

Teesside University
School of Computing
2011 - 2012

Final Year Project
BSc Computer Science

Social network library development

Thibaut Havel
L1247434

Supervisor : João F. Ferreira

Abstract

The main aim of this project is to develop a low-level library that is able to grab and store data from a social network. This library works in an embedded device and stored data had to be used to provide a simple service. To demonstrate the effectiveness of the final library, I created a demo service that interacts with a known social network (for example, Twitter).

Acknowledgements

I would like to thank my supervisor João Ferreira for his support all over the development of my project.

Contents

I	Abstract	1
II	Acknowledgements	2
III	Development of the library	5
1	Introduction	6
2	Methodology	7
3	Research	8
3.1	Operating System: FreeRTOS	8
3.1.1	Overview	8
3.1.2	Real-Time System	8
3.1.3	Libraries	8
3.1.4	POSIX simulator	8
3.2	Twitter authentication: OAuth protocol	8
3.2.1	Overview	8
3.2.2	Existing library in C	9
3.2.3	Register an application on Twitter	9
3.2.4	Required libraries	9
4	Design	10
4.1	Goal of the library	10

4.2	Requisites	10
4.3	Representation	10
5	Implementation	11
5.1	The library	11
5.1.1	Requisites	11
5.1.2	Simplification layer	11
5.2	A demo application	11
6	Testing of the library	12
7	Evaluation of the ptoject	13
7.1	Goal	13
7.2	Schedule	13
7.3	Improvements	13
8	Conclusion	14
IV	References	15

Development of the library

1 Introduction

Our society tends to use more and more social networks (for instance, Twitter or Facebook). At the same time, we are increasingly dependent on the use of embedded devices on a day-to-day basis (for instance, home automation). The goal of this project is to develop a platform that allows an effective and efficient communication between embedded devices and social networks.

The main aim is to develop a low-level library that is able to grab and store data from a social network. This library works in an embedded device and stored data has been used to provide a simple service. To demonstrate the effectiveness of the final library, I created a demo service that interacts with a known social network: Twitter.

One of my personal objectives was to improve my knowledge in low-level development and become familiar with the C language. Also I was looking forward to improving my software development skill while working with a very specific hardware.

Firstly, this report will present the way I started my initial research and how I designed the library according to my functional choices. Secondly, it will present how I've implemented these functionalities, and what I did to test my library. Finally, this report will end by an evaluation of the final product regarding the design choices, the way I've implemented them, and his reliability.

2 Methodology

My initial approach was to become familiar with the embedded device technology. I had to find an adapted, a small and a simple operating system to work with, thereby I chose **FreeRTOS**¹ supported from my supervisor.

(Existing library: no -¿ become familiar with FreeRTOS -¿ search about the way to development my own library: protocols, useful libraries -¿ parallel tests and design -¿ implementation -¿ final testing)

(Designed along the research, tests, and implementation)

(Use a diary)

(My schedule)

¹FreeRTOS is a light-weight Real-Time Operating System.

3 Research

3.1 Operating System: FreeRTOS

3.1.1 Overview

(Quick overview of the system: Free, open source, GP Licence, light-weight)

3.1.2 Real-Time System

(Kernel mechanism: priorities and scheduling)

3.1.3 Libraries

(Existing libraries: non-free libraries for specific hardware, light system library)

3.1.4 POSIX simulator

(Compilation of a library and a task)

3.2 Twitter authentication: OAuth protocol

3.2.1 Overview

(Common authentication mechanism: token, secret key system, include graphic representations)

3.2.2 Existing library in C

(Downloaded and tested library: samples hard to understand, idea: create a simple-to-use library layer)

3.2.3 Register an application on Twitter

(Way and proprieties of the registered application)

3.2.4 Required libraries

(libcurl: overview and it's seem hard to adapt to FreeRTOS, idea: create a very simple HTTP request library)

(OpenSSL: overview and it's seem hard to adapt to FreeRTOS, idea)

4 Design

4.1 Goal of the library

(Layer which simplify the authentication, using the adapted libraries of OAuth, libcurl and OpenSSL)

4.2 Requisites

(libcurl or any other HTTP library and OpenSSL, both used by OAuth)

4.3 Representation

(UML-like representation of the way it will work)

5 Implementation

5.1 The library

5.1.1 Requisites

(What did I do to solve the OpenSSL and Libcurl problem)

5.1.2 Simplification layer

(What the programmer need to know before coding: private and public key)

(How le library use OAuth to do the authentication)

5.2 A demo application

(Graphic representation of the use of my library layer)

6 Testing of the library

7 Evaluation of the ptoject

7.1 Goal

(Is my goal achieved, why/why not?)

(Is my work could be use by someone else, why/why not?)

7.2 Schedule

(Did I follow my schedule, why/why not?)

7.3 Improvements

(What is it possible to do to improve my library?)

8 Conclusion

References

- LastName, FN. (year), "Title" - *<http://www.link.com/document.pdf>*