Università di Genova | DIBRIS DIPARTIMENTO DI INFORMATICA, BIOINGEGNERIA, ROBOTICA E INGEGNERIA DEI SISTEMI

Genova Lost and Found System

Software System Design and Modeling

Professor Maura Cerioli and Gianna Reggio

Hussein Hijzi- 5467332

Enrico Pezzano – s4825087

Mohammad Torabi- 5806782

Business Case

Lost & Found is a software system for helping the regain of an object, so it supports respectively the loser and the finder. In the following paragraphs we'll describe entirely the concerning system's business case:

- **Recovery Rates:**

  To improve the recovery rates of lost items, the objective is to address the challenge users face in retrieving their belongings. This involves implementing efficient categorization, creating searchable databases, and providing timely notifications to users. The aim is to enhance the overall Lost and Found system, making it easier for users to locate and reclaim their lost items.

- **Loser and Finder:**

  The proposed revamp of the Lost and Found system focuses on streamlining reporting for both Losers and Finders. For Losers, it includes a user-friendly interface, real-time updates, and secure identification methods. Finders benefit from an intuitive reporting interface, recognition programs, and secure item handling. The approach emphasizes transparency, community recognition, legal protections, and a comprehensive registry for a seamless and collaborative system.

- **Costs:**

  To alleviate the challenge of inefficient processes leading to higher operational costs, the objective is to implement a cost-effective system within the Lost and Found framework. This includes optimizing resource allocation, minimizing the requirement for physical storage space, and leveraging technology to streamline operations. The goal is to create a more efficient and economically viable system that reduces manual effort and operational expenses.

- **Police:**

  The proposed comprehensive Lost and Found system for law enforcement prioritizes efficient evidence management by implementing secure documentation, storage, and tracking procedures. Enhanced security protocols, integration with case management systems, and collaboration with neighboring agencies ensure integrity and information exchange. Swift identification of stolen property is facilitated through cross-referencing with databases. User authentication, training programs, and public collaboration mechanisms address security concerns. Legal compliance measures and forensic integration expedite evidence analysis. Proactive crime prevention strategies leverage analytics, while community outreach and education programs promote collaboration and shared responsibility.

- **Secondhand shops:**

  The goal is to establish a Lost Items Reclamation Partnership with secondhand shops to prevent unclaimed items from going to waste. This collaboration involves redirecting non-personal and non-evidentiary items for resale, reducing waste and potentially generating revenue. The integration with secondhand shops also promotes sustainable practices by encouraging the resale and reuse of lost items, contributing to environmental conservation. To incentivize the return of high-value items, a partnership with secondhand shops is proposed, offering finders rewards like discounts or vouchers, fostering responsible item recovery and sustainability.

- **Administrative Processes:**
  To address the issue of manual and time-consuming administrative tasks in lost and found management, the objective is to automate various processes. This includes automating item logging, categorization, and notifications to reduce staff workload and minimize errors. The goal is to streamline administrative functions, making the lost and found management system more efficient, accurate, and less labor-intensive.

Design purpose (Step 0)

This greenfield system, developed using the ADD method, aims to enhance the recovery and return of lost items by leveraging technologies like tracking devices, mobile apps, and online platforms. The system incorporates customer-facing databases with search capabilities and image recognition to streamline retrieval processes and increase efficiency. This comprehensive design is intended to drive the project's development and support the system's construction.

Attribute-Driven Design - First round

The first step of the ADD method involves reviewing the inputs and identifying which requirements will be considered as drivers (i.e., which will be included in the design backlog). The inputs are summarized and available in the following table.
We know that they’re correct because we already reviewed them during the “Phase 2” of the course.
As we are saying before, we have already a prioritization view of our overall requirements, so we report the requirements that we consider as drivers in the following:

Step 1: review inputs

The inaugural step of the Attribute-Driven Design (ADD) methodology necessitates a meticulous examination of the inputs to identify and prioritize those requirements that are deemed as drivers. As previously mentioned, our requirements have already been prioritized, and we present herein the subset of requirements that we consider as drivers for the ADD process.

Req	Description	priority
G1	People should be able to notify the finding of an item	MUST
G2	People should be able to consign a notified item to a point	MUST
G3	People should be able to browse the existing points with their position and opening hours	SHOULD
G4	People should be able to notify the loss of an item	MUST
G5	After having notified a loss the loser should be allowed to browse the found items matching their lost one, and possibly confirm one of them	MUST
G6	Immediately after the notification of a finding, losers of matching items should be notified to browse again the found items and possibly confirm one of them	SHOULD
G9	Second-hand shops should be able to register to the system	COULD
G11	Public establishments should be able to register to act as points	MUST
G12	Police should be able to browse the found items	MUST
G13	People with a digital identity valid in EU should be able to register in the system	MUST
G15	People should always be able to know the status of their loss and findings, including the current available matches	MUST
G17	People and points should be assisted in determining the value of an item by means of dedicated searches on the web using in a critical way the item info	COULD
G18	People should be able to cancel the notification of a loss	COULD
G19	System users should be informed of the rules about getting the non-claimed items, rewards, ... and they should accept them before to use the system	MUST

Req	Description	priority
NF4	Users should access the system by means of a dedicated mobile app	MUST
NF5	Points should access the system by means of a dedicated mobile app	MUST
NF6	Second-hand shops should access the system by means of a dedicated mobile app	SHOULD
NF7	Police should access the system by means of a dedicated mobile app	COULD
OSS1	Establishing an Overall System Structure	

Those requirements reported above are deemed drivers due to their substantial impact on the system's overall behavior. Therefore, we shall prioritize addressing these requirements in subsequent iterations.

Iteration 1: Establishing an Initial Overall System Structure

Given that we are currently in the initial phase of the Attribute-Driven Design (ADD) methodology for a greenfield system, we must carefully evaluate which requirements are most influential in establishing an overarching architectural framework. This assessment will be documented in the following step.

Step 2: Establish Iteration Goal by Selecting Drivers

During this pivotal phase, we shall focus on identifying and evaluating the critical requirements that will shape the overall architectural framework.

Req	Description	priority
NF4	Users should access the system by means of a dedicated mobile app	MUST
NF5	Points should access the system by means of a dedicated mobile app	MUST
NF6	Second-hand shops should access the system by means of a dedicated mobile app	SHOULD
NF7	Police should access the system by means of a dedicated mobile app	COULD
OSS1	Establishing an Overall System Structure	

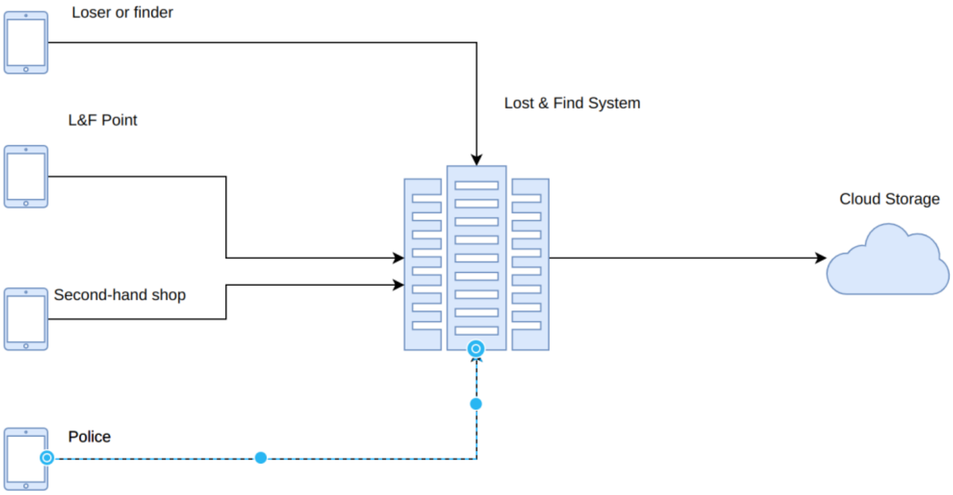
Motivation:
These drivers are important for the overall structure of the system as they ensure that lost and found system that includes a dedicated mobile app and multilingual interfaces caters to the need for user-friendly, inclusive, and efficient item reporting and retrieval. This approach enhances user experience, promotes accessibility, and strengthens the system's reputation in both Italian and English.
NF4, NF5, NF6 and NF7 is related to who can access the system so it has a great impact on
the overall system. And OSS1 is taken into consideration as best practice in case of starting to design a greenfield system likes in our case.

OSS1 is regarded as a best practice when initiating the design of a greenfield system, as applicable in our situation.

Discarded: To establish an encompassing system architecture, the primary focus of this iteration is to identify and prioritize requirements that shape the overall structure as opposed to specific functionalities.

Step 3: Choose One or More Elements of the System to Refine

This being a greenfield development endeavor, the element under scrutiny is the entirety of -the Lost & Found system.



The selection of a cloud storage solution is underpinned by its recognized efficiency in the storage and retrieval of data facilitated through a remote server infrastructure, as opposed to reliance on local storage mechanisms.

This selection holds the potential to yield various advantages for the Lost and Found system, including:

- **Scalability:**

  Cloud storage provides on-demand scalability, allowing you to easily adjust your storage capacity based on the evolving needs of the Lost and Found system. As the volume of lost items and user interactions increases, cloud storage can seamlessly accommodate the growing data requirements without the need for significant upfront investments or infrastructure modifications.

- **Accessibility:**

  Cloud storage enables universal accessibility to your data from anywhere with an internet connection. Users, administrators, and authorized personnel can access the Lost and Found system's data and functionalities regardless of their physical location. This flexibility is especially beneficial in scenarios where users need to report lost items or claim found items remotely.

- **Cost-Efficiency:**

  Cloud storage often follows a pay-as-you-go pricing model, where you only pay for the storage resources you use. This can be more cost-effective, particularly for smaller-scale deployments, as it eliminates the need for large upfront investments in hardware and infrastructure. Additionally, cloud providers typically handle maintenance, reducing operational costs.

- **Automated Backups:**

  Many cloud storage providers offer automated backup solutions, ensuring that your Lost and Found data is regularly and securely backed up. Automated backups help mitigate the risk of data loss due to accidental deletion, system failures, or other unforeseen events. This feature simplifies the process of data recovery and contributes to the overall robustness of your system.

- **Collaboration and Integration:**

  Explanation: Cloud storage facilitates seamless collaboration and integration with other cloud-based services. You can easily integrate the Lost and Found system with other cloud applications, databases, or services, enhancing the overall functionality of your project. This interoperability fosters a more connected and efficient ecosystem.

- **Resource Management:**

  Cloud storage offloads the responsibility of hardware provisioning, maintenance, and management to the cloud provider. This frees up your team's resources, allowing them to focus on developing and optimizing the Lost and Found system's features and functionality rather than managing the underlying infrastructure.

- **Security Measures:**

  Reputable cloud storage providers implement robust security measures, including data encryption, access controls, and regular security audits. By leveraging these built-in security features, your Lost and Found project can benefit from a secure storage environment. Cloud providers invest heavily in maintaining the confidentiality, integrity, and availability of stored data.

- **Automatic Updates and Maintenance:**

  Cloud storage providers handle routine maintenance tasks, such as software updates, security patches, and hardware maintenance. This ensures that your Lost and Found system operates on the latest software versions and benefits from enhanced security features without requiring manual intervention from your team.

Overall, a cloud-based storage solution is poised to furnish the Lost & Found system with the essential attributes of scalability, reliability, accessibility, cost-efficiency, and security requisite for the storage and retrieval of both loser and finder data. Distinguishing L&F Points, Police, and Secondhand shops as distinct entities, we advocate for separate access points to the Lost & Found system to address the specified NF4, NF5, and NF6 requirements outlined in the present iteration. Further elucidation on the intricacies of these applications will be expounded upon in subsequent stages.

Step 4: Choose One or More Design Concepts That Satisfy the Selected Drivers

In this first iteration, aligned with the overarching objective of system structuring, design concepts are meticulously chosen in accordance with the ADD method roadmap. The ensuring table encapsulates a succinct representation of the designated design decisions.

Selected Design Concept	Alternative Design Concept	Description
Mobile application Architecture		This architecture is selected to guide Finder or losers and external individuals in developing applications deployed on handheld devices. The server side aligns with the Microservices Architecture. This strategic choice enables the fulfillment of NF4, NF5, NF6 and NF7 requirements.
N-layer Architecture		The deliberate adoption of the N-Layer approach facilitates a methodical separation of system components, fostering essential decoupling. This architectural strategy not only improves maintainability but also lays the groundwork for enhanced scalability, ensuring adaptability to evolving requirements.
N-Tier Architecture		The N-Tier approach is implemented to physically segregate components within our system, a strategic measure aimed at bolstering both security and performance.

	Microservices Architecture	Microservices is an architectural style where a complex application is decomposed into a set of small, independent services that can be developed, deployed, and scaled independently. Each service represents a specific business capability and communicates with others via APIs. This architecture is selected for its focus on developing applications with a rich user interface within a web browser environment.
	Progressive Web Apps (PWA)	PWAs use web technologies to deliver an app-like experience, combining the best features of web and mobile applications. They are designed to work offline and provide a seamless user experience.
Cloud storage		Cloud-based data storage enhances scalability and availability, offering seamless backup and recovery capabilities essential for meeting the demands of NF8 in the event of system failures.
	Local Database	Dismissed due to the limitations of local storage or database solutions, which may lack the scalability, reliability, and availability inherent in cloud-based alternatives.
RESTful architecture		RESTful architecture, based on the principles of Representational State Transfer (REST), is an architectural style for designing networked applications. REST is often used for building scalable and maintainable web services, providing a set of constraints that guide the design and behavior of these services. Here are the key aspects of RESTful architecture:
	GraphQL Architecture	GraphQL is a query language and runtime for APIs that enables clients to request only the data they need. It was developed by Facebook and has gained widespread adoption as an alternative to RESTful APIs.

Step 5: Instantiate Architectural Elements, Allocate Responsibilities, Define Interfaces

During this phase, our focus shifts to customizing the previously selected architectures. The table below provides a concise overview of the decisions already finalized:

Design Decision and Location	Foundation
Cloud Storage within both applications	This depiction aims to provide an overview of the integration of Cloud Storage.
Layered logical structure architectural pattern	A structured logical framework comprised of four layers is delineated as follows: • **Presentation Layer:** This layer is dedicated to the implementation of requisite functionalities facilitating user interaction with the application. • **Business Layer:** At the core of the system's functionality, the Business Layer is responsible for implementing and encapsulating the fundamental operations and processes integral to the system. • **Data Access Layer:** This layer serves as the gateway, providing access to the data stored either in the database server or external systems. Its primary function is to facilitate interaction with and retrieval of data resources. Subsequently, an elaboration is provided on the nature of services or modules embedded within each of these layers.
Layered physical structure deployment pattern	The adoption of a three-tier architectural model for our system is predicated upon several considerations: • **Client-Tire:** This facilitates client access to the mobile application, enabling individuals to function either as a "loser" reporting lost items or as a "finder" contributing to the Lost and Found system. • **Services-Tier:** Integrating the business logic, situated within the application tier, this layer serves as an intermediary between the client interface and the database.

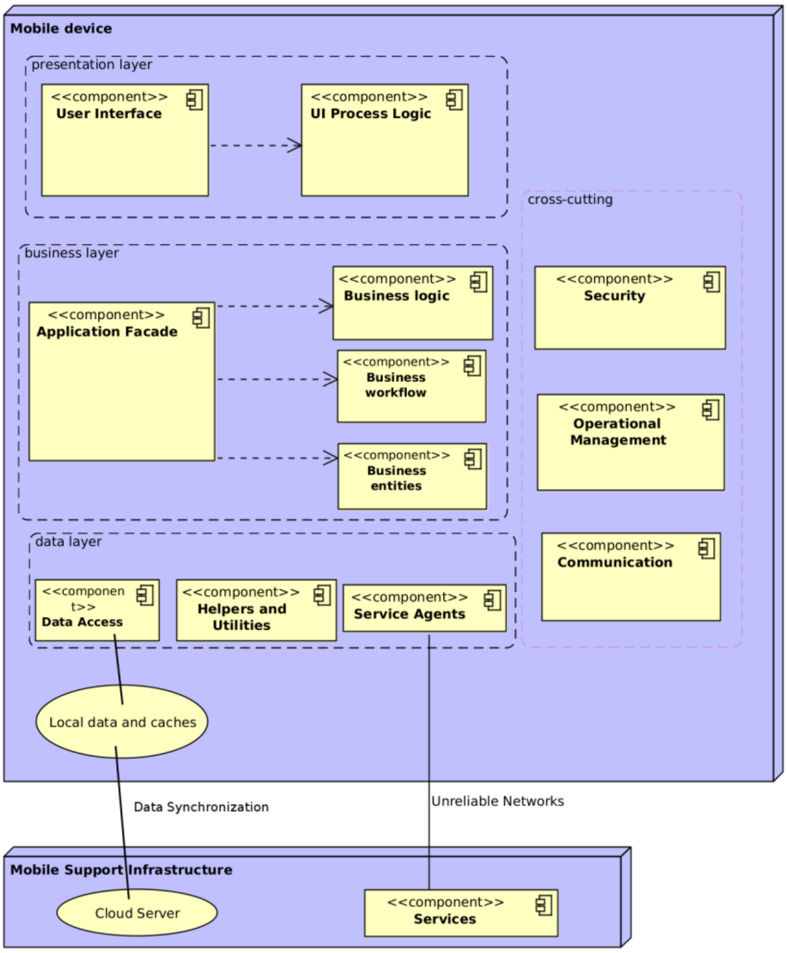
	• **Data Base-Tier:** The migration of an existing database server to cloud storage is imperative.

The outcomes of these decisions are documented in the subsequent phase. In our present iteration, it is premature to precisely delineate functionalities and interfaces, as these aspects will undergo refinement in subsequent iterations.

Step 6: Sketch Views and Record Design Decisions

The diagram in the subsequent figure illustrates the outline of a module view encapsulating the selected reference architecture for clients' applications. These architectures have been subsequently modified in alignment with the design decisions that have been established.

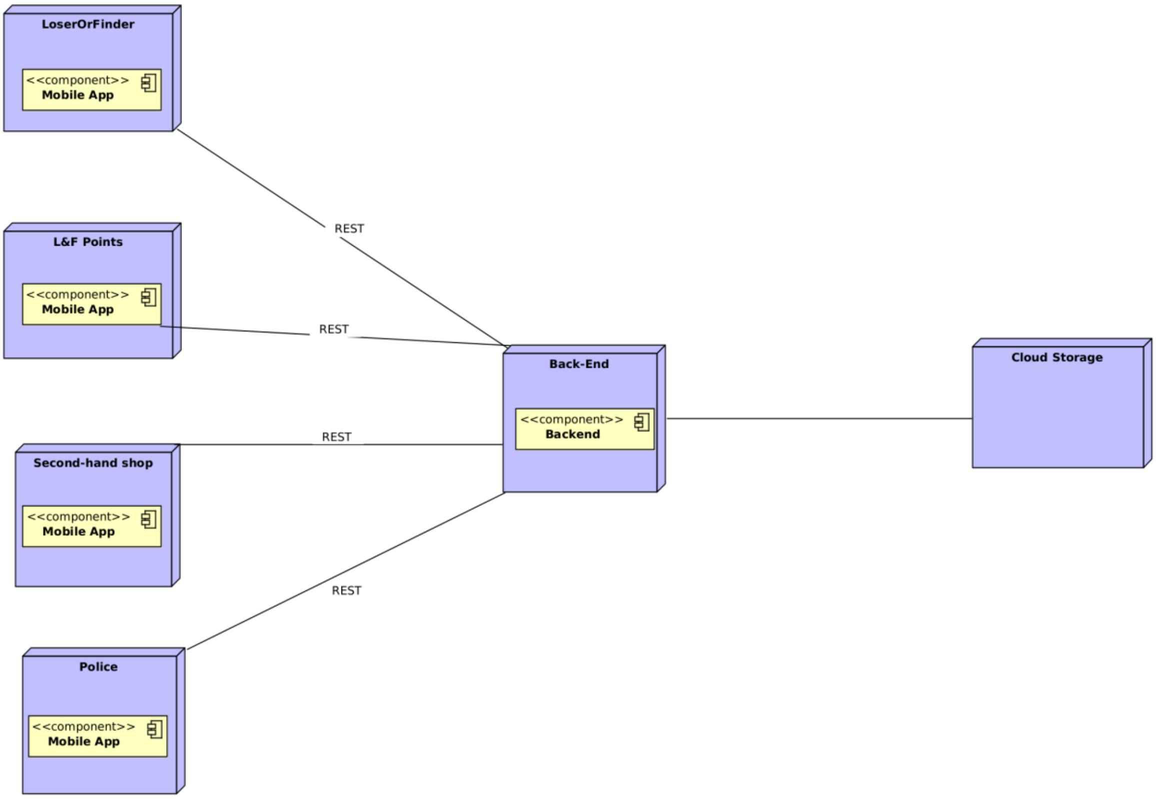
For L&F points, Losers and Finders, Second-hand shop, and Police


Mobile device
presentation layer
<<component>> User Interface
<<component>> UI Process Logic
cross-cutting
business layer
<<component>> Business logic
<<component>> Application Facade
<<component>> Business workflow
<<component>> Business entities
<<component>> Security
<<component>> Operational Management
data layer
<<component>> Data Access
<<component>> Helpers and Utilities
<<component>> Service Agents
<<component>> Communication
Local data and caches
Data Synchronization
Unreliable Networks
Mobile Support Infrastructure
Cloud Server
<<component>> Services

All users employ a nearly identical application, with distinctions primarily existing in the "Business layer" and "Presentation layer".

So, in this table are summarized the main layers and their responsibilities:

Layer	Description
Presentation Layer	This layer is dedicated to implementing the functionality necessary for users to interact with the application.
Business Layer	This layer actualizes the logical functionality of the application. The application façade facilitates the combination of multiple and diverse operations into a unified operation.
Data Layer	This layer encapsulates all the data essential for the proper functioning of the application. The service agent is employed for interactions with external services, while the data helper and utility components collect common logic utilized for data extraction. ld access the system by means of a dedicated mobile app
Cross-Cutting Layer	This layer encompasses shared components utilized across different layers, encompassing: • **Security Layer:** Comprising components dedicated to authentication, authorization, and validation processes. • **Operational Management Tasks:** Encompassing components utilized for logging, configuration, and tracing functions. • **Communication:** Encompassing components dedicated to implementing communication with other services and applications.

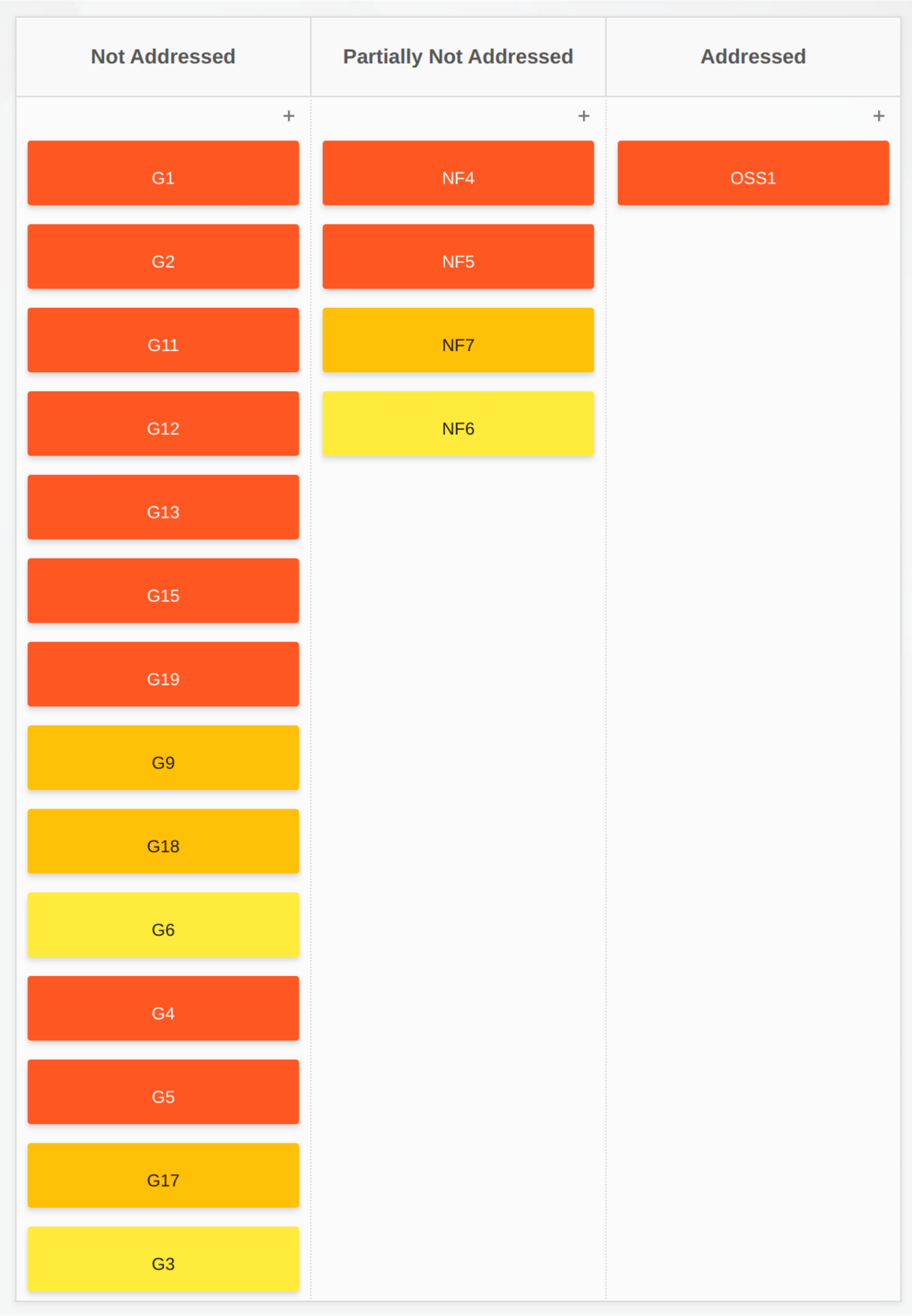


Tier	Description
Mobile Applications	All mobile devices encompass the Mobile App, providing functionality for the elaboration, visualization, and interaction with our application.
Backend	In the mobile application architecture, leveraging the Backend Tier enables the asynchronous synchronization of all data.
Database	The Cloud storage tier incorporates the database responsible for organizing the data.

Given our selection of a RESTful architecture, requests from the mobile application are executed using REST requests.

Step 7: Choose One or More Design Concepts That Satisfy the Selected Drivers

The design progress is summarized in the following table, utilizing the Kanban board:



Not Addressed	Partially Not Addressed	Addressed
+	+	+
G1	NF4	OSS1
G2	NF5	
G11	NF7	
G12	NF6	
G13		
G15		
G19		
G9		
G18		
G6		
G4		
G5		
G17		
G3		

Explanation of Kanban Board: The Red Flag signifies High Priority, the Orange Flag denotes Normal Priority, and the Yellow Flag represents Low Priority.

Design Decisions Made in this Iteration:

- *Not Addressed*: All functionality requirements, as they are not necessary for the description of the overall architecture.
- *Partially Addressed*: NF4, NF5, NF6, NF7: The chosen reference architecture delineates the modules supporting the Mobile applications on the Client-side.
- *Addressed*: OSS1: Established the overall system structure. Addressed by defining a reference architecture for our applications, opting for a layered architecture, and identifying cloud storage for an encompassing view of the overall system structure.

Iteration 2: Defining architectural structures to reinforce operational capabilities

Step 2: Establish Iteration Goal by Selecting Drivers

The primary objective of this iteration is to systematically attend to the overarching architectural imperative of refining the comprehensive structure. Furthermore, the emphasis lies in delineating structural elements essential for buttressing primary functionality. The considerations for this iteration encompass the following requirements:

Req	Description	priority
NF4	Users should access the system by means of a dedicated mobile app	MUST
NF5	Points should access the system by means of a dedicated mobile app	MUST
NF6	Second-hand shops should access the system by means of a dedicated mobile app	SHOULD
NF7	Police should access the system by means of a dedicated mobile app	COULD

Req	Description	priority
G1	People should be able to notify the finding of an item	MUST
G3	People should be able to browse the existing points with their position and opening hours	SHOULD
G4	People should be able to notify the loss of an item	MUST
G17	People and points should be assisted in determining the value of an item by means of dedicated searches on the web using in a critical way the item info	COULD
G19	System users should be informed of the rules about getting the non-claimed items, rewards, etc and they should accept them before to use the system	MUST

Motivation:

In this subsequent iteration, our concentration is directed towards these specific requirements, as they align with the ones previously employed for garnering an understanding of the holistic system structure. However, it is imperative to refine these requirements comprehensively to facilitate informed decision-making regarding the system's functionality in subsequent iterations.

Step 3: Choose the elements to refine

In this iteration, our paramount objective is the meticulous refinement of the previously delineated overall architecture. Given this focus, we abstain from providing further specifications in this step, as we defer to the detailed description of the structure articulated in the preceding iteration.

Step 4: Choose one or more design concepts

Here, the consideration of various design concepts becomes paramount. The ensuing table encapsulates the outlined design decisions:

Selected Design Concept	Alternative Design Concept	Description
Java		Java is extensively used in mobile app development, primarily for building Android applications. Android, the mobile operating system developed by Google, officially supports Java as one of its primary programming languages. Here's a description of how Java is utilized in mobile app development for Android:
	Kotlin	Kotlin is a modern, concise, and interoperable programming language developed by JetBrains and officially supported by Google for Android development. It aims to enhance developer productivity, code readability, and maintainability.
Swift		Swift is a powerful and expressive programming language developed by Apple. It is designed for building iOS, macOS, watchOS, and tvOS applications. Swift is known for its safety features, modern syntax, and performance.
	Flutter	Flutter is a UI toolkit developed by Google for building natively compiled applications for mobile, web, and desktop from a single codebase. It uses the Dart programming language and provides a rich set of pre-designed widgets.

Spring Boot framework		Spring Boot is a Java-based framework for building standalone, production-grade Spring-based applications. It simplifies the configuration and deployment of Spring applications.
	Node.js	Node.js is a JavaScript runtime built on the V8 JavaScript engine. It is commonly used for server-side development, enabling developers to use JavaScript for both frontend and backend development.
Git		Git is a distributed version control system used for tracking changes in source code during software development. It facilitates collaboration among developers.
Amazon RDS (Relational Database Service)		Amazon RDS is a fully managed relational database service that enables you to set up, operate, and scale relational databases in the cloud. It supports several popular relational database engines, allowing you to choose the one that best fits your application's requirements. Amazon RDS automates routine administrative tasks, such as hardware provisioning, database setup, patching, and backups, freeing you to focus on your application.
	Azure SQL Database	Azure SQL Database is a fully managed relational database service provided by Microsoft Azure. It is based on the SQL Server database engine, but it is hosted in the Azure cloud. Azure SQL Database is designed to provide a scalable, secure, and performant platform for hosting and managing relational databases without the need for the user to manage the underlying infrastructure.

Step 5: Adapt design concept to your needs

In this phase, we proceed to refine our earlier determinations, specifically focusing on delineating responsibilities and defining interfaces:

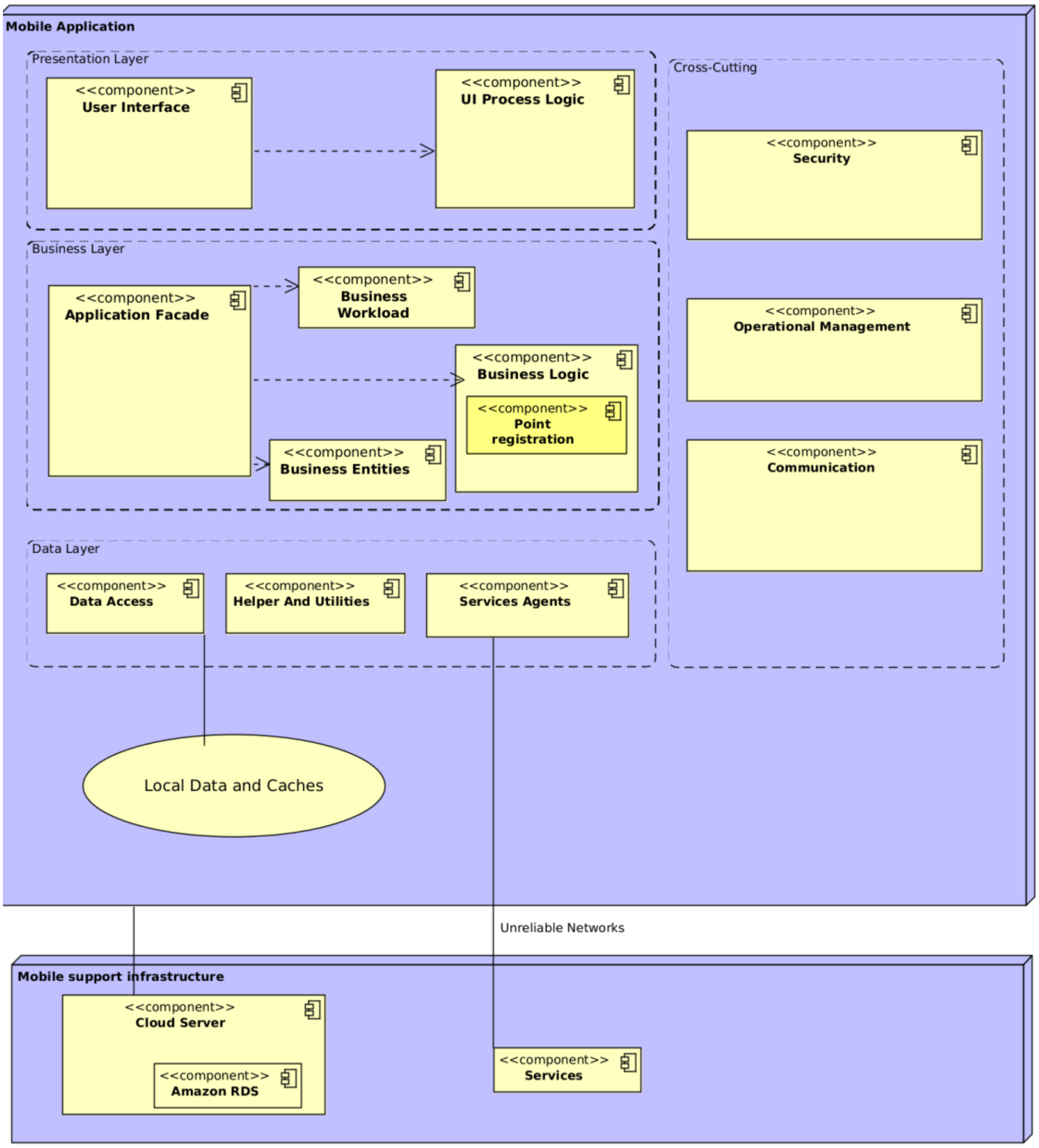
1. Added Spring Boot Framework instead of RESTful architecture
    a. Instead of opting for a RESTful architecture, our preference is to leverage the Spring Boot framework. Spring Boot, renowned for streamlining the development of Java-based applications, including RESTful services, aligns with our choice of a native programming language, offering enhancements to the overall system development.

2. Added Git
    a. Git, as a distributed version control system, is employed to track changes in source code throughout the software development process, making it a viable choice for our project.
3. Added Amazon RDS
    a. Amazon RDS provides comprehensive support for MySQL, rendering it a suitable database solution for mobile applications.

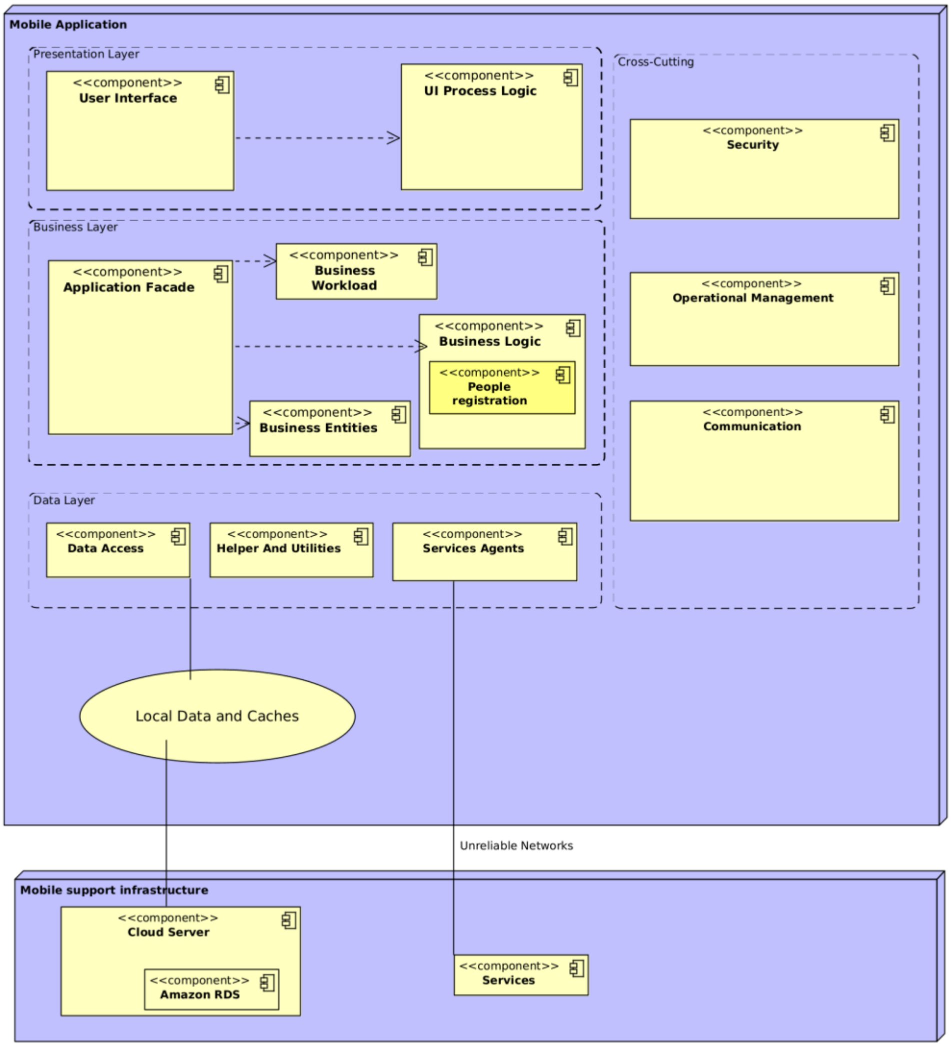
Step 6: Sketch views & record design decisions

Following the previously outlined modifications, the Mobile Application reference architecture undergoes the following changes:

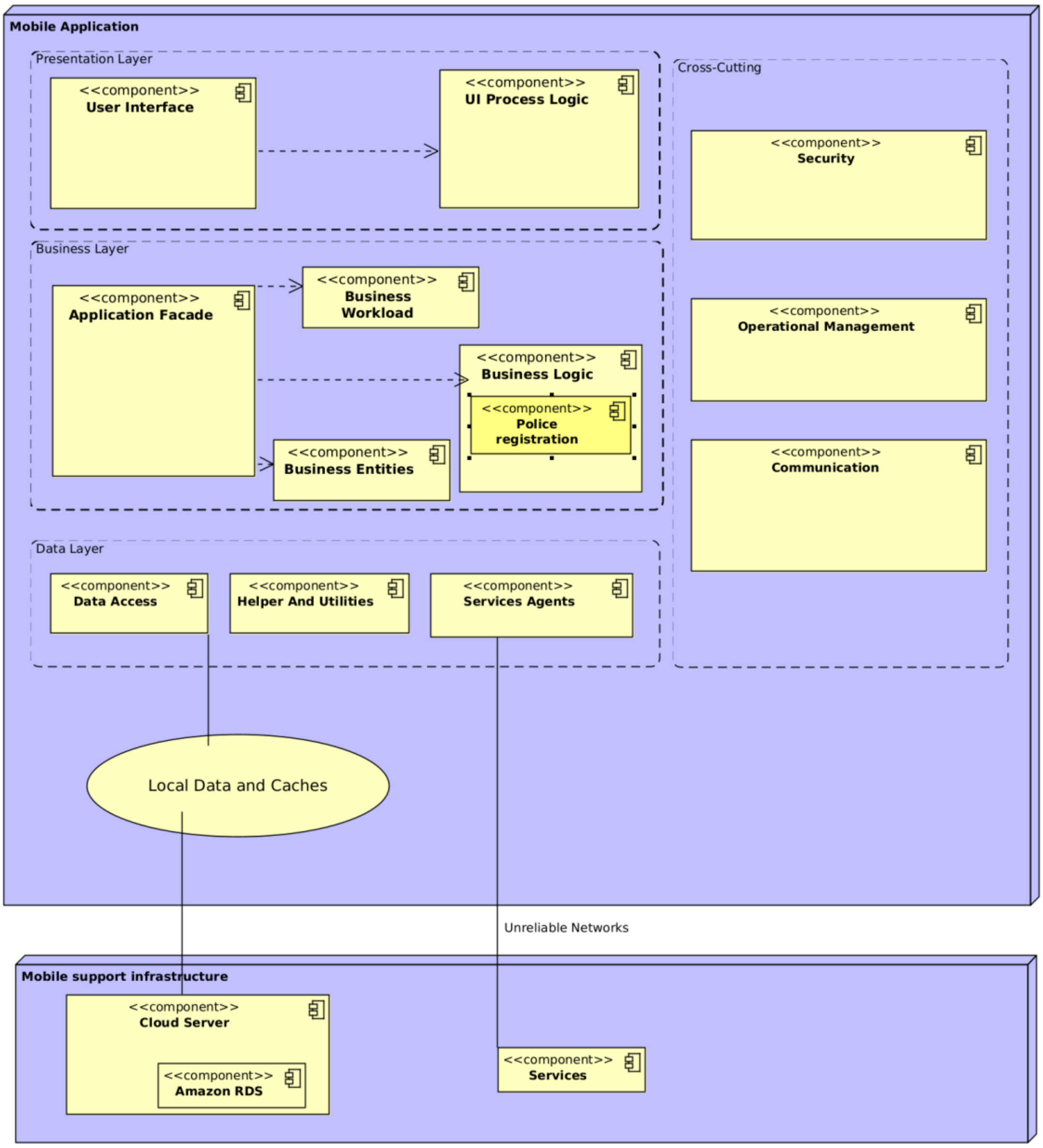
For Points application:



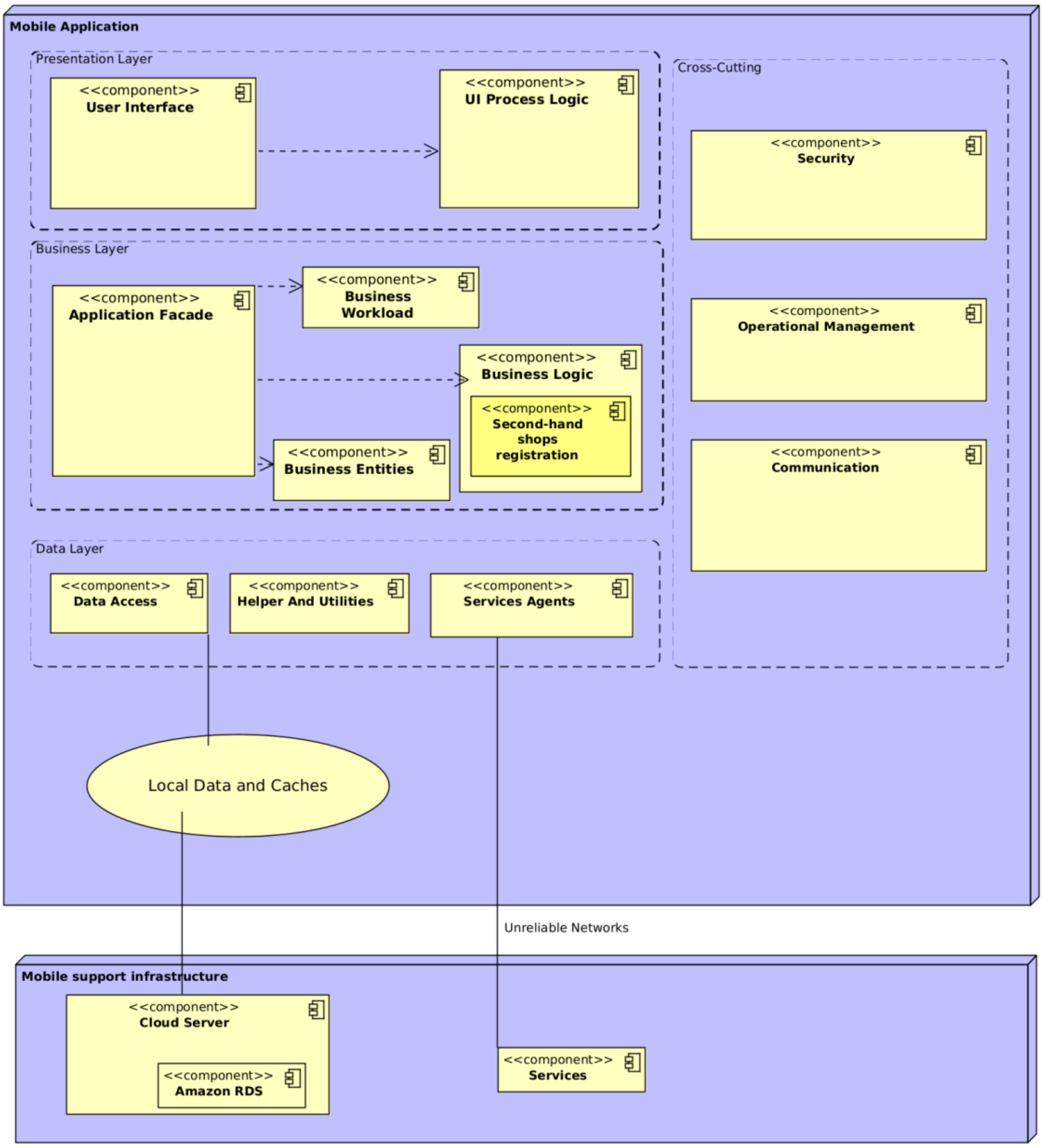
For People application:



For Police application:



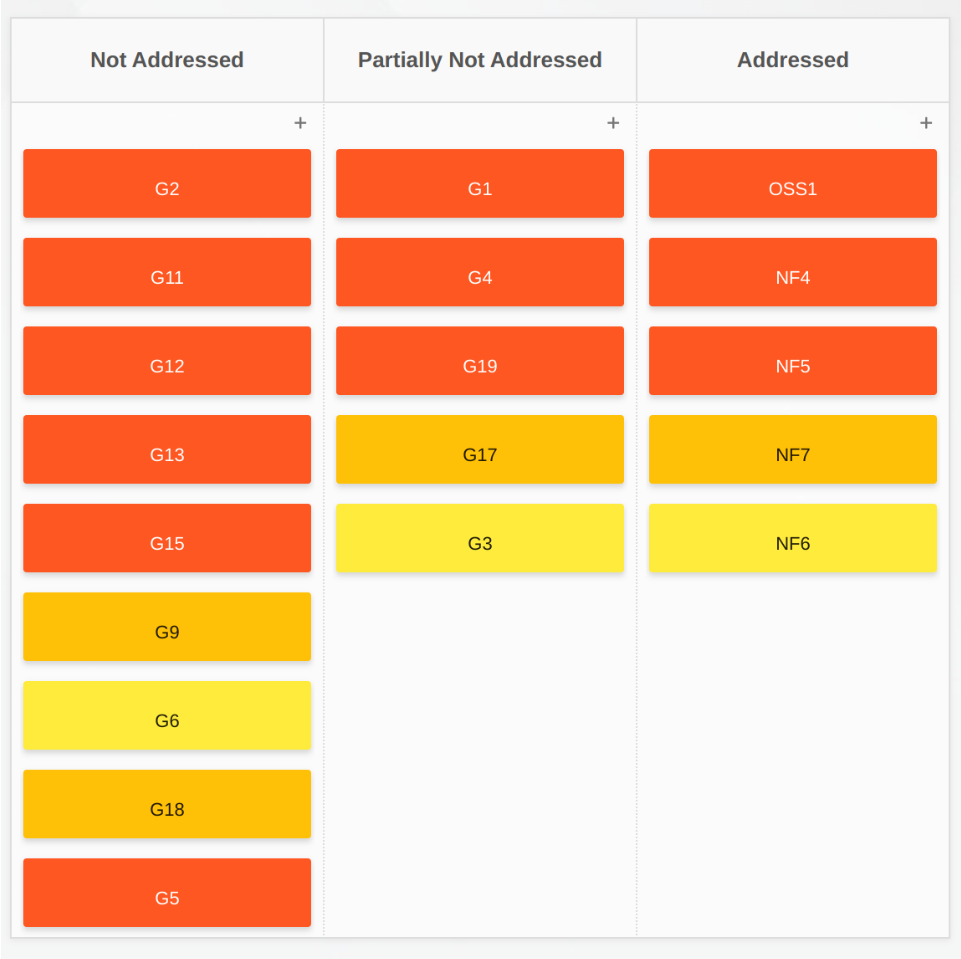
For Second-hand shops application:



In our iterative design process, we endeavored to conceptualize four distinct applications tailored for the functionalities of Points, People, Police, and Second-hand Shops, respectively. Subsequently, we systematically imbued each application with specific objectives pertinent to its designated domain and proceeded to refine the system incrementally, enhancing both its architecture and design iteratively.

Step 7: Perform analysis of current design

Now, we encapsulate the design progress through the visualization provided by the Kanban board.



Not Addressed	Partially Not Addressed	Addressed
G2	G1	OSS1
G11	G4	NF4
G12	G19	NF5
G13	G17	NF7
G15	G3	NF6
G9		
G6		
G18		
G5		

Design Decisions Made in this Iteration:

We have successfully addressed all Non-Functional requirements.

The chosen programming language, storage solution, and reference architecture form the foundation for establishing modules that will support the creation of a mobile application. This application is designed to cater to users of varying priorities and usage scenarios, accommodating the specifications outlined in NF4, NF5, NF6, and NF7 requirements for both Android and iOS devices.

Partially Addressed:

G1, G3, G4, G5, G17: The chosen goals make it easy for users to find nearby locations and their opening hours. It also helps users report lost items. Plus, it helps both users and the locations find out how much items are worth by searching the internet. Before using the system, users are told about and must agree to the rules about what happens to items that aren't claimed and any rewards.

Iteration 3: Addressing Remaining Drivers through Structural Enhancement

Within the current iteration, due regard is given to the functional requirements, with the overarching objective of delineating the primary functionality inherent in the system.

Step 2: Establish Iteration Goal by Selecting Drivers

The objective of this iterative phase is to systematically address the overarching architectural concern pertaining to the identification of structures supporting primary functionality. Within this iteration, due consideration is given to the following specified requirements.

Req	Description	priority
G1	People should be able to notify the finding of an item	MUST
G3	People should be able to browse the existing points with their position and opening hours	SHOULD
G4	People should be able to notify the loss of an item	MUST
G9	Second-hand shops should be able to register to the system	COULD
G11	Public establishments should be able to register to act as points	MUST
G12	Police should be able to browse the found items	MUST
G13	People with a digital identity valid in EU should be able to register in the system	MUST
G17	People and points should be assisted in determining the value of an item by means of dedicated searches on the web using in a critical way the item info	COULD
G18	People should be able to cancel the notification of a loss	COULD
G19	System users should be informed of the rules about getting the non-claimed items, rewards, etc, and they should accept them before to use the system	MUST

Motivation:

Following two iterations dedicated to achieving a well-defined overall structure, we now turn our attention to incorporating functional requirements. This shift allows us to systematically address the primary functionality of the system.

A user must consent to G19 to utilize the application. G1, G2, G3, and G11 were chosen due to their inherent tight coupling and high correlation, as they are interdependent and unable to operate in isolation, as elucidated in subsequent steps. Furthermore, G4, G5, and G15 exhibit coupling among themselves, prompting a dedicated focus on understanding their impact on the system. The correlation between G12 and NF7 is acknowledged. Additionally, G6 and G18 are essential for facilitating easy and seamless claims processing, making them integral components in our considerations. G13, when integrated into the system, plays a pivotal role in user authorization. Leveraging G17, users can access pricing information for individual items. Additionally, G9 enables second-hand shops to register within the system, facilitating the buying and selling of properties.

Step 3: Choose the elements to refine

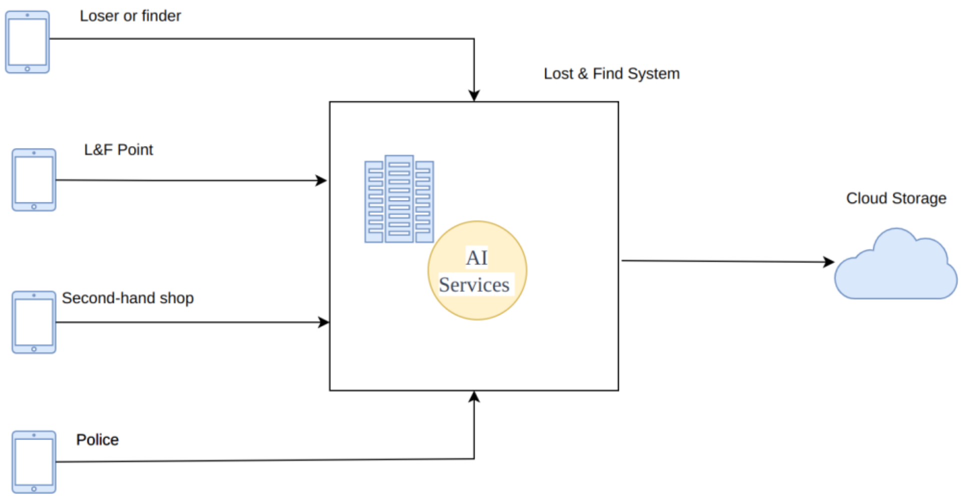
Here, our emphasis lies in the meticulous refinement of structures underpinning the primary function of the system. Our focus extends to the identification of key components and their intricate interactions. As we delve deeper into the conceptual landscape, we articulate the refinement of layers, elucidating the nuanced considerations of external services integral to this step.

Identifying and Enhancing System Elements	***Consideration***
Presentation Layer	The system necessitates the development of four distinct presentation layers tailored to different user roles. Lost and Found (L&F) points will access the system as administrators with comprehensive control over individual points. Losers or finders will have a user interface allowing them to report losses or finds. Police personnel will be provided with access to all data to ensure

	legality verification. Second-hand shops will access the system with interfaces catering to both buying and selling activities.
Business Layer	The system must incorporate logic to accommodate various user roles, including the loser or finder, Lost and Found (L&F) points, police, and second-hand shops. Furthermore, the chat functionality will be situated within the business layer, given its correlation and usage in the context of item loss or discovery events.
Data Layer	The system must have the capability to store exhaustive information pertaining to the discovery or loss of items. Furthermore, seamless data sharing is imperative with Lost and Found (L&F) points, law enforcement agencies, and second-hand shops. Effective communication channels, including chat functionality, need to be established. The system must also adeptly handle communication with external services, such as AI Services.
AI Services	The intended external service is designed to oversee authorization processes and categorize items reported as lost or found within a specific day. Lost and Found (L&F) points are anticipated to leverage this service for legality verification. Moreover, in cases of suspicion, the system is configured to promptly report such instances to the police.

In the subsequent section, a graphical representation of these refinements is presented for visual clarity and conceptual understanding.

Considering the outlined choices, the Overall System Structure undergoes the following modifications:



In subsequent stages, we will delineate the behavior of the AI Service within the system and engage in discussions regarding the necessary changes at the specified layers mentioned earlier. This iterative process aims to achieve a more nuanced and refined understanding of the system.

Step 4: Choose one or more design concepts

In this iteration, we undertake various design decisions aimed at integrating externally developed components into the existing system structure. The following table provides a summary of our design decisions:

Selected Design Concept	Alternative Design Concept	Description
Watson Visual Recognition		Watson Visual Recognition is an AI service provided by IBM Watson that allows developers to analyze and understand the content of images and videos. It uses deep learning algorithms to recognize objects, scenes, faces, colors, and other visual attributes within images and videos. We use this for users' identification.
Watson Natural Language Understanding		Watson Natural Language Understanding is a natural language processing (NLP) service offered by IBM Watson. It analyzes text to extract metadata such as concepts, entities, keywords, sentiment,

		emotion, and relationships. It helps developers understand the meaning and context of unstructured text data. We use this for users' identification.
Watson Machine Learning		Watson Machine Learning is a comprehensive platform provided by IBM Watson for building, training, deploying, and managing machine learning models. It offers tools and services to streamline the end-to-end machine learning workflow, from data preparation and model development to deployment and monitoring. Using this service, L&F point and police would be possible identify robbed items. This also helping us to notify the finding of an item and consign a notified item to a point
	Microsoft Azure Machine Learning	Microsoft Azure Machine Learning is a cloud-based platform provided by Microsoft Azure that enables developers and data scientists to build, train, deploy, and manage machine learning models at scale. It offers a range of tools and services to streamline the entire machine learning lifecycle, from data preparation and model development to deployment and monitoring.
Amazon Web Services (AWS)		Using Amazon API Gateway and AWS Lambda you can create RESTful APIs to serve different presentation layers of your mobile application. API Gateway allows you to manage and secure your APIs, as well as integrate with other AWS services. And execute code in response to API calls. You can use Lambda functions to customize data before sending it to the presentation layer, or to handle specific business logic. Amazon Cognito with AWS AppSync and AWS Amplify can be a powerful combination for specific AWS service that can provide real-time chat in your system.

Step 5: Adapt design concept to your needs

We refined the Presentation and Business logic of the Lost and Found (L&F) Application by incorporating the following components. Additionally, we obtained an updated view of the AI Service, aligning with the previously established design concept.

Presentation layer:

Components	Description
Showing the rules	System users showing the rules about getting the non-claimed items, rewards ,... and they should accept them before to use the system
Notification of the finding items	This component allows users to notify a find item in the system immediately.
Notification of the losing items	This component allows users to notify a loss item in the system immediately.
Showing existing points with location and opening hours	This component allows users to see the existing points with location and opening hours to provide them an information about the points.

Business Logic:

Components	Description
People registration	People with a digital identity valid in EU able to register in the system. We use Watson Visual Recognition to handle this part to do not allow fake people using or system.
Report a lost item	Enables users to report a lost item within the system.
Report a find item	Facilitates users in reporting a found item within the system.
Cancel a lost item	Allows users to cancel a reported loss item within the system.
Second-hand shops registration	Second-hand shops should be able to register to the system to start buying or selling the items they achieve.
Police browsing	Equip the police with a system to ascertain the origin of an item, aiding in the identification of potentially stolen items.

Register to act as points	It uses Amazon Cognito with AWS AppSync and AWS Amplify to manage sign-up, sign-in, and provides a set of tools and libraries for building scalable and secure cloud-powered applications
Showing people and points an item's value	People and points assist in determining the value of an item by means of dedicated searches on the web using in a critical way the item info

Data Layer:

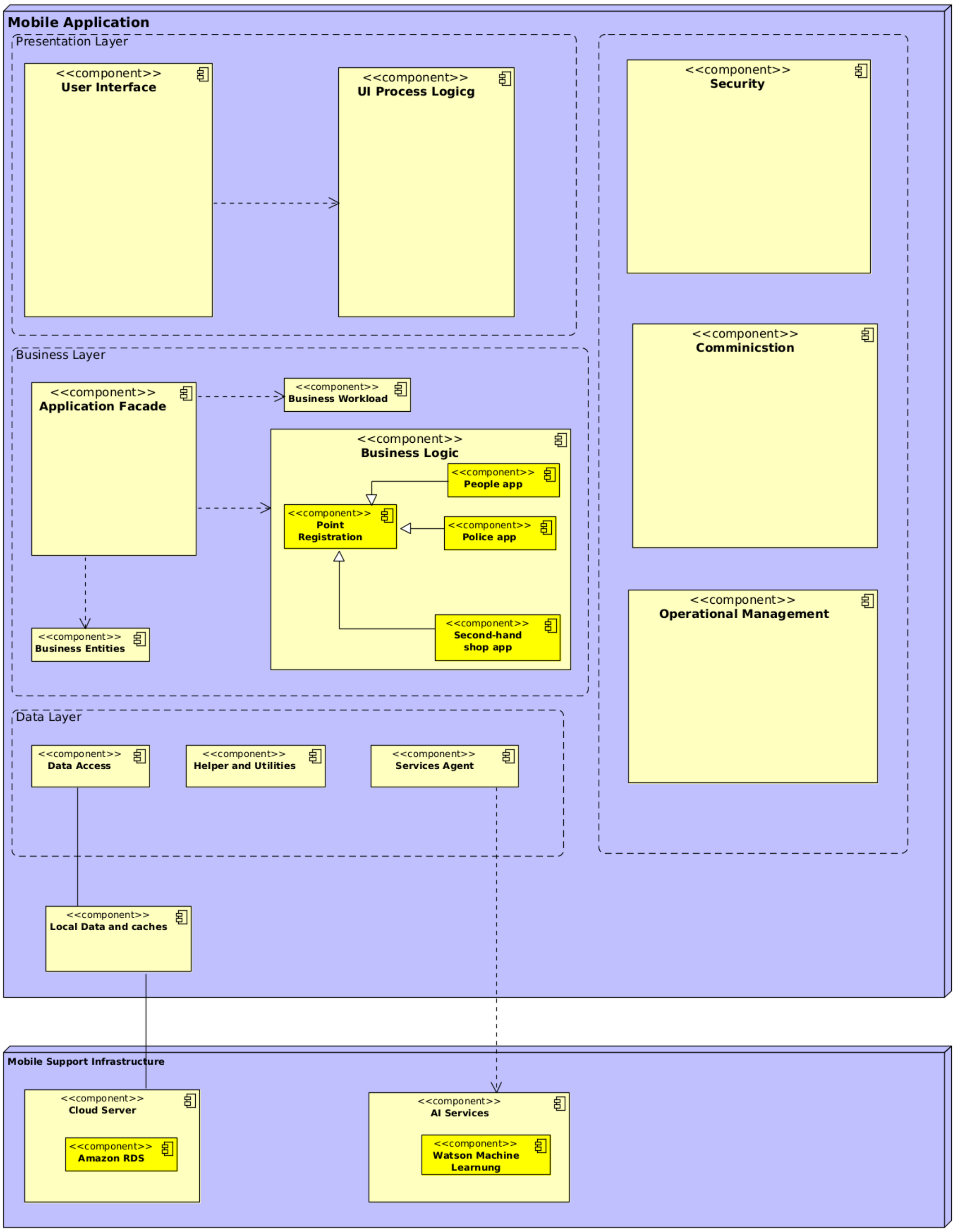
Components	Description
Service Agents	This component within the data layer serves to establish the connection between the data layer and the AI Services.
AI Services	The evaluation service, informed by the design concepts established in the prior step, employs Watson Visual Recognition and Watson Machine Learning for training AI services utilized by Lost and Found (L&F) points. This training aims to enable the identification of stolen items and ensure accurate person identification to bolster system security. Once trained, the AI service can discern the originality of each item. To achieve this, the AI service retrieves data from the system via the Service Agents component within the Data Layer, as previously outlined, or establishes a direct connection with the Cloud Storage.

Step 6: Sketch views & record design decisions

In this step, we illustrate the system prior to the implemented changes, followed by a depiction of the system post-implementation. Additionally, we outline the modifications made on the admin side.

Following the previously defined modifications, the Mobile App Package Diagram for points and people undergoes the following changes:

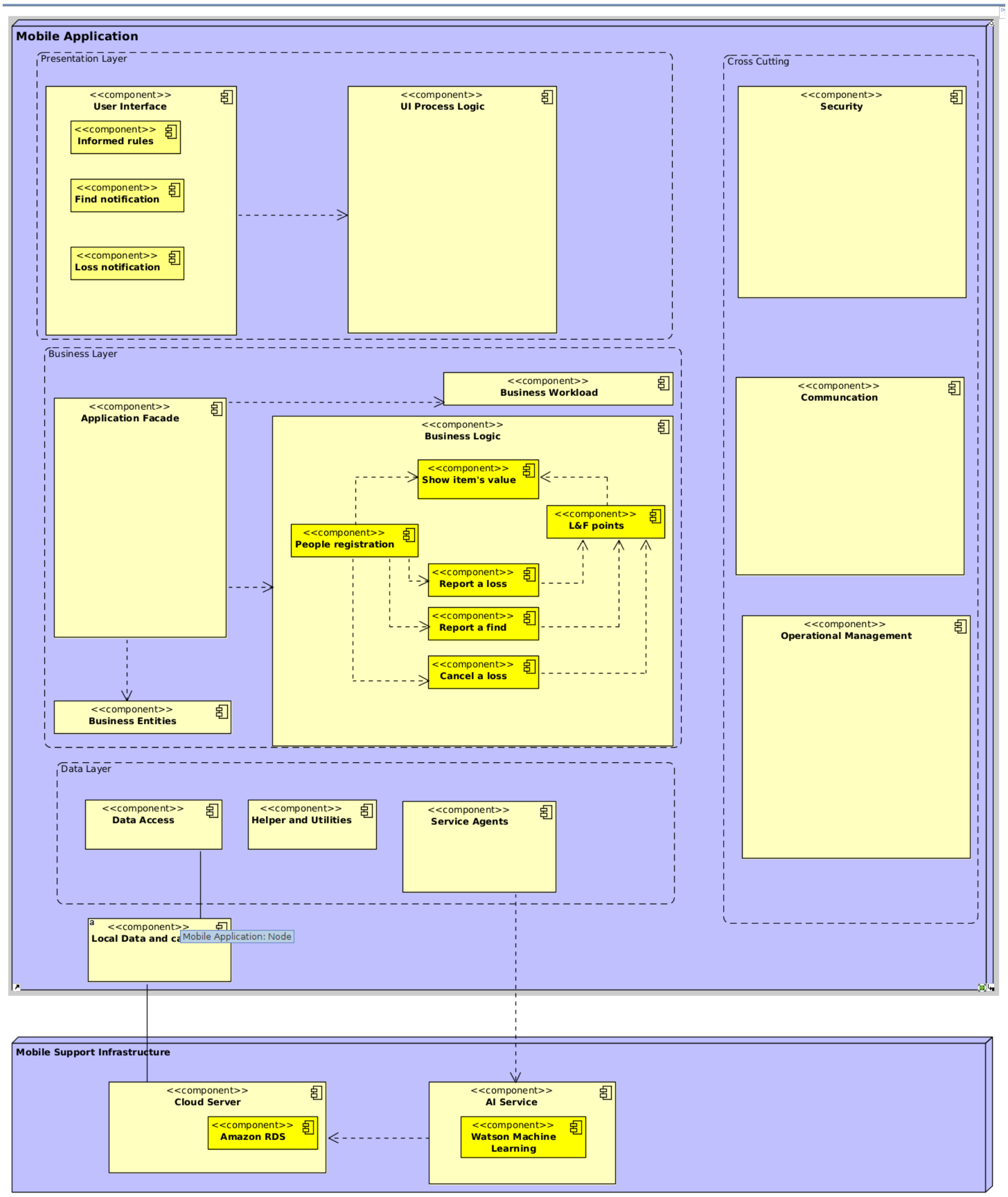
For Points application:



The changes made are the following:

We inserted in give an availability to control and view people, police, and second-and shop within the business logic when user register as a point.

For People application:

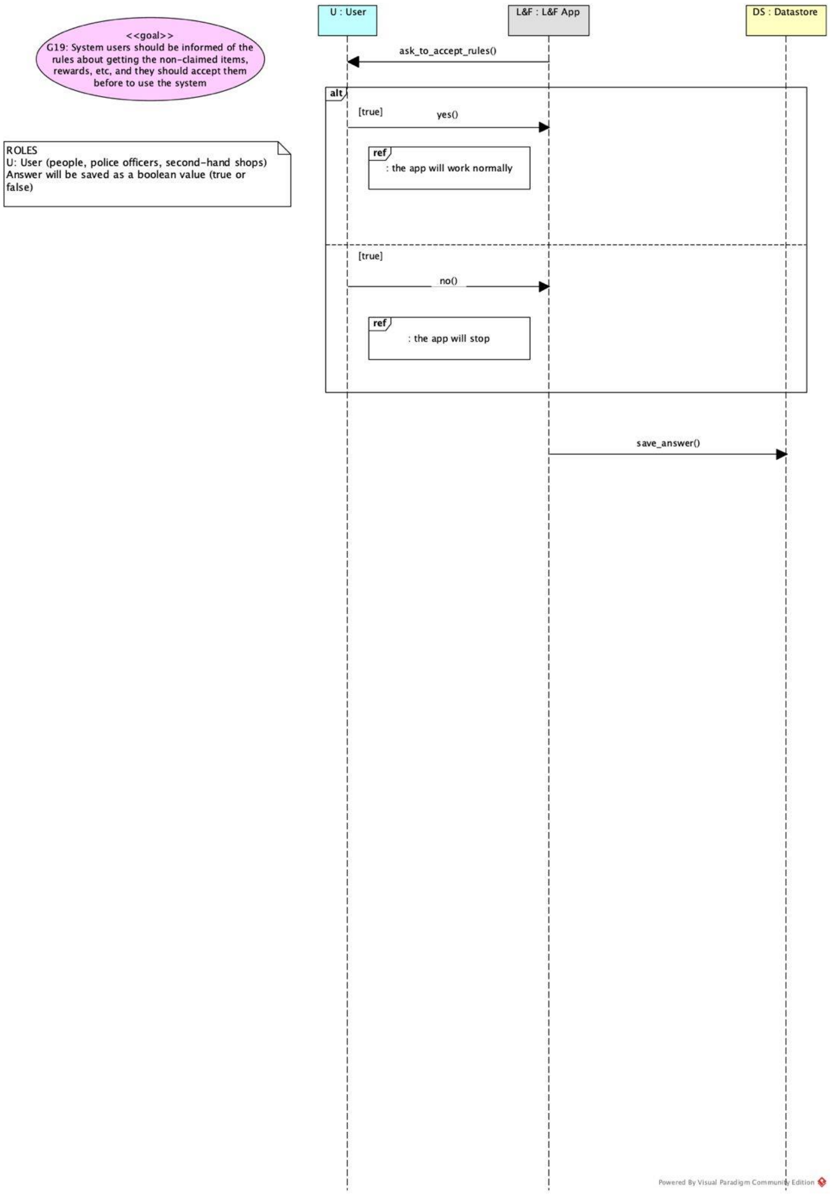

Mobile Application
Presentation Layer
<<component>> User Interface
<<component>> Informed rules
<<component>> Find notification
<<component>> Loss notification
<<component>> UI Process Logic
Cross Cutting
<<component>> Security
<<component>> Communcation
<<component>> Operational Management
Business Layer
<<component>> Business Workload
<<component>> Application Facade
<<component>> Business Logic
<<component>> Show item's value
<<component>> L&F points
<<component>> People registration
<<component>> Report a loss
<<component>> Report a find
<<component>> Cancel a loss
<<component>> Business Entities
Data Layer
<<component>> Data Access
<<component>> Helper and Utilities
<<component>> Service Agents
<<component>> Local Data and ca
Mobile Application: Node
Mobile Support Infrastructure
<<component>> Cloud Server
<<component>> Amazon RDS
<<component>> AI Service
<<component>> Watson Machine Learning

The changes made are the following:

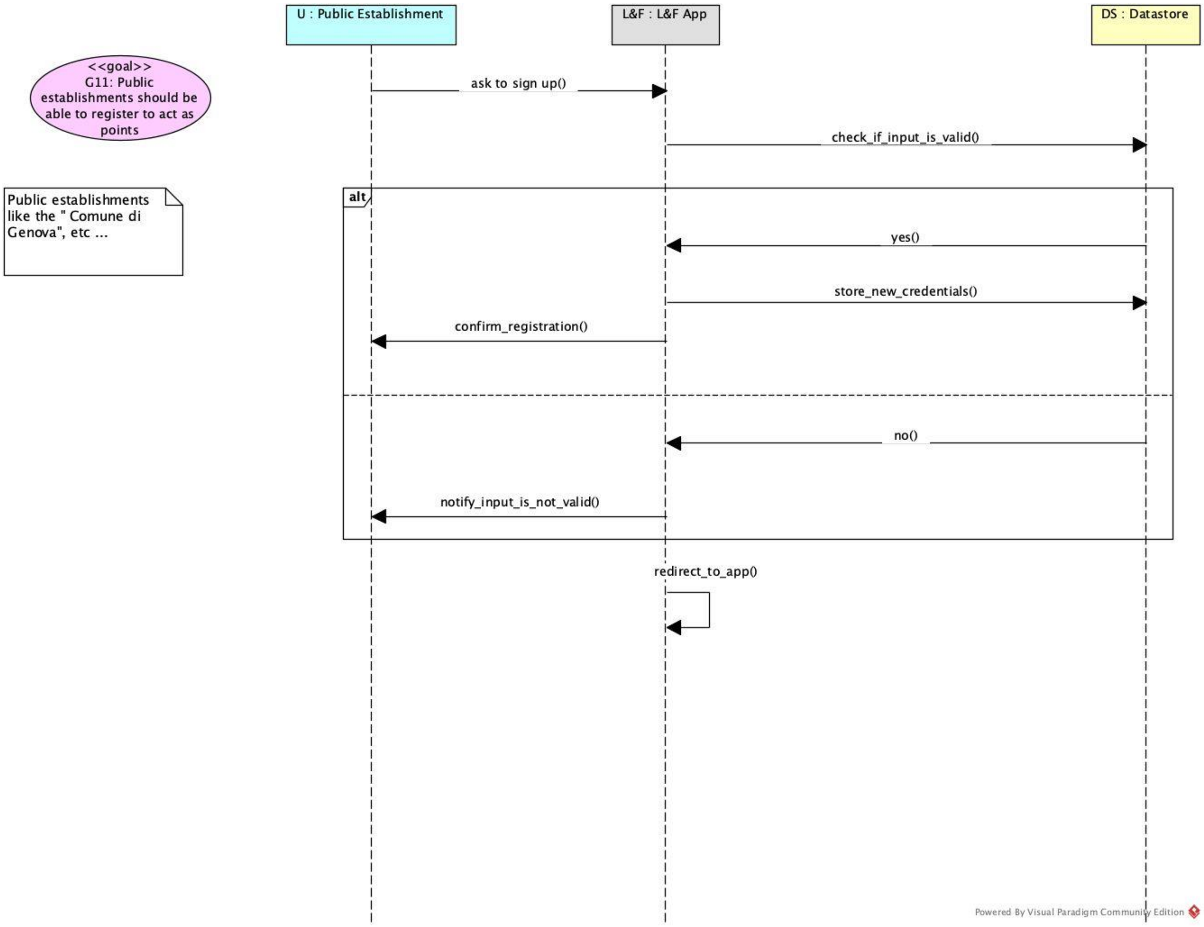
To use the system, people should accept rules and we addressed this goal in the presentation layer. Also, people can receive find or loss notifications as a pop-up in their application.
Certainly! When individuals sign up in our system, they gain the ability to report various actions such as finding an item, losing an item, or cancelling a previous loss report. We've integrated these functionalities into our business logic to ensure seamless operation and efficient handling of user interactions.

In the following, we draw the goals sequence diagrams which addressed for future implementation:

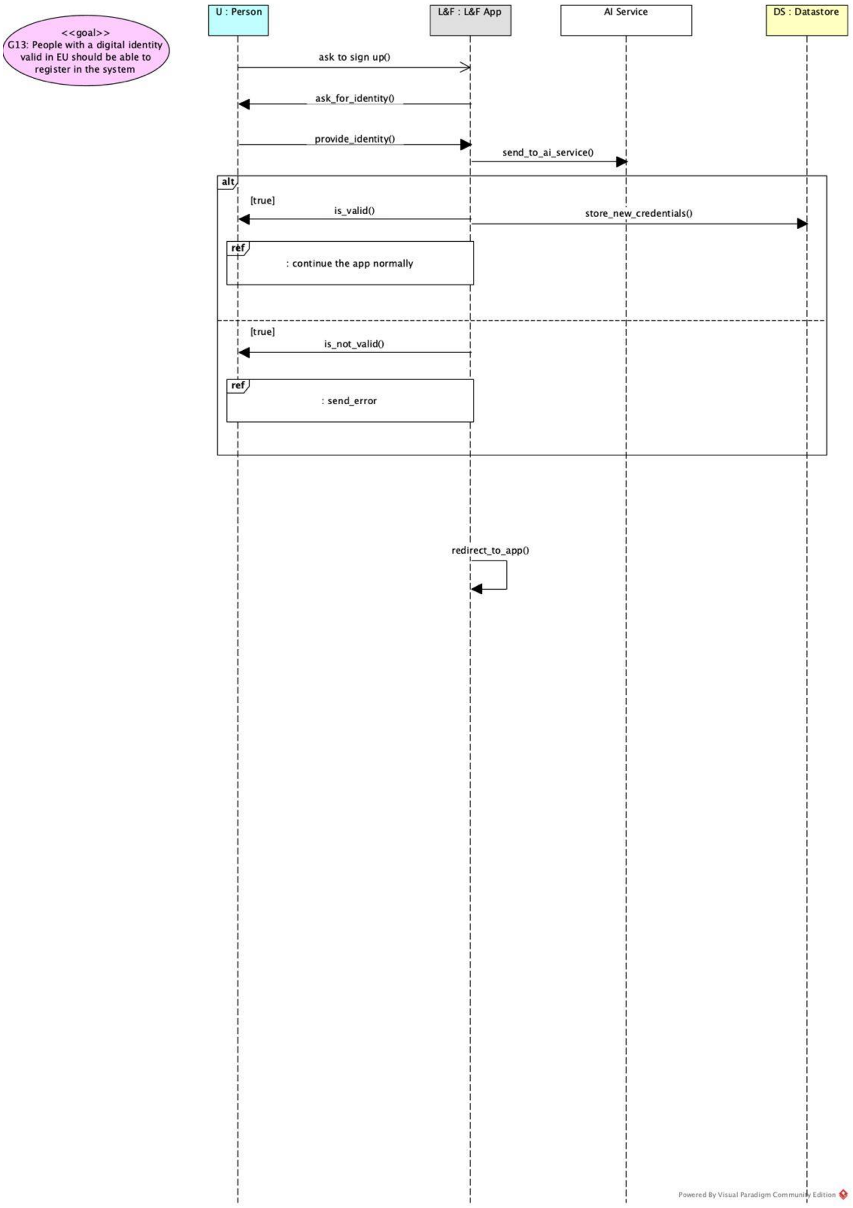
When a person register to the system, he have to first accept the rules about getting the non-claimed items, rewards, and all the other rules.


<<goal>>
G19: System users should be informed of the rules about getting the non-claimed items, rewards, etc, and they should accept them before to use the system
ROLES
U: User (people, police officers, second-hand shops)
Answer will be saved as a boolean value (true or false)
U : User
L&F : L&F App
DS : Datastore
ask_to_accept_rules()
alt
[true]
yes()
ref
: the app will work normally
[true]
no()
ref
: the app will stop
save_answer()
Powered By Visual Paradigm Community Edition

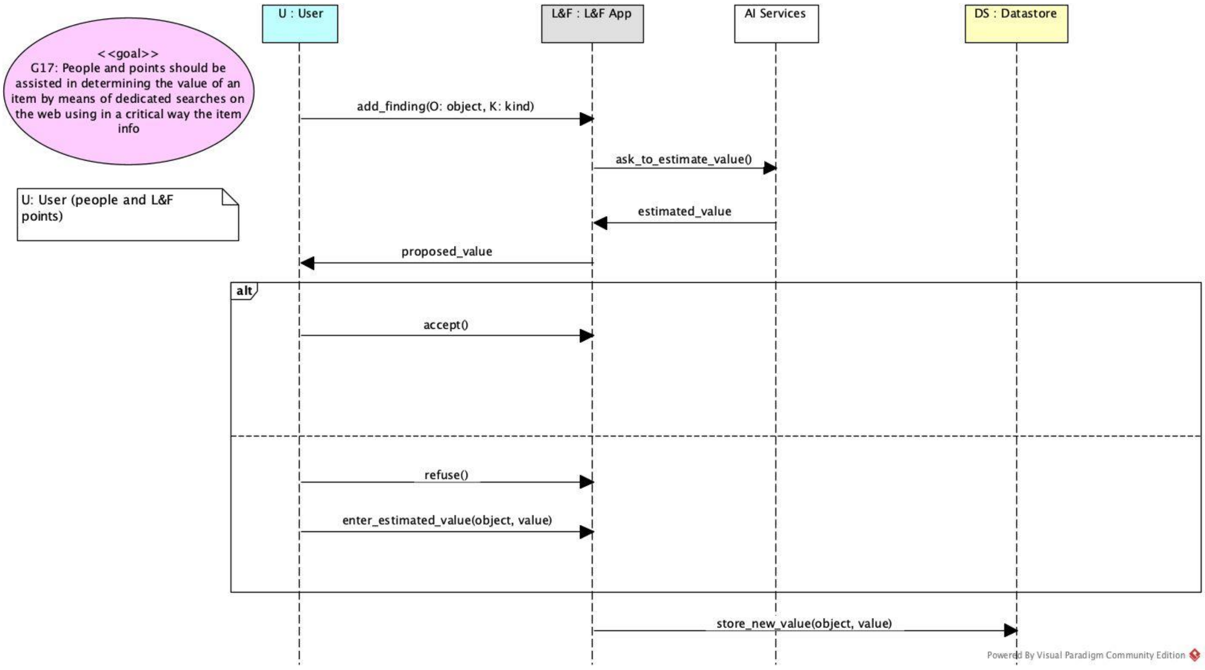
In our system, users can register as points of access. To obtain complete accessibility and the ability to manage various functions within the system, they must complete a registration form. As a point of access, users are granted full access to three other applications integrated within our system. This ensures that they have the necessary permissions and capabilities to effectively handle tasks and operations across the platform.


U : Public Establishment
L&F : L&F App
DS : Datastore
<<goal>>
G11: Public establishments should be able to register to act as points
Public establishments like the " Comune di Genova", etc ...
ask to sign up()
check_if_input_is_valid()
alt
yes()
store_new_credentials()
confirm_registration()
no()
notify_input_is_not_valid()
redirect_to_app()
Powered By Visual Paradigm Community Edition

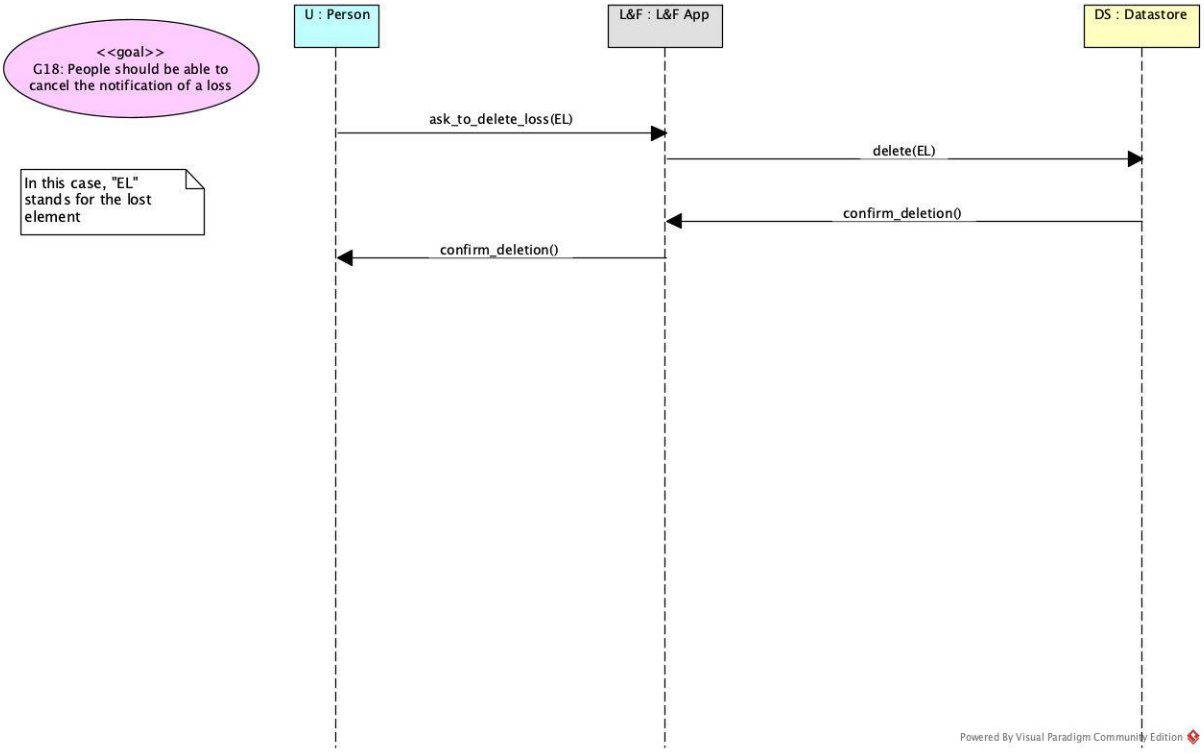
In order to maintain a secure system and prevent fraudulent activity, individuals are required to register using a digital identity that is valid within the European Union (EU). This entails providing an official identity card that is recognized and accepted by government authorities. By adhering to this requirement, we can ensure the integrity of our platform and safeguard against potential scams or unauthorized access.



By this diagram, we offer help to both individuals and points in assessing the value of an item by conducting thorough web searches. During these searches, we carefully analyze the item's information to determine its market value. This ensures that users have accurate information to make informed decisions about their items.

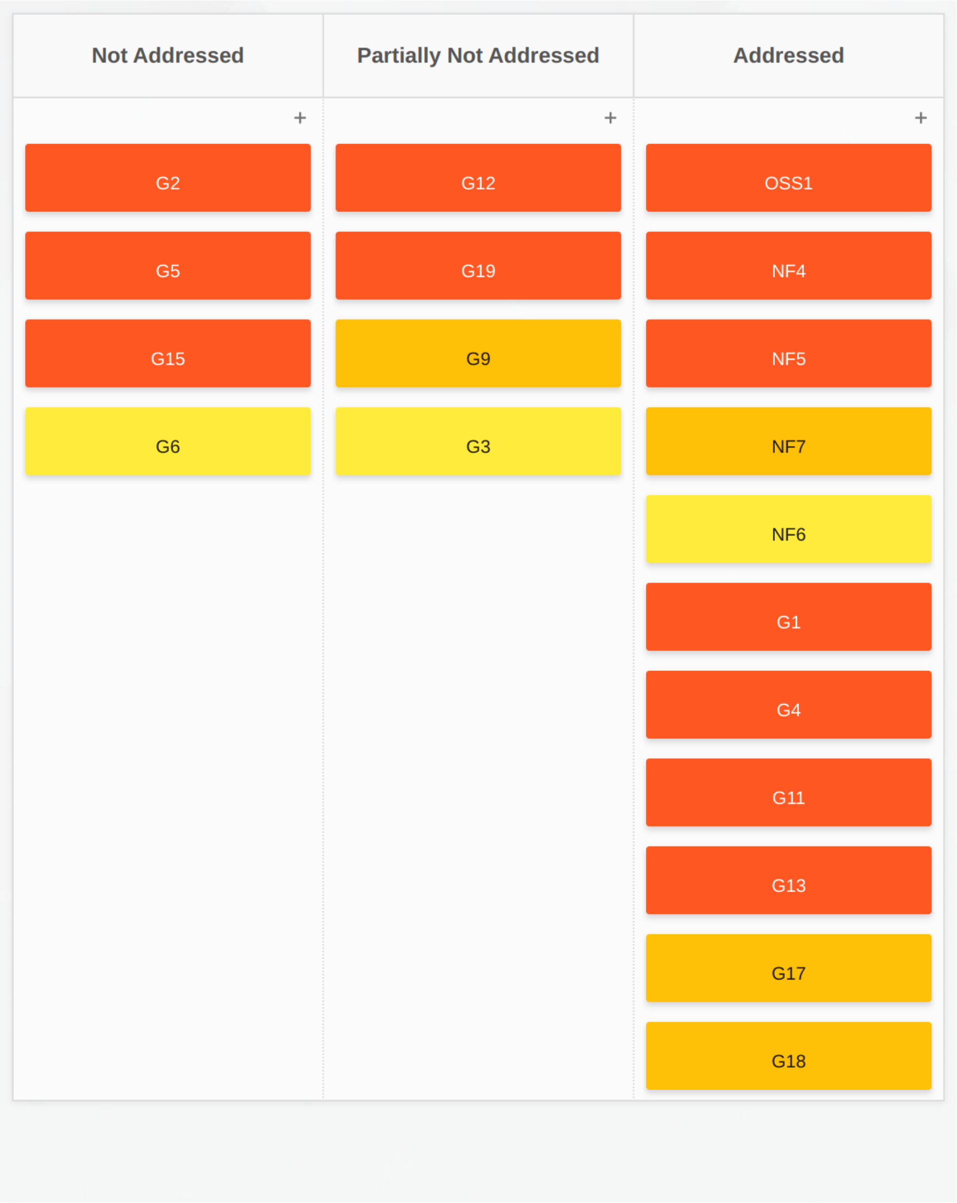

U : User
L&F : L&F App
AI Services
DS : Datastore
<<goal>>
G17: People and points should be assisted in determining the value of an item by means of dedicated searches on the web using in a critical way the item info
U: User (people and L&F points)
add_finding(O: object, K: kind)
ask_to_estimate_value()
estimated_value
proposed_value
alt
accept()
refuse()
enter_estimated_value(object, value)
store_new_value(object, value)
Powered By Visual Paradigm Community Edition

Users should have the ability to cancel a loss notification within the system, for instance, if they happen to find their lost item shortly after submitting a loss report or if they decide they do not wish to proceed with paying the loss amount.


U : Person
L&F : L&F App
DS : Datastore
<<goal>>
G18: People should be able to cancel the notification of a loss
In this case, "EL" stands for the lost element
ask_to_delete_loss(EL)
delete(EL)
confirm_deletion()
confirm_deletion()
Powered By Visual Paradigm Community Edition

Step 7: Perform analysis of current design

The table below outlines the design progress using the Kanban board:



Not Addressed	Partially Not Addressed	Addressed
G2	G12	OSS1
G5	G19	NF4
G15	G9	NF5
G6	G3	NF7
		NF6
		G1
		G4
		G11
		G13
		G17
		G18

Addressed:

To fulfill the requirements outlined in goals G1, G4, G13, and G18, individuals can register on the mobile application specifically tailored for their use. Registration entails providing a valid EU card for authentication purposes. Once registered, users gain the ability to report various actions such as finding an item, reporting a loss, or cancelling a previously reported loss. This streamlined process ensures compliance with EU regulations while providing users with efficient means to manage their interactions within the application.
We designed a system for points by addressing G11 to control overall system and other application. By put G17 in system people and point assisted in determining the value of an item by means of dedicated searches on the web using in a critical way the item info.

Partially Addressed:
In the next iteration, we are going to design 2 applications for police and second-hand shops that can cover G9 and G12.
In the next iteration, we refine the system to allow our people to browse the existing points with their position and opening hours by addressing G3.

For G19, we partially addressed accept rules only for people. Later we expand it for second-hand shop and police application as well.

Not Addressed:
G2: We decided to able people to consign a notified item to a point after creating their application. In the following iteration, with G5 after having notified a loss the loser should be allowed to browse the found items matching their lost one, and possibly confirm one of them and with G15 could track their loss or find report. We will implement a system the immediately after the notification of a finding, losers of matching items that can cover G6.

Conclusion

The implementation of the Architecture Design Document (ADD) method for developing a system from scratch proved to be an illuminating journey. It allowed us to apply our theoretical knowledge in a practical setting, resulting in the creation of a comprehensive architecture. Throughout this project, we recognized the significance of collaborative teamwork within contemporary organizations.

Furthermore, this experience prompted us to explore a plethora of solutions and tools tailored to different stages of the ADD method. We delved into various frameworks and design concepts, some of which were entirely novel to us. Additionally, we gained a deeper understanding of tools and technologies that we were already acquainted with but hadn't thoroughly explored previously.

In summary, this project provided us with a valuable opportunity to refine our skills and expand our knowledge by tackling a real-world problem. It instilled in us a deeper appreciation for the complexities inherent in system design and underscored the importance of adopting a systematic approach.