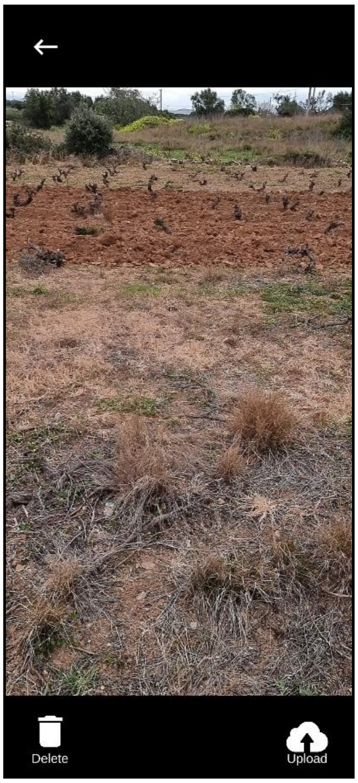


Figure 18: Existing photo preview

Offline mode

The nature of the farming activities and the geotagged mobile application's purpose mean that the most significant actions in the application's lifecycle will take place outdoors. It is not rare the fact of agricultural parcels being situated in remote and mountainous places that are not covered by mobile network signal. Therefore, it is crucial that some of the functionalities of the application can be performed when no mobile signal reception is available.

Since the data presented to the user need to be the latest ones, the first steps of the process require an internet connection to fetch the relevant data (Parcels, Tasks). This data, in turn, is temporarily stored and is available while the application is "running" subsequently offline.

Also unavailable is the access to the map view via Mapbox. The map relies on getting tile information via the internet so no useful information can be presented otherwise. However, a user can download (automatic process) some initial map tiles in the map view, disconnect from the internet, and have these initial tiles as guiding reference to the parcel in question.

The entirety of the AR session is working offline. The AR content that is superimposed is based on the initial data fetched for the user from the Toolbox API. The photos taken are stored locally in the phone's internal memory so that when a network connection is available, the user can browse through them and upload the most appropriate ones. As mentioned, the time and date of a taken photo are very important to the project as they provide a timestamp for the snapshot of the evolving crops in a parcel. The time integrity component is working offline as well. It can provide a date and time irrelevant of the phone's settings or other external providers that require network to function. The only requirement is the reception of GNSS signals which is a trivial task when outdoors.

Location Accuracy

In the context of the application alpha version, different methods have been researched and analysed in order to ensure high location accuracy including the exploitations of multiple location differentiators (i.e. dual frequency devices) and the use of EGNOS Data Access Service (EDAS) for ground-based access to EGNOS data through the Internet..

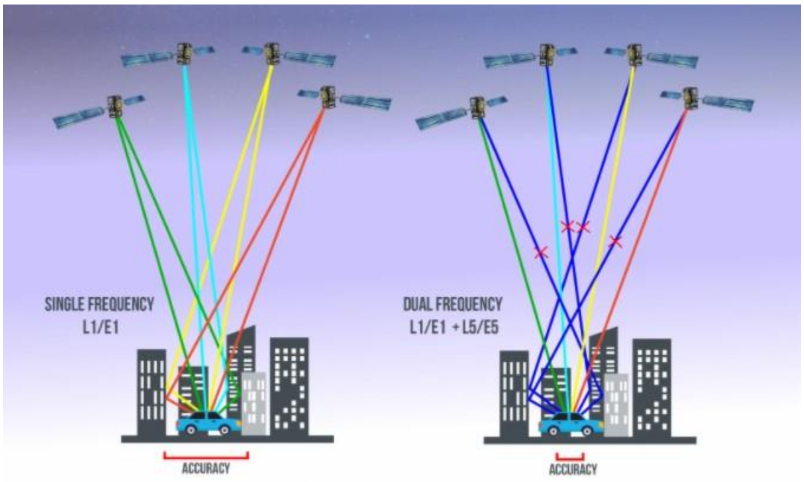


Figure 19: Dual frequency GNSS signals concept

Dual-frequency capability means that the GNSS receivers are able to receive two GNSS signals at different frequencies from a satellite. In the case of Galileo, these frequencies are E1 and E5a. Dual frequency provides increased reliability to users – if one of the frequency bands fails, the other can be used as backup.

The selection of this dual frequency combination is particularly appealing for two main reasons:

- Spectral efficiency: all the signals are with the same central frequency and bandwidth.
- Wide-band signals: most signals in the L5/E5a band are wideband with BPSK (10) modulation.

Users of dual-frequency smartphones are able to benefit from the reduced signal acquisition time and improved accuracy of positioning and timing. Dual frequency also reduces problems caused by obstructions such as buildings and other obstacles, thanks to the fact that the L5/E5a signals are lower in frequency, making them are less prone to multipath interference errors.

It is important to stress that the final positioning accuracy in mass-market devices is not only driven by GNSS measurements, either single or dual frequency. In addition to GNSS, a very important role is played by the smartphone integrated inertial sensors and additional terrestrial based signals, including for example cellular network (4G/5G), WiFi, NFC, Bluetooth, etc. All these ingredients contribute to the fused location and its ultimate accuracy.

EGNOS

The European Geostationary Navigation Overlay Service (EGNOS) provides an augmentation service to global navigation satellite systems (GNSSs), such as GPS and Galileo. EGNOS[6] provides three services:

- Open Service (OS), freely available to any user.
- Safety of Life (SoL) Service, that provides the most stringent level of signal-in-space performance to all Safety of Life user communities.
- EGNOS Data Access Service (EDAS) for users who require access to specific GNSS data streams for the provision of added-value services, professional applications, commercial products, R&D, etc.

EGNOS is a Satellite Based Augmentation System (SBAS). SBAS systems are designed to augment the navigation system constellations by broadcasting additional signals from geostationary (GEO) satellites. The basic scheme is to use a set of monitoring stations (at very well-known positions) to receive the navigation signals from core GNSS constellations that will be processed in order to obtain some estimations of these errors that are also applicable to the users (i.e. ionospheric errors, satellite position/clock errors, etc.). Once these estimations have been computed, they are transmitted in the form of "differential corrections" by means of a geostationary satellite.

The EGNOS Data Access Service (EDAS) provides ground-based access to EGNOS data, through a collection of services, which are accessible to registered users through the Internet.

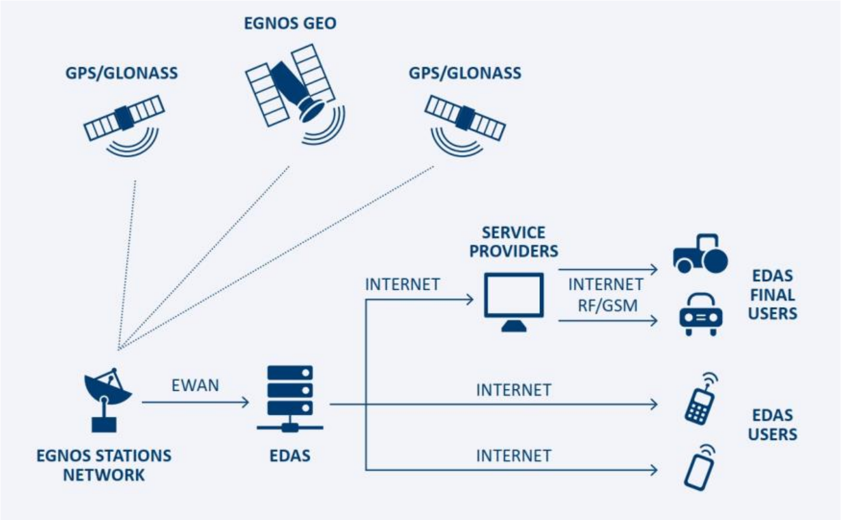


Figure 20: EDAS high level architecture

[6] https://www.gsa.europa.eu/egnos/what-egnos

More specifically, he EDAS Signal In Space (SISNeT) service provides access to the EGNOS geostationary satellites messages transmitted over the Internet through the SISNeT protocol. Doing so, the EGNOS signal is available even if geostationary satellites are not visible from the user location.

Thus, in the context of DIONE, the EGNOS messages received from the EGNOS Data Access Service (EDAS) over the internet shall be used to augment the GPS signal received. This option requires an application capable of processing and applying the EGNOS corrections (obtained from the EDAS SISNeT service) to the computed position solution. This approach will also cover the mobile devices built with chipsets and firmware able to access GPS raw data but are not SBAS-compatible (not being able to access and process EGNOS raw due to their operative system/firmware).

Registration to EDAS and access to the EGNOS messages has been already provided, whilst the implementation will be included in the next version of the geotagged photos mobile application.

3.1.3 Toolbox services

In order to leverage the system integration with the Paying Agencies infrastructures several Toolbox API endpoints (https://api.dione.gilab.rs/api/) are used in the context of the geotagged mobile application.

Table 2: Toolbox endpoints used by the geotagged mobile application

Endpoint	Description	Comment
POST /authentication/login/	Login function	Ask to start a DIONE session with user credentials
POST /authentication/logout/	Logout function	Terminate user session
POST /authentication/token-refresh/	Refresh token	Returns a refreshed token (with new expiration) based on existing token
GET /parcel/	Fetch parcels	Fetch all parcel information related to the user in order to populate the related page and visualise AR content
GET /photo/	Fetch photos	Fetch all photo history related to the user in order to populate the related page
GET /task/	Fetch tasks	Fetch all task history related to the user in order to populate the application pages, trigger actions and visualise AR content
PATCH /task/	Modify task	When a user reads (dismisses) a notification, the application updates the backend with this information so as not to present it to the user in the future

3.2 Data integrity, validation and anonymization

3.2.1 Outline of integrity framework

In order to ensure that the file of the photo captured by the mobile application is not manipulated and is the same with respect to the location and time the photo was taken, an integrity framework has been developed. Different algorithms have been implemented for a multifaceted solution and better security. The framework consists of two parts, i) a mobile module which is an android library responsible for running some initial processing and ii) a backend module that validates the origin of the photo and on a successful validation creates a blurred image keeping potential personal information that may exist in the data private.

The chosen language for the implementation of the mobile library is Java. Java is one of the two languages with which native android applications are developed. Moreover, being one the most used programming languages provided a variety of possible APIs and modules that can be exploited in the context of the library. One equally used language is Python. By taking into consideration the good support for image processing and the security libraries available, it was considered as the right choice for the backend part of the framework.

3.2.2 Integrity algorithms implemented

3.2.2.1 Lightweight digital signature scheme

The mobile library is triggered in different times while using the application. The first time is during the authentication of the farmer, when a unique pair of a private and a public key is generated[7]. The public key is communicated and stored in the database, being bonded with a specific farmer. However, the private key never leaves the smartphone and is used only by the application. When a farmer takes a photo, the digital signing takes place, using the private key, and a signature is generated. The signature is produced and sent to the API (Central Database, see section 4.1 & 4.4) along with the photo. This signature cannot be verified even if a bit is changed in the photo. As a result, during the verification in the backend, it is ensured that the original photo has been sent.

For the generation of the keys and the digital signing in the mobile library, the Security library available in Java is utilised. The corresponding library in the backend framework is the Crypto Python library. For the DIONE scheme, the RSA with a SHA256 algorithm is chosen due to the smaller key sizes and the faster computations compared to the elliptic curves. It is crucial to mention here that the steganography technique and the metadata embedding (see sections below) are conducted before the digital signature takes place because otherwise the signature verification would fail.

3.2.2.2 Steganography

Another mechanism that it is used to ensure the authenticity of the photo is the steganography technique. The steganography technique uses the least significant bits of every pixel, in order to embed a secret message in the photo that can only be extracted by using the reverse algorithm, and is not identifiable by common tools. As only the least significant bit of a pixel is used there is no visual alteration in the photo.

For the implementation of the steganography technique a java algorithm has been implemented. For the extraction zsteg[8] was used, which is a bash command tool implemented in Ruby. The integration

[7] https://ieeexplore.ieee.org/document/507642

[8] https://github.com/zed-0xff/zsteg

of this bash tool with the framework was straightforward using the OS Python module. A successful extraction of the message would mean that the file is not tampered.

3.2.2.3 Photo metadata

A number of information, like GPS, date and focal length, which are checked for their integrity (see next paragraphs) are collected during the usage of the mobile application. This information is embedded in the photo in the form of metadata[9]. Later the metadata are extracted in the backend and cross checked with the farmer's declared data stored in the database.

For the embedding, the pngj[10] library and for the extraction the imagemagick[11] suite are utilised. The integration with the framework is identical to zsteg tool described above.

3.2.2.4 Copy Move Forgery Detection

One popular digital modification technique is the copy move forgery. The copy-move forgery is the process of copying and pasting from one region to another location withing the same photo. An algorithm has been implemented which is able to detect such cases. The algorithm is based on principal of component analysis, but sensitive to noise and post region duplication process[12].

This is pure python implementation, using Pillow module for the photo processing.

3.2.2.5 Error level analysis

Every photo file is compressed using a specific algorithm (i.e. JPEG etc.). Taking this into consideration, the entire photo should be roughly at the same level, if a difference is detected, then it likely indicates a digital modification. This technique is called Error level analysis.

For this technique a pure python script was implemented. The result of the algorithm is photo with the parts of the photo which are at different compression levels in brighter color. Different parts of the photo could be in different compression levels, but if these parts form a meaningful figure (ie. A face or a tree), this implies that the photo is modified.

However, the automation of this process didn't yield satisfactory results, and thus this algorithm wasn't included eventually in the geotagged photos integrity framework.

3.2.2.6 Camera identification

To further ensure the origin of the photo, each user is paired with his mobile device. This is achieved with a module which calculates the pattern noise and identifies the camera used to take the photo. The identification becomes possible due to the fact that each component in a digital camera leaves intrinsic fingerprints in the final image output, which due to manufacturing choices are unique for each device. The component of the camera that makes possible the identification is the complementary metal oxide semiconductor (CMOS) image sensor, which inserts a pattern noise in the photos which is unique for each device.

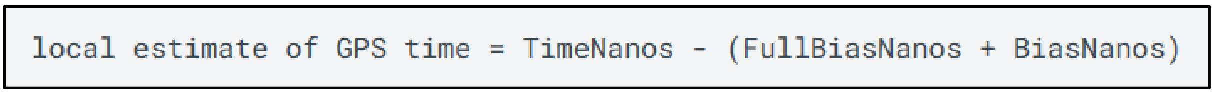
The method used for the development of this module is presented in the "Estimation of Gaussian, Poissonian–Gaussian, and Processed Visual Noise and Its Level Function"[13]. However, due to the fact photos are collected through the mobile application AR features, the characteristics of the camera

[9] https://en.wikipedia.org/wiki/Exif
[10] https://github.com/leonbloy/pngj
[11] https://imagemagick.org
[12] https://www.semanticscholar.org/paper/Exposing-Digital-Forgeries-by-Detecting-Duplicated-Popescu-Farid/b888c1b19014fe5663fd47703edbcb1d6e4124ab
[13] https://ieeexplore.ieee.org/document/7506318

cannot be imprinted in the photos and thus the pattern noise cannot be computed properly. To encounter this, the focal length is read which constitutes a fundamental characteristic of the camera. After the photo is taken, using the Android API, the focal length is extracted and embedded in the photo so as to cross check it later in the backend.

3.2.2.7 Time Integrity

The time integrity module uses the Android API and based on the equation below, manages to recreate a near-precise clock independent of network access. By doing this, a user cannot fool the DIONE platform and all the photos taken are tagged with their true date and time. The date and time issue are sensitive in the context of in-situ checks since the evidence data is only relevant at the time period that they are requested.



```
local estimate of GPS time = TimeNanos - (FullBiasNanos + BiasNanos)
```

3.2.2.8 Location integrity

One critical piece of information that has to be verified is the location from which the photos are being captured.

For this purpose, a module has been developed, being capable to detect any external process/application that attempts to alter the position information/GPS of the mobile device. In such cases, the data collection is not permitted.

Furthermore, in order to enhance the location integrity verification, a dedicated algorithm has been developed allowing the exploitation of the open service navigation message authentication scheme (OSNMA). OSNMA allows a GNSS receiver to verify the authenticity of the GNSS information and of the entity transmitting it, and ensures that it comes from a trusted source. By exploiting the OSNMA, the geotagged photos integrity framework shall be able to ensure that navigation message received is identical to the message transmitted and that was generated by a trusted source.

Regarding the implementation, the C++ programming language was used due to the faster computation times. For the cryptographic primitives the CryptoPP libarry[14] is utilised. This application will be integrated as a library in the next version of the DIONE geotagged photos application using the Java Native Interface (JNI) of the Android platform.

3.2.3 Anonymization component

The last component of the framework aims to address privacy considerations. Personal information that may exist in the collected photos should not be communicated to other users of the data (i.e. PA Inspector) in order to comply with the provisions of the General Data Protection Regulation (GDPR). Thus, an anonymisation component has been implemented, which is responsible for blurring any faces or license plates exposed in the photos.

The anonymization component is basically a pre-trained convolutional neural network. For the training of the network various core libraries have been used such as tensorflow, numpy, scipy and Pillow.

[14] https://www.cryptopp.com/

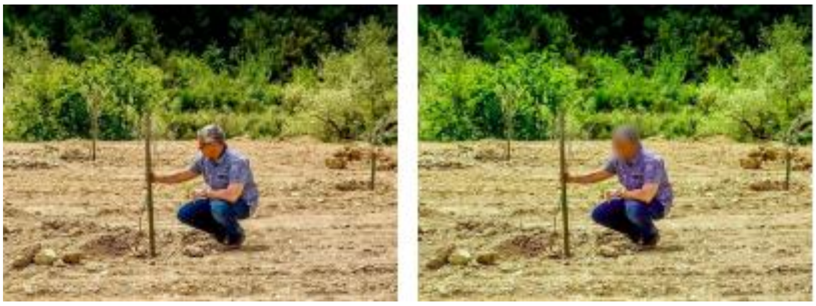


Figure 21: Example of anonymized photo

The overall process (both mobile and backend) is depicted in the next photo.

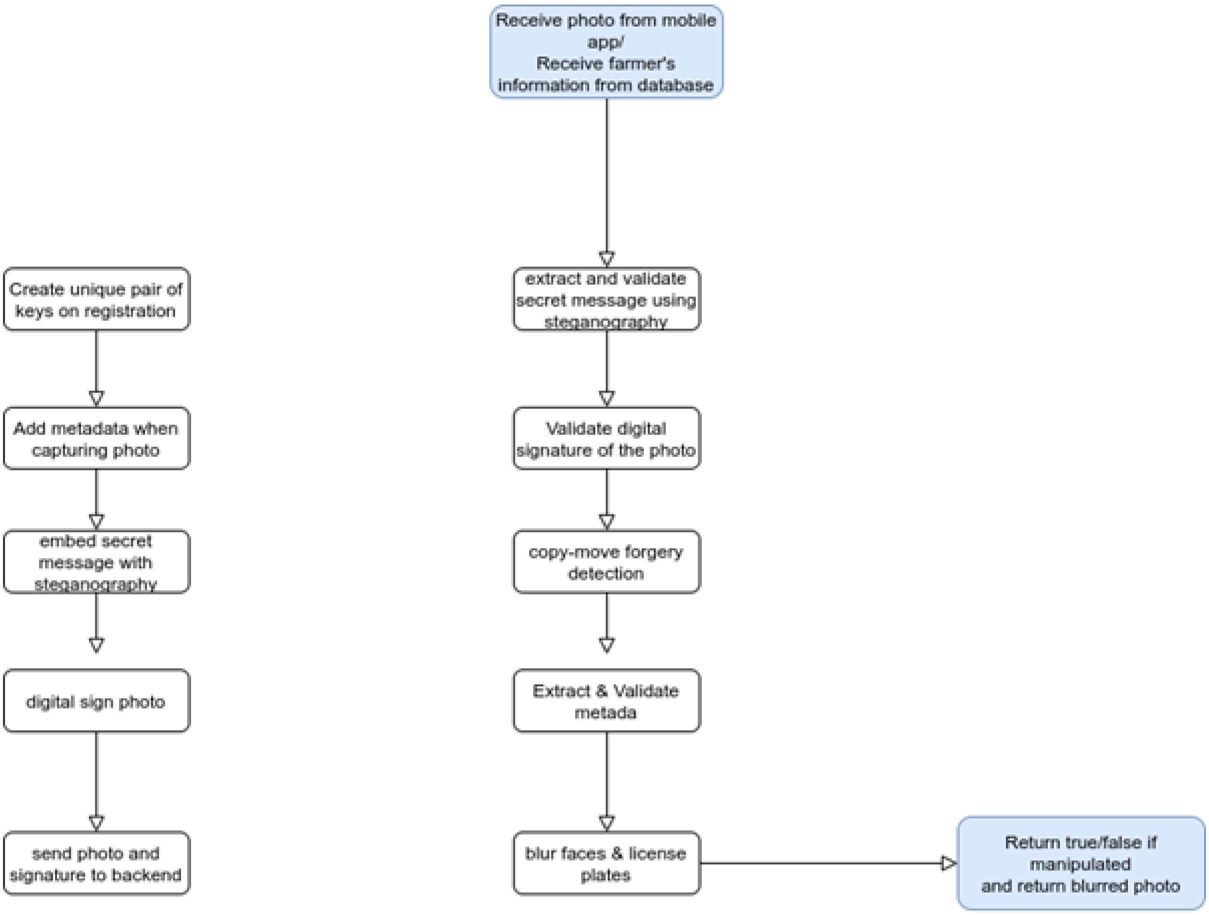

Receive photo from mobile app/ Receive farmer's information from database
Create unique pair of keys on registration
Add metadata when capturing photo
embed secret message with steganography
digital sign photo
send photo and signature to backend
extract and validate secret message using steganography
Validate digital signature of the photo
copy-move forgery detection
Extract & Validate metada
blur faces & license plates
Return true/false if manipulated and return blurred photo

Figure 22: Schematic overview of the mobile and backend process of the geotagged photos integrity, validation and anonymization framework

One critical consideration of the development phase was the smooth communication between the application and the forensics library. This task was successfully done, due to the AndroidJavaClass class in the Unity game engine, which is basically a wrapper for an android class.

3.3 Data transmission

The photo related data is stored in the Android device in a .json file as follows:

Key	Value
userId	Unique identifier of the user
uuid	Unique identifier of the image
parcel	Unique identifier of the Parcel the photo is taken for
task	Unique identifier of the Task the photo is taken for
date_time	Date and time directly decoded from GNSS signals
coordinates	Coordinates of the photo location from native Android
bearing_angle	Compass reading from the device sensor at the time of the photo
image	The photo itself in base64 PNG format
blurred_image*	The resulting image after the integrity checks and the blurring applied
validated_image*	Boolean flag
signature	The hash of the file used for integrity
publicKey	Public key for signature check
focalLength	Device camera focal length
status	Image status (uploaded, not uploaded, dismissed by integrity)

3.4 Compliance of alpha version with technical specifications

The table lists an overview of the compliance of the GEO in the alpha version with the technical specifications.

Table 3: Compliance of alpha version of the geotagged photos framework with technical specifications

Component	Requirement ID	Short description	Included in alpha version
GEO	SYS_F_GEO_1	Access to registered farmers	Yes
GEO	SYS_F_GEO_2	EGNSS systems for positional accuracy and location integrity	Partially (OS-NMA implementation)
GEO	SYS_F_GEO_3	Positional metadata extracted from raw GNSS data	No
GEO	SYS_F_GEO_4	Current position on map and parcel information	Yes
GEO	SYS_F_GEO_5	User navigation	Yes
GEO	SYS_F_GEO_6	User tutorial	No
GEO	SYS_F_GEO_7	Geotagged photos capturing restrictions	Yes

GEO	SYS_F_GEO_8	Geotagged photos must be sent and saved to a database	Yes
GEO	SYS_F_GEO_9	Offline photo capture	Yes
GEO	SYS_F_GEO_10	Local language application menus	No
GEO	SYS_F_GEO_11	More detailed parcel information	Yes
GEO	SYS_NF_GEO_2	Application must achieve the best possible positional accuracy	Partially (use of native Android location in place of Unity)
GEO	SYS_F_SEC_2	Geotagged photos timestamp integrity	Yes

4 Data processing and storage system

The goal of this section is to present the central data processing and storage system for the two mobile applications presented in Sections 2 and 3, namely the in-situ soil scanning system and the farmers' geotagged photos framework. The high-level overview of this subcomponent is presented below and in Figure 8.

In essence, the data collected by those two mobile applications are transmitted Over the Air (OTA) to a server that hosts a) the database and b) the RESTHeart service required to expose this database to the public. Moreover, the server further contains automated hooks which are executed on specific instances (e.g., when a new datum is received) to check the integrity and validity of the data, ensuring a) the integrity of the data is not comprised during the transmission, b) that the data are error-free and have not been tampered with by external actors, c) the data are sound (contain no outliers or are of the expected input). Should the aforementioned conditions be met, then further data-specific processing pipelines are also automatically triggered; these entail for example the estimation of soil properties from the spectral data and the automatic posting of the geo-tagged photos to the DIONE toolbox.

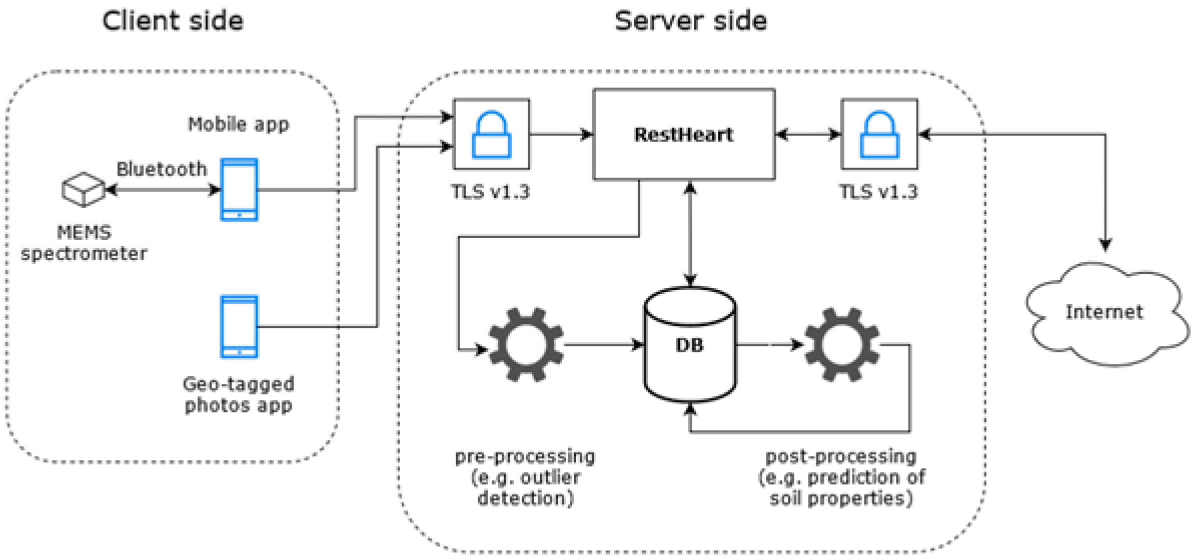


Figure 23: A simplified view of the data processing and storage system

The rest of the section is structured as follows:

- Section 4.1 describes the implementation of the database management system including the deployment steps and development of RESTHeart plugins.
- Section 4.2 lays out the mechanisms used to ensure data integrity, the data pre-processing, and the outlier and novelty detection mechanisms.
- Section 4.3presents the way data are post-processed.
- Section 4.4presents the current API as deployed for the alpha version.
- Finally, Section 4.5 lists the compliance of the alpha version with the technical specifications.

4.1 Database management system

At a higher level, the database management system deployed uses:

- The MongoDB document-oriented database, which stores data records as BSON documents. BSON is a binary representation of JSON documents, though it contains more data types than JSON. It can thus handle the heterogeneous data that are required within the scope of the DIONE project.
- RESTHeart which connects to the MongoDB instance and exposes the data using a RESTful interface over TLS.
- The TLS components to connect to the outside world securely.

4.1.1 Deployment and hosting

The software developed may be used to deploy a RESTHeart instance and DB for the DIONE project at any server. A development server was first deployed at i-BEC's premises and a production server was then deployed at ICCS's premise. The production virtual machine has the following specifications:

- 2 x CPU cores
- 16 GB of RAM
- 1TB of HDD space
- Ubuntu 20.04.02 LTS operating system

For security reasons the ufw (uncomplicated firewall) software is used, and access is only allowed from ICCS's internal network and from the VPN. The only ports open are the 80 and 443 (standard web traffic via http and https). Port 80 redirects to port 443 (i.e. http redirects by default to http over TLS or https), while port 443 is forwarded using an apache webserver to the internal 8082 port used by RESTHeart.

The production server may be reached at: **https://dione.iccs.gr/**

4.1.2 Security overview of RESTHeart

When a new request reaches the RESTHeart interface, the first step is to ensure that the request is authenticated and authorized (Figure 24). The Authenticator verifies the credentials extracted from the request by Authentication Mechanism. The authentications mechanisms used currently are a Google Token authenticator, the token authenticator from the DIONE toolbox (for single sign on purposes), and Basic Auth (username / password combination). The Authorizer is responsible of checking if the user can actually perform the request against an Access Control List. For example, if the authenticated user is indeed authorized to perform the requested action on a specific collection.

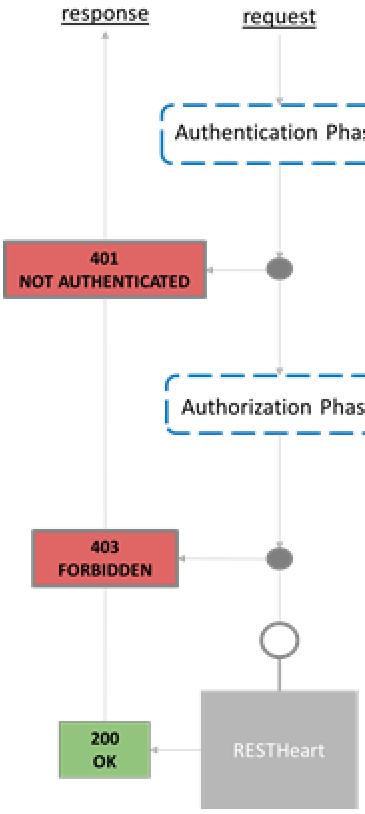


Figure 24: Requests in RESTHeart should be both authenticated and authorized.

This authentication and authorization scheme may work for both input (e.g., POST, PUT or PATCH) and output (e.g., GET) requests. The following restrictions apply:

- Users authenticated via a Google token or a DIONE toolbox token are allowed to POST, PUT and PATCH documents in the respective collections for DIONE. Currently, users authenticated via a Google token are authorized to have access to the collection about the soil scanning system, while users authenticated via the toolbox API are authorized to have access to the collection about the geotagged photos.
- A specific user instance has been created who is the only one (excluding the administrator) that can be authenticated via the basic authentication mechanism (using a username / password combination). This user has only GET permissions on the collections generated for DIONE.

4.1.3 RESTHeart extensions

RESTHeart follows a pluggable architecture and programmers can extend it by implementing custom extensions, such as Application Logic Handlers and Request Hooks. As RESTHeart is a Java application, to package custom Maven or any Maven-compatible tools, such as Gradle, ought to be used.

Within the frame of DIONE, the following extensions have been implemented in the Java language:

Google Authenticator (DEPRECATED)

In the alpha version of the software, the soil scanning system uses google authentication[15] [16]to securely authenticate the users to submit their data through the RESTHeart interface. That is, the Android application uses the google account to which the given smartphone is connected to; it is noted that a pre-requisite in Android phones is to be connected using a google account. The whole flow is presented in Figure 25. The user first authorizes the application to access the e-mail of the google account. Subsequently, the tokens received by Google's API server are sent to RESTHeart (along with the rest of the payload). The Google authenticator plugin then confirms the validity of the token with Google's API server; if the token is valid the request is authenticated, otherwise the request is denied, and the server replies with a standard HTTP 401 message (Unauthorized).

It is noted that after the first review of the project and in direct consultation with the other project partners it was decided that the single sign-on mechanism incorporated in the DIONE toolbox will be used by all components. Thus, this plugin will be deprecated and replaced by the authentication mechanism for the DIONE toolbox.

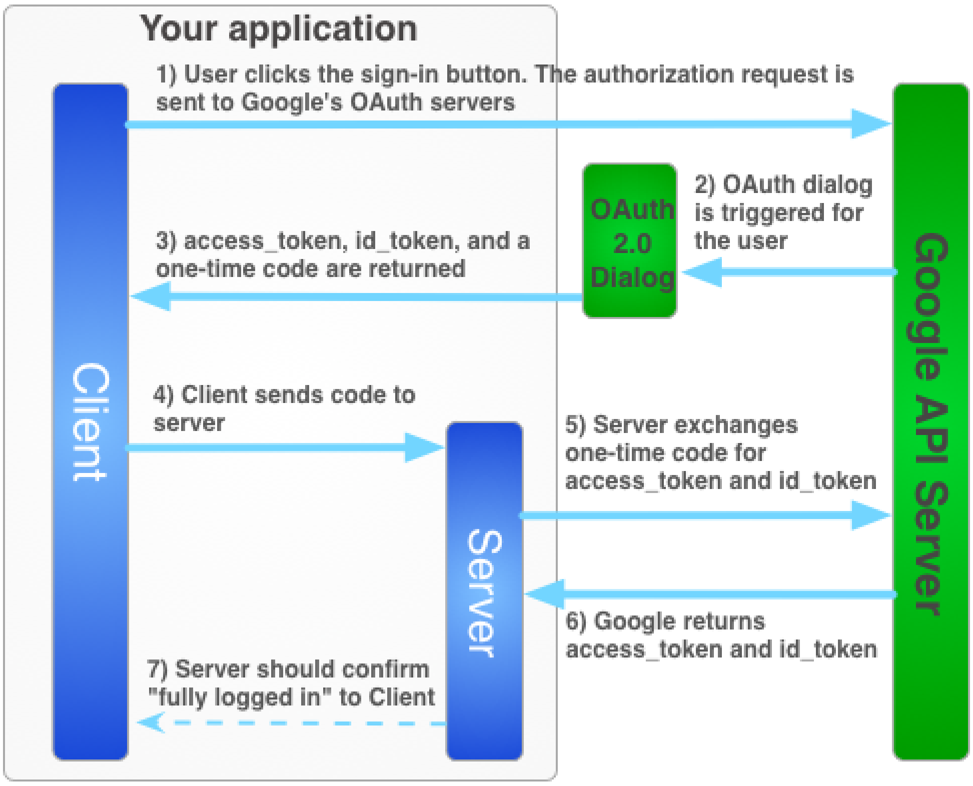

Your application
1) User clicks the sign-in button. The authorization request is sent to Google's OAuth servers
OAuth 2.0 Dialog
2) OAuth dialog is triggered for the user
3) access_token, id_token, and a one-time code are returned
Google API Server
Client
4) Client sends code to server
5) Server exchanges one-time code for access_token and id_token
Server
6) Google returns access_token and id_token
7) Server should confirm "fully logged in" to Client

Figure 25: The Google sign-in flow for obtaining an access token for the server-side application. Image taken from https://developers.google.com/identity/sign-in/web/server-side-flow

DIONE toolbox authenticator (Single Sign On)

In the alpha version of the software, the geotagged photos integrity system authenticates the incoming requests by leveraging the DIONE platform authorisation system of the Toolbox API. All users of the geotagged photos mobile application are registered in the DIONE platform since it serves centrally the Paying Agencies' data.

[15] https://developers.google.com/identity/sign-in/android/start-integrating

[16] https://developers.google.com/identity/sign-in/web/server-side-flow

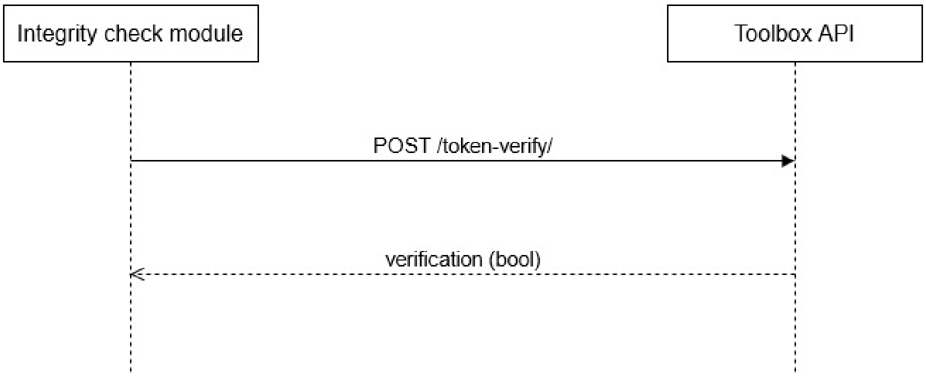


Figure 26: Authentication scheme for integrity requests

Requests towards the /geotag/ collection require the JWT provided by the Toolbox API to be included in the Authorization query parameters. This token, is subsequently queried for validity using the /token-verify/ endpoint of Toolbox API. Based on the response, we process the geotag request or not. User authenticated through the DIONE platform are allowed to POST documents to the respective collection.

Spectra Interceptor

The goal of this plugin is to execute the complete pipeline for a given document that contains a single measurement from the soil scanning system. Mainly, to i) transform the transmitted raw sensor output to standardized reflectance, ii) to detect potential outliers or novel signatures, and iii) to predict the soil properties for the clean and validated soil spectral signatures. For reasons described in the following section, the implementation of this interceptor has focused only on the software architecture and currently performs simplified actions instead of the more elaborate ones that will be delivered in the final version.

Integrity Interceptor

This plugin is responsible for executing the integrity framework for each photo that is uploaded. The goal of our interceptor is to pass the photo through all the integrity algorithms described in the corresponding section. The initial request to the server is intercepted and on a successful validation of the photo, a new item is appended to the request. This item is a base64 encoded image which corresponds to the blurred image produced by the anonymization component. Also, a field which contains a Boolean indicating the result of the framework, namely if the photo is original.

4.2 Data preprocessing, outlier and novelty detection techniques

Data integrity refers to the accuracy and consistency of data over its entire lifecycle, as well as compliance with necessary constraints and other security measures. In effect, it is the trustworthiness of the data; high data integrity means data has not been altered, corrupted or misused in any form. Achieving data integrity requires identifying the potential risks during data transfer and storage, which involves the risk of corruption (either voluntarily or involuntarily) during that process.

If data integrity can broadly be summed up by the trustworthiness of the data, then data quality pertains to its analytic value. Data quality refers to the characteristics that determine the reliability of information to serve an intended purpose including planning, decision making and operations. It is the state of complete features and attributes that define the usability of information to address specific needs in context of real-world circumstances and implications.

Within the scope of the DIONE project, different checks were focused on tackling issues which may arise:

- Through human error, whether malicious or unintentional, e.g., by purposefully tampering the geotagged photos, erroneous usage of the spectrometer, and more;
- Due to transfer errors, including unintended alterations or data compromise during transfer from the mobile apps to the database;
- Through compromised hardware, such as a device or disk crash.

4.2.1 Data integrity

Following the technical specifications, the data integrity was focused on three aspects:

- Error detection / data validation to overcome errors in data transmission;
- Integrity check of the transmitted documents (to ensure conformity) with a JSON Schema Validator;
- Data backup and duplication to prevent data loss prevention.

Error detection in data transmission

To overcome potential transfer errors when data from the mobile applications are transmitted to the server, the TLS protocol was used. TLS stands for Transport Layer Security and is the successor to SSL (Secure Sockets Layer). TLS provides secure communication between web browsers and servers. The connection itself is secure because symmetric cryptography is used to encrypt the data transmitted. The keys are uniquely generated for each connection and are based on a shared secret negotiated at the beginning of the session, also known as a TLS handshake. The newest version is TLS 1.3, which is a faster and more secure version of TLS (compared to previous ones). By default, the connection is reliable because each message transmitted includes a message integrity check using a message authentication code to prevent undetected loss or alteration of the data during transmission.

The DBMS is exposed to the internet via the https://dione.iccs.gr/ API endpoint. The SSL/TLS report for this endpoint is given in the Appendix C.

JSON Schema Validation

JSON Schema specifies a JSON-based format to define the structure of JSON data for validation, documentation, and interaction control. A JSON Schema provides a contract for the JSON data required by a given application, and how that data can be modified.

The JSON schemas will be finalized in the next months and their final version will be presented in D4.4. An initial version for the mems documents has been added and is integrated into the alpha version.

Data backup and duplication

The DBMS is hosted on a Linux VM (see section 4.2.1) which uses the ZFS file system that in effect combines both the file system and a volume manager. One major feature that distinguishes ZFS from other file systems is that it is designed with a focus on data integrity by protecting the user's data on disk against silent data corruption caused by data degradation, power surges (voltage spikes), bugs in disk firmware, phantom writes (the previous write did not make it to disk), misdirected reads/writes (the disk accesses the wrong block), DMA parity errors between the array and server memory or from the driver (since the checksum validates data inside the array), driver errors (data winds up in the wrong buffer inside the kernel), accidental overwrites (such as swapping to a live file system), etc. Moreover, instead of hardware RAID, ZFS employs "soft" RAID, offering RAID-Z (parity based like RAID 5 and similar) and disk mirroring (similar to RAID 1).

In terms of data backup, in the final version of this component an automated process will ensure that the data are regularly duplicated to a separate physical location over the Internet. The separate location for the MEMS spectral data will be the servers of i-BEC, while for the geotagged photos it will be an ICCS server.

Processing of geo-tagged photos

A complete list of further actions that are undertaken to further ensure the integrity of the geotagged photos is presented in Section 3.2.

4.2.2 Sensor output to standardized reflectance

The sensor data from the MEMS spectrometer are uncalibrated by the Spectral Engines company that created the spectrometer; they instead provide only the digital output of the sensor. Essentially, the sensor turns the incoming energy into a voltage, which an analog to digital converter turns into a single value (called the Digital Number, or DN) for the energy. This value however does not correspond to a particular S.I. unit of measurement and is different depending on the sensor (both the type of spectrometer and the physical instance) used. To have standardized values it is necessary to perform a standardization step; this ensures that the measurements are compatible among different spectrometers of the same type and company and with other commercial spectrometers (i.e., inter and intra standardization).

This step will be carefully evaluated in the next two months (i.e., May and June of 2021) in dedicated field visits in order to ensure its efficacy. It is of paramount importance to ensure it works properly so that further processing works as intended. Therefore, in the present deliverable which describes the alpha version of the software there is no delivery of a standardization module.

4.2.3 Identification of outliers

Pending the finalization of the transformation from raw sensor output to standardized reflectance, no outlying detection mechanism is included in the alpha version. After this calibration step is finished, a new dataset will be generated which will entail spectral signatures from various materials (i.e. not limited to soil). A binary classification model will then be developed to distinguish between spectra from soil and spectra from non-soil materials.

4.2.4 Novelty detection

Pending the finalization of the transformation from raw sensor output to standardized reflectance, no outlying detection mechanism is included in the alpha version. After this calibration step is finished and following the development of the models from the data of the soil spectrometer, a new model will be developed which will identify whether the recorded spectra are novel.

4.3 Data processing

The spectrometer data that have been cleaned and validated are automatically processed by the model developed from the machine learning algorithms, so that the spectral data are finally transformed into soil properties. Pending the finalization of this step, the development shifted focus on implementing the proper software architecture that will enable an easy update and usage of the model. That is, currently the spectral data received are assumed to be clean and validated and a first simple version of the model is used.

4.3.1 Software interface and containerization

This interface is part of the Spectra Interceptor extension. As far as the software architecture is concerned the following was implemented: First, it was noted that both the integrity checks of the geo-tagged photos and the models pertaining to the processing of the spectral data (outlier detection, novelty detection, and prediction of soil properties) are implemented in the Python language. As noted above, RESTHeart is written in Java. To enable effortless updates, a wrapper functionality was used to interface between the two. In simple terms, the RESTHeart plugin written in Java calls an abstracted implementation in python of each service and parses its formatted output. For example, in the case of the model prediction the abstracted implementation in python expects as input the spectral signatures and outputs the soil properties. From the other side of the software, the Java applications handles the interface with RESTHeart and dictates when the model prediction should take place (i.e., when a new POST request is processed). By building this interface it is possible entirely decouple the abstraction from its implementation and allows both to vary independently.

Moreover, an additional goal was set to have containerized applications that can be executed independently from one another. The benefits of this approach are in line with the technical requirements (and particularly SYS_F_SOIL_2) and further include:

- Portability and reproducibility between different platforms
- Improved security by isolating applications from each other
- Projects are self-contained by capturing all dependencies in a single requirements file

This was implemented through the usage of python environments. A Python environment is a dedicated directory where specific dependencies can be stored and maintained. Environments have unique names and can be activated when needed, allowing for control over the libraries that are installed at any given time. Therefore, different environments may have different versions of specific libraries installed without conflicts. The environments are installed and managed via conda, an open-source package manager that:

- Can quickly install, run, and update packages and their dependencies.
- Can create, save, load, and switch between project specific software environments.

Thus, in a nutshell each specific data processing application is standalone, portable and easily updateable thanks to the abstraction layer that interfaces the java and the python code. The software written in Java handles the logical processing of the query and the lifetime cycle of the python software. In turn, the python code tackles the problem itself (e.g. prediction of soil property from a soil spectrum).

Finally, it should be noted that with respect to the geotagged photos, the clean and validated data are automatically transmitted to the DIONE's toolbox API for further storage and integration with the services provided therein (see section 3.1.3).

4.3.2 Model for predicting soil properties

In the alpha version of the software, from the data processing modules pertaining to the soil spectra, namely the outlier and novelty detection and soil property detection, only the latter is delivered in a preliminary format. With respect to the database management system, the software development is complete, and pending is only the finalization of the standardization protocol which will consequently lead to the update of the underlying model.

The initial model was established from the (unstandardized) spectra recorded from soil samples during the testing phase of the soil scanning system (see Section 2.2). The training dataset was comprised from 102 soil samples belonging to the open eastern Mediterranean Soil Spectral Library that was co-developed by i-BEC. As far as the technical aspect of the model is concerned, the machine learning algorithm used was a Partial Least Squares Regression model that predicts the soil organic matter content from the reflectance spectra. This was used as a simple and baseline model, with future expansions focusing on the use of more sophisticate learning algorithms and data processing toolchains. The model is saved in a pickle format and the abstraction interface works as follows:

1. It reads as command line input the reflectance spectrum.
2. It loads the stored ML model in pickle format.
3. It predicts the soil property given the reflectance spectrum.
4. It outputs the soil property.

4.3.3 Processing of geo-tagged photos

Similarly, using the exact same software interface architecture, the processing of the geotagged photos is automatically triggered when a new document is posted to the /geotag endpoint. Based on the outcome of these checks, one of two things may happen. In case of successful pass, the photo is posted to the DIONE Spatial Data Infrastructure via the Toolbox API (https://api.dione.gilab.rs/api/photo/) for final storage and to become available for inspection. In the event of failure, the backend triggers a push notification towards the mobile device to let the application know of the failed result so that the user takes correction measures either by uploading other photos they have taken or by going again to the field to take new photos. The details of the processing steps for the geotagged photos are given in Section 3.2.

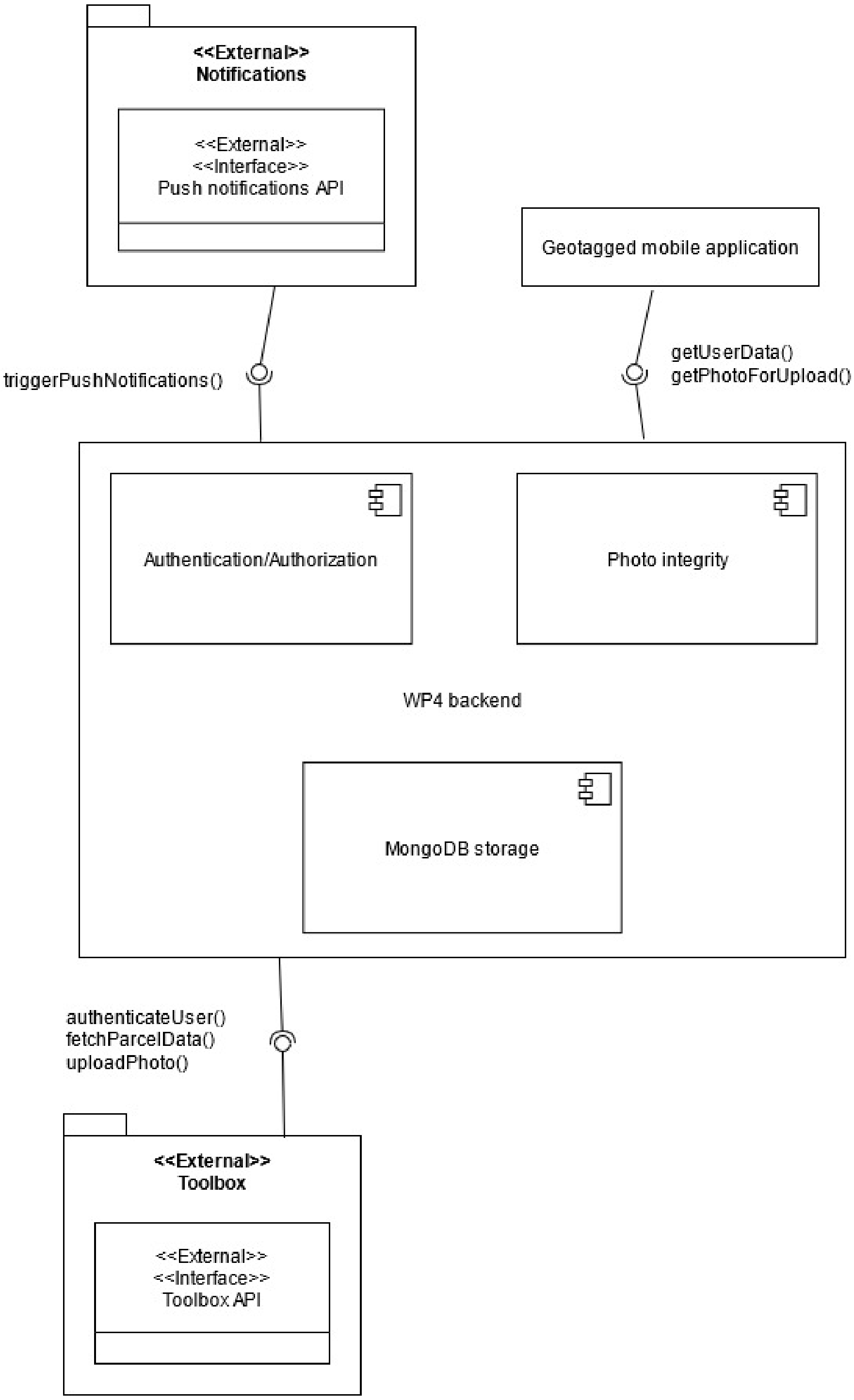


Figure 27: Geotagged photos backend component connections

4.4 System API

All the API endpoints of RESTHeart are available out of the box for the administrator of the system. A full reference may be found in the official docs[17]. In this section the API endpoints of the alpha version

[17] https://restheart.org/docs/

are presented that are available only to outside actors and meaningful as parts of the external interface. Table 4 lists the currently operation API endpoints from external actors.

With respect to the proposed API endpoints in D4.1 the only change is the name of the collections: instead of spectra in the alpha version they are called mems, and instead of photos the collection is called geotag. The change is at this stage deliberate and has been performed to enable a quick switch between a development environment and a production environment.

Table 4: API endpoints for the database

Endpoint	Description	Comment
POST /mems	Submit new in-situ soil spectral measurement	Enables smartphones to transmit data to the DB. ObjectId is autogenerated. Outside authenticated and authorized users use this API endpoint to push new data.
GET /mems	Browse spectral measurements	Supports counting, paging, filtering, sorting and projection[2]. User dione may use this API endpoint to get the spectral data.
POST /geotag	Submit new geo-tagged image in the image management component	Ability to catalogue geo-tagged images. Outside authenticated and authorized users use this API endpoint to push new data.
POST /geotag_user_data	Submit fresh user data	Centrally store regularly used data
GET /geotag	Browse a list of possible geo-tagged photos	Supports counting, paging, filtering, sorting and projection. User dione may use this API endpoint to get the geotagged data.
POST/notification	Submit or edit a notification for geotagged image retrieval by a farmer	Farmers must be notified about the need of geotagged photos for a given parcel from the DIONE mobile app.

4.5 Compliance of alpha version with technical specifications

Table 5 lists an overview of the compliance of the DBMS in the alpha version with the technical specifications.

Table 5: Compliance with technical specifications

Component	Requirement ID	Short description	Included in alpha version
DBMS	SYS_NF_INTEG_1	Server online with min. technical requirements	Yes
DBMS	SYS_NF_INTEG_2	Secure data transmission	Yes
DBMS	SYS_NF_INTEG_3	Data redundancy and automated back-ups	Partially (offline back-ups not implemented)
DBMS	SYS_F_INTEG_1	Ensure incoming data integrity	Yes
DBMS	SYS_F_INTEG_2	Outlier and novelty detection	No (pending finalization of the standardization step)
DBMS	SYS_F_INTEG_3	Storage of clean & validated values	Partially
DBMS	SYS_F_INTEG_4	RESTful API	Yes
DBMS	SYS_F_INTEG_5	Handle heterogeneous data	Yes
DBMS	SYS_F_INTEG_6	Data authorization	Yes
DBMS	SYS_F_SEC_1	TLS/SSL	Yes
DBMS	SYS_F_SOIL_1	Call ML model automatically	Yes
DBMS	SYS_F_SOIL_2	Updateable ML model	Yes
DBMS	SYS_F_SOIL_3	Ingest and serve maps	No (part of D4.4)
DBMS	SYS_F_SOIL_4	ML models for soil maps	No (part of D4.4)

5 Next steps and future outlook

5.1 In-situ soil scanning system

Further features that will make the application friendlier to the end user will be included to the final release. As mentioned before, the help section is unpopulated and will contain a detailed guide and a video that will facilitate users on how to successfully perform a soil scan. This guide will be textually delivered in English, Greek and Lithuanian, and will include images for easier understanding. Added to that, a video walkthrough will be also included. Of course, it should be noted that within the context of the DIONE project a hands-on training session will be scheduled (either virtually due to COVID-19 restrictions or with a physical presence).

The interface will also be reformed, adopting a high contrast palette designed for field use and to be aligned with DIONE's colour palette. On top of that, since the Soil Scanning System will be a component of the broader DIONE's toolbox, the application will adopt the single sign-on authentication method that will be used from the rest of DIONE's components. A new feature that will help capturing the moisture class will be also included, where the user will be asked to assess the topsoil moisture to a coarse classification in three easily distinguishable classes of "Dry", "Normal" and "Wet" and one class of measurements that will be rejected due to high moisture presence and labeled as "Extremely Wet".

Finally, the Soil Scanning System will be tested in operational environment and the standardisation protocol will be finalised too.

5.2 Geotagged photos

The application user interface will be further assessed by end users, prior to the conduction of the pilot demonstrations in each country and relevant adaptations will be realised according to the feedback received.

Additionally, more guidance will be needed for the application user, especially anyone unfamiliar with such technologies. Hence, a screen will be added with brief instructions on what is required of them, how to take a photo, the restrictions applied etc. An instructional video will be also added.

Since AR systems rely heavily on location accuracy as explained in the abovementioned sections, the implementation of the EGNOS-EDAS augmentation will be finalised harnessing the required augmentation messages provided by the SISNet service of the EDAS platform. In parallel, various filtering methods may be utilized to stabilize existing position.

In the backend side of things, a JSON schema will be created in order to annotate and validate the JSON documents uploaded by the geotagged mobile application. By describing the data format, the quality of the submitted data can be ensured.

With respect to the geotagged photos integrity framework, the OSNMA implementation will be integrated and subsequently a full test will be realised aiming to assess all the different cases.

5.3 Data preprocessing, outlier and novelty detection techniques

With respect to this module, an important step needs to be finalized: the standardization of the spectral measurements which ensures the quality of the spectra and is the server-side implementation of the theoretical framework. This key step is expected to be complete within the next two months of the project (by June of 2021), which after finalization will enable the acquisition of the final datasets required for:

- Developing and integrating the outlier analysis
- Developing and integrating the novelty detection algorithm, and
- Developing and integrating the final ML models that predict the soil properties

Finally, once the mobile app for the in-situ soil scanning system switches to using the SSO approach, the relevant updates from the server side should also take place.

Although the scaffolding for these components is set, what remains is to provide the final versions of these models and plug them in the relevant plugin of the DBMS.

Appendix A

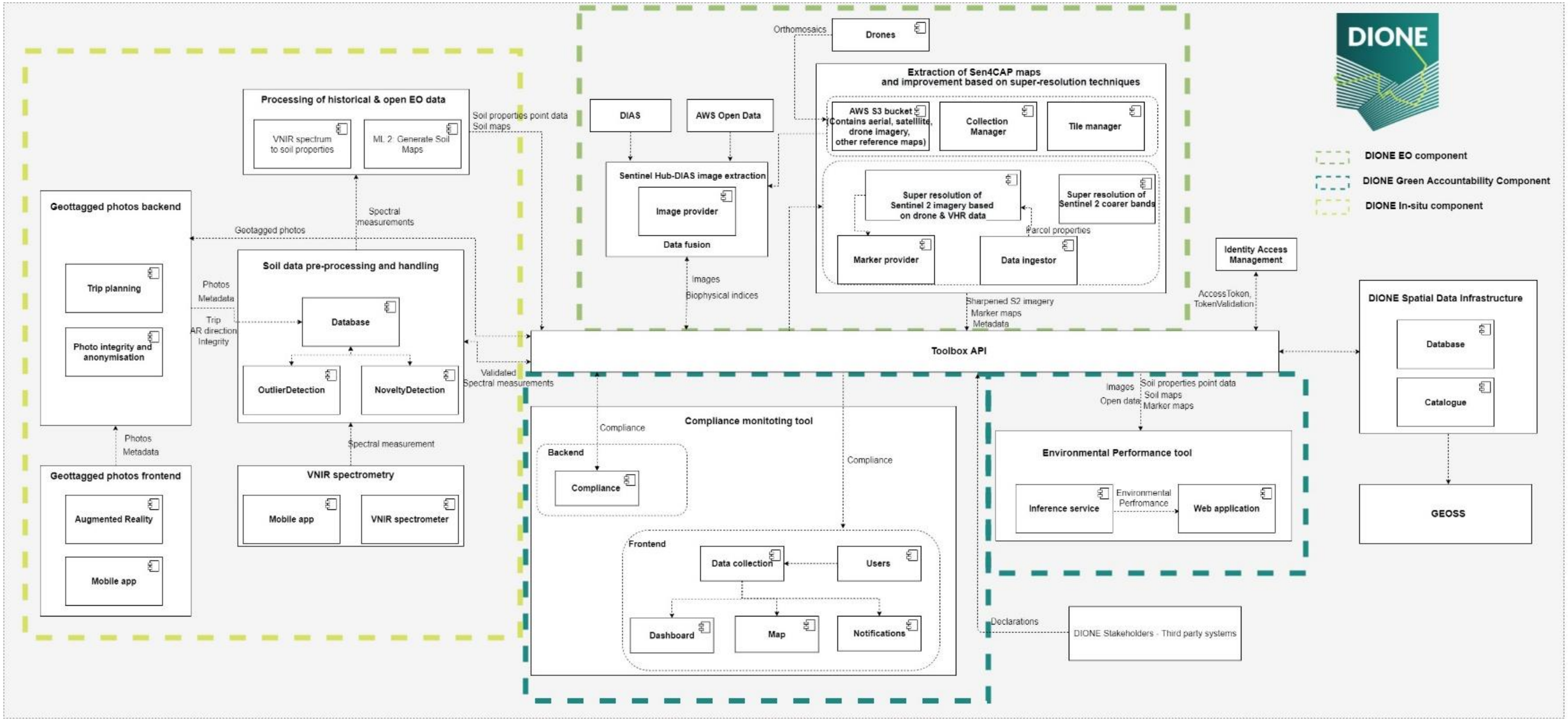


Figure 28: DIONE toolbox high-level architecture

Appendix B

SSS usage protocol for in-situ spectral measurements

Device preparation:

- Before usage, the device must be fully charged having the sensor's protective gorilla glass cleaned from leftovers from previous measurements.
- Switch on the device by pressing for three consecutive seconds the Power On button, on top of the device.
- Open the mobile application and switch on the mobile device's Bluetooth reception. Tap search, and when MEMS device appears, select it.

Sampling site selection and preparation:

The selected field must not have been irrigated and no rainfall must be observed during the last 3 days. The sensing site must be as much dryer and fine as possible. Find a small area (about 0.25m^2) and remove any external material that may exist (small rocks, vegetation, etc.).

- Measure surficial moisture content and record it at the application's interface field "Soil Moisture Content %" (This feature is not included in the alpha version). If a soil moisture sensor is not available, please characterize the soil moisture according to the following classification:
  - "A" – The sampling point is dry or slightly wet
  - "B" – The sampling point is moderately wet
  - "C" – There is increased moisture without the existence of stagnant water
  - "D" – There is stagnant water (the measurement will be ignored)
- Flatten the selected field with a spatula (Figure 29)
- If an undisturbed soil sampling tube exists, gently place it with the sharp edge facing bottom, and with steady hammer pounds (Figure 30) make sure that it is marginally concealed to the ground (Figure 31).

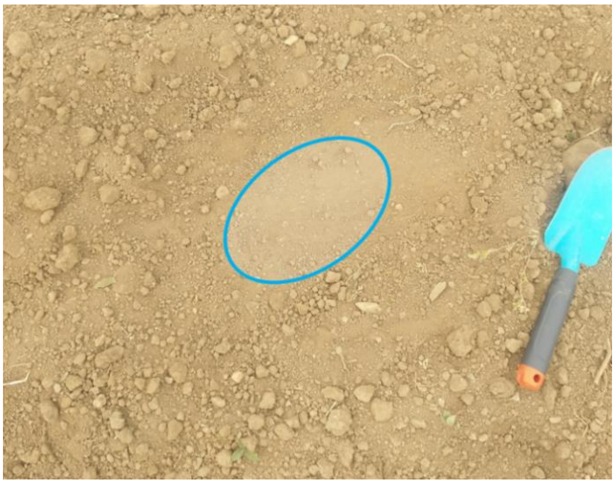


Figure 29: Spatula flattened soil surface

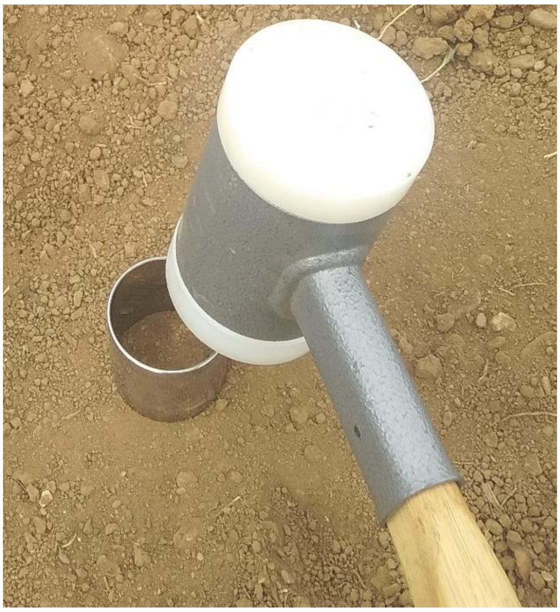


Figure 30: Undisturbed sampling tube placement with the aid of hammer

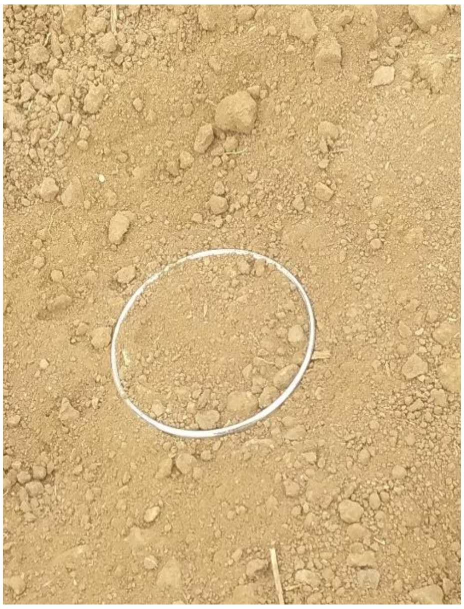


Figure 31: Undisturbed sampling tube marginally concealed to the ground

Calibration measurement:

- Place the white reference disk on a solid surface (ground), centered in front of the optic tube aperture with the white surface facing the sensor.
- Acquire a measurement by pressing the "White reference" button through the mobile application.
- Remove the panel and proceed to the next step.

Reflectance measurement:

- Place the device centered to the undisturbed soil sampling tube **Error! Reference source not found.Error! Reference source not found.**and acquire a spectral signature by pressing scan.
- On the prompted field, name the measurement as "FIELD_ID _DATE"
- Remove the device and repeat the procedure 4 more times.

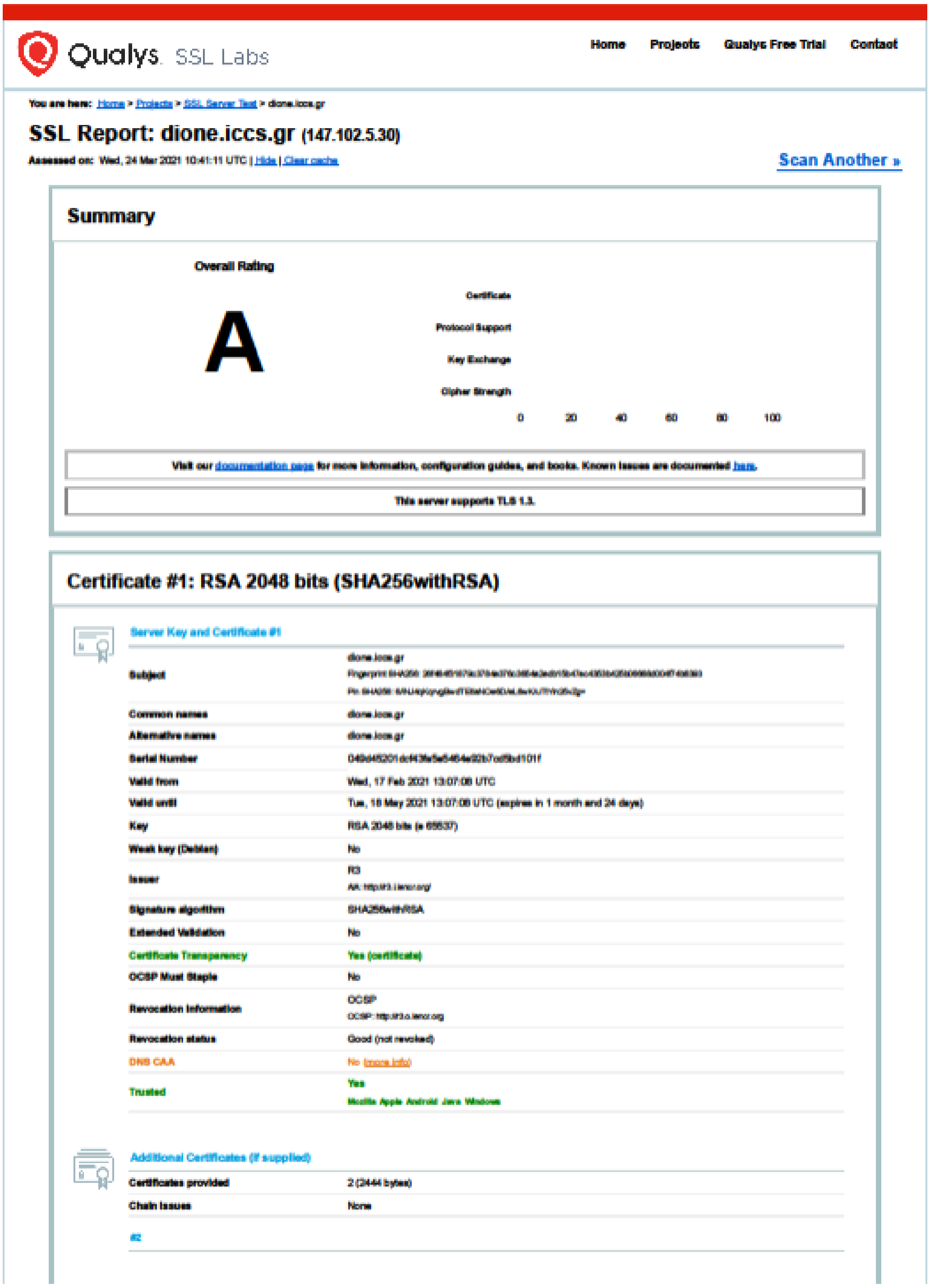
At most five distinct soil samples can be measured at once. For every quintet, calibration and standardization measurements must be renewed.

The data are automatically transmitted to the central database, but in absence of internet connection, the user must press the “Sync now” button at a later stage, when internet connection is established.

Appendix C

SSL/TLS Report by SSL Labs for dione.iccs.gr, which demonstrates the support of TLS v1.3.

SSL Server Test: dione.iccs.gr (Powered by Qualys SSL Labs) https://www.ssllabs.com/ssltest/analyze.html?d=dione.iccs.gr



You are here: Home > Projects > SSL Server Test > dione.iccs.gr

SSL Report: dione.iccs.gr (147.102.5.30)

Assessed on: Wed, 24 Mar 2021 10:41:11 UTC | Hide | Clear cache

Scan Another »

Summary

Visit our documentation page for more information, configuration guides, and books. Known issues are documented here.

This server supports TLS 1.3.

Certificate #1: RSA 2048 bits (SHA256withRSA)

Server Key and Certificate #1

Subject	dione.iccs.gr Fingerprint SHA256: [illegible] Pin SHA256: [illegible]
Common names	dione.iccs.gr
Alternative names	dione.iccs.gr
Serial Number	049d45201dcf43fe5e5464e92b7cd5bd101f
Valid from	Wed, 17 Feb 2021 13:07:08 UTC
Valid until	Tue, 18 May 2021 13:07:08 UTC (expires in 1 month and 24 days)
Key	RSA 2048 bits (e 65537)
Weak key (Debian)	No
Issuer	R3 AIA: http://r3.i.lencr.org/
Signature algorithm	SHA256withRSA
Extended Validation	No
Certificate Transparency	Yes (certificate)
OCSP Must Staple	No
Revocation information	OCSP OCSP: http://r3.o.lencr.org
Revocation status	Good (not revoked)
DNS CAA	No (more info)
Trusted	Yes Mozilla Apple Android Java Windows

Additional Certificates (if supplied)

Certificates provided	2 (2444 bytes)
Chain issues	None
#2	

1 of 5 3/24/2021, 12:43 PM

References

M. Fajardo P, F. Nájera, V. Pino, B. Minasny, and A. B. McBratney, “Spectral information related to soil slaking: An example from Australia,” *Geoderma Regional*, vol. 25, p. e00386, 2021, doi: https://doi.org/10.1016/j.geodrs.2021.e00386.

G. L. Roderick, *A History of Particle-Size Limits*, 1st ed. 1962.

https://insidegnss.com/galileo-hits-the-spot-testing-gnss-dual-frequency-with-smartphones/

https://www.gsa.europa.eu/newsroom/news/market-understands-value-dual-frequency#:~:text=Dual%2Dfrequency%20capability%20means%20that,can%20be%20used%20as%20backup.