How to port Application in Raspberry Pi

Use Case - Knowledge Sharing Platform

Vivekanand JHA Student ID (UvA): 10865004 Student ID (VU): 2566951	Daoud URDU Student ID (UvA): 10576436 Student ID (VU): 2566817	Mustafa GÜLKAR Student ID (UvA): 10892 Student ID (VU): 2566

Introduction

This document helps the researchers to port the application in the context of Ghana. The prototype application is not finished, but the setup and design is there. The application is build on the use case of Knowledge Sharing.
Main purpose of the use case is to stimulate knowledge sharing between farmers in benefit of the farmers. Main technologies which supports the use case are: voice technologies (IVR, asterisk), web application and Linked Data technologies.

In this document we putted different sections to help the reader. In the first chapter we added a list of what is needed.
The second section takes the reader to the process of starting up the application. Following chapters show step by step what actually is achieved and how the .conf looks like. In the last chapter we added some questions which can be asked to the farmers. These answers might help the use case description and the software.

What is needed?

Technology	Example	Remarks
Hardware	Raspberry PI	At least the specifications mentioned in chapter "Infrastructure" of the assignement
OS	Linux Debian	VM Image provided
PBX Voice Server	Asterisk	
Voice Application	Ekiga/Linphone	
Web application	PHP/Java/HTML5	
Scripting	Python/LXTerminal	
Linked Data	Graph Data/ RDF/ Semantics/SPARQL	

Steps to take to start the process

1. Start the image/OS
2. Start serveral services completely:
    a. Asterisk
    b. Ekiga (if needed for debugging purposes)
    c. LXTerminal
3. Copy the .conf code (attached in the submittion) in the orginally .conf file

    a. Run the script *sudo leafpad /etc/asterisk/extensions.conf* in LXTerminal
    b. Replace the code in the document with the attached .conf code
    c. Save the document
    d. Run the script *asterisk -rx 'dialplan reload'* In LXTerminal
4. Now you can call with ekiga, the record application still doesnt work to record the question.

What is achieved in the prototype?

- Downloading and installing the VM did go successful. Only remarks where the network configurations should be set on NAT and the usb controllers disabled. This counts for both of the images (including Ekiga). This took eventually a lot of time, because the getting started (github) suggested the *network bridge* protocol.
- http://www.voip-info.org/wiki/view/Asterisk+cmd+Vxml
    - Link VXML with Asterisk is not working somehow.
- Recording a users voice is not working. This had the following reasons:
    - The USB connector was disabled in the VM. We did that intentionally due to errors. The microphone in the laptop uses USB on the motherboard.
    - For fixing the errors, and thus enabling the USB connector, we had to download an extension pack: https://www.virtualbox.org/wiki/Downloads
    - After installing the extension pack, the system needs a reboot.
    - After testing the application after the reboot, Asterisk gave the following error:

    [Apr 23 17:26:59] WARNING[3273]: file.c:766 ast_readaudio_callback: Failed to write frame
    [Apr 23 17:26:59] WARNING[3273]: app_playback.c:475 playback_exec: ast_streamfile failed on SIP/10.0.2.15-0000000c for beep
    [Apr 23 17:26:59] WARNING[3273]: file.c:663 ast_openstream_full: File /tmp/asterisk-recording-box-box does not exist in any format
    [Apr 23 17:26:59] WARNING[3273]: file.c:958 ast_streamfile: Unable to open /tmp/asterisk-recording-box-box (format 0x4 (ulaw)): No such file or directory
    [Apr 23 17:26:59] WARNING[3273]: app_playback.c:475 playback_exec: ast_streamfile failed on SIP/10.0.2.15-0000000c for /tmp/asterisk-recording-box-box

- Due to running out of time, we stopped with the technical part after this step.

Asterisk .conf file

```
; The "General" category is for certain variables.
[general]
```

```
clearglobalvars=yes

; The "default" section is what Asterisk looks into by default for calls
[default]
; Whoever is calling to the menu
exten => _.,1,Goto(menu,s,1)

; This is a custom section
[menu]
; Play a ring sound for 2 seconds
exten => s,1,Ringing
; Answer the phone
exten => s,n,Answer
; Start playing the background music
exten => s,n,Background(/var/lib/asterisk/sounds/custom/nature)
; Wait for the caller to press something
exten => s,n,WaitExten
; If the caller pressed "1" go to the submenu
exten => 1,1,Goto(submenu,s,1)
; If the caller pressed "2" hangup
exten => 2,1,Hangup
; If the caller pressed "3" go to the testmenu
exten => 3,1,Goto(testmenu,s,1)

; Another custom section
[submenu]
exten => s,1,Playback(beep)
exten => s,n,Hangup

[testmenu]
exten => _.,1,Playback(beep)
exten =>
_.,n,Record(/tmp/asterisk-recording-${CALLERID(name)}-${CALLERID(num)}.sln,2,10,q)
exten => _.,n,Playback(/tmp/asterisk-recording-${CALLERID(name)}-${CALLERID(num)})
exten => _.,n,Hangup
```

Questions to be asked from Farmer’s in Ghana:

1. Out of 3 scenarios described in knowledge sharing platform which one is more preferred by farmers?
2. Do they think of any other way of knowledge sharing which we can add as additional option in our application?

3. Are there specific questions or subjects which can be pre-defined in our application from farmer's perspective?
4. General feedback from farmer how do they think about the concept behind knowledge sharing platform?
