Bilkent University

Department of Computer Engineering

Senior Design Project

Water Vision

High Level Design Report

Maryam Shahid

Hammad Khan Musakhel

Yiğit Emiray Dinç

Supervisor: Salim Aksoy
Jury Members: Shervin Arashloo and Hamdi Dibeklioglu

December 21, 2021

This report is submitted to the Department of Computer Engineering of Bilkent University in partial fulfillment of the requirements of the Senior Design Project course CS491/2.

Contents

High Level Design Report

Project Short-Name: WaterVision

1 Introduction

WaterVision is a web application that aims to detect groundwater using satellite imaging. It will not only highlight the areas consisting of groundwater but will also show the level of water present. It will act as a groundwater detecting and monitoring tool.

This report describes the proposed software architecture, subsystem services, the factors considered in engineering design and team work details [1].

1.1. Purpose of the system

Water is one of the most crucial natural resources present as it sustains life itself. Water sustainability – especially in the wake of climate change, global warming, water depletion – is at the core of a sustainable environment.

Similarly, groundwater supplies drinking water for over half the population in almost every country [2]. It is used to grow food and sustain irrigation for growing crops. Groundwater is also one of the integral components in many industrial processes. Likewise, for regions prone to floods, groundwater detection can play a key role in flood management. In addition, diseases that spring from water such as dengue – which is carried by mosquito larvae dwelling in still groundwater – can be controlled by effective groundwater detection and moderation [3]. Thus, groundwater detection carries a plethora of applications

WaterVision will bring this to life by detecting groundwater using satellite imaging of various regions. Along with detection of groundwater, it will also report the level of water present. For regions where satellite imaging is difficult to obtain, there will be machine learning algorithms to predict the presence of groundwater. After detection, the application will showcase a variety of layers with various use cases. Such as, a layer for drinking water supply will distinguish it from the rest of groundwater. Areas which are prone to floods and heavy rainfall will be highlighted in the flood dashboard for effective flood management. Similarly, a water irrigation layer will also be present which separates water being used in agriculture. If time allows, we will also add a layer highlighting water regions which are prone to water borne diseases such as dengue [3].

Purpose of the system is to predict possible unwanted natural events and warn the user. This may be imagined as a fortune teller for natural events where the traditional Turkish coffee left its place to satellite images, heat maps, and possible different types of data sources. Ultimate goal is to achieve some valuable precautionary warning beforehand anything unwanted occurs.

1.2. Design goals

1.2.1. Accuracy

Since this program is based on groundwater detection from satellite imaging, accuracy is one of its integral components. The program will ensure that the results of detection have an accuracy rate of over 96%.

1.2.2. Usability

Since the program should be easily used in different resolutions and screen sizes, responsive design of the UI is accepted. In every step of the UI development the first concern is having a properly working UI on distinct types of screens.

1.2.3. Scalability

Although our ML models will be trained on only a few cities' satellite images, it will be scalable so that the datasets of other cities and regions can be fed to the ML models and groundwater detection can be carried out through the already trained models.

1.2.4. Maintainability

The outcome of the process would be assumed as something works as a blackbox, for a given input should produce a close enough prediction. Hence, it may not be seen as something requiring high maintenance. However, there is always a risk of losing the data source which makes the system stop working. Until such case happens it is believed that the blackbox not requires any work after it is made.

1.3. Definitions, acronyms, and abbreviations

Map: Any type of data that has corresponding place information in it, not just the world map.

Intelligence: Data after its first process. Knowledge.

1.4. Overview

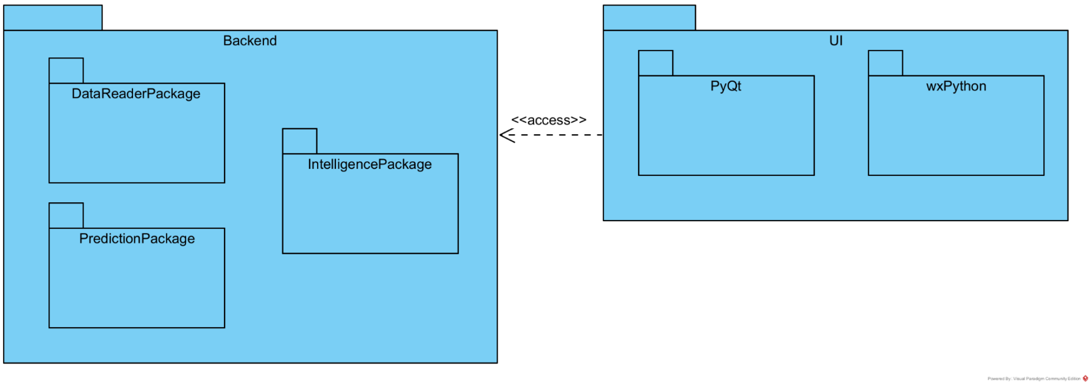
General structure of a high-level program is as follows: there exist two subsystems, namely Backend, UI. Backend reads, extracts and means the information obtained from maps, where the UI is where they are shown. To reach what is aimed at, data from maps may be read using image processing, thus creating a knowledge base, and by using machine learning algorithms on them we would like to reach a prediction of a risk happening.

2 Proposed Software Architecture

2.1. Overview

Ultimate purpose of this project is to predict harmful natural events. Thus, achieving accurate predictions was the main concern. Hence, the overall architecture is based on easily adaptable prediction producers, so that any unforeseen aspects of possible natural disasters would be eliminated through new adepts.

2.2. Subsystem Decomposition



2.3. Hardware/Software Mapping

Hardware/software mapping of the system is simple, since there is only one machine that operates the expected functionality i.e. predicting possible events. That is direct out service is required.

2.4. Persistent Data Management

Images, maps, graphs used for the predictions are kept in a corresponding file pathway i.e. either data is put in its correct place before the program execution or data will be downloaded to the correct place. However, the aforementioned solution may not be applicable in our case, since all those data are not available realtime.

2.5. Access Control and Security

WaterVision will only be accessible by those who have a username and password. The account information will only be given out by the admin, thus enhancing security of the application. This information will be encrypted and stored in the main server to protect from theft and misuse.

The machine learning models implemented for groundwater detection purposes will be kept secure and not shared with the client. These models will be kept in a local environment and thus will not pose security concerns.

2.6. Global Software Control

Since the program is assumed to be installed after purchase there is not going to be a live update system, instead the sold product is planned to have only one version. If the program is required to have new updates, then the solution for this problem is again selling the upgraded product as a new solution. Hence no global software control is planned.

2.7. Boundary Conditions

- **Initialization**

  For a new user, the application will be installed in the user's computer. Account information will be created and supplied by the admin. The server will be initialised for the user consisting of sockets and connection ports. The database will be set up and client communication will be initiated.

- **Inaccurate Results**

  Since most of the data being processed in the program consists of trained/tested datasets from machine learning models, accuracy and confusion matrices will be calculated periodically for all datasets. In case the accuracy falls below a set threshold, the model will be retrained. This will provide a more robust response to inaccuracy.

- **Failure**

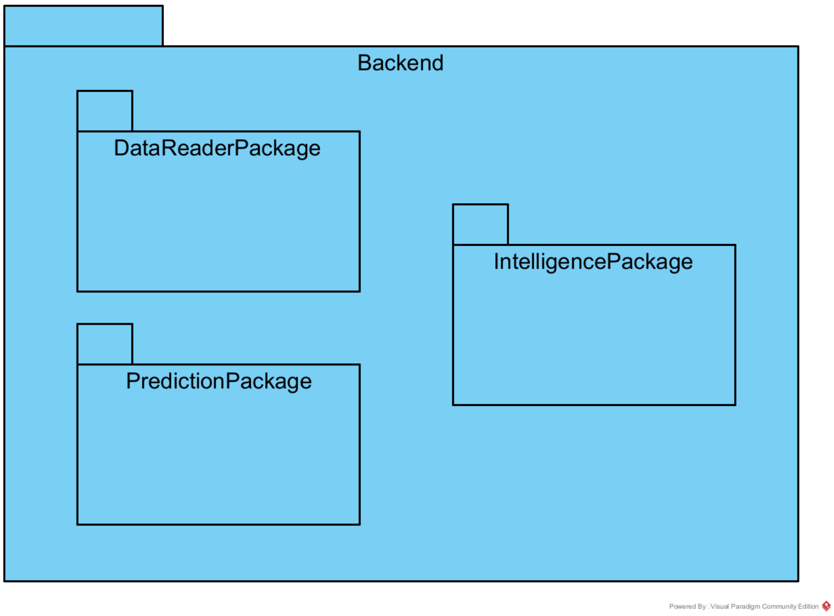
  In case user authentication failures will occur if the user inputs an incorrect username or password. Network failures can also occur in case the user loses connection to the network or database storing the results of the ML models.

  Moreover, since the server needs to constantly be active, it will be robust against failures. it will handle boundary cases and exceptions using various mechanisms. It will also inform the user about the respective errors.

3 Subsystem services

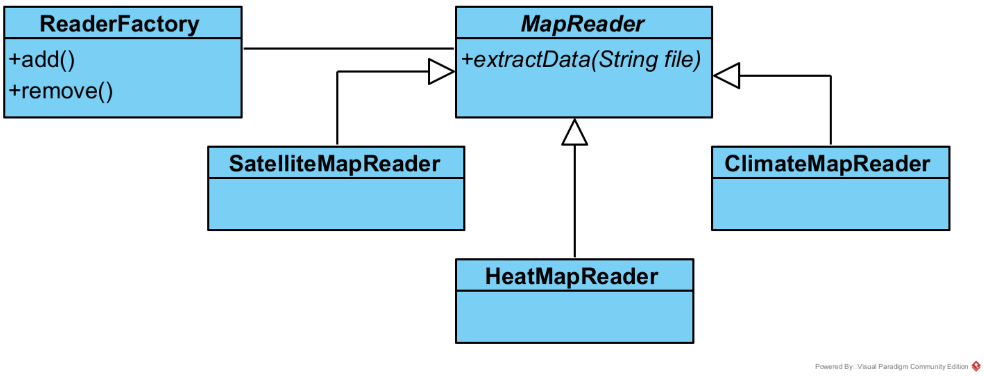
Subsystems are composed of two distinct ends of the program. They are called Backend and UI in a genuinely old fashioned way. Backend is assumed to be the part where all of the data is read and processed, also where the predictions are made. UI as one may assume, is where the aesthetics of the program is handled. To achieve a neat and responsive design we would consider outsource libraries such as PyQt, wxPython.

3.1. Backend Subsystem Service



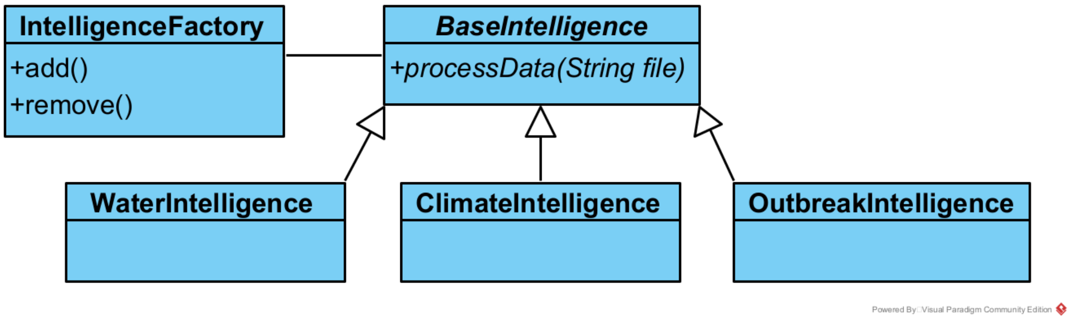
Backend structure with corresponding DataReader, Prediction and Intelligence packages.

3.1.1. DataReader Package



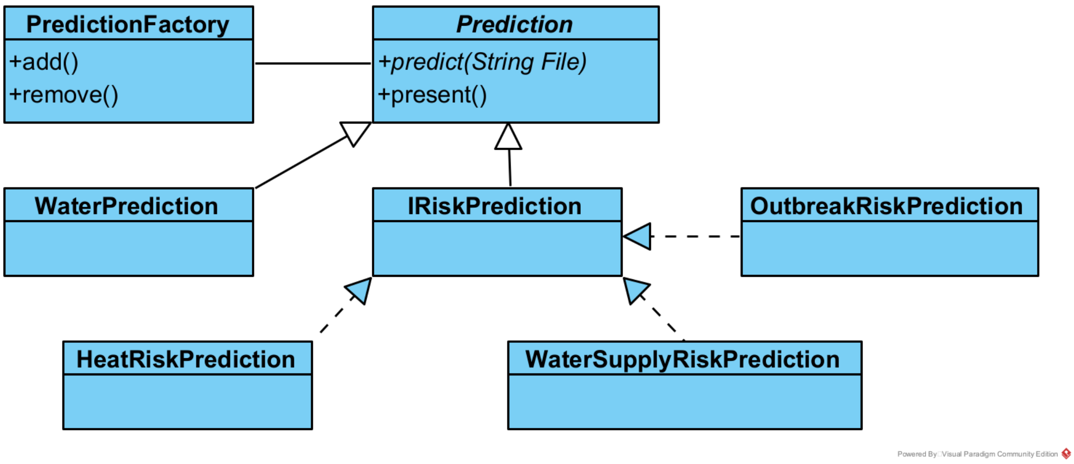
It may be predicted as quite likely to have different types of maps to read, thus data to accumulate. That is the main reason why we may need to have a factory design pattern. Other than that, illustrated Readers are for acquiring data from Satellite maps, Heat maps, Climate maps, such that for a given map, the reader must convert it into mathematical data to be used in the intelligence package.

3.1.2. Intelligence Package



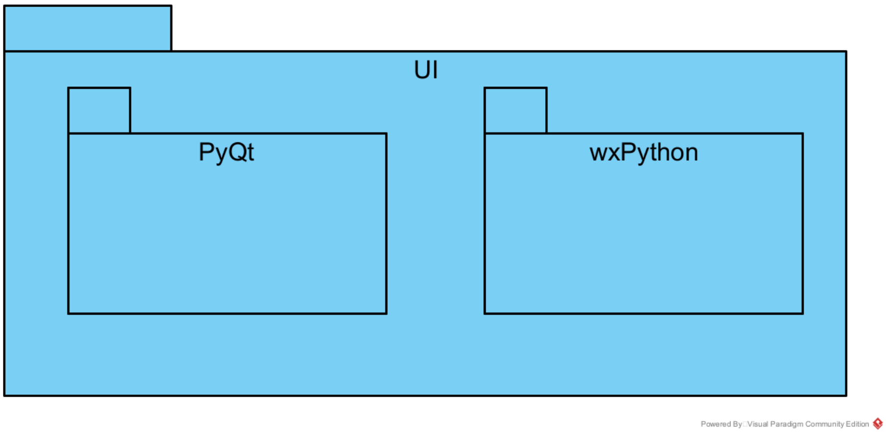
Aforementioned reason to implement a factory pattern is valid for the Intelligence. Moreover, it is assumed that generally for each type of distinct map there should be corresponding Intelligence to process its data. The Intelligence part is where the data from the map is filtered and forwarded to prediction.

3.1.3. Prediction Package



Aforementioned reason to implement a factory pattern is valid for the Prediction. Moreover, it is assumed that generally for distinct types of Intelligences there should be corresponding Prediction to oversee possibilities from past data. The prediction part mainly arose from know-how relationships.

3.2. UI Subsystem Service



Since some parts of the UI are designed using functions from PyQt and wxPython it is quite not simple to divide libraries in a structured way. Instead, for an object drawn by PyQt, it was not obligated to color it using the same library such that sometimes QML is handled that process and sometimes wxPython. Hence, libraries are shown together as given.

3.2.1. PyQt Library Dependencies

PyQt is a GUI design library for Python, however by adding such flexibility it lets programmers benefit from QML.

3.2.2. wxPython Library Dependencies

wxPython library is a general purpose GUI development library for Python. We would like to benefit from it as we face obstacles in PyQt.

4 Consideration of Various Factors in Engineering Design

There are numerous aspects of engineering design, many different nuances to be aware of when developing a software project. It is essential to consider the far-reaching limiting factors to our project's design.

- **Human Factor:** The project is aimed to be developed as an application with an interactive interface. Although our project is bound to have complex technical issues during and after development, it is pertinent to notice that it will be mainly used by humans. The user of our application could and could not be a computer expert, thus, we have decided to develop a highly dynamic and interactive graphic user interface for the application. This way, it will be easier for the user to navigate through numerous functionalities easily. It is highly important to optimize this factor during design in order to maximize the reach of this project. The goal of technology is to maintain simplicity at the front with functioning complexity at the back.

- **Aesthetics Factor:** In this ever-growing technology-oriented industry, competition is sky high. There are multiple similar applications with similar performance metrics, however, their aesthetics differ. In this era of technological bloom, it has become highly important to develop an aesthetic design for the software one is aiming to develop. The aesthetics of the software need to be pleasing as we have witnessed a rise in dark and light mode options for the GUI in various web and mobile applications. Thus, we have decided to use colors that allow the user to interact with the application for longer periods of time. Not only this, we have also decided to focus immensely on the dynamic features of the GUI of our application. The primary focus of our project will be to showcase various filters on a map in a distinctive manner in a dynamic manner.

- **Data Accuracy Factor:** Accuracy is a major part of our project. What must distinguish our project from other similar projects should be the functioning part of it. WaterVision will be considered fully successful when it records high accuracy results for different filters. High accuracy is our goal because our project can ultimately be used by different municipalities across the globe to learn about the ground-water level in different regions. We have resorted to spending most of our time learning different techniques/algorithms to maximize the accuracy. The dataset we are dealing with is enormous in size and it is critical to produce high performance metrics as giving low accuracy statistics will hinder the optimal goal of our project: practicality. The scope of our project is to be used for flood management as well, which in itself is a sensitive issue; thus, high accuracy is important.

- **Economical Factor:** There are numerous datasets available on the internet which we need to develop a more dynamic project in terms of regions to inspect. However, most of those datasets have exorbitant prices. The dataset we will use for our project is available for free [5]. Hence, we have not incorporated the option of choosing numerous regions in our design process, rather we have decided to use a single region's dataset for our project. However, during design we will make sure that our program is scalable; we will develop models that will also allow different datasets to be trained on it. This way, we can incorporate different regions in our project if we have access to their satellite imagery data.

5 Teamwork Details

5.1. Contributing and functioning effectively on the team

Water Vision comprises different functionalities such as the application of different layers, producing high accuracy results, designing different machine learning models, etc. Therefore, we were able to divide the tasks amongst ourselves thoroughly. The division of work is described below. Nonetheless, each member participated in meetings regularly and contributed to report writing, software design process, and brainstorming equally.

- **Maryam Shahid:** Maryam is responsible for the GUI development of the program as the leader of this part. She is responsible for deploying and testing machine learning algorithms as KNN, CNN. She carried out the preprocessing of the test dataset.
- **Hammad Khan Musakhel:** Hammad is responsible for helping the GUI development of the program including the main screen. He is responsible for deploying and testing machine learning algorithms as KNN, CNN. He carried out the preprocessing of the train dataset.
- **Yiğit Dinç:** Yiğit is responsible for the relation between subsystems, main program logic. He is responsible for deploying and testing machine learning algorithms as SVP, PCA. He carried out the preprocessing of the labels of the dataset.

5.2. Helping creating a collaborative and inclusive environment

Since we are a group of three members, it was relatively easier to divide work amongst ourselves. However, the work was divided but it is as if all of us are equalling helping one another in their tasks. Thankfully, the collaborative environment has allowed us to assist one another in a more professional manner. It has also allowed us to share problems with each other reluctantly. Moreover, due to being a group of three only, each member was bound to carry more load, but the inclusive environment we have established has allowed us to share the load with one another. This has helped us to function as one single body.

5.3. Taking lead role and sharing leadership on the team

Since we were a group of three people, all of us collaborated equally on leadership roles. Maryam initiated the project since she came up with the idea of this project. She led the project when it came to understanding the fundamentals and specifications, and divided the reports amongst the group members. Hammad supervised preprocessing of the datasets which was fundamental to the project. He guided the team on preprocessing satellite images. The team then undertook leadership roles for various machine learning models - Yigit conducted CNN along with KNN and acted as a leader for that model. Meanwhile Hammad led feature selection and extraction. Maryam conducted the UI design of the program and assigned tasks to the team. The entire team worked collaboratively on the reports.

6 References

[1] Object-Oriented Software Engineering, Using UML, Patterns, and Java, 2nd Edition, by Bernd Bruegge and Allen H. Dutoit, Prentice-Hall, 2004, ISBN: 0-13-047110-0.

[2] Water for Life Decade. https://www.un.org/waterforlifedecade/water_and_sustainable_development.shtm. [Accessed: Oct 09, 2021].

[3] Groundwater Foundation. https://www.groundwater.org/get-informed/basics/groundwater.html. [Accessed: Oct 09, 2021].

[4] Automatic detection of surface-water bodies from Sentinel-1 images for effective mosquito larvae control. https://www.spiedigitallibrary.org/journals/journal-of-applied-remote-sensing/volu me-15/issue-01/014507/Automatic-detection-of-surface-water-bodies-from-Sentin el-1-images/10.1117/1.JRS.15.014507.full?SSO=1 [Accessed: Oct 09, 2021].

[5] "Spacenet 3 satellite imagery," [Online]. Available: https://spacenet.ai. [Accessed: 21-Dec-2021].