Lost Item detector

Introduction to Software Engneering

Sonia Jalal 22F-3712

Imama Kainat 22F-3661

Zainab Eman 22F-3738

[Email address]

GRP 34

Table of Contents

[Process Model 1](#_Toc132366138)

[Flexibility and Adaptability: 1](#_Toc132366139)

[Continuous Feedback: 1](#_Toc132366140)

[Iterative Development: 2](#_Toc132366141)

[Rapid Prototyping: 2](#_Toc132366142)

[Collaborative Development 2](#_Toc132366143)

[Faster Development Cycles: 2](#_Toc132366144)

[Iterative Testing: 2](#_Toc132366145)

[Continuous Testing and Improvement: 2](#_Toc132366146)

[Quality Focus: 2](#_Toc132366147)

[Time and Cost-Efficient: 2](#_Toc132366148)

[User-Centered: 3](#_Toc132366149)

[Requirements 3](#_Toc132366150)

[User requirements: 3](#_Toc132366151)

[System Requirements: 4](#_Toc132366152)

[Non-Functional Requirement: 5](#_Toc132366153)

[Functional Requirement: 6](#_Toc132366154)

[Design 6](#_Toc132366155)

[Use Case Daigram 6](#_Toc132366156)

[Class Daigram 8](#_Toc132366157)

[Activity Daigram-01 8](#_Toc132366158)

[Activity Daigram-02 9](#_Toc132366159)

[Activity Daigram-03 11](#_Toc132366160)

[Sequence Daigram-01 11](#_Toc132366161)

[Sequence Daigram-02 12](#_Toc132366162)

[Sequence Daigram-03 13](#_Toc132366163)

Lost Item Detector

**Technology That Can Help You Track and Find Your Lost Items**

**No more pacing the parking lot, checking the couch cushions, rifling through drawers**

A lost item detector is a system or device that is designed to help individuals locate misplaced or lost items. These devices typically use wireless technology such as Bluetooth, GPS, or Wi-Fi to communicate with a smartphone or other mobile device to help users track down lost items.

Some lost item detectors come in the form of small tags or stickers that can be attached to items such as keys, wallets, or bags. When the user misplaces the item, they can use their smartphone to trigger an alarm on the lost item detector, helping them locate the lost item.

Other lost item detectors may be standalone devices, such as trackers or beacons, that can be placed in a specific location or attached to an item to track its location in real-time. These devices may also have additional features such as geofencing, which can alert the user when the lost item has been moved outside of a specific area.

Overall, lost item detectors can be useful tools for individuals who frequently misplace their belongings or want to keep track of their valuables.

# Process Model

An Agile or iterative approach is more appropriate for the development of a lost item detector system for several reasons:

Flexibility and Adaptability: An Agile or iterative approach allows for flexibility and adaptability to change, which is critical for a project like a lost item detector system, where the technology and user needs may evolve over time. This approach allows for continuous feedback from users and stakeholders, enabling the development team to adjust and adapt the system to meet the evolving needs of the users.

Continuous Feedback: An Agile or iterative approach involves continuous feedback from the user or customer. In the case of a lost item detector system, the user's needs and expectations may evolve over time, and feedback can help ensure that the system meets those needs.

Iterative Development: An iterative approach allows for testing and iteration throughout the development cycle, rather than only at the end of the process. In the case of a lost item detector system, iterative testing can help ensure that the system is accurate and reliable, and can help identify and address any issues early on.

Rapid Prototyping: An Agile or iterative approach allows for rapid prototyping and development of the system. This can be especially beneficial for a lost item detector system, where quick development and testing cycles may be necessary to keep up with the evolving technology and user needs.

Collaborative Development: An Agile or iterative approach involves collaboration and communication between all stakeholders, including developers, testers, and users. This can help ensure that everyone is on the same page and working towards the same goal, which may be especially important for a lost item detector system that must be accurate and reliable.

Faster Development Cycles: An Agile or iterative approach typically has faster development cycles compared to the waterfall model. This may be important for a lost item detector system, which needs to be developed quickly to address the market needs.

Iterative Testing: In Agile or iterative approaches, testing occurs throughout the development cycle. This means that issues can be identified and addressed early in the development process, resulting in a better and more reliable product.

Continuous Testing and Improvement: An Agile or iterative approach allows for continuous testing and improvement of the system throughout the development cycle. This approach ensures that the system is reliable, accurate, and meets the user's needs. Continuous testing can identify issues early in the development cycle, reducing the cost and time required to fix them.

Quality Focus: An Agile or iterative approach emphasizes quality. This means that the team is focused on delivering a high-quality product that meets the user's needs.

Time and Cost-Efficient: An Agile or iterative approach can be time and cost-efficient, as it allows for quick iterations and testing cycles. This approach can help to identify issues early in the development cycle, which reduces the time and cost required to fix them. The shorter development cycles also allow for the development team to deliver functional components incrementally, providing stakeholders with tangible value throughout the development process.

User-Centered: An Agile or iterative approach is more user-centered and emphasizes delivering value to the users. This approach helps to ensure that the system meets the needs of the users and provides them with a positive experience.

Overall, an Agile or iterative approach may be more appropriate for the development of a lost item detector system due to its flexibility, continuous feedback, iterative development, rapid prototyping, and collaborative development. By following such an approach, developers can ensure that the system meets the user's needs and expectations while also being accurate, reliable, and up-to-date with the latest technology.

# Requirements

## User requirements:

* **Compatibility with multiple devices: The** software should be compatible with various devices, including smartphones, tablets, laptops, and other portable electronic devices.
* **Loud alarm:** The software should be able to activate a loud alarm on the lost item to help the user locate it quickly.
* **Compatibility with multiple devices:** The software should be designed to work with various operating systems and hardware configurations. It may require a certain version of the operating system or specific hardware features, such as GPS or Bluetooth capabilities.
* **GPS tracking:** The software should have access to GPS data and be able to process it in real-time. It may require a GPS chip or an internet connection to access GPS data.
* **Bluetooth connectivity:** The software should be compatible with Bluetooth 4.0 or higher, which is the standard for low-energy Bluetooth devices. It should be able to establish a secure connection with the lost item.
* **Real-time notifications**: The software should be able to send notifications instantly, which may require an internet connection or a push notification service.
* **Loud alarm:** The software should be able to produce a loud sound, which may require access to the device's speaker or the use of an external speaker.
* **Customization:** The software should provide a user interface that allows users to customize various settings, such as the range for notifications, the alarm sound, and the frequency of location updates.
* **Battery life management**: The software should be able to monitor the battery level of the lost item and use power-saving techniques to reduce energy consumption.
* **User-friendly interface:** The software should have a simple and intuitive user interface that is easy to navigate and understand.
* **Data privacy:** The software should adhere to standard privacy and security practices, such as encryption and data anonymization, to protect users' personal data.
* **Customer support:** The software should provide customer support channels, such as email, chat, or phone, to help users troubleshoot issues and answer questions

## System RequirementsTop of Form:

* **Accuracy**: The system should be able to accurately detect and locate lost items within a reasonable margin of error. The accuracy of the system should be high enough to minimize false positives and false negatives.
* **Range**: The system should be capable of detecting lost items over a wide range of distances and environments, such as indoors and outdoors, and in various weather conditions.
* **Battery Life:** The system should have a long battery life to ensure that it can operate continuously for extended periods of time without requiring frequent charging or replacement of batteries.
* **Portability:** The system should be small, lightweight, and easy to carry or attach to the items it is designed to detect, such as keys, wallets, or bags.
* **Connectivity:** The system should be able to connect to a smartphone or other mobile device via Bluetooth or other wireless technology to provide real-time location updates and alerts.
* **Durability:** The system should be designed to withstand wear and tear from daily use, as well as accidental drops or impacts.
* **Security:** The system should provide secure and encrypted communication between the lost item detector and the mobile device to ensure that sensitive information is not intercepted or compromised.
* **User-friendliness**: The system should be easy to set up and use, with clear instructions and intuitive user interfaces that do not require technical expertise to operate.

## Non-Functional Requirement:

* **Detecting Lost Items**: The lost item detector should be able to detect lost items and determine their location. This is the primary function of the system.
* **Alerting the User:** The lost item detector should alert the user when the lost item is detected. The system should use an audible or visual alert to notify the user of the item's location.
* **Tracking Lost Items**: The lost item detector should be able to track the location of the lost item over time. This can be useful in cases where the item is moved or stolen.
* **Compatibility**: The lost item detector should be compatible with various types of devices, such as smartphones, smartwatches, and tablets.
* **Battery Life**: The lost item detector should have a long battery life to ensure that the system can operate for extended periods without requiring frequent charging.
* **Range**: The lost item detector should have a reasonable range to detect the lost item. The system should be able to detect the item from a reasonable distance, such as 50-100 feet.
* **User Interface**: The lost item detector should have a user-friendly interface that is easy to navigate. The system should provide the user with clear instructions on how to use the system.
* **Device Management**: The lost item detector should allow the user to manage the devices that are connected to the system. This can include adding new devices, removing devices, and managing the device settings.
* **Security**: The lost item detector should have robust security features to protect the user's data and privacy. The system should use encryption to secure the data and prevent unauthorized access.

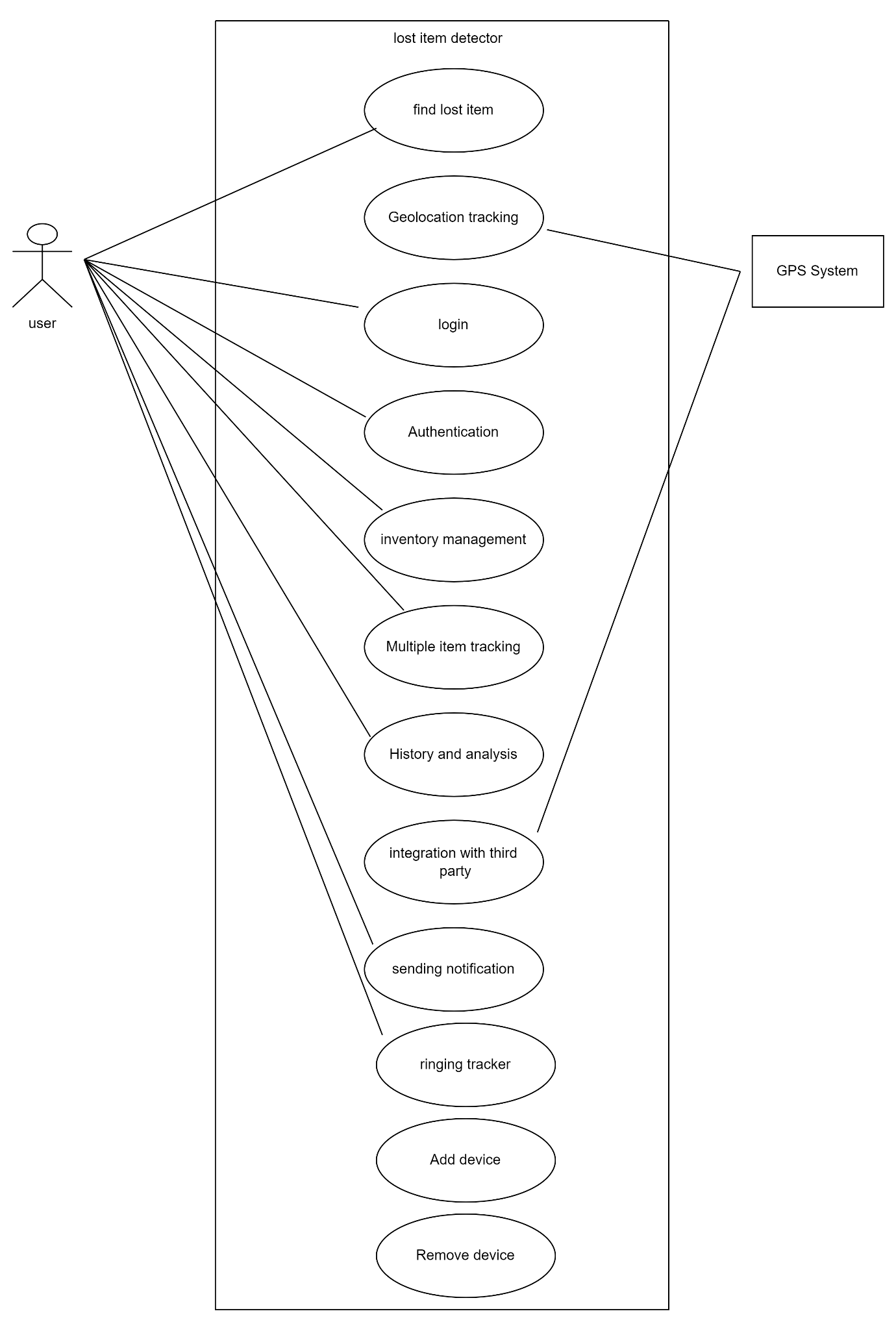
## Functional Requirement:

* **Item Detection:** The system should be able to detect lost items by using various sensors such as GPS, Bluetooth, Wi-Fi, or RFID.
* **Item Tracking:** The system should be able to track the lost item's location and provide real-time updates to the user.
* **Notification:** The system should send alerts and notifications to the user when a lost item is detected or when the item is near the user.
* **Item Retrieval:** The system should provide the user with directions and instructions on how to retrieve the lost item.
* **User Authentication:** The system should authenticate users and ensure that only authorized users can access the system's features and data.
* **Data Management**: The system should store and manage user data such as user profile information, lost item history, and device information.
* **Reporting**: The system should provide reporting features such as item location history, user activity logs, and system performance metrics.
* **Integration**: The system should be able to integrate with other systems and services such as social media, messaging, or e-commerce platforms.
* **Customization**: The system should allow users to customize their preferences such as notification settings, search parameters, and user interface.
* **Maintenance:** The system should provide maintenance features such as software updates, bug fixes, and data backups to ensure the system's functionality and security

# Design

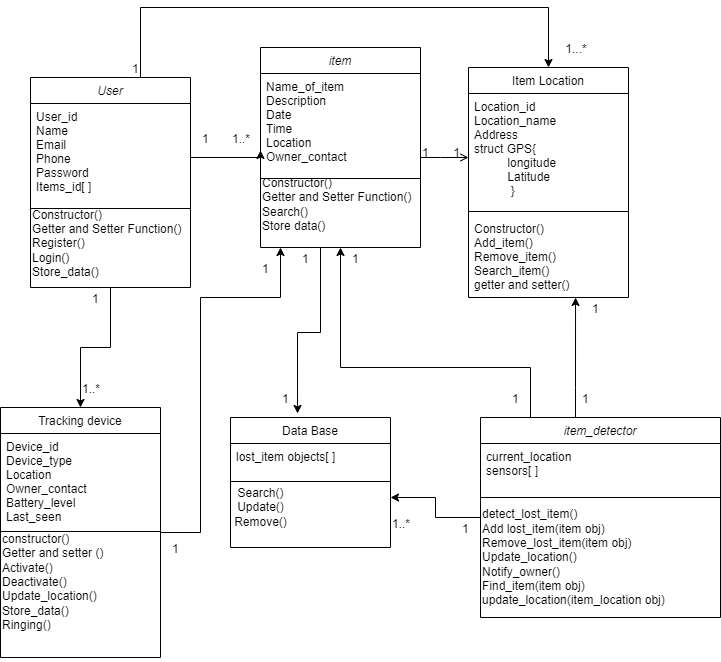
## Use Case Daigram

Use Case Diagram for system of Lost Item Detection is as follow:



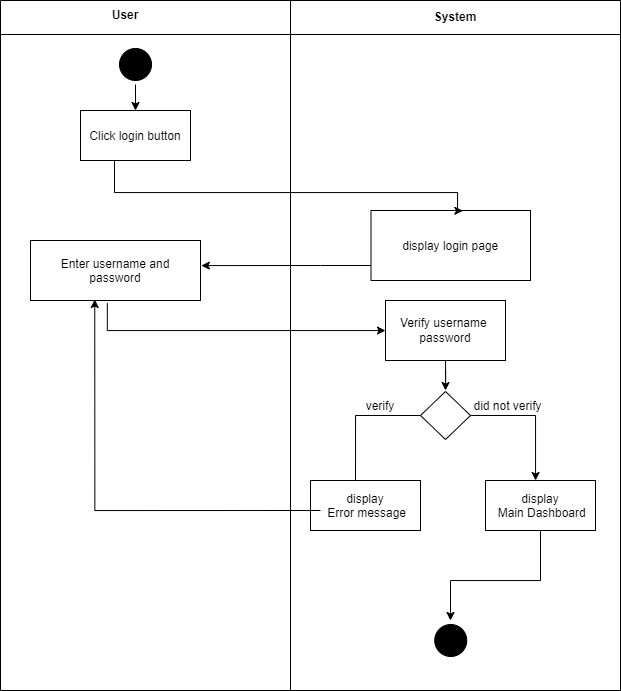
## Class Daigram

Class Diagram for system of Lost Item Detection is as follow:



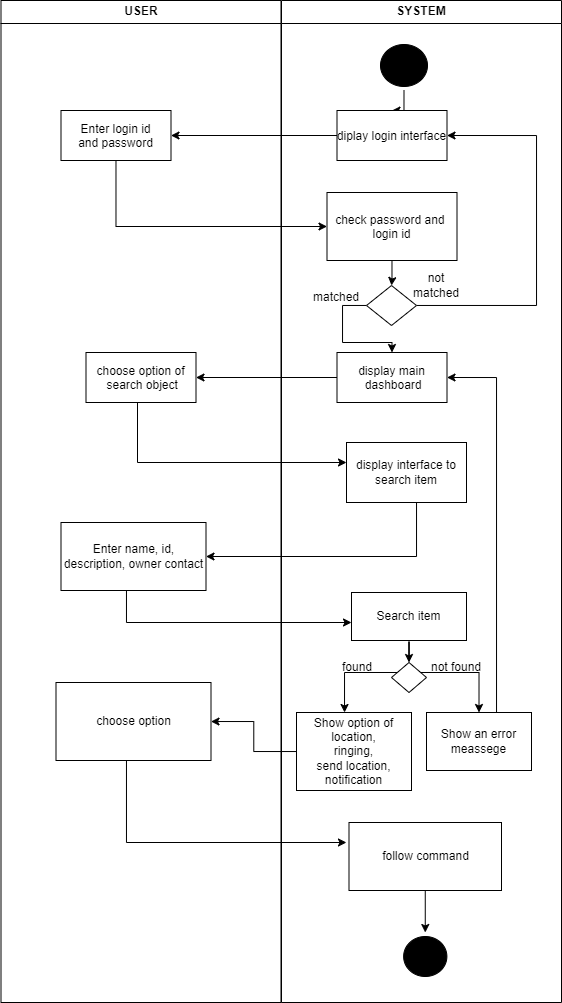
## Activity Daigram-01

Activity diagram for function of “Login”:



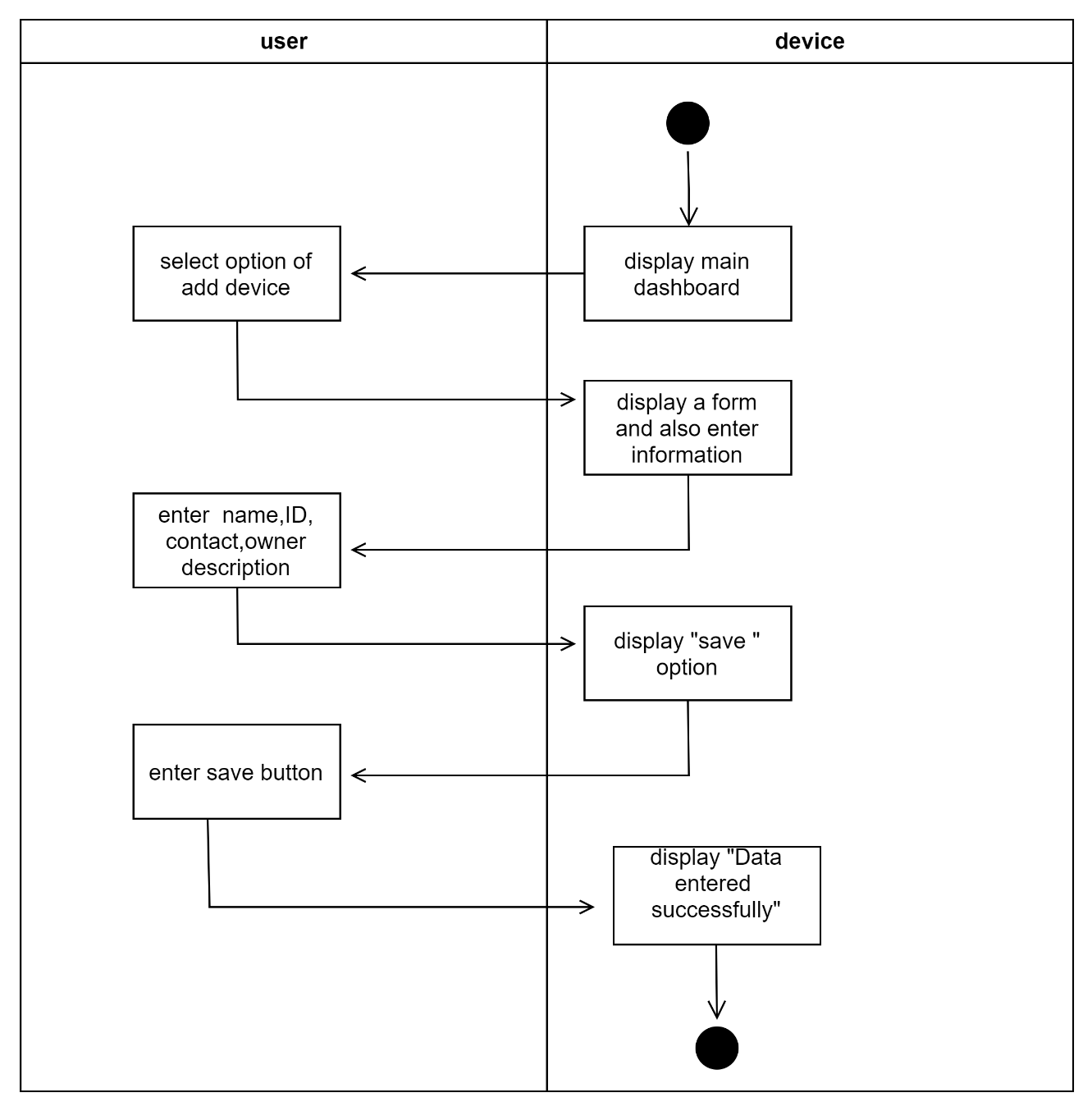
## Activity Daigram-02

Activity diagram for function of “lost item detection”:



## Activity Daigram-03

Activity diagram for function of “Add Device”:



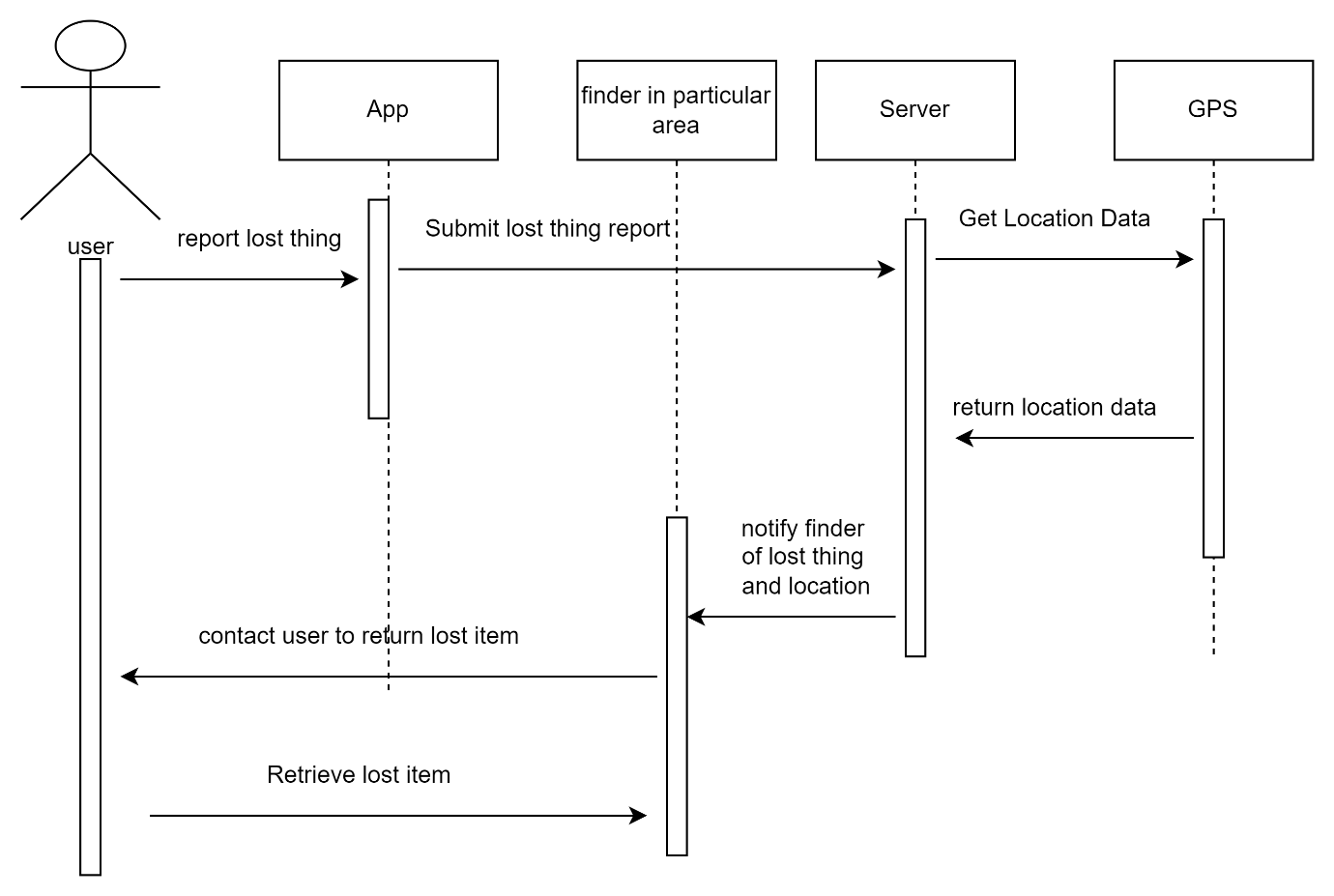
## Sequence Daigram-01

sequence diagram for” login function” :



## Sequence Daigram-02

sequence diagram for” finding lost item function” :



## Sequence Daigram-03

sequence diagram for” Adding new item ” :

