SOFTENG 2022

The Eighth International Conference on Advances and Trends in Software Engineering

ISBN: 978-1-61208-946-1

April 24 - 28, 2022

Barcelona, Spain

SOFTENG 2022 Editors

Luigi Lavazza, University of Insubria - Varese, Italy

SOFTENG 2022

Forward

The Eighth International Conference on Advances and Trends in Software Engineering (SOFTENG 2022) continued a series of events focusing on challenging aspects for software development and deployment, across the whole life-cycle.

Software engineering exhibits challenging dimensions in the light of new applications, devices and services. Mobility, user-centric development, smart-devices, e-services, ambient environments, e-health and wearable/implantable devices pose specific challenges for specifying software requirements and developing reliable and safe software. Specific software interfaces, agile organization and software dependability require particular approaches for software security, maintainability, and sustainability.

We take here the opportunity to warmly thank all the members of the SOFTENG 2022 technical program committee, as well as all the reviewers. The creation of such a high-quality conference program would not have been possible without their involvement. We also kindly thank all the authors who dedicated much of their time and effort to contribute to SOFTENG 2022. We truly believe that, thanks to all these efforts, the final conference program consisted of top-quality contributions. We also thank the members of the SOFTENG 2022 organizing committee for their help in handling the logistics of this event.

SOFTENG 2022 Chairs

SOFTENG 2022 Steering Committee
Paolo Maresca, VERISIGN, Switzerland
Zeeshan Ali Rana, NUCES, Lahore, Pakistan

SOFTENG 2022 Publicity Chairs
Javier Rocher, Universitat Politècnica de València, Spain
Lorena Parra, Universitat Politècnica de València, Spain

SOFTENG 2022
Committee

SOFTENG 2022 Steering Committee

Paolo Maresca, VERISIGN, Switzerland
Zeeshan Ali Rana, NUCES, Lahore, Pakistan

SOFTENG 2022 Publicity Chairs

Javier Rocher, Universitat Politècnica de València, Spain
Lorena Parra, Universitat Politècnica de València, Spain

SOFTENG 2022 Technical Program Committee

Khelil Abdelmajid, Landshut University of Applied Sciences, Germany
Mo Adda, University of Portsmouth, UK
Bestoun S. Ahmed, Karlstad University, Sweden
Issam Al-Azzoni, Al Ain University of Science and Technology, UAE
Mahmoud Alfadel, Concordia University, Montreal, Canada
Vahid Alizadeh, College of Computing & Digital Media - DePaul University, USA
Washington Almeida, Cesar School | Center of Advanced Studies and Systems of Recife, Brazil
Hussein Almulla, University of South Carolina, USA / University of Anbar, Irak
Vu Nguyen Huynh Anh, Université Catholique de Louvain, Belgium
Pablo O. Antonino, Fraunhofer IESE, Germany
Darlan Arruda, University of Western Ontario, Canada
Jocelyn Aubert, Luxembourg Institute of Science and Technology (LIST), Luxembourg
Lerina Aversano, University of Sannio, Italy
Ali Babar, University of Adelaide, Australia
Doo-Hwan Bae, Software Process Improvement Center - KAIST, South Korea
Musard Balliu, KTH Royal Institute of Technology, Sweden
Mohamed Basel Almourad, College of Technological Innovation - Zayed University, Dubai, UAE
Imen Ben Mansour, University of Manouba, Tunisia
Maya Benabdelhafid, Ecole Supérieure de Comptabilité et de Finances (ESCF) de Constantine, Algeria
Marciele Berger, University of Minho, Portugal
Marcello M. Bersani, Politecnico di Milano, Italy
Anna Bobkowska, Gdansk University of Technology, Poland
Antonio Brogi, University of Pisa, Italy
Luigi Buglione, Engineering Ingegneria Informatica SpA, Italy
Azahara Camacho, University of Cádiz, Spain
Qinglei Cao, University of Tennessee, Knoxville, USA
José Carlos Metrôlho, Polytechnic Institute of Castelo Branco, Portugal
Pablo Cerro Cañizares, Universidad Complutense de Madrid, Spain
Allaoua Chaoui, University Constantine 2 - Abdelhamid Mehri, Algeria
Andrea D'Ambrogio, University of Rome Tor Vergata, Italy
Lilian Michele da Silva Barros, Instituto Tecnológico de Aeronáutica, Brazil
Luciano de Aguiar Monteiro, Institute of Higher Education iCEV - Teresina-Piauí, Brazil
Amleto Di Salle, University of L'Aquila, Italy
Sigrid Eldh, Ericsson AB, Sweden
Gencer Erdogan, SINTEF Digital, Norway

Fernando Escobar, PMI-DF Brasilia, Brazil
Naser Ezzati Jivan, Brock University, Canada
Faten Fakhfakh, National School of Engineering of Sfax, Tunisia
Stefano Forti, University of Pisa, Italy
Barbara Gallina, Mälardalen University, Sweden
Atef Gharbi, National Institute of Applied. Sciences and Technology, Tunisia
Pablo Gordillo, Universidad Complutense de Madrid, Spain
Jiaping Gui, NEC Laboratories America Inc., USA
Adriana Guran, Babes-Bolyai University, Cluj-Napoca, Romania
Ulrike Hammerschall, University of Applied Sciences Munich, Germany
Xue Han, University of Southern Indiana, Evansville, USA
Noriko Hanakawa, Hannan University, Japan
Qiang He, Swinburne University of Technology, Australia
Philipp Helle, Airbus Group Innovations - Hamburg, Germany
Samedi Heng, Université de Liège, Belgium
Birgit Hofer, Institute of Software Technology | Graz University of Technology, Austria
Jang Eui Hong, Chungbuk National University, South Korea
Fu-Hau Hsu, National Central University, Taiwan
LiGuo Huang, Southern Methodist University, USA
Rui Humberto Pereira, ISCAP/IPP, Portugal
Carlos Hurtado Sánchez, Tecnológico Nacional de México - campus Tijuana, Mexico
Miren Illarramendi, Mondragon University, Spain
Shinji Inoue, Kansai University, Osaka, Japan
Anca Daniela Ionita, University Politehnica of Bucharest, Romania
Jiajun Jiang, Tianjin University, China
Mira Kajko-Mattsson, Royal Institute of Technology, Sweden
Atsushi Kanai, Hosei University, Japan
Afrina Khatun, BRAC University, Bangladesh
Wiem Khlif, Mir@cl Laboratory | University of Sfax, Tunisia
Sang Kil Cha, KAIST, Korea
Alexander Knapp, Universität Augsburg, Germany
Johann Krautlager, Airbus Defence and Space GmbH, Germany
Dieter Landes, University of Applied Sciences Coburg, Germany
Seyong Lee, Oak Ridge National Laboratory, USA
Maurizio Leotta, University of Genova, Italy
Bruno Lima, INESC TEC | FEUP, Porto, Portugal
Qinghua Lu, CSIRO, Australia
Yingjun Lyu, University of Southern California, USA
Damian M. Lyons, Fordham University, USA
Jianbing Ma, Chengdu University of Information Technology, China
Ivan Machado, Institute of Computing - Federal University of Bahia, Brazil
Eda Marchetti, ISTI-CNR, Pisa, Italy
Paolo Maresca, VERISIGN, Switzerland
Johnny Marques, Aeronautics Institute of Technology, Brazil
Imen Marsit, University of Sousse, Tunisia
Danilo Martínez Espinoza, ESPE, Ecuador / Technical University of Madrid, Spain
Núria Mata, Fraunhofer Institute for Cognitive Systems, Germany
Ibéria Medeiros, University of Lisboa, Portugal

Mahshid Helali Moghadam, RISE Research Institutes of Sweden, Sweden
Edgardo Montes de Oca, Montimage, Paris, France
Fernando Moreira, Universidade Portucalense, Portugal
Ines Mouakher, University of Tunis El Manar, Tunisia
Krishna Narasimhan, Itemis AG, Stuttgart, Germany
Risto Nevalainen, FiSMA (Finnish software measurement association), Finland
Thien Nguyen, Oak Ridge National Laboratory, USA
Virginia Niculescu, Babes-Bolyai University, Cluj-Napoca, Romania
Stoicuta Olimpiu, University of Petrosani, Romania
Rafael Oliveira, UTFPR - The Federal University of Technology - Paraná, Brazil
Nelson Pacheco Rocha, University of Aveiro, Portugal
Luca Pascarella, Delft University of Technology, Netherlands
João Pascoal Faria, University of Porto, Portugal
Antonio Pecchia, Università degli Studi di Napoli Federico II, Italy
Fabiano Pecorelli, University of Salerno, Italy
Michael Perscheid, SAP Technology & Innovation, Germany
Dessislava Petrova-Antonova, Sofia University, Bulgaria
Fumin Qi, National Supercomputing Center in Shenzhen (Shenzhen Cloud Computing Center), China
Zhengrui Qin, Northwest Missouri State University, USA
Stefano Quer, Politecnico di Torino, Italy
Aamir Raihan, University of British Columbia, Canada
Raman Ramsin, Sharif University of Technology, Iran
Zeeshan Ali Rana, National University of Computer and Emerging Sciences (FAST-NUCES), Lahore, Pakistan
Mohammad Reza Nami, Islamic Azad University-Qazvin, Iran
Oliviero Riganelli, University of Milano - Bicocca, Italy
Michele Risi, University of Salerno, Italy
Simona Riurean, University of Petrosani, Romania
António Miguel Rosado da Cruz, Higher School Technology and Management - Polytechnic Institute of Viana do Castelo, Portugal
Gunter Saake, Otto-von-Guericke-Universitaet, Magdeburg, Germany
Sébastien Salva, University Clermont Auvergne, France
Hiroyuki Sato, University of Tokyo, Japan
Daniel Schnetzer Fava, University of Oslo, Norway
Ruth Schorr, Frankfurt University of Applied Sciences, Germany
Tomas Schweigert, SQS Software Quality Systems AG, Germany
Ali Sedaghatbaf, RISE Research Institute of Sweden, Sweden
Rocky Slavin, University of Texas at San Antonio, USA
Cristovão Sousa, Polytechnic Institute of Porto / INESC TEC, Portugal
Yoshihisa Udagawa, Tokyo University of Information Sciences, Japan
Miroslav Velev, Aries Design Automation, USA
Colin Venters, University of Huddersfield, UK
Roberto Verdecchia, Gran Sasso Science Institute (GSSI), Italy / Vrije Universiteit Amsterdam (VU), Netherlands
Flavien Vernier, Université Savoie Mont Blanc, France
László Vidács, University of Szeged, Hungary
António Vieira, University of Minho, Portugal
Shaohua Wang, New Jersey Institute of Technology, USA

Ralf Wimmer, Concept Engineering GmbH / Albert-Ludwigs-Universität Freiburg, Freiburg im Breisgau, Germany
Xiaofei Xie, Nanyang Technological University, Singapore
Cemal Yilmaz, Sabanci University, Istanbul, Turkey
Levent Yilmaz, Auburn University, USA
Peter Zimmerer, Siemens AG, Germany
Alejandro Zunino, ISISTAN, UNICEN & CONICET, Argentina

Copyright Information

Table of Contents

An Integrated EO-based Toolbox for Modernising CAP Area-based Compliance Checks and Assessing Respective Environmental Impact

Orestis Sampson, Nikos Iliakis, Valantis Tsiakos, Maria Krommyda, Angelos Amditis
Institute of Communication and Computer Systems
Athens, Greece
emails: {orestis.sampson, nikos.iliakis, valantis.tsiakos, maria.krommyda, a.amditis}@iccs.gr

***Abstract*—As part of its ongoing move to simplify and modernise the EU's Common Agricultural Policy (CAP), the European Commission has adopted new rules that allow a range of modern technologies to be used in the check systems for area-based CAP payments. This includes the possibility to use geotagged photos to support and complement checks when the latter do not lead to conclusive results and additionally to help avoid wasting time and money on the spot checks. They can also be used as ground truth information provided by farmers or other stakeholders. A system that can support the farmer in collecting the needed geotagged photos is presented here. The system will help with the automation and acceleration of a heavily manual process by facilitating the interaction between the farmers and the relevant authorities.**

Keywords-Common Agricultural Policy; Earth Observation; remote sensing; drones; geotagged photos; environmental performance; farmers' compliance.

I. Introduction

The DIONE project proposes a close-to-market area-based direct payments monitoring toolbox that addresses the Modernised Common Agricultural Policy (CAP) regulation of using automated technologies to ensure more frequent, accurate and inexpensive compliance checks. In particular, the toolkit showcases the capability of Sentinel data to monitor the crop diversification rules and integrates the generated crop-type maps in a way directly exploitable by the paying agencies. It includes in the analysis the so far neglected Ecological Focus Area (EFA) types, such as fallow land of all sizes, buffer strips, hedges and trees, by making use of super-resolution technology that improves the 10-20m Sentinel resolution to an improved resolution range. The toolkit also complements the use of Earth-Observation (EO) data with a system of reliable, ground-based geotagged photos, captured by the farmers, that exploits advances that allow for improved positional accuracy, low-footprint encryption techniques for improved data security and reliability and image detecting manipulation techniques. The system allows for an improved Land Cover/Land Use annotation and ensures the process is untampered.

The main objective of this paper is to provide an overview of the geotagged photos in-situ component for complementing EO-data which are developed within the frame of the DIONE project. The geotagged photos are captured using smartphones and allow for an improved method for the provision of additional evidence regarding the CAP compliance monitoring. 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 (AR) features.

Finally, a central data processing and storage system gathers the data, ensures their quality and provides user-friendly Application Program Interfaces (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 [10] is not compromised, that no data with highly variable/outlier nature is going to be used and that appropriate forms of data storage are used to ensure easy retrieval.

The rest of this paper is organized as follows. Section II describes the design of the system, core tech functionalities and use cases. Section III addresses the front-end side of the mobile application and describes the User Interface (UI)/User Experience (UX) aspects. Section IV offers some thoughts on the outcomes and future work. The acknowledgement and references close the article.

II. System Design

The data collection process is supported by a mobile application that exposes to the user all the related content about his/her parcels while enabling the conclusion of the process and the provision of the final photos to the Paying Agencies (PAs). On the other hand, a set of backend processes [15] [16] provide the necessary instructions for farmers to reach a given parcel as well as directions regarding the process of capturing appropriately a photo of a given parcel.

In particular, the application enables the user to check if there are any requests from the Paying Agency for the provision of geotagged photos and, based on this, facilitate the overall process. The latter includes the provision of information on map related to the user's parcels and his/her current position, navigation to the correct spot in the parcel for the photo acquisition, as well as use of AR to dictate the exact spot and direction he/she needs to place his/her mobile device while superimposing parcel boundaries to facilitate the process.

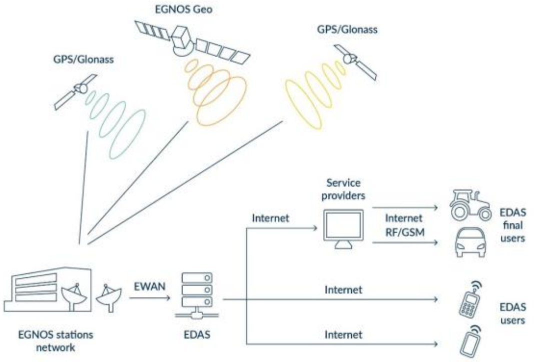


Figure 1. EDAS Hig-Level Architecture

Ensuring the best possible positional accuracy is another essential requirement [12], whilst the application allows the exploitation of raw Global Navigation Satellite System (GNSS) measurements and enables the user to evaluate his/her positional accuracy. The photo acquisition process can also be initiated without an initial request from the Paying Agencies, allowing users to act proactively and provide geotagged photos in advance so as to assist compliance evaluation for their parcels.

Moreover, aiming to support ease of use, the application provides local language menus, user friendly visualisation of detailed parcel information, as well as a tutorial with step-by-step information of how to capture photos.

Following the collection of the needed photos, the application enables the transmission of the collected figures along with their associated metadata. The minimum required metadata that are stored during the data acquisition process include:

- Time, date and geographical location of the photo acquisition: This information is extracted directly without manual interference from the GNSS antenna embedded in the mobile devices. The positional accuracy can be improved to a meter or even sub-meter one, for the devices that can harness multiple location differentiators and European Geostationary Navigation Overlay Service (EGNOS) services.
- Orientation, heading, of the camera at the time of photo capture: Its goal is to ensure the proper acquisition of the photo, pointing at the correct parcel. This information can be extracted from the compass system embedded in the device (10 degrees deviation) while also being deduced by exploiting the AR features of the geotagged photos framework. Regarding the latter, AR allows to identify and superimpose on the photo the land parcel border or other identifiable landmarks, and subsequently to capture multiple photos as needed and properly adjust the device positioning.
- Identification of the operator.
- Basic information on the mobile device and inbuilt camera i.e., mobile device brand, camera model, and focal length.

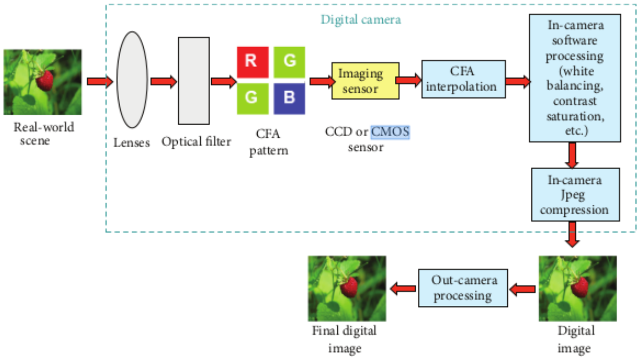


Figure 2. A scheme representing the steps composing the usual life cycle a digital image undergoes. Imperfections in the CMOS manufacturing process introduce noise in the photos taken.

The application is available to operate both in online and offline mode. In this context, all data needed for the guidance of the user must be fetched beforehand and preferably while on a Wi-Fi connection. Thus, all actions and data acquired on-site must be stored locally in order to be sent to the backend later.

The format of the acquired photos is stored according to the most commonly used standards in digital cameras. In this context, appropriate settings were considered towards ensuring appropriate quality of the photo as well as avoid losing image details due to compression levels.

Users of the application are able to receive detailed instructions in order to fulfil the data acquisition process. In line also with the EU guidelines, a macro/panoramic photo is captured with the camera pointing higher at the horizon showing the corresponding field. In some cases, this type of format may represent the optimal evidencing option (i.e., mixture of crop as EFA cover crop).

On the other hand, in order for the photos to be reliable there is a need to validate their integrity and origin and detect any attempts of digital manipulation of the photos. This is achieved through a server-side process (framework) that runs successively different image forensics techniques to locate any digital manipulation and ensure that the photo is taken by the correct user (farmer) in the correct parcel. The component firstly ensures that the file of the photo comes from the same device that the farmer used to authenticate himself/herself in the mobile application. In the second step, the photo file is checked aiming to detect any tampering from the time the image was taken and until it was uploaded to the system while also using methods to ensure location integrity. In the next steps of the process, forensics techniques specifically designed for photos are executed. In these steps, the goal is to verify that the user did not manipulate the photo digitally.

This component also addresses the need to preserve private and personal information that may be exposed in the taken photos when they are reviewed by the inspectors of the Paying Agency. Thus, the last part of the server-side framework is an anonymization tool that is responsible for blurring any faces or license plates detected in the photos.

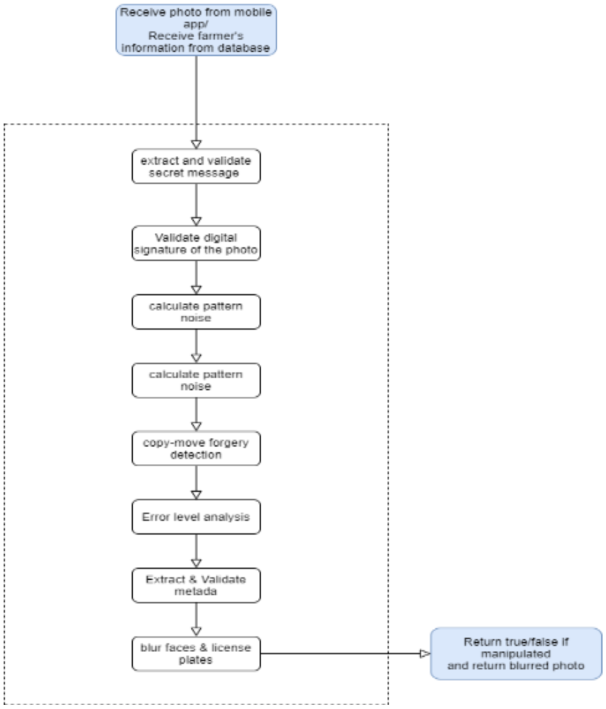
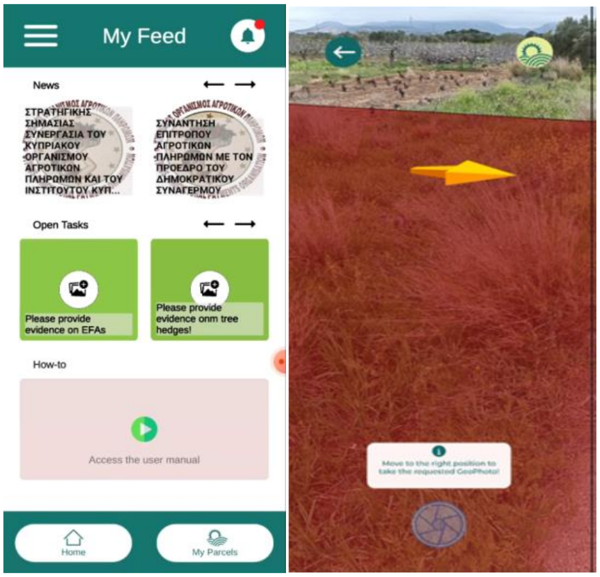

Receive photo from mobile app/
Receive farmer's information from database
extract and validate secret message
Validate digital signature of the photo
calculate pattern noise
calculate pattern noise
copy-move forgery detection
Error level analysis
Extract & Validate metada
blur faces & license plates
Return true/false if manipulated and return blurred photo

Figure 3. The overall integrity process.

Data acquisition. The starting point for the application use is through the transmission of a push notification by the toolbox API to the geotagged photos component regarding the parcel requiring the acquisition of geotagged photos. A geotagged photo consists of saving at least the location as latitude and longitude coordinates, date, time, orientation and the mobile device/camera information into a JavaScript Object Notation (JSON) file for each Portable Network Graphics (PNG) photo file. The notification is transmitted downstream to the mobile application in order to notify farmers regarding the parcel requiring geotagging. Through the geotagging photo app, farmers can view the route towards the parcel under investigation. Subsequently, they receive guidance regarding the process of taking a photo of the parcel with the use of AR techniques. The photo taken is uploaded and checked for integrity by the backend of the geotagging component and it is stored in a database. The compliance dashboard requests the geotag photo object which is fetched through the toolbox API.

All app users are linked to their respective farmer unique id. This is achieved during the authentication-authorisation process and is facilitated by the toolbox API that is dynamically connected with the Paying Agencies systems. Moreover, in order to benefit from all modern differentiators, a recent Android device is needed, that provides access to real time data extracted from raw GNSS data. The platform has strict limitations since the photos taken must represent the real state of a parcel at a very specific time and day. As many techniques as possible are needed in order to maximise the accuracy of the position.

The user is able to monitor positional metadata extracted from raw GNSS data.



ΣΤΡΑΤΗΓΙΚΗΣ ΣΗΜΑΣΙΑΣ ΣΥΝΕΡΓΑΣΙΑ ΤΟΥ ΚΥΠΡΙΑΚΟΥ ΟΡΓΑΝΙΣΜΟΥ ΑΓΡΟΤΙΚΩΝ ΠΛΗΡΩΜΩΝ ΚΑΙ ΤΟΥ ΙΝΣΤΙΤΟΥΤΟΥ ΚΥΠ...

ΣΥΝΑΝΤΗΣΗ ΕΠΙΤΡΟΠΟΥ ΑΓΡΟΤΙΚΩΝ ΠΛΗΡΩΜΩΝ ΜΕ ΤΟΝ ΠΡΟΕΔΡΟ ΤΟΥ ΔΗΜΟΚΡΑΤΙΚΟΥ ΣΥΝΑΓΕΡΜΟΥ

Please provide evidence on EFAs

Please provide evidence onm tree hedges!

Access the user manual

Move to the right position to take the requested GeoPhoto!

Figure 4. Screenshots from the mobile application: (a) Overview page, (b) AR support for photo capturing

Based on the user's device and Internet connectivity, they are able to benefit from different European Global Navigation Satellite System (EGNSS) differentiators (e.g., dual frequency GNSS [13]) to achieve improved positional accuracy and to assure positional data integrity. Positional accuracy requires data connection to be able to receive correctional data from the EGNOS Data Access Service (EDAS) service (Fig. 1). EDAS provides a wide variety of products, in different formats and different protocols [14]. Among these products, EDAS provides the EGNOS augmentation messages, as normally received by users via the EGNOS geostationary satellite. This message is transmitted in real-time by the EDAS Signal-In- Space (SISnet) service and is useful for users that won't have an EGNOS supporting device.

Positional data integrity can be achieved by analysing raw GNSS data for Open Service Navigation Message Authentication (OSNMA) [1][2][3] verifying Galileo signalling. In order to identify the Galileo messages, the raw bits of E1 I/NAV messages are taken from the receiver using Android calls and from there, the OSNMA relevant bits are extracted for authentication. Since all photo metadata must be as accurate as possible, the application makes sure that the positional accuracy is below the required threshold before the image capturing.

Data integrity. In order to ensure that the file of the photo is not manipulated and is the same from the time the photo was taken, the technique of steganography [4] is utilised. Steganography is the practice of concealing a message within a file. In our case, we conceal a secret message right after the photo is taken and try to extract this message on the server side. Knowing that the message is concealed within the file, a successful extraction of the message would mean that the file is not tampered with. In the second step, a light-weight signature scheme [5] [6] aims to facilitate the secure transmission of the photo and operates as a pairing mechanism between the farmer and the mobile device used. During the sign in process, a pair of a private and a public key would be generated, and the public key is sent to the server to be saved. Every time the farmer takes a photo, the private key is used to sign the photo. The signing procedure ensures that the photo which was sent to the server was taken by the same device the farmer used to sign in. The signing scheme operates as a second level of assurance that the photo was not tampered on top of the steganography, as in the case of tampering, the signature is not able to be verified in the server process.

In the next phase of the server-side procedure, an algorithm is executed so as to identify the device used to capture 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 CMOS image sensor, which inserts a pattern noise in the photos which is unique for each device [7]. The process is depicted in Fig. 2. This step is useful to confirm that the noise matches the device the farmer is using.

In what follows, two photo specific forensic algorithms are implemented. The first one is the copy-move forgery detection algorithm. In the copy-move forgery, the malicious actor replicates a portion of the photo inside the photo [8] [9]. The detection algorithm allows to avoid this kind of forgery. The second technique regards the way the information is saved inside the photo. As shown in Fig. 2, different processing steps take place before the final photo/file is created. The final step is the appliance of a compression. Every photo file is compressed using a specific algorithm. 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. So, the second forensic algorithm constitutes an investigation of the compression levels. One example of such an investigation is the error level analysis.

In the next step of the procedure, a tool is used to extract the metadata that is embedded in the photo. This extracted data is verified with the data that have been stored in the database i.e., parcel location, time of photo etc. The main information to be obtained involves the location of the photo. Using the Galileo OSNMA, it can be guaranteed that users are utilising non- counterfeit navigation data coming from the Galileo satellites.

Finally, a pre trained convolutional neural network is used so as to locate faces and license plates and blur them. The overall process is depicted in Fig. 3.

For the implementation of the data integrity framework, the python programming language is utilised. For the first two steps, there are respective libraries (stegano and crypto libraries) that are exploited for the implementation. One thing to consider here is that we have to be careful in the choice of the techniques used due to the fact that some parts are implemented in the mobile application. So, we have to ensure that the respective parts should be possible to be implemented in C#, which is the language of the implementation of the mobile applications. Finally, one more benefit that python gives is the ability to execute bash commands and interact with other tools. A final consideration that should be taken into account, is which variation of the steganography technique to implement. That is because, we could make the algorithms in the last steps unable to be used due to the structure alteration of the photo, by embedding a secret message.

III. Mobile Application

The application is developed in the Unity game engine to benefit from the integrated AR solution that ships with the engine. Unity is a cross-platform game engine developed initially as a Mac OSX 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, AR applications. AR Foundation is a cross-platform framework built for the Unity engine that allows to build AR experiences once, then build for either Android or iOS devices. The package presents an interface for Unity developers to use, but does not 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: (i) ARCore XR Plugin on Android (ii) ARKit XR Plugin on iOS.

Along with Unity, some native android plugins were developed, mainly to handle the low-level operations required for the raw measurements handling and integrity aspects [11]. The AR component aims to provide directions to farmers in order to enable retrieval of representative photos of a given parcel. 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 and to be stored in the Central Database.

Home screen – Authentication/Authorization. The geotagged mobile application provides translated text interface to accommodate for the users in different countries. The application is not open for general use, it is available only for registered users. In order to achieve this, the user must have registered themselves in the platform.

Content visualization. The application 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, along with the latest pending Tasks for the user (Fig. 4), 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. More than that, the settings page is there to provide visual feedback in the form of the traffic light approach about the EGNSS differentiators supported by the device.

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 Toolbox API triggers a push notification to be sent towards the mobile device. The Tasks have an associated status based on their progress which can be open, pending or completed. The push notification system for the mobile application, relies fully on the OneSignal service.

Navigation to parcel and defined spots. After launching a specific Task, the user is presented with a map view. Its purpose is to guide the user to the Parcel this Task is associated with. 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 Software Development Kit (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. The routing functionality is provided by the Mapbox Directions API external service. The 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 intention is to obtain sufficient information in order to avoid any physical field visits 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 (Fig. 4). There are two types of photos that are required from the user, landscape or portrait.

A landscape photo should depict a larger part of the field and include elements other than the main object such as crop and activity, 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 portrait photo must serve to enable the robust identification of the element to evidence. This subject could be a mixture of crop as EFA cover, presence of rare crops that cannot be reliably discriminated in the Sentinel data etc. In the context of a Task created by a Paying Agency Inspector, the type of photo as well as the preferred camera orientation are specified during the creation of the Task in the Compliance monitoring tool.

By 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 allowed to take a photo from inside the parcel or near each one of the parcel's corners. 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.

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. In conjunction with ARFoundation, 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. 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.

The positional accuracy is being tracked and the digital content is drawn only when the accuracy is below a specific ceiling. 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 Exchangeable image file format (EXIF) metadata to the file as it usually happens. Thus, a dedicated mechanism is employed, that takes the background of the image and in the resulting PNG file a custom method is applied to decode and embed the required metadata.

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 take place outdoors. Agricultural parcels are often situated in remote and mountainous places that are not covered by mobile network signal. Therefore, it is crucial that the functionalities of the application can be performed when no mobile signal reception is available.

Since the data presented to the user needs to be as recent as possible, the first steps of the process require an Internet connection to fetch the relevant data. 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.

IV. Conclusion And Future Outlook

The presented solution offers a reliable and secure way to modernise the EU's CAP. The farmers are given the ability to capture geotagged photos to support their applications and complement other systems when the latter do not lead to conclusive results. More importantly, it strengthens the interaction between the farmers and the relevant authorities.

In order to improve the UX of the application, more guidance will be needed for the application user, especially anyone unfamiliar with such technologies. Hence, a screen needs to be added with brief instructions on what is required of them, how to take a photo, the restrictions applied etc. An instructional video may also be added. Since AR systems rely heavily on location accuracy as explained in the above-mentioned sections, the implementation of the EGNOS-EDAS augmentation needs to 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, the creation of a JSON schema is 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 needs to be integrated and subsequently a full test to be realised aiming to assess all the different cases.

Acknowledgment

This work is a part of the DIONE project. This project has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 870378.

References

[1] I. Fernandez-Hernandez et al., “A Navigation Message Authentication Proposal for the Galileo Open Service,” Navigation, Journal of The Institute of Navigation, vol. 63, no. 1, pp. 85-102, 2016.

[2] Galileo Navigation Message Authentication Specification for Signal-In-Space Testing – v1.0

[3] B. Motella, M. Nicola, and S. Damy, ”Enhanced GNSS Authentication Based on the Joint CHIMERA/OSNMA Scheme,” in IEEE Access, vol. 9, pp. 121570-121582, 2021, doi: 10.1109/ACCESS.2021.3107871.

[4] N. F. Johnson and S. Jajodia, ”Exploring steganography: Seeing the unseen,” in Computer, vol. 31, no. 2, pp. 26-34, Feb. 1998, doi:10.1109/MC.1998.4655281.

[5] R. Kaur and A. Kaur, ”Digital Signature,” 2012 International Conference on Computing Sciences, 2012, pp. 295-301, doi: 10.1109/ICCS.2012.25.

[6] An Introduction to Digital Signature Schemes arXiv:1404.2820 [cs.C]

[7] Digital Camera Identification from Sensor Pattern Noise

http://www.ws.binghamton.edu/fridrich/research/double.pdf [retrieved: May, 2021]

[8] W. Luo, J. Huang, and G. Qiu, ”Robust Detection of Region-Duplication Forgery in Digital Image,” 18th International Conference on Pattern Recognition (ICPR'06), 2006, pp. 746-749, doi:10.1109/ICPR.2006.1003.

[9] A. C. Popescu and H. Farid, “Exposing Digital Forgeries by Detecting Duplicated Image Regions.” , 2020.

[10] The Metadata in PNG files, https://dev.exiv2.org/projects/exiv2/wiki/The Metadata in PNG files [retrieved April, 2021]

[11] Android plugins in Unity, https://docs.unity3d.com/Manual/PluginsForAndroid.html [retrieved September, 2021]

[12] Use of geotagged photographs in the frame of Common Agriculture Policy checks https://marswiki.jrc.ec.europa.eu/wikicap/images/c/ce/Geotagged JRC Report1.pdf [retrieved: October, 2021]

[13] Advantages of dual-frequency GNSS in smartphones

https://www.geospatialworld.net/blogs/advantages-of-dual-frequency-gnss-in-smartphones/ [retrieved: November, 2020]

[14] RTCA DO-229 Minimum Operational Performance Standards (MOPS) for Global Positioning System/Satellite-Based Augmentation System Airborne Equipment

[15] MongoDB: the application data platform https://www.mongodb.com/ [retrieved: March, 2021]

[16] The Open Source API Service for the Modern Web https://restheart.org/ [retrieved: March, 2021]

Requirements for an AI-enabled Industry 4.0 Platform - Integrating Industrial and Scientific Views

Eichelberger Holger
Software Systems Engineering
University of Hildesheim
Hildesheim, Germany
e-mail: eichelberger@sse.uni-hildesheim.de

Stichweh Heiko
Head of Innovation
Lenze SE
Hamelin, Germany
e:mail: heiko.stichweh@lenze.com

Sauer Christian
Software Systems Engineering
University of Hildesheim
Hildesheim, Germany
e-mail: sauer@sse.uni-hildesheim.de

Abstract*—Intelligent manufacturing is one goal of smart industry/Industry 4.0 that could be achieved through Artificial Intelligence (AI). Flexibly combining AI methods and platform capabilities, such as dynamic offloading of code close to production machines, security or interoperability mechanisms are major demands in this context. However, recent Industry 4.0 software platforms fall short in various of these demands, in particular in upcoming ecosystem scenarios, e.g., when data or services shall be shared across platforms or companies without vendor lock-ins. The aim of the funded Intelligent Industrial Production (IIP) IIP-Ecosphere project is to research concepts and solutions for 'easy-to-use' AI in Industry 4.0 and to demonstrate the results in a prototypical software platform. Core questions are which demands shall drive the development of such a platform and how a feasible set of requirements can be determined that balances scientific and industrial interests. In this paper, we discuss our approach on eliciting requirements in this context for two interlinked requirements perspectives, a usage and a functional view. In summary, we collected 67 usage view activities / scenarios and 141 top-level requirements with 179 detailing sub-requirements. About 35% of the requirements have so far been realized in a prototype and some of the identified concepts are currently being taken up by a standardization initiative for edge devices in Industry 4.0.

***Keywords*—Industry 4.0 platforms; intelligent production; AI; requirements; edge; adaptation; asset administration shell.**

I. Introduction

The digitization of industry increases the performance of technical systems and their processes, but also their complexity. Intelligent manufacturing (smart industry, Industry 4.0) can be realized through application of Artificial Intelligence (AI) in the production context. This is perceived as an enabler for an increase of productivity of up to 50% [1]. However, currently more than 75% of AI applications are ultimately not deployed [2], e.g., as they are not considered to be production ready or as they are not easily applicable by domain users.

One further trend in Industry 4.0 are edge devices. As an evolution of Programmable Logic Controllers (PLC), they are frequently used for retrofitting, e.g., equipping legacy manufacturing machines with recent communication protocols. Moreover, modern edge devices combine hard real-time functions connected to the manufacturing machines with soft/non real-time IT capabilities. Some recent edge devices even ship with modern hardware accelerators, such as Graphic Processing Units (GPUs) or Tensor Processing Units (TPUs), which are often beneficial for AI calculations. While edge devices allow for offloading IT functionality close to production machines, e.g., to operate AI at low latency, they also significantly increase the management and deployment complexity in Industry 4.0 setups by emphasizing distributed on-premise computing.

To support companies in managing this complexity, several software platforms for Cyber-Physical Production Systems (CPPS) or Industrial Internet of Things (IIoT/IoT) applications are available, e.g., Siemens MindSphere or PTC ThingWorx. As we discussed in [3], these platforms significantly differ in their capabilities, in particular with respect to AI, edge offloading or cloud usage. Moreover, they often fall short in providing capabilities for consistent (distributed) system customization, one key capability to cope with the complexity, but also in data protection or data/service sharing for ecosystem setups.

In the funded project IIP-Ecosphere, we are researching concepts for easy-to-use AI in the manufacturing domain. The overall mission of IIP-Ecosphere is to create an ecosystem of involved stakeholders for the mutual transfer of experience and knowledge. For demonstrating the approaches, the partners develop a prototypical IIoT platform. On the one side, the requirements for such a platform must reflect the scientific goals and pave the way for experimenting with and demonstrating of novel approaches. On the other side, such a platform must also be interesting for industrial stakeholders and support production requirements. Thus, an elicitation of platform requirements needs to be carefully balanced.

Our main questions are 1) how to collect and combine scientific and industrial requirements in an Industry 4.0 context and 2) can different views on the requirements be used to improve their mutual completeness? As contributions we present a pragmatic combination of scientific methods, e.g., surveys, with requirements elicitation techniques in the context of an industrial reference process for systems design. This involves the creation two complementing views, a usage, as well as a functional/quality view on the requirements and allows for a more encompassing requirements collection, but also a discussion of mutual influences. We provide insights into elicited requirements and experiences that we made.

In summary, we collected 67 usage view activities/scenarios and 141 top-level requirements further detailed by 179 sub-requirements. These requirements characterize the (prioritized) desires for an AI-enabled Industry 4.0 platform. Intentionally, we were open to requirements that will probably not be realized during the lifetime of IIP-Ecosphere in order to provide inspiration for future works. At the point of writing, about 35% of the requirements have been realized in a prototypical open source platform and several of the identified concepts are being taken up by a standardization initiative for edge devices in Industry 4.0. Moreover, some industrial IIP-Ecosphere partners adopted our integrated requirements approach to improve their internal software development processes.

This paper is structured as follows: In Section II, we provide a brief overview of the IIP-Ecosphere project. In Section III, we introduce our approach for requirements collection and discuss results from that approach in the following sections, i.e., on a detailed platform survey in Section IV and for the requirements collection with two views in Section V. In Section VI, we discuss related work and in Section VII we conclude this paper and outline future work.

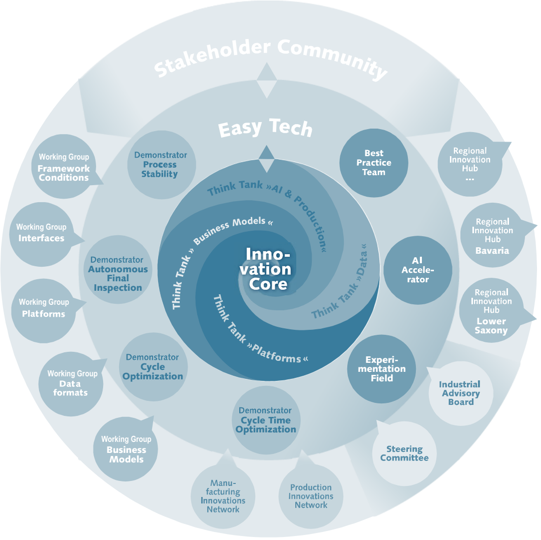

Stakeholder Community
Easy Tech
Working Group Framework Conditions
Demonstrator Process Stability
Best Practice Team
Regional Innovation Hub ...
Working Group Interfaces
Regional Innovation Hub Bavaria
Think Tank »AI & Production«
Think Tank »Business Models«
Innovation Core
Think Tank »Data«
Think Tank »Platforms«
Demonstrator Autonomous Final Inspection
AI Accelerator
Working Group Platforms
Regional Innovation Hub Lower Saxony
Working Group Data formats
Demonstrator Cycle Optimization
Experimentation Field
Industrial Advisory Board
Demonstrator Cycle Time Optimization
Working Group Business Models
Steering Committee
Manufacturing Innovations Network
Production Innovations Network

Figure 1. IIP-Ecosphere project structure.

II. IIP-Ecosphere Project Overview

Our work takes place in the context of the IIP-Ecosphere project, which is funded by the German ministry for Economics and Energy in its AI innovation competition program. IIP-Ecosphere aims at achieving an innovative leap in the field of industrial production exploiting networked, intelligent, autonomous system capabilities to increase productivity, flexibility, robustness and efficiency of Industry 4.0. The goal is to build a novel ecosystem of humans (through companies and organizations), software, machines and products with a specific focus on mutual experience and knowledge transfer.

To achieve this, the activities in IIP-Ecosphere are structured in three layers, as illustrated in Figure 1. The *Innovation Core* is at the heart of the ecosystem and is constituted by four so called think tanks performing research on core topics, such as platforms, AI, business models and data. The *Easy Tech* layer aims at demonstrating the research results and transferring them into industrial practice, in particular through the *AI Accelerator*, which works, e.g., on a public catalog of AI solutions and on generalized, (re-)usable AI services for manufacturing. Finally, the *Stakeholder Community* conducts activities for external parties, e.g., workshops on the core topics or linking of linking start-ups, SMEs, large companies and multiplicators with the project (*Regional Innovation Hubs*). This paper is based on joint activities of the platform think tank, the AI accelerator and the demonstrators. After the end of the project's lifetime, the created community/ecosystem shall continue the project activities on its own.

One core activity in IIP-Ecosphere is the realization of a virtual platform that connects existing devices and factory installations in a vendor-independent manner. A *virtual platform* [4] takes up functionality and services of existing, already installed protocols and platforms, integrates them and offers additional services on top of these. In IIP-Ecosphere, we aim at enabling intelligent manufacturing applications based on an open set of re-usable AI and platform services. These services shall be flexibly distributed to available resources, such as edge devices, on-premise servers or clouds. The service distribution shall be determined by the platform before starting an application, but also during run-time, i.e., in a self-adaptive manner. As requested by the funding scheme, IIP-Ecosphere strives for concepts and methods to achieve/increase vendor-neutrality, interoperability and flexible uptake of Industry 4.0 related standards, e.g., Message Queuing Telemetry Transport (MQTT) [5], or Open Platform Communications Unified Architecture (OPC UA) [6].

III. Approach

The realization of such a platform is not only a technical endeavor that commands the application of software engineering methods, such as requirements engineering or architectural design. It forms a data-driven system and, thus, faces challenges that are, e.g., discussed in [7]. Particular challenges are highly interdisciplinary teamwork (production, AI, data science, software engineering, economics) including researchers and practitioners, but also volatile and unclear requirements due to explorative AI and data science processes.

As stated above, we head for a research-integrated requirements collection, which is based on relevant standards/approaches for Industry 4.0 and IIoT. For system development, the German Standardization Roadmap Industry 4.0 [8] advocates the Industrial Internet Reference Architecture (IIRA) [9], in particular the so-called 'Industrial Internet Viewpoints'. Figure 2 a) illustrates these interlinked viewpoints, consisting of a *business view* (roles attributed with business interests), an *usage view* (a use case collection for all involved roles and system entities), a *functional view* (domain decomposition of system functions) and an *implementation view* (detailed architecture). This approach is also used in relevant inputs for our work, particularly in an international effort to standardize edge computing in manufacturing [10]. However, like several other works [11]–[13], the IIRA approach focuses on the technical side, neglecting research demands.

ISBN: 978-1-61208-946-1

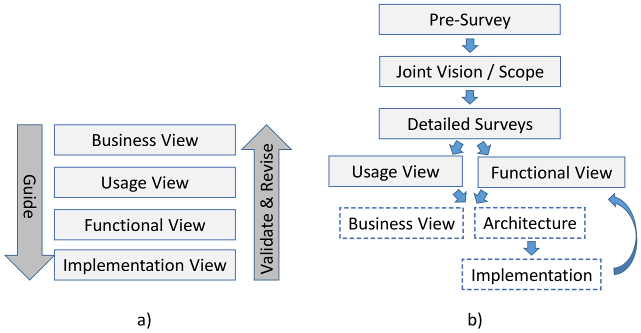


Figure 2. Steps towards requirements: a) IIRA [9] b) our approach.

For our requirements collection, we adopted the IIRA usage and functional views in Figure 2 b) as follows:

- Start with an open-minded **pre-survey**: We conducted surveys on research literature for IIoT platforms and on economically predominant IIoT platforms. As result, we identified (research-)gaps in dynamic and adaptive deployment, semantic data integration, security, and consistent customization/configurability (in the sense of variability modeling in software product lines [14]).
- Create a **joint vision**: Based on the pre-surveys, we identified further (research-)relevant topics and integrated them into a joint vision. One topic is to explore the upcoming Asset Administration Shell (AAS) [15] standard, which aims at interoperable modeling of Industry 4.0 ”assets”, i.e., products, machines or digital twins, similar to the “Smart Manufacturing Profile” concept in the US. From a software perspective, AAS can be viewed as (distributed) functional interfaces allowing for transparent remote access [16]. One aim is to identify benefits and limitations of AAS, e.g., platform interfaces can efficiently be realized by AAS. Further platform challenges target transparent mechanisms for data privacy, secure data sharing (along the lines of the International Data Spaces Association (IDSA) [17]) or the optimization of code deployment to computational resources.
- Stabilize the vision by **detailed surveys**, i.e., assure the gaps and identify supporting arguments for the vision through focused surveys. In Section IV, we will report on a survey of IIoT platforms, while an accompanying systematic literature review is out of scope here.
- Create a **usage** and a **functional** view: Using the vision as scope, elicit the requirements in terms of the two views so that they can complement each other. In our case, two teams created the views and performed a comparison of the results to assess and improve the comprehensiveness of the requirements collection. We will report our results for both views, the identified similarities and gaps, as well as our experiences in Section V.

Our results act as input for further works, e.g., the IIP-Ecosphere think tank “Business Models” uses our usage view to derive a business view for the platform and ultimately for the ecosystem. Further, the technical partners design an architecture and create an implementation of the platform based on the collected research-integrated requirements. In turn, this will act as a basis for the think thanks and demonstrators to demonstrate their results in an integrating environment.

It is important to emphasize that the collected requirements are so far based on the input of the IIP-Ecosphere partners. Ongoing work with the stakeholder community may lead to additional input and a refinement of the existing views. This input may be taken up in an iterative manner or induce requirements that document future work for the community.

IV. Survey of current IIoT Platforms

To stabilize the joint vision, we performed a survey of current IIoT platforms [3]. We selected 21 platforms, among them 9 platforms due to a competitive stakeholder analysis (including AWS IoT, PTC ThingWorx, SAP Leonardo or Siemens MindSphere), as well as 12 further platforms of particular interest to the project (such as Adamos, Deviceinsight Centersight, or Software AG Cumolocity). Based on a pre-survey and the joint vision, we defined 16 analysis dimensions including (re-usable) AI, Edge/IoT/cloud capabilities, uptake of standards, security, data privacy, ecosystem building and systematic configurability. We systematically analyzed the platforms along these dimensions based on vendor material and web pages in the period from June to August 2020.

- Although stated as relevant to almost all platforms, only 77% detail their **AI capabilities**. 48% enable customizations of the AI capabilities, while only 14% support user-defined orchestration or third party AI functionality.
- 95% of the platforms offer some form of **cloud** integration, which is frequently used to argue the scalability of the platform. Although a (mandatory) cloud integration is sometimes perceived by customers as an adoption barrier [18], only 19% offer an optional cloud integration, and only 24% support an **on-premise** installation.
- 85% support **edge** devices, but the functionality is rather diverse, ranging from data storage (67%) to customer-specific deployments (29%). 33% support AI on edge devices, however, this is currently often limited to functionality shipped with the platform. 38% of the platforms rely on container technology (usually Docker [19]) and 4 platforms (19%) utilize containers for edge deployment.
- 57% are characterized as (soft-)**real-time** capable. This roughly correlates with the edge findings. 76% employ some form of data stream or complex event processing, partially offering query languages, "low code" or "no code" environments to customize the data processing.
- Usually, the platforms offer extensive support for modern and legacy **protocols**, as well as (secure) device management. More recent approaches like OPC UA are used rarely. Most of the platforms offer some vendor-specific (REST) **interfaces**, while none of the platforms seems to uptake recent Industry 4.0 interfacing works like AAS [15] or OPC UA companion specs.

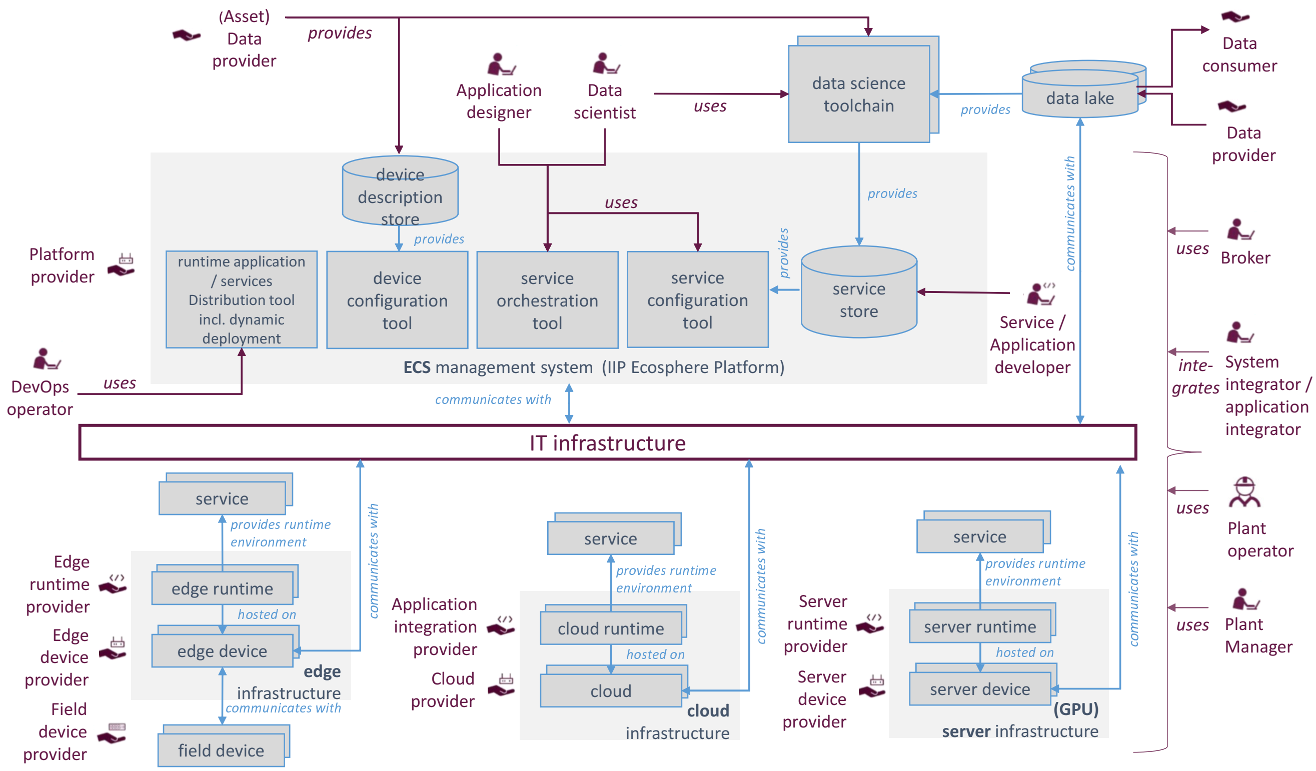


Figure 3. IIP-Ecosphere System under Consideration (SuC).

- **Security** and **data protection** seem to be essential for all surveyed platforms, in particular for cloud integration. 86% describe employed authorization measures, 71% allow limiting the data retention, but only 48% implement mechanisms to control processing of personal data.
- 81% of the platforms appear to be **customizable**, e.g, 62% of the surveyed platforms allow for external (AI) components. However, the utilized mechanisms, e.g., to ensure a consistent platform configuration, remain unclear despite the fact that customization approaches for the manufacturing/CPPS domain do exist, e.g., [20].
- Openness and customization often correlate with platform **ecosystems** [4]. Usually, the platforms focus on developer and community support, while only some platforms build up an ecosystem around their own platform (19%).

With the advent of AI, the demand for real-time processing and flexible deployments of (customer-defined) AI methods will become more prevalent. This coincides with demands for flexible offloading including edge devices for latency reduction and cloud capabilities. However, issues in standardization, openness, interoperability among platforms, security and data protection/privacy impact this trend, as well as the user's freedom of choice. We used these results to confirm the gaps/topics identified for the joint platform vision (cf. Section III) and as a scope for the subsequent requirements collection.

V. Requirements Collection

We now detail the requirements collection for the IIP-Ecosphere platform, the results and experiences that we made.

The requirements collection was conducted by two distinct teams. The input was mostly elicited through document analysis (relevant papers, standards and documents also attesting prior work, as well as the IIP-Ecosphere grant agreement as described in [21]) and interactive workshops, with stakeholders from research, industry and multipliers with backgrounds in AI (research, application), industrial production, factory construction, device supply, software engineering in individual cases also with experience in requirements collection. The workshops introduced the vision/scope based on the findings from our review of selected IIoT platforms [3], gave an explanation of the respective approach to requirements collection, and typically led to many interactions and lively discussions. The results of these interactions were scripted, summarized in a document and reviewed by the participants.

A. Usage view

An IIRA usage view consists of an initial architecture, the "System under Consideration" (SuC), a definition of the used *entities*, the interacting *roles*, as well as *activities* on when/how the *roles* interact with the *entities*. Activities can be specified in terms of a template with a sequence enumerating the interactions (similar to use cases). The edge configuration usage view [10] provided a good basis for our work, but it does not cover all relevant topics for IIP-Ecosphere, particularly deployment to on-premise/cloud resources or AI activities. For these topics we organized focus workshops, where the participants discussed existing/new roles, entities and activities. Finally, we integrated the collected information into the usage view in [22] with the following results:

- A significantly **extended SuC**, as illustrated in Figure 3. The entities are colored in gray, the roles in purple, and the interactions between the SuC and roles are drawn as purple arrows. Areas indicating systems of the IIP-Ecosphere platform, such as the ECS (Edge, Cloud and Server) management system or the ECS devices/infrastructures, are shaded in light gray, covering all entities that belong to the indicated systems. Data flows between entities are drawn as light blue arrows. The underlying IT Infrastructure of the SuC, connecting the ECS devices/infrastructures with the IIP-Ecosphere platform is depicted as a white box with a purple outline, as it is neither entity nor role in the SuC.
- As in [10], **field devices**, such as sensors, actuators or (parts of) manufacturing machines are only connected to edge devices. In contrast to [10] where entire applications are deployed to edge devices, the IIP-Ecosphere platform shall focus on **applications** that are composed of orchestrated services that can be distributed across ECS devices.
- Applications and services are specified in a **configuration model** (not shown in the SuC) that allows for creating the runtimes of applications and services for the deployment into containers and for determining optimizations or incompatibilities in the orchestration of services. Means for validating the consistency of the model shall be provided and integrated with the user interactions. Services can be added to/updated in a service store as needed.
- Each ECS device runs a **device abstraction** (ECS runtime) being responsible for executing the services/containers and for reporting their runtime measurements. The platform decides about the deployment, dynamically composes service containers for the target resource, and adapts the distribution runtime. For developing applications and services, the stakeholders indicated that **pre-deployment testing or simulation** of new or updated services is highly desirable.
- All resources and services provide a **self-description** in terms of an AAS [15] information model and communicate only via **Industry 4.0 protocols** to foster interoperability, but also to explore limitations.
- The elicited **AI and data science activities** form an orthogonal space. The stakeholders contributed activities for data exploration, AI model design/testing and the integration of external data science toolchains. In the context of developing new AI services, the stakeholders also expressed the need for the ability to use **pre-deployment testing** of new and updated AI services/models, which are key elements in the continuous development and operation of machine learning (MLOPs, [23]) of applications and services within the IIP-Ecosphere platform.
- AI methods typically operate on models that may incur **data protection, IPR or further legal issues**. Some issues may be addressed by limiting deployment targets, e.g., through the exclusion of certain cloud spaces. Issues, such as data protection could be addressed by modifying the data close to the source, e.g., through anonymization or pseudonymization. However, the impact of such modifications on AI and further data processing is currently unclear and shall be researched using the platform.

In summary, as illustrated in Figure 4, the usage view for the IIP-Ecosphere platform consists of 18 entities, 19 roles, as well as 43 deployment and 24 AI activities (as opposed to [10] with 5 entities, 7 roles and 27 activities). Although the joint vision focused the discussions to a certain system scope and one might expect that this also limited the contributions of the participants, several results were creative and surprising to us. We will detail some examples below.

For example, the data scientists argued about alternatives on how to integrate a data science toolchain into the platform. An initial idea on a tight integration was rejected in favor of a loose integration, i.e., the toolchain shall remain flexible so that a data scientist may use his/her favorite tools while the platform supports the process in terms of provisioned resources, access to experimental and life data, as well as available (AI) services. This insight let to 10 activities specifically on 'activities for model training and evaluation', which cover many of the 9 stages of the machine learning workflow in [24], some are exemplified in Figure 4. Furthermore, the aspect of continuous delivery of AI services following MLOps concepts was emphasized by the AI experts. This led to activities like 'continuous application of AI models on new data' or 're-calibration of AI model parameters' shown in Figure 4. Moreover, the stakeholders requested capabilities to measure the accuracy of productive or simulated AI models to observe the quality of predictions and to early on detect model-drift, e.g., the loss of accuracy due to slow changes in the application environment. This induced 5 activities on 'analysis and prediction of performance and accuracy'.

During the workshops, the industry experts expressed the necessity to provide simulation- and testing-capabilities to allow for simulation-driven development of applications and services. A key approach that was identified here is the development, simulation and monitoring of applications and services but also of ECS devices, based on digital twins. Similar to MLOps, these activities target DevOps [25] capabilities, for example, allowing for pre-deployment testing of any application, service or ECS device. We represented this desire in terms of 8 'activities for (distributed) applications'.

As described, the presented results focus on service deployment and AI activities. Initially, we planned to explore also further topics, such as data sharing or data privacy. However, we also experienced that interactively creating a usage view is a significant effort. Thus, in particular for a research-integrating usage view, it is important to focus on the most important topics first. It is noteworthy that, as outlined above, a

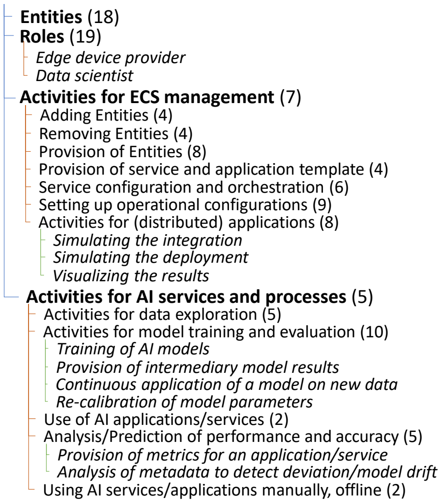

Entities (18)
Roles (19)
Edge device provider
Data scientist
Activities for ECS management (7)
Adding Entities (4)
Removing Entities (4)
Provision of Entities (8)
Provision of service and application template (4)
Service configuration and orchestration (6)
Setting up operational configurations (9)
Activities for (distributed) applications (8)
Simulating the integration
Simulating the deployment
Visualizing the results
Activities for AI services and processes (5)
Activities for data exploration (5)
Activities for model training and evaluation (10)
Training of AI models
Provision of intermediary model results
Continuous application of a model on new data
Re-calibration of model parameters
Use of AI applications/services (2)
Analysis/Prediction of performance and accuracy (5)
Provision of metrics for an application/service
Analysis of metadata to detect deviation/model drift
Using AI services/applications manually, offline (2)

Figure 4. Usage view overview with sections, (number of) contained roles / activities, and example roles / activities (in italics).

significant number of capabilities needed for the platform and subsequent activities that enable theses, were identified by the close communication and interaction between the AI experts, who were focused on research aspects of the IIP-Ecosphere platform and the Industry experts with their focus lying on the technical aspects of the platform. Hence, the integration of the scientific and the industrial view in these two groups yielded deep and very valuable results for the elicitation and formulation of the IIP-Ecosphere platform requirements. It is also worth to mention that during the cooperation of both expert groups a significant amount of "mutual understanding" was established, clarifying for both groups of experts specific vocabulary, views and motivations present in the two groups and thus enabling a productive discourse and collaboration.

B. Functional View

A second team collected required platform functions, as well as quality requirements, initially independent of the usage view activities. We performed a requirements collection combining the recording of ideas and desires mentioned in specific discussions with structured approaches, such as interviews or questionnaires. In particular, within the consortium we conducted a requirements questionnaire with 8 questions driven by the joint vision, ranging from a summary of the planned applications over envisioned AI methods, relevant data protection measures up to imaginable run-time changes for self-adaptation.

We documented the requirements in terms of phrase templates [26], i.e., based on a *template sentence* indicating the acting role, the required functionality and the prioritization of the requirement. All requirements were reviewed by the stakeholders, categorized, prioritized (must/should/can) - with more emphasis on scientific goals, required basis functionality and the grant agreement of IIP-Ecosphere - annotated with their source, and, if needed, detailed by an explaining text.

Ultimately, we compared the usage view with the functional view. While more than two third of the topics do occur in both views, we also identified differences. We found entire topics in the usage view that the stakeholders did not touch at all in the requirements discussions, e.g., the pre-deployment simulation. Moreover, the interaction steps in the usage view activities pointed us to details that were not covered by the requirements, e.g., how IIoT applications shall be managed. For the opposite direction, we found, e.g., that run-time adaptation was treated in the questionnaire as an interesting feature, which was also viewed with caution, i.e., some stakeholders requested explicit human approvals rather than autonomously changing a deployment (or, similarly, a re-trained AI model).

In summary, we elicited 141 top-level and 179 refining sub-requirements as documented in [21]. 17% of all requirements were added due to the comparison of the two requirements views. Figure 5 illustrates the requirements categories that we identified along with the number of contained top-level and sub-requirements, as well as selected example requirements (without explaining text). About 16% of all requirements target quality, among them 7 on data processing, e.g., on the expected data frequency and volume. The largest group of quality requirements focuses on security and data protection. The main sources are the think tanks (41%), the IIP-Ecosphere demonstrators (20%) and the grant agreement (12%). Further sources are standards, the platform survey from Section IV and the comparison with the usage view.

Although our set of functional requirements is rather detailed, we are aware that it is potentially incomplete. On the one side, the IIP-Ecosphere platform forms a data-driven system and, as mentioned above, requirements in such systems are known to be volatile, unclear or incomprehensive due to the explorative nature of data science and AI processes [7]. On the other side, resource limitations in this research project prevented us from conducting further/deeper usage view and

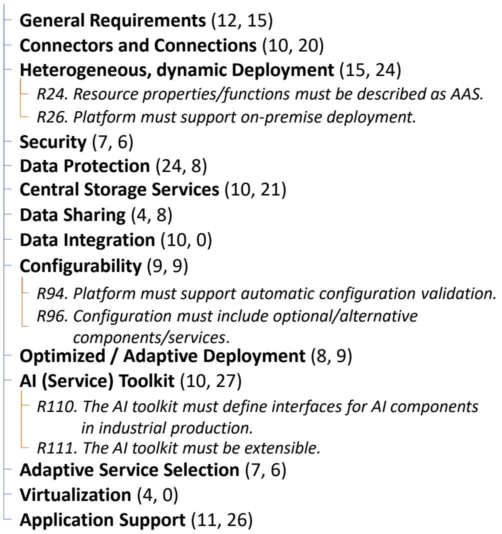

General Requirements (12, 15)
Connectors and Connections (10, 20)
Heterogeneous, dynamic Deployment (15, 24)
R24. Resource properties/functions must be described as AAS.
R26. Platform must support on-premise deployment.
Security (7, 6)
Data Protection (24, 8)
Central Storage Services (10, 21)
Data Sharing (4, 8)
Data Integration (10, 0)
Configurability (9, 9)
R94. Platform must support automatic configuration validation.
R96. Configuration must include optional/alternative components/services.
Optimized / Adaptive Deployment (8, 9)
AI (Service) Toolkit (10, 27)
R110. The AI toolkit must define interfaces for AI components in industrial production.
R111. The AI toolkit must be extensible.
Adaptive Service Selection (7, 6)
Virtualization (4, 0)
Application Support (11, 26)

Figure 5. Requirements overview with categorization topics, number of top-/sub-requiements and selected example requirements (in italics).

requirements workshops, e.g., on data sharing. Thus, in both views, particular topics/requirements could still be missing. However, the described results allow for discussing the effects of an interactive multi-view requirements collection. Moreover, the collected requirements are sufficient to incrementally realize the IIP-Ecosphere platform and, if needed, to elicit missing details during platform realization in agile manner.

C. Experiences

The two teams recorded the experience that they made during the elicitation that we briefly summarize in this section.

Both views are based on templates (activities and phrases) that seemed to give a certain form of guidance to the participants. In comparison, the usage view workshops seemed to have allowed for more creativity, i.e., the participants tended to talk more freely about desirable system interactions or known limitations. The workshops also allowed for more interactions, such as agreements among the participants as stated in Section V-A. However, this impression may be biased by the first workshop, where the participants were asked to name any missing topic. Some of the topics were taken up by later workshops where then mostly 'experts' participated, which allowed also more quiet persons to participate more actively.

The discussions on the functional view were focused on capabilities for developing applications on top of the platform and, thus, more technical. Here the stakeholders did not interact so lifely, which might be one reason why we missed application-related topics. Moreover, we noticed that different persons participated in the usage view and in the requirements workshops. In the latter, the participants seemed to have more technical background, probably as the workshop invitation asked for contributions to functional requirements for the platform. Yet, the functional view also revealed interesting aspects, e.g., as mentioned above, (different levels) of adaptation approvals or the need for explainable adaptation decision making. This may be biased by our questionnaire, where we explicitly asked for these topics and the participants could overthink their answers or discuss them with their team.

We also experienced that research-integrated requirements do not come for free. Questions like “Why do we need this?” or “Isn't that too risky?” for certain research topics, e.g., for self-adaptive capabilities, arise and must even be defended against more practical/industrial requirements.

VI. Related Work

We now review briefly work related to our core topics, i.e., surveys and requirements collections for IoT platforms.

Various comparisons and *surveys of IoT platforms* are published. As stated by Mijuskovic at al. [27], this is often done for a specific set of criteria lacking a sound comparison framework. Moreover, comparisons are typically based on a selection of platforms as the market is rather dynamic and encompasses hundreds of platform vendors [28], i.e., typical numbers of platforms are 11 in [29], 13 in [30], 20 in [31], 24 in [28] or 26 in [32]. Often, such surveys are based on vendor material, while in [28] interviews with vendor representatives were used. Regarding comparison criteria, the topics are frequently device management [28] [30] [31], fog/edge/cloud deployment [29] [30] [32], connectivity/protocols [28] [29] [30] [31] [32], security [28] [29] [31], data management [28] [29] [30] [32], data analytics [28] [31] [32], visualizations/UI [28] [31] [32], application development [28] [29] [30] [32], system/service management [30] [32], or licensing/payment [29] [30]. In contrast, in our survey we also analyzed AI capabilities, edge usage, ecosystem building, data protection and consistent configurability and used that survey as a basis for our research-integrated requirements approach.

There is also a body of work on requirements management for IoT/CPPS platforms or ecosystems, e.g., [11]–[13]. However, we do not aim at proposing a completely new requirements approach rather than performing a *requirements collection* for IoT/CPPS systems while balancing scientific and practical interests. Many technical publications motivate their work with a focused set of requirements, while overview work with collections of platform requirements is less common. Among those, we identified the following topics for functional requirements: device/resource/distribution management [33] with heterogeneous deployment [27], communication/networking [27] [34], data (base) management [27] [33] [34], data processing [27] [33], data analytics including AI [27], monitoring [27] service management [34], security/privacy [27] [33] [35] [36], or visualization [27] [33]. Moreover, we found non-functional topics, such as scalability [33] [35], performance [27] [33] [34], standardization/interoperability [33]–[35], development support [27] [33] and even self-adaptation [33]. In contrast to our work, the cited publications typically focus on a single (functional/quality requirements) view, i.e., do neither take the scientific site nor interactions of multiple views into account.

VII. Conclusion

IIoT, CPPS and Industry 4.0 platforms form the software foundation of complex manufacturing systems. The introduction of Artificial Intelligence into such systems will enable new opportunities, but further increases the complexity and causes challenges for all involved disciplines. Eliciting requirements for future platforms is not trivial, in particular if scientific and industrial interests must be balanced and integrated.

In this paper, we reported on a pragmatic approach to perform a requirements collection of a platform that shall demonstrate research approaches in an upcoming Industry 4.0 ecosystem. Driven by pre-studies, we used a joint vision as scope for the further steps, a surveying phase and a requirements collection phase. For the surveys, we reported on an overview of 21 recent Industry 4.0 platforms that helped us to identify gaps and to stabilize the vision. The vision then guided an intensive requirements collection for two perspectives, a usage view and a functional/quality view, which, in summary, integrates research and industrial interests. We discussed our experiences with such a requirements elicitation, in particular that different views can successfully complement each other.

The joint work, in particular on documenting the results, helped the involved partners to clarify and synchronize their view on the system to be built, e.g., the terminology or the needed components. Based on these experiences, first companies in IIP-Ecosphere started applying such a requirements elicitation approach as part of their own activities. Moreover, concepts and ideas on service-based Industry 4.0 platforms as outlined in the usage view [22] were fed back to the originating Labs Network Industrie 4.0 (LNI 4.0) organization and at the time of writing are being integrated into a revised version of [10].

Current and future work is on developing the IIP-Ecosphere platform based on both requirements documents, including incremental architecture design or integration of research and industrial approaches. At the time of writing, about a third of all platform requirements have been realized and validated. We also plan for evaluations of the platform approaches in terms of industrial use cases.

ACKNOWLEDGMENT

The IIP-Ecosphere project (https://www.iip-ecosphere.de/) is partially supported by the German Ministry of Economics and Energy (BMWi) under the grant numbers 01MK20006C and 01MK20006D. The authors would like to thank the partners who participated in and contributed to our requirements workshops.

REFERENCES

[1] J. Delsing, “Local Cloud Internet of Things Automation,” *IEEE Industrial Electronics Magazine*, pp. 8–21, 2017.

[2] R. Wöstmann *et al.*, “Conception of a Reference Architecture for Machine Learning in the Process Industry,” in *Intl. Conference on Big Data*, 2020, pp. 1726–1735.

[3] C. Sauer, H. Eichelberger, A. S. Ahmadian, A. Dewes, and J. Jürjens, “Current Industry 4.0 Platforms - An Overview,” retrieved 03-2022. [Online]. Available: https://www.iip-ecosphere.eu/wp-content/uploads/2021/02/IIP-2020_001-en.pdf

[4] K. Schmid, H. Eichelberger, and C. Kröher, “Domain-Oriented Customization of Service Platforms: Combining Product Line Engineering and Service-Oriented Computing,” *Journal of Universal Computing*, vol. 19, no. 2, pp. 233–253, 2013.

[5] “Website of the Organization for the Advancement of Structured Information Standards on MQTT,” retrieved 03-2022. [Online]. Available: https://mqtt.org/

[6] “Website of the OPC UA foundation,” retrieved 03-2022. [Online]. Available: https://opcfoundation.org/

[7] O. Hummel, H. Eichelberger, A. Giloj, D. Werle, and K. Schmid, “A Collection of Software Engineering Challenges for Big Data System Development,” in *Proceedings Euromicro Conference on Software Engineering and Advanced Applications (SEAA’18)*, 2018, pp. 362–369.

[8] H. Bedenbender *et al.*, “Industrie 4.0 Plug-and-Produce for Adaptable Factories: Example Use Case Definition, Models, and Implementation,” 2017. [Online]. Available: https://www.zvei.org/fileadmin/user_upload/Presse_und_Medien/Publikationen/2017/Juni/Industrie_4.0_Plug_and_produce/Industrie-4.0-_Plug-and-Produce-zvei.pdf

[9] Industrial Internet Consortium, “The Industrial Internet Reference Architecture,” retrieved 03-2022. [Online]. Available: https://www.iiconsortium.org/pdf/IIRA-v1.9.pdf

[10] A. Graf, U. Löwen, M. Rentschler, B. Vojanec, and T. Roehrl, “LNI 4.0 Testbed Edge Configuration - Usage View,” retrieved 03-2022. [Online]. Available: https://www.plattform-i40.de/PI40/Redaktion/EN/Downloads/Publikation/LNI4.0-Testbed-Edge-Configuration_UsageViewEN.html

[11] P. Loucopoulos, E. Kavakli, and N. Chechina, “Requirements Engineering for Cyber Physical Production Systems,” in *Advanced Information Systems Engineering (CAiSE’19)*, 2019, pp. 276–291.

[12] S. Kaleem *et al.*, “A Review on Requirements Engineering for Internet of Things (IoT) Applications,” in *HCT Information Technology Trends (ITT’19)*, 2019, pp. 269–275.

[13] D. Silva, T. G. Gonçalves, and A. R. C. da Rocha, “A Requirements Engineering Process for IoT Systems,” in *Brazilian Symposium on Software Quality (SBQS’19)*, 2019, pp. 204–209.

[14] F. van der Linden, K. Schmid, and E. Rommes, *Software Product Lines in Action*. Springer, 2007.

[15] S. Bader *et al.*, “Details of the Asset Administration Shell,” retrieved 03-2022. [Online]. Available: https://www.plattform-i40.de/PI40/Redaktion/DE/Downloads/Publikation/Details_of_the_Asset_Administration_Shell_Part1_V3.html

[16] M. G. Casado and H. Eichelberger, “Industry 4.0 Resource Monitoring - Experiences with Micrometer and Asset Administration Shells,” in *Symposium on Software Performance (SSP ’21)*, 2021.

[17] “Website of the International Data Space Association,” retrieved 03-2022. [Online]. Available: https://internationaldataspaces.org/idsa-industrial-community/

[18] C. Esposito, A. Castiglione, B. Martini, and K. R. Choo, “Cloud manufacturing: Security, privacy, and forensic concerns,” *IEEE Cloud Comput.*, vol. 3, no. 4, pp. 16–22, 2016.

[19] “Website of Docker Inc.” retrieved 03-2022. [Online]. Available: https://www.docker.com/

[20] K. Meixner, R. Rabiser, and S. Biffl, “Towards Modeling Variability of Products, Processes and Resources in Cyber-Physical Production Systems Engineering,” in *Intl. Systems and Software Product Line Conference (SPLC’19) - Vol. B*, 2019, pp. 49–56.

[21] H. Eichelberger *et al.*, “IIP-Ecosphere Platform Requirements (Functional and Quality View),” retrieved 03-2022. [Online]. Available: https://www.iip-ecosphere.eu/wp-content/uploads/2021/03/IIP-2021_002-en.pdf

[22] C. Sauer, H. Stichweh, and H. Eichelberger, “IIP-Ecosphere Platform Requirements (Usage View),” retrieved 03-2022. [Online]. Available: https://www.iip-ecosphere.eu/ergebnisse/

[23] S. Mäkinen, H. Skogström, E. Laaksonen, and T. Mikkonen, “Who needs mlops: What data scientists seek to accomplish and how can mlops help?” *arXiv preprint arXiv:2103.08942*, 2021.

[24] S. Amershi *et al.*, “Software engineering for machine learning: A case study,” in *International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP)*, 2019, pp. 291–300.

[25] R. Jabbari, N. bin Ali, K. Petersen, and B. Tanveer, “What is DevOps? A systematic mapping study on definitions and practices,” in *Scientific Workshop Proceedings of XP2016*, 2016, pp. 1–11.

[26] K. Pohl and C. Rupp, *Requirements Engineering Fundamentals*. Rocky Nook, 2011.

[27] A. Mijuskovic, I. Ullah, R. Bemthuis, N. Meratnia, and P. Havinga, “Comparing Apples and Oranges in IoT context: A deep dive into methods for comparing IoT platforms,” *IEEE Internet of Things Journal*, pp. 1797–1816, 2020.

[28] T. Krause *et al.*, “IT-Plattformen für das Internet der Dinge (IoT),” 2017. [Online]. Available: http://publica.fraunhofer.de/documents/N-470532.html

[29] P. Agarwal and M. Alam, “Investigating IoT Middleware Platforms for Smart Application Development,” in *Smart Cities - Opportunities and Challenges*, ser. Lect. Notes Civ. Eng. Springer, 2020, vol. 58.

[30] M. Asemani, F. Abdollahei, and F. Jabbari, “Understanding IoT Platforms: Towards a comprehensive definition and main characteristic description,” in *Intl. Conference on Web Research (ICWR’19)*, 2019, pp. 172–177.

[31] H. Hejazi, H. Rajab, T. Cinkler, and L. Lengyel, “Survey of platforms for massive IoT,” in *Future IoT Technologies*, 2018, pp. 1–8.

[32] P. P. Ray, “A survey of IoT cloud platforms,” *Future Computing and Informatics Journal*, vol. 1, no. 1, pp. 35 – 46, 2016.

[33] P. Agarwal and M. Alam, “Open Service Platforms for IoT,” in *Internet of Things (IoT): Concepts and Applications*, 2020.

[34] E. Patti and A. Acquaviva, “IoT platform for Smart Cities: requirements and implementation case studies,” in *Research and Technologies for Society and Industry Leveraging a better tomorrow (RTSI’16)*, 2016.

[35] P. Bhuyan and A. Ray, “IoT Service Platform,” in *Interoperability in IoT for Smart Systems*. CRC Press, 2021, pp. 91–97.

[36] S. Oh and Y. Kim, “Security Requirements Analysis for the IoT,” in *International Conference on Platform Technology and Service (PlatCon’17)*, 2017, pp. 1–6.

Software Architecture Evolution of a Physical Security Information Management System

Oğuzhan Özçelik
ASELSAN A.Ş.
Ankara, Turkey
e-mail: oozcelik@aselsan.com.tr

Halit Oğuztüzün
Department of Computer Engineering
Middle East Technical University
Ankara, Turkey
e-mail: oguztuzn@ceng.metu.edu.tr

***Abstract*—The planned reuse mentality of software product line engineering makes it possible to deliver similar products within a short amount of time. Physical Security Information Management (PSIM) system customizations tend to be similar to each other with fundamental requirements being more or less the same in different projects. One of the most common difference in these projects is the used sensors. Some sensors could be integrated into the PSIM system easily if they are compatible with a standard communication interface such as Open Network Video Interface Forum (ONVIF) protocols. But sensors that use a special communication interface need to be integrated one by one. A PSIM system is always expected to integrate additional sensors to its catalog. In order to do this easily, the parts that need to be developed to integrate a sensor must be segregated and developed individually for each sensor. In this work, we aim to segregate the sensor integration of a PSIM system and compare the old and new generations of the architecture qualitatively, based on architecture models.**

Keywords-Physical Security Information Management Systems; Physical Protection Systems; Software Product Line Engineering.

I. INTRODUCTION

A Physical Security Information Management (PSIM) system integrates diverse independent physical security applications and devices. Applications such as building management or network video recorder systems, and devices such as security cameras, access control systems, radars and plate recognition systems are used interconnectedly. It is designed to ensure the physical security of a facility, city or an open field, while providing a complete user interface to the security operators to monitor and control them.

The subject PSIM system of this work is called SecureX, which is not the name of the actual system but a placeholder used for confidentiality reasons. SecureX is a PSIM system that aims to satisfy the needs mentioned above and also to provide an easy integration environment for new sensors and applications. The ever-increasing number of such new systems and different security needs of different customers drove SecureX team to embrace a software product line engineering approach in order to reduce the response time to reply to the customers' demands. These demands vary from practical improvements to integrating a new sensor or security application as a feature to the system. SecureX is deployed with the full feature set and only at runtime these features are reduced to the ones required by a given customer, using different configuration files. Any new integration required by a customer needs to be developed as a feature in SecureX. Afterwards, a new SecureX build must be generated. Following every new integration, a new testing process takes place and because the previously integrated system might not always be available for testing, it must be guaranteed that the new integration will not affect the other integrations. In this work, a new method for integrating such new systems while reducing the number of required tests is proposed.

The rest of the paper is structured as follows. In Section II, several PSIM products and their specializations are mentioned. Also, we briefly explain how they approach the sensor integration problem and why that is not enough in the case of SecureX. In Section III, the general architecture of SecureX is described and the point where sensor integration takes place is shown. In Section IV, this sensor integration point is described in more detail. In Section V, the problems with the current architecture are explained and in Section VI, a new architecture that solves those problems is described. In Section VII, the benefits of the new architecture are shown by explaining how it solves each problem of the current design.

II. RELATED WORKS

There are several companies offering PSIM products. Although they provide every essential feature of a PSIM system, they may have different specializations. Genetec [1] provides a video analytics tool to detect intrusions. Milestone [2] uses its own Network Video Recorder (NVR) systems and provides an easy to use video management system. Nedap [3] is specialized in access control systems. However, not many details exist on how they work internally. These products integrate some general communication standards like ONVIF [4] protocols and also release Software Development Kits (SDK) and expect sensor manufacturers or customers to integrate their custom subsystems into the PSIM system as well. This way, they accelerate sensor integration by including numerous 3rd parties. While developing an SDK to use in integrations is a feasible solution, in the SecureX's case, the main objective is developing an architecture that can simplify not only the sensor integrations, but also the component selection to deploy because different customers have different requirements. Another requirement is that the new architecture will be able to remove the update and test overhead. A software product line architecture would be suitable to accomplish this goal.

Recently, Tekinerdogan et al. [5] described how a PSIM system should be designed with software product line engineering methodologies to reduce the cost of development by improving reuse. The present work describes a step in architectural evolution toward a product line architecture.

III. ARCHITECTURE OF SECUREX

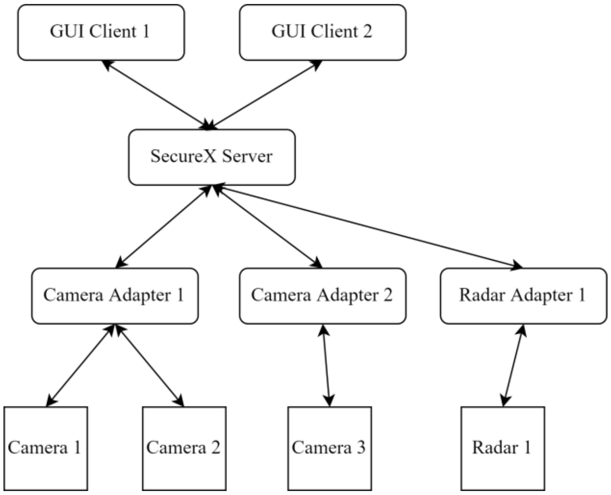


Figure 1. Deployment model of SecureX

SecureX has a distributed architecture which can be seen in Figure 1. Graphical User Interface (GUI) Clients of SecureX are installed on the computers of security officers, enabling them to monitor the entire security infrastructure of the area under surveillance. These clients are connected to the SecureX Server application which handles the communication between SecureX components. The server is also responsible for recording events, including detections and errors sent from adapter components to the central database. SecureX could also be installed in a hierarchical fashion in which higher servers could also control and monitor the security components that are connected to the servers under them. Under the SecureX Server, there are adapter applications for each sensor group such as camera, radar, plate recognition systems, access control systems, etc. These adapters are the points where the SecureX environment makes its connections to the outer world.

When a user wants to perform some action with a sensor, after pressing a button in the SecureX GUI Client, a message will be sent to the SecureX Server. Then, the server delegates this message to adapters and other servers that are hierarchically under that server. The message arrives at the sensor's adapter and, according to the Interface Control Document (ICD) used in its integration, a message would be sent to the sensor to perform the desired action. Events and detections caught by the sensors would follow the reverse route and find their way to the SecureX GUI Clients.

IV. EXISTING ADAPTER ARCHITECTURE

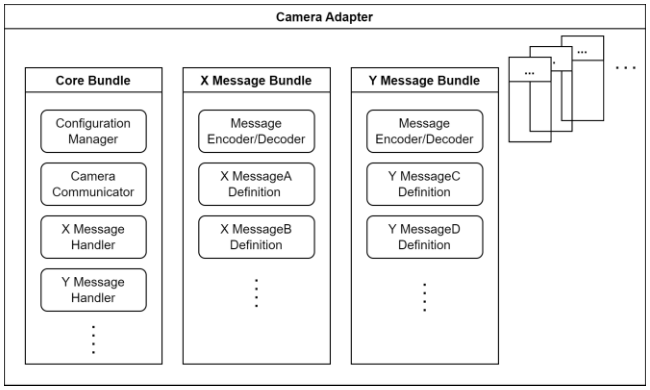


Figure 2. Simplified Camera Adapter model in the existing architecture

SecureX is developed using the Open Services Gateway Initiative (OSGi) framework, which is a Java [6] framework to develop modular software [7]. These modules are called "bundles" and the framework could install, uninstall and update them, even at runtime [8]. The bundles to be installed and their start levels are stated in bundle configuration files. A few of these bundles can be seen in Figure 2. SecureX uses this framework to take advantage of its service architecture. We use the Camera Adapter application to describe the adapter architecture, but all adapter applications of SecureX are quite similar.

The Camera Adapter application consists of many OSGi bundles whose purposes vary from providing network connection interfaces or utility tools, to message definition of sensors. These message definition bundles contain the methods for encoding and decoding messages to and from the sensor. Generally, the message formats for each sensor are different. They have different data types, header types, checksum calculation methods, big or little endian formats. Some sensors accept JSON formatted string messages and some require encoding messages in a certain length byte arrays and sending them. Information about how to communicate with a sensor is given in its ICD. A message bundle is basically an implementation of the related ICD.

The *Configuration Manager* class in the *Core* bundle is mainly responsible for opening a Transmission Control Protocol (TCP) port to accept incoming server connections and initializing the *Message Handlers*. Each sensor's type, model, unique identifier key and required information about establishing a connection to it is written in a configuration XML file. The *Configuration Manager* constantly iterates over these files, creating a *Camera Communicator* and a specific *Message Handler* for every new or updated file. Messages are received by the TCP server and forwarded from there to the *Camera Communicator* and lastly to the sensor's *Message Handler*.

A *Camera Communicator*, which extends from the *Sensor Communicator* class as in every other sensor family, is the class where the processing of messages that came from the server starts. It handles generic messages or preprocesses them before the messages arrive at the *Message Handler*. When a message is received from the server, it is added to the message buffer of every active *Camera Communicator* in that adapter. *Camera Communicators* take this message and decide if this message is meant for their sensor. To do this, they use the sensor identifiers in the messages. If the identifier is the same with the *Message Handler* they have, the message gets processed as will be explained in the subsequent paragraph, otherwise it is discarded.

The processing of the messages starts at the *Camera Communicator* level. Some messages are not specific to different sensor integrations and can be handled at the *Camera Communicator* level. Alternatively, some messages require a preprocessing step such as transforming some variables before they get forwarded to the *Message Handler*. After the initial processing is done, the *Camera Communicator* sends the message to the *Message Handler*.

The *Message Handler* is where the connection to the sensor is established using the protocol the sensor uses, which could be TCP, User Datagram Protocol (UDP), WebSocket, serial port, (Representational State Transfer) REST or any other network connection method that is stated in its ICD. The *Message Handler* knows how the connection should be established and how the incoming and outgoing messages should be processed. It receives the incoming message from the communicator and sends necessary commands to the sensor. The *Message Handler* needs a utility bundle to do the message conversions. When it needs to encode/decode messages to/from the sensor, it uses the Message bundle of that sensor that contains the message types, formats, checksum methods and the information of exactly how a message should be generated. After a message is generated, the *Message Handler* sends it to the sensor using the connection interface.

V. The Integration Problem

When the adapter starts, the *StartLevelEventDispatcher* thread in the OSGi framework initializes all bundles that are marked for auto-start in the bundle configuration file. In Figure 3, initialization of the *Core* bundle is shown. The *Core* bundle is the one that starts the main Camera Adapter process with its thread "*ConfigurationMonitor*". In the initialization of the *Core* bundle, a single *Configuration Manager* instance gets created. The *Configuration Manager* then opens a port to listen to incoming SecureX Server connections. After that, it starts a thread that periodically checks sensor configuration files to find new or updated configurations. If there is such a file, then the *Configuration Manager* creates a *Camera Communicator* and the *Message Handler* for that sensor. In the existing architecture, in order to create a *Message Handler* instance, the *Configuration Manager* has to know which *Message Handler* needs to be used for which sensor configuration. In the configuration file, the identifier of the correct *Message Handler* is given and the *Configuration Manager* uses that identifier to construct the *Message Handler*. But these *Message Handler* classes are inside the *Core* bundle and the *Configuration Manager* has a class dependency for them. This is the root problem in the current architecture.

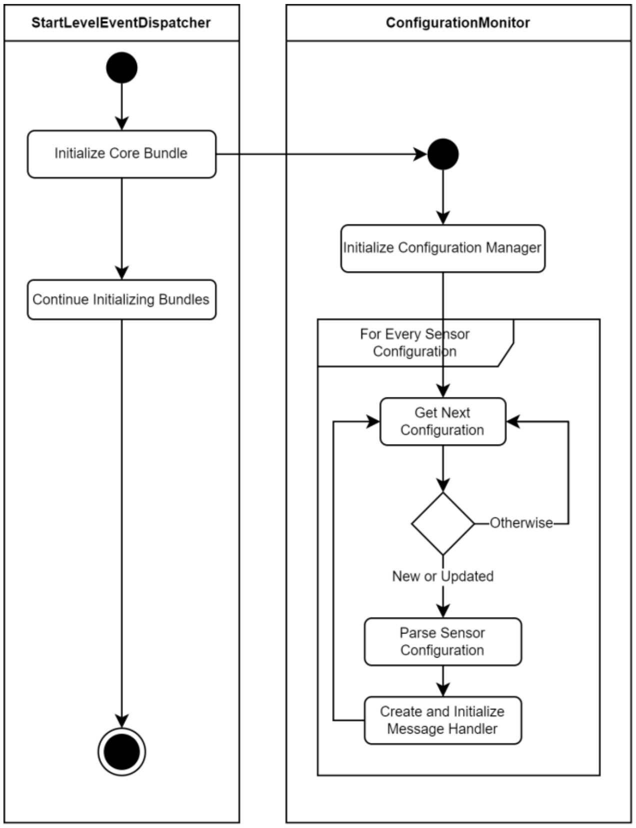

StartLevelEventDispatcher
ConfigurationMonitor
Initialize Core Bundle
Continue Initializing Bundles
Initialize Configuration Manager
For Every Sensor Configuration
Get Next Configuration
Otherwise
New or Updated
Parse Sensor Configuration
Create and Initialize Message Handler

Figure 3. Message Handler initialization in the existing architecture

A. Difficulties with the Existing Architecture

In order to carry out a new sensor integration, the message definition bundle has to be added in the Camera Adapter product file and its *Message Handler* has to be included in the *Core* bundle. The *Configuration Manager* class needs to know with which configuration identifier the new *Message Handler* should be constructed beforehand, hence the dependency. Because of this design, integrating or updating the integration of a sensor requires updating the *Core* bundle in the adapter. The components in the *Core* bundle, such as *Configuration Manager* and *Camera Communicator*, are used in every *Message Handler* and need to be compatible with all of them too. Therefore, any change in those components in the integration of a sensor could affect the already integrated sensors and cause them not to function as intended. Alarms detected by the sensor might start not to be forwarded to the server or changing the orientation of the sensor becomes difficult because of a change in some movement speed calculations.

In the current design, to update an already deployed system, a complete new build needs to be generated and tested. But testing of the previous sensor integrations are not always easy or even possible. These sensors could be produced in very limited numbers and they can only be found in the customer's facilities, working with the previous SecureX version. The location of these facilities might be difficult to access too and trips to these locations are not only costly, but sometimes, also dangerous. Because these sensors are almost always used in closed networks, the only way to test them is by going to these facilities, increasing the test cost. Also, customers would not want testers to separate these sensors from the PSIM system to test with the new version, creating a window of vulnerability.

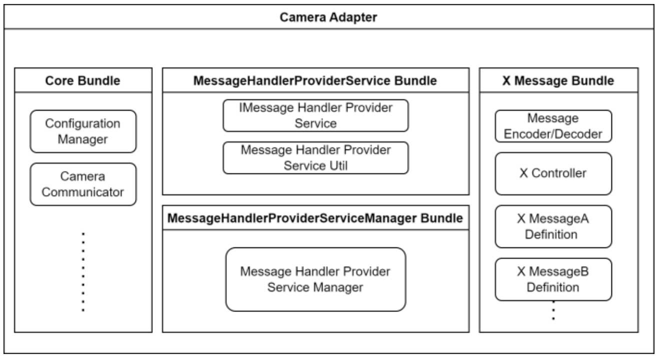

Camera Adapter
Core Bundle
Configuration Manager
Camera Communicator
MessageHandlerProviderService Bundle
IMessage Handler Provider Service
Message Handler Provider Service Util
MessageHandlerProviderServiceManager Bundle
Message Handler Provider Service Manager
X Message Bundle
Message Encoder/Decoder
X Controller
X MessageA Definition
X MessageB Definition

Figure 4. Simplified Camera Adapter model in the new architecture

Even if the tests are somehow completed, the update procedure has its own problems. To quickly update systems used in remote locations with little to no network access, or used in thousands of mobile locations without stable internet access, the update size must be minimal. But, with the current architecture, the whole adapter build needs to be updated, rather than just a couple of bundles.

Also, to catch up with new and updated sensors or security systems, 3rd party companies are employed for integrations. But this process is done through signing a Non-Disclosure Agreement (NDA) and sharing huge parts of the adapter code with them to be used to integrate the sensors. Any one of them could expose the code at any point and this indeed is a security vulnerability.

Because of these reasons, there is a need for an architecture that ensures that the new integrations will not affect the existing ones. The main problem with the current design is, for every new integration, it has a need to update the *Core* bundle. The reason for that is the *Configuration Manager* class needs to know all available *Message Handlers* and for what kind of sensor they need to be used beforehand via class dependencies. In the new architecture, this problem is targeted with the aim to reduce testing overhead, reducing the amount of code that is shared with 3rd parties and also enables updating the deployed systems with very low data.

VI. New Adapter Architecture

To solve the problems with the existing architecture, a new adapter architecture shown in Figure 4 is developed. With this new architecture, all *Message Handler* classes moved to their message definition bundles and an OSGi service called *IMessage Handler Provider Service* that provides a *Message Handler* constructor for a given configuration identifier is developed. With that change, now the *Core* bundle does not depend on the *Message Handlers* or message bundles, but it depends on the *Message Handler Provider Service* bundle. Message bundles also depend on this service bundle too. This fixes the problem of the *Core* bundle depending on *Message Handlers* and its need to be updated to include a dependency with every new sensor integration. These message bundles, similar with every other OSGi bundle, can be extracted as a compiler .jar file and be

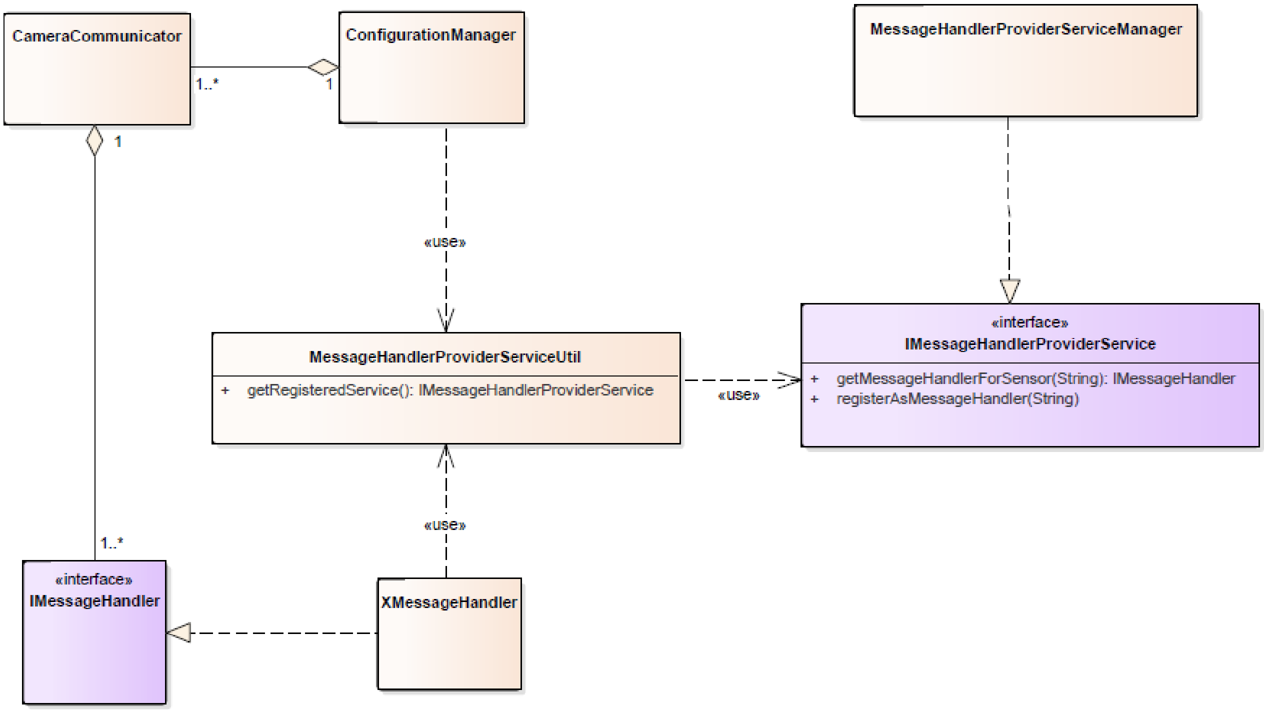

CameraCommunicator
ConfigurationManager
1..*
1
1
MessageHandlerProviderServiceManager
«use»
MessageHandlerProviderServiceUtil
+ getRegisteredService(): IMessageHandlerProviderService
«use»
«interface»
IMessageHandlerProviderService
+ getMessageHandlerForSensor(String): IMessageHandler
+ registerAsMessageHandler(String)
«use»
1..*
«interface»
IMessageHandler
XMessageHandler

Figure 5. Camera Adapter Class Diagram (Simplified)

installed externally.

Figure 5 shows the new classes and their hierarchies while Figure 6 shows the new message handler initialization procedure. The *Message Handler Provider Service Manager* implements the *IMessage Handler Provider Service* interface and when it is initialized by the *StartLevelEventDispatcher*, it reads a directory in which the new sensor integration bundles are placed as .jar files. The manager installs those new integrations and, after the initialization of every new bundle, it registers itself as an instance that implements the *IMessage Handler Provider Service* interface to the OSGi context.

While those bundles are initialized, they register themselves with the *IMessage Handler Provider Service* in the OSGi context using the configuration identifier to indicate the sensor they should be used for. Accessing the registered *IMessage Handler Provider Service* is made possible through the *Message Handler Provider Service Util* class. This access technique blocks the requester thread until a service instance registers. The *Message Handler Provider Service Manager* registers itself after it initializes every integration file. Because *Message Handlers* access this manager using the same blocking technique, they can only register themselves after the service manager finishes its job. This causes all *Message Handlers* to register almost simultaneously.

While this process continues, the *Core* bundle also starts by the *StartLevelEventDispatcher* thread and continues its regular processes. But this time, the *Configuration Manager* class does not know any *Message Handler* itself. The dependencies for *Message Handler* classes are removed. When the *Configuration Manager* reads a sensor configuration, it uses its configuration identifier and asks a *Message Handler* constructor from the registered *IMessage Handler Provider Service*. It uses the *Message Handler Provider Service Util* class to access the service, so it also waits until an *IMessage Handler Provider Service* finishes its initializations and registers itself. After that, if a *Message Handler* for a given configuration identifier exists in the application, the *Configuration Manager* uses its constructor to create an instance and initialize it. The initialized *Message Handler* connects to the sensor and starts its regular processes. If a *Message Handler* does not exist for that identifier, the *Configuration Manager* skips that configuration for this iteration.

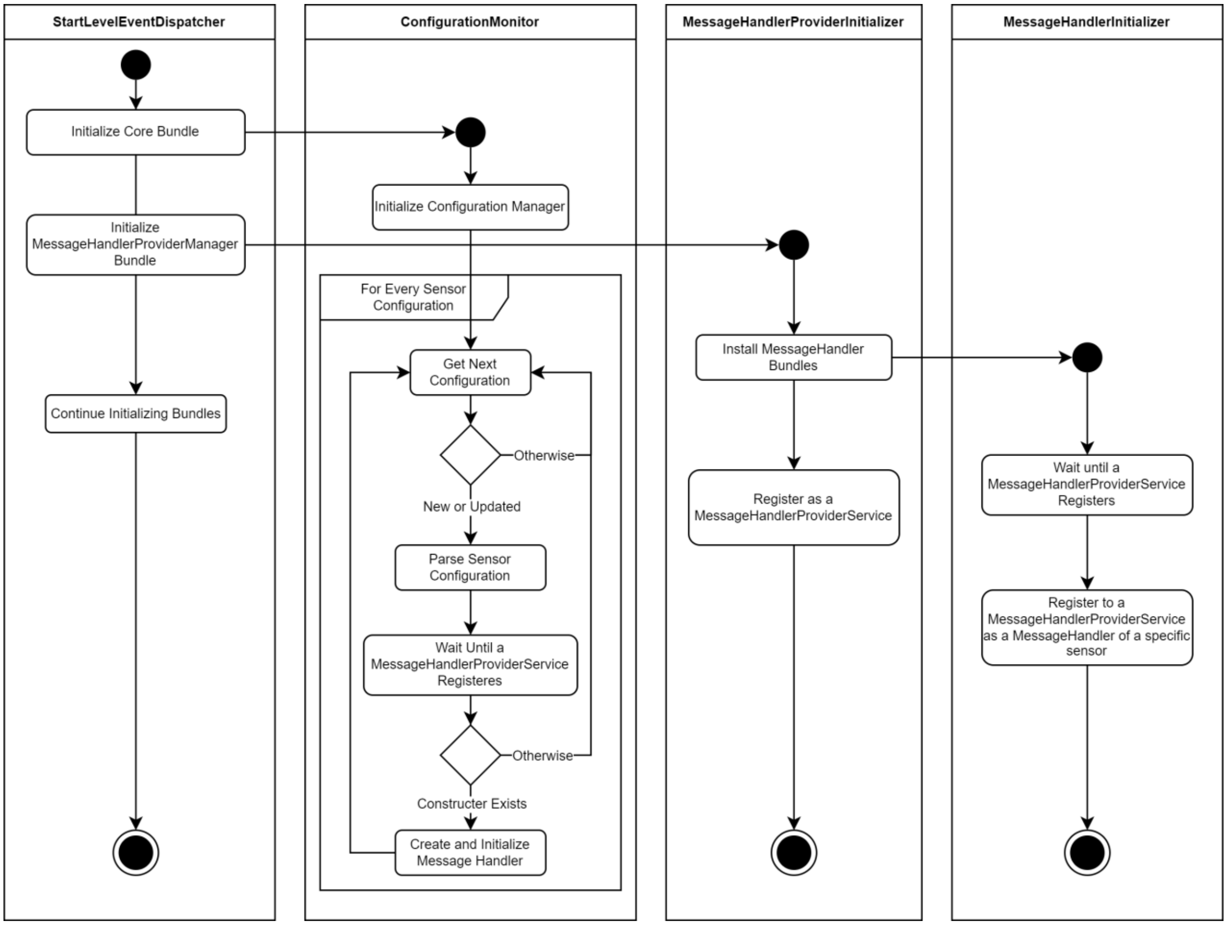

StartLevelEventDispatcher
ConfigurationMonitor
MessageHandlerProviderInitializer
MessageHandlerInitializer
Initialize Core Bundle
Initialize Configuration Manager
Initialize MessageHandlerProviderManager Bundle
For Every Sensor Configuration
Install MessageHandler Bundles
Get Next Configuration
Continue Initializing Bundles
Otherwise
New or Updated
Wait until a MessageHandlerProviderService Registers
Register as a MessageHandlerProviderService
Parse Sensor Configuration
Register to a MessageHandlerProviderService as a MessageHandler of a specific sensor
Wait Until a MessageHandlerProviderService Registeres
Otherwise
Constructer Exists
Create and Initialize Message Handler

Figure 6. Message Handler initialization in the new architecture

VII. Conclusion

The proposed adapter architecture allows us to integrate additional sensors into the already deployed PSIM systems, without requiring to generate another complete build of an adapter. Because previous integrations are not touched, integration tests of only the newly integrated sensors would be sufficient. When the sensor is integrated, it will most probably be available and going to the field and using the sensor of a customer will no longer be needed.

The .jar files of the integration bundles are smaller than one MB so system updates can be completed even with unstable or slow networks. Even if new sensor integrations have a problem working with previously integrated sensors, simply removing the .jar file would be enough to revert back to the previous deployment.

Segregating sensor integration also enables easily selecting and combining different integration bundles according to the project's requirement, as one could expect from a system developed with software product line principles. The new design also enables employing 3rd party companies for integrations without sharing the bulk of the adapter code. Now, any integrator could develop an integration bundle only with the *Message Handler*, *IMessage Handler Provider Service* and the *Message Handler Provider Service Util* classes.

The new architecture provides a helpful pattern towards transforming SecureX into a Software Product Line (SPL). An external .jar installer service could be used not only for sensor integrations, but also for features such as additional GUI views or in the server, new alarm evaluation algorithms. Because every feature is developed as an OSGi bundle, they all could be externalized.

The sensor integration problem could be solved by developing an SDK, similar to the products given in the Section II, but this design also eliminates the need of deploying the SecureX with a full feature set and stripping it off with configuration files at runtime. As this design gets implemented in other parts of the SecureX, they could all be removed from the base build and can be added per customer demand. The new design opens a path for segregating such different aspects in the SecureX and is expected to be even more beneficial in the future.

References

[1] Genetec KiwiVision. [Online], retrieved March 2022 Available: https://www.genetec.com/products/

[2] Milestone XProtect. [Online], retrieved March 2022 Available: https://www.milestonesys.com/solutions/

[3] Nedap Aeos Access Control. [Online], retrieved March 2022 Available: https://www.nedapsecurity.com/solutions/

[4] Open Network Video Interface Forum (ONVIF). [Online], retrieved March 2022 Available: https://www.onvif.org/

[5] B. Tekinerdoğan, İ. Yakın, S. Yağız, and K. Özcan, “Product Line Architecture Design of Software-Intensive Physical Protection Systems”. IEEE International Symposium on Systems Engineering (ISSE), 2020, pp. 1-8, doi: 10.1109/ISSE49799.2020.9272239.

[6] “The Java Language Specification, Java SE 8 Edition” J. Gosling, B. Joy, G. Steele, G. Bracha, and A. Buckley. Apr. 2015. [Online]. retrieved March 2022 Available: https://docs.oracle.com

[7] R. S. Hall, K. Pauls, S. McCulloch, and D. Savage. “OSGi in Action - Creating Modular Applications in Java”. Manning Publications, 2011

[8] “OSGi Service Platform, Core Specification, Release 8,” The OSGi Alliance, April. 2018. [Online]. retrieved March 2022 Available: http://docs.osgi.org/specification/

Requirements Engineering in Healthcare: Lessons Learned from Practice

Malak Baslyman
Information and Computer Science
King Fahd University of Petroleum & Minerals
Dhahran, Saudi Arabia
email: malak.baslyman@kfupm.edu.sa

***Abstract*—Healthcare systems are facing grand challenges in improving current processes and meeting the high demand on resources while maximizing the quality of delivered services. Although technology is a key enabler of improvement, it still fails, in healthcare, due to several reasons, such as poor acceptance by users/physicians, disturbance to existing practices, and lack of comprehensive analysis prior to the implementation of solutions. Hence, we found an opportunity to investigate the effectiveness of some Requirement Engineering (RE) methods, such as goal-oriented and process modeling, in capturing the context of a process under improvement, collecting requirements, and analyzing multiple views and conflicting opinions to support decision-making in healthcare. In this paper, I'm reporting on the challenges and opportunities that were learned while observing and applying some RE modeling and analysis methods in five real-world projects, over five years, in healthcare. In addition, some future research directions are discussed.**

***Index Terms*—Requirements Engineering; Healthcare; URN; Goal-oriented modeling; Process modeling; Industry.**

I. Introduction

These days, most healthcare systems are going through different types of transformations such as changing the purchasing system, from service-based to value-based, and digital health transformation [1] [2]. The transformation aims to deliver a high quality of services, provide patient-centered solutions, and enable technology to improve and digitalize current processes while controlling the budget [2] [3]. As a result, many core changes may be introduced to the structure of healthcare institutions, the role of physicians, healthcare processes and workflow, and the definition of measures and performance targets. Some changes in healthcare, which are related to technological solutions, are still perceived as time-consuming while preventing physicians from doing their jobs, and are difficult to use, with risk exposure and security threats [2] [4] [5]. Thus, physicians, patients, and all stakeholders who belong to the context under change have to be fully engaged in the decision-making process where their goals and concerns are addressed and analyzed adequately.

Requirements Engineering (RE) and its methods regroup proven practices for the elicitation, modeling, analysis, and validation of requirements. It gives a holistic view of the context, including stakeholders, their goals and practices, and enablers and threats. It also supports the evaluation of the potential impact of solution alternatives on those goals and practices in order to select the appropriate solution [6]. The absence of sufficient RE effort can lead to systems that result in unsatisfied users, time/effort lost, low performance, or ignorance about impactful changes [7]. Hence, we were motivated to investigate the use of RE modeling and analysis methods in healthcare-related projects and assess its usefulness in introducing changes and emerging technology effectively [8].

In this study, we report on lessons learned while practicing RE, over five years, in five healthcare projects. we started by exploring RE practices in real-world cases (in one project), then applying advanced RE-based methods to integrate technology effectively into current processes (in three projects) [7] [8]. User Requirements Notation (URN) and its sub-languages Goal-oriented Requirements Language (GRL) and Use Case Maps (UCM) were used for modeling and analysis of stakeholder intentions, values, and processes [6] [9]. In addition, jUCMNav was used for illustration and analysis [10]. One of the major findings in this study is the promising potential for RE methods to be used effectively in healthcare; however, domain-specific solutions and appropriate tool support are needed.

The rest of the paper is organized as follows. Section II provides background about the study motivation and the RE methods used. Section III presents the lessons learned in each project including challenges and opportunities. Lastly, Section IV discusses some of future research opportunities and Section V draws conclusions.

II. Background

This section describes the projects and RE methods used.

A. Study Plan

Briand et al. argued that in a practical field such as Software Engineering, which relies intensively on customers and industry, studies shall be driven by industry needs tailored to a certain context [11]. Context-driven research makes clear assumptions and a well-defined context in addition to achievable objectives and attainable results [11]. As we share the same beliefs, we had planned to study RE practices in healthcare over five years in multiple projects that belong to Canadian and Saudi hospitals. In all projects, managers and their teams were not familiar with most RE practices. In addition, their

RE practices did not go beyond requirements gathering, which are technical, security and functional requirements. Microsoft Excel and Word were used to document the requirements. As illustrated in Figure 1, the study plan consists of three main phases that are observation, design, and implementation. The first project (2015) was meant to investigate RE practices in a real case and observe how the decision on technology selection is made in practice (see Section III-A), which is the observation phase. The investigation resulted in discovering some technology selection and integration problems. Hence, we designed an RE-based framework, which is described in Section II-C, that integrates technology into healthcare practices effectively. Lastly, the framework was implemented successfully in four projects (2016 -2019) that are related to technology integration and context modeling (see Sections III-B and III-C). The next section presents the RE modeling language (URN) that was used to implement the proposed RE-based framework, and to capture and analyze requirements in the projects.

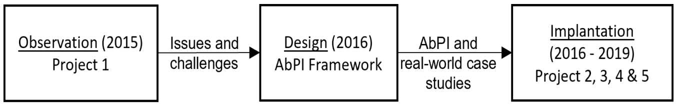


Fig. 1. The study plan of investigating and practicing RE in healthcare

B. User Requirements Notation

URN is the first standardized modeling language that supports requirements engineering activities in a graphical representation way [6]. URN provides two complementary sub-languages that are Goal-oriented Requirement Language (GRL) and Use Case Maps (UCM) [6]. GRL has the capabilities of capturing and modeling stakeholders and intentional elements which include operation goals, softgoals, and tasks. It has three types of relationships between the intentional elements (decomposition, contribution, and correlation) that show how intentional elements are linked to each other and contribute to the satisfaction of stakeholders' goals. GRL also provides a trade-off analysis of design alternatives. The analysis is enabled by the propagation mechanism that propagates the initial evaluation values of goals and the weighted contributions to the root goals to compute their satisfaction values, and the satisfaction of stakeholders [10].

UCM is a causal graphical representation of functional requirements and system behavior. A UCM model consists of start and end points, responsibilities (activities), directions and conditions to guard the transition from one responsibility to another. GRL and UCM can be linked together through URN links to provide a holistic view of the system quality and stakeholder goals, and the system functionalities and behavior. URN has a tool-support (jUCMNav) that enables requirements analysts to model GRL and UCM effectively, and apply appropriate analysis [12]. URN was used to implement the AbPI framework that is presented in the next section.

C. Activity-based Process Integration (AbPI) Framework

The AbPI is a RE-based analysis framework that provides technology integration alternatives into current processes. It also provides a holistic and comprehensive analysis of the impact those alternatives have on stakeholder needs and practices, long-term values, and healthcare urgent needs. The AbPI takes the goal models and the process models of the context under improvement and the new technology to be integrated as inputs. Then, it applies to the main methods: the integration and the evaluation.

In the integration method, the activities of the technology-related process are integrated sequentially into current processes where the relationship between the new activity, to be integrated, and existing ones is captured. For example, a new activity may eliminate, replace, or add to existing activities. Having multiple relationships between new activities of the technology-related process and existing activities of the current process results in several integration alternatives. Hence, the evaluation method analyzes the integration alternatives and assesses the impact of each alternative on predefined criteria. The output of the AbPI framework is the best integration alternative that increases the satisfaction of stakeholders, achieves performance targets, and satisfies selection criteria. The AbPI profiled URN to model and analyze the integration context, GRL was used to capture goal models and UCM to model business and technology-related processes [1] [7] [8].

The AbPI is meant to overcome challenges identified in practice (project 1), and to fill the gap, found in literature, of comprehensively analyzing technology integration in the context of process improvement [7]. There were few RE-based studies that were conducted specifically for the healthcare domain. Most of those studies focus either on requirements elicitation and system design [13]–[16] or process analysis with regard to business objectives [17] [18].

III. Lessons Learned

This section reports on observations, opportunities, and challenges faced during the practice of RE in healthcare projects.

A. Project 1: Technology Selection

The project was about selecting the most appropriate technology for physicians to communicate through. It was led by the IT department. The tasks of the projects were to meet physicians, identify the communication issues, gather their requirements and needs, and map them to a set of off-the-shelf technologies. According to the mapping results and the analysis of the requirements/goals, we reported on the technologies that could be used in this context. The following are observations collected during our work.

1) Requirement analysis: collecting and analyzing requirements started after the business case was prepared. The functional requirements were collected from some physicians; meanwhile, technical and security requirements were identified later by the IT team and the Security and Privacy Office during multiple meetings with service providers. It was observed that little work was done to gather and analyze requirements, especially user requirements, of all involved stakeholders/users in different units, which led to a reluctance to change and an unsatisfied group of users. In addition, there was a paucity in considering non-functional requirements, such as usability, safety, and regulatory compliance.

2) Premature solutions: the main issue was the premature discussions of solutions before identifying current problems and user needs. Also, the lack of an achievable vision, long-term values, and convincing reasons for new changes did not help to negotiate the changes successfully with some groups. Resistance to change was a big obstacle due to the different computer literacy levels, urgent needs, and current goals of each unit. It was clear that the IT and the Security and Privacy teams were in agreement on requirements and the possible solutions; however, physicians were not. Some physicians refused to collaborate at all as they did not see the changes as reasonable; on the other hand, some were very involved and welcoming to those changes.

3) Flexible integration: in a critical environment, such as healthcare, where lives are saved, it is not feasible to impose changes on physicians and obligate them to use certain technology as it may cause delay or deterioration of the quality of provided services. Hence, flexible integration of new changes into current processes is needed, where the current situation, processes, and goals/needs of different stakeholders of different units are captured and analyzed, resulting in integration alternatives. Each alternative, and the status quo, would be evaluated against goals, long-term values, and performance objectives to get a shared understanding of the best way to achieve desired outcomes. This was absent and not thought through in the project.

At the end of the project, the opportunity of using RE-based methods to tackle some of the above-mentioned issues was discussed with the IT manager and team, and caregivers at the hospital. Both groups encouraged applying RE methods as they would be able to have a holistic view of the situation, including the interests and concerns of other units and stakeholders. Also, they emphasized the need for considering long-term values, urgent needs, and sustainability of solutions before implementing them. As a result, we developed the AbPI framework (presented in Section II-C) that was used in the next projects.

B. Projects 2, 3 & 4: Technology Integration

The AbPI framework, discussed in Section II, was applied in three projects: two in Canadian hospitals and one in a Saudi hospital (2016-2018). The three projects were about emerging technologies to automate existing processes: patient information documentation, real-time tracking of lab samples, and real-time waiting estimation systems. Two projects were led by the Security and Privacy Office, and one by the Quality and Patient Safety Department [8]. The tasks were to model goals and processes of different stakeholders, design integration alternatives, and recommend the best integration alternative. Below are lessons drawn on using the AbPI and the tool, where the effectiveness of the framework was proven in practice and some important technical issues arose.

1) Effectiveness: the AbPI framework guided the integration process in the three projects effectively; in addition, the results of the AbPI supported the project managers' decisions. In one project, the project manager decided to suspend the project temporarily, based on the recommendation of the AbPI framework, until a better solution was found. While in the other two, managers chose to implement current solutions partially to satisfy some urgent needs, even though the cost was high, and some stakeholders were unsatisfied. In the projects, one of the major challenges faced was the definition of measures. The AbPI supported the definition of measures and linked them to goals and activities of the processes for analysis.

2) Tool support: it is challenging to use current RE tools in industry. In the context of AbPI, there are many types of relations within activities of processes, and between activities and goals. In jUCMNav for example, the relations could only be captured through URN links between UCM and GRL models, which require many interactions and are not entirely visible on diagrams. For analysis, tasks were used in the goal model to represent the impact of activities on goal satisfaction. Hence, there is a need for usable context-specific tools, as in the integration, that provide appropriate support. The tool shall automate the creation of models, especially alternatives, and provide semantic correctness and consistency checking. Also, the impact of activities of processes on goals shall be illustrated automatically when appropriate data is available, such as the time or cost of an activity.

3) Context-specific goals: urgent needs and long-term values are examples of special types of goals that are used in healthcare [8]. Assessing potential solutions against urgent needs was fairly straight forward. However, capturing and analyzing long-term values by GRL intentional elements was challenging. On one hand, healthcare always strives for maximizing values in delivered services. For example, would the satisfaction level of a goal of long-term value type exceed 100 be considered positive or a desired outcome? What does 100 mean in long-term values evaluation? On the other hand, most solutions evolve over time until long-term values are achieved. Accordingly, some solutions may not fully satisfy long-term values at a certain time; however, they build the basis for more advanced solutions to be developed. Hence, there should be a way to distinguish between low satisfaction values resulting from poor solutions and low satisfaction values which were produced due to the evolution of solutions and current capabilities of hospitals; the former is negative, while the latter is positive.

4) Conflicting opinions of stakeholders: in healthcare, physicians are a special type of stakeholder. They are the owners and users of most of the processes and e-systems. Hence, capturing all their requirements and opinions is essential as it will influence greatly the selection of solutions. The challenge faced was modeling the conflicting opinions of stakeholders who belong to one group. For example, a group of physicians may see system X as a facilitator, while another group sees it as an obstacle on their way to save lives. GRL does not give the flexibility to model the conflicting opinions of the same stakeholder (actor) in one model, which happens always in healthcare. However, this could be solved partially, in jUCMNav, using a contribution override option in the strategy evaluation or having another actor of the same stakeholder type but with a different name.

5) Scalability and effort: one process may cut across multiple units of an organization, or even across different organizations (e.g., hospitals and clinics). Each unit has its own processes, roles, goals, and quality criteria. Although large URN models were created and analyzed in the past, URN models may not scale well at modeling, analyzing, and maintaining multiple large processes across the organization or across organizations. Also, we modeled the context manually; hence, automation will be required for a large set of processes and wider contexts. A considerable amount of time was spent to collect data and build the models as we had to build our own domain expertise along the way; processes were not documented anywhere.

6) Usability: the usability of AbPI was assessed by real users in healthcare participating in a usability study [8]. The participants were given a task of applying AbPI to design integration alternatives and choose the best integration alternative. Even though the unit under analysis was the AbPI, direct comments about GRL and UCM were received. Participants perceived UCM as easy to understand and implement. They described GRL to be a powerful analysis method as it includes stakeholders, goals, and measures. However, GRL also was considered complicated and difficult to use, especially the contribution and propagation mechanism. GRL and UCM seemed to give a holistic vision and evaluation of the context as participants reported. A comment was received to customize GRL and UCM to the healthcare domain, or potentially develop a domain-specific language for healthcare, and consider the use of healthcare wording rather than using RE vocabulary.

7) Change Management: combining change management methods, such as Lean management, and the AbPI led to better analysis. The strongest points of Lean is defining measures and assigning performance targets. However, Lean focuses only on customer (patient) value, while ignoring other stakeholders. Hence, the AbPI was leveraged by the data collected in the Lean approach; at the same time, AbPI was used to capture other stakeholder goals and needs, and analyze solutions designed by the Lean. Combining them both brought another benefit that is reducing the number of the integration alternatives as the design of alternatives is guard, in the Lean, by a condition such as add-value or non-added value activities. Hence, this minimized the effort associated with designing and evaluating all integration alternatives [19].

C. Project 5: Context Modeling and System Design

In a Canadian hospital, a department that was responsible for managing research projects was facing issues of 1) monitoring the projects after the funding was granted, 2) unifying the process for receiving and approving those projects, and 3) dealing with a high workload for staff. In addition, staff did not use the system that was designed specifically to solve some of those issues. Hence, in this project, we applied the AbPI framework partially as there was no technology to be integrated. First, we attempted to analyze the problem and identify the opportunities and issues through several meetings with stakeholders. Then, I prepared the input of the AbPI that are the goal and the process models. Following that, a design thinking session was conducted, which resulted in an initial design of the system to be used to facilitate monitoring and tracking the projects and the workload for staff. The initial design of the system was the base point for several modifications, features and additions that appeared in following meetings and brainstorming sessions. The evaluation method of the AbPI, later, was used to select the best system design alternative based on stakeholders' requirements and goals, and other criteria defined by the hospital.

The project manager found three major benefits of the AbPI framework that are:

1) Visualization: the UCM model helped in visualizing the main obstacles in the process that prevented them from achieving their goals. It was to the base point to agree and disagree on the processes' definitions and roles. In addition, the UCM model became the first source in which the process was defined completely and formally.

2) Goal model evaluation: the team, around seven stakeholders, was interested in the capabilities of GRL and the evaluation model. They all agreed that it reflected how far they were from achieving their goals and how likely the new solution may satisfy the goals. Moreover, it helped them to focus on points of improvement rather than guessing what to be improved and why.

3) Tool support: some comments were left also on jUCMNav; they found it effective and very useful throughout the project; however, it required technical expertise, and it was not user friendly. It is worth mentioning that the designed solution was implemented, later, in the hospital.

IV. Discussion

As seen in the previous sections, RE methods were used effectively in practice in the context of technology selection and integration, and system design. The AbPI influenced the decisions made on technology selection in the projects and provided rationals. Also, it is obvious that stakeholders of the projects agreed on the usefulness of the tool support (jUCMNav) and its visualization capabilities, but also agreed that it is unusable in practice and required special technical skills. In addition, it was suggested, in the three projects, that RE practices have to be customized and tailored to the specific needs of healthcare, especially as stakeholders have, almost, equal power of influencing decisions and have conflicting opinions. Another reason is that, now, the domain is going through major transformations, such as shifting from service-based to value-based payment systems and digital health transformation. The transformations expand the circle of stakeholders and decision-makers to include patient and direct community, leverage data-driven techniques, change the model of care, and change the role of caregivers. In the following sections, we discuss those grand changes and highlight some future research opportunities.

1) Value-based Healthcare System (VBHC): aims to provide high-quality healthcare services for individuals and the population while optimizing the distribution and allocation of resources [20] [21]. It puts healthcare under pressure as running processes, technologies, and stakeholder practices always have to be questioned and reassessed for optimization and improvements. Also, value-based healthcare system is different from the traditional system as it brings a new model of care along with new concepts and implementation, such as segmentation of population based on healthcare needs, moving from corrective to preventive model, better patient, and provider experience, etc. [20] [22].

One of VBHC strategies is the Integrated Care Model (ICM) which refers to having a multidisciplinary team, of diverse views, (physicians of different specialties, policy-makers, social workers, managers, etc.) to provide the best services to patients while putting patients in control of their health decisions. It is one the most agreed on, globally, care models in VBHC [23]. There is a big opportunity for RE to contribute greatly to this matter in different ways. For example, having different perspectives on patient health, while providing a high quality of service and optimizing resources is a very interesting case to investigate for an informative and evidence-based decision-making process. Also, it is interesting to investigate the opportunity of providing domain-specific modeling and analysis methods that speak healthcare language, and model healthcare environment (processes, roles, units, strategies, etc.) and characteristics of its entities. That is to identify, quantify, and analyze value in delivered care services.

Adequate and usable tool support is needed that provides automated analysis for the continued evaluation of current solutions, identification of improvements opportunities, and synthesis of models. In addition, there are important aspects to investigate and questions to answer, empirically, in this context such as *What is value in healthcare?*, *How do we model and analyze value in healthcare?*, *How do we quantify value in delivered care services?*, *Are current RE methods sufficient to capture and analyze value in healthcare?*. An interesting challenge in VBHC is defining and using the right measures. While the value definition is still not unified or agreed on globally, there are too many measures of VBHC that have been published by healthcare organizations. That emphasizes the need to define value formally, as mentioned before, support practitioners to identify appropriate goals and measure, and align measures to those goals systematically. Goal-measure alignment is important not only to quantify and assess goals, but also to avoid wasting resources on using too many irrelevant measures. In addition, pathway-measure alignment is essential too because VBHC strategies, such as the integrated care model, change the traditional pathways; hence, it is important to ensure that measures' definitions are aligned with pathways' definitions and correct observations/measurements will be collected from those pathways.

2) Digital Health Transformation: is another essential change that most healthcare will be going through intensively in the coming decades. It is meant to emerge advanced technologies, such as AI and data-driven solutions, to minimize the load on healthcare providers, and to ensure that services are delivered to patients [24]. It aims, in the long term, to shift the nature of healthcare services from being corrective, where treatments are provided to patients, to preventive, where users are treated and diagnosed before they become patients [24]. As a result, dramatic changes will be brought to the structure of hospitals, workflows, service delivery, and physician-patient relationships. This creates a situation where culture change, physician resistance, risks, ethics, privacy and security issues become obvious [2]. Hence, RE could play a pivotal role in many directions starting from assessing the healthcare system readiness for such change, to the user acceptance of such a model for delivering care. Moreover, it could be used effectively to analyze associated risks, user acceptance, concerns, and compliance. Also, RE-based methods could be used to elicit domain knowledge, anticipate events, guide decisions in the presence of uncertainty, and provide customized care delivery processes that are specific to the needs of each patient [25]. Another interesting research dimension is personalized care where RE can support in identifying opportunities for personalization, trade-off analysis of conflicting interests and preferences of patients, and optimizing the patient experience.

The pandemic of Covid-19 fostered the implementation of healthcare digital transformation in some countries and in many directions [26]. For example, in Saudi Arabia, the Ministry of Health launched many healthcare applications to minimize the number of cases in which patients need to go to hospitals. One of the applications is Sehaty (My Health) which provides virtual clinics where patients can see and talk to caregivers online [27]. However, there is no available literature or technical reports that assess the usability of those applications and how users felt when they interacted with the application (User Experience), especially, for elderly and special needs users. The healthcare digital transformation embraces patient-centric strategies. It leverages technology to increase accessibility to healthcare services. Hence, some technologies are meant to be used directly by patients, such as self-triage apps [28], virtual clinics [29], etc. In this context, we believe more focus should be given to usability requirements and user emotions because they affect patients' perception of the effectiveness of provided services directly [28]. Usability requirements and user emotions should be treated as first-class citizen requirements and appropriate support to model and analyze them (frameworks, modeling languages, and analysis tools) is needed. Also, human values, privacy, and information security should be given more attention and addressed formally to avoid any harm for end-users and to preserve their rights.

3) Industry-Academia Collaboration: we want to emphasize the need for more collaborations with the healthcare sector. RE research is growing rapidly with many new methods and algorithms; at the same time, the healthcare context is changing quickly and facing grand challenges, which could be resolved by RE support. we believe that RE research should not be kept in the laboratory or, mainly, for academic illustrations; it should be driven by real-world needs and its solutions should be practical and used by end-users.

V. Conclusion

In this paper, it was shown that RE methods were used effectively in five healthcare-related projects and brought real and tangible positive results. The discussed lessons learned also showed that it is essential for both researchers and practitioners to continue investigating the applicability of requirements engineering practices in healthcare, the gap between current practices and desired outcomes, and the needed tools for the RE to be an effective part of healthcare practices. In addition, URN-GRL is perceived as powerful at analysis while URN-UCM is easy to understand and follow. However, they need to be customized to healthcare needs and to use healthcare vocabulary.

Moreover, some grand challenges that healthcare is facing these days are discussed too. The value-based healthcare system brings many research opportunities and areas of improvement, such as defining and analyzing value in delivered care services, where RE-based methods can contribute greatly. Also, the health digital transformation puts end-user (patients) face to face with new technologies that they might not be familiar with or not be confident dealing with it; hence, user needs, emotions, values, and rights shall be addressed adequately in RE research.

Acknowledgment

We would like to thank Prof. Daniel Amyot for his ultimate support and guidance in all projects. Also, we would like to thank the University of Ottawa, Montfort Hospital, Dr. El Mostafa Bouattane, Mr. Yasser Alshalahi, and Itqan initiative for supporting this research. This research paper is supported by King Fahd University of Petroleum and Minerals.

References

[1] M. Baslyman, B. Almoaber, D. Amyot, and E. M. Bouattane, "Using goals and indicators for activity-based process integration in healthcare," in *7th Int. Conf. on Current and Future Trends of Information and Communication Technologies in Healthcare (ICTH 2017)*, ser. Procedia Computer Science, vol. 113. Elsevier B.V., 2017, pp. 318 – 325.

[2] B. Meskó, "Digital health technologies and well-being in the future," *IT Professional*, vol. 22, no. 1, pp. 20–23, 2020.

[3] C. Booker, A. Turbutt, and R. Fox, "Model of care for a changing healthcare system: are there foundational pillars for design?" *Australian Health Review*, vol. 40, no. 2, pp. 136–140, 2016.

[4] R. J. Holden, "What stands in the way of technology-mediated patient safety improvements? a study of facilitators and barriers to physicians' use of electronic health records," *Journal of patient safety*, vol. 7, no. 4, p. 193, 2011.

[5] J. H. Weber-Jahnke, M. Price, and J. Williams, "Software engineering in health care: Is it really different? and how to gain impact," in *Proceedings of the 5th International Workshop on Software Engineering in Health Care*. IEEE Press, 2013, pp. 1–4.

[6] D. Amyot and G. Mussbacher, "User Requirements Notation: the first ten years, the next ten years," *Journal of Software (JSW)*, vol. 6, no. 5, pp. 747–768, 2011.

[7] M. Baslyman, B. Almoaber, D. Amyot, and E. M. Bouattane, "Activity-based process integration in healthcare with the user requirements notation," in *E-Technologies: Embracing the Internet of Things. 7th International Conference, MCETECH 2017*. Springer, Cham, 2017, pp. 151–169.

[8] M. Baslyman, "Activity-based process integration framework to improve user satisfaction and decision support in healthcare," Ph.D. dissertation, University of Ottawa, Canada, 2018.

[9] ITU-T, *Recommendation Z.151 (10/12) User Requirements Notation (URN) - Language definition*, International Telecommunication Union Std., 2012. [Online]. Available: https://www.itu.int/rec/T-REC-Z.151, Retrieved: March, 2022

[10] D. Amyot, S. Ghanavati, J. Horkoff, G. Mussbacher, L. Peyton, and E. Yu, "Evaluating goal models within the goal-oriented requirement language," *International Journal of Intelligent Systems*, vol. 25, no. 8, pp. 841–877, 2010.

[11] L. Briand, D. Bianculli, S. Nejati, F. Pastore, and M. Sabetzadeh, "The case for context-driven software engineering research: Generalizability is overrated," *IEEE Software*, vol. 34, no. 5, pp. 72–75, 2017.

[12] G. Mussbacher and D. Amyot, "Goal and scenario modeling, analysis, and transformation with jUCMNav," in *31st Int. Conf. on Software Engineering - Companion Volume*. IEEE CS, 2009, pp. 431–432.

[13] L. Teixeira, C. Ferreira, and B. S. Santos, "User-centered requirements engineering in health information systems: a study in the hemophilia field," *Computer methods and programs in biomedicine*, vol. 106, no. 3, pp. 160–174, 2012.

[14] C. Weng *et al.*, "An integrated model for patient care and clinical trials (impact) to support clinical research visit scheduling workflow for future learning health systems," *Journal of biomedical informatics*, vol. 46, no. 4, pp. 642–652, 2013.

[15] S. AlHajHassan, M. Odeh, S. Green, and A. Mansour, "Goal-oriented strategic modelling for cancer care in systems of systems context using the i* framework," in *2018 1st International Conference on Cancer Care Informatics (CCI)*. IEEE, 2018, pp. 100–109.

[16] N. Al Kilani, R. Tailakh, and A. Hanani, "Automatic classification of apps reviews for requirement engineering: Exploring the customers need from healthcare applications," in *2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS)*. IEEE, 2019, pp. 541–548.

[17] G. R. Hayes, C. P. Lee, and P. Dourish, "Organizational routines, innovation, and flexibility: The application of narrative networks to dynamic workflow," *International journal of medical informatics*, vol. 80, no. 8, pp. e161–e177, 2011.

[18] C. Damas, B. Lambeau, and A. van Lamsweerde, "Transformation operators for easier engineering of medical process models," in *Proceedings of the 5th International Workshop on Software Engineering in Health Care*. IEEE Press, 2013, pp. 39–45.

[19] M. Baslyman, D. Amyot, and Y. Alshalahi, "Lean healthcare processes: effective technology integration and comprehensive decision support using requirements engineering methods," in *2019 IEEE/ACM 1st International Workshop on Software Engineering for Healthcare (SEH)*. IEEE, 2019, pp. 37–44.

[20] E. Teisberg, S. Wallace, and S. O'Hara, "Defining and implementing value-based health care: a strategic framework," *Academic Medicine*, vol. 95, no. 5, p. 682, 2020.

[21] M. E. Porter, "Value-based health care delivery," *Annals of surgery*, vol. 248, no. 4, pp. 503–509, 2008.

[22] M. E. Porter *et al.*, "What is value in health care," *N Engl J Med*, vol. 363, no. 26, pp. 2477–2481, 2010.

[23] World Health Organization - Regional Office for Europe, *Integrated care models: an overview*, 2016. [Online]. Available: https://www.euro.who.int, Retrieved: March, 2022

[24] B. Meskó, Z. Drobni, É. Bényei, B. Gergely, and Z. Győrffy, "Digital health is a cultural transformation of traditional healthcare," *Mhealth*, vol. 3, 2017.

[25] A. Sutcliffe *et al.*, "Known and unknown requirements in healthcare," *Requirements engineering*, vol. 25, no. 1, pp. 1–20, 2020.

[26] J. N. Olayiwola, C. Magaña, A. Harmon, S. Nair, E. Esposito, C. Harsh, L. A. Forrest, and R. Wexler, "Telehealth as a bright spot of the covid-19 pandemic: Recommendations from the virtual frontlines (" frontweb")," *JMIR public health and surveillance*, vol. 6, no. 2, p. e19045, 2020.

[27] Omnia Health, "Saudi arabia's evolving healthcare system: What's new in 2020." 2020. [Online]. Available: https://insights.omnia-health.com, Retrieved: March, 2022

[28] M. K. Ziabari, D. Amyot, W. Michalowski, E. M. Bouattane, and N. Hafez, “Creating mobile self-triage applications: Requirements and usability perspectives,” in *2021 IEEE 29th International Requirements Engineering Conference Workshops (REW)*. IEEE, 2021, pp. 268–277.

[29] Ministry of Health, “Moh news - moh: e-health application launched.” 2017. [Online]. Available: https://www.moh.gov.sa, Retrieved: March, 2022