**Software Requirements**

**Specification**

**for**

**A5MMS**

**Version 1.0 approved**

**Prepared by <author>**

**<organization> <date created>**

# Table of Contents

[Table of Contents ii](#_Toc6260)

[Revision History ii](#_Toc6261)

[1. Introduction 1](#_Toc6262)

[1.1 Purpose 1](#_Toc6263)

[1.2 Document Conventions 1](#_Toc6264)

[1.3 Intended Audience and Reading Suggestions 1](#_Toc6265)

[1.4 Product Scope 1](#_Toc6266)

[1.5 References 1](#_Toc6267)

[2. Overall Description 2](#_Toc6268)

[2.1 Product Perspective 2](#_Toc6269)

[2.2 Product Functions 2](#_Toc6270)

[2.3 User Classes and Characteristics 2](#_Toc6271)

[2.4 Operating Environment 2](#_Toc6272)

[2.5 Design and Implementation Constraints 2](#_Toc6273)

[2.6 User Documentation 2](#_Toc6274)

[2.7 Assumptions and Dependencies 3](#_Toc6275)

[3. External Interface Requirements 3](#_Toc6276)

[3.1 User Interfaces 3](#_Toc6277)

[3.2 Hardware Interfaces 3](#_Toc6278)

[3.3 Software Interfaces 3](#_Toc6279)

[3.4 Communications Interfaces 3](#_Toc6280)

[4. System Features 4](#_Toc6281)

[4.1 System Feature 1 4](#_Toc6282)

[4.2 System Feature 2 (and so on) 4](#_Toc6283)

[5. Other Nonfunctional Requirements 4](#_Toc6284)

[5.1 Performance Requirements 4](#_Toc6285)

[5.2 Safety Requirements 5](#_Toc6286)

[5.3 Security Requirements 5](#_Toc6287)

[5.4 Software Quality Attributes 5](#_Toc6288)

[5.5 Business Rules 5](#_Toc6289)

[6. Other Requirements 5](#_Toc6290)

[Appendix A: Glossary 5](#_Toc6291)

[Appendix B: Analysis Models 5](#_Toc6292)

[Appendix C: To Be Determined List 6](#_Toc6293)

# Revision History

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

# Introduction

## Purpose

The purpose of the project is to make a Human Resource System that will improve the management of employees of the corporation and an Automated Payroll system that will decrease the work time greatly.

## Document Conventions

The document will be written on standard Arial, 12, single-spacing. Every requirement is to have its own priority

## Intended Audience and Reading Suggestions

The documents of the product are intended for the A5MMS corporation and for the developers who wants to further enhance this product. For the A5MMS, the user must at least read the business case/project overview and the user manual to have a good knowledge on how the system will work, and for the developers, it is suggested to go through the diagrams of the system first rather than the business case.

## Product Scope

Refer to the project Vision and Scope

## References

<http://projects2.apc.edu.ph/wiki/index.php/Project_-_A5MMS_-_102> [1]

# Overall Description

## Product Perspective

The product will be a full replacement of the manual encoding system of the corporation since it their system will be almost completely automated due to its automated payroll system and employee management system.

## Product Functions

The major functions of the products are:

* Automated Payroll
* Employee Management System
* Client Management System

## User Classes and Characteristics

* A5MMS Office Staff - has complete admin access, can modify data within the system.
* A5MMS Contracted Employees - has no admin access, can only access personal data within the system.
* Ordinary users- has no access to the system internally, can inquire to the system externally

## Operating Environment

The system can run at a standard PC with ‘Windows’ as its operating system. The minimum version of the operating system should only be Windows 7.

## Design and Implementation Constraints

There will be a limitation on the development of the automated payroll system since the corporation did not want to share information on how the employees are paid due to confidentiality

## User Documentation

Together with the finished product will also include a user-manual on how to operate the system. There is also a hard copy of documentation included on how the system is integrated, and can also be accessed via the projects wiki (look at references [1]).

## Assumptions and Dependencies

<List any assumed factors (as opposed to known facts) that could affect the requirements stated in the SRS. These could include third-party or commercial components that you plan to use, issues around the development or operating environment, or constraints. The project could be affected if these assumptions are incorrect, are not shared, or change. Also identify any dependencies the project has on external factors, such as software components that you intend to reuse from another project, unless they are already documented elsewhere (for example, in the vision and scope document or the project plan).>

# External Interface Requirements

## 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.>

## Hardware Interfaces

<Describe the logical and physical characteristics of each interface between the software product and the hardware components of the system. This may include the supported device types, the nature of the data and control interactions between the software and the hardware, and communication protocols to be used.>

## Software Interfaces

<Describe the connections between this product and other specific software components (name and version), including databases, operating systems, tools, libraries, and integrated commercial components. Identify the data items or messages coming into the system and going out and describe the purpose of each. Describe the services needed and the nature of communications. Refer to documents that describe detailed application programming interface protocols. Identify data that will be shared across software components. If the data sharing mechanism must be implemented in a specific way (for example, use of a global data area in a multitasking operating system), specify this as an implementation constraint.>

## Communications Interfaces

<Describe the requirements associated with any communications functions required by this product, including e-mail, web browser, network server communications protocols, electronic forms, and so on. Define any pertinent message formatting. Identify any communication standards that will be used, such as FTP or HTTP. Specify any communication security or encryption issues, data transfer rates, and synchronization mechanisms.>

# System Features

A5MMS Have two (2) module which are HR and Payroll. For the HR module have three (3) subdivided modules the Announcement, Client Management and Employee Management Module. Also for Payroll modules have three (3) subdivided modules which are Pay slip, Payroll Generation, Attendance management.

## System Feature 1

<Don’t really say “System Feature 1.” State the feature name in just a few words.> 4.1.1 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).>

4.1.2 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.>

4.1.3 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:

## System Feature 2 (and so on)

# Other Nonfunctional Requirements

## 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.>

## 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.>

## 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.>

## 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.>

## 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.>

# 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

<Define all the terms necessary to properly interpret the SRS, including acronyms and abbreviations. You may wish to build a separate glossary that spans multiple projects or the entire

organization, and just include terms specific to a single project in each SRS.>

# Appendix B: Analysis Models

<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams.>

# 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.>