Vendu!

Assignment 3 - Final report

By:
Christian Valladares (2612946)
Maaike Maas (2618179)
Sanjay Sheombar (2589918)

Table of contents

Introduction

We would like to aid in spontaneous regreening management practices in the sahel region through incentivizing the sales of non-timber tree goods. We base our approach on the Radio Marche use case implemented in Mali. We propose to introduce supply and demand driven market prices through the use of an accessible, and localized auction system.

Farmers can have difficulties with finding a consumer market for their products. The goal of the project is to help farmers find such a market and receive higher margins for their products as an incentive for producing non-timber tree products. Additionally, the project aims to address the economic sustainability of the project by diverting some of the profits to the upkeep costs of the project.

The name for our application is “Vendu!” which means “Sold!”. The use case we selected for this project is “market prices”. Further iterations of the project can expand to a more general marketplace with more product categories and involve more stakeholders.

1. The Vendu! auction application

1.1. Key idea

Selling your products for the right price can be difficult. With Vendu! we will let the market decide the right price for the producers of goods in rural Ghana. Vendu! grants producers/farmers access to a digital and voice controlled auction system. Producers/farmers become sellers and enjoy the advantages of an auction system. They can set their base price, and the quantity of the product they are selling and the system will schedule an auction for their products.

On the scheduled time the auction opens and customers can bid on the products in the Vendu! System via phone or computer. Naturally the auction is only active for a predetermined amount of time (for instance 15 minutes). When it closes the products are sold to the highest bidder(s). An auctioning schedule is available via phone or computer. Also, radio stations will be informed and asked to transmit the schedule over the radio (for marketing purposes).

When the auction opens customers can place bids on the product. They can decide on the quantity of products they want to place a bid for. With the calling system, customers who have been outbid will receive a text or voice message that they have been outbid. The message will contain the price of the new bid. To provide auction-like behavior the customer can reply on the message (either through voice or text if possible) to up their bid again. The web interface will show the current offered bid and customers can place a new bid by clicking on a button.

Because of supply and demand a producer can expect a higher income on his goods. The auction system can be funded (or even yield profit) by taking a percentage of the profit. We are considering different options to monetize the project in this document.

1.2. Actors and goals

Actor	Role/ Responsibilities	Goals
Seller	Producer of goods	To increase sales
Radio Station	Broadcast auctions	Increment revenue through a percentage of profits earned in the auctions
Customer	Buyer of goods	Acquire goods against a fair market price and to be informed of when a good is being sold
Telephone Service Company	Provides the supporting infrastructure for Vendu	Increment revenue by expanding their clientele

Table 1. Overview actors with roles and goals

Table 1 above shows the actors and their roles and goals. For extended documentation of stakeholders and concerns we refer to appendix A.3.

1.3. System Design

Figure 1 shows the interactions that take place with the actors and the system. There are mobile and web interfaces available. Appendix A.5 and A.6 contains flow diagrams to explain the interaction flows.

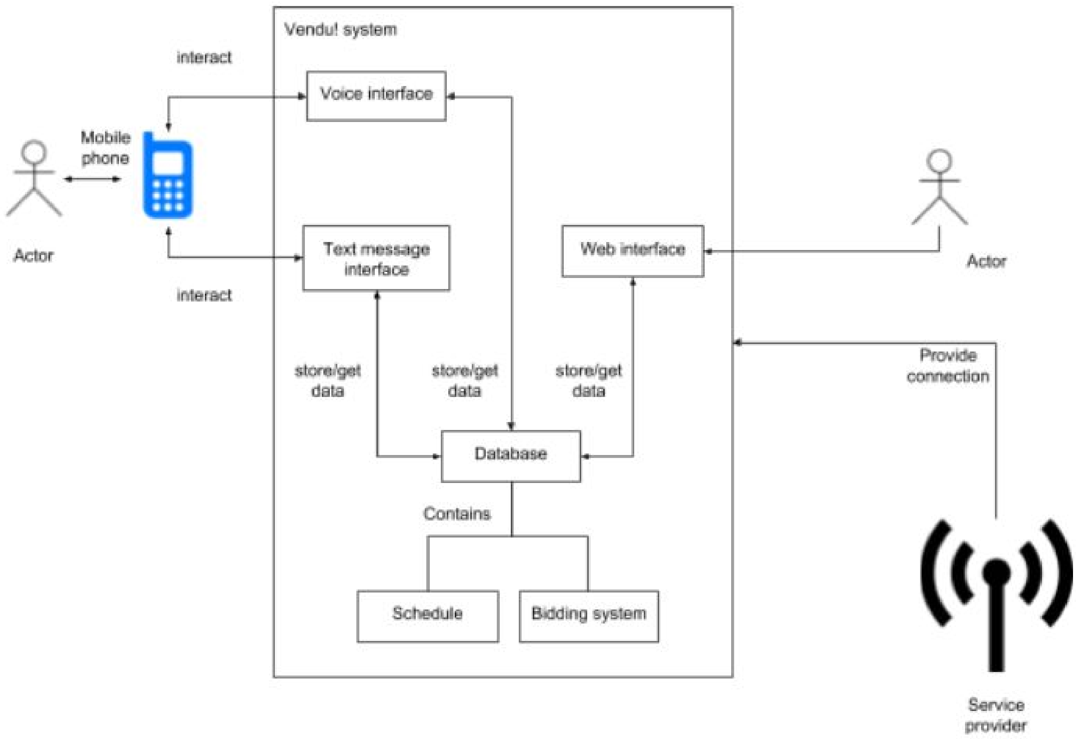


Figure 1. Vendu! System interactions

1.4. Key requirements

Below, the functional (F) and non-functional (N) requirements are described with the MoSCoW method.

Functional Requirements (Must have)	
Code	**Description**
FM1	The seller must be able to call a number to submit his products to the auction.
FM2	The seller must be able to choose the product, enter minimum price and quantity of the sale when submitting his products to the auction.
FM3	People must be able to call a number to get the auction schedule of today.
FM4	The consumer must be able to place his bid with a standard amount of money by calling a number.
FM5	The consumer must be able to select the quantity he wants to bid for.
FM6	The consumer must be notified when he has been outbid.
FM7	The system should be able to communicate the outcome of the auction and provide contact information for the delivery and payment.
FM8	Auctions, bids and contact information has to be stored (in a db).
FM9	The system must be able to communicate the new offer via text or voice message.

Non-functional Requirements (Must have)	
Code	**Description**
NM1	The system must be navigated with a voice and interface menu.
NM3	The system must be 90% available and reachable during an auction.
NM4	The system must be fast enough to inform users of a new bid in less than 2 seconds.
NM5	The hardware must be cheap.

Functional Requirements (Should have)	
Code	**Description**
FS1	The consumer should be able to adjust the amount of money of the new bid before it is placed.
FS2	The system should -besides English- also have the possibility to choose French as a language.
FS3	People should be able to follow the auction and place bids via a Web interface.

FS4	The seller should be able to go to a website to submit his products to the auction.
FS5	People should be able to get the auction schedule of today via a web interface.

Non-functional Requirements (Should have)	
Code	**Description**
NS1	The system should be communicating in two other local languages.
NS2	The system should be 95% available and reachable during an auction.
NS3	The system should be safe for the most basic hacking activities.

Functional Requirements (Could have)	
Code	**Description**
FC1	People must be able to call a number to get the auction schedule of today, tomorrow and the day after tomorrow.
FC2	People can view the auction schedule via a text message after they text a number.
FC3	The system could have more than two selection languages.

Non-functional Requirements (Could have)	
Code	**Description**
NC1	The system should be safe for normal hacking activities.

Functional Requirements (Won’t have)	
Code	**Description**
FW1	The system won’t have the delivery system.
FW2	The system won’t be able to handle payments.

2. Business aspects

2.1. Context and potential of the idea

To analyze the potential of the idea, we made a SWOT analysis which is shown in table 2. Below table 2 some aspects of the SWOT analysis are explained in more detail.

Strengths	Weaknesses
+ The app is easily extendable for languages + The app is available via phone and the web which reaches a bigger audience	- Further development of the app is not guaranteed
Opportunities	**Threats**
+ There is demand for an product selling app + There is not much competition + 56% of work in Ghana is found in agriculture + High estimated economical growth in Ghana	- Not everyone is familiar with the concept of an auction - There is a risk that the auction concept is not well received - Internet is not available anywhere - There are a lot of different languages in Ghana

Table 2. SWOT analysis Vendu!

Languages
We want to start our project in Ghana which is mostly English-speaking. Therefore, we focused our app on the English language. However, it is quite easy to extend the app with another language in the future.

As can be seen in figure 2, most countries surrounding Ghana are French-speaking so this will probably be the next language extension.

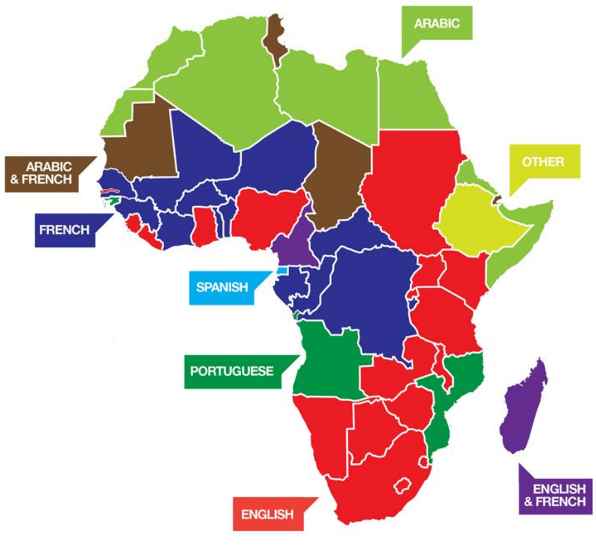


Figure 2. Business languages in Africa[1]

Demographics and economics
The currency of Ghana is Cedi. This could be extended with other currencies in the future. A report from PWC from 2014 (PWC Ghana, 2014) shows some interesting facts. The population is around 25 million people (estimated in 2013). Around 60% of the people falls between 15 and 65 years of which 71% was economically active. In 2014, 56% of the work was in agriculture as opposed to 29% in services and 15% in industry. This is favorable for our app since this means a lot of products that could potentially be sold with our auction app.

Ghana is interesting as an investment country as well. According to the World Bank, the African Development Bank, the International Monetary Fund an the Brookings Institution, Ghana is likely one

[1] http://www.corporatecouncilonafrica.com/business_languages_in_Africa, retrieved on 19-05-18

of the world's fastest-growing economies in 2018 (McDonell, 2018). Its growth in 2018 is estimated between 8.3 and 8.9 percent. Economical growth could stimulate the selling markets which is good for the spread of the app.

Consumer market for the app

From our user interviews in class we learned that auctions are not known in the rural areas because they don't use the system there. However, people in the city are familiar with the concept of an auction or have used it before. For rural Ghana, we therefore need to market the idea very clearly. We also learned that there would probably be interest for our idea among farmers because they sometimes have trouble selling their products. Every new opportunity to sell something is welcomed. In 2001, Henrich et al. concluded that people in small-scale societies with a higher degree of market integration were more willing to cooperate in economic experiments (Henrich et al., 2001). From the user interviews, we also gathered that they would also be prepared to pay a fee for the service. This is good for the sustainability of the app. This is explained further in the next section 'sustainability of the app'.

A quick online search shows that there already are some online auction markets in Ghana such as auctionexport.com and tomaton.com. However, most seem to focus on car auction as opposed to our focus on local food products like shea nuts. Also, these auctions are online only. This is a problem for farmers because there is low literacy (de Boer et al., 2012). To our best knowledge, there is no other auction service in Ghana that is also available via mobile phone. Another project from the VU, radio marché (de Boer et al., 2012), also developed an application for sales. It also focused on the market for local products and made use of alternative communication channels as mobile phone and radio. That was a big success. Therefore, we certainly think there is a market for our application.

2.2. Sustainability of the app

It is important that the app is sustainable. Therefore, we made a sustainability plan with a business model. We explain this with a value diagram that is shown in figure 3 on the next page.

With Vendu! we provide a way to sell or buy items. Because of our app, sellers will sell more products and maybe also sell their products for a higher price. We will ask for a small fee of the sellers for every auction. We choose this business model because it's simple to understand and it covers the costs per auction. After searching for prices of products in Ghana on the internet (Mar(k), 2010), we decided to make the fee 1 Zedi. However, we want to do some further user analysis to establish if this is an acceptable fee.

In the future, we want to experiment with different business models:

1. Payment of a fixed fee per auction
2. Have a monthly subscription fee for a (maximum) number of auctions
3. Payment of a variable fee per auction, for example a certain percentage of the profit gained

The costs of the calls will be carried by the sellers and buyers when they make the call. The Telco delivers the calls and gets money in return. The website and mobile phone platforms have to be hosted. This is also done by the Telco for which he gets money. Through the platform the buyers get their products after payment. The radio station is involved for advertisement purposes. They also will be paid for that.

An explanation of the cost considerations can be found in the appendix A.8.

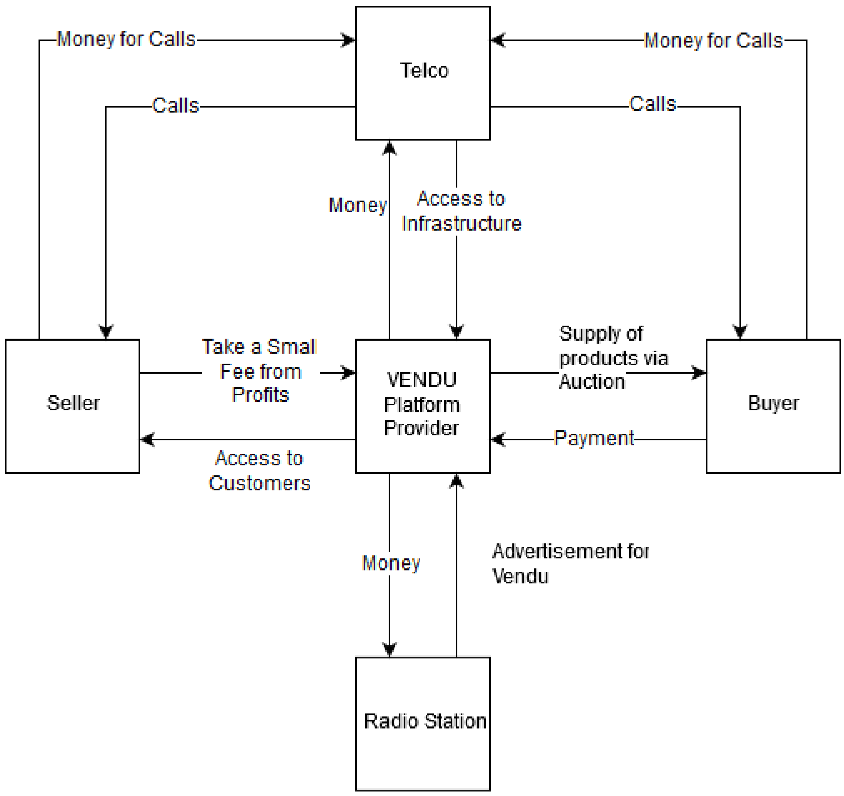


Figure 3. Value diagram of Vendu!

2.3. Deployment and marketing strategy

We distinguish two different target groups for deployment and marketing in Ghana: people in the city and people in rural Ghana. Table 3 on the next page shows the steps in the launch of the application and the first few weeks.

We will start with implementing the phone system and advertising the app on two different local radio stations. After that we will also implement the web interface and target city people.

The user experience is tested in the following ways. For the rural people we will held a living lab session. After speaking with one of the lecturers, we also want to do a demonstration of an auction so that they get what it actually is. There is also some related literature on this. Morawetz, De Groote and Kimenju (2001) did some analysis on experimental auctions in Africa and how they can differ from auctions in other countries. Citizens that use the web interface can read and right, so they will be asked to fill in a survey. Also, we will conduct interviews with them.

Period	Task
Week 1	Implement and test phone system.
Week 2 and 3	Arrange an interview at a local radio station to get the word out there that an app is being launched. Also, visit villages to explain how the app works and the arrange at least two farmers who would sell their items at the auction. Advertise at two local radio stations.
Week 4 and 5	Check user experience and improve the app.
Week 6	Advertise at four local radio stations.
Week 8	Start the hosting of the web interface.
Week 9 and 10	Flyer around in the city for the auction app and advertise at two local radio stations in the city.
Week 11 and 12	Check user experience and improve the app web interface.

Table 3. Deployment and marketing

Deployment and marketing risks

There are some risks for this application. First of all, people in rural Ghana might not get the auction system. Therefore, we need to be clear what an auction is. Also, we have to ensure that we startkick the application by searching for farmers who want to sell items in the first few weeks. If there are no auctions, then the app also will not come from the ground.

There is another risk in that the project team doesn't live in Ghana and therefore cannot fix problems on site. We will have to educate local people so that they could fix problems on site. This could be a problem since local people are not standard digital savvy.

Lastly, if we want to combine bids on the web interface with bids on the phone database, we need internet on the phone hardware. The first prototype uses the same database for phone and web interface but that requires internet for the phone hardware to sync the database. For the first launch, we will offer auctions for the interfaces separately to ensure the phone hardware doesn't require internet. So some auctions are reachable by phone and some by web interface. However, in the future this might be needed to also allow both interfaces for the same auctions.

3. Technical aspects

3.1. Theoretical background on auction mechanisms

One of the complications we reached in our project is the nature of selling multiple goods on a single auction. How do we determine which birds have more weight than the others? By prematurely selling goods on a first come first serve basis, we may indeed address the issue of exhausting the supply of goods, thereby preventing their spoilage or expiration, but it is also the case that this may also mean that we may be sacrificing the highest possible price as determined by a high demand on a scarce good. By introducing an adequate model, we intend to increase the profit for farmers while at the same time ensuring that the supply of goods is exhausted.

We explored two different models for our project: a uniform price auction and a descending-price auction. In a uniform price auction, you have a number of identical products like tomato's. People can bid on the products but the bids are generally blind (Markakis & Telelis, 2015). At the end of the auction, the highest bidder gets the number of products that he requested, then the second highest bidder gets his products and so on until all products are sold. The price that everyone pays is the same, namely the price of the lowest winning bid.

With a descending-price auction, the auction starts with the highest price. As the auction continues the price lowers until someone is willing to place a bid or until it has reached the minimum price (Mishra & Garg, 2006).

An advantage of the uniform price auction is that it is flexible enough to allow the combination of goods by multiple farmers into a single auction. It is also possible for this bidding mechanism to work for single farmer bids. Also, it tries to ensure that the goods are sold to the people who are willing to pay the most for them. However, this mechanism is complicated, and can make the adoption of our product more difficult. Also, the fact that this model requires that bids are kept secret until the end of the auction also has negative usability implications. The lack of transparency of this model can be perceived negatively by the end user. Since a high adoption rate for our product is a critical metric to the projects success, this model is inappropriate for our context.

An advantage of the descending price auction is that pricing is transparent, as it is determined by time. It will sell goods until they are exhausted, but given the fact that the goods start from high price and descend through the course of time means that people that are willing to pay a high price for a good will be served first, until goods are exhausted. This mechanism should allow for the supply of goods to be exhausted by advertising attractive prices for different buyer profiles. A major weakness for this model is that an initial price where the auction can descend from must be determined. In other words, knowledge about fair market prices must be available for the system to use. In the context of the application, this can prove to be difficult if sellers are made responsible of determining such a fair market price. This can be mitigated in future work by establishing predefined initial prices that are based on pricing for goods that can be purchased in urban supermarkets (where goods can be assumed to be more expensive than goods that are bought at the source of productions). By placing the responsibility on developers or application administrators to determine fair pricing, usability is not hindered and fair pricing can be achieved. Furthermore, this model can also be flexible enough such that auctions can aggregate goods from multiple farmers into single auctions. This may especially become necessary to allow the platform to scale whenever the demand for auction schedule slots becomes difficult to manage. However this is ultimately a design decision that must be determined by feedback from the end users.

In our project, we have opted for the descending-price auction model, as it is more intuitive the alternative. We opted to allow for a modification where farmers may suggest a minimum price for the goods, such that the system may not trickle down beneath this value. This is based on the double-auction model in which buyers and sellers both bid to suggest a fair price where both buyers and sellers are satisfied (Wurman, Walsh & Wellman, 1998). Other models can be explored, but this model is comparatively more intuitive, and is reasonably easy to develop for Vendu!.

3.2. Solution design

Current state of the prototype

Last sprint, we rebuild our calling service within the VSDK framework, made some set-up for the data models and added a web interface. This sprint we added the logic for auctioning an item, complete with data models.

Currently, we have a working application with a mobile phone and a web interface. In the mobile interface, you can buy or sell an item. The menus are built with audio files. In appendix A.6, call flow diagrams of the current prototype can be viewed. The choices of the caller are stored in a database. The mobile interface is integrated in the provided VSDK and the switcher interface.

The web interface is a seperate application. It has an landing page, a page for the current auction and a page where you can view the schedule. The schedule page now contains a table with auctions from the database. The current auction-page contains information about the auction like the seller and product. Instead of a hardcoded bidding, we implemented real logical for a real auction.

Design choices

During our last sprint we could not implement multi-numbered input as this was not implemented in the VSDK. We decided to replace the VSDK and serve vxml files through django directly to allow for more flexible voice interfaces. There are trade offs in productivity due to the difficulty of debugging vxml manually, and also considering that VSDK has the advantage of being a closed environment that prevents common bugs in voiceXML. However, this was mitigated by implementing prototypes in voxeo evolution, which has a complete debugger implementation and also by using syntax parsers from within the Kasadaka virtual interface.

The fact that we developed our voiceXML manually had negative implications in productivity due to the difficulty of producing a bug free implementation. We decided to prioritize developing a prototype which illustrates the dynamics of an auction, but certain aspects were deemed appropriate for the next iteration of the project. Such as the management of edge conditions in the voice interface, and the development of feedback from within the voice interface to ensure the user has not mistyped any selections.

Also, after feedback that the most spoken language in Ghana was English, we decided to focus on that instead of on French. As described in section 3.1, we encountered design challenges in determining an effective market model that could both exhaust farmer supply to prevent spoilage and also generate a fair profit for farmers. Upon analysing various existing models, we opted for a descending-price model.

The design choices for the auction models have already been described in 3.1.

Data model

Figure 4 shows an entity relation data model. In tables 4 until 6, also show the database structure per table.

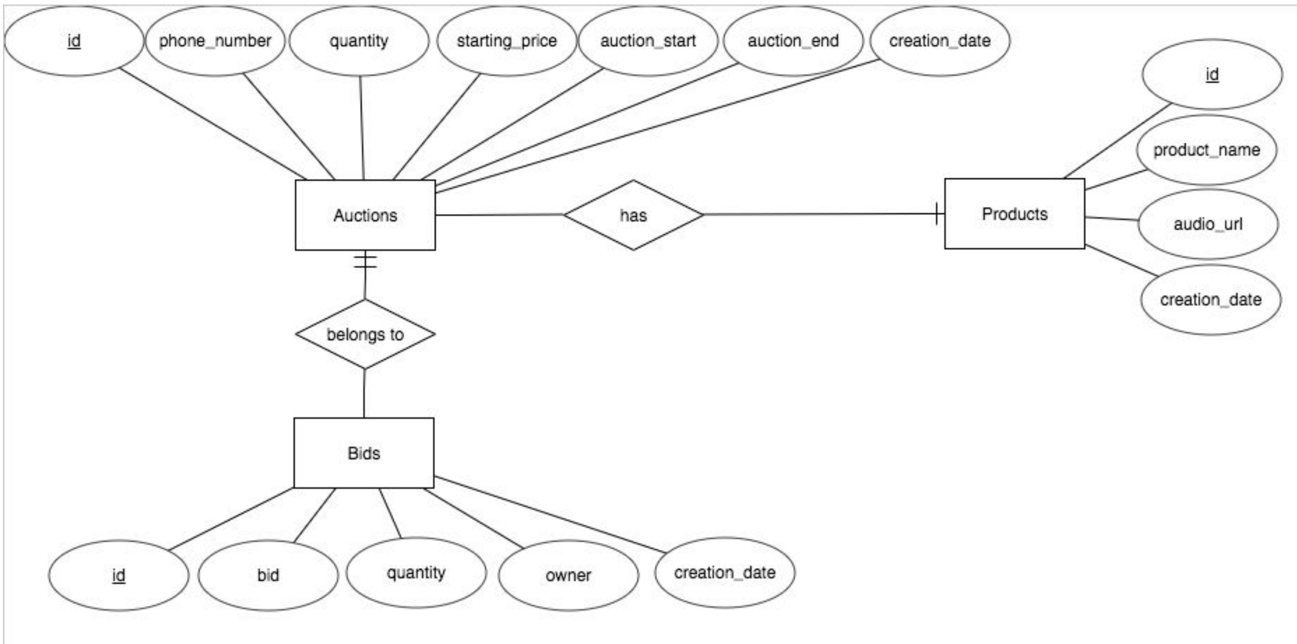


Figure 4. Entity relation data model

Auctions		
id	int	Primary key, unique identifier of the auction.
product	int	Foreign key reference to specific product, one auction has one product.
phone_number	string	String containing the phone number of the owner / creator of the auction.
quantity	int	The number of products he is willing to sell.
starting_price	long	The price at which the auction should end, the minimal amount which the seller wants to get from his product.
auction_start	datetime	The date/time at which the auction is scheduled to start.
auction_end	datetime	The date/time at which the auction is scheduled to end.
creation_date	datetime	The date/time at which the auction was created.

Table 4. Database table for ‘Auctions’

Products		
id	int	Primary key, unique identifier of the product.
product_name	string	Then name of the product we are going to sell.
audio_url	string	The URL of the audio file stating the name of the product.
creation_date	datetime	The date/time when the product was created.

Table 5. Database table for ‘Products’

Bids		
id	int	Primary key, unique identifier of the product.
auction	int	Foreign key reference to an auction. One bid belongs to one auction.
bid	int	The actual bid, price buyer wants to pay for the product.
quantity	int	The amount of produce buyer wants.
owner	string	The name of the buyer, currently restricted to solely being a phone number where the seller can reach the buyer on.
creation_date	datetime	The date/time when the bid was created.

Table 6. Database table for ‘Bids’

Third party
Our application currently does not use any third party data. The application is written in Python and ‘extended’ from the Kasadaka repository found on Github. The only third party content we currently access is a CDN hosted Bootstrap for styling the web application. The URL’s can be found on: https://www.bootstrapcdn.com/

3.3. Infrastructural issues

In order for our scenarios to work, customers need to have a phone with a working telephone connection or a computer with internet. We also need a device capable of handling incoming phone calls, for convenience we call this the Vendu! device. The Vendu! device needs to have a stable phone connection. Also to store and create auction, store bids and store contact details of customers, we need a database. The database has to be accessible from the Vendu! Device. For the web interface, we need a hosted website as well. The website must constantly be online during an auction because otherwise bids will be lost. This also means customers have to have a stable internet connection themselves when they want to bid on a product.

We can market the project via the radio, so we need a radio station were we can broadcast information about the auction and what the schedules are. There are also the telephone costs to consider. We need to send text messages to people with higher bids for the mobile interface.

The consequences of using this technology are mostly constraints to user experience. We have to make use of voice and phone technology. For example, we can't use multi-digit input. The other constraint within this category is the lack of any graphical feedback. Users do not have a graphical user interface on their phones, menus have to be navigated by voice or numerical input from the phone. Also the menus are spoken by the system and each option is being told over the phone, would the user be wanting to use the last option, the user has to listen to all other options first.

3.4. Instructions to test and use the applications

In this section, we described how to test and use the applications on the provided Heroku environment. In appendix C, we present a detailed installation guide to replicate our setup on different hardware.

Vendu! mobile interface
Our application is accessible for voice calls by consuming the following endpoint with a kasadaka device:
https://quiet-lake-72654.herokuapp.com/auction/vxml

Vendu! web interface
Go to the following url: https://quiet-lake-72654.herokuapp.com/auction/

The web interface prototype has two features. First of all, you can see the schedule for the auctions. Secondly, you can go the the auction that is currently taking place. It is also possible to add new items to the system by creating providing urls to the item's recording:

https://quiet-lake-72654.herokuapp.com/auction/product

Go to http://quiet-lake-72654.herokuapp.com/auction/ to add auctions and see the schedule.

From this interface, you can navigate to the most current auction to place a bid. You see the information about the current auction, how much time is left and what the last bid is. After the auction you get a message if you succeeded in buying the items or not.

Usage scenario
Below is a use case scenario for the Vendu! System. More scenarios can be found in appendix A.4.

<u>Use case scenario script - Bid on product through Vendu</u>

- Customer hears about the auction of honey on the radio
- Customer calls Vendu at the time the auction opens
- Customer chooses the open auction for honey from the voice menu
- Vendu tells the current minimum offer to bid for the honey per one unit
- Customer places a price offer
- Customer selects quantity of product being bought
- Customer confirms the placement of a bid on the product

3.5. Scope and fidelity of the prototype

The prototype at the moment implements a bidding system that is connected to the voice interface and the web-interface. This prototype illustrates the dynamics that we intend to introduce to extend the reach for farmers to sell their products. At this moment, you can both place bids and start new auctions, both in the web interface and the voice interface. Overall, this prototype demonstrates the dynamics of an auction system, and addresses issues of many market items going to waste due to lack of market access. In other words, there is a large supply of items being produced, however there is little demand for them. Furthermore, the use case describes that there are challenges for farmers in determining fair prices for their goods. Our system aims to address this issue by shifting this responsibility by allowing prices to be determined by the interaction of supply and demand. By implementing an economic model, we hope to develop a project that is able to generate a fair profit for farmers, while also ensuring that the supply of goods is exhausted to avoid waste.

There are certain aspects that still need to be developed in order to produce a robust system. First, it is necessary for the system to be able to prevent bids that are lower than the minimum price. Also, the instance where lower bids are offered than the current highest bid must be taken care of. This also has to do with race conditions, especially for users who buy products using the voice interface due to the length of the transaction. Furthermore, it is necessary to provide feedback to voice interface users regarding their purchase and sale options during the calls. This is important, as this prevents cases where users mistype their choices, thus unintended bids and auctions may be placed. Other features that may be implemented is a recording interface, where users may record new items to sell which are not yet defined in the system. The system as it is implemented at the moment is able to dynamically list the items for sale; also, it allows for new items to be created, assuming that an url to a valid recording is provided. However, this implies that the list of items that can be put for auction may grow very quickly. This means that a sensible solution must be developed to make popular items easy to access in the current web interface while still providing a wide gama of items for farmers to sell. This problem has implications with the usability of the project, as crowded voice interfaces are less effective.

4. Conclusion and future work

We made some big improvements with making the auction app work. We implemented the logic, designed the database structure and made the phone and web interfaces work together. We coupled the front end with the backend so that we can use actual data instead of mockup data.

For the future, we could improve the app on the following points:

- Getting user feedback to determine if users accept this business model (with 1 Cedi fee) and if the app is understandable.
- Adding more languages for localization.
- Making sure bids under the minimum price are not possible.
- Better handle race conditions.
- Providing feedback to voice interface users regarding their purchase and sale options during the calls.

Bibliography

de Boer, V., de Leenheer, P., Bon, A., Baah Gyan, N., van Aart, C., Guéret, C., Tuyp, W., Boyera, S., Allen, M., Akkermans, H. (2012). Radio Marché: Distributed Voice- en Web Interfaced Market Information System under Rural Conditions, *Proceedings of 24th International Conference on Advanced Information Systems Engineering, CAiSE'2012*, Gdansk, Poland

Corporate Council of Afrika, (2011, April 11). Business languages in Africa. Retrieved from http://www.corporatecouncilonafrica.com/business_languages_in_Africa on May 19, 2018

Henrich, J., Boyd, R., Bowles, S., Camerer, C., Fehr, E., Gintis, H., and McElreath, R. (2001). In Search of Homo Economicus: Behavioral Experiments in 15 Small-Scale Societies. *American Economic Review 91*(2), 73–78.

Mar(k), (2010, January 28). What one Ghanaian Cedi can get you… Retrieved from https://knoxshircore.wordpress.com/2010/01/28/what-one-ghanaian-cedi-can-get-you%E2%80%A6/ on May 20, 2018

Markakis, E., & Telelis, O. (2015). Uniform price auctions: Equilibria and efficiency. *Theory of Computing Systems*, *57*(3), 549-575.

Mishra, D., & Garg, R. (2006). Descending price multi-item auctions. *Journal of Mathematical Economics*, *42*(2), 161-179.

Morawetz, U. B., De Groote, H., & Kimenju, S. C. (2011). Improving the use of experimental auctions in Africa: Theory and evidence. *Journal of Agricultural and Resource Economics*, 263-279.

PWC Ghana, (2014). Growth accellerated; doing business in Ghana. Retrieved from https://www.pwc.com/gh/en/assets/pdf/doing-business-in-ghana-2014.pdf on May 19, 2018

Tim McDonelly, (2018). What's the World's Fastest-Growing Economy? Ghana Contends for the Crown. Retrieved from https://www.nytimes.com/2018/03/10/world/africa/ghana-worlds-fastest-growing-economy.html on May 19, 2018

Wurman, P. R., Walsh, W. E., & Wellman, M. P. (1998). Flexible double auctions for electronic commerce: Theory and implementation. *Decision Support Systems*, *24*(1), 17-27.

Appendices

A. Use Case Description

A.0 NAME

The name for our application is “Vendu!” which means “Sold!”. The use case we selected for this project is “market prices”.

A.1 SUMMARY OF KEY IDEA

Selling your products for the right price can be difficult. With Vendu! we will let the market decide the right price for the producers of goods in rural Ghana. Vendu! grants producers/farmers access to a ‘digital’ voice controlled auction system. Producers/farmers become sellers and enjoy the advantages of an auction system. They can set their base price, and the quantity of the product they are selling and the system will schedule an auction for their products.

On the scheduled time the auction opens and customers can bid on the products in the Vendu! System via phone or computer. Naturally the auction is only active for a predetermined amount of time (for instance 15 minutes). When it closes the products are sold to the highest bidder(s). An auctioning schedule is available via phone or computer. Also, radio stations will be informed and asked to transmit the schedule over the radio (for marketing purposes).

When the auction opens customers can place bids on the product. They can decide on the quantity of products they want to place a bit for. With the calling system, customers who have been outbid will receive a text or voice message that they have been outbid. The message will contain the price of the new bid. To provide auction-like behavior the customer can reply on the message (either through voice or text if possible) to up their bid again. The web interface will show the current offered bid and customers can place a new bid by clicking on a button.

Because of supply and demand a producer can expect a higher income on his goods. The auction system can be funded (or even yield profit) by taking a percentage of the profit. We are considering different options to monetize the project in this document.

A.2 ACTORS AND GOALS

Table A1 below shows the actors and their roles and goals.

Actor	Role/ Responsibilities	Goals
Seller	Producer of goods	To increase sales
Radio Station	Broadcast auctions	Increment revenue through a percentage of profits earned in the auctions
Customer	Buyer of goods	Acquire goods against a fair market price and to be informed of when a good is being sold
Telephone Service Company	Provides the supporting infrastructure for Vendu	Increment revenue by expanding their clientele

Table A1. Overview actors with roles and goals

A.3 CONTEXT AND SCOPE

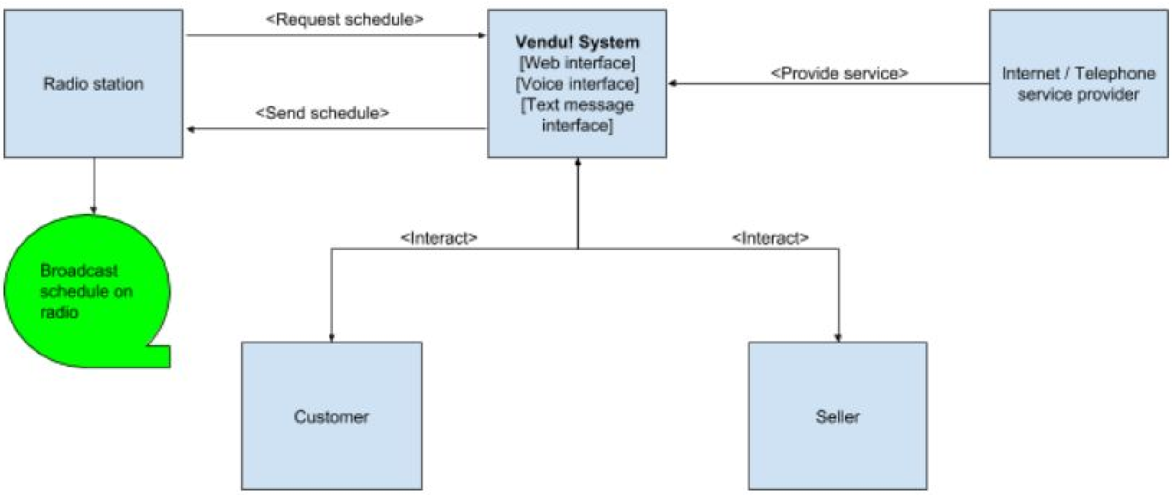


Figure A1. Interactions between parties.

External Stakeholders	
Stakeholder	**Concern**
Customer	• Wants to acquire products at a fair market price • Wants to be informed of when a product is being sold
Seller	• Wants to improve the profit from his products by increasing sales and overall revenue
Radio Station	• Wants a schedule which they can advertise on radio
Internet/ Telephone Service Provider	• A stable telephone service is asked from the internet / telephone service provider • A stable internet connection would be nice-to-have from the internet / telephone service provider

Internal Stakeholders	
Stakeholder	**Concern**
Maintainer of the auction system	• Wants to provide a good auction system for sellers in rural Ghana • Wants to create profit out of the auction system
Sahel Eco	• Wants to improve soil conditions in the Sahel region through spontaneous regeneration • Wants to incentivise the sale of non-timber goods to preserve trees

What is the scope of the scenario?
Out of scope:

- Delivery of products: we do not provide a service for the delivery of the products.
- Location: we assume the seller can reach the customer and vice versa. We assume they physically in acceptable / travelable range of each other.
- Confirmation that the producer physically owns the products he is selling. We are assuming he trades truthfully and genuine.
- Managing payments: transactions between producer and customer should be solved between them.
- Quantity: at this stage of development we haven't thought of a way to deal with multiple (local) measurement systems.
- Localization: at this stage of developing the prototype we will not include different languages other than English.

General pointers which should be considered in-scope:

- Customer
  - Customer can place bids on products.
  - Customer can place new bids when he is outbid.
  - Customer can request today's schedule through voice by calling the system or by going to the website.
- Seller
  - Seller can create an auction by calling the system or going to the website. He is asked the product's name, the number and a preferred auction day (today, tomorrow, the day after tomorrow).
  - Seller can get information on highest bidder by calling the system or by going to the website..
- Other
  - System can produce a schedule trough voice and text messaging.
  - System can produce a schedule via the web interface.

- System should support English.
    - System should be expendable in more languages.
    - We are restricting our prototype to a few products being the shea butter, honey.

What are success or performance measures for the scenario?
It is important to be able to measure measure the success of the application, and much of this should be dependent on the feedback of the buyers and sellers, as it is not the intent to impose an ICT solution which is not necessary. These aspects must be true:

- Both buyers and sellers must find the application accessible, such that they do not find it difficult to purchase and sell goods
- The application must be economically sustainable such that its users can find its value and anticipate the generation of enough funds for its user's profit and to sustain the application in the long run
- The application's business model must be intuitive and not go against cultural norms of its users and must not be incompatible with current market practices
- The application must not disrupt the end user's market negatively in the long run such that economic gaps are not exacerbated

Important (pre)conditions that must be or are assumed to be satisfied for our product:

- Customer
    - The customer has a phone with a working connection to call the system or go to the website.
    - The customer has a phone with the capability to send and receive text messages or has access to a computer with internet connection.
- Seller / Producer (farmer)
    - The producer physically owns the product he is selling (you cannot sell what you do not have).
    - The producer is able to ship the products to the customer.
- Radio station
    - Has a phone with a working connection or an internet connection to see or request the schedule.

A.4 USE CASE SCENARIO SCRIPT

The use cases below are some use cases for the Vendu! System.

<u>Use case scenario script - Bid on product through Vendu</u>

- Customer hears about the auction of honey on the radio
- Customer calls Vendu at the time the auction opens
- Customer chooses the open auction for honey from the voice menu
- Vendu tells the current minimum offer to bid for the honey per one unit
- Customer places an price offer
- Customer selects quantity of product being bought
- Customer confirms the placement of an bid on the product

<u>Use case scenario script - Start an auction with Vendu</u>

- Farmer is willing to sell honey, he has a batch of 50 pots to sell
- Farmer calls Vendu
- Farmer places a request for an auction to sell his batch of 50 pots with honey
- Vendu asks him at what price per unit (jar/pot) can the honey be sold
- Farmer inserts the starting price on his phone

<u>Use case scenario script - Being outbid, placing a new bid (dealing with competition)</u>

- Customer has placed an offer on a honey
- Customer receives a text message a few moments later that he is outbid by someone and that in order to still get the honey has to up his bid to x amount
- Customer can do two things:
    - He calls Vendu and places a new offer on the honey
    - He replies to the text message with a new offer on the honey
- A new offer is accepted and the customer is now the highest bidder.

Use case scenario script - Auction has concluded

- Farmer gets a text message his auction has concluded
- The message says his possible profit
- In order for farmer to unlock the contact details of his buyers, a fee must be paid, this can be done by calling Vendu.
- Farmer unlocks contact details by paying the fee (calling Vendu and opt to pay his fee)
    - Farmer can now request contact details of his buyers by calling Vendu
    - Farmer can now request contact details of his buyers by requesting a text message
    - Farmer can now request contact details of his buyers by visiting the web interface and logging in with his phone number and his auction code (id)

Use case scenario script - Radio broadcast using schedule from Vendu

- Radio advertises auction schedule
- Radio operator calls Vendu
- Radio operator selects from the voicemenu that he wants a schedule
- Vendu system tells the schedule to the operator

We found that placing bids and creating auctions are the most important use case scenarios.

A.5 INTERACTION AND COMMUNICATION

Figure A2 shows an example of the interaction between the bidder and the seller.

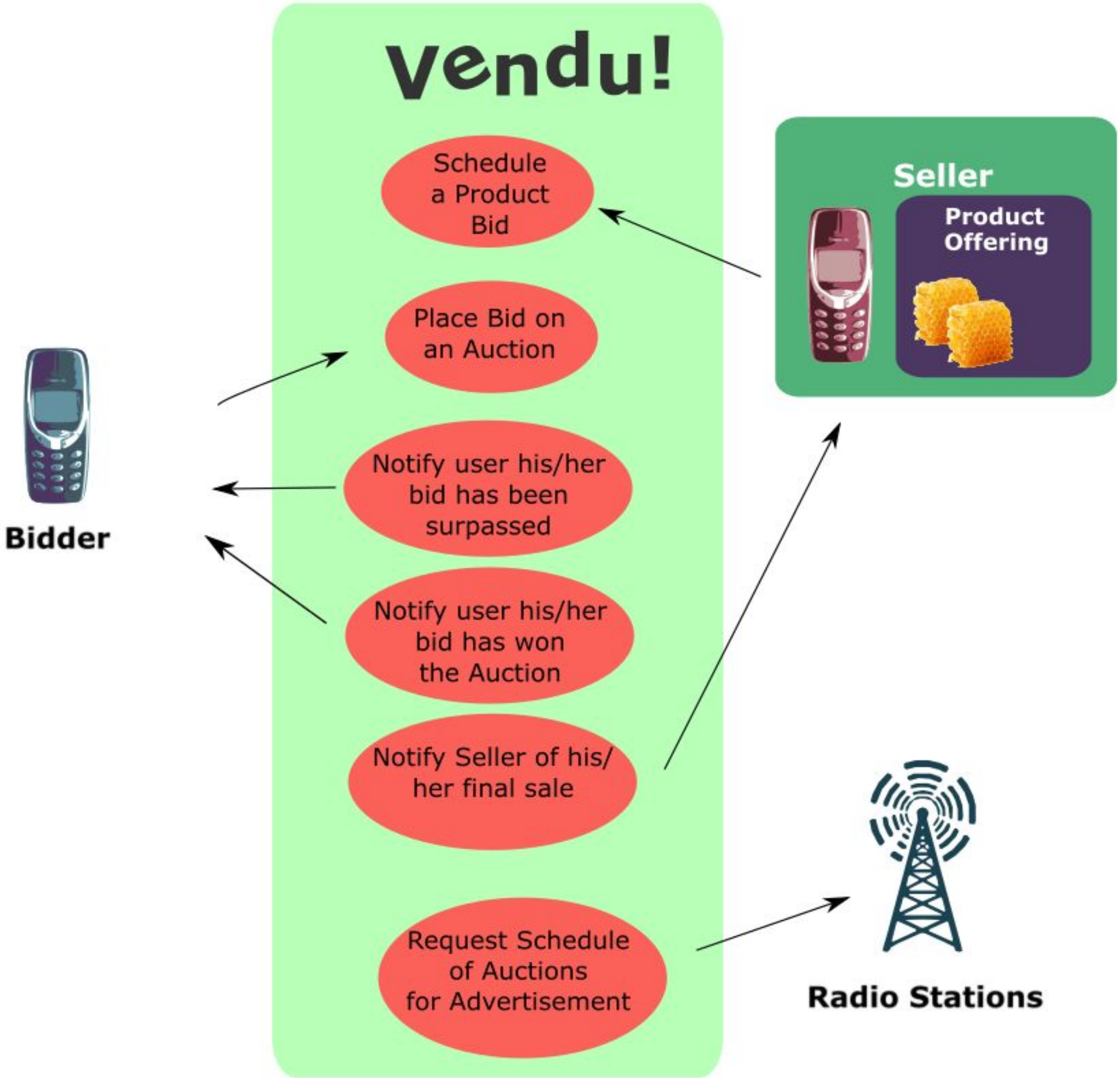


Figure A2. Example interaction between bidder and seller.

A.6 INFORMATION CONCEPTS

Figure A3 contains a flow diagram that represent the flow of the calling interface.

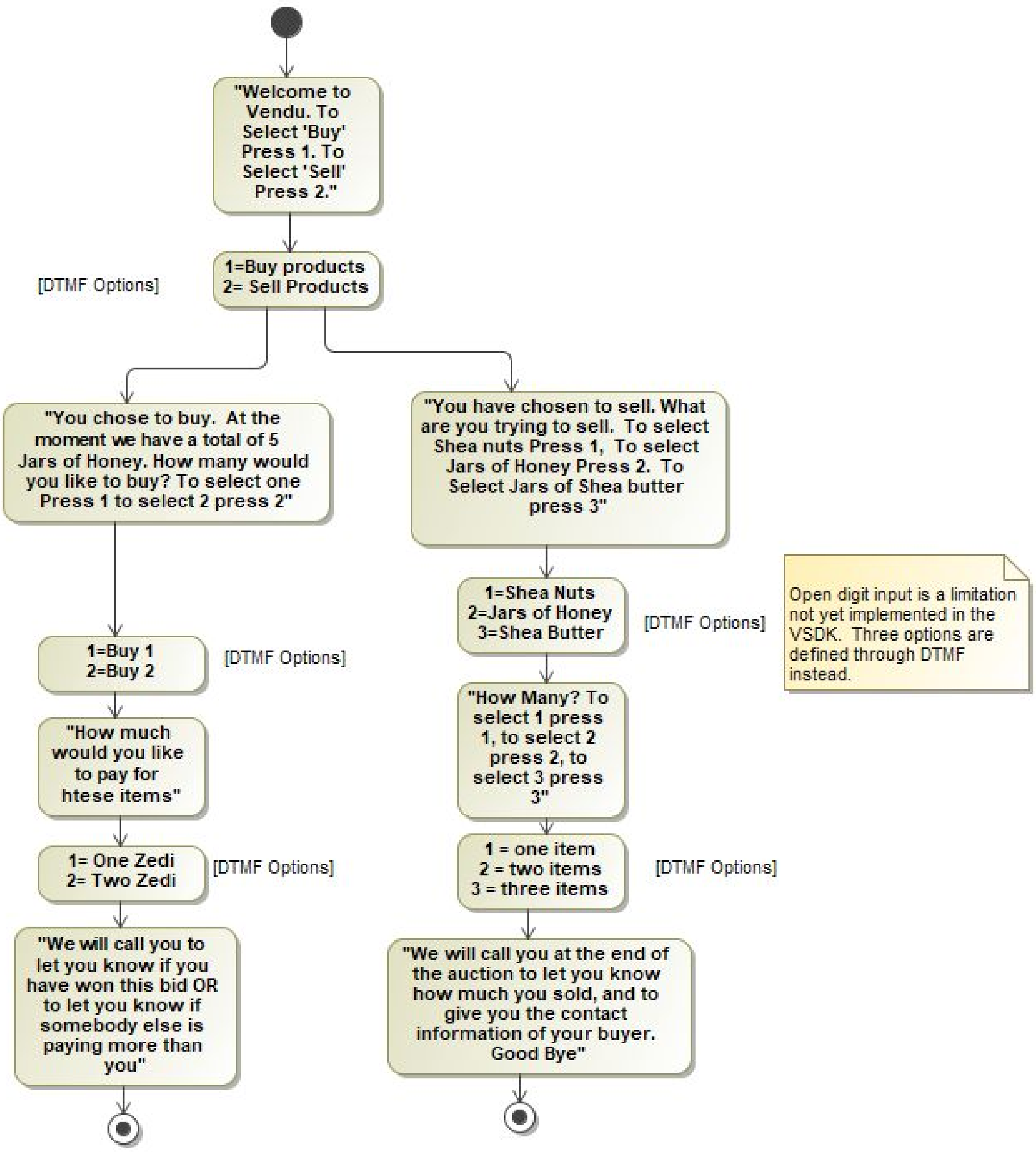


Figure A3. Flow diagram

A.7 TECHNOLOGY INFRASTRUCTURE

In order for our scenarios to work, customers need to have a phone with a working telephone connection or a computer with internet. We also need a device capable of handling incoming phone calls, for convenience we call this the Vendu! device. The Vendu! device needs to have a stable phone connection. Also to store and create auction, store bids and store contact details of customers, we need a database. The database has to be accessible from the Vendu! Device. For the web interface, we need a hosted website as well. The website must constantly be online during an auction because otherwise bids will be lost. This also means customers have to have a stable internet connection themselves when they want to bid on a product.
We can market the project via the radio, so we need a radio station were we can broadcast information about the auction and what the schedules are. There are also the telephone costs to consider. We need to send text messages to people with higher bids for the mobile interface.

The consequences of using this technology are mostly constraints to user experience. We have to make use of voice and phone technology. For example, we can't use multi-digit input. The other constraint within this category is the lack of any graphical feedback. Users do not have a graphical user interface on their phones, menus have to be navigated by voice or numerical input from the phone. Also the menus are spoken by the system and each option is being told over the phone, would the user be wanting to use the last option, the user has to listen to all other options first.

A.8 COST CONSIDERATIONS

What are estimated associated costs (operational, investment, development, in/outsource) for these technology infrastructure and components? Who carries these costs?

Item	**(Minimal) costs**	**Payment by**
GSM dongle, Raspberry PI and extra storage	140 euros per device	Project money
Text messages	0,09 euros per text message	Seller (/ project money or telco sponsorship)
Radio marketing	8 euros per advertisement	Project money
Implementation costs	160 hours - 30 euro per hour - 4800 euro	Project money
Maintenance	2 hours a week - 8 hours a month - 30 euros per hour - 240 euro monthly	System owner (which is funded by project profit and telco sponsorship)
System administrator	A local administrator for the system between 2200 - 3150 GHS which is between 399 - 570 EUR monthly	Project money
Hosting the website	30 euros per year	Project money

Table A2. Overview of costs and who pays them

Table A2 shows what the costs are and who is going to be paying for them. There is some project money needed to start the whole project. After the project is started only the text messages and maintenance need to be paid for. In the beginning, we could help carry these costs with project money or with a sponsorship with a telco (as to reduce text message costs). If more products with bigger batches are sold, sellers could carry the costs. But this needs to be tested in the field.

A.9 FEASIBILITY AND SUSTAINABILITY

What is the technical feasibility of the scenario (e.g. [project] risk analysis, technical obstacles to overcome, system-level impacts)?
There are several risks for this system. The system must be fast to process the offers and send the new price to other bidders. Secondly, there are scaling problems. Because there is only a small timeframe for the auction, the system should be able to handle more callers at once.

A technical obstacle would be dealing with different quantities (liters, grams or any local measurement system in use).
Another technical obstacle would be dealing with supply demand. For instance we have 50 products at base price which all have been sold to different bidders.
Bidder A 20 products at 5 cedi (current base price for this product)
Bidder B 10 products at 5 cedi (current base price for this product)
Bidder C 20 products at 5 cedi (current base price for this product)

Now we have bidder D to buy 15 products at 10 cedi per product. How to solve this issue?

What is the business and (socio-)economic feasibility and sustainability of the scenario?
The Radio Marché project showed that there was an interest in selling products via the radio. One of the problems of that project was that the demand was higher than the supply. Vedu makes sure people can call in and can get the product if they are willing to up their offer. Vendu also poses an advantage for sellers. They can determine a minimum selling price, so they can cover their costs. But they also have a chance to earn more. This extra money can be used for example to invest and produce more in the future. Stimulated in the right way, we can create an investment climate which can boost the local economical situation.

A risk to our product is that people don't really have auctions in the area. Therefore, there could be resentment to this way of selling products. Also, people might not understand the concept right away. Thirdly, the costs for texts will rise significantly if we have a lot of bids in place. Lastly, when prices are driven up all the time, people might not call anymore because they do not have enough money. This might lead to more divide in right and poor people.

What are possible goal conflicts and dependencies in the collaboration between the actors in the scenario?
There could be a goal conflict where the telephone company wants us to send as much texts as possible (because then they earn more) where the exploiter of the system wants to send a minimum amount.
Goal conflict of the seller wanting the highest possible price for his product whereas the buyer wants to have the lowest price available.

Are there important general preconditions for the scenario to work, and is it sufficiently interoperable with the wider context both in a business process and a technical sense?
As mentioned before, people should be willing to participate in an auction. Also, there should be enough products to sell. Thirdly, people should have phones and also the money to place a call. Lastly, knowledge of and trust in the auction system has to be created.

A.10 KEY REQUIREMENTS

Below, the functional (F) and non-functional (N) requirements are described with the MoSCoW method.

Functional Requirements (Must have)	
Code	**Description**
FM1	The seller must be able to call a number to submit his products to the auction.
FM2	The seller must be able to choose the product, enter minimum price and quantity of the sale when submitting his products to the auction.
FM3	People must be able to call a number to get the auction schedule of today.
FM4	The consumer must be able to place his bid with a standard amount of money by calling a number.
FM5	The consumer must be able to select the quantity he wants to bid for.
FM6	The consumer must be notified when he has been outbid.
FM7	The system should be able to communicate the outcome of the auction and provide contact information for the delivery and payment.
FM8	Auctions, bids and contact information has to be stored (in a db).
FM9	The system must be able to communicate the new offer via text or voice message.

Non-functional Requirements (Must have)	
Code	**Description**
NM1	The system must be navigated with a voice and interface menu.
NM3	The system must be 90% available and reachable during an auction.
NM4	The system must be fast enough to inform users of a new bid in less than 2 seconds.
NM5	The hardware must be cheap.

Functional Requirements (Should have)	
Code	**Description**
FS1	The consumer should be able to adjust the amount of money of the new bid before it is placed.
FS2	The system should -besides English- also have the possibility to choose French as a language.
FS3	People should be able to follow the auction and place bids via a Web interface.
FS4	The seller should be able to go to a website to submit his products to the auction.
FS5	People should be able to get the auction schedule of today via a web interface.

Non-functional Requirements (Should have)	
Code	**Description**
NS1	The system should be communicating in two other local languages.
NS2	The system should be 95% available and reachable during an auction.
NS3	The system should be safe for the most basic hacking activities.

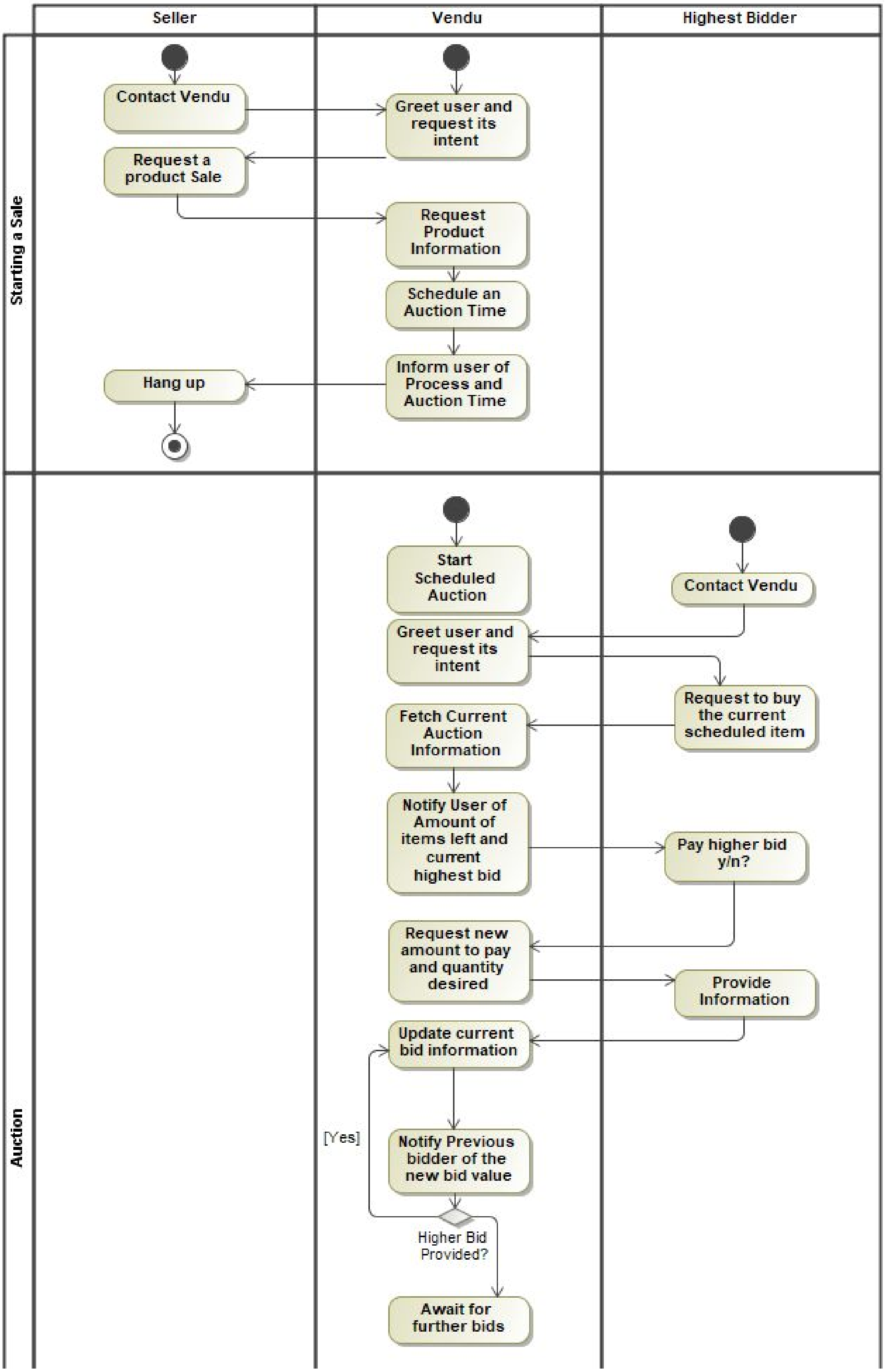
Functional Requirements (Could have)	
Code	**Description**
FC1	People must be able to call a number to get the auction schedule of today, tomorrow and the day after tomorrow.
FC2	People can view the auction schedule via a text message after they text a number.
FC3	The system could have more than two selection languages.

Non-functional Requirements (Could have)	
Code	**Description**
NC1	The system should be safe for normal hacking activities.

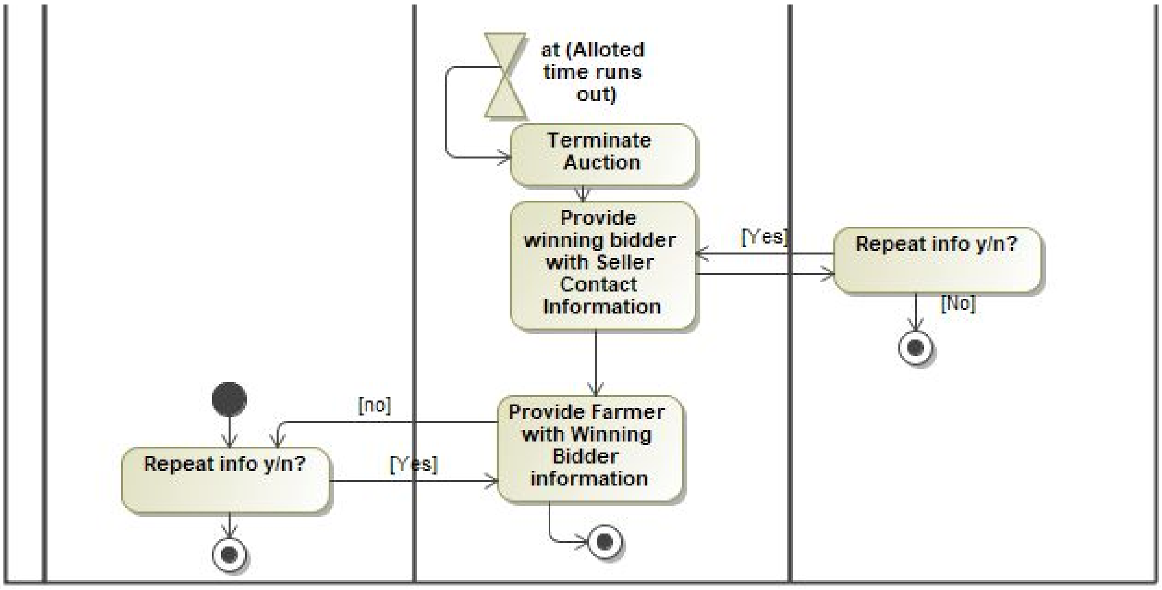
Functional Requirements (Wont have)	
Code	**Description**
FW1	The system won't have the delivery system.
FW2	The system won't be able to handle payments.

B. Sequence Diagram

On the next two pages, you will find the sequence diagram for our application.



Seller
Vendu
Highest Bidder
Starting a Sale
Contact Vendu
Greet user and request its intent
Request a product Sale
Request Product Information
Schedule an Auction Time
Hang up
Inform user of Process and Auction Time
Auction
Start Scheduled Auction
Contact Vendu
Greet user and request its intent
Request to buy the current scheduled item
Fetch Current Auction Information
Notify User of Amount of items left and current highest bid
Pay higher bid y/n?
Request new amount to pay and quantity desired
Provide Information
Update current bid information
[Yes]
Notify Previous bidder of the new bid value
Higher Bid Provided?
Await for further bids


at (Alloted time runs out)
Terminate Auction
Provide winning bidder with Seller Contact Information
[Yes]
Repeat info y/n?
[No]
[no]
Provide Farmer with Winning Bidder information
Repeat info y/n?
[Yes]

C. Detailed installation guide

Our application was developed using the VSDK platform. For the most part, the application was developed using a local version up to the final stretches of the project where changes were made directly to the heroku application in order to fix recording issues in a more rapid fashion. Our codebase is accessible and cloneable via the following link:

https://github.com/Cvalladares/KasaDaka-VSDK.git

Our application is accessible for voice calls by consuming the following endpoint:
https://quiet-lake-72654.herokuapp.com/auction/vxml

This endpoint must be consumed with a Kasadaka device with a fully deployed stack, which should include a VoiceXML interpreter and a means to connect to the telephony infrastructure. In order to replicate the relations generated from the database for local hosting, we generated a dump file for the relations created by the VSDK. Our latest dump file is called db_backup.json. The database relations, and thus the voiceXML application, can be loaded to a clone of the our Kasadaka repository or to a clone of the kasadakaVSDK by running in a Cloud hosting service such as Heroku. Relations can be imported to either of these contexts by issuing the following command in the project's root directory:

python manage.py loaddata db_backup.json