**Software Requirements**

**Specification**

**for**

**Cruz-Rabe Pharmacy Request System**

**Version 1.0 approved**

**Prepared by Carl Bueno**

**Asia Pacific College**

**November 8, 2016**

***Software Requirements Specification for Cruz-Rabe Pharmacy Request System*** ***Page ii***

**Table of Contents**

|  |  |  |  |
| --- | --- | --- | --- |
| **Table of Contents ..........................................................................................................................** | | | **ii** |
| **Revision History ............................................................................................................................** | | | **ii** |
| **1.** | **Introduction ..............................................................................................................................** | | **1** |
|  | 1.1 | Purpose ............................................................................................................................................ | 1 |
|  | 1.2 | Document Conventions .................................................................................................................... | 1 |
|  | 1.3 | Intended Audience and Reading Suggestions .................................................................................. | 1 |
|  | 1.4 | Product Scope .................................................................................................................................. | 1 |
|  | 1.5 | References ........................................................................................................................................ | 1 |
| **2.** | **Overall Description ..................................................................................................................** | | **2** |
|  | 2.1 | Product Perspective ......................................................................................................................... | 2 |
|  | 2.2 | Product Functions ............................................................................................................................ | 2 |
|  | 2.3 | User Classes and Characteristics ..................................................................................................... | 2 |
|  | 2.4 | Operating Environment .................................................................................................................... | 2 |
|  | 2.5 | Design and Implementation Constraints .......................................................................................... | 2 |
|  | 2.6 | User Documentation ........................................................................................................................ | 2 |
|  | 2.7 | Assumptions and Dependencies ...................................................................................................... | 3 |
| **3.** | **External Interface Requirements ...........................................................................................** | | **3** |
|  | 3.1 | User Interfaces ................................................................................................................................. | 3 |
|  | 3.2 | Hardware Interfaces ......................................................................................................................... | 3 |
|  | 3.3 | Software Interfaces .......................................................................................................................... | 3 |
|  | 3.4 | Communications Interfaces ............................................................................................................. | 3 |
| **4.** | **System Features ........................................................................................................................** | | **4** |
|  | 4.1 | System Feature 1 .............................................................................................................................. | 4 |
|  | 4.2 | System Feature 2 (and so on) ........................................................................................................... | 4 |
| **5.** | **Other Nonfunctional Requirements .......................................................................................** | | **4** |
|  | 5.1 | Performance Requirements .............................................................................................................. | 4 |
|  | 5.2 | Safety Requirements ........................................................................................................................ | 5 |
|  | 5.3 | Security Requirements ..................................................................................................................... | 5 |
|  | 5.4 | Software Quality Attributes ............................................................................................................. | 5 |
|  | 5.5 | Business Rules ................................................................................................................................. | 5 |
| **6.** | **Other Requirements ................................................................................................................** | | **5** |
| **Appendix A: Glossary....................................................................................................................** | | | **5** |
| **Appendix B: Analysis Models .......................................................................................................** | | | **5** |
| **Appendix C: To Be Determined List ............................................................................................** | | | **6** |

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason for Changes** | **Version** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

***Software Requirements Specification for Cruz-Rabe Pharmacy Request System*** ***Page 1***

**1. Introduction**

* 1. **Purpose**

The *System Requirements Specification* (SRS) document is intended to provide detailed overview of our software product, therefore, its constraints and goals. Our sponsored client, Cruz-Rabe Maternity and General Hospital, given us the opportunity to build a system wherein its scope is within pharmacy transactions of the hospital, including the interactions of patient and the management itself. The system from which we are building to provide convenient and more efficient in resources, namely *Cruz-Rabe Pharmacy Request System*.

The purpose of this document is to discuss how the system works, its functionalities and its features. Therefore, the developers will use this document as a guide to completely comprehend the requirements to build and integrate the software.

* 1. **Document Conventions**

Software Requirement Specification document is published on Microsoft Word 2016, using font “Times New Roman” and font size of “12dp” for context and 14dp” for headings and titles with Bold font styles.

This document has six (6) parts with sub-topics, namely:

1. Introduction
2. Overall Description
3. External Interface Requirements
4. System Features
5. Other Nonfunctional Requirements
6. Other Requirements
   1. **Intended Audience and Reading Suggestions**

Involve readers that were participated to the project:

* Project Manager -
* Developers -
* Project Adviser -
* Project Consultant -
* Project Client -
* Hospital Pharmacist -
* Nurse -
* Billing/Cashier -
* Patient -

**1.4 Product Scope**

Pharmacy Request System is intended for pharmacists to use in their service that assists patients’ drug requests that is prescribed by a doctor. The patients will present their prescription given by the doctor into the nurse that will be processed by the pharmacist. The pharmacist will do request of orders that will be paid upon the cashier and billing section. Once the orders are paid, the orders will be ready to dispense to the patient. Alternatively, the Pharmacy Requests will be automated and the data will be on a database for data warehouse. On discharging In-Patients will be provided with their records as fast as possible. The stocks they took will be listed down on their records in real time. Therefore, in paying their bills, the process will take lesser time. This scheme will ease the work of the auditing staff and improve the discharging process of patients.

**1.5 References**

Software Requirements Specification (SRS) document reference/s:

* GITHUB repository

[*https://github.com/seans888/Group-2*](https://github.com/seans888/Group-2)

* APC Wiki

[*http://projects2.apc.edu.ph/wiki*](http://projects2.apc.edu.ph/wiki)

* IBM Bluemix

*https://console.ng.bluemix.net*

* MOODLE

*https://moodle2.apc.edu.ph*

**2. Overall Description**

**2.1 Product Perspective**

*<Describe the context and origin of the product being specified in this SRS. For example, state whether this product is a follow-on member of a product family, a replacement for certain existing systems, or a new, self- contained product. If the SRS defines a component of a larger system, relate the requirements of the larger system to the functionality of this software and identify interfaces between the two. A simple diagram that shows the major components of the overall system, subsystem interconnections, and external interfaces can be helpful.>*

**2.2 Product Functions**

Based on the user and software function:

DOCTOR

* check-up patient (in and out patient)
* create prescription for the patient (in and out patient)

NURSE

* create a request based on the prescription coming from the doctor
* distribute order/s to in-patient
* process the discharge request slip of in-patient.

PHARMACY

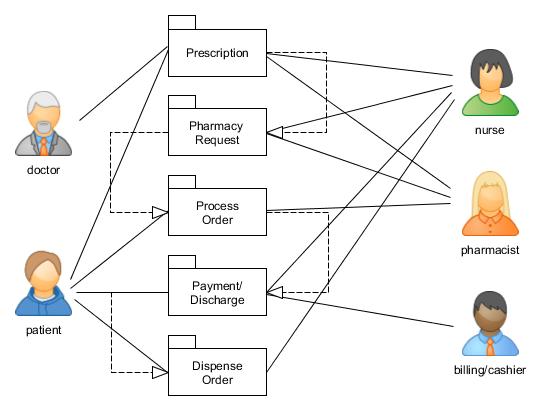
* check for the prescription’s validity (if the drug or supply is necessary to give to patients)
* check the availability of orders and its effectivity
* approve the request charge slip
* prepare orders for dispense to patients

PATIENT

* present prescription to the pharmacy for out-patient
* admit to the hospital if in-patient
* confirm order/s before processing
* take necessary and available medicine and supply upon request

BILLING & CASHIER

* confirm corresponding amount before generating to the system
* process service and order charge/s for in-patient for discharge clearance
* accept payment of orders from the patient
* generate official receipt



**2.3 User Classes and Characteristics**

The first hand user of Cruz-Rabe Pharmacy Request System is the hospital pharmacist. The system should only interact with a licensed pharmacist with sufficient knowledge on the software.

The first person user should consist of the following characteristic:

* Licensed pharmacist
* Computer-literate
* Can interact with the user interface and know how the system works
* Can do basic software operations
* Have knowledge on troubleshooting

**2.4 Operating Environment**

The system will run and be supported by the following:

* [Core 2 Duo](https://en.wikipedia.org/wiki/Core_2_Duo) or [Athlon X2](https://en.wikipedia.org/wiki/Athlon_X2) at 2.4 GHz or Higher
* At least 2GB of RAM
* 30GB of ROM or Higher
* Network Connection
* Windows 10 or Later
* Internet Browser (Google Chrome, Mozilla Firefox, etc)
* PHP
* MySQL
* XAMPP Control Panel

**2.5 Design and Implementation Constraints**

*<Describe any items or issues that will limit the options available to the developers. These might include: corporate or regulatory policies; hardware limitations (timing requirements, memory requirements); interfaces to other applications; specific technologies, tools, and databases to be used; parallel operations; language requirements; communications protocols; security considerations; design conventions or programming standards (for example, if the customer’s organization will be responsible for maintaining the delivered software).>*

**2.6 User Documentation**

*<List the user documentation components (such as user manuals, on-line help, and tutorials) that will be delivered along with the software. Identify any known user documentation delivery formats or standards.>*

***Software Requirements Specification for Cruz-Rabe Pharmacy Request System*** ***Page 3***

**2.7 Assumptions and Dependencies**

**AS-1:** Additional equipment, services, and maintenance will be improvised for the authorized personnel to do the business processes.

**AS-2:** The hospital will be developed to ensure the system was implemented well-lesser hardcopies, more softcopies for the records.

**DE-1:** If the pharmacy records will be accessible with the other departments to support the system for consistency of the data.

**3. External Interface Requirements**

**3.1 User Interfaces**

*<Describe the logical characteristics of each interface between the software product and the users. This may include sample screen images, any GUI standards or product family style guides that are to be followed, screen layout constraints, standard buttons and functions (e.g., help) that will appear on every screen, keyboard shortcuts, error message display standards, and so on. Define the software components for which a user interface is needed. Details of the user interface design should be documented in a separate user interface specification.>*

**3.2 Hardware Interfaces**

* [Core 2 Duo](https://en.wikipedia.org/wiki/Core_2_Duo) or [Athlon X2](https://en.wikipedia.org/wiki/Athlon_X2) at 2.4 GHz or Higher
* At least 2GB of RAM
* 30GB of ROM or Higher
* Network Connection and Peripherals
* Mouse
* Keyboard
* Printer/Copier Machine

**3.3 Software Interfaces**

* Windows 10 or Later
* Internet Browser (Google Chrome, Mozilla Firefox, etc.)
* PHP
* MySQL
* XAMPP Control Panel

**3.4 Communications Interfaces**

***Software Requirements Specification for Cruz-Rabe Pharmacy Request System*** ***Page 4***

**4. System Features**

*<This template illustrates organizing the functional requirements for the product by system features, the major services provided by the product. You may prefer to organize this section by use case, mode of operation, user class, object class, functional hierarchy, or combinations of these, whatever makes the most logical sense for your product.>*

**4.1 System Feature 1**

*<Don’t really say “System Feature 1.” State the feature name in just a few words.>*

1. Description and Priority

*<Provide a short description of the feature and indicate whether it is of High, Medium, or Low priority. You could also include specific priority component ratings, such as benefit, penalty, cost, and risk (each rated on a relative scale from a low of 1 to a high of 9).>*

1. Stimulus/Response Sequences

*<List the sequences of user actions and system responses that stimulate the behavior defined for this feature. These will correspond to the dialog elements associated with use cases.>*

1. Functional Requirements

*<Itemize the detailed functional requirements associated with this feature. These are the software capabilities that must be present in order for the user to carry out the services provided by the feature, or to execute the use case. Include how the product should respond to anticipated error conditions or invalid inputs. Requirements should be concise, complete, unambiguous, verifiable, and necessary. Use “TBD” as a placeholder to indicate when necessary information is not yet available.>*

*<Each requirement should be uniquely identified with a sequence number or a meaningful tag of some kind.>*

REQ-1:

REQ-2:

**4.2 System Feature 2 (and so on)**

**5. Other Nonfunctional Requirements**

**5.1 Performance Requirements**

*<If there are performance requirements for the product under various circumstances, state them here and explain their rationale, to help the developers understand the intent and make suitable design choices. Specify the timing relationships for real time systems. Make such requirements as specific as possible. You may need to state performance requirements for individual functional requirements or features.>*

***Software Requirements Specification for Cruz-Rabe Pharmacy Request System*** ***Page 5***

**5.2 Safety Requirements**

*<Specify those requirements that are concerned with possible loss, damage, or harm that could result from the use of the product. Define any safeguards or actions that must be taken, as well as actions that must be prevented. Refer to any external policies or regulations that state safety issues that affect the product’s design or use. Define any safety certifications that must be satisfied.>*

**5.3 Security Requirements**

*<Specify any requirements regarding security or privacy issues surrounding use of the product or protection of the data used or created by the product. Define any user identity authentication requirements. Refer to any external policies or regulations containing security issues that affect the product. Define any security or privacy certifications that must be satisfied.>*

**5.4 Software Quality Attributes**

*<Specify any additional quality characteristics for the product that will be important to either the customers or the developers. Some to consider are: adaptability, availability, correctness, flexibility, interoperability, maintainability, portability, reliability, reusability, robustness, testability, and usability. Write these to be specific, quantitative, and verifiable when possible. At the least, clarify the relative preferences for various attributes, such as ease of use over ease of learning.>*

**5.5 Business Rules**

*<List any operating principles about the product, such as which individuals or roles can perform which functions under specific circumstances. These are not functional requirements in themselves, but they may imply certain functional requirements to enforce the rules.>*

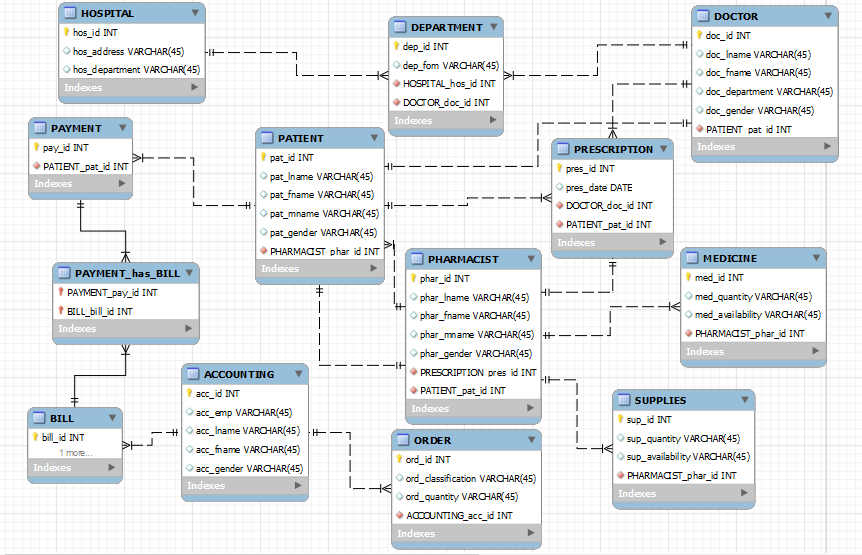
**6. Other Requirements**

*<Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.>*

**Appendix A: Glossary**

**Appendix B: Analysis Models**

Entity-Relationship Diagram



***Software Requirements Specification for Cruz-Rabe Pharmacy Request System*** ***Page 6***

**Appendix C: To Be Determined List**

*<Collect a numbered list of the TBD (to be determined) references that remain in the SRS so they can be tracked to closure.>*