CS807 - Final Project

André E. dos Santos
Jhonatan S. Oliveira

March, 2016

University of Regina

Department of Computer Science

Resource Constrained Computing

CS807 - Final Project

André E. dos Santos
Jhonatan S. Oliveira

1. Reviewer Dr. David Gerhard
Department of Computer Science
University of Regina

2. Reviewer Mr. Trevor Tomesh
Department of Computer Science
University of Regina

March, 2016

André E. dos Santos
Jhonatan S. Oliveira
CS807 - Final Project
Resource Constrained Computing, March, 2016
Reviewers: Dr. David Gerhard and Mr. Trevor Tomesh

University of Regina
Department of Computer Science
3737 Wascana Pkwy
Regina, S4S 0A2

Abstract

[TODO]

Contents

1 Introduction

”

—
(TODO)

A *terminal information systems* is a computation system which [blank]. As a computation system, a terminal information systems is has specific constrains associated with its components. For instance, power constrains applies since the terminal usually is implemented in low power devices. In this report we present the proposal and implementation of a low cost terminal information systems with voice recognition feature.

Accessibility is a key assistive accommodation to enable people overcome physical, technological, or informational barriers [1]. In our project we sought for tools to enhance the experience and quality on a terminal information systems.The first tool we chose was Wit.ai, [blank]. Wit.ai is good [blank].

As foe an efficient system to manage the data and improve the information retrieval, the implementation was executed with meteor.js, [blank] Meteor is goof [blank].

The cost of the project is manly dictated by the hardware component that holds the system. Some platforms usually utilized for low cost system are [blank]. In our project the hardware chosen was Reaspberry.py, [blank]. Rep is goof [blank.]

With the implementation some issues were faced. For instance, one of our first goal of the project was to provide a map localization and direction to the user. This feature is commonly seen in big centers such malls, hospitals or airports. Althouhgt some systems are availabe, such as [blank]. we realized they are not at level for small systens, and when they are, not simply. This constarins are better descripted in Section [Blanck] We then turn our attentions for the soctware design and search engine of the termnal. We developted a semantic seach that permforms efficienty search on a tree structure. Well know algorith as breadth-fisrts and deapth serach was utilizled for this task.

The project presents many salient feature. Firts, it cn be voice activated and has state-of-the art speech recognition trhoeght the usage of the wit.ai. Second, it

performs an efficienty seach on the tree strcytue implemented on metero. Fetures as uncertanty decision,imcplemete infromation are well handled by the sustem. Thir, The implementations was condutc on a a reps.py, which is wel suited for the project since it is (i) [blank] and (ii) [blank]. This report is organized as follows. Section . Section . Section . Section .

2 Background

> *TODO.*
>
> — **TODO**
> (TODO)

2.1 Terminal Services

2.2 Reactive Programming

2.3 Graph Data Structures

2.4 Resource Constrains

3 The Project

> *TODO.*
>
> — **TODO**
> (TODO)

3.1 Introduction

3.2 Related Works

3.3 The Platforms

Meteor.

Raspberry Pi.

3.4 Implementation

Design process.

How the project works.

- Administration scheme: - Services - Data - Knowledge tree - The tree is built in server not in client - External - Map with key-value being the key the name of the external service and the value a function which runs the service

- Searching the tree - scoring the tree with one pass - problem: when we have the query found in the parent and child. Eg: tainara -> tainara@gmail.com

3.5 Results

Failures and successes.

4 The Ex-Project

> *TODO.*
>
> — **TODO**
> (TODO)

The initial plan for our final project included a service inside the app for indoor navigation. The idea was to allow the user to ask for a location within the University of Regina main campus and the app would trace a route from where the device hosting the app is to where the user requested. Indoor navigation itself is a complex and well known task problem in computer science and engineering. Our primary focus was to use some already available solution for indoor navigation, instead of trying to come up with our own solution. In this way, we wanted to show how resource constrain devices can still be used to provide such service by using cloud computing.

We did a broad research on publicly available solutions for indoor navigation, including paid, free or open source ones.

5 Conclusion

> *TODO.*
>
> — **TODO**
> (TODO)

Bibliography

[1] Federal disability reference guide. Human Resources and Skills Development Canada, Gatineau, Québec (2012)