MIDDLE EAST TECHNICAL UNIVERSITY

DEPARTMENT OF COMPUTER ENGINEERING

TREELOGY

LEAF BASED TREE IDENTIFICATION SYSTEM

SOFTWARE DESIGN DESCRIPTIONS

Date of Issue:

03.06.2016

Project Advisor:

Assoc. Prof. Ahmet Oğuz AKYÜZ

Project Team:

Pay Inekereg

Project Team Members:

Burak BALCI, Çağrı ERCİYES, Emre AKIN, Eren ŞENER, İlke ÇUĞU

Table of Contents

List of Figures

1 INTRODUCTION

1.1 Purpose

The purpose of Software Design Description is to describe and visualize the design and architecture of TREELOGY by using different viewpoints.

This document aims to provide an understanding of the software system which is structured to the needs specified in Software Requirements Specification Document and how this system is expected to be built. This document will be the primary reference for implementation phase.

1.2 Scope

This document contains a complete description of the design of identification system by giving information about the overall software architecture and the design methods for each module of the software project.

This document will serve as a guideline through the implementation phase.

1.3 Context

The design viewpoints of the system are listed below the Design Viewpoints chapter. This document covers the most commonly used design viewpoints which are essential for software design. The design viewpoints that are included in this document are context, composition and interaction.

1.4 Summary

This document contains the software design descriptions for Android application for leaf based tree identification system. This document is prepared according to the ''IEEE Standard for Information Technology – System Design – Software Design Descriptions – IEEE 1016 – 2009''.

This software design description provides the details for how the TREELOGY software should be built. The details are represented by using graphical notations such as viewpoints, use case models, sequence diagrams and other supporting design information.

1.5 Design Languages

Design language is UML with supporting material, such as data dictionary, free or structured text, other kinds of diagrams, tables, pictures, and so on.

1.6 Authorship

Version	**Primary Author(s)**	**Description of Version**	**Date Completed**
1.0	Burak BALCI Çağrı ERCİYES Emre AKIN Eren ŞENER İlke ÇUĞU	Software Design Descriptions	03.06.2016

2 REFERENCES

- IEEE Standard for Information Technology – System Design – Software Design Descriptions – IEEE 1016 – 2009.
- ISO – IEC – IEEE 29148 reqeng.pdf available at https://cow.ceng.metu.edu.tr/Courses/?semester=20142&course=ceng350&cedit=0
- SRS Document of TREELOGY, TREElOGY_SRS_1.0, December, 2015.

3 GLOSSARY

Term	Definition
Software Requirements Specification (SRS)	A document that completely describes all of the functions of a proposed system and the constraints under which it must operate.
Software Design Description (SDD)	A document that describe and visualize the design and architecture of software by using different viewpoints
TREELOGY	Android application for leaf based tree identification system
UML	Unified Modeling Language
IEEE	Institute of Electrical and Electronics Engineers
Web Site	A place on the world wide web.

4 IDENTIFIED DESIGN STAKEHOLDERS AND THEIR CONCERNS

This section discusses the stakes and concerns of the various stakeholders.

4.1 Customer

Customer of TREELOGY is basically individuals which wants the entire system.

- Effectiveness is the extent to which the software-based information system (or architecture, or anything else) contributes to achieving the business goals.
- Schedule estimation
- Budgeting
- Feasibility and risk assessment
    - Technical feasibility
        - whether the proposed solution can be implemented with the available hardware, software, and technical resources
    - Economic feasibility
        - whether the benefits of the proposed solution outweigh the costs
    - Operational feasibility
        - whether the proposed solution is desirable within the existing managerial and organizational framework
- Requirements traceability
- Progress tracking

4.2 Developing Organization Management

The concerns of the developing organization management are similar to those of the customer, but on a various level. The customer is interested in the ways the architecture and the information system contribute to his business, and the developing organization management is concerned with how the project (of developing an architecture and an information system) conduces to its business.

Developing organization team of BSS is author of this description.

- Effectiveness
- Schedule estimation
- Budgeting (low costs, keeping people employed)
- Feasibility and risk assessment
- Requirements traceability
- Progress tracking

4.3 User

The users' interest in the architecture typically is ensuring that the final product will meet their functional and non-functional requirements. They are also interested in the impact that prospective requirements might have on the system.

User of the BSS is basically bicycle users in Izmir.

- Consistency with requirements and usage scenarios
- Non-functional requirements (performance, reliability etc.)
- Accommodate future requirements

4.4 Architect

Software architect is a computer manager or expert who makes high-level design choices and dictates technical standards, including software coding standards, tools, and platforms according to requirements.

Architect of the TREELOGY is basically authors of this description.

- Support of trade-off analysis
- Requirements traceability
- Completeness, consistency of architecture
- Context definition

4.5 Developer

Developers typically use the architecture as a reference for developing the system and assembling system components.

- Sufficient detail for design
- Reference for selecting/assembling components
- Maintain compatibility with existing systems

4.6 Maintainer

The maintainers are mostly concerned with how easy it will be to change the system in the future.

- Maintain compatibility with existing systems
- Guidance on software modification
- Guidance on architecture evolution
- Non-functional requirements (performance, reliability etc.)

5 IDENTIFIED DESIGN CONCERNS

Identified Design Concerns related to viewpoints is represented in their own Design Viewpoint section.

6 DESIGN VIEW

One of the design concerns is that the extendibility of the project. All of the classes and relations are designed to make the system as flexible and extendable as possible.

A context view and interaction view of the product is explained and also it is supported by use case descriptions. A composition view and structure view of the product shows organizational components with each other. A pattern use view of the product shows general architectural patterns. A logical view of the product is explained and also it is supported by diagrams. Relationships of the classes are easily perceived. Dependency and information viewpoints show possible future problems and how the information is stored and shared among the users. Finally, state dynamic views shows the state transitions and flow of actions with diagrams.

The design views are overseen by the design viewpoints which are used in section 7.

7 DESIGN VIEWPOINTS

7.1 Introduction

In this part we try to give various viewpoints about the system to describe from all angles to the stakeholders for understanding the system totally.

7.2 Context Viewpoint

Context viewpoint is mainly concerned with the services the system provides and the users of the system and describes the relationships, dependencies, and interactions between the system and its environment.

Context diagram, use case diagram and use case descriptions are presented to make a picture of context viewpoints by showing the relationships and interactions between the system and actors of system.

USE CASE DIAGRAM AND CORRESPONDING USE CASE DESCRIPTIONS

The following diagram represents use cases to show a general view of the correlations and interactions between the system and the actors.

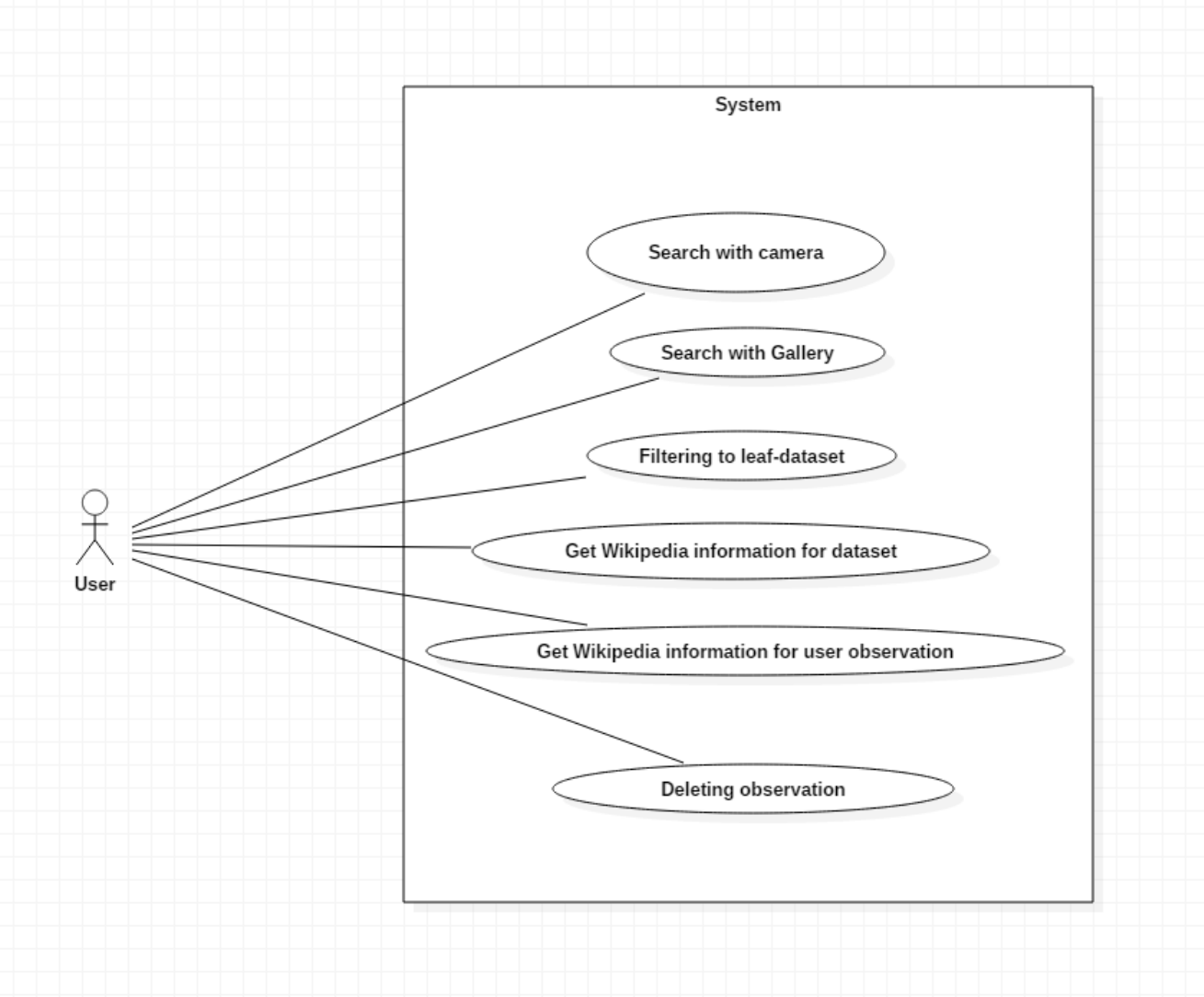

FIGURE 1: Use Case Diagram

Use Case Names Respectively:

1. Search with camera
2. Search with gallery
3. Filtering to leaf-dataset
4. Get Wikipedia information for dataset
5. Get Wikipedia information for user observations
6. Deleting Observations

Use Case Descriptions

Use Case Number	1
Use Case	Search with Camera
Summary	Taking image with camera is returned back to user with 5 different tree class results.
Actor	User
Trigger	User clicks “New observation” button in the Entrance activity.
Primary Scenario	1. User goes to first tab in the Main activity, and click Camera button 2. User takes a leaf photo with Camera and clicks search button. 3. Image is sent to Server, and respectively background elimination, stem removal process are handled and final image is sent to Caffe. 4. Caffe return 5 results, SVM returns 1 result and we merge them. 5. Final 5-results are sent to user device, and they are displayed. 6. Use case terminates.
Alternative Scenario	**None**
Exceptional Scenario	**None**
Pre-Conditions	User has Network Connection
Post-Conditions	5 tree results are shown to the user with latin name and leaf images.
Assumptions	User’s Android device has Camera.

Use Case Number	2
Use Case	Search with Gallery
Summary	Selecting image from is returned back to user with 5 different tree class results.
Actor	User
Trigger	User clicks “New observation” button in the Entrance activity.
Primary Scenario	1. User goes to first tab in the Main activity, and click Gallery button 2. User selects a leaf image from Gallery and clicks search button. 3. Image is sent to Server, and respectively background elimination, stem removal process are handled and final image is sent to Caffe. 4. Caffe return 5 results, SVM returns 1 result and we merge them. 5. Final 5-results are sent to user device, and they are displayed. 6. Use case terminates.
Alternative Scenario	**None**
Exceptional Scenario	**None**
Pre-Conditions	**None**
Post-Conditions	5 tree results are shown to the user with latin name and leaf images.
Assumptions	**None**

Use Case Number	3
Use Case	Filtering to leaf-dataset
Summary	Filters to the dataset that user queries
Actor	User
Trigger	User clicks to “Leaves and Trees” button in the Entrance activity.
Primary Scenario	1. User goes to second tab in the Main activity, and enters what s/he is looking for the search area. 2. Dynamically filter is applied to the dataset. 3. Filtering results are shown to the user instantly. 4. Use case terminates.
Alternative Scenario	**None**
Exceptional Scenario	**None**
Pre-Conditions	**None**
Post-Conditions	Filtering results are displayed
Assumptions	**None**

Use Case Number	4
Use Case	Wikipedia information for dataset
Summary	User gets Wikipedia information using dataset
Actor	User
Trigger	User clicks to “Leaves and Trees” button in the Entrance activity.
Primary Scenario	1. User goes to second tab in the Main activity 2. User clicks list item on the List view. 3. Wikipedia site is opened in our application with our leaf and tree images. 4. Use case terminates.
Alternative Scenario	**None**
Exceptional Scenario	**None**
Pre-Conditions	**None**
Post-Conditions	Wikipedia site is opened in the application
Assumptions	**User has Network connection**

Use Case Number	5
Use Case	Wikipedia information for user observation
Summary	User gets Wikipedia information using her/his observations
Actor	User
Trigger	User clicks to “My Observations” button in the Entrance activity.
Primary Scenario	1. User goes to third tab in the Main activity 2. User clicks list item on the List view. 3. Wikipedia site is opened in our application with our leaf and tree images. 4. Use case terminates.
Alternative Scenario	**None**
Exceptional Scenario	**None**
Pre-Conditions	**None**
Post-Conditions	Wikipedia site is opened in the application
Assumptions	**User has Network connection**

Use Case Number	6
Use Case	Deleting observation
Summary	User deletes her/his observations
Actor	User
Trigger	User clicks to “My Observations” button in the Entrance activity.
Primary Scenario	1. User goes to third tab in the Main activity 2. User long-presses list item on the List view. 3. In the opening small window, user clicks delete button. 4. User observation is deleted from gallery and file system. 5. Use case terminates.
Alternative Scenario	**User clicks outside of window and nothing will happen.**
Exceptional Scenario	**None**
Pre-Conditions	**None**
Post-Conditions	User observation is deleted from gallery and file system.
Assumptions	**None**

7.3 Composition Viewpoint

The composition viewpoint of the system concerns about the architecture of the whole system.

For this aim, for the logical representation UML Component Diagram and for the physical representation UML Deployment Diagram is provided.

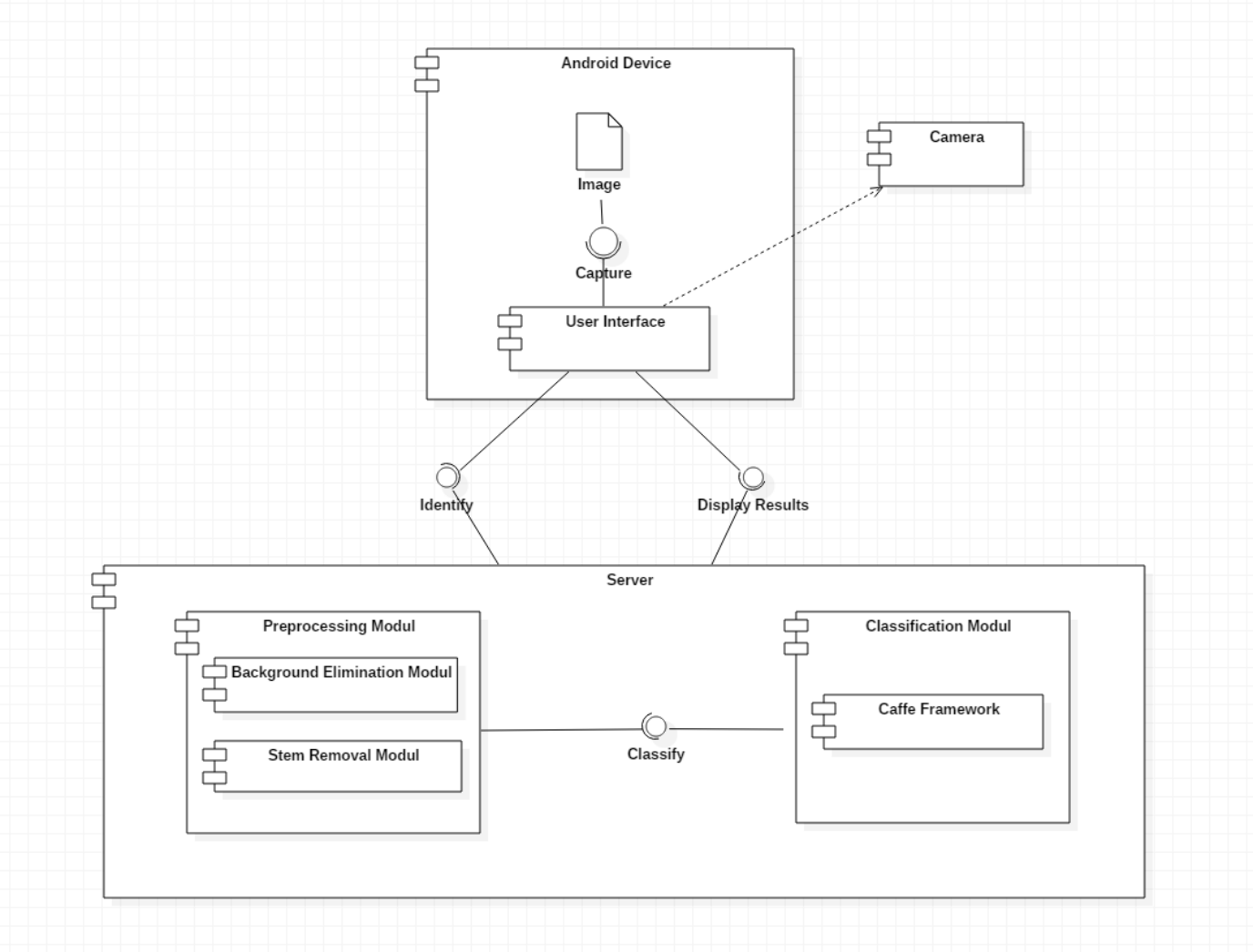

FIGURE 2: Component Diagram

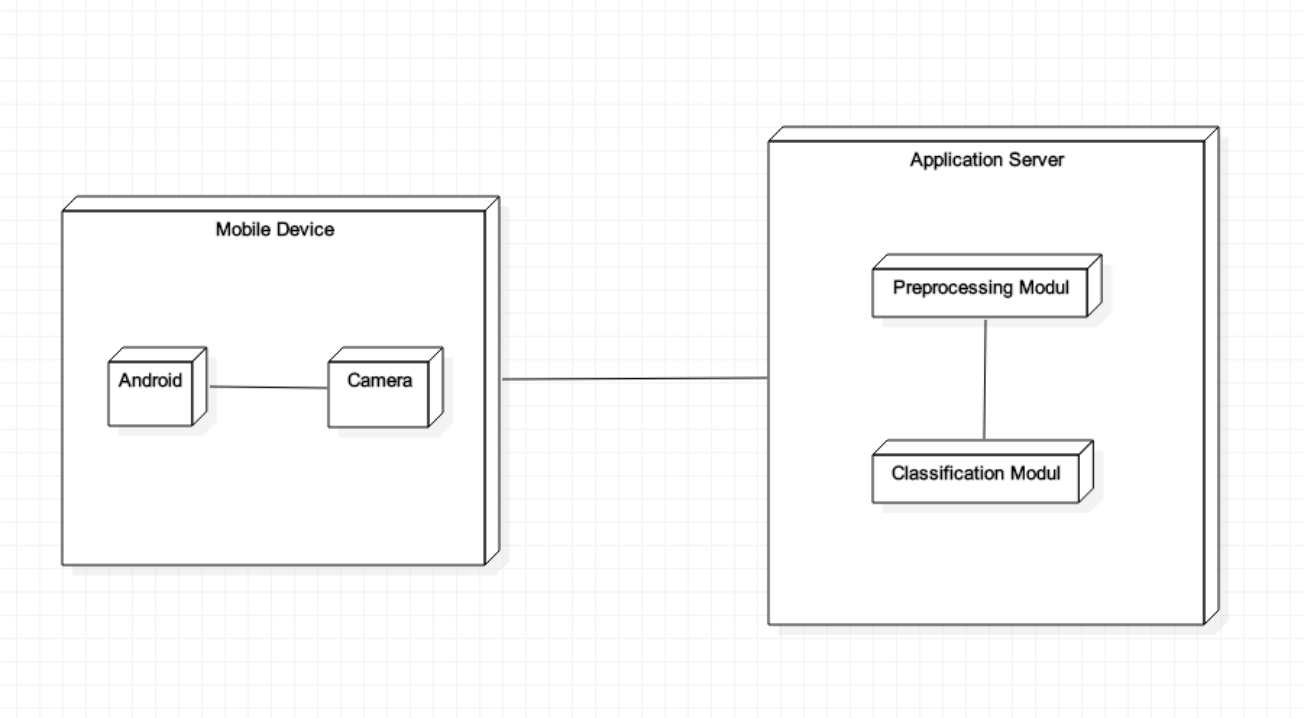

FIGURE 3: Deployment Diagram

7.4 Interaction Viewpoint

Interaction Viewpoint is mainly concerned with object communication and messaging.

The interaction viewpoint of the system represents how each object operates one another and in what order. The sequence diagrams in below show objects' interactions in time sequence. These sequence diagrams are related with the use-case diagrams that are shown in Software Requirements Specification document and use-case descriptions that are shown in 5.2.

Sequence Diagram Names Respectively:

1. Search with camera
2. Search with gallery
3. Filtering to leaf-dataset
4. Get Wikipedia information for dataset
5. Get Wikipedia information for user observations
6. Deleting Observations

1. Search with camera Sequence Diagram

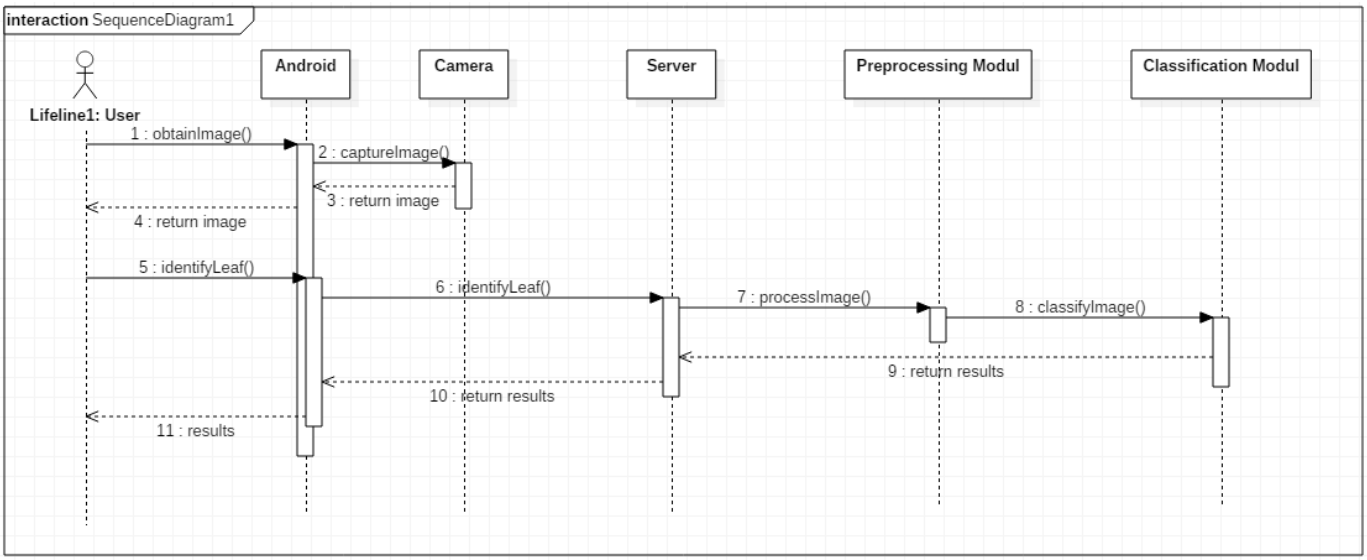

FIGURE 4: Search with camera Sequence Diagram

2. Search with gallery Sequence Diagram

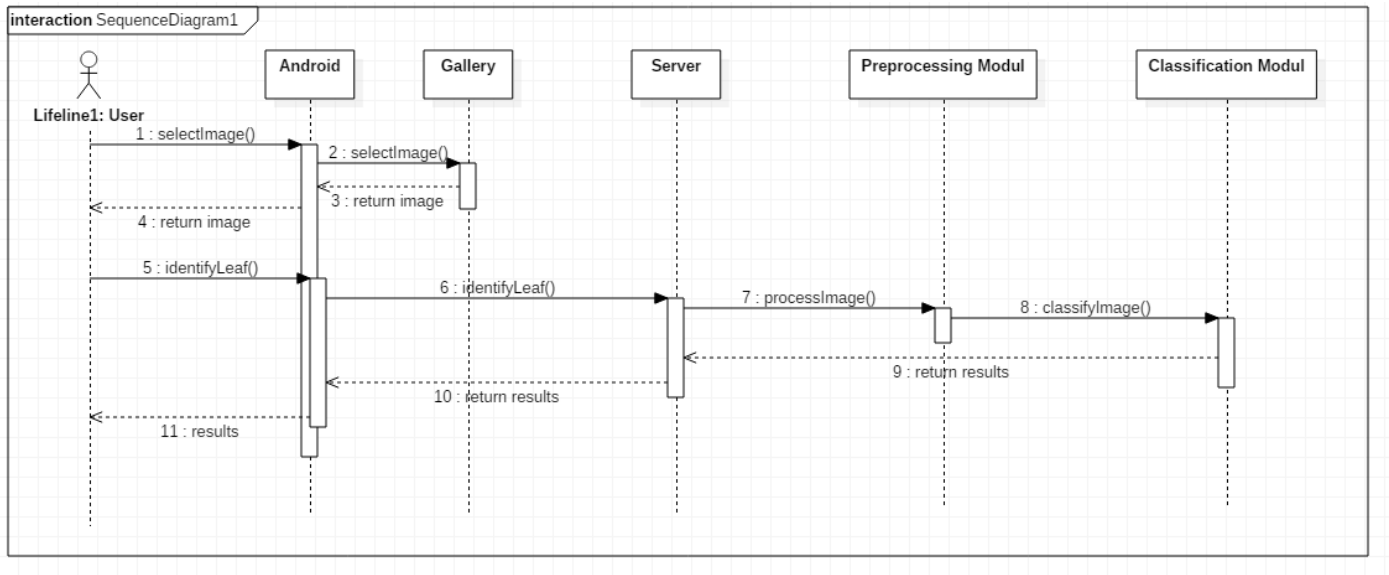

FIGURE 5: Search with gallery Sequence Diagram

3. Filtering to leaf dataset Sequence Diagram

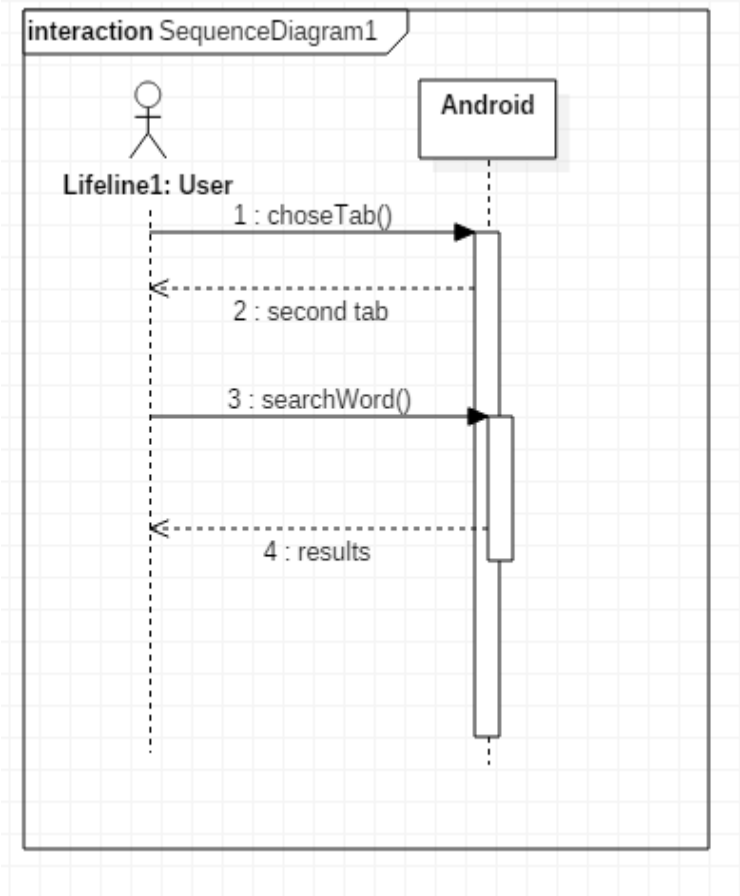

FIGURE 6: Filtering to leaf dataset Sequence Diagram

4. Get Wikipedia information for dataset Sequence Diagram

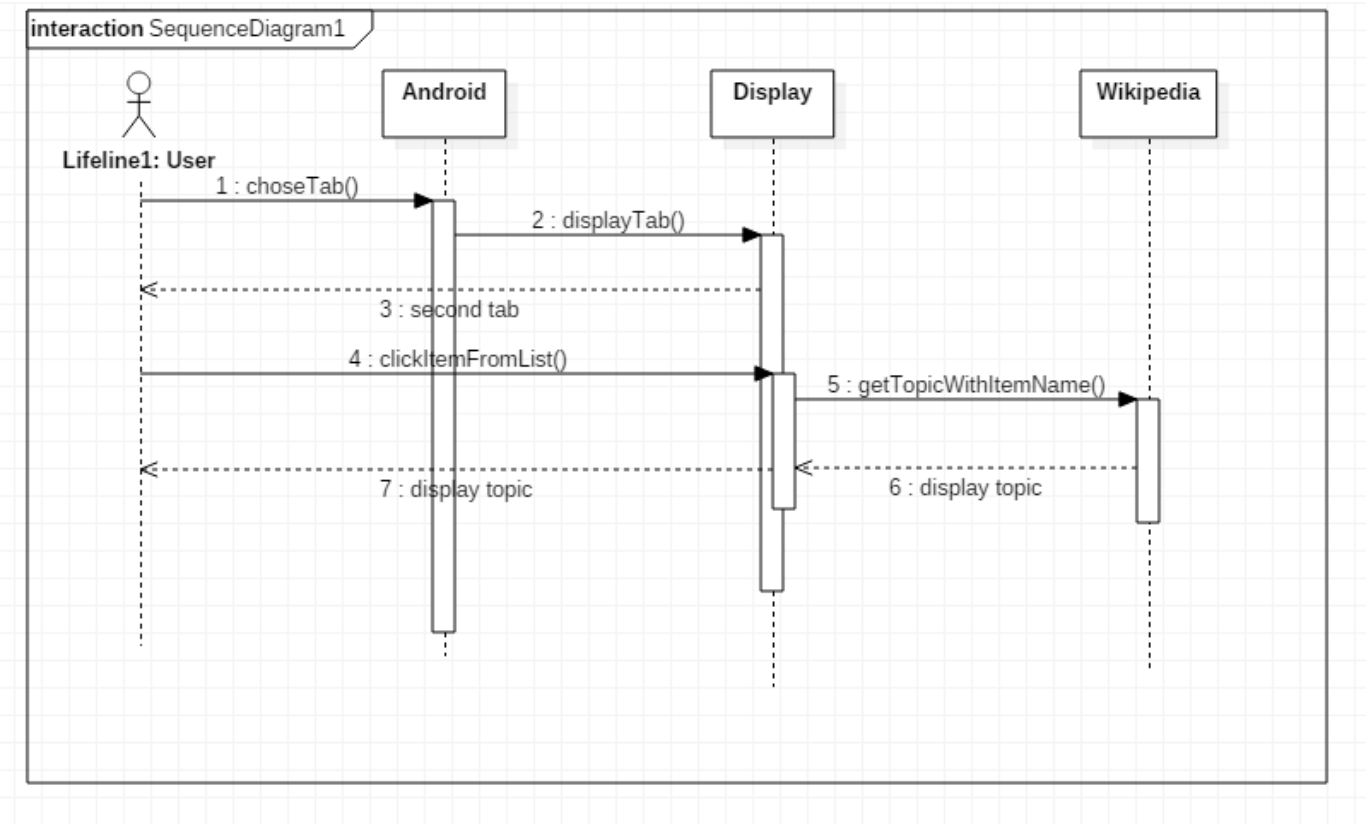

FIGURE 7: Get Wikipedia Information for dataset Sequence Diagram

5. Get Wikipedia information for user observation Sequence Diagram

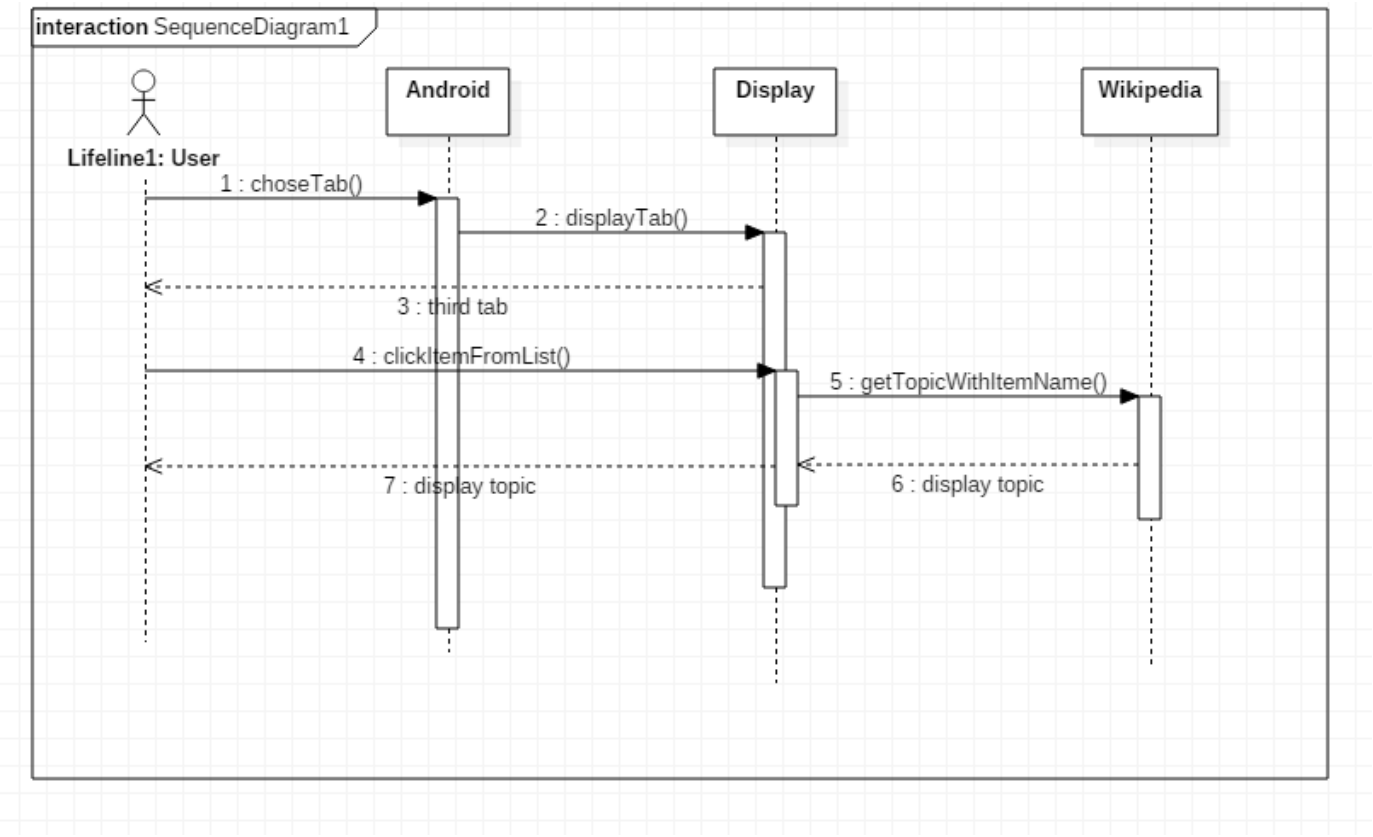

FIGURE 8: Get Wikipedia information for user observation Sequence Diagram

6. Deleting Observation Sequence Diagram

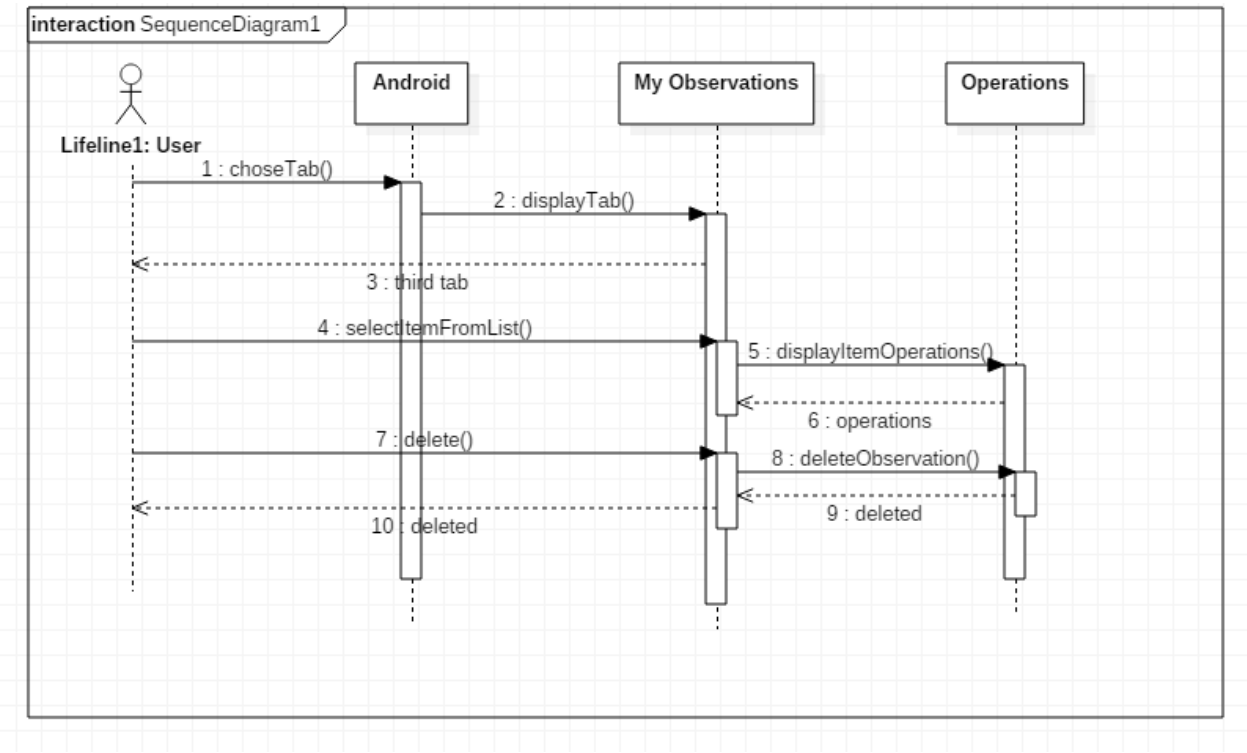

FIGURE 9: Deleting Observation Sequence Diagram

8 DESIGN RATIONALE

Design choices are made according to some significant features like sustainability, integrating another project. It can be updateable according to stakeholders and users requirements. Each function in the software will be commented so that it can be understandable for the other developers and also they can change the code by help of these comments.