CMS SYSTEM Requirements Specification (IJG)

Version 1.0

By Joel Chipoya Kalimbwe

Revision History

Date	Version	Role	Name
3/03/2025	v1	Developer	Joel Chipoya Kalimbwe

Table of Contents

Software Requirements Specification

1. Introduction

1.1 Purpose

The purpose of this Software Requirements Specification (SRS) is to define the functionalities, constraints, and scope of a multi-system platform integrating four interconnected applications:

1. ***Jobhant App** – A job marketplace where companies post job listings, and users apply for jobs. Employers can schedule interviews and manage applicants. Candidates also manage their profiles, follow companies and like posts*
2. ***Jobhant Admin panel** - to manage the app. Manage companies and candidate profiles, add admins and manage admins, companies will manage their profiles, there must be a master admin, . This panel needs to be installed manually with a company that partner with ozomcapital for their job listing and recruitments partner, where they configure it with their own settings (domains and hosting servers)*
    - *Each company or client will install the Desktop app and first lunch asks for a company subdomain or company domain. And after that they access the API for sending information to Hr and others. Companies can also use the web app if they want. (example: first lunch,*
    - *company subdomain*
    - *company name*
    - *database name*
    - *password*
    - *e.t.c*
    -
    - *for connection always when the subdomain or the domain is entered in the app the desktop app or web app returns the credential page where users need to login in for the first lunch with their database name and passwords and than type of role, these database based can either point to their own servers or AWS, depends. (we register the subdomains on AWS and point the database either on AWS or Own servers)*
    - *Or login page provides the list of subdomains registered within us and users select from them and enter their login details and their roles as well.*
    - *This can also be implemented on the web, because both the desktop app and the Web App will be pulling the registered subdomains and they will be logining with their details. These applies to all admin apps (A**dvertisement Partner Portal, Hr management System, Payroll System, OzomAccounting** and both of them will have web and desktop)*
3. ***Myozom App** – An employee management app and self services for employees that integrates with Jobhant to transition successful candidates into employees and employees can also apply for loans within the app, like car loans, houses and manage them there.*
4. ***Advertisement Partners portal** - This portal is meant for Advertisement Partners that want to*

*advertise their products or services with our softwares (**Jobhant** app and **Myozom** app). E.g for MyOzom app, companies can advertise their business like giving loans and getting loan applications from **Myozom** App to the portal and approving their loans there. (This one of the advertising or services involves in this)*

5. ***HR Management System** – A platform to onboard employees, manage employee records, leave requests, time and attendance tracking, and other HR functions.*
6. ***Payroll System** – A system for managing employee salaries, tax deductions, bonuses, and payment processing.*
7. ***ozomAccouting**- manage and track financial transactions, ensuring accurate record-keeping, compliance, and financial analysis. Calculate VAT returns, accept payments and send payments. Invoices.*
8. ***Setups and Hostings** - Jobhant Admim panel, HR Management system, Payroll system and **ozomAccounting** needs to be setup manually with the companies that partners ups with ozomcapital, companies will have their own domains and hosting servers setups*

This document ensures that all stakeholders understand the system's requirements and functionality to develop a seamless and integrated solution.

1.2 Scope

The multi-system platform is designed to support job hiring, employee management, human resources processes, and payroll management. The system will offer:

- ***Jobhant App**: Job postings, candidate applications,candidate profile managements , interview scheduling, and hiring workflow management.*
- ***Myozom App**: Employee self services, access to HR services, performance tracking, and internal company communication.*
- ***HR System**: Employee onboarding, Employee record keeping, attendance management, leave management, and compliance tracking.*
- ***Payroll System**: Salary calculations, tax processing, payment scheduling, PAYE compliance (Annual recon), and reporting. And SSC & VET compliance.*

All four systems will be interconnected to provide a streamlined experience, ensuring smooth transitions from candidate selection to employee management and payroll processing.

1.3 Definitions, Acronyms, and Abbreviations

- ***HR** – Human Resources*
- ***SRS** – Software Requirements Specification*
- ***UI** – User Interface*
- ***API** – Application Programming Interface*
- ***Payroll** – Employee salary processing system*
- ***Jobhant** – The job marketplace application*
- ***Myozom** – The employee management application*
- ***Candidate** – A job applicant using the Jobhant app*
- ***Employer** – A company posting jobs on Jobhant*

1.4 References

- *IEEE 830-1998: Software Requirements Specification Standard*
- *ISO/IEC 25010:2011 – Software Quality Model*
- *Payroll Compliance Regulations (Namibia)*
- *HR Best Practices for Employee Management*
-

1.5 Softwares and Hardware required to build this System

- ***Software Required***
    1. *Intellij*
    2. *Vscode*
    3. *Microsoft Visual Studio Code*
    4. *MySQL*
    5. *Android Studio*
    6. *Electric Mobile Studio*
    7. *draw.io*
- ***Hardware Required***
    1. *Intel i5, windows (16Ram)*
    2. *Monitors*
- ***Languages***
    1. *Flutter (Frontend)*
    2. *Java (Backend)*
    3. *Spring Boot (Backend)*

---

2. Overall Description

2.1 Product Perspective

This system integrates multiple applications into a single ecosystem for recruitment, employee management, HR operations, and payroll processing. The applications will share a centralized database to ensure seamless data exchange.

2.2 Product Functions

- ***Jobhant App***
    - *Job posting and searching*
    - *Candidate applications and tracking*
    - *Candidate Profile managements*
    - *Likes, followers*
    - *Interview scheduling*

    - *Hiring workflow automation*
- ***Myozom App***
    - *HR service requests*
    - *Attendance and leave tracking*
    - *Employee performance monitoring*
    - *Announcements*
    - *Project management*
    - *Tasks*
    - *Budgeting and track expenses*
    - *Apply for loan e.t.c*
- ***HR Management System***
    - *Employee records management*
    - *Employee onboarding*
    - *Leave and attendance management*
    - *HR policy compliance tracking*
    - *Performance appraisals*
    - *And many more activities*
- ***Payroll System***
    - *Salary calculations and deductions*
    - *Tax and benefits processing*
    - *Automated payroll reports*
    - *Payment disbursement*
    - *Payslip distributions*

2.3 User Characteristics

Jobhant App Users:

- ***Job Seekers****: Individuals applying for jobs via the Jobhant app.*
- ***Employers****: Companies managing job postings and hiring.*
- ***Recruiters****: HR professionals or agencies responsible for screening and shortlisting candidates.*
- ***System Administrators****: Users managing the Jobhant platform via the admin panel, overseeing job postings, employer activities, and ensuring compliance. Monitor both candidates and companies.*
- ***Master Admin:*** *Manage Companies and Candidate profiles, jobs listings*

Myozom App Users:

- ***Employees****: Hired candidates who transitioned from Jobhant to Myozom.*

HR Management System Users:

- ***HR Administrators****: Employees managing records and HR functions.*
- ***Team Leaders****: Overseeing employee schedules, leave requests, and work performance.*
- ***Executives****: High-level decision-makers accessing employee analytics and reports.*
- ***Supervisors/Managers****: Users monitoring employees' attendance, leave requests, and*

performance evaluations within the HR system.

Payroll System Users:

- ***Finance/Payroll Staff****: Users handling employee salaries, tax processing, and payment disbursement.*
- ***Auditors****: Users reviewing financial transactions and ensuring compliance with company policies.*
- ***System Administrators****: Ensuring payroll calculations and reports are accurate and properly maintained.*
-

2.4 Constraints

- *Must comply with **Namibia's labor laws** and **payroll regulations**.*
- *Should support **multi-device access** (web and mobile).*
- *Data security must be ensured with **encryption and authentication mechanisms**.*
- *System should be **scalable** to support future expansions.*

2.5 Assumptions and Dependencies

- *The system will be **cloud-hosted** with a secure database.*
- *It will rely on **third-party payment APIs** for payroll processing.*
- *Employers must verify job postings to ensure authenticity.*

3. Specific Requirements

*[This section of the **SRS** should contain all the software requirements to a level of detail sufficient to enable designers to design a system to satisfy those requirements, and testers to test that the system satisfies those requirements. When using use-case modeling, these requirements are captured in the Use-Cases and the applicable supplementary specifications. If use-case modeling is not used, the outline for supplementary specifications may be inserted directly into this section, as shown below.]*

3.1 Functionality

*[This section describes the functional requirements of the system for those requirements which are expressed in the natural language style. For many applications, this may constitute the bulk of the **SRS**Package and thought should be given to the organization of this section. This section is typically organized by feature, but alternative organization methods may also be appropriate, for example, organization by user or organization by subsystem. Functional requirements may include feature sets, capabilities, and security.*

Where application development tools, such as requirements tools, modeling tools, etc., are employed to capture the functionality, this section document will refer to the availability of that data, indicating the location and name of the tool that is used to capture the data.]

3.1.1 <Functional Requirement One>

[The requirement description.]

3.2 Usability

[This section should include all of those requirements that affect usability. For example,

- specify the required training time for a normal users and a power user to become productive at

particular operations

- specify measurable task times for typical tasks or base the new system's usability requirements on other systems that the users know and like

- specify requirement to conform to common usability standards, such as IBM's CUA standards Microsoft's GUI standards]

3.2.1 <Usability Requirement One>

[The requirement description goes here.]

3.3 Reliability

[Requirements for reliability of the system should be specified here. Some suggestions follow:

- Availability-specify the percentage of time available (xx.xx%), hours of use, maintenance access, degraded mode operations, etc.

- Mean Time Between Failures (MTBF) - this is usually specified in hours, but it could also be specified in terms of days, months or years.

- Mean Time To Repair (MTTR)-how long is the system allowed to be out of operation after it has failed?

- Accuracy-specify precision (resolution) and accuracy (by some known standard) that is required in the system's output.

- Maximum Bugs or Defect Rate-usually expressed in terms of bugs per thousand of lines of code (bugs/KLOC) or bugs per function-point(bugs/function-point).

- Bugs or Defect Rate-categorized in terms of minor, significant, and critical bugs: the requirement(s) must define what is meant by a "critical" bug; for example, complete loss of data or a complete inability to use certain parts of the system's functionality.]

3.3.1 <Reliability Requirement One>

[The requirement description.]

3.4 Performance

[The system's performance characteristics should be outlined in this section. Include specific response times. Where applicable, reference related Use Cases by name.

- response time for a transaction (average, maximum)

- throughput, for example, transactions per second

- capacity, for example, the number of customers or transactions the system can accommodate

- degradation modes (what is the acceptable mode of operation when the system has been degraded in some manner)

- resource utilization, such as memory, disk, communications, etc.

3.4.1 <Performance Requirement One>

[The requirement description goes here.]

3.5 Supportability

[This section indicates any requirements that will enhance the supportability or maintainability of the system being built, including coding standards, naming conventions, class libraries, maintenance access, maintenance utilities.]

3.5.1 <Supportability Requirement One>

[The requirement description goes here.]

3.6 Design Constraints

[This section should indicate any design constraints on the system being built. Design constraints represent design decisions that have been mandated and must be adhered to. Examples include software languages, software process requirements, prescribed use of developmental tools, architectural and design constraints, purchased components, class libraries, etc.]

3.6.1 <Design Constraint One>

[The requirement description goes here.]

3.7 On-line User Documentation and Help System Requirements

[Describes the requirements, if any, for on-line user documentation, help systems, help about notices, etc.]

3.8 Purchased Components

[This section describes any purchased components to be used with the system, any applicable licensing or usage restrictions, and any associated compatibility and interoperability or interface standards.]

3.9 Interfaces

[This section defines the interfaces that must be supported by the application. It should contain adequate specificity, protocols, ports and logical addresses, etc. so that the software can be developed and verified against the interface requirements.]

3.9.1 User Interfaces

[Describe the user interfaces that are to be implemented by the software.]

3.9.2 Hardware Interfaces

[This section defines any hardware interfaces that are to be supported by the software, including logical structure, physical addresses, expected behavior, etc.]

3.9.3 Software Interfaces

*[This section describes software interfaces to other components of the software system. These may be purchased components, components reused from another application or components being developed for subsystems outside of the scope of this **SRS** but with which this software application must interact.]*

3.9.4 Communications Interfaces

[Describe any communications interfaces to other systems or devices such as local area networks, remote serial devices, etc.]

3.10 Licensing Requirements

[Defines any licensing enforcement requirements or other usage restriction requirements that are to be exhibited by the software.]

3.11 Legal, Copyright, and Other Notices

[This section describes any necessary legal disclaimers, warranties, copyright notices, patent notice, wordmark, trademark, or logo compliance issues for the software.]

3.12 Applicable Standards

[This section describes by reference any applicable standard and the specific sections of any such standards which apply to the system being described. For example, this could include legal, quality and regulatory standards, industry standards for usability, interoperability, internationalization, operating system compliance, etc.]

4. Supporting Information

*[The supporting information makes the **SRS** easier to use. It includes:*

- Table of contents

- Index

- Appendices

*These may include use-case storyboards or user-interface prototypes. When appendices are included, the **SRS** should explicitly state whether or not the appendices are to be considered part of the requirements.]*