VP: Vaccination des Poulets 3.0 

A voice service application that helps chicken farmers in Mali to remember and to keep track of multiple chicken vaccines

Group 4
Linh Tran (2537431)
Rang Salih (2526302)
Denice Tuinhof (2537507)
Rick van Rheenen (2504112)

ABSTRACT
In this paper a system design for local farmers in Mali is presented. This system will remind farmers about (re-)vaccinations of their newborn chickens. While these farmers in Mali are often illiterate the designed server will exist out of an offline, multiple-language server which is reachable by call and fully functional by simple DTMF and voice in and output. The application is designed to support farmers in their daily life by performing a call reminder so that the chickens will be (re-)vaccinated within time and that chicken death will be lowered. Beside the needs of the user, the system itself and the context of the system, and additionally sustainability issues will be discussed is this paper. Context sustainability and economic sustainability are important aspects to think about when developing such a ICT4D-project. Maintenance of the systems is namely most difficult in these kind of projects, nevertheless with the V3-value net value sheets we tried to improve our chances on maintenance and sustainability. Lastly open data and linked data will shortly be discussed in connection to our system. In the future open data and linked data can be of major influence in retrieving knowledge. Therefore this is also an interesting topic to think about while developing and integrating such a system in Mali.

1. INTRODUCTION

Poultry diseases is a huge problem for bird farmers all over the world. The types of poultry where disease are the most common are chicken, turkey, duck, goose and ostrich [1]. Especially in poor countries a lot of the villagers live from livestock. Mali is one of these poor countries where 80% of the people rely on raising livestock (such as chickens), growing crops and fishing [2]. Raising chickens in Mali is a logical decision for the farmers, because chickens are easy to breed, feed and bring profit [2]. This profit can then be used to cover expenses and buy food. Nevertheless, the problem with chicks is that 70-80% of them dies before they are even one month old [2].

The most important diseases they are dealing with in Mali are the Newcastle Disease (Maladie de Newcastle), Chickenpox (Variole Aviaire), Infectious Bronchitis (Bronchite Infectieuse) and the Disease of Gumboro (Maladie de Gumboro). The Newcastle Disease virus is a bird disease which affects all types of birds and can also contaminate people [3]. The Newcastle disease virus was first identified in 1926 in Java (Indonesia) and later in 1927 in Newcastle (England), where it got its name [4]. Chickenpox, also known as Fowlpox, is a disease of frequent occurrence, especially for chickens that are not vaccinated. Chickenpox is not killing for every chicken, but very weak and young chickens could possibly die [5]. Infectious bronchitis is a respiratory disease of chickens and it very contagious. The disease is characterized by respiratory signs like; gasping, coughing and sneezing. Young chickens are often infected with this disease and can lead to severe respiratory distress. Gumboro, also known as “Infectious Bursal Disease”, is also a contagious disease that is often observed in young chickens. This is characterized by immunosuppression, which is the process of reduction of the activation of the immune system. The Gumboro disease can cause death at the age of 3 to 6 weeks [6].

To support farmers in Mali to protect their chickens from diseases, a system called 'Vaccination des Poulets' (VP) was set up. Vaccinating chicks when they are young, improves the survival rate immensely [7]. Chickens are mostly vaccinated with pigeon pox virus [5]. This vaccine is usually given between the age of 12 and 16 weeks and is done via the wing web method [5]. Nevertheless, there are certain features that are important while vaccinating [8]. One of these features is that following up vaccination should be done within a certain amount of time. Nevertheless without the proper technology this might be difficult. Therefore the in this paper proposed system, will remind the farmers when they should re-vaccinate their chickens to improve their survival rate. In the future such a system can be extended to also inform local farmers about diseases in the neighbourhood.

Unfortunately this system faces some difficulties. With the still developing technologies in Mali, the currently used technologies are still a bit deprecated. Without proper electricity and internet, the basic mobile phone is still the best option. With 0-30% of the people in Mali being illiterate, an information system which uses text as input and text as output, will not be very useful. In this system a information system with database running in the background is mainly focused on speech with basic DTMF. The voice system will use audio files to design an easy in use speech menu for the farmers in which they will be informed. The system is designed to lower the chicken death in Mali by implementation of the 'Vaccination des Poulets' system. In this paper the third iteration prototype is shown and described.

2. CONTEXTUAL ISSUES

2.1 Context and Scope

In Mali there are still a lot of people illiterate, in addition, due to the lack of internet and electricity, connection by a mobile phone, with only basic features, is mostly used. Due to this illiteracy, contact by text message (SMS) is not often used, information should therefore mostly be transferred by speech (these audio files are then manually written down by a person and stored in the database). In addition, the system should also give speech feedback as output. Therefore, a network connection in the environment and access to a mobile phone within the network of the farmers is required for this system to work.

Besides the farmers who can benefit from this VP service, there are a lot of people who can also have benefit from this cooperation between the farmers and VP. When farmers vaccinate their chickens, and re-vaccinate them within the amount of time, the chance that their chickens will become ill and die becomes smaller. In addition, when farmers in the future would receive a notification about diseased chicken outbreaks, farmers might let their chickens stay inside, and undertake actions to make sure their chickens stay healthy. This will lower the death rate of chickens which is not only beneficial for the farmer, but also for the inhabitants of the village. When there are more chickens, there is more food for everyone, the prices can stay low and farmers might have a good negotiation position with other villages because they sell their chickens for a lower price. Even though food security for the inhabitants, cost price and other economical aspects are interesting, for now we will only focus on the basic features of the system and its interaction with the farmers, and the sustainability of the system. When the death rate of chickens is lowered after implementation of the system, the systems performance can be seen as an success.

2.2 Use Case Scenario Script

Yaro is a farmer in Mali, he has just started to take over his father's company, even though he already knows a lot about farming he still can use some support. Yaro has just started to use VP (Vaccination des Poulets) which is locally used by many farmers. VP knows a lot about Yaro, he knows the amount of chicken batches he has, when the chickens are born and Yaro receives a notification when his chickens should get what kind of (re-)vaccinations. With VP in Yaro's life he feels more secure about the health of his chickens. Yaro also receives a phone call when his chickens should be vaccinated two days before every vaccination date. Yaro really likes that VP works with speech, because he has never learned how to read or write, and now he is still able to make use of the system and to understand it.

2.3 Interaction and Communication

The system and the farmer have a lot of communication and interaction with each other. First of all the farmer needs to registrate. This is done by speech input which is saved as a .wav. This audiofile then is listened to by a person working at the server (which can write and read) and this person will add the name of the farmer to the database. If the farmer is already registered, he can enter the amount of newborn chickens. These chickens will be saved in the database with the vaccination date. The farmer will receive a reminder after an X-amount of days by the VP-server. For now this reminder will be done by a phone call by the same person working at the server (because the current hardware of the server cannot call back) nevertheless in the future we want our server to be as automatic as possible.

3. THEORETICAL BACKGROUND

Mali is one of the poorest countries in the world and that is why Mali is dependent on foreign aid. The current population of Mali is 19 million [14]. The average salary of a (experienced) worker is $1,500 per year, but half of the population of Mali lives with a day spending of $1,25 per day. The amount that half of the population of Mali have is below the international poverty line [12].
According to statistics of Unicef 89.5 persons per 100 have a mobile phone [13]. These phones are not per se smartphones, these are probably all mobile phones without internet access. Average cost for a one minute call is $0,26 in Mali [16].The number of internet users in Mali is very low: 1.9 million people have internet access, that was 11.1% of the total population in 2016 [15]. The average monthly costs for internet is $17.96, which is expensive if you compare that to the amount they have to spend on daily basis [16].

4. RELATED LITERATURE

4.1 Identifying rale sounds in chickens using audio signals for early disease detection in poultry

The paper of M. Rizwan et al. [17] shows how audio data can be used to detect diseases in poultry. By making use of Extreme learning machine (ELM) and support vector machine (SVM). The ELM is a feed forward neural network with multiple layers of (randomly assigned) hidden nodes. The SVM is used to separate classes in data, which helps to map data. By analyzing audio data with complex algorithms they made is possible to detect diseases. They made use of two groups (control & experiment) of six newborn chicks and where placed in two Horsfall isolators. The isolators were equipped with microphones, cameras, and weatherboards. They recorded audio for 25 days long and after around 20 days they detected symptoms of infectious bronchitis virus in the experiment group. The control group did not expose anything during the experiment.

If the developers can make a mobile application of the system, then it will be accessible for farmers around the world and then it can also be used in ICT4D to help diagnose chicken disease with cheap equipment. The concept can also expand our solution, where the farmers detect diseases by themself with an application (the application could be hosted and maintained by us as the service provider because of its complexity). This disease they could then mention to the server, which will store this disease in the database. This would be a nice way to locally check for any rising epidemics and also warn local farmers for certain disease outbreaks.

4.2 Development of a Mobile Image-Based Reminder Application to Support Tuberculosis Treatment in Africa

The paper of H. Ali Haji [18] introduces the design of an application to encourage patients to follow their treatment for tuberculosis (TB) from Zanzibar and Tanzania. Thirty-eight participants were given mobile phones with the application installed to test the application and to evaluate their experience. The developers used case studies to understand the challenges that were faced during treating TB. The mHealth image-based system that was made had two advantages. Firstly, it does not have literacy and language barriers and secondly it does not require any party to host and control the service that is provided with the created system. To adhere patients with their treatment the system helped as expected. The system helped literate and non-literate patients with TB. The system even worked properly in remote rural areas, where the mobile network was not working as it should. This last thing, that the system can also work when the mobile network is not well, should also be implemented in our system in the future by others to make the system more sustainable.

4.3 Characteristics of commercial and traditional village poultry farming in Mali with a focus on practices influencing the risk of transmission of avian influenza and Newcastle disease

The paper of S. Molia et al. [19] aims to show characteristics of commercial and traditional village poultry farming in Mali, to influence the risk that avian influenza and Newcastle disease will be spread. By making use of surveys in 2009 until 2011 in a specific area data was collected of 98% of the Malian poultry population. The study area consisted of five ot the eight regions in Mali. The five areas are called; Kayes, Koulikoro, Sikasso, Segou and Mopti. The conclusion was that some biosecurity and health measures are very bad implemented and could be improved without any big investments. Improving biosecurity and health is likely to happen in Mali because of education. Some villages made use of programs like PDAM, where farmers were educated with new knowledge to improve their biosecurity. This is also a lesson that should be learned to implement our solution, to make sure that the farmers are well educated and understand the service that we provide with our application. We as the service provider need to listen carefully to the needs of the farmers with implementing the service to make it useable for them without big investments.

5. SOLUTION DESIGN FOR VACCINATION DES POULETS 3.0

In the previous assignment, three questions were proposed to the end users. These three question were:

1. Are vaccinated chicks kept seperate per batch?
    a. Meaning that if on date X, I would vaccinate 10 chicks, and on date Y, I would vaccinate 10 other chicks, are those chicks kept seperate per batch and from the group?
2. How often does a farmer receive new chicks?
    a. Are these born on the farm or bought on the local market?

3. Would counting the diseased chickens on one day, on a certain time, be a good representation of the diseased chicken population of a farmer?
    a. Should we also take into account dead chickens for the 'diseased chicken' database?

These questions were formulated with the previous system design in mind. They were relevant for us to design the database for newborn chickens and diseased chickens while the idea was to implement two kinds of services for the VP system. One is to send out alerts for chicken disease breakouts in a certain local area, and the other one is to send the farmers notifications to remind them to vaccinate their chickens on time. However, after feedback from the contact person in Mali who works a lot with chicken farmers there, we have decided to narrow down our scope to just one main service for now, which is only the chicken vaccination reminder service (see fig. 5.1). This is due to the feedback about local farmers who are not able to diagnose certain chicken diseases while they do not have the right knowledge for that. Our contact person told us to design the system in such a way that it can help farmers diagnose chicken diseases and tell them what to do afterwards. Certainly, this seems feasible for the VP system as well, however because of our time scope for this project, it is not possible for us to include this implementation as well. Therefore, reporting chicken diseases is not an option anymore, and hence the whole disease breakout alert service was excluded from our VP system. Thus, for the current implementation, the focus is on a reminder service for chicken vaccinations only.

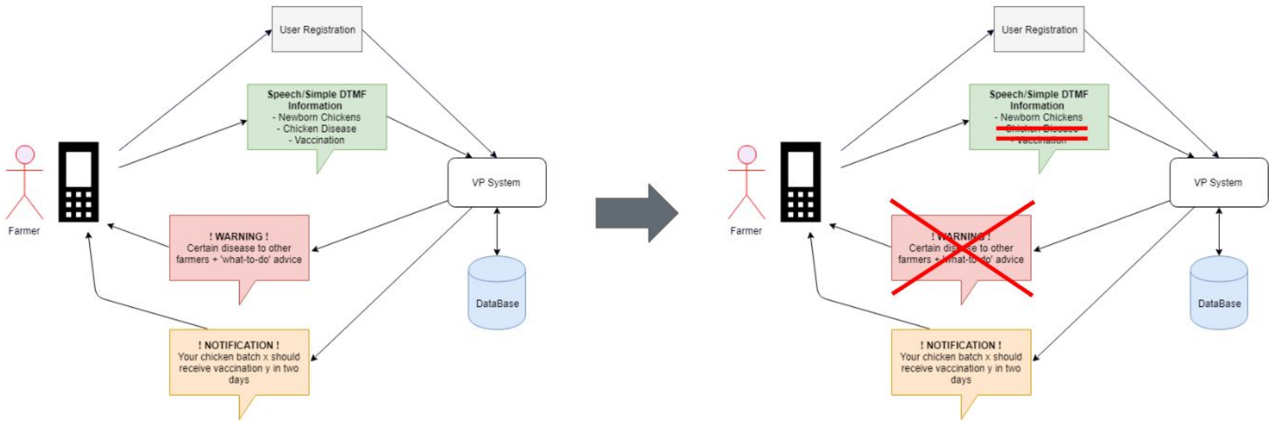

Figure 5.1: Modifications of the VP System Design

5.1 System design

The current VP system is designed with the functionalities to register new users and to save users' reports of newborn chicken batches in a database in order for the system to create a vaccination schedule for each registered user. An additional functionality to the system would be a notification mechanism that makes use of the vaccination schedules to send reminders back to the users.

For the purpose of scheduling reminders for vaccinations, it is not crucial to store the amount of newborn chickens. It is the simplest for the system's database to only store the day when an entire chicken batch was born and from there on use that information to calculate the time for the reminders that is corresponding to the actual vaccination calendar relevant for our use case (see Appendix A). Therefore the system will not ask the user to report the amount of newborns anymore. Next, it might be unnecessary for the farmers to call and to report every time a chicken batch got vaccinated as well, as it is expensive for the farmers to call that often. Moreover, it is almost certain that farmers will vaccinate their chickens after they have received the reminder to do so. Therefore, to keep the reminder schedule simple, the option to report vaccinations is also left out of the system and our VP system now needs to assume that chicken vaccinations indeed happen after notifications in order to adhere to the scheduled reminders.

In Figure 5.2, the interaction design between the users and the system is presented in an action diagram. There are two kinds of users: (1) the farmers, and (2) the admin person(s). The farmers can interact with the VP voice server by calling with a GSM phone. If the farmers are calling for the first time, the VP system will ask them for their language preference. For now, the choices are between English and French, however another local language option can always be added later. After language preference is set, the farmers are then redirected to user registration where they can register as a new user with a recording of their first and last names. Subsequently, when the registration is done, the farmers can enter the voice service where they first will be ask whether they would like to register a new batch of chicks. When they are choosing to do so, they will be asked how many days ago this chicken batch was born. After the system receives this input it will submit and save it in the database. The farmers then reach the end of the voice service where the system tells them what to do immediately and when they can expect a reminder for the next vaccination date. Figure 5.2 also shows what information is stored in the database. Further explanation about the VP database is elaborated in section 7 of this paper. The other users of the VP server are the admin persons. These people are needed to manually adding the recorded user names as text in the user database. As for now, because of our limited Kasadaka system that does not have an option for outbound calls yet, we imagine the admin people to also call the farmers two days before the scheduled vaccination date to remind them what chicken batch they need to vaccinate in two days, and what kind of vaccination is needed.

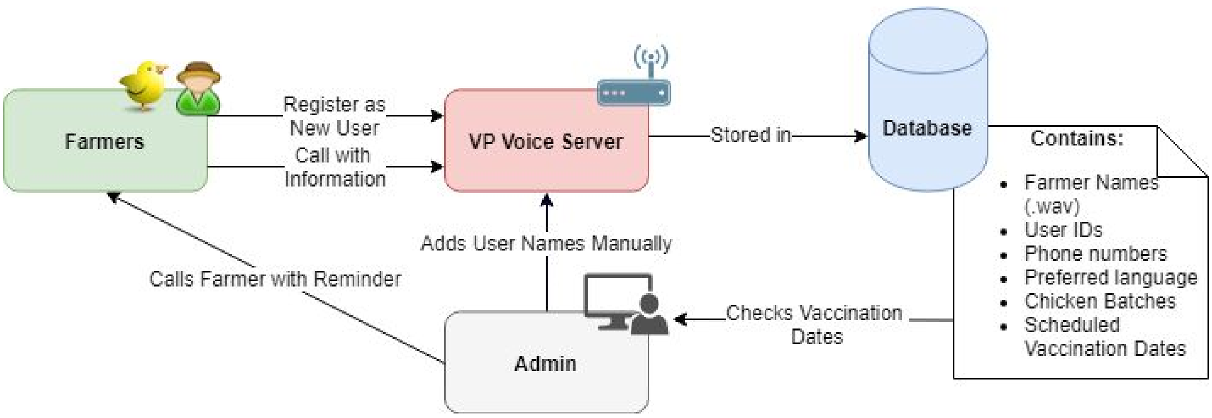

Figure 5.2: Data Model with Interaction Features

6. INFRASTRUCTURE ISSUES

For the system's infrastructure we made use of the Voice Service Development Kit (VSDK) which is based on the sustainable voice-service platform called KasaDaka and is built using the Django web-framework. The KasaDaka platform is used to host a voice-service locally and is runned on a Raspberry Pi with a GSM dongle that connects to a locally available GSM network. The VSDK is made as an extension of the KasaDaka platform. To test our system online, we make use of Heroku. Heroku is a cloud application platform to run the application online. The advantage of using KasaDaka instead of Voxeo is that KasaDaka can be hosted locally. Unfortunately, there are no text-to-speech and speech-to-text converters in VSDK. Therefore, there should be someone that enters all the recorded account names that is linked to the caller-id, as mentioned before for the system design.

In Appendix B, we made a simplified representation of how our current infrastructure of our system works. Using Django[9] to build our KasaDaka-VSDK platform and the structure of our call flow, and using Heroku[10] to upload this platform to a local server where it can be called by the users. Django can be locally used while designing the platform, nevertheless Heroku is needed to upload the platform and make the .wav's and call flow available for the public. Unfortunately, Heroku cannot host static files such as our audio files for the voice fragments, therefore a SFTP server is used for access to these .wav files. When the KasaDaka-VSDK platform is uploaded, the users can call the server and make use of our system's voice service.

6.1 Challenges

In our current implementation, the system can register a new user according to their registration name in a .wav file. Nevertheless, the support of a person working at the server is needed to manually enter and transform this name from speech to text and fill it into the belonging data in the database. As for now, there is a locally hosted SQLite database in Django and a PostgreSQL database when using Heroku. The VSDK is extended with custom made models, views, templates and admin elements, on the simplified system design. Therefore we now can store the given user input with the belonging context. Our current application also saves the caller_id (telephone number), the preferred language_id, date of the newborn chicken batch and the corresponded vaccination reminder dates. To show a functional prototype, we have adapted our previous VoiceXML template to the current available structures and modified it again by simplify it after the potential users' feedback. For the user registration process, there is no speech to text and text to speech conversions, therefore to register a new user by their names, the user should say their names out loud and this recording (.wav) can then be stored in the database. Later, the recorded account names that is linked to the id-number of the phone should be entered in the database manually by a person.

Another adaptation that we have to make due to the current infrastructure in our previous system, is that we have to put the entire application in one voice service as we only have one phone number which the user can use to reach our application. In real life, it is more reasonable to use multiple voice services with different phone numbers linked to these services. In our previous designed system, this could mean a different service to report newborns, a different service to report diseases, and a different service to report a vaccination in the future. However, because of the mentioned adjustments to the system, our current application now only needs one service to let farmers report new chicken batches. Therefore one service is not a limitation anymore for our current system. Nevertheless, if we would extend our system in the future, for example with diagnosing chicken diseases, using multiple services could save the user time while they do not have to loop through the entire menu to report different things. In that case different services can be set up for different report purposes.

7. IMPLEMENTATION

A database (see fig. 7.1) was used to store the system's information. The database contained the following application-relevant tables:

User: Stores the individual users and their basic information, being phone number, first and last name, chosen language and a link to SpokenUserInput

Session: The users' sessions in the system, storing when the sessions started and ended, the caller_id (which is the phone number), the session's language (derived from the user and/or the service), the service being used and the user who's session it is.

Service: This is the service the user is making use of, this contains a name, a description, when it was created and modified, whether it is active and whether this service requires name and language registration.

Language: This contains the languages the system is compatible with, consisting of a name (eg. English) and a code (eg. en) and various constants such as numbers.

SpokenUserInput: This contains recordings made by users, consisting of the audio file (.wav), the time of recording, a reference to the session in which it was recorded, a link to which category it falls in and a description.

UserInputCategory: This stores the various categories for SpokenUserInput, by name and description, linked to the service in which it is active

Disease: This stores the different diseases in the system, the name and description of the disease and whether is has vaccinations.

DiseaseVaccinationday: This stores how many days after birth a vaccination has to be given, and a reference to the disease for which it is relevant.

UserDisease: A ManyToMany table linking disease and user, storing detected diseases as well as the amount detected.

Batch: Contains the batches the referenced user has submitted to the system, and the date on which the batch was born.

Vaccination: This contains the vaccinations that have been scheduled in the system, referencing the relevant disease as well as the batch that is to be vaccinated and the data on which it was scheduled and when it was actually given.

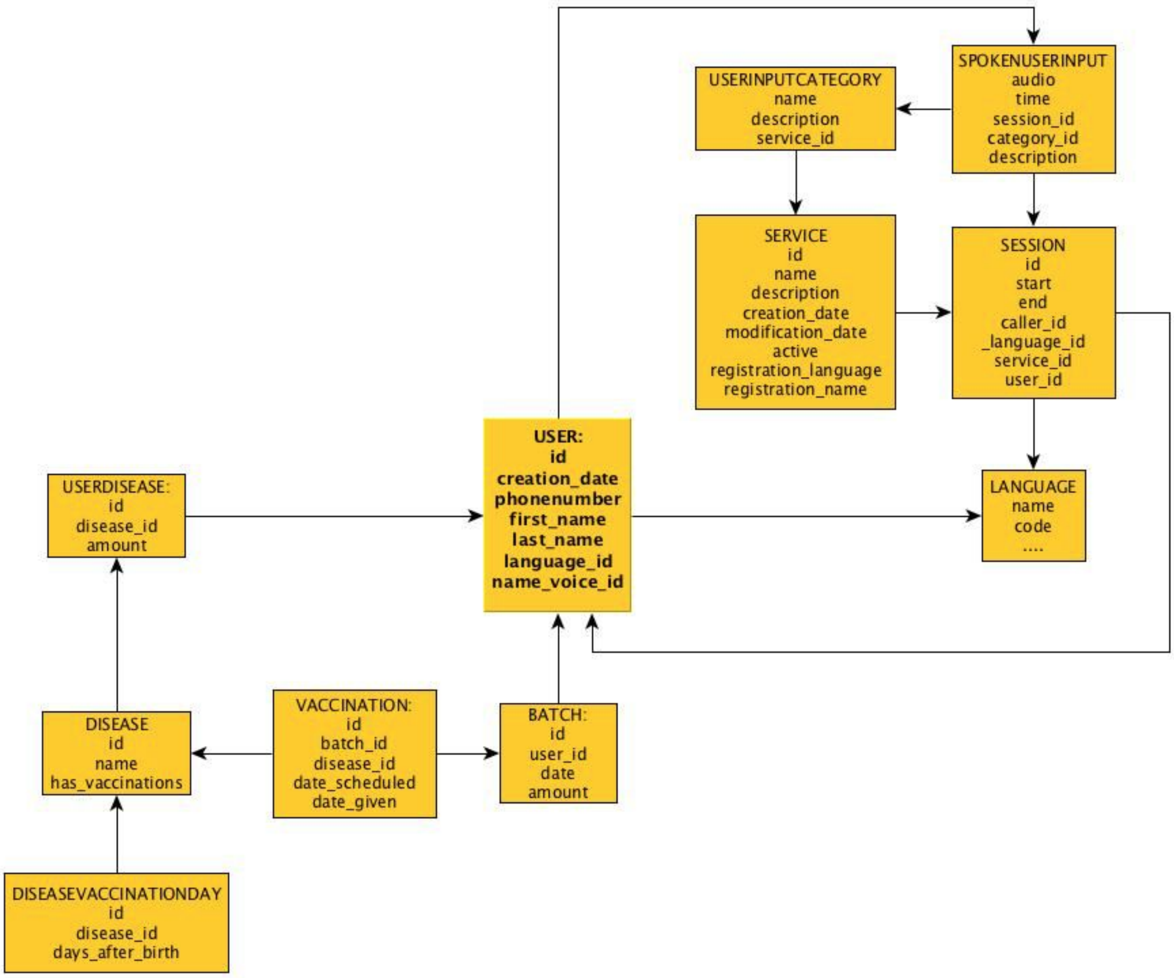

Figure 7.1: Visualisation of the VP Server Database

This structure was embedded in an aforementioned Django framework, facilitating interaction between the users and the stored information. Our current VP system implementation can be found through the following link to GitHub, in the 'test' branch: https://github.com/linhtr/KasaDaka-VSDK.

8. USAGE SCENARIO

To use the prototype follow these instructions:

- *Step 1*: Go to http://ict4d.kasadaka.com/
- *Step 2:* Enter the following information:
  - Your name: group 4
  - VoiceXMLURL: http://mysterious-sea-88468.herokuapp.com/vxml/start/2
  - Lock Password: ICT4D123!
  - URL to call on incoming bip: *keep this empty*
  - *Click*: "verzenden"
- *Step 3:* *Call* 020-3697664

Below in Figure 8.1, the call flow of the current VP system implementation is presented.

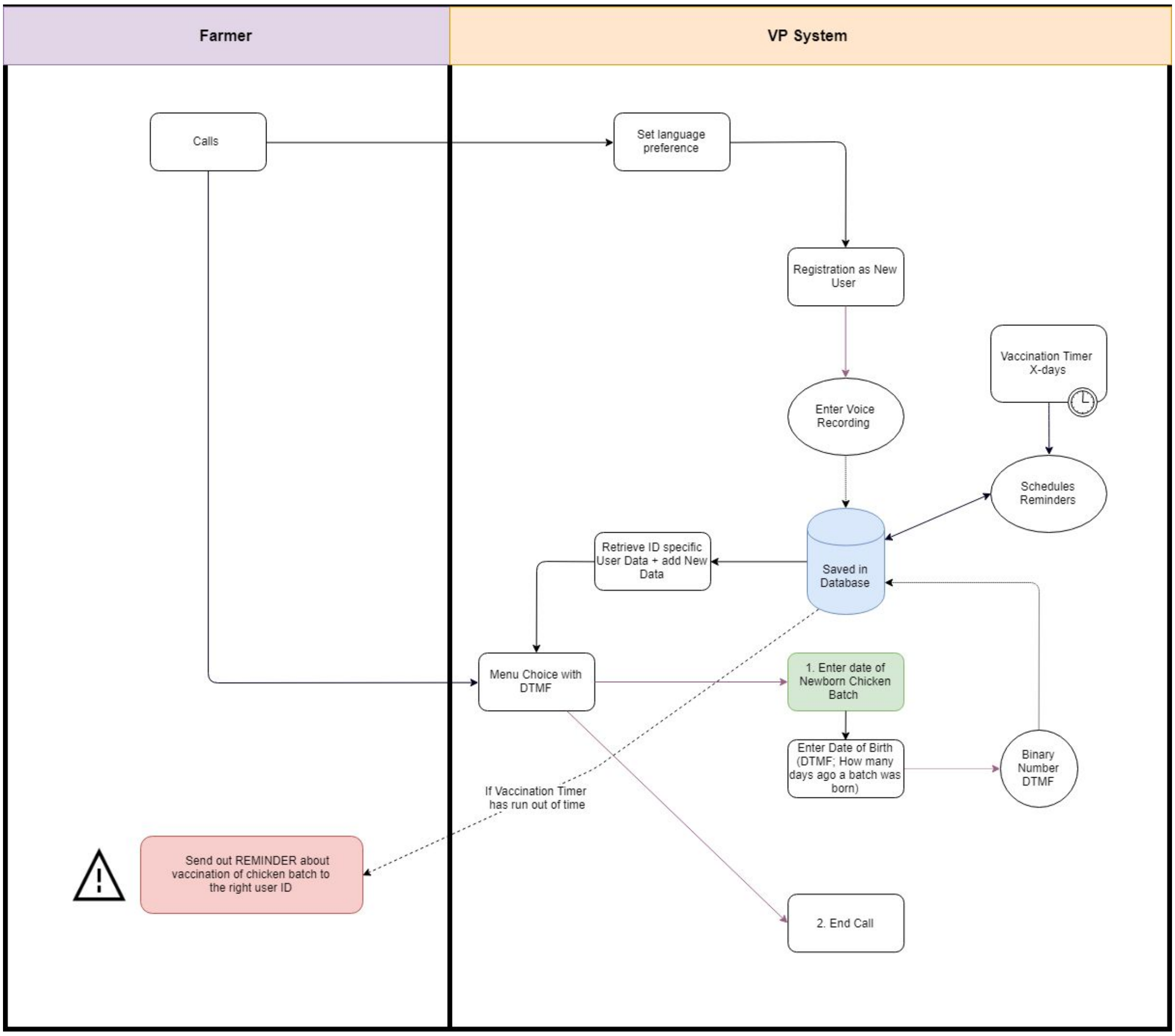

Figure 8.1: Call Flow of the current VP system implementation

9. SCOPE AND FIDELITY OF THE PROTOTYPE

While developing our Vaccination des Poulets prototype we tried to optimize our prototype with every iteration. The main problem after the first iterations was that the server was not available offline and that there were no multiple language options. These two characteristics we therefore implemented in our second prototype and iteration. Nevertheless, there was still one more problem, the lack of a database that combined all the aspects together. In this third iteration we implemented the database, however we also simplified our current system. Therefore, we do meet all the ‘must have’ and ‘should have’ requirements. Yet, we did not focus on the kind of diseases chickens have anymore (because farmers often do not know this themselves). We also made sure that the DTMF input still stayed simple (only simple number digit input) and that the system makes use of voice input whenever the input would be too difficult for the nonliterate user to enter in de system (in our case, this is for the registration with first and last names).

Furthermore, despite a structured database for our use case, our current system still depends on the interaction and support of a human. This human should listen to the saved .wav files to manually add the names of the farmer into the database. In addition, this person should check the database and call the farmer when it is time to re-vaccinate his or her chicken batch.

Moreover, the ‘could have’ we eventually did meet. For now our system can recognize and differentiate between different chicken batches. In addition when the farmer needs to know which kind of vaccination he or she should, this will also be mentioned in the notification. All this info is easy accessible by the person working at the service, in the database. through notification sending processes with daily checks once the system is populated by users.

Lastly, we almost meet our ‘won’t have’. The now designed database is very structured, therefore if the farmer would call the person at the server and would ask some personal information he or she could provide this to him or her based on the designed database. For this option, we could see a possibility in our voice service menu to go to batch submission or to end the call. Instead of choosing 2 to end the call, the user could be redirected to call an actual person at the server who can give the requested information to the farmer.

Below, an overview of the must haves, should haves, could have, and won’t have for our current VP system is presented, where [x] means aspect is present, and
[~] means aspect is half presented, however some extra implementations could be made.

Must Have:

- Speech Input [x]
- Speech Output [x]
- Database [x]
- ~~Notification system of diseases [ ]~~
- Notification system of vaccination [x]
- Input of newborn chicken batch [x]
- Available Offline [x]

Should Have:

- Easy input for farmers (if dates are difficult or numbers) [x]
- Different language options [x]

Could Have:

- Alert for different batches of newborn chicks [x]

Won’t Have:

- Database access that is easy available for the farmers to retrieve information about their chicken batches [~]

10. BUSINESS MODEL

In Figure 10.1 a visualisation of the developed business model is shown. In this business model a sustainable process is shown in which farmers can store their information about their chickens in a system and in addition receive phone calls about their chickens revaccination. In return the VP service will receive service fee and information from the farmer and will receive a calling service from the telephone company and hosting service from the hosting provider. The telephone company and the hosting provider, will both receive payment for providing calls and the hosting service. The dotted lines represent money flow and the blue lines data flow. This small subprocces on it’s own could be sustainable when the costs are covered by the farmers. Nevertheless in the beginning this probably will be quite difficult. Therefore we have set up a sustainability plan which can be found in section 12.

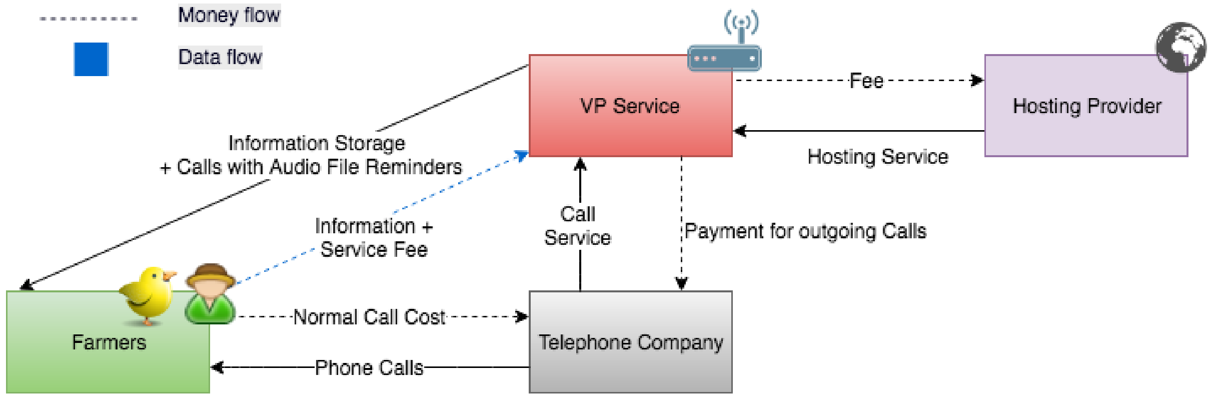

Figure 10.1 Visualisation of the Business Model

11. NET VALUE FLOW SHEETS

In this section we discuss the net value flow sheet of our designed server from three different perspectives. Namely: the Vaccination des Poulets server, the farmer and the telephone company. In the sustainability plan we have shown the hosting provider as an external company, nevertheless, depending on the certain characteristics of the telephone company, they could also provide this service. To make it easy, we for now have chosen to make the telephone company also the hosting provider in the following net value flow sheets.

11.1 VP Server

Below, the value flow sheet for the Vaccination des Poulets server is shown in Table 12.1. First, the VP server will need income to provide the VP service. When the VP server is called, the caller will pay 105 CFA per call. The reason for this value is that call costs are 100 CFA per minute [16], and when the service calls the farmer back; it will need this 100 CFA to call. In addition, the server will pay service cost to the provider, we are not sure how much this will cost, but we think that this will be a monthly fee of around 10.000 CFA (based on the unlimited data, Cable/ADSL of around 17.96 $ which is roughly 10.000 CFA a month [16]), this value is covered when 5 CFA is paid on top of the 100 CFA in general. We think that approximately 500 farmers will use the server [11] four times a month. This gives an occurance of 2000 per month of the service. The actual usage, the actual amount of users, the average usage under farmers and exact prices should of course should be considered more precisely in future work, nevertheless this is for now as close as we can get to an approximation without being their in Mali and asking providers personally.

Table 11.1: Net Value Flow Sheet of the VP Server

Interface	Port	Transfer	Occurrence	Valuation	Value	Total
Hosting Service Incomes: from Vaccination de Poulet						
	Out: Service	MONEY	Payment per Usage ~2000	105 CFA	210.000 CFA	210.000 CFA
	In: Money		(500 farmers calling			

			once a week)			
Phone Calls: From Server VP to Farmers (Fee)						
	Out: Audio Files	MONEY	Once per Incoming Call ~ 2000	- 100 CFA	- 200.000 CFA	- 200.000 CFA
	In: MONEY					
Service Costs						
	Out: MONEY	MONEY	Once per month	-10.000 CFA	-10.000 CFA	-10.000 CFA
	In: IP-connect ion					
TOTAL:						0

In Figure 11.1 below, a visualisation of the net value flow sheet is shown. In this visualisation we have three different subparts in which farmers interact with the service by a GSM network. Money flow from the farmers to the service is exchanged for a reminder service. The usage of the GSM network and the IP-network is exchanged for money by the server.

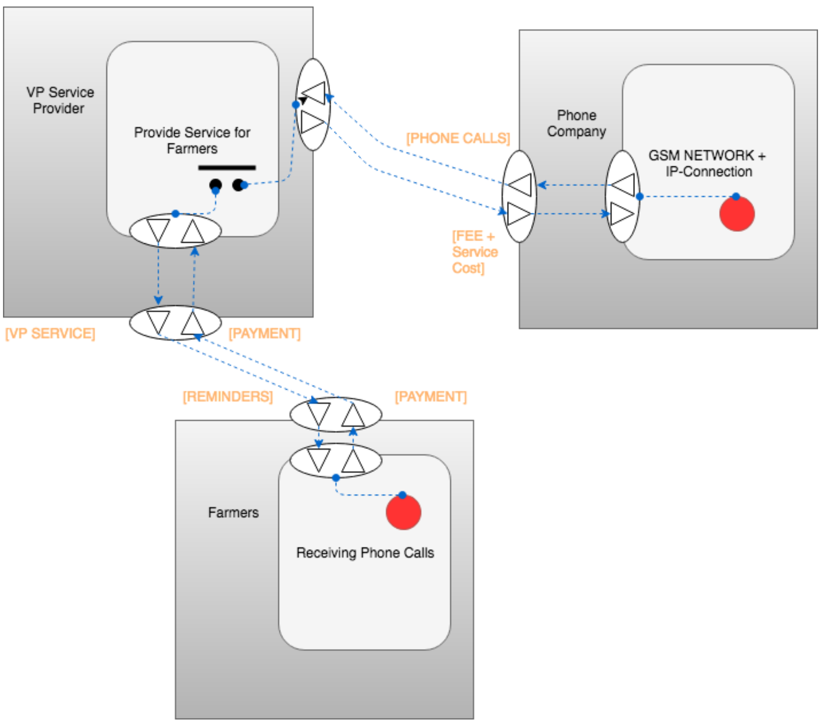

Figure 11.1: Net Value Flow Model

11.2 Farmer

In Table 11.2 a representation of the farmers costs are shown. The farmer will pay two types of costs, namely: fee (to call the server), and service costs (to use the server). The server cost will include all the costs to keep the service running, meaning money to call back the farmer and money to keep the IP-address existing. As can be seen in Figure 11.1 the farmer will receive reminders (as outbounding calls with audio files) for chicken vaccinations by the VP service, in exchange for money.

Table 11.2: Net Value Flow Sheet of the Farmer

Interface	Port	Transfer	Occurrence	Valuation	Value	Total
Service Costs: to Vaccination des Poulets Server						
	Out: MONEY	MONEY	Once per Call ~ 2000x	- 105 CFA	- 210.000 CFA	- 210.000 CFA
	In: Reminder service		(500 farmers calling once a week)			
Calling Costs: to telephone company						
	Out: MONEY	MONEY	Once per Call ~ 2000	- 100 CFA	- 200.000 CFA	- 200.000 CFA
	In: Audio files (reminder calls)		(500 farmers calling once a week)			
TOTAL:						- 410.000 CFA

11.3 Telephone Provider and Hosting Service

Below in Table 11.3, the net value flow sheet for the telephone provider is shown. The telephone provider will receive fee (shown in orange) and the hosting service, service costs (shown in blue). The fee will exist out of calling cost from the farmer to the server, and from the server calling the farmer (and playing an audiofile). The service cost will exist of a monthly pay to keep the IP-connection existent. In Figure 11.1, a visualisation of this process is shown. The phone company will receive fee and the hosting service the service cost. The telephone company gets money to provide a GSM network and the hosting service to provide a IP-connection and make the call service possible.

Table 11.3: Net Value Flow Sheet of the Telephone Provider

Interface	Port	Transfer	Occurrence	Valuation	Value	Total
Phone Calls: To Vaccination de Poulet Server (Fee)						
	Out: Call		(500 farmers calling once a week)			
	In: MONEY	MONEY	Once per Call ~ 2000	100 CFA	200.000 CFA	200.000 CFA
Phone Calls: To Farmers (Fee)						
	Out: Audio		Once per incoming			

	Files		Call			
	In: MONEY	MONEY	Once per incoming Call ~ 2000	100 CFA	200.000 CFA	200.000 CFA
Server Vaccination de Poulet						
	Out: IP-Connection		Once a month			
	In: MONEY	MONEY	Once a month	10.000 CFA	10.000 CFA	10.000 CFA
TOTAL:						410.000 CFA

12. SUSTAINABILITY PLAN

To deploy Vaccination des Poulets we have set up a sustainability plan. This plan is set up to make sure the service is maintained after implementation and after funding of sponsors. As can been seen in Figure 12.1 five different subparts will play a role in this process.
First of all, farmers will make use of the VP service and will receive phone calls with vaccination reminders, and information about their chickens that will be locally stored. In exchange, they will provide the server with information, pay service fee and call fee. The telephone company will exchange calls between the farmer and the VP service and the hosting service will maintain the IP-connection for the VP service. In exchange the telephone company and the hosting provider will be paid by VP and by the farmers. The VP server will be the central point between all the subparts, it will receive funding by sponsors (or the government) to set up the server and develop the server in exchange for data. As could be seen in the Net Value Flow Sheet farmers pay quite an amount of their income to keep the VP server running. Therefore there will be two phases in this process. In phase1, funding or subsidy is needed to make the system affordable in the first place and keep the prices within payable limits. While the popularity of the system grows and the usage with it, the hosting provider cost can be divided over more farmers and will drop, making maintenance without an external provider possible. In addition when the systems grows, a better negotiation position with the telephone company might ensure even lower prices for calling to the system and to the farmers. The moment the prices of the service cost and calling cost lower, we will work towards phase 2, in which is the system is completely sustainable without the outside influence of sponsors or the government, nevertheless, this may take a while.

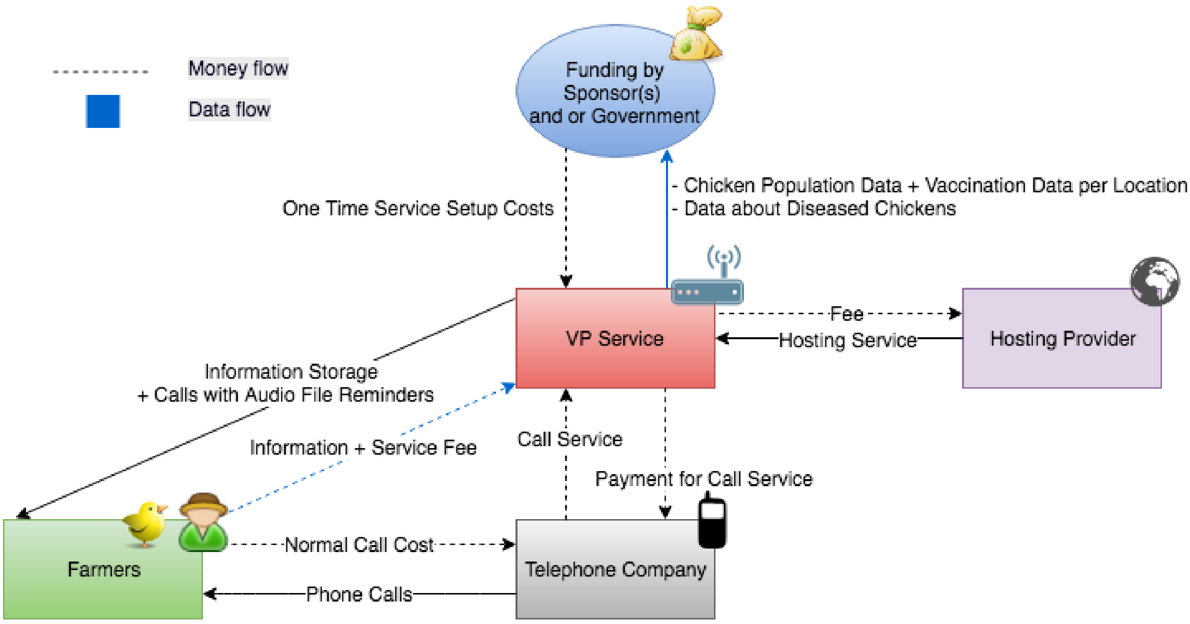

Figure 12.1: Business Model Sustainability

13. EVALUATION

Despite that our system design went from three kinds of services of newborn, disease, and vaccination reports to only one service to report newborn batches, in our opinion our system is now better structured due to the fact that we have a better overview and understanding of our current database. Also because the amount of data that needs to be stored in the database is reduced, we were able to simplify the system and make it more usable for future work. All the information that is needed for a notification system is basically there. Unfortunately we were not able to implement this ourselves yet, however we certainly see the feasible aspect of it and hope that with our current database design we have made a great setup for the notification functionality that checks for reminders daily to be easily implemented in the future. However, for now our VP service can still be used with a real human in between the the system and the end users who is seeing the database with the scheduled reminders as presented for instance in Figure 13.1.

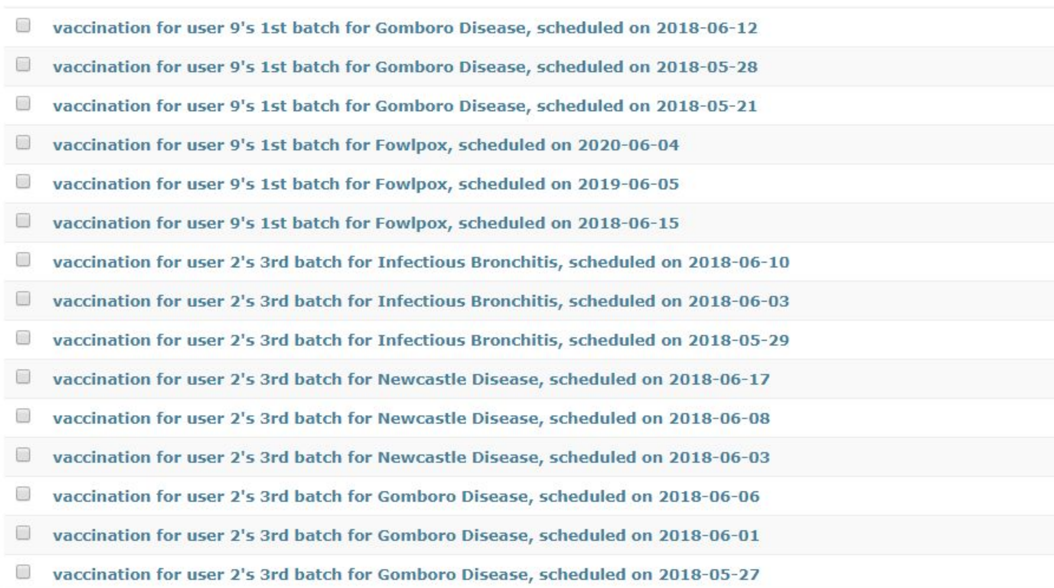
vaccination for user 9's 1st batch for Gomboro Disease, scheduled on 2018-06-12
vaccination for user 9's 1st batch for Gomboro Disease, scheduled on 2018-05-28
vaccination for user 9's 1st batch for Gomboro Disease, scheduled on 2018-05-21
vaccination for user 9's 1st batch for Fowlpox, scheduled on 2020-06-04
vaccination for user 9's 1st batch for Fowlpox, scheduled on 2019-06-05
vaccination for user 9's 1st batch for Fowlpox, scheduled on 2018-06-15
vaccination for user 2's 3rd batch for Infectious Bronchitis, scheduled on 2018-06-10
vaccination for user 2's 3rd batch for Infectious Bronchitis, scheduled on 2018-06-03
vaccination for user 2's 3rd batch for Infectious Bronchitis, scheduled on 2018-05-29
vaccination for user 2's 3rd batch for Newcastle Disease, scheduled on 2018-06-17
vaccination for user 2's 3rd batch for Newcastle Disease, scheduled on 2018-06-08
vaccination for user 2's 3rd batch for Newcastle Disease, scheduled on 2018-06-03
vaccination for user 2's 3rd batch for Gomboro Disease, scheduled on 2018-06-06
vaccination for user 2's 3rd batch for Gomboro Disease, scheduled on 2018-06-01
vaccination for user 2's 3rd batch for Gomboro Disease, scheduled on 2018-05-27

Figure 13.1: An example of the scheduled reminders from the VP database

14. DISCUSSION

While developing our server in three different phases we run into quite some different problems, some were easy to overcome and some took some time. In our first implementation of the prototype we developed a VoiceXML service which was online accessible. The service worked quite well, nevertheless it lacked the feature to be played in different languages and was not accessible offline. While working on the second prototype we tried to overcome these problems, making the voice server accessible offline and available in different languages. Nevertheless the offline server brought the problem that text to speech was not available anymore so that audio files where needed and in addition the whole voice server structure needed to be implemented over again. With the third implementation we received feedback from potential users in Mali. This gave use insights in different aspects of our system which we tried to implement. This made us take the decision to simplify our prototype relative to the first and second prototype. We now only where focusing on revaccination of newborn chickens. Nevertheless still the offline programm had limitations which we had to overcome. First of all outbounding calls with KasaDaka are not yet possible. We wanted to create .wav audio files which would be played once the KasaDaka performed an outbounding call. To create a temporary solution we will need a real person which will support our system manually. This person will call the farmers to remind them of the vaccines. This person has access to the user database, where he/she can see a table with the users and their chicken batches together with the scheduled vaccination program.

Also contextual issues are still a problem. Even though we tried to look up as much contextual information as possible and ask as many question as we could, sometimes a real world setting gives most valuable insights. For instance for future work and implementation we would still like to know who is providing the vaccinations to the farmers. Who is vaccinating the chickens and do the farmers have knowledge about vaccinations themselves?

15. CONCLUSION

The system VP that we provide for local farmers in Mali is to remind them to re-vaccinate their newborn chickens. This VP systems keeps in mind the context of these farmers, by taking into account their illiteracy and access to only simple technologies. The easy and user-friendly interface of the voice server makes it possible to be used by any farmer in Mali and can be accessed with the most simple and old mobile phone. Farmers will be more confident about the health of their chickens with the help of VP. Unfortunately, we did not have the chance to test the application with farmers in Mali. If we had the chance to do this in some regions in Mali, we could get more insights to improve the application even more based on feedback after usage by the farmers. If only the health rate of the chicks would go up and the death rate would go down, our system implementation could already be seen as an succes.

15. Technical Conclusion

The framework in which the system was built, is highly suitable for applications in this field, as it enables developers to build accessible and scalable solutions in a short amount of time, while still being very understandable and we consider it an essential tool in the repertoire of ICT4D researchers and developers.

Finally we think that the VP application is a valuable ICT4D project that can be expanded and/or customized in the future to be used in different contexts.

16. FUTURE WORK

When we would combine our VP system with a disease diagnosing system, we could update farmers about local diseases by radio. This would be a nice angle of approach for future work. Another group has already been working on a voice system in which farmers answer certain question and a disease diagnosis is given as output. When farmers know which diseases their chickens have, they can notify the new VP service and when an epidemic is taking place, the radio can broadcast this information and make local farmers aware of the dangers. In Figure 16. a representation of this process is shown. The orange block with the radio is added. As can be see they will broadcast information of local diseases for information and money.

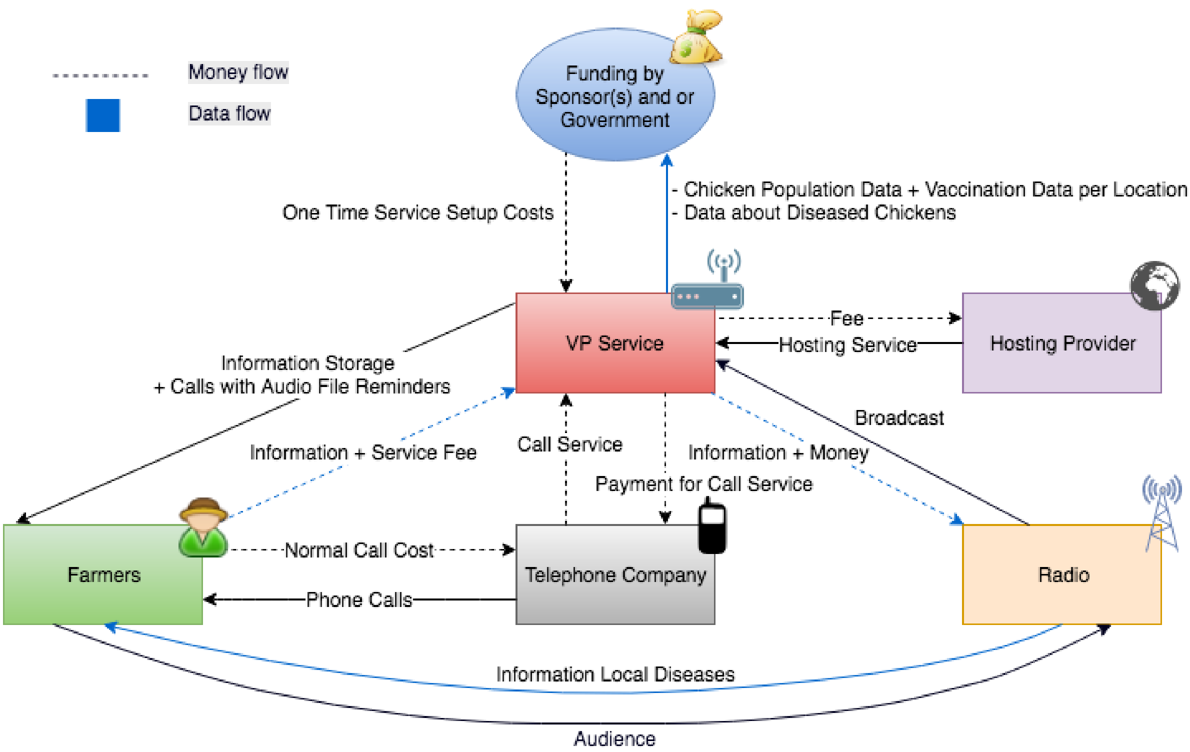

Figure 16. Future Work Broadcasting possibilities in the VP server.

In addition in the future we would like our system to have outgoing calls itself and play audio files automatically so that human interference is not needed in the system. Nevertheless this was for now to difficult with the implementation of the database.

16.1 Linked Data

While the world wide web is developing, we are trying to structure the web by creating databases that are as complete as possible. Adding new information to these databases, by including the data received data by the VP server, could be of valuable input. Keeping track of chicken diseases and chicken population in Mali might create insights in certain trends. These trends can then be stimulated if good and tried to be broken once not. In addition creating awareness about local farmer problems in these areas might attract potential sponsors.

REFERENCES

[1] Pattison M. (2008). – Poultry diseases, 6th Ed. Elsevier, Edinburgh.

[2] Farm Radio (accessed 17-04-18) http://scripts.farmradio.fm/radio-resource-packs/103-2/4-nomadic-people-fight-fowl-pox-in-mali/

[3] Nelson, C. B., B. S. Pomeroy, K. Schrall, W. E. Park, and R. J. Lindeman.1962. An outbreak of conjunctivitis due to Newcastle disease virus (NDV) occurring in poultry workers. Am. J. Public Health 42:672.

[4] MACPHERSON, L.W. (1956) Some observations on the epizootiology of Newcastle disease. Canadian Journal of Comparative Medicine, 20: 155–168.

[5] Butcher GD, Rossi F. Prevention and control of fowl pox in backyard chicken flocks [Fact Sheet VM65]. Gainesville: Florida Cooperative Extension Service, Institute of Food and Agricultural Sciences, University of Florida; 1990.

[6] Luque D, Saugar I, Rejas MT, Carrascosa JL, Rodriguez JF, Caston JR: Infectious Bursal disease virus: ribonucleoprotein complexes of a double-stranded RNA virus. J Mol Biol 2009, 386:891-901.

[7] Veits, Jutta, et al. "Newcastle disease virus expressing H5 hemagglutinin gene protects chickens against Newcastle disease and avian influenza." Proceedings of the National Academy of Sciences 103.21 (2006): 8197-8202.

[8] Daily Nation (accessed 17-04-18) https://www.nation.co.ke/business/seedsofgold/Different-poultry-breeds-require-different-vaccines/2301238-3181538-4lc8hoz/index.html

[9] Django (accessed 03-05-18) https://www.djangoproject.com/

[10] Heroku (accessed 03-05-18) https://signup.heroku.com/

[11] https://ag-transition.org/3703/a-small-village-in-mali-is-farming-toward-a-viable-food-future/

[12] http://www.ophi.org.uk/wp-content/uploads/Mali.pdf

[13] https://www.unicef.org/infobycountry/mali_statistics.html

[14] http://www.worldometers.info/world-population/mali-population/

[15] https://www.indexmundi.com/mali/internet_users.html

[16] https://www.numbeo.com/cost-of-living/country_result.jsp?country=Mali

[17] Rizwan et al. “Identifying rale sounds in chickens using audio signals for early disease detection in poultry.” Signal and Information Processing (2016).

[18] Ali Haji et al. “Development of a mobile image-based reminder application to support tuberculosis treatment in Africa.” International Journal of Biomedical and Biological Engineering (2015) Vol:9, No:8.

[19] Molia et al. “Characteristics of commercial and traditional village poultry farming in Mali with a focus on practices influencing the risk of transmission of avian influenza and Newcastle disease”

APPENDIX A: An Example View of the Chicken Vaccination Calendar

	A	B	C	D	E
1	PROPHYLAXIS PLAN FOR PRIORITY DISEASES OF THE CHICKEN				
2	Period (days)	NEWCASTLE disease	GOMBORO disease	Infectious bronchitis	Fowlpox
3	1	First vaccination between the 1st and the 4th day (not beyond the 15th day) with PESTOS (bottle of 500 doses, 1000 doses for beverage and soaking)			
4	2				
5	3				
6	4				
7	5		First vaccination on the 5th day with GOMBOPEST		
8	6				
9	7			First vaccination on the 7th day with H120 (bottle of 500 doses, 1000 doses for beverage and soaking)	
10	8	1st reminder a week after first vaccination with ITANEW (bottle of 100 doses)			
11	9				
12	10				
13	11				
14	12		1st reminder with GOMBOPEST 7 days after first vaccination		
15	13				
16	14				
17	15				
18	16				
19	17				
20	18				
21	19				
22	20				
23	21				
24	22			1st reminder 15 days after the first vaccination, that is to say the 22nd day with H120	

APPENDIX B: Infrastructure of Vaccination des Poulets