Weather Report System for Mali

Group 17:

Yuyu Bai [2732696]

Zhining Bai [2728339]

A.W. Gerritzen [2522283]

Simei Li [2738043]

Yiran Li [2730767]

May 22, 2022

0. Project History Table

Version:	Date	Modification	Pages
0.1	April 18, 2022	Doc Creation	1-5
0.2	April 29, 2022	Section 6-10, code update(Voxel framework) can suppo	5-11
0.3	May 16, 2022	Change input, multi-input (add voice input)	prototype
0.4	May 17, 2022	Complete the French menu	prototype
0.5	May 20, 2022	Correct report based on feedback	1-10
0.6	May 22, 2022	Finish new sections	8-12

1. Name

To be able to clearly deliver the idea of our project, we find a suitable logo to represent our application as Figure 1. The app's name is How's the Weather Tomorrow.

Figure 1 Weather Report System for Mali Logo [1].

2. Summary of key idea

Mali is considered one of the world's least developed countries and its economy is based on agriculture. Climate change has a big impact on the country. The insecurity of output has increased as a result of extreme weather. On the other hand, it is difficult for the local people to know the weather precisely due to a lack of technology and network.

This prototype can provide farmers in Mali with weather forecast and severe weather alerts. This service provides an intuitive voice-based interface that converts quantitative weather data into understandable information and supports many languages. When severe weather is observed, the information will be broadcast on the radio station to help achieve the goal of informing local people as swiftly as possible.

3. Actors and goals

Table 1 Stakeholders and their operational goals

Stakeholder	Operation goal	Responsibility in the envisaged system
User (Farmer)	Receive weather forecasts and severe weather alert	To avoid the harmful impact of atypical weather on crops and other daily activities, preventive steps should be done in advance based on weather forecasts.
Weather Report System	Provide weather information to users in different languages and determine whether there is extreme weather	Qualitative weather forecast data should be translated into many languages and made available to users. Investigate historical data to find unusual weather indicators, and provide weather to radio stations beyond the indicators.
Weather Data Provider	Provides weather forecast information	Provide accurate localized weather forecasts for specific time, including temperature, rainfall, wind speed
Radio Station	Broadcast extreme weather alert to residents	Receive severe weather alerts and disseminate them at the appropriate moment
Hosting Agency (NGO)	Maintain the application's presence online and functionality	Infrastructure funding and management

4. Context and scope

Extreme weather and climate events (such as floods, landslides, and droughts) have a disastrous effect on the Malian people's agricultural development and livelihoods, particularly for farmers and other outdoor laborers [2]. Farmers and traders in need of weather information can dial +31 20 8082848, then enter the code 9990521878 to access the Weather Report System menu. The parties' interaction is depicted in Figure 5 in Appendix.

Since Mali is a multilingual country, when calling the Weather Report Service System, the users would first select a language, and then from a list of options choose the city for which they want to know the climate. Next users select the date since the weather data provider can provide weather forecasts for the next seven days. The Weather Report System then sends a request to the weather data center for relevant climate information. The system analyses and collates the data to determine whether there will be extreme weather soon and informs the user results in the form of a voice announcement via radio station. During this progress, the service provider should simplify complex data and offer it to the user in an easily comprehensible format. The weather data API, as the data provider, needs to ensure that the data is accurate, immediate and comprehensive.

The service will be based on the OpenWeather API and therefore the weather data could only contain the areas for which it is provided. In addition to regular weather forecasting, our service also includes forecasting extreme weather and provide timely alerts. The success of the weather report system

therefore depends on the ability to accurately predict future weather and extreme climate, and to be of use to local farmers and other users in Mali.

Important prerequisites for the success of this program are that the weather alert system is hosted on a stable server with suitable resources, and the weather data provider guarantees the accuracy and integrity of the data. Furthermore, the system should be connected to a mobile phone network so that the users can call into the system.

5. Use case scenario

<The end-user> <calls> <weather report system>

<weather report system> <provide a list of language to> <The end-user>

<The end-user> <select language and return it to> <weather report system>

<weather report system> <provide a list of report date to> <The end-user>

<The end-user> <select date and return it to> <weather report system >

<weather report system> <provide a list of regions to> < The end-user>

<The end-user> <select region and return it to> < weather report system >

<weather report system> <give weather and alert information based on selected info to> <user>

6. Interaction and communication

To be able to clearly articulate the interactions of our users with the application, both internally and to potential stakeholders, we also created a UML diagram as Figure 7 in the Appendix. The diagram below illustrates that: Users (Farmers) should be able to call the Weather Service to choose their service language and submit their service requirements. Weather Service provider should be able to request and analyze weather data from Weather Data provider. Weather Data provider should be able to send the weather data result back to the Weather Service provider.

7. Information concepts

Figure 8 in the Appendix is a UML data model showing the relational database of Weather Report System. Where PK is a Primary key constraint, and FK stands for Foreign key constraint [3]. While the highlighted sections are the ones that have already been realized, the remaining parts are related to the adoption of other languages. Where WeatherAPI is the data retrieved directly from the Open Weather Map, we extract the relevant data and save it in WeatherReportSystem. The cities with weather forecast data are shown in Location, and Radio Station is a list of local radio stations with contact information, and an alert is sent from the system to the appropriate station. Currently, there is no database linked with languages because the present prototype only uses English and French. If more national languages are required for application, more audio data should be considered.

8. Technology infrastructure

Figure 9 in the Appendix is a technology infrastructure diagram in our current stage of development. Users will take the VoiceXML browser as their entry point. The weather report system chooses Flask

as the web application framework, PostgreSQL as the database, also there is a separate file database to store .wav files. Heroku, a platform as a service (PaaS) platform, enables to build, run and operate applications entirely in the cloud after deploying the applications.

9. Cost considerations

In general, this project does not need a large budget. On one hand, this is a student-designed product that is still in its early stages, on the other hand, it has not been placed into the official application, and factors such as maintenance and development do not require cost consideration. If it is formally put into action, however, it will need professional maintenance as well as additional fees. In terms of basic investment consumption, our present design makes use of the free Heroku server, but in practice, it may rely on more mature hardware, such as the Kasadaka system [4]. Furthermore, some revenues can be acquired through sponsors, therefore some of the operational cost can be covered, and the remainder possibly undertake by non-governmental organizations (NGOs). Users need to spend on a simple mobile phone and a sim card and communication expenses.

Table 2 Cost considerations for Weather Report System

Type of Cost	Specific Term	Description
Operational	Electricity and Internet	There would be energy consumption in operation, mostly electrical demand, but the overall consumption should be small enough to be met by sponsors or NGO. Similar situation for internet requirement.
	Maintenance	Zero cost because the system is still at development stage. Although follow-up maintenance is essential, the Weather Report System's designers would also be liable.
	Weather forecast data acquisition	Available for free on the Open Weather MAP API.
	Radio	Radio broadcasting costs around $3 per minute. Radio is mostly used for severe weather alerts, which have a significant influence on people's lives. Because the warnings are not often, the expense is not high, thus the government can handle this aspect.
Investment	Hardware (Kasadaka system) [4]	It is an appropriate hardware solution in this scenario since the platform supports voice-based information services. Each device costs about 100 euros but this is a one-time fee, it may be covered by NGO.
Development	Prototype development	Because the prototype is being produced for educational reasons, there are no compensation expenses associated with its development.

Others	Cell phone	The users requires mobile phones with basic calling functionality
	Calling cost	Calls need to pay the call tariff of $0.14 per minute to the communication operator. . Individuals' need for precise weather is seasonal, therefore it's impossible to have a more exact estimate of the number of people who enquire about the weather prediction, which is expected to be 100 people every day. [5]
	Advertisement Income	Can insert some ads at the end of voice calls to get some income from the sponsors

10.Feasibility and sustainability

To realize the technical feasibility of the scenario, we adopt the technology infrastructure as Figure 9. Firstly, we need to consider the development efficiency during the development progress. Instead of choosing to manage the local server, the Weather Report System chooses Heroku as the Paas platform. Heroku is distributed and remote, which means it's faster, more secure, and easier to scale than traditional private servers. Through Heroku, the Weather Report System developers save time and money to manage the expensive local servers. Secondly, as the project expands and grows, we hope to gradually complete the separation of the front and back ends to reduce maintenance costs. Separation of front and back ends requires separation of business, development, and deployment. The biggest challenge is code development efficiency. Without a concrete front-end and back-end interface standard, we could only develop the back-end after the front-end development is finished.

To realize business and economic feasibility, the first solution would be to attract advertising investors. On the one hand, investors can raise the awareness of their products by placing advertisements. On the other hand, the Weather Report System can use sponsorship funds for system maintenance and operation. Secondly, we can also recruit volunteer engineers and workers from all over the world. Considering the current corona situation, volunteers can work on development and maintenance through remote internships, flexible working, etc.

The goal conflicts could appear between advertising investors and end-users. As mentioned earlier, we hope to bring in advertising investors to fund the Weather Report System. If the banners are too long or cumbersome, the users will gradually lose patience while waiting. Conversely, if the banners are too short, they will not meet the needs of the investors. It is therefore very important to design a tagline that is suitable for telephone services.

The prerequisites for Weather Report System to work effectively are the effectiveness and accuracy of the weather forecast data and the availability of telephone signals. In terms of effectiveness, the Weather Report System service can provide weather forecast data for up to one hour recently. The service can also provide weather forecasts for up to a week to meet different needs that vary among different users. In terms of accuracy, our external weather data is from OpenWeatherMap API. As a third-party service, we cannot guarantee if the data itself we receive is accurate. However, we can minimize our forecast range by asking for the precise user location.

Figure 2 shows the current E3 value model of the project. We will refine this model as we expand the project and improve the functionality.

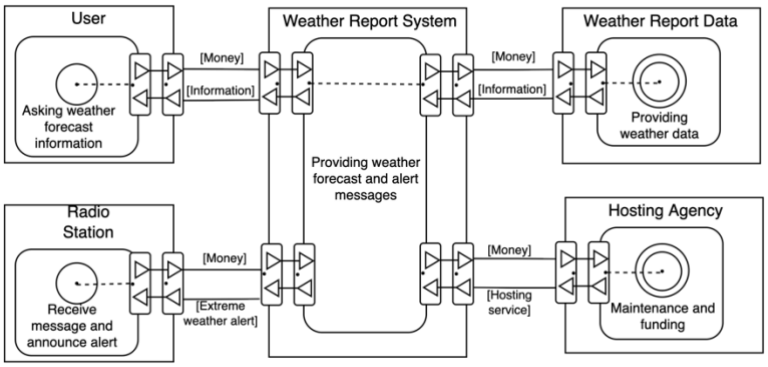

Figure 2 The e3value model of the Weather Report System.

It is essential to consider the ethical aspect. The Weather Report System service is currently web and telephone signal based. It is necessary to have internet when providing end-user with the weather forecast service. The poor farmer in Mali may not have their own phone or internet, which will raise ethical challenges. In another word, not all the farmers have the same right to use the service. Moreover, it could be a high risk that the internet is interrupted in Mali. One solution could be back up the critical information in the local database which could serve as an alternative database. Another solution could be to set up volunteer services in the cities, which collaborate with the local NGOs. In this way, the poor citizens could use the service in the volunteer station.

11.Key requirements

The prototype would be built based on the priority of the following requirements, with the "Must have" being fully shown and the "Won't have" being disregarded in the design.

Table 3 Key requirements for Weather Report System

Must have	• Support service in English and French • Support voice interaction with the users • A database to store daily updated weather data • A database to store pre-recorded multilingual audio files • Retrieve weather forecast from OpenWeatherMap API • Provide rain, temperature, wind related data • Transfer quantitative data to comprehensive audio information • Establish a criterion for determining if the weather is abnormal • Send alert message to radio station while there is abnormal weather
Should have	• Analyze the historical weather for Mali and qualitatively decide the threshold value for extreme weather (e.g.: 1% from the top) • Web interface for this system
Could have	• Include extreme weather alert in the voice response • Advertisement in the end of the phone call

	• Service support for other local languages
Won't have	• Advertisement in a variety of languages

12.Prototype description

Deployed and hosted on Heroku, the prototype, built mostly in Python 3.10, offers an insight into what the finished product could look like. The prototype can be accessed with the following information:

a. Description of the design decisions of your application.

Flask is a micro web framework written in Python. It is classified as a microframework because it does not require tools or libraries [6]. Flask is also easy to get started with as a beginner and easy to be getting a simple app up and running. For Database, we choose PostgreSQL because it's well supported in the Heroku Platform and easy to use.

b. Describe front-end and back-end implementation results.

Front-end:

1.Voice XML

Our service presents a simple and user-friendly voice call system with two main levels of files, the first being *language.xml*, which is used to select the user's familiar language at the first point of access. If the user's voice or numeric input does not correspond to the language we provide, then it returns to the main menu. In addition to this, the "language.vxml" file calls the initialization url on each user call-in, to keep all backend data up to date.

The second level of files is for the different language menus: "EN_menu.vxml" and "FR_menu.vxml". The logic of both files is basically the same, "FR_menu.vxml" has the setting for <xml:lang>. The following will be illustrated using the "EN_menu" file as an example.

When users select English, they will firstly be asked where and when to search. There are several choices for them to type or say, the events are triggered when the *location* and *day* are selected. The variables in the xml file are assigned values and a request is sent using <data method="post"> to transfer data to the backend. To prevent data reading errors, the id is assigned to each user in the backend. When we post user information, the user id is returned to the front-end in time.

Next is the weather broadcast module, where the user id is again posted into the backend, and the database created for that user is queried and returned as JSON data. Finally, end or return to the main menu option pops up and the service ends.

2.Web visualization

In order to access the weather report system with different applications, our team also implement a web-based front-end. The home page of the web weather report system is shown in Figure 3. Users can easily select the location and date from the Location and Data drop lists. After clicking the 'Ask Weather' button, the web page will be linked to the weather report page which is shown in Figure 4.Also, the web application also provides an alert page to notify the user if there is an extreme weather during the consult day.

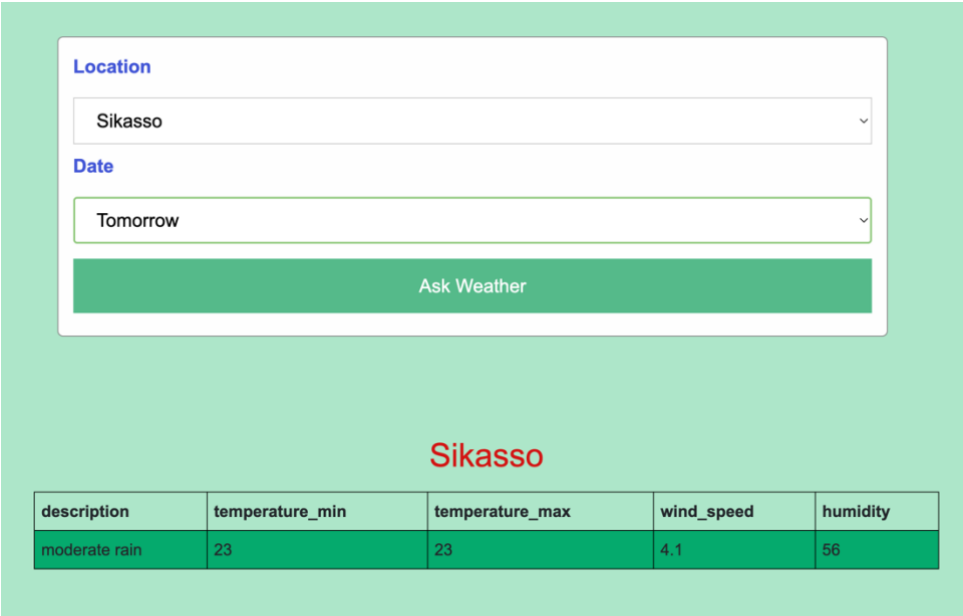

Figure 3 Web visualization example of weather inquiry and alert content.

Figure 4 Webpage visualization example for alert content.

Back-end:

In the back-end part, we use a Flask framework to implement all the functions. The three main functions of our application are as follows:

1. Getting data from local weather report providers. As mentioned in the Front-end part, the main purpose of the interaction between the front-end and back-end is to transfer the weather data. So we must prepare weather data in our back end. The source of the weather data is OpenWeatherAPI [7]. The OpenWeatherAPI provides comprehensive and informative weather data for most of the countries in the world. Due to the tight timeline, now we only use some of the basic information such as Temperate, wind, and humidity from this API.

2. Providing the data storage and manipulation tools. The raw data from API cannot directly be returned to end-users. We also developed a tool to easily transform the raw data into what we need in the database. With this tool, developers can easily manipulate the PostgreSQL database which is our main data container. Data analysis is also needed in our application. Our alert information is produced by processing the raw data using some human-defined rules.

3. Providing several endpoints to handle the requests. The requests from Voxeo and the webpage are the route to different endpoints by different APIs. The complete workflow is:

1)Voxeo and webpage send a request to our API.

2)These requestion are routed to different functions.

3)Retrieve the needed data from the corresponded data table and transfer it to the right format.

4)Finish the execution and return the information to the Front-end.

13.Pointer to the Application code

This URL contains all related codes: https://github.com/WillemGerritzen/ict4d

14.Pointer to how to access the application

The application would be deployed on Heroku.

Heroku welcome page: https://hows-the-weather-tmmr.herokuapp.com/

Query weather page: https://hows-the-weather-tmmr.herokuapp.com/webForm

Alert info page: https://hows-the-weather-tmmr.herokuapp.com/alert

To run the application:

- Call the voxel service on: 0208082848, pin code: 9990521878.
- Open query weather page to get weather detail information.
- Open Alert information page to check alert information.

Where in this webform, we can check the forecast information of a specific city for seven consecutive days.

15.Short Usage scenario

Users may get weather information by dialing a phone number and selecting options from the phone menu using the dial pad or directly saying the number. It would first be necessary to choose a language, including English or French. Second, users can pick from Bamako, Ségou, Sikasso, Kayes, and Nara as the location for forecast information retrieval. The third option is to choose a time from today, tomorrow, or the day after tomorrow. The users would then be informed about the weather, maximum and minimum temperatures, wind speed, and humidity. If the users desire, they can select a different city for forecast information.

16.Feedback questions

Our group met with Francis Dittoh at the start of the project to get feedback on the prototype development. These are some of the questions we asked.

- When do users need a weather forecast the most, and how long in advance do they would like to know the information?
- In which regions people can only retrieve weather information through phone calls.
- Which weather information will users be primarily interested in?
- In addition to calling for weather information, are there any other preferred ways to obtain weather information?
- How much are users willing to spend on this, like are they willing to pay a subscription fee?
- How comfortable you are with hearing the advertisement during the weather broadcast.

- What form of information should we convey to the radio station?

17. Discussion of Scope and Fidelity

We only implemented English and French in our prototype since VXL provides us with common languages to convert text to speech, but Mali has 13 native languages that are still spoken. These language speakers are mostly from rural locations, and they will use this weather prediction system more than urban residents. It is important to record voice files for these rare languages, which is an aspect that should be improved.

The technology is designed to deliver voice-based weather prediction information to all Malians. However, only five main cities may be chosen, including Bamako, Ségou, Sikasso, Kayes, and Nara, because the OpenWeatherMap API can only access data from these locations. For the timeframe, only the weather information that allows the user to select three days is finally implemented. The weather API can only provide information for the next seven days, because the closest prediction information should be more precise, only three days of query is implemented in the prototype of this phone call. It is possible to enquire on the web page version about it for the seven following days.

The digital information in this prototype is not fully translated into descriptive information that is simpler to understand. Because they have no idea of these statistics, this may not be particularly user-friendly for individuals who do not frequently follow weather forecasts. However, knowing precise meteorological data should be more intuitive for farmers, therefore if we want to make the service more accessible for more people, we could add an option to the menu to offer precise numbers. For example, if you want to offer a comprehensive weather forecast, you may break the numbers into intervals and give matching descriptions.

18. User evaluation

We invite several volunteers to assist in the testing of the calling service, as finding local Mali customers to test this prototype is difficult. Overall, collecting current weather information is a simple process that all participants can complete. However, there are a few things that have an impact on the user experience. The first is that English interaction is not as good as French communication, which might be a Voxeo problem. Second, the pronunciation of many English place names is a little ambiguous. The French service, on the other hand, sounds more coherent, but has larger sentence spacing and alternates between male and female voices. These issues usually do not prevent the function from being completed, so if time allows, they can still be handled. Some participants also claimed that inputting numbers into mobile phone menus in a loud environment was affected. To enhance, it may allow users to select whether to utilize voice input or the keypad from the start.

19. Conclusions

In conclusion, *How's the Weather Tomorrow* is a multilingual weather report system for Mali, using Voice XML and Heroku cloud application platform. This application considers the low literacy and multilingualism of the Malian people, with a simple and practical process. Furthermore, in addition to the usual weather forecast function, we have provided an extreme weather alert function for a safe life of the Malian people.

In the early stages of the project, due to the uncertainty and changeability of the project requirements, we did not have a defined interface standard, which led to a separation of development. In the rest of the development phases, we achieved parallel development by developing interfaces standard between the front-end and back-end. Benefit from the concrete interface standard, our development process becomes more efficient.

However, the current prototype still has potential for development, such as the fact that it only realizes functionalities in English and French. There are still a variety of national languages that are extensively being used by Malians, especially farmers in rural regions. This could be solved by adding the audio files for each weather cases in the database. Furthermore, this prototype has only been tested by a few volunteers, and there are no participants who have similar experience with the local people, thus this prototype just passed the technical testing, not sure that if the provided information would be enough, although receiving some suggestions from user representatives at the beginning of the project.

Reference

1. *Elo Merchant Category Recommendation | Kaggle*. (2019). Kaggle. Retrieved April 17, 2022, from https://www.kaggle.com/competitions/elo-merchant-category-recommendation/overview
2. ACCORD. (2022, February 8). *Climate change and violent conflict in Mali*. Retrieved April 18, 2022, from https://www.accord.org.za/analysis/climate-change-and-violent-conflict-in-mali/
3. *Database Modeling with UML | Sparx Systems*. (2010). SPARX SYSTEMS. Retrieved April 27, 2022, from https://sparxsystems.com/resources/tutorials/uml/datamodel.html
4. Baart, André & Bon, Anna & Boer, Victor & Dittoh, Francis & Tuijp, Wendelien & Akkermans, Hans. (2019). Affordable Voice Services to Bridge the Digital Divide: Presenting the Kasadaka Platform.
5. *Mali: Orange tariff plans for terminatio*. (2019). Learn and Support. Retrieved April 29, 2022, from https://goantifraud.com/en/blog/1160-mali-orange-tariff-plans-for-termination.html#studysubscribe
6. *Flask (web framework)*. (2022) Wikipedia. Retrieved May 22, 2022, from https://en.wikipedia.org/wiki/Flask_(web_framework)
7. OpenWeatherMap.org. (n.d.). *Weather API*. OpenWeatherMap. Retrieved May 22, 2022, from https://openweathermap.org/api

20.Appendix
20.1 Figures

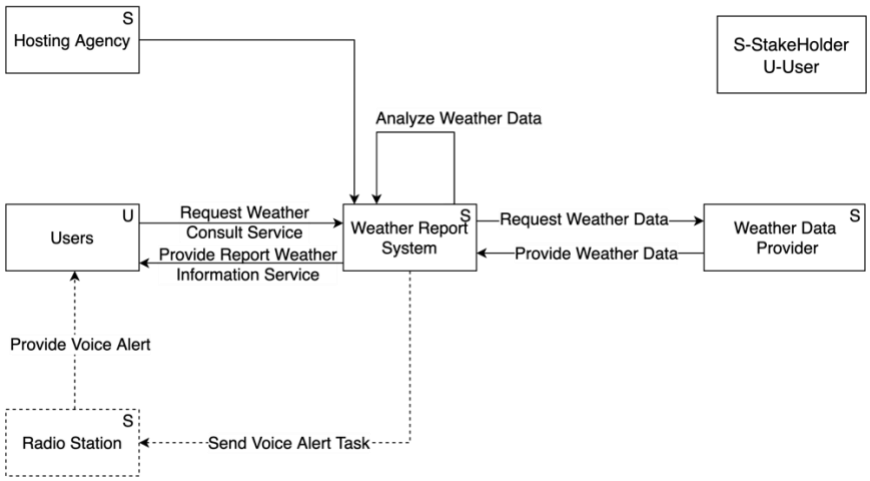

Figure 5 Business domain of the Weather Report System.

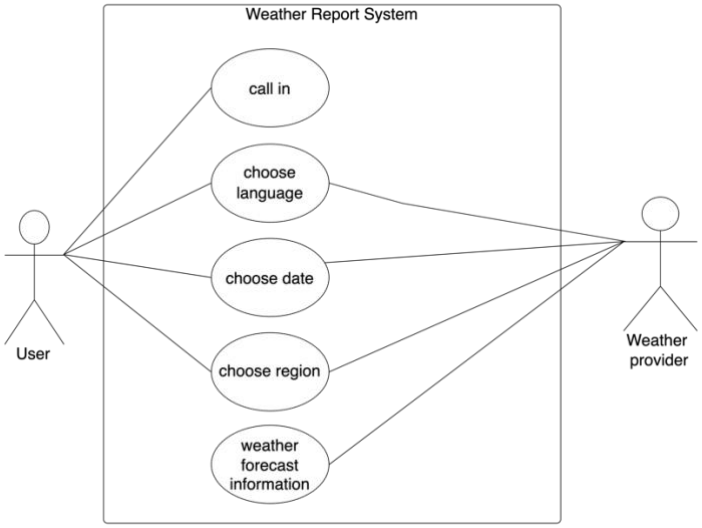

Figure 6 Use case of the Weather Report System.

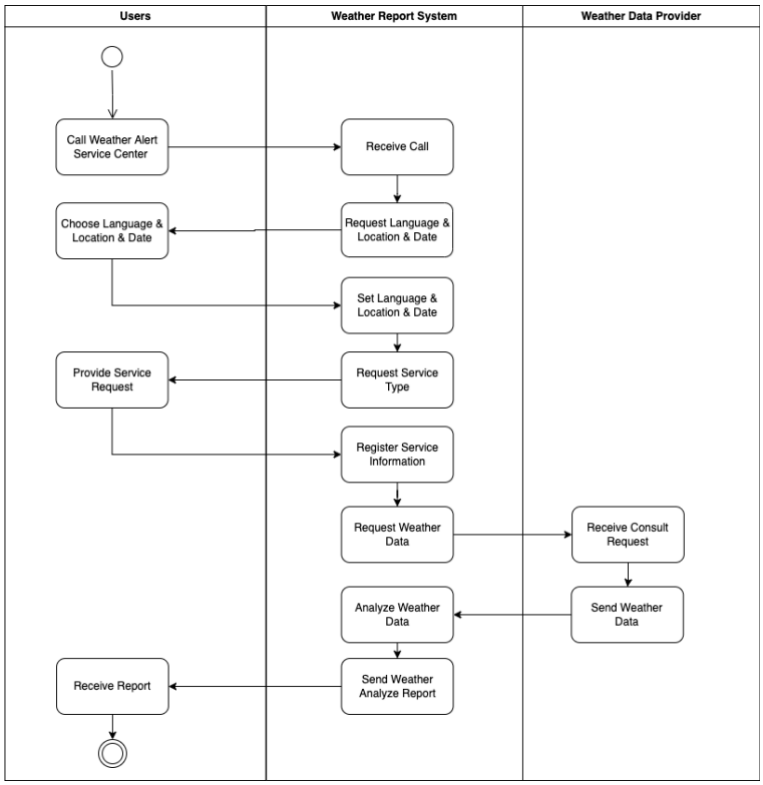

Figure 7 UML Diagram showing the interactions within users, Weather Report System and Weather Data Provider.

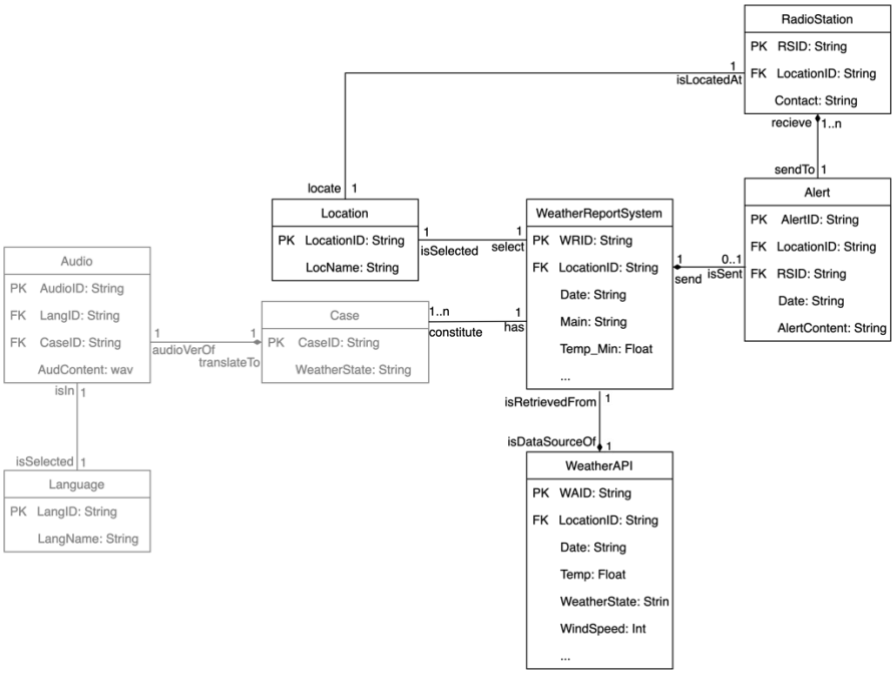

Figure 8 Class diagram for database of Weather Report System.

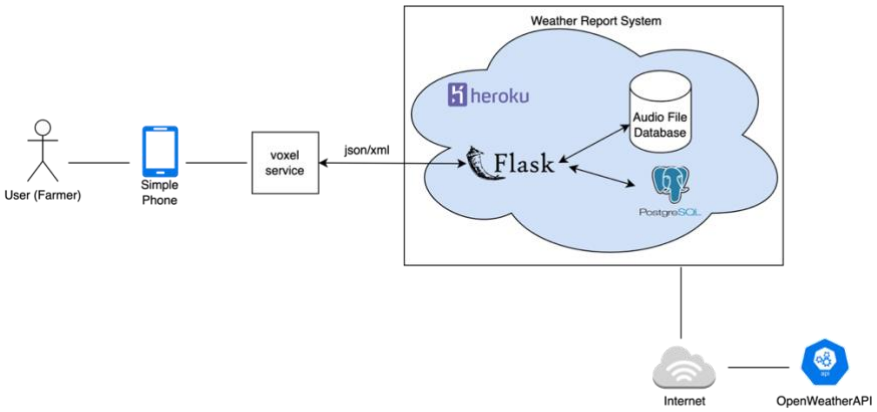

Figure 9 Technology infrastructure diagram of Weather Report System.

20.2 Application

Phone Number: +31 20 8082848

Code: 9990521878