**MINISTRY OF EDUCATION AND TRAINING**

**FPT UNIVERSITY**

Capstone Project Document

**Design and Implement**

**The model of Handheld POS**

**(point of sale) system for retailers**

|  |  |
| --- | --- |
| **Group 2** | |
| **Group members** | Nguyễn Thanh Hải – HaiNTSE90195  Đặng Minh Hiếu – HieuDMSE60969  Lê Long Hồ - HoLLSE61291  Huỳnh Hữu Nghị - NghiHHSE61055 |
| **Supervisor** | Bùi Đại Trí - TriBD |
| **Ext. Supervisor** | N/A |
| **Capstone Project code** | HHPOS |

-Ho Chi Minh City, ***1st Sep 2016*** *-*

*This page is intentionally left blank*

# Table of Contents

[Table of Contents 4](#_Toc464576986)

[A. System Implementation & Test 11](#_Toc464576987)

[1. Introduction 11](#_Toc464576988)

[1.1 Overview 11](#_Toc464576989)

[1.2 Test Approach 11](#_Toc464576990)

[2. Database Relationship Diagram 11](#_Toc464576991)

**Definitions, Acronyms, and Abbreviations**

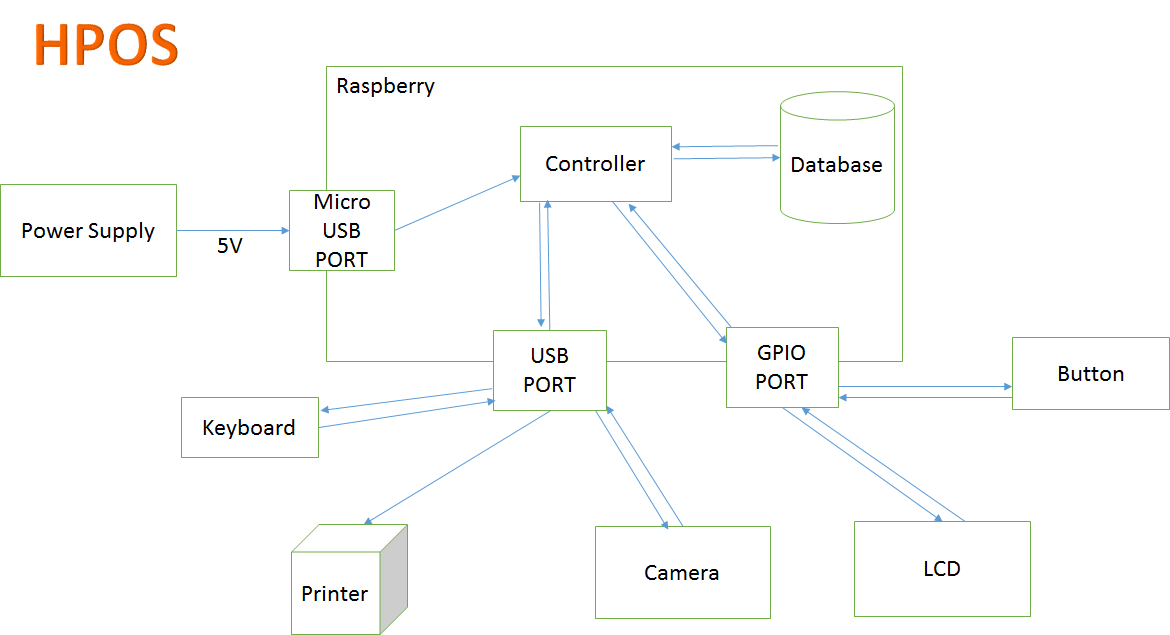
|  |  |
| --- | --- |
| **Name** | **Definition** |
| HHPOS | Handheld POS (point of sale) system for retailers |
| I/O | Input / Output |
| GPIO | General purpose input output |
| PC | Personal computer |
| IPC | Industrial PC |
| USB | Universal Serial Bus |
| UART | Universal Asynchronous Receiver/Transmitter |

D. Report No.4 Software Design Specification

1. Design overview

* This document describes the technical and user interface design of **HHPOS System**. It includes the architectural design, the detailed design of common functions and business functions.
* The architectural design describes the overall architecture of the system and the architecture of each main component and subsystem.
* The detailed design describes static and dynamic structure for each component and functions. It includes class diagrams, class explanations and sequence diagrams for each use cases.
* Document overview:
* Section 2: gives an overall description of the system architecture design.
* Section 3: gives component diagrams that describe the connection and integration of the system.
* Section 4: gives the detail design description which includes class diagram, class explanation, and sequence diagram to details the application functions.
* Section 5: describe screens design.
* Section 6: describe a fully attributed ERD.
* Section 7: describe algorithms.

2. System Architectural Design



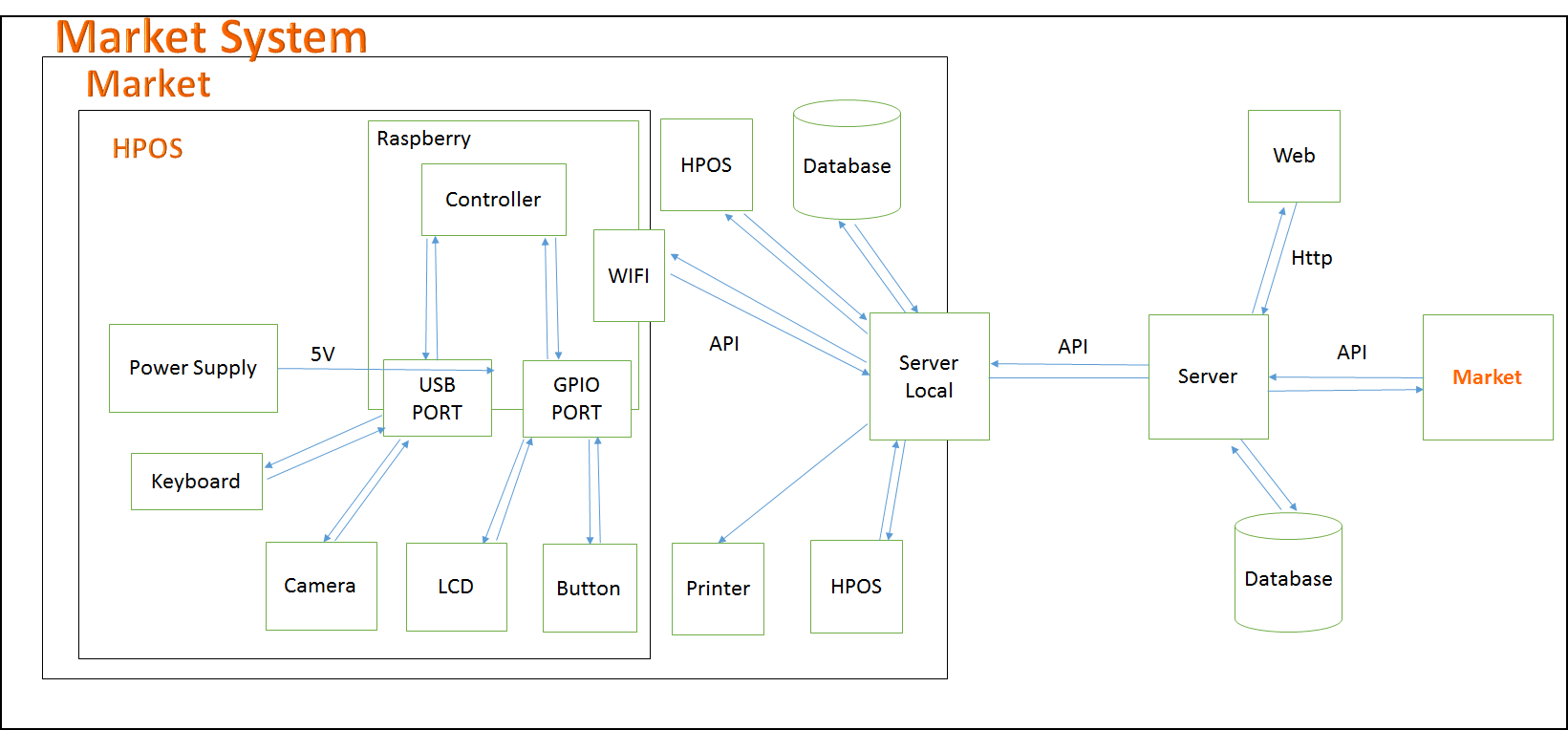


Figure: System overview architecture

2.1 External Interface Requirement

2.1.1. User interface

User interface use Graphical User Interface must be simple, clear and easy to use.

2.1.2. Hardware interface

Raspberry Pi B3 with SDRAM 1GB, Quad-core, 1.2GHz ARMv8 CPU

Camera 5.0 megapixcel.

LCD touch screen 3.5 inch.

SD card Sandisk 16GB class 10.

Power Supply 5V.

3. Component diagram

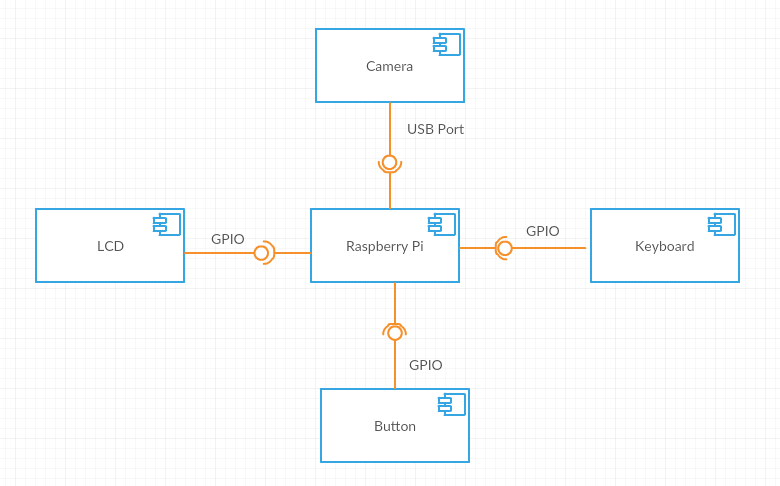


Figure: Component diagram

4. Detail description component

4.1. Hardware layer

4.1.1. Raspberry Pi B3

4.2. Software layer

Raspbian is an unofficial port of Debian Wheezy armhf with compilation settings adjusted to produce optimized "hard float" code that will run on the Raspberry Pi. This provides significantly faster performance for applications that make heavy use of floating point arithmetic operations. All other applications will also gain some performance through the use of advanced instructions of the ARMv6 CPU in Raspberry Pi.

Although Raspbian is primarily the efforts of Mike Thompson (mpthompson) and Peter Green (plugwash), it has also benefited greatly from the enthusiastic support of Raspberry Pi community members who wish to get the maximum performance from their device.

5. Detail diagram

5.1. Class diagram

5.2. Class diagram explanation

|  |  |
| --- | --- |
| **Class dictionary: Describe Class** | |
| **Class name** | **Description** |
| Gsm | Describe all information and methods relative to |

Table: Class diagram explanation

5.2.1. Gsm

Method

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Return type** | **Visibility** | **Description** |
| getUart\_filestream | Int | Public |  |
| setUart\_filestram | Void | Public |  |
| Init | Void | Public |  |
| sendSMS | Void | Public |  |

Table: Method of Gsm class

5.3. Flowchart diagram

6. User interface design

6.1. A

**Fields**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Description** | **Read only** | **Mandatory** | **Control Type** | **Data Type** | **Length** |
| 1 | Menu |  | Yes | No | Menu | N/A | N/A |
| 2 | Instruction |  | Yes | No | Label | N/A | N/A |

Table: Field of “ ” screen

6.2. B

**Fields**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Description** | **Read only** | **Mandatory** | **Control Type** | **Data Type** | **Length** |
| 1 | Menu |  | Yes | No | Menu | N/A | N/A |
| 2 | Instruction |  | Yes | No | Label | N/A | N/A |

Table: Field of “ ” screen

7. Algorithm

OpenCV (Open Source Computer Vision) is a library started by Intel in 1999. It focus on realtime image processing.