The Kasadaka Weather Forecast Service
ICT4D Assignment 3

Kim Bosman 2524547

Jari Ferguson 2529692

May 26, 2016

1 Introduction

During trips to Mali and Ghana, the ICT4D department of the VU Amsterdam talked to local villagers through a couple of Living Labs sessions. During these sessions, they could express their needs and wishes for developmental support. These Living Labs resulted in use cases which - once solved - could potentially improve the lives of the villagers drastically. An example of the obtained use cases is receiving more information on animal diseases. If farmers can identify or further specify the illness of an animal, they will be able to estimate the seriousness of the illness. A proper evaluation on the illness could mean the difference between life and death for the animal. A veterinarian can be consulted further, at an earlier stage and the proper care can be provided. Another use case gathered on the field trips to Ghana and Mali concerns weather forecasts. This use case will be further explained in this paper.

1.1 Current situation

In 2010, 66% of Mali's gross domestic product was agriculture[1]. Even though agriculture is extensive, it is comparatively unproductive. For example, in 2010, the sorghum yield in Mali was 1 ton/ha against 4.5 tons/ha in the United States.

One of the main reasons for this difference is the lack of reliable weather information. Accurate information about the near-future weather is very important for farmers. Especially information on incoming rain is very important. Based on rain forecasts, farmers will be able to adjust the watering plans for their crops. If a rainstorm is expected, farmers will not water their crops. Also, the farmers will be able to collect the rain water and safe it for days of drought.

At the moment, weather forecasts for farmers in Mali are sparse. Radio stations either do not broadcast at all, or the broadcast is for a region that is too spread out and thus not specific enough for different regions. Increasing the quality of the forecasts should increase the crop production, since it would prepare the farmers better for the weather that is to come.

[1] http://www.fao.org/fileadmin/templates/mafap/documents/Mali/MALI_Country_Report_EN_Feb2013.pdf

1.1.1 Key idea

In an attempt to help the farmers reach their goal of growing more crops, the idea for the weather service Mali came to mind. The key idea of this prototype is to provide farmers in Mali with an opportunity to call directly to a weather service, in order to obtain weather forecasts for the area of interest. After the initial deployment, the system should work autonomously. The idea is to assist farmers in deciding when to take action regarding sowing and harvesting of crops. However, this service can be useful to many other businesses that are dependent on the weather as well.

1.2 Actors and goals

Table 1: Actors and operations goals

Actors	Operations goals
Farmers and other users depending on weather information	Learn when the weather is suitable for sowing or harvesting
System owner	Provide a working system
Weather Stations	Provide accurate weather information

1.3 Scope

Mali is a country of 1.240.192 km^2, and is home to a population of 14.5 million[2] in 2009. Of the nine districts that can be found in Mali, the three Northern districts Tombouctou, Gao and Kidal only host 10% of these 14.5 million. This is why the decision has been made to focus on the Southern part of Mali for now, which means that the prototype will only feature the remaining six districts. The used districts are marked in figure 1.

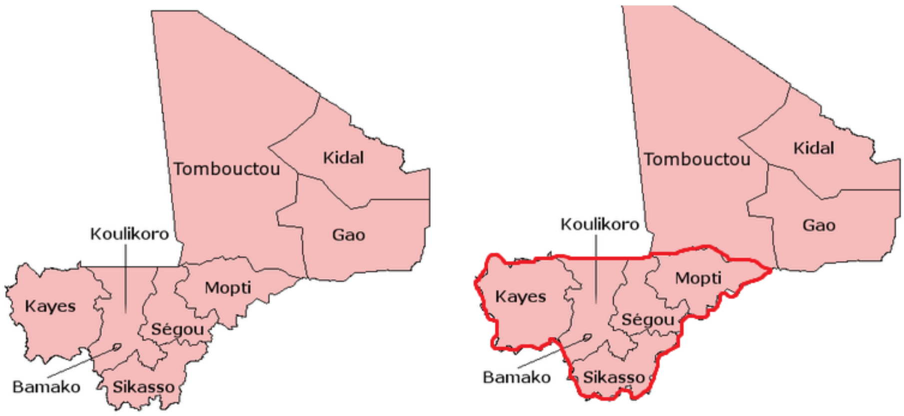

Figure 1: Mali (left) and the used districts (right)

In order to provide the most accurate weather forecast, each district will be represented by four cities. These four cities in turn represent the North, East, South and West of each district. The only exception is Bamako, which is the

[2] http://populationcommunication.com/wp-content/uploads/2014/06/Mali_report.pdf

capital of Mali. The capital represents its own district, and thus all regions of this district as well. The chosen cities for each district can be found in table 2.

1.3.1 Language

Mali is a multilingual country which offers around forty different languages. Since French is the official language of Mali[3], the prototype will feature both French and English voice options. The English language is mostly used for the tests, since none of the developers speak French extensively enough for proper testing.

Table 2: Districts and their cities

District	Kayes	Kolikoro	Sikasso	Ségou	Mopti	Bamako
City	Kayes	Nara	Sikasso	Ségou	Mopti	Bamako
	Kita	Kolokani	Koutiala	Alatona	Boni	
	Béma	Dioïla	Bougouni	Falo	Koro	
	Sitakilly	Karan	Kadianna	San	Dogo	

1.4 Context

Although aimed mainly at farmers, anybody in need of weather forecasts can phone the forecast service. People in need of a weather forecast could include people who plan to travel and would like to check if the weather is expected to be suitable. When users call the system via the number +31 6 83 86 33 17, they can select a region and a place for which they want to hear the forecast. Forecasts are regularly updated and are provided for up to five days including the current day. The system will convey the forecast using basic verbal expressions in multiple languages, so it is accessible for everyone.
If a user calls the service, the first menu that will be provided is the language selection. Then the user selects a region from a list of possible regions. Next, the user selects a place or town closest to her or his location. Following, the users selects for how many days the forecast should be given. The system will obtain the raw weather forecast data from an external service provider and translate the data into verbally expressed weather predictions that the caller can hear in their chosen language. Figure 2 shows an overview of the system. The system can be considered to be successful if it is used by farmers in all the regions and places that the system provides forecasts for. This should be made measurable by analyzing the usage of the system (i.e. when is it used, how often and for which places are the forecasts being asked for). The success of the system also depends on the accuracy of the forecasts - do the forecasts match the weather that occurs? If the predictions are accurate, it should translate into a higher adoption rate. Conversely, if the predictions are often incorrect, the adoption rate can be expected to drop.
Important preconditions are that the system is hosted at a location with the suitable resources. These include a clean and secure environment, with a reliable power supply, and a steady Internet connection for data retrieval. Furthermore, the system will need access to the mobile phone network, in order for end-users

[3] https://en.wikipedia.org/wiki/Languages_of_Mali

to be able to call into the system. Lastly, the systems depends on the end-users being aware of the system and having the phone-number that is needed to call the service.

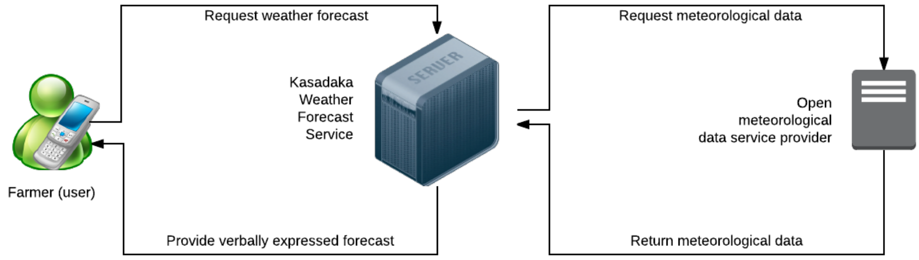

Figure 2: Overview of the system

1.5 Use case scenario script

For end-users, such as farmers, using the intended system is straightforward. The steps involved are as follows:

1. The end-user calls the number of the service
2. The system offers a choice of languages
3. The end-user selects a language
4. The system offers a selection of regions to choose from
5. The end-user selects a region of interest
6. The system offers a selection of places in the chosen regions
7. The end-user selects a place of interest
8. The system offers the user a number of days (up to five) for the forecasts
9. The end-user selects a number of days
10. The system gives a verbal weather forecast for each day the user has selected

1.6 Interaction and communication

Figure 3 shows the activity diagram for the Kasadaka Weather Service system.

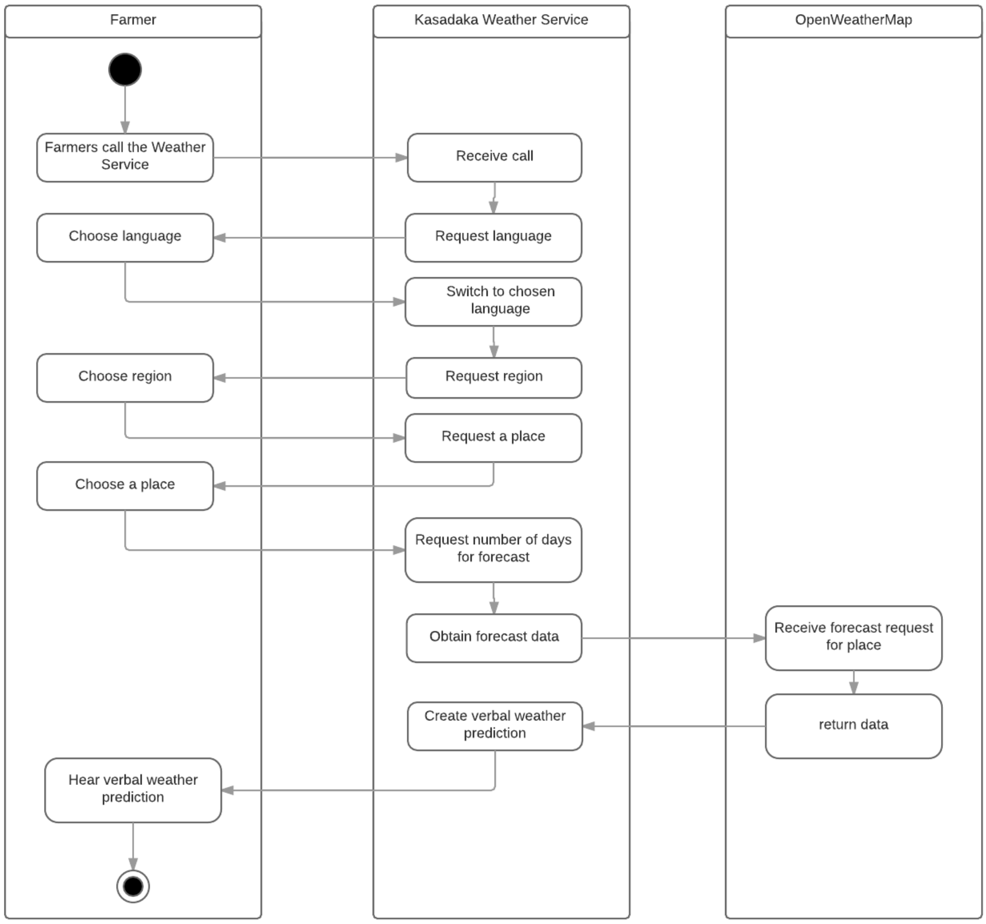

Figure 3: Activity diagram for the Kasadaka Weather Service

1.7 Information concepts

Central to the system is the weather forecast data. This meteorological data is retrieved from OpenWeatherMap (OWM) for individual locations and is stored in numerical format in a local database. Since the number of places for which the system should provide forecasts is larger than the number of possible choices of a phone's keypad, the supported places are split into regions. To make a recognizable distinction between regions, it was decided to use geographically recognized regions.
In order to request forecast data from OWM, it is necessary to provide GPS coordinates of the place in question. For each queried place, OWM returns meteorological data. Of all provided data, the following is considered to be relevant:

- (dt) = time-stamp of the forecast data (to which time does it apply)
- (temp) = predicted temperature for the forecast (given in Kelvin)

- (rain) = predicted rainfall for the forecast (given in millimeters)
- (windspeed) = predicted wind speed for the forecast (given in m/sec)
- (winddirection) = predicted wind direction for the forecast (given in degrees)

This data is summarized in the entity relationship diagram shown in figure 5 on page 14. In order for the data to be meaningful to the end-user, the data needs to be translated into verbal expressions. Also, the granularity of the predictions needs to be considered. OWM provides up to eight prediction per twenty-four hours, but it does not mean that the end-user should be provided with an equal amount of predictions per day. Based on feedback on the second prototype, this has been reduced to two predictions, namely night and day. This means that the eight prediction need to be translated not only into verbal expressions, but they also need to be summarized. Numerical temperature predictions are averaged and translated into *freezing*, *cold*, *warm*, *hot* and *very hot*. Numerical rainfall predictions are averaged and translated into *rain*, *little rain*, *moderate rain* and *intense rainfall*. Wind speed is also averaged and then translated into *light air*, *light breeze*, *gentle breeze*, *breeze*, *fresh breeze*, *strong breeze*, *near gale*, *gale*, *strong gale*, *storm*, *violent storm* and *hurricane*. Wind direction is averaged and translated into *north*, *north-east*, *east*, *south-east*, *south*, *south-west* and *west*.

1.8 Technology infrastructure

Although the system is designed to function autonomously, it is recommended that a system administrator can at least access the system remotely (via the web interface or SSH). Furthermore, due to the limited life expectancy of the storage card (microSD), it is recommended that the system administrator maintains a backup strategy, at least whenever changes have been made to the system. The most basic version of the Kasadaka Weather Service system consists of the following components (hardware and software):

- A RaspberryPI (model 2B or 3B) (hardware)
- A GSM dongle (hardware) with SIM card for voice cards
- Internet access via local network
- Apache HTTPD 2, PHP5 (with SQLite support enabled)
- Asterisk PBX software
- Either a remote computer with access to the system or a screen, mouse and keyboard connected to the system
- A reliable power supply and a UPS system[4] to prevent data-loss

1.9 Cost considerations

In order to properly offer the service, it is important to have an idea of the predicted costs necessary for maintaining the system. We identified maintenance, infrastructure and development costs.

[4]`https://en.wikipedia.org/wiki/Uninterruptible_power_supply`

1.9.1 Infrastructure costs

The hardware used for this prototype consist of the KasaDaka[5]. This is a low-cost, credit card-sized computer which runs on Linux. The next thing needed is a GSM network, which is delivered via a GSM dongle combined with a SIM-card. Adding a local telecom service could either be a prepaid service or a monthly one. Also, an internet connection is needed in order to retrieve the most accurate weather information. If this isn't available the system will make a prediction based on old weather data.
Keeping in mind that the average temperature of Mali lies around the thirty degrees Celsius [6], we have to consider that there might be a need of a storage place, which will bring along additional costs like rent and facilities.

1.9.2 Maintenance and development costs

As of right now, there are no development and maintenance costs since this is a student project. However, if the prototype turns out to be desired it would be interesting to consider continuing with this project. In order to support the maintenance costs, we could ask the users to pay a small amount of money each time they use the service. This will at least partially fund new developments and needed maintenance down the road.

1.9.3 User costs

The system itself will mainly be used by local farmers which need information about the upcoming weather. Right now, the only costs the user will make is the cost of a phone call to the provided number. However, as mentioned before, we might consider asking for a small donation in the future.

1.10 Feasibility and sustainability

Making a weather service is not something new. It's actually something that is pretty well used in Western countries. So the possibilities of designing a weather service for countries like Mali are high enough to make it feasible.
Sustainability is something that has to be thought of as well, since sustainability is important for the earth we live on. Working with the Weather Service will result in a more sustainable use of the available water, both gained from rain and stored for emergencies. By making accurate predictions, farmers do not have to use their water supply if it will rain within a short period of time, and they can even store the rain that is falling to fill their water supplies. This will not only lead to a more efficient way of distributing the water, but it will also lead to a more regular watering system for the crops. The crops will not get too much or too little water anymore, and thus the harvest might increase. This will lead to a worldwide increase of food, which is needed desperately.

[5] `https://kasadaka.wordpress.com/hardware/`

[6] `https://weather-and-climate.com/average-monthly-Rainfall-Temperature-Sunshine,bamako,Mali`

1.11 Key requirements

The key requirements of the system can be divided into three sections, namely: must have, should have and could have.
Must-have:

- Internet access to OpenWeatherMap in order to retrieve forecast data
- Phone-access for end-users to dial into
- Language support for the most-used languages
- A local database for buffering/storing forecast data

Should have:

- Language support for all languages
- A web interface for remotely managing the system
- A report tool to analyze the usage of the system

Could have:

- Configurable translations of numerical data into verbal expressions

2 Contextual issues

One of the biggest contextual issues that had to be conquered during this project was the cultural difference. With cultural differences, we are mostly talking about interpretations. For example, thirty degrees Celsius is hot in the Netherlands, but is this also considered hot in Mali? The same can be asked for rain, four millimeters of rain is not considered a lot in the Netherlands, but it might make a huge difference in Mali. Working on a use case that was provided did not make this issue any easier, since the information is not gathered by the developers. This made it difficult to make a real local-usable prototype, that was understandable by the end users. The feedback provided from Mali was therefore extra helpful, and the prototype got adjusted according to it.
Another contextual issue was the language barrier. Of course the first implementation was in English, but we soon figured out that the main language in Mali isn't English at all. The most spoken language is actually Bambara, but due to time and skill constraints we decided to implement French as the second supporting language. We do believe that this will not satisfy all end users, and that implementing local languages like Bambara are a must.
Finally, the poor infrastructures in Mali were a contextual issue as well. In the Netherlands, everything is taken care of perfectly. In Mali however, the infrastructures are lacking or sometimes not available at all. The Weather Service needs some basic infrastructures to be able to work. Developing and testing the service in the Netherlands meant always testing a working system. Bringing the service to Mali however does not guarantee that the service will actually work. In order to compensate for at least the unsteady internet connections, we decided to incorporate an "offline mode" which will make predictions based on the weather data that is in the database already if there is no internet connection available.

3 Theoretical background

The prototype introduced in this paper is largely based on research that was already performed by the ICT4D department of the VU Amsterdam. The Kasadaka[7], that is the center of our prototype, is a project that build information acquired by the Web Alliance for Re-greening in Africa (W4RA)[8]. The Kasadaka contains the technical requirements that are context specific. For example, most of the farmers in rural Mali are illiterate. The Kasadaka supports voice-based communication tools to make the information exchange accessible for everyone. Other than that, the Kasadaka is affordable due to the low cost components it consist of, which is ideal for rural areas, in which the infrastructures are not that optimal.
A big inspiration for this prototype was the project RadioMarché[9], which works with both voice-based technology and a web service. Combining these technologies makes the service accessible for literate users, but also for illiterate users. This should be the end goal for every development project, since you want to help as many people as possible with your service.

4 Solution

The solution to the irregular weather forecasts is the Kasadaka Weather Forecast Service. As in any prototype design, some design choices had to be made. The following subsections provide an overview of these decisions and the rationale behind it.

4.1 Design choices

4.1.1 Use Beaufort scale for wind speed

The Beaufort scale for wind speed is internationally accepted and provides verbal expressions for a large range of wind types.

4.1.2 Use compass directions for wind direction

Using the compass directions for wind directions is more natural than using degrees. People will understand these compass directions better than degrees.

4.1.3 Store forecast data in a database

Since OpenWeatherMap only updates the forecast data every three hours, it was considered to be a waste of bandwidth to retrieve the data for each received call. By comparing the time-stamp of the last update with the current time, unnecessary requests to the API can be avoided.

4.1.4 Only retrieve meteorological data when needed

The decision was made to only query the OpenWeatherMap system when the meteorological data is needed. When a call is received, requesting forecasts for

[7]http://kasadaka.com/about.html
[8]http://w4ra.org/
[9]http://w4ra.org/radiomarche-voice-based-market-information-system/

a particular region, the system will first query its local database. If the data does not exist or it is considered to be too old (older than three hours), the system will query OpenWeatherMap first and store the response in the local database. Then it will use this data to provide the forecasts to the caller. This is illustrated in figure 8 on page 22.

4.1.5 Start out with English and French as main languages

Since French is the official main language spoken in Mali, it was decided that this language had to be implemented in the first prototype. However, since none of the developers speak French, it was decided to also implement English for proper testing.

4.1.6 Provide wind and temperature forecasts

The first prototype that was built only provided forecasts for rainfall, as stated in the original use case description for this project. However, it was suggested by Francis Dittoh that the service should provide forecasts for wind and temperature as well. These additional factors are regarded as important by the end-users in Mali.

4.1.7 Using OpenWeatherMap for the source of meteorological data

In order to provide the intended system with weather forecast data, the decision was made to use the freely accessible data provided by OpenWeatherMap. Accessing global forecast data can be done via an API that they provide. Furthermore, the service provides forecasts up to five days, which is also requested in the original use case description. This service was also considered to be suitable, as the API provides data on predicted wind speed, wind directed, rainfall and temperature.

4.1.8 Using SQLite for local storage

For local storage, SQLite[10] was considered to be suitable, as it does not require any additional software installation and is well integrated with PHP. Furthermore, PHP is also well integrated with XML, so it was considered to be very suitable for generating VoiceXML[11] scripts, used by the Asterisk[12] component installed on the Kasadaka platform.

4.1.9 Provide web-based user interface

French is not the only language spoken in Mali. So the decision was made to allow the system to support multiple languages in a user-friendly manner. In the second prototype there was no easy means for doing this. Someone would have to install software in order to make the necessary changes in the local database, determine which additional audio-clips would be needed and upload these in a folder on the server. This limitation was overcome by providing a web-based user interface in which additional languages can be defined and linked to the

[10] `https://www.sqlite.org/`
[11] `https://en.wikipedia.org/wiki/VoiceXML`
[12] `http://www.asterisk.org/`

country that the system serves. Furthermore, the web interface should provide the option to upload the necessary audio-clips into a folder where the system expects to find these.

4.1.10 Supporting multiple countries

The decision was made that the system should not be built around one county only. It should be able to be deployed in multiple countries in a user friendly manner. In order to do this, a web-based interface was created in which countries can be defined. However, it was also decided that a single instance of this system should only support one country at a time. It was considered to be user-unfriendly if the user has to first select a country. Also, it was considered to be more useful if each country would have at least one system of its own. The web-interface provides a means of selecting the country that the system should serve.

4.1.11 Assigning languages to countries

Although the decision was made to support multiple languages, it was recognized that not all languages should be supported by each defined country. Therefore, it should be possible to specify which languages should be supported for which countries.

4.1.12 Web-based interface for specifying regions

It was decided that the system administrator should be able to manage the regions that the system supports. For instance, some regions may require a split into more regions. For the third prototype the decision was made that a user-friendly web interface would be more suitable for this than having to do this directly in the local database.

4.1.13 Web-based interface for specifying places

For the same reason as the previous design decision, it was considered to be useful if the system manager is able to specify places that are part of a supported region. Furthermore, by providing these using an interface, it would reduce the risk of introducing errors into the system.

4.1.14 Using GPS coordinates for specifying places

Because the OpenWeatherMap service does not recognize all the places in the world by name, it was decided that it would be more suitable if the GPS coordinates (latitude and longitude) are used instead. Furthermore, this would make it possible to specify places that do not have a formal name or are too large to be considered as one place.

4.1.15 Provide an export function of the caller history

It was considered to be very useful if the system can provide an overview of the usage. It was decided that the system should register each call made and store the selected country, language, region, place, number of days for the forecast and the time of the call. By allowing the system manager to download this data in a CSV-file, it can be used in, for instance, Excel, to analyze the system usage. It could, for instance, be used to determine which places or regions are most active. Also, it can be used to determine which languages are used most. The data can also be used to determine how much the system is used and if it may need to be scaled. Furthermore, it can also be for determining the expected revenue. It should be noted that this export function does not contain any personal information, such as the caller's phone number.

4.2 Field trip feedback

For the field trip to Mali, a few questions for specified feedback were designed. The feedback received was very helpful, the three main feedback points are further explained below.

- *Report frequency:* Prototype 2 of this system used four forecasts for each day, which included morning, afternoon, evening and night. However, from feedback it was learned that this frequency was too high. It was considered sufficient it the system provides a forecast for night and day only. This was therefore implemented as: night (00:00 - 11:59) and day (12:00 - 23:59). It should be noted that the same amount of meteorological data is used, despite the reduction of granularity.

- *Languages:* The second prototype only supported English and French. However, this would not be adequate for a large section of the target audience. Unfortunately, it was not viable to introduce additional languages at the moment. Instead, the decision was made to provide the system with a user-friendly method of introducing additional languages themselves.

- *Temperature index:* The translation of the temperatures used by OpenWeatherMap into verbal expressions was not considered to be suitable according to local standards. The rules that were used where considered to be very European. Using the feedback provided, a new set of translation rules was created.

5 Implementation

The following two sections describe the end-user interaction with the system and the web interface for managing the system.

5.1 The voice interaction component

The end-user of the system can interact with the system using a basic mobile phone. The user listens to the messages generated by the system and can respond to questions using the keypad on the phone. This kind of interface is provided by Asterisk. It relies on voiceXML scripts to determine how the interaction should take place. These can be static scripts, in which all possibilities are created beforehand. However, because this particular system relies on very dynamic weather data for a large number of different places and potentially for a large number of languages, it was considered to be better if the voiceXML scripts are generated at run-time. In order to generate these scripts, the decision was made to use PHP. VoiceXML, as implemented by Asterisk, allows additional voiceXML scripts to be loaded during execution, using hyperlinks. Furthermore, these hyperlinks allow GET-parameters[13] to be used to pass information to the scripts. As illustrated in figure 8 on page 22, the same PHP-script is called during the interaction between the system and the users. The choices that the end-user makes are passed on in order for the script to determine subsequent actions. When the user initially calls, the system will ask the user to select a language. The sentence of this particular question is expressed in each language. The possible language options are determined by the languages that are assigned to the country that the system is serving. The information is available in the local database. After the user selects a language, the remainder of the interaction will be spoken in the language of choice. In the following menu, the user is greeted. The greeting is personalized to the current time (for instance good morning or good afternoon). The user is then asked to select a region. The possible regions are listed in the database. After the user has selected a region, the user is asked to select a place. This division of regions and places is used in order to minimize the number of choices for each level in the interaction. Lastly, the user is asked to select the number of days for the forecast. The minimum is one day, which represents the current day. The maximum number of days is five. Once the user has provided the right information, the PHP-script can query the internal database for the requested forecast data. The user is then presented with the forecasts for each day. Each day consists of a night and a daytime forecast. After the forecasts have been given, the user is thanked for using the service and informed that the call will end.

[13] https://en.wikipedia.org/wiki/Query_string

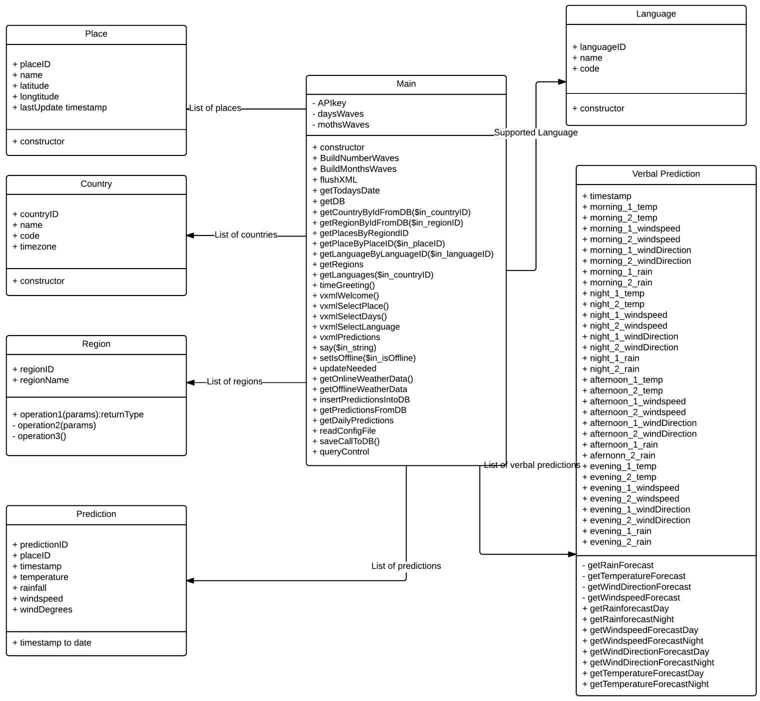

Figure 4: Voice interaction class diagram

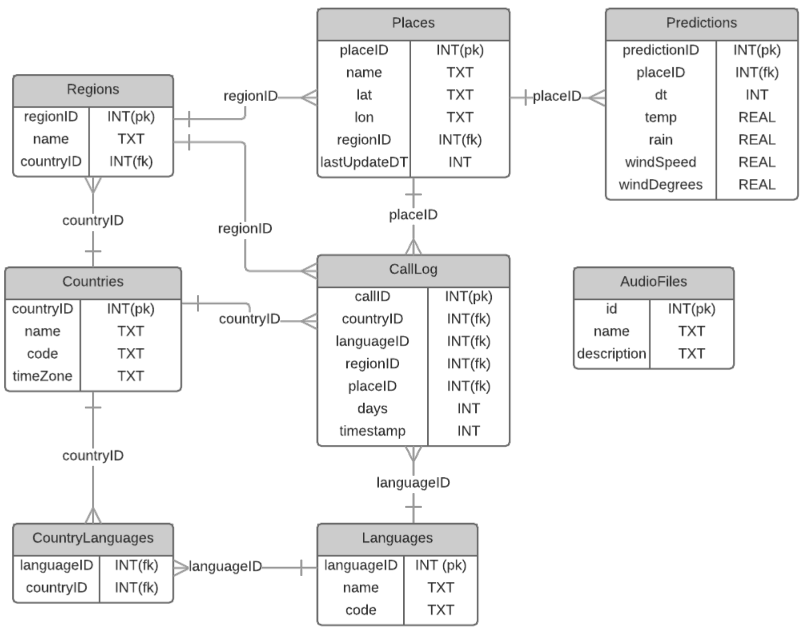

Figure 5: Database ERD

5.2 System Management Interface

Although the second prototype was already constructed to support multiple countries and languages, there was no easy way to implement these. In order to allow a systems administrator or implementer to deploy the system in other countries or to support additional languages, the decision was made to build a web-interface that provides the following functionality:

- Add, modify and delete supported languages, shown in figure 9, page 23
- Add, modify and delete supported countries, shown in figure 10, page 23
- Assign and remove languages to countries, shown in figure 10, page 23
- Add, modify and delete regions, shown in figure 11, page 24
- Add, modify and delete places, shown in figure 12, page 24
- Indicate which audio-clips need to be uploaded in order to support languages, shown in figure 13, page 25
- Provide upload functionality for a selected language and indicate missing files, shown in figure 13, page 25
- Allow the user to change the API key that is used to query OpenWeatherMap, shown in figure 14, page 25
- Allow the user to switch the system from offline mode (demo) to online mode, shown in figure 14, page 25
- Allow the user to choose the operational country (the system only serves one country at a time), shown in figure 14, page 25
- Allow the user to download a CSV-file of the caller history

The web interface was constructed using a single HTML5 script (indicated as index.html) and a single JavaScript script (indicated as kasadaka.js). This script is used to control the interactions of the various components on the web page. Any queries, inserts, updates and deletes for the database are performed by calling PHP-scripts that perform the actions on the server-side. Furthermore, the PHP-scripts are used to create directories on the server, for storing the audio-clips for each language. Also, a PHP-script is used to update the configuration file (config.json) on the server, which is used by the voice interaction component in order to determine the operational mode (online or offline), the operations country and the API key for OpenWeatherMap. Lastly, a PHP-script is used to generate the CSV-file that contains the caller history.

6 Usage scenario

The complete code for the application can be downloaded from
`https://www.dropbox.com/s/n8k5osd11v9ewg1/code.zip?dl=0`.
Also, an image of the system (for Raspberry Pi 2) can be downloaded from
`https://www.dropbox.com/s/lz7k9i2k5j25946/KasadakaV3%28rpi2%29_group_2.zip?dl=0`.

The system can be tested by calling the following phone number:
+31 6 83 86 33 17, and the web interface can be reached via:
`http://www.jsferguson.nl:6009/`

6.1 Example of voice interaction

Next follows an example of the interaction between a caller and the weather information service.
(Spoken in French) Select one for French
(Spoken in English) Select two for English
User presses 2
Good afternoon, this is the weather service for Mali. The date is 19 May.
Please select a region.
Select one for Bamako
Select two for Kayes
Select three for Koulikoro
Select four for Mopti
Select five for Sikasso
User presses 2
Please select a city.
Select one for Kayes
Select two for Kita
Select three for Bema
Select four for Sitakilly
User presses 4
Please select the number of days for the forecast, up to five days.
User presses 1
The weather for 19 May.
In the night it will be warm with light rain with a gentle breeze from the south-west.
In the afternoon it will be hot with no rain with a gentle breeze from the west.
You have reached the end of the forecast.
Thank you for using this service. This call will end now.
The system ends the call

7 Scope and fidelity of the prototype

All the functions that have been listed in the previous sections have been implemented and are fully operational in the third prototype. However, some functions are somewhat limited. These are:

- When a place is added to a region, it is necessary to enter the GPS coordinates manually.
- Audio-files that are uploaded to the system need to be in the correct format, before they can be used. A description of this process is available online[14].

[14] `http://www.voip-info.org/wiki/view/Convert+WAV+audio+files+for+use+in+Asterisk`

- Rules regarding the translation of meteorological data into verbal expressions are hard-coded into prototype 3. Although these rules may satisfy users in Mali, based in their feedback, it may differ from other countries or even regions. An option could be to allow a system manager to use the web interface to create these rules separately for each country. For example, the manager could give a range of values that determine when the temperature is regarded to be hot.
- The voice interaction component does not offer the user to repeat a message or to go back to a previous menu. This could be implemented in the voiceXML script using additional `<goto>` elements.
- The web interface does not restrict the number of regions or places to be added. However, because the user can only interact with single keypad numbers, these numbers should be limited to 9 options at each menu. This can be enforced in the web interface.

8 Plan

This section will describe the future deployment plans and the sustainability of the weather service.

8.1 Deployment

In order for the system to be deployed in a country, the following steps need to be undertaken:

- Determine the number of regions (up to nine) and create these in the system using the web based management tool
- Determine the places for each region (up to nine per region) and determine the GPS coordinates of these places
- Enter the data for each place and for each region
- Determine the languages (up to nine) needed to support the end-users
- Create and upload the audio-clips, using the web based management tool
- Find a suitable location for the system, taking technological requirements into account
- Provide the system with an activated SIM-card for phone access
- Test the system extensively, for each language and each place
- Announce the phone number for accessing the system, for instance via radio broadcasts
- Provide a phone number for gathering feedback with regards to the system

8.2 Sustainability

As described in the costs analysis, keeping the system operational will mean the system will generate costs. As the system can be expected to provide value to the farmers that will use the system, it is necessary to ensure there is a flow of revenue. It will be important to keep track of the costs that are generated and to distinguish fixed and variable costs. For instance, housing and electricity costs can be considered to be fixed costs, as these are independent of how much the system is used. However, variable costs will depend on how often the system is used. For instance, the amount of data that the system will retrieve from OpenWeatherMap will depend on the number of calls retrieved. The system is designed to only refresh forecast data when it is needed so that the amount of data exchanged over the Internet is reduced as much as is possible. By analyzing the call history data, which is provided using the web interface, the number of calls per period can be precisely determined. This data can also be used to determine the costs per call once the amount of calls stabilizes.

Besides determining the actual costs per call, it is also important to learn how much an end user is willing to pay for the service. An important question arises: How can end users pay for using the service? The system does not keep track of who actually called (caller-ID), although it might be possible with the Asterisk component. One approach may be to provide the service through a service number, which could be provided by a commercial telephone company. The end users would then call the service-number, which charges an additional predetermined fee and then automatically routes the call to the actual number of the Kasadaka-system. Such an approach would ensure that revenue is received and the end user only pays for services consumed. This might result in more users, since obliging users to pay a standard fee every month is not possible for a lot of users. Only calling when a forecast is needed is more feasible.

Furthermore, for such an approach, it is necessary to predict the expected amount of calls, in order to calculate the fee. This figure can be adjusted on a periodic basis, as more usage data becomes available. The value flow diagram in figure 6 on page 19 illustrates this.

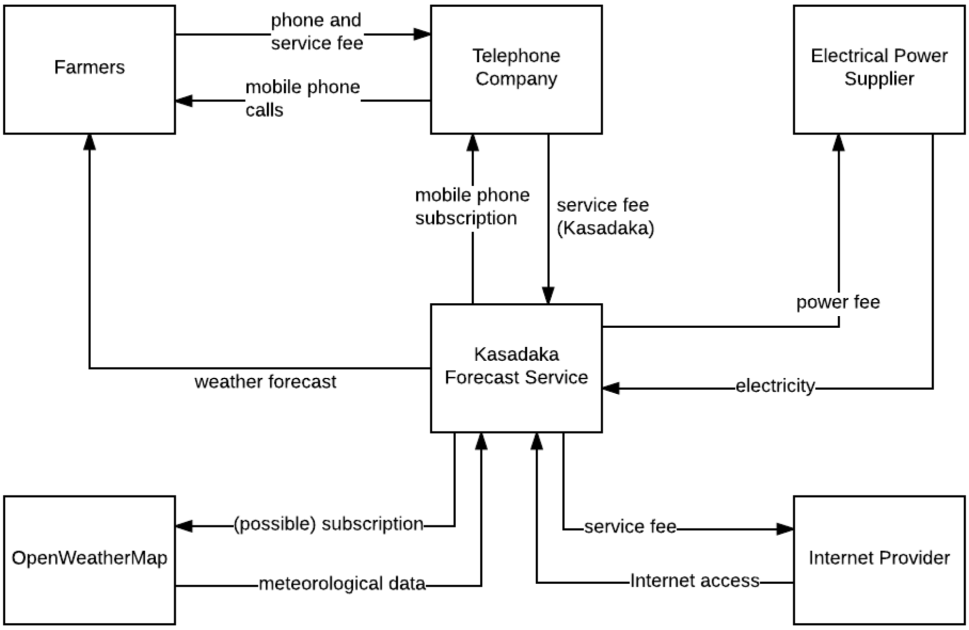

Figure 6: Value Flow Diagram

9 Evaluation

The feedback that was provided by the end users in Mali was very positive. This leads us to believe that the application was well liked. Besides the positive feedback, it is apparent that the application itself works like it should. The caller enters its region and the system will provide the accurate weather forecast as feedback. The current limitations are not a big influence on the end users of the Southern part of Mali, which is currently our most important region of interest.

10 Discussion, Conclusion and Future Work

10.1 Discussion

Besides the limitations mentioned in the "Scope and Fidelity" section, another limitation would be the fact that there are not enough weather stations in Mali, and a lot of other rural countries for that matter, to provide real accurate weather data specified for a small region. The goal should be to work towards adding more weather stations in these regions, especially since a lot of the worlds food is produced there. Accurate weather information is very important and can actually help improve the succession rates of each harvest.

10.2 Conclusion

The overall process of the creation of this prototype was a satisfying one. The prototype features everything that the concept included and even more that was thought of along the way. We hope that the final result will be perceived as useful and desirable for the end users. We believe that the Kasadaka Weather Service can be helpful in the battle against nature, retrieving more crops every harvest.

10.3 Future Work

A next big step would be to add more weather stations in rural areas like Mali or Sub Saharan Ghana. Currently, Tahmo is working on this project for Sub Saharan Ghana[15]. Unfortunately, we cannot influence this process for Mali. For us, the next step will be to work on the current limitations while waiting for more weather stations.
After that, we want to make sure that the implementation of the weather service will be completed for the whole of Mali and in the desired languages as well. After that is completed, we can focus on starting the same service in another country like Ghana.

11 Bibliography and Related literature

Most of the used literature and information used for this project can be found in the footnotes of the pages the information is used at. Other than those information sources, we would like to mention that the course material of ICT4D -both the lectures and the extra reading materials- were used as well.

Extra readings that contribute to the overall knowledge behind this paper:

- The Role of Technology in Environmentally Sustainable Development (1995), Chapter: Pathways to Sustainability. The chapter can be found on: `http://www.nap.edu/read/9236/chapter/5`
- Learning to see: Making value flow from End to End, John Shook (2012). The slides can be found on: `http://www.lean.org/Search/Documents/503.pdf`
- VOICES - VOIce-based Community-cEntric mobile Services for social development, seventh framework programme (2011)
- Communicating agrometeorological information to farming communities, A. Weiss (2000). The paper can be found on: `http://www.sciencedirect.com/science/article/pii/S0168192300001118`

[15] http://tahmo.org/

Figure 7: The logo for the Kasadaka Weather Service Mali

12 Appendix

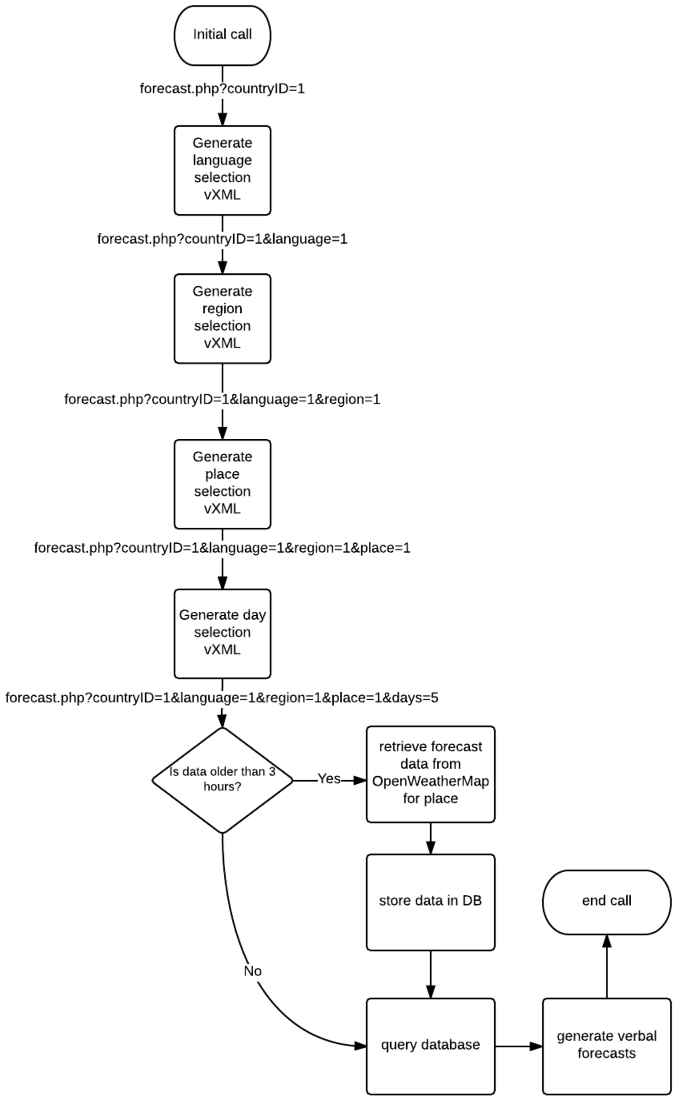

Figure 8: System flowchart

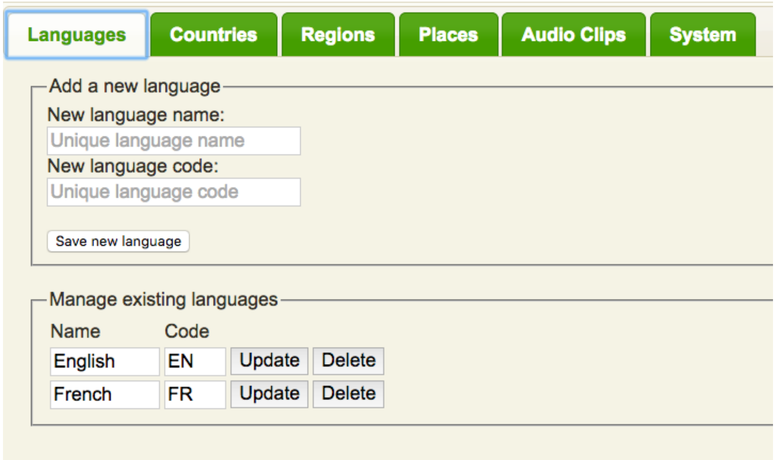

Figure 9: Adding, modifying and deleting languages

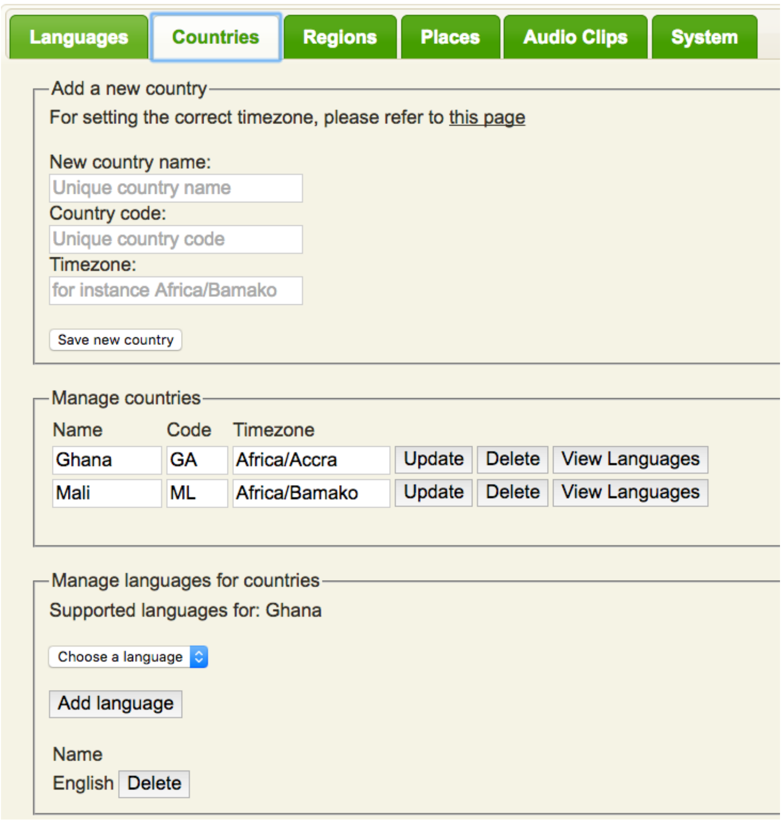

Figure 10: Adding, modifying and deleting countries and assigning languages.

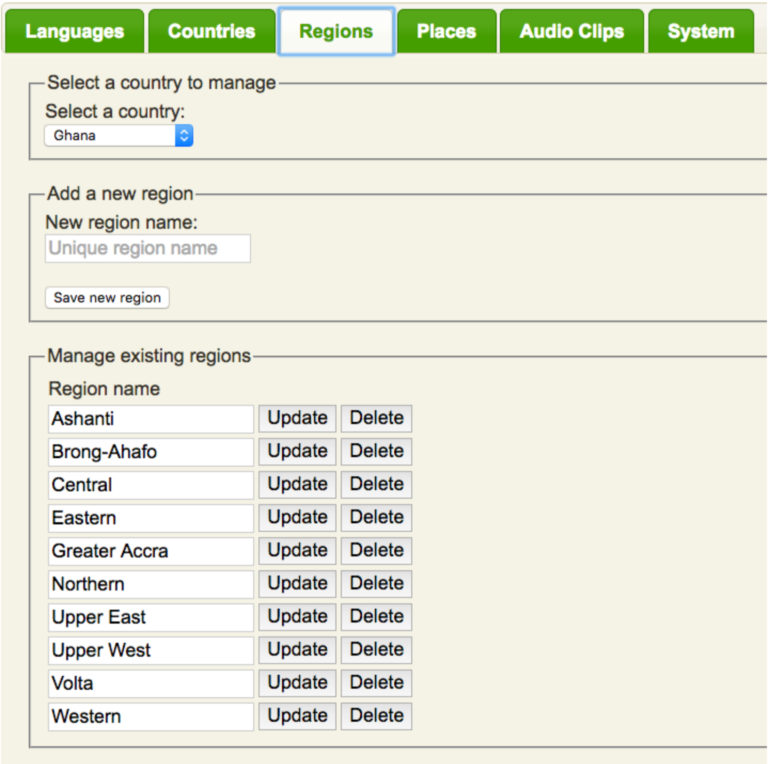

Figure 11: Adding, modifying and deleting regions.

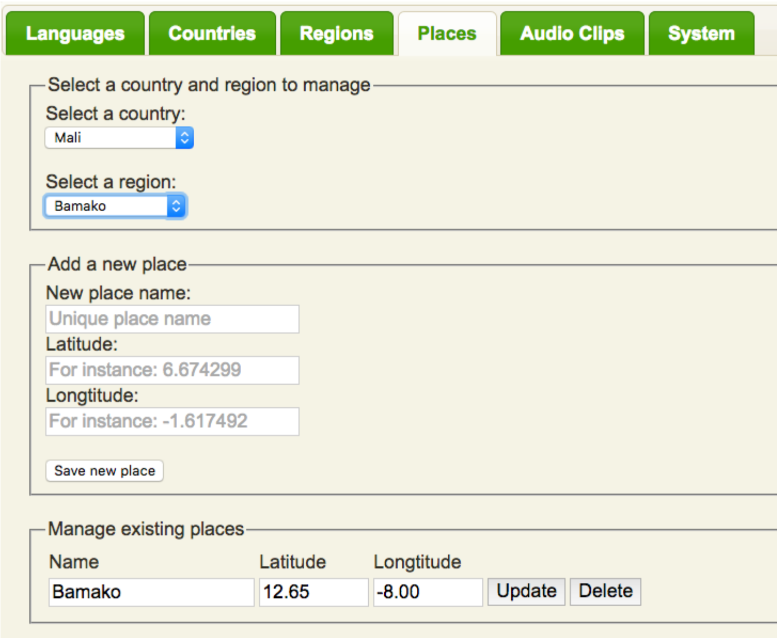

Figure 12: Adding, modifying and deleting places.

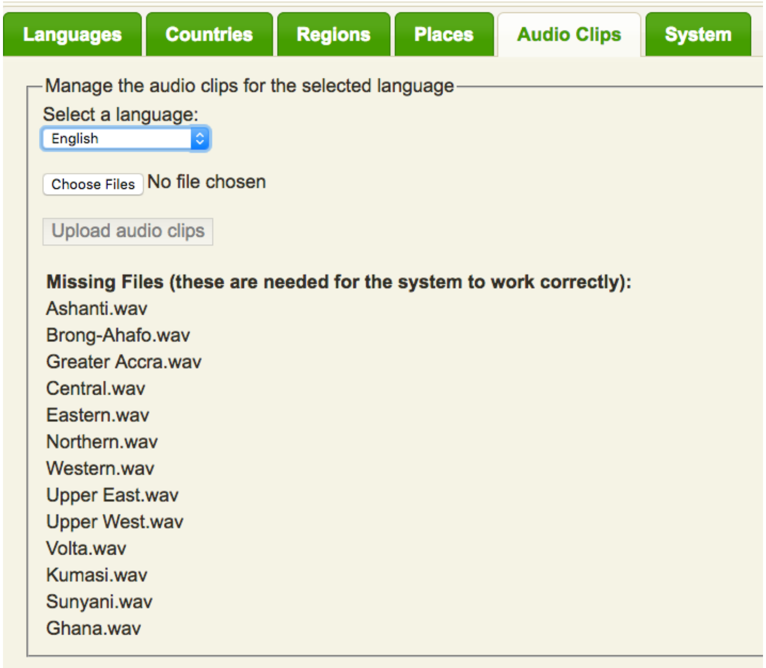

Figure 13: Uploading audio-clips for supported languages.

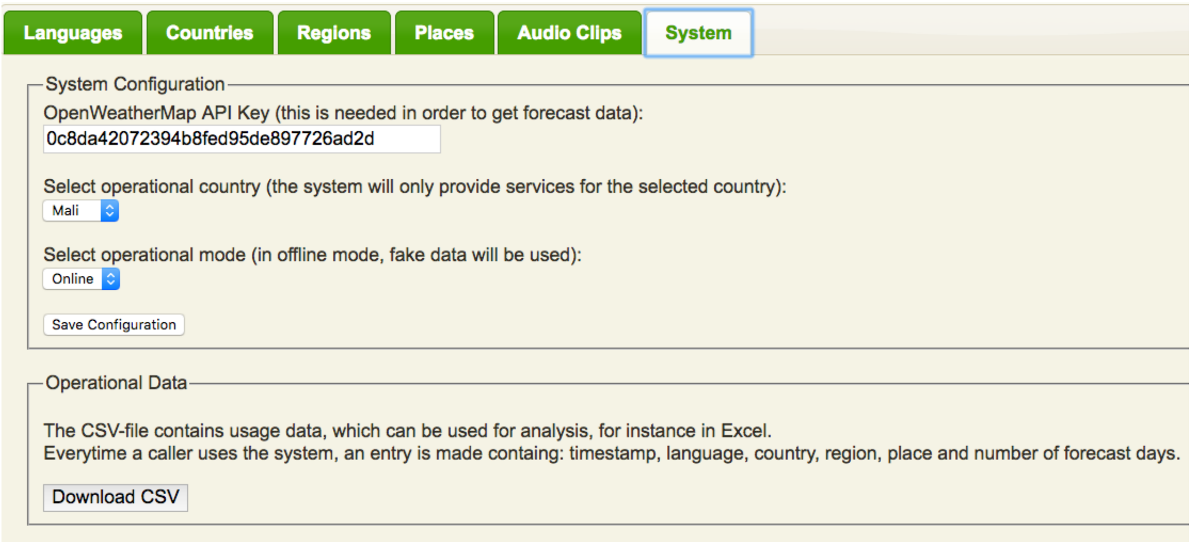

Figure 14: System configuration and downloading caller history data.