

International School

**Capstone Project 2**

CMU-SE 451

**Architecture Document**

**Version 2.4**

**Date:** 13/05/2023

**Craft Village Pollution Monitor System**

**Submitted by**

**Ca, Van Cong Le**

**Huy, Bui Duc**

**Phuc, Hua Hoang**

**Trung, Nguyen Thanh**

**Nhan, Huynh Ba**

**Approved by**

**Ph.D. Nguyen Thanh Binh**

**Proposal Review Panel Representative:**

Name Signature Date

**Capstone Project 2 - Mentor:**



Name Signature Date

**PROJECT INFORMATION**

|  |  |  |  |
| --- | --- | --- | --- |
| **Project acronym** | CVPMS | | |
| **Project Title** | Craft Village Pollution Monitor System | | |
| **Start Date** | 01/03/2023 | **End Date** | 15/05/2023 |
| **Lead Institution** | International School, Duy Tan University | | |
| **Project Mentor** | Ph.D. Thanh Binh, Nguyen | | |
| **Scrum master / Project Leader & contact details** | Ca, Van Cong Le  *Email:* [cascabusiness@gmail.com](about:blank)  *Tel:* 0352707895 | | |
| **Partner Organization** |  | | |
| **Project Web URL** | [*https://github.com/Casca113s2/craft-village-pollution-monitor-system*](https://github.com/Casca113s2/craft-village-pollution-monitor-system) | | |
| **Team members** | **Name** | **Email** | **Tel** |
| 25211207666 | Ca, Van Cong Le | [cascabusiness@gmail.com](http://cascabusiness@gmail.com) | 0352707895 |
| 25211215894 | Huy, Bui Duc | [duchuyltt122@gmail.com](http://duchuyltt122@gmail.com) | 0818648090 |
| 25211204084 | Phuc, Hua Hoang | [phuchuho0402@gmail.com](http://phuchuho0402@gmail.com) | 0905639682 |
| 25211215133 | Trung, Nguyen Thanh | [nguyenttrung2601@gmail.com](http://nguyenttrung2601@gmail.com) | 0774496838 |
| 25211203702 | Nhan, Huynh Ba | <huynhbanhan1491999@gmail.com> | 0935430785 |

DOCUMENT NAME

|  |  |  |  |
| --- | --- | --- | --- |
| **Document Title** | Architecture Document | | |
| **Author(s)** | Bui Duc Huy | | |
| **Date** | 13/05/2023 | **File Name** | C2SE.01\_CVPMS\_Architecture-Document\_v2.4.docx |

REVISION HISTORY

| **Version** | **Date** | **Comments** | **Author** | **Approval** |
| --- | --- | --- | --- | --- |
| 1.0 | 21/08/2022 | Initial Release | Bui Duc Huy |  |
| 1.1 | 17/10/2022 | Update C&C, Module View diagrams | Bui Duc Huy |  |
| 2.0 | 14/03/2023 | Update Activity, Module View diagrams | Hua Hoang Phuc |  |
| 2.1 | 29/03/2023 | Update Activity, Context Diagram | Hua Hoang Phuc |  |
| 2.2 | 13/04/2023 | Update Activity, Module View diagrams, | Hua Hoang Phuc |  |
| 2.3 | 28/04/2023 | Overview, Prediction AI | Bui Duc Huy  Van Cong Le Ca |  |
| 2.4 | 13/05/2023 | Allocation View, Prediction AI | Huynh Ba Nhan  Bui Duc Huy |  |

**Approve Document:** Sign in to approve the document

|  |  |  |  |
| --- | --- | --- | --- |
| **Mentor** | Binh, Nguyen Thanh | Date | 13/05/2023 |
| Sign |  |
| **Scrum Master** | Ca, Van Cong Le | Date | 13/05/2023 |
| Sign |  |
| **Scrum Member** | Huy, Bui Duc | Date | 13/05/2023 |
| Sign |  |
| **Scrum Member** | Phuc, Hua Hoang | Date | 13/05/2023 |
| Sign |  |
| **Scrum Member** | Trung, Nguyen Thanh | Date | 13/05/2023 |
| Sign |  |
| **Scrum Member** | Nhan, Huynh Ba | Date | 13/05/2023 |
| Sign |  |

Table Of Contents

[1. Introduction 7](#_Toc135326321)

[1.1. Purpose 7](#_Toc135326322)

[1.2. Business needs 7](#_Toc135326323)

[1.3. Proposed solution 7](#_Toc135326324)

[1.4. Business drivers 8](#_Toc135326325)

[1.5. Project goal 8](#_Toc135326326)

[2. Architectural drivers 9](#_Toc135326327)

[2.1. Functional requirements 9](#_Toc135326328)

[2.2. Business constraints 10](#_Toc135326329)

[2.3. Technical constraints 10](#_Toc135326330)

[2.4. Quality Attribute 10](#_Toc135326331)

[2.5. Context Diagram 13](#_Toc135326332)

[3. Activity Diagram 16](#_Toc135326333)

[4. C&C view 20](#_Toc135326334)

[5. Module View 21](#_Toc135326335)

[6. Allocation view 22](#_Toc135326336)

[7. Proposed architecture for Image Classification 23](#_Toc135326337)

[8. Proposed architecture for Random Forest Classification 24](#_Toc135326338)

[9. References 25](#_Toc135326339)

**Table Of Figures**

[**Figure 1.** Craft Village Pollution Monitor System 7](#_Toc135326431)

[**Figure 2.** System Context Diagram Overview 13](#_Toc135326432)

[**Figure 3.** Personal System Context Diagram 13](#_Toc135326433)

[**Figure 4.** Household System Context Diagram 14](#_Toc135326434)

[**Figure 5.** Authority System Context Diagram 14](#_Toc135326435)

[**Figure 6.** Admin System Context Diagram 15](#_Toc135326436)

[**Figure 7.** Activity Diagram (Personal) 16](#_Toc135326437)

[**Figure 8.** Activity Diagram (Household) 17](#_Toc135326438)

[**Figure 9.** Activity Diagram (Authority) 18](#_Toc135326439)

[**Figure 10.** Activity Diagram (Admin) 19](#_Toc135326440)

[**Figure 11.** Component & connector view 20](#_Toc135326441)

[**Figure 12.** Module view 21](#_Toc135326442)

[**Figure 13.** Allocation view 22](#_Toc135326443)

[**Figure 14.** Image Classification Architecture 23](#_Toc135326444)

[**Figure 15.** Random Forest Classification 24](#_Toc135326445)

**Table Of Tables**

[**Table 1.** Functional requirements 9](#_Toc135326462)

[**Table 2.** Quality Attributes: Availability 10](#_Toc135326463)

[**Table 3.** Quality Attributes: Performance 11](#_Toc135326464)

[**Table 4.** Quality Attributes: Performance 12](#_Toc135326465)

[**Table 5.** Quality Attributes: Usability 12](#_Toc135326466)

# Introduction

## Purpose

This document will cover the following information:

* Brief description of the project (project overview, business goals, general constrains about technical and business problems).
* Architectural drivers (functional requirements, quality attributes and constraints).
* Architectural design (C&C View type, Module View type, Allocation View type).

## Business needs

* Desiring to help people have more awareness of pollution from craft villages, a system that will allow people to submit data of craft village and keep track its pollution status.
* A system that will provide the user the abilities such as take a survey, include the image of the pollution, view survey history, etc.
* An AI system will also provide the application to analyze the pollution based on the user's provided image.
* An AI takes the production information and then give the prediction of pollution types that the craft village will facing.
* The system should also give the administrator the ability to manage their user and data.

## Proposed solution



**Figure 1.** Craft Village Pollution Monitor System

Our team will wrap around the above problems and help everyone monitor and resolve the pollution problems from their craft village area more effectively. Some aspects that will make our system that the people will find much more effective:

* Our system will help everyone to monitor your local craft village despite where your location. This means whether you stay in a big city or a small village at the top of the mountain our system will still function.
* Our system will provide a function that lets people make an instant report to the local environment department.
* Our system will detect all kinds of pollution instead of focusing on some specific kinds. This will help to collect a variety of data that could help people to a bigger picture about their pollution levels in the area.
* Our system will have an AI that will take images from people then analyzes the image to know what kind of pollution the user is facing and the result will be automatically filled into the form which will be sent directly to the local environment department to resolve the problem. This is so easy to use that even an elementary school child can do and it also reduces a massive amount of the cumbersome and bureaucratic process that people have to go through.
* Our system will have an AI that will take the production information and then give the prediction of pollution types that the craft village will facing. By combining the pollution from the image and the prediction then the result will be more accurate.
* Our system is also integrated with location-based technology to detect the location of the pollution and layout the data on the map for the user to monitor.

## Business drivers

Based on the business needs and business solution our team decides to make a Craft Village Pollution Monitor System.

## Project goal

The goal of the project is to build a Craft Village Pollution Monitor System (CVPMS) within the budget of $3000 and deliver on time by the end of May of 2023.

# Architectural drivers

## Functional requirements

**Table 1.** Functional requirements

|  |  |  |
| --- | --- | --- |
| **ID** | **Function** | **Description** |
| FE01 | Login | Use username/password to login into the system, can use the function of the system. |
| FE02 | Register | Use to register a new account |
| FE03 | Forgot Password | Use to recover password |
| FE04 | Change Password | Use to change current password to new password |
| FE05 | Send Mail | Use to sent verify code or new password to registed email |
| FE06 | Edit User’s Information | Use to change registed information |
| FE07 | Change Language | Use to change application language from English to Vietnamses and vice versa |
| FE08 | Take Pollution Photo | Use to take pollution photo |
| FE09 | Detect Location | Use to detection location of user automatically |
| FE10 | Auto Fill Information | Use to autofill necessary information after application detect pollution types from image and get location |
| FE11 | Add New Village | Use to add new village to database |
| FE12 | Detection Pollution Types | Use to auto detect pollution types from image |
| FE13 | Add Additional Information | Use to add additional information for the survey |
| FE14 | Submit Survey | Use to sent survey to database |
| FE15 | View Finished Survey | Use to load the finshied survey for user to review |
| FE16 | View In Progress Survey | Use to load the in progess survey for user to review |
| FE17 | Accept/Decline New Village | Use to give an accept/decline a new village |
| FE18 | Create Local Authority Account | Use to create a new local authority account |
| FE19 | Display Dashboard | Use to load necessary according to user role |

## Business constraints

* Project begins from Mar 1st, 2023 to May 15th, 2023. After delivery, the team will rectify defects in the deliverable (no additional functionalities or features).
* Resource availability is defined below: 01/03 – 15/05 with 5 members.
* Product follows Mentor's requirement.

## Technical constraints

* **Technical to develop**
  + Language: Java (Spring Boot), Dart (Flutter), Python (Flask, FastAI)
  + Develop tool: Visual Studio Code, SpringToolSuite4
  + Version Control System: Git/GitHub
  + Database Management System: Oracle SQL Developer
* **Environment**
  + Operation systems: Microsoft Windows, MacOS, Android, iOS

## Quality Attribute

**Table 2.** Quality Attributes: Availability

|  |  |
| --- | --- |
| **Scenario** | **A1** |
| **Attribute concern** | **Downtime of system** |
| **Description** | The operating time of the system should be 95% to have time for backup data, maintenance and repair. |
| **Source** | Internal to system |
| **Stimulus** | System pause |
| **Artifact** | System |
| **Environment** | The system works normally |
| **Response** | Be temporarily unavailable while backup data, maintenance and repair are being effected |
| **Response Measure** | Uptime of the system should be 95%, downtime is about 1.2 hours per day |

**Table 3.** Quality Attributes: Performance

|  |  |
| --- | --- |
| **Scenario** | **P2** |
| **Attribute concern** | **The latency of initiating transactions** |
| **Description** | Users initiate transactions under normal operations. The system processes the transactions with latency less than 5 seconds. |
| **Source** | Users |
| **Stimulus** | Initiate transactions |
| **Artifact** | System |
| **Environment** | Under normal operations |
| **Response** | Transactions are processed |
| **Response Measure** | With latency less than 5 seconds |

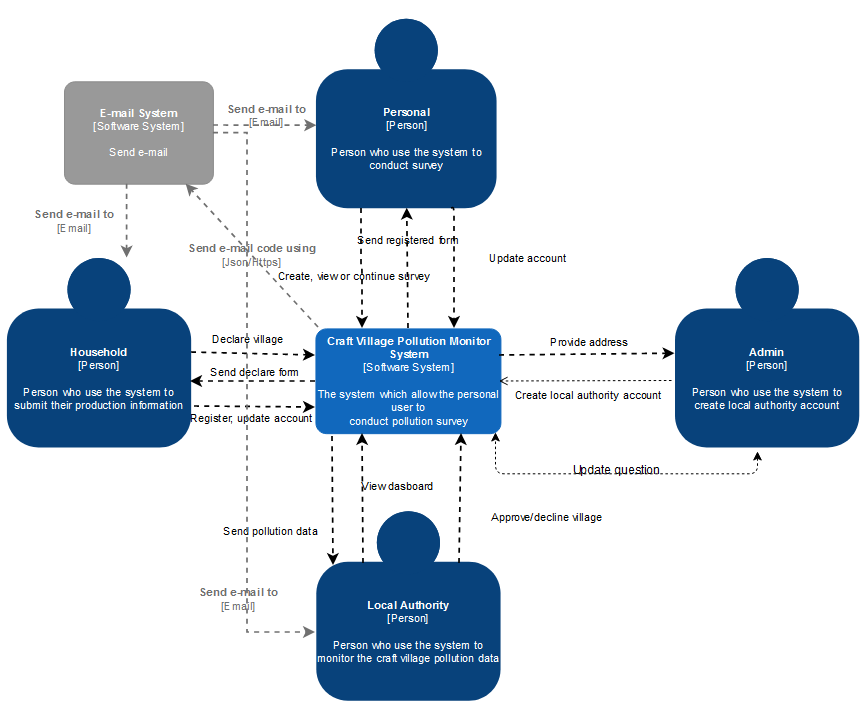
**Table 4.** Quality Attributes: Performance

|  |  |
| --- | --- |
| **Scenario** | **P3** |
| **Attribute concern** | **The throughput of the system** |
| **Description** | At peak load, the system is able to complete 100 normalized transactions per second. |
| **Source** | Internal to system |
| **Stimulus** | Multiple transactions at the same time |
| **Artifact** | System |
| **Environment** | Peak load |
| **Response** | Throughput |
| **Response Measure** | Throughput is 100 transactions per second |

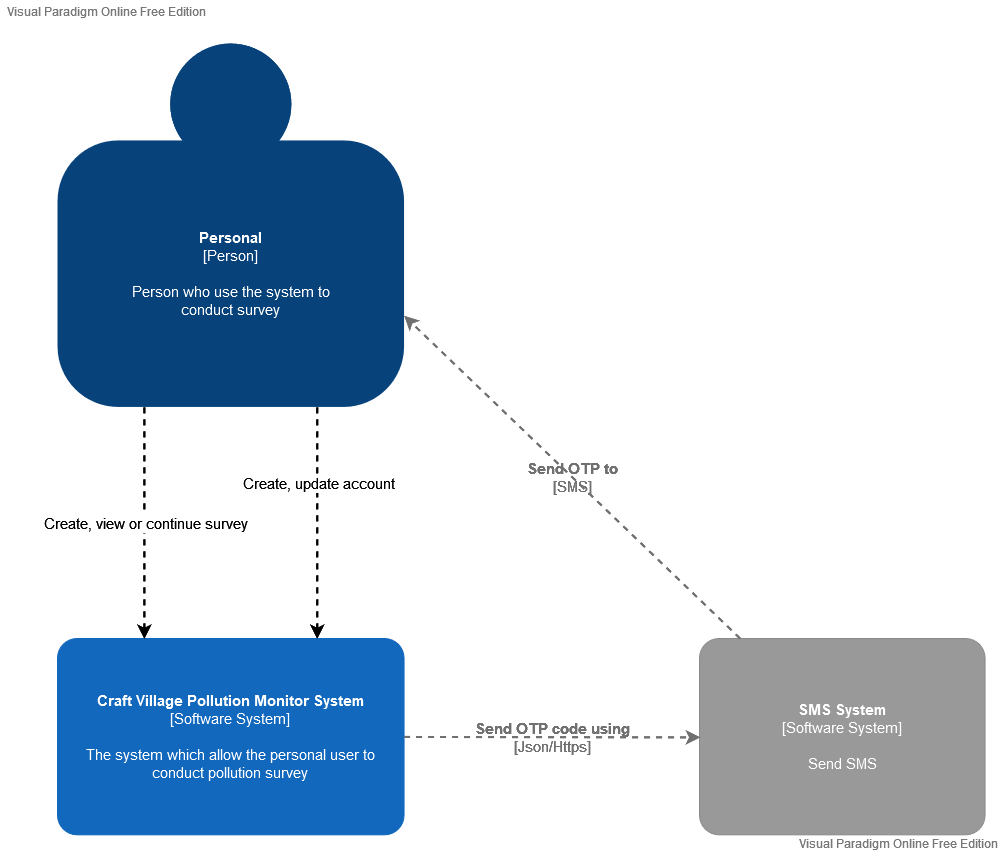
**Table 5.** Quality Attributes: Usability

|  |  |
| --- | --- |
| **Scenario** | **U4** |
| **Attribute concern** | **Using effectively** |
| **Description** | Craft Village Pollution Monitor can be easy for end-users to create a report after 10 minutes using. |
| **Source** | End-users |
| **Stimulus** | Create a report |
| **Artifact** | System |
| **Environment** | The system work normally |
| **Response** | Easy to use |
| **Response Measure** | Easy to use after 10 minutes using |

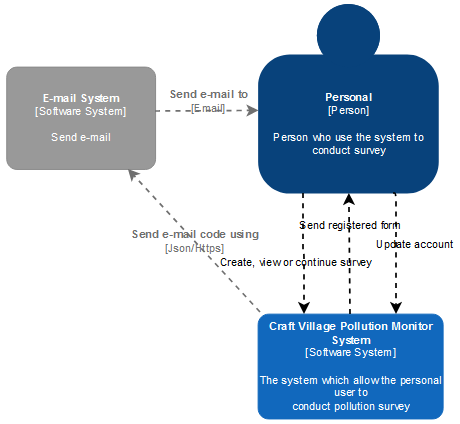
## System Context Diagram



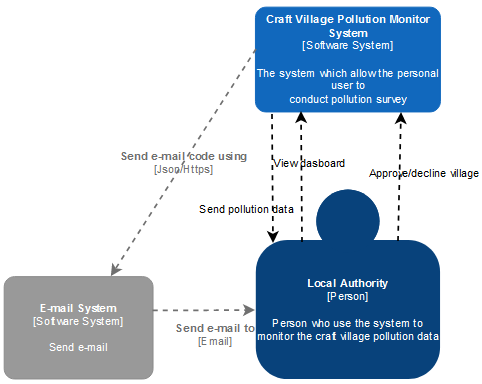
**Figure 2.** System Context Diagram Overview



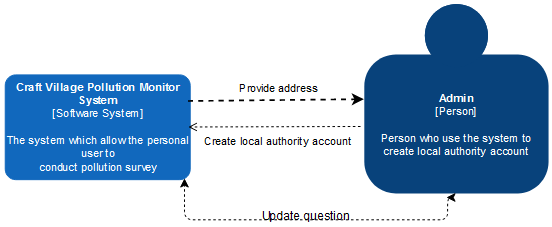
**Figure 3.** Personal System Context Diagram



**Figure 4.** Household System Context Diagram



**Figure 5.** Authority System Context Diagram



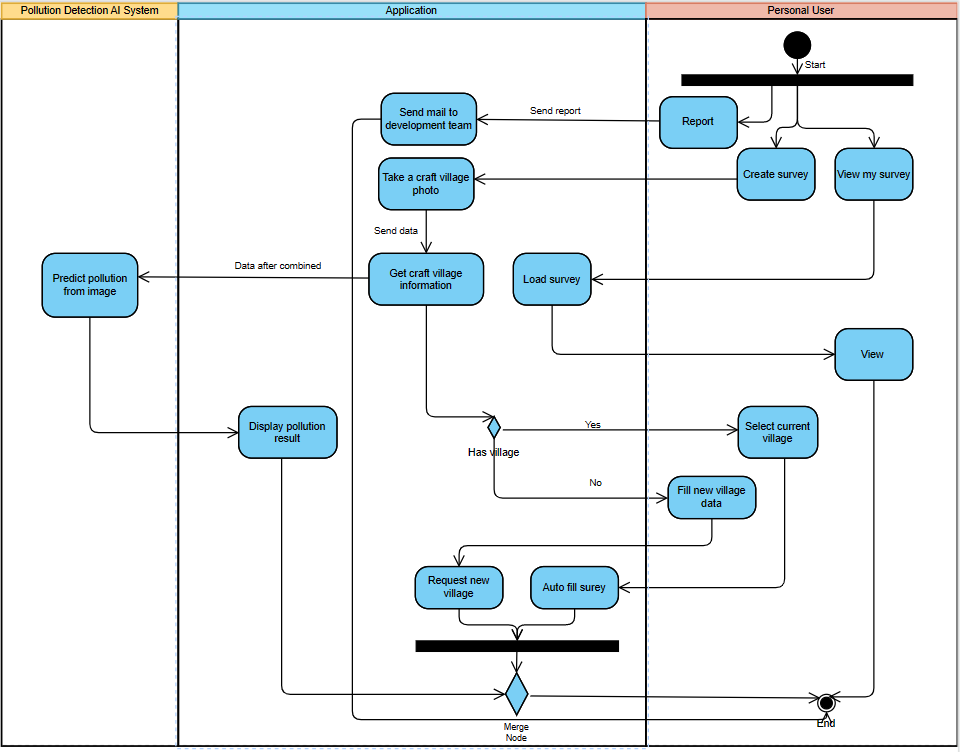
**Figure 6.** Admin System Context Diagram

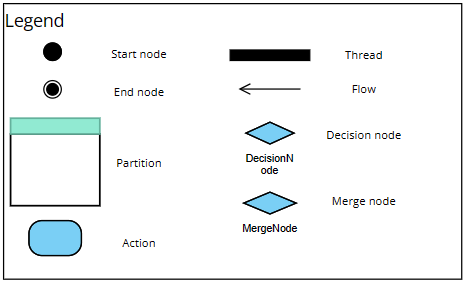
**Prose:**

* The personal user, they can:
* Create anew survey;
* View previous survey;
* Report;
* The household, they can:
* Submit their production information;
* Request add their village;
* Declare their current village;
* Report.
* The authority, they can:
* View craft village’s data (village production information, pollution status, etc);
* Approve/decline a new village.
* Update village**.**
* Report**.**
* The admin, they can:
* Create authority account.
* Update question to training data.
* Download data set.

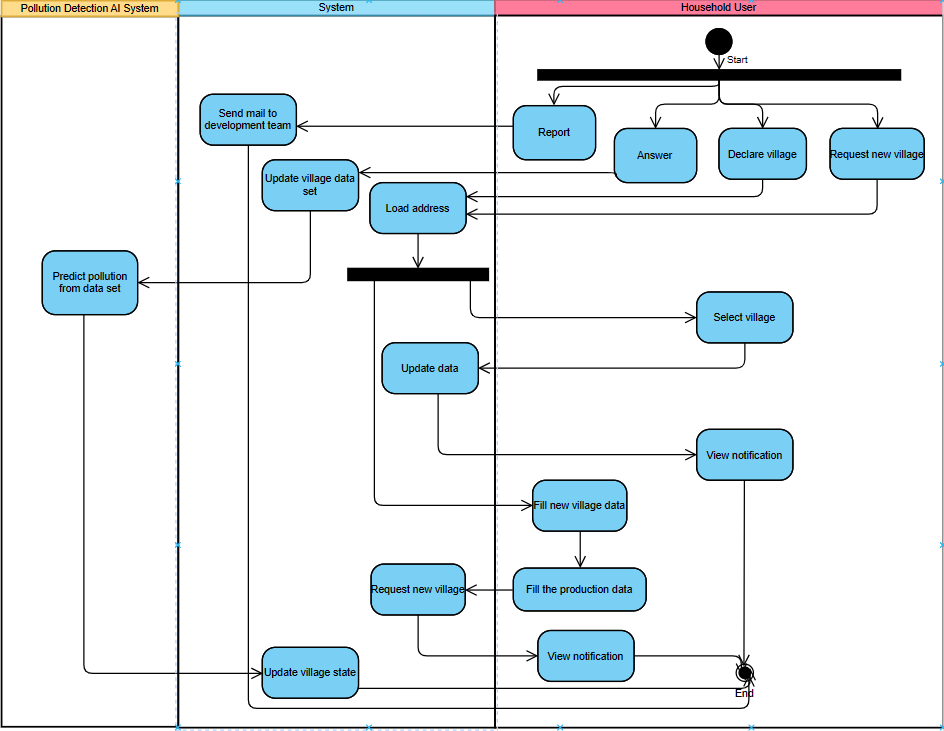
# Activity Diagram

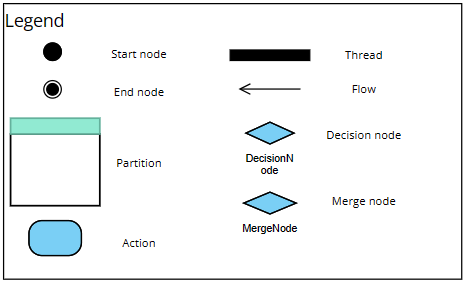
Activity diagram is a graphical representation of workflows of stepwise activities and actions with support for choice, iteration, and concurrency.



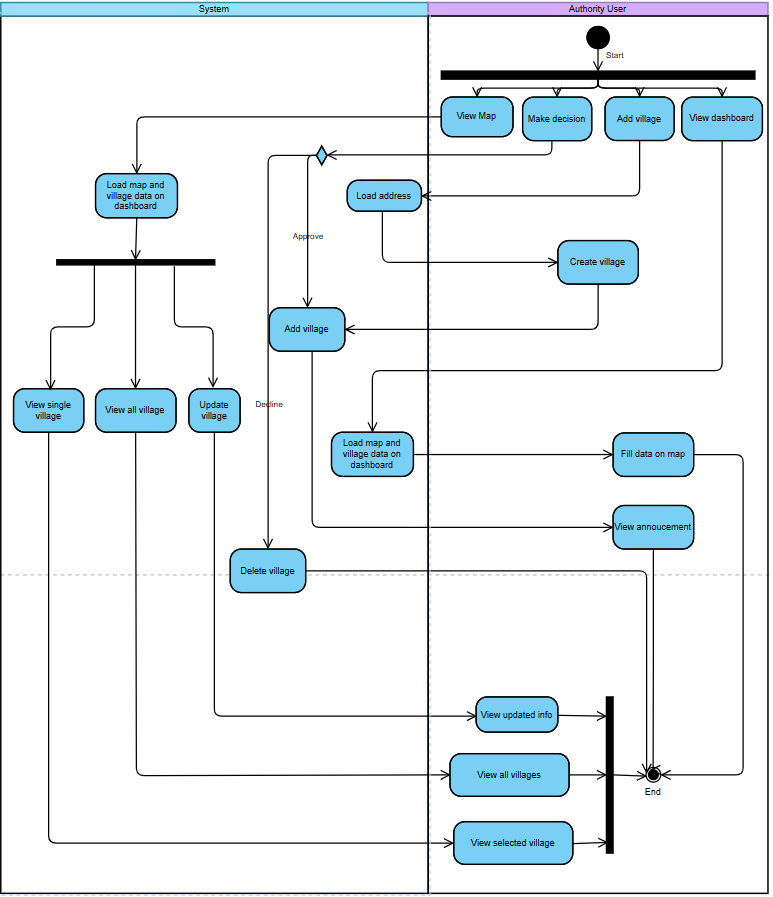


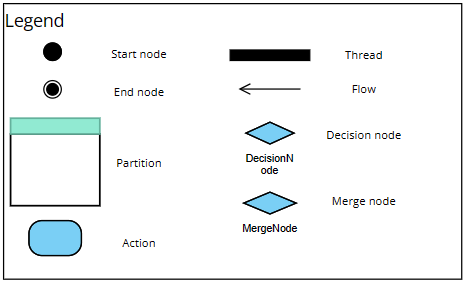
**Figure 7.** Activity Diagram (Personal)

******

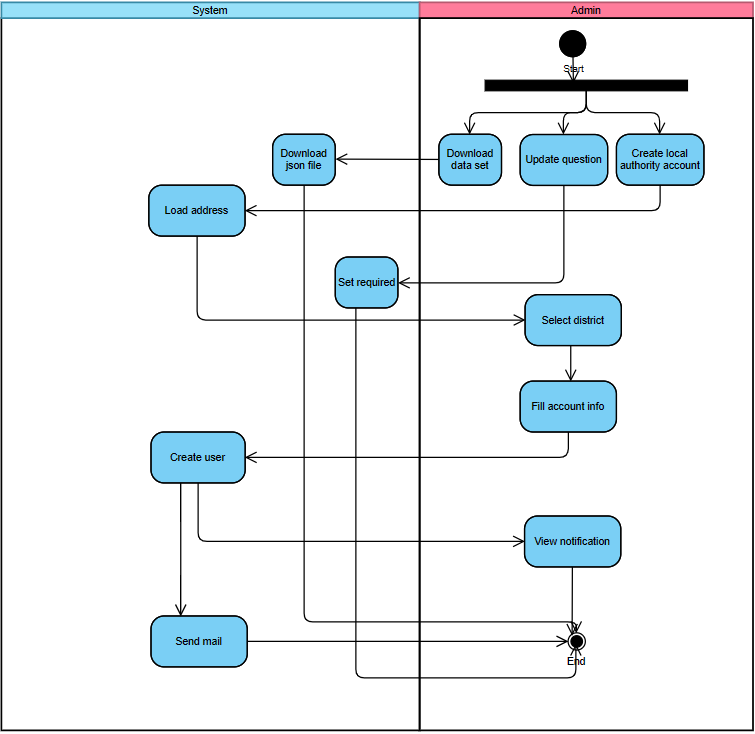
******

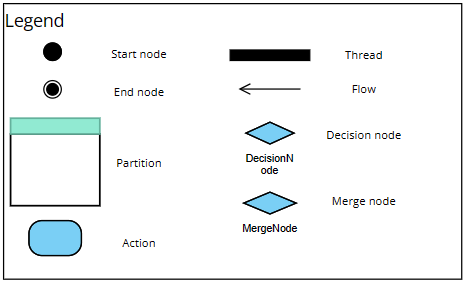
**Figure 8.** Activity Diagram (Household)

******

******

**Figure 9.** Activity Diagram (Authority)

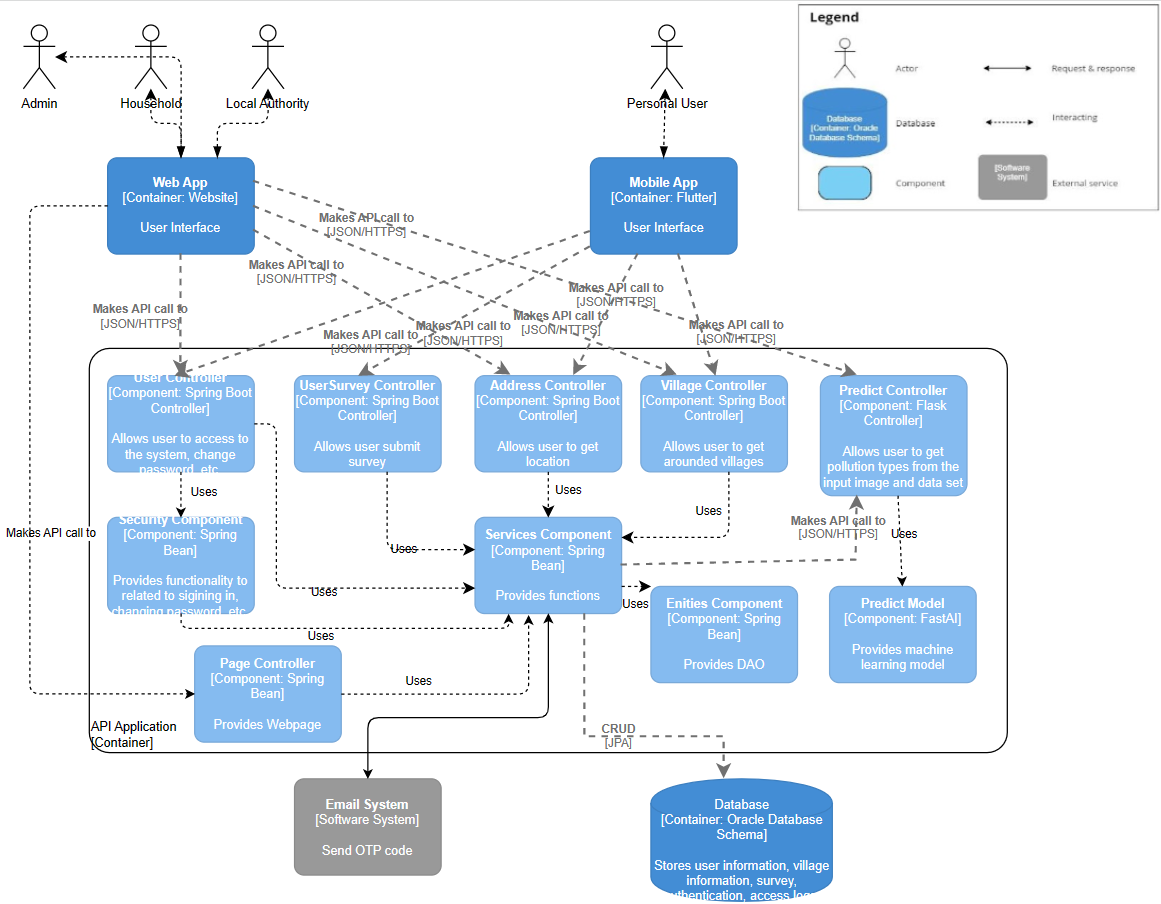
**

**

**Figure 10.** Activity Diagram (Admin)

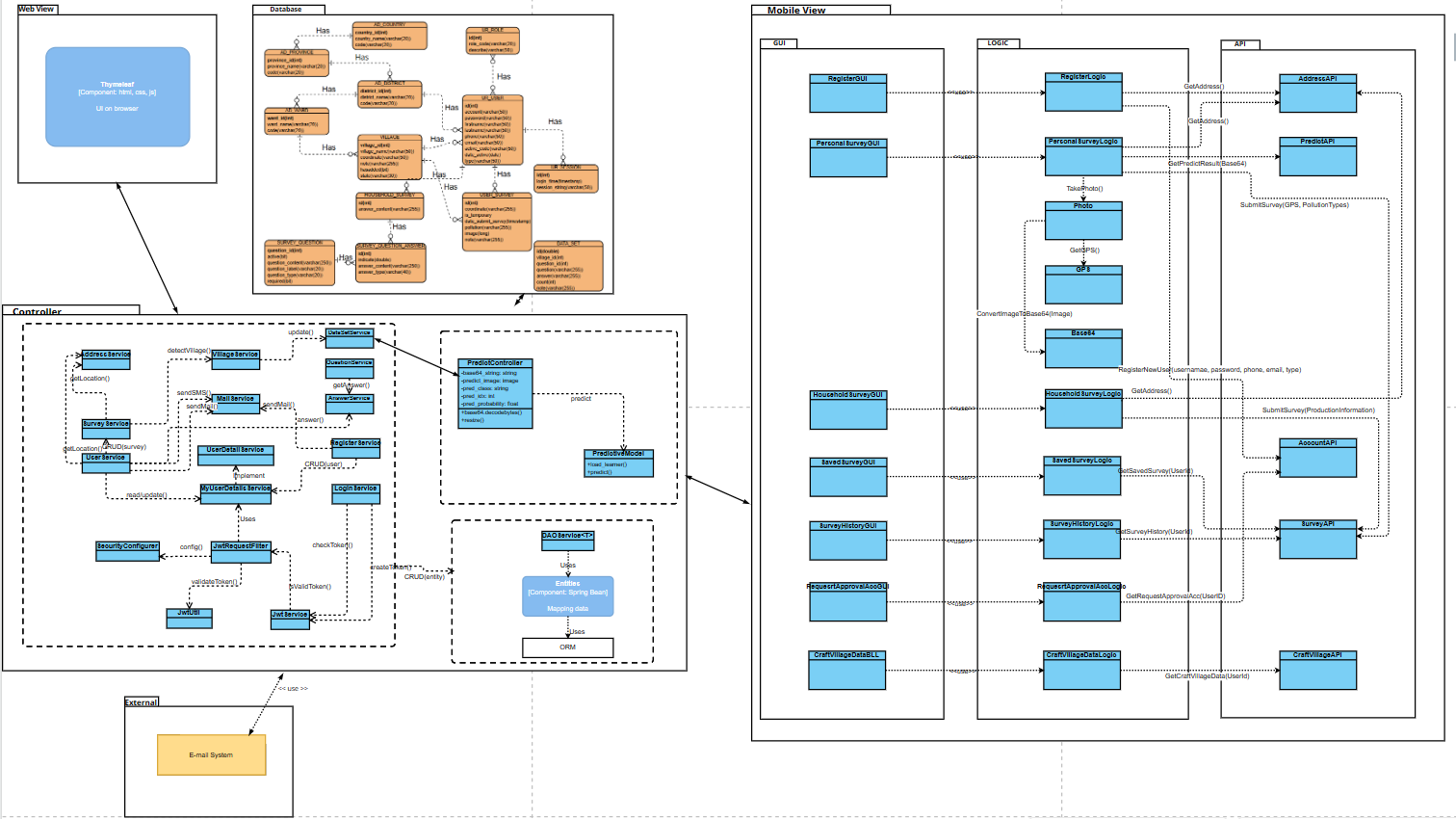
# C&C view

The diagram below shows the overview architecture including components and other related components.



**Figure 11.** Component & connector view

# Module View

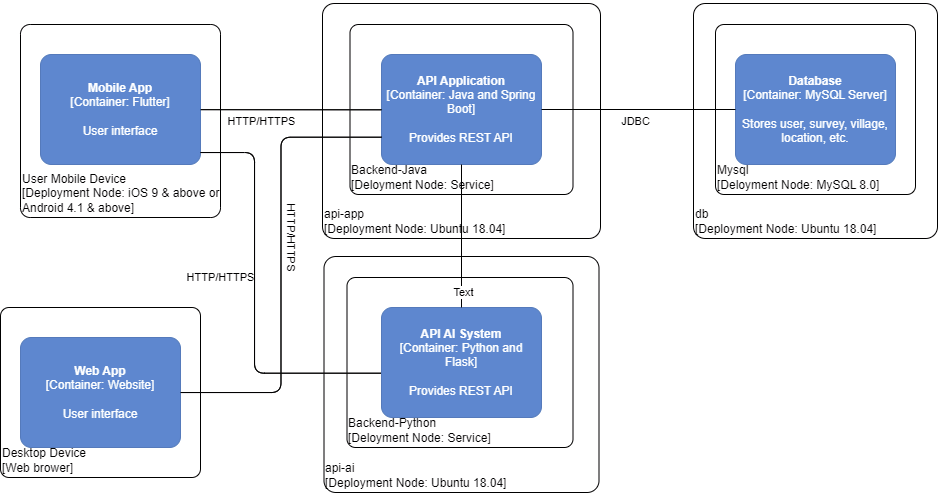


**Figure 12.** Module view

**Prose:**

* The CVPMS includes 5 packages and a database that helps the app run effectively.
* In the Web View package, we use Thymeleaf (Java template engine) to process and generate HTML, Javascript, and CSS.
* The App View package which has 22 classes are often used and we customize it to fit our requirements.
* The Controller package contains 16 classes, the Entities component, and the ORM model. The PredictController and the PredictiveModel are used to predict and return the types of pollution. We build a "bridge" between the software and relational databases using the ORM model and the Entities component.
* The Module package contains 10 models and the relation between them.
* In the External package, we use the E-mail System and the SMS System.
* Finally, the app is connected to the Oracle Database.

# Allocation view

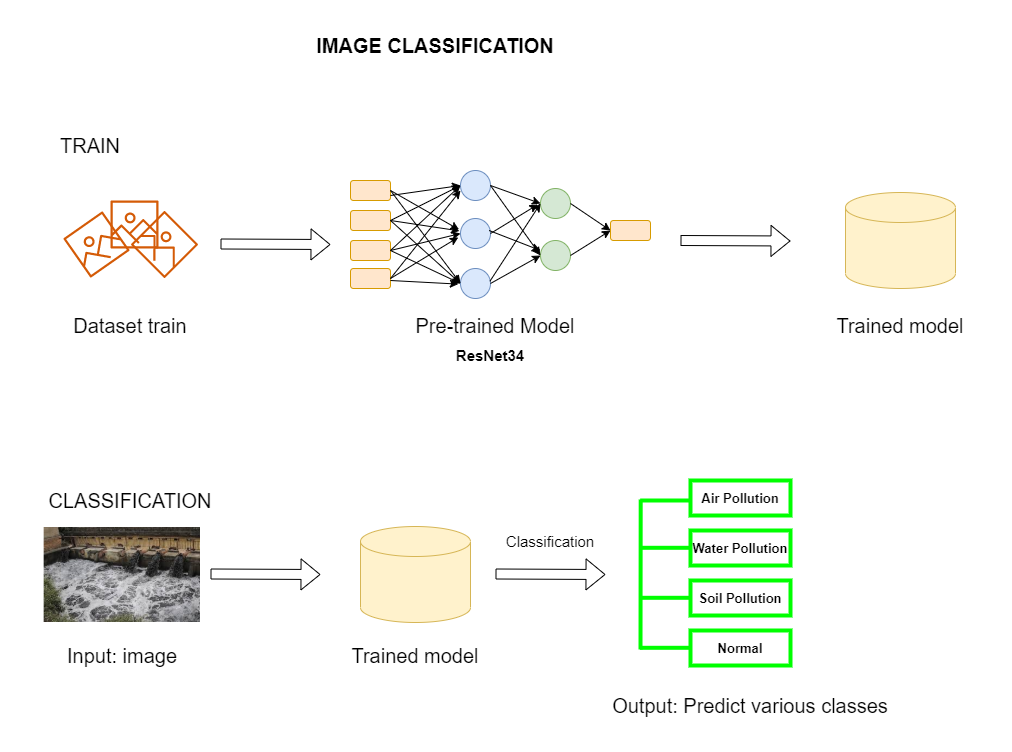


**Figure 13.** Allocation view

**Prose:**

The user can access our system by using Web App (Household User and Authority User and Admin) and Mobile App (Personal User) via internet.

# Proposed architecture for Image Classification

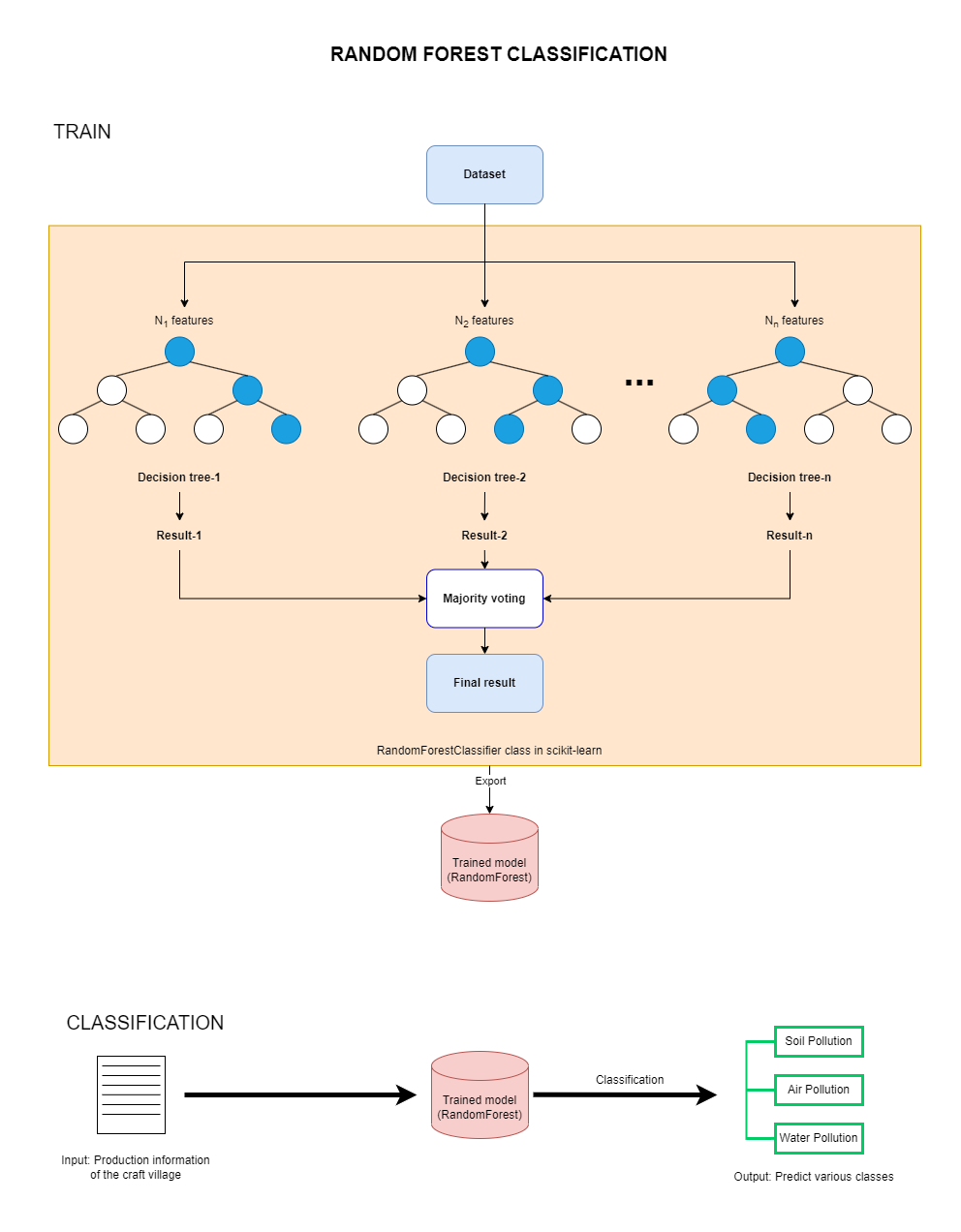


**Figure 14.** Image Classification Architecture

**Prose:**

To train a machine learning model, we use the dataset and the Pre-trained Model (ResNet34). From an input image, the trained model can predict various pollution classes.

# Proposed architecture for Random Forest Classification



**Figure 15.** Random Forest Classification

**Prose:**

To train a Random Forest classifier model, we use the dataset and the RandomForestClassifier class in scikit-learn. The trained model (RandomForest) can make predictions on the Production information of the craft village. The output of the prediction will be the predicted various classes.

# References

|  |  |  |
| --- | --- | --- |
| **No.** | **References** | **Document Information** |
| 1 | Design standards,  Document standards | <https://www.softwarearchitecturebook.com/svn/main/slides/ppt/26_Standards.ppt> |
| <https://standards.ieee.org/standard/1471-2000.html> |
| <https://c4model.com/> |
| <https://machinelearningcoban.com/tabml_book/ch_model/random_forest.html> |
| 2 | Patterns | <https://en.wikipedia.org/wiki/Architectural_pattern> |
| 3 | Evaluation standards | <https://www.iso.org/obp/ui/#iso:std:iso-iec-ieee:42030:ed-1:v1:en> |
| <https://gabrielfs7.github.io/software-architecture/2019/10/18/atam-analyze-evaluate-architecture/> |
| 4 | Draw.io | <https://www.draw.io> |
| 5 | Visual Paradigm Online | https://online.visual-paradigm.com/ |