Software Requirements Specification

For

Newspaper Agency Automation Software

-Version 1.0 Approved Prepared By

Telu Amrutha Varshini(21CS30055)
Parala Siva Sai Yeswanth(21CS30034)

Gottipalli Sri Vishnu Satwik(21CS10027)

1. Introduction

1.1 Purpose

It is proposed to develop software for the newspaper agency automation software(NAAS) specifically for the Master newspaper. This software provides data and daily updates to users like delivery agents, managers, and customers. This software works as an accountant in the newspaper agency industry.

1.2 Document Conventions

MNAAS: Masters Newspaper agency automation software
Manager: The person who supervises the MNAAS software.
Delivery Agent: The person who delivers the newspapers and magazines door to door.
Customer: The person who purchases the newspaper.

1.3 Intended Audience and Reading Suggestions

This document contains the requirements and specifications for this project and is meant for project developers, testers, and users. This paper provides a detailed explanation of the software's numerous functional and non-functional needs.

1.4 Product Scope

NAAS has the following benefits over a current manual newspaper system:

1. It offers a shared platform for managers, delivery agents and customers.

2. Time-saving and straightforward to use.

3. Lower potential for human error.

4. An intuitive interface.

5. Information can be retrieved quickly, easily, and securely.

1.5 References

- Given the SRS document template.
- IEEE. IEEE Std 830–1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.

2. Overall Description

2.1 Product Perspective

This software is a new, self-contained product that gives the newspaper agency a management system. The NMS is developed as a replacement for the work previously done on paper and manually. This project makes it easy for customers, delivery agents, and managers to do their jobs.

2.2 Product Functions

Our product function includes:

- Manager sign-in
- Delivery Agent Login
- Customer login
- Details of newspapers and customers' data
- Keeping records of financial information
- Manager/ Deliverer login
- Add Publication
- Edit publication details
- Add customer
- Edit customer details
- Publications to be delivered
- Delivery summary
- Customer bills
- With-Hold subscription

- Payment receipt
- Deliverer payment

2.3 User classes and characteristics

2.3.1 Manager

- Login with ID and password
- Can add and access details of staff and customers and newspapers.
- Estimate the financial status of the agency
- Generates monthly bills for customers and staff.

2.3.2 Delivery Agent

- Login with ID and password
- Can access information about the addresses to be delivered.
- Can view monthly salary details

2.3.3 Customer

- Register with required credentials
- Login with ID and password
- Can change the subscription list
- Can view the bill status

2.4 Operating environment

Python and virtual environment is required.

NAAS runs smoothly on any personal computer with Linux, Windows, or MacOS with an internet connection.

2.5 Design and Implementation Constraints

The software runs on any device with windows/Linux/macOS Operating System and that supports Python 3. The interface language is English.

2.6 User Documentation

A simple register/login guide is provided.
For further queries, customers can mail to the given service mail ID.

2.7 Assumptions and Dependencies
Python environment is installed and setup in the system

3. External Interface Requirements
3.1 User interface
The user interface for the system must be compatible to windows ,linux and MacOS.

3.2 Hardware Interfaces
Personal class computers are capable of running all components of the software. An Internet connection is a prerequisite component.

3.3 Software Interfaces
- **OS-** Windows, Linux, MacOS
- Python environment (SQL)
- HTML -user interface
- Django or Flask to interlink front end and backend

3.4 Communication Interfaces
Communication between software and users is through valid IDs assigned during registration.
Communication between users is via software and further can contact through the Mail(email-ID is provided).

4. System Features
14.1 Function 1 Manager/ Deliverer login
Input -Username, Password

Processing -Check if there exists a user in the database with the

entered credentials.

Output -The user is directed to his main page if the entered credentials are correct, else the system will ask the user to enter his correct credentials.

Function 2- Add Publication

Input Newspaper name, language, description, price

Processing- A new record with the above details is inserted into the Database.

Output- Notify the user whether the data is successfully inserted or not.

Function 3- Add customer

Input- Customer name, address, phone, subscription

Processing- A new record with the above details is inserted into the database.

Output- Notify the user whether the data is successfully inserted or not.

Function 4- Publications to be delivered

Input Customer address, Customer subscriptions, Deliverers

Processing Each deliverer is assigned an address to which the subscribed publication must be delivered. The addresses are assigned in such a way that the communication of deliverer is minimal.

Output Print the publications to be delivered.

Function 5- Delivery summary

Input -Customers, subscribed publications

Processing- Check which publications the customers have received

for the whole month.

Output- Print summary of the delivered publication details in a
month.

Function 10- Payment Receipt

Input -Customer id, paid amount, cheque number

Processing- Update the database about the payment.

Output- Print the payment receipt of the customer.

Function 11 Deliverer payment

Input Deliverers, deliveries made

Processing Calculate the amount a deliverer should receive by
adding 2.5% of the value of the publications delivered by
him.

Output Print the amount payable to each delivery boy.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

Operating System: Windows/Linux/MacOS.

Coding: Python

Performance: It should perform according to the functional requirements.

Internet connection: a stable connection is sufficient.

5.2 Safety Requirements
The users must remember their usernames and passwords since there is no facility for recovering forgotten passwords and usernames. The software does no harm to the device. The system used to implement also must not contain any viruses.

5.3 Security Requirements
Any user can enter the system only when they have the correct login credentials as per the encrypted data stored in the database.

5.4 Software Quality Attributes

● Correctness: This system should satisfy the demands of the user precisely to fulfill the user's objectives.

● Efficiency: Enough resources to be implemented to achieve a particular task efficiently without any hassle.

● Portability: The system should run in any of the Windows/Linux/MacOS environments.

● Usability: The system should provide a user manual to every level of users.

● Testability: The system should be able to be tested to confirm the performance and all kinds of user specifications.

● Maintainability: The system should be maintainable.

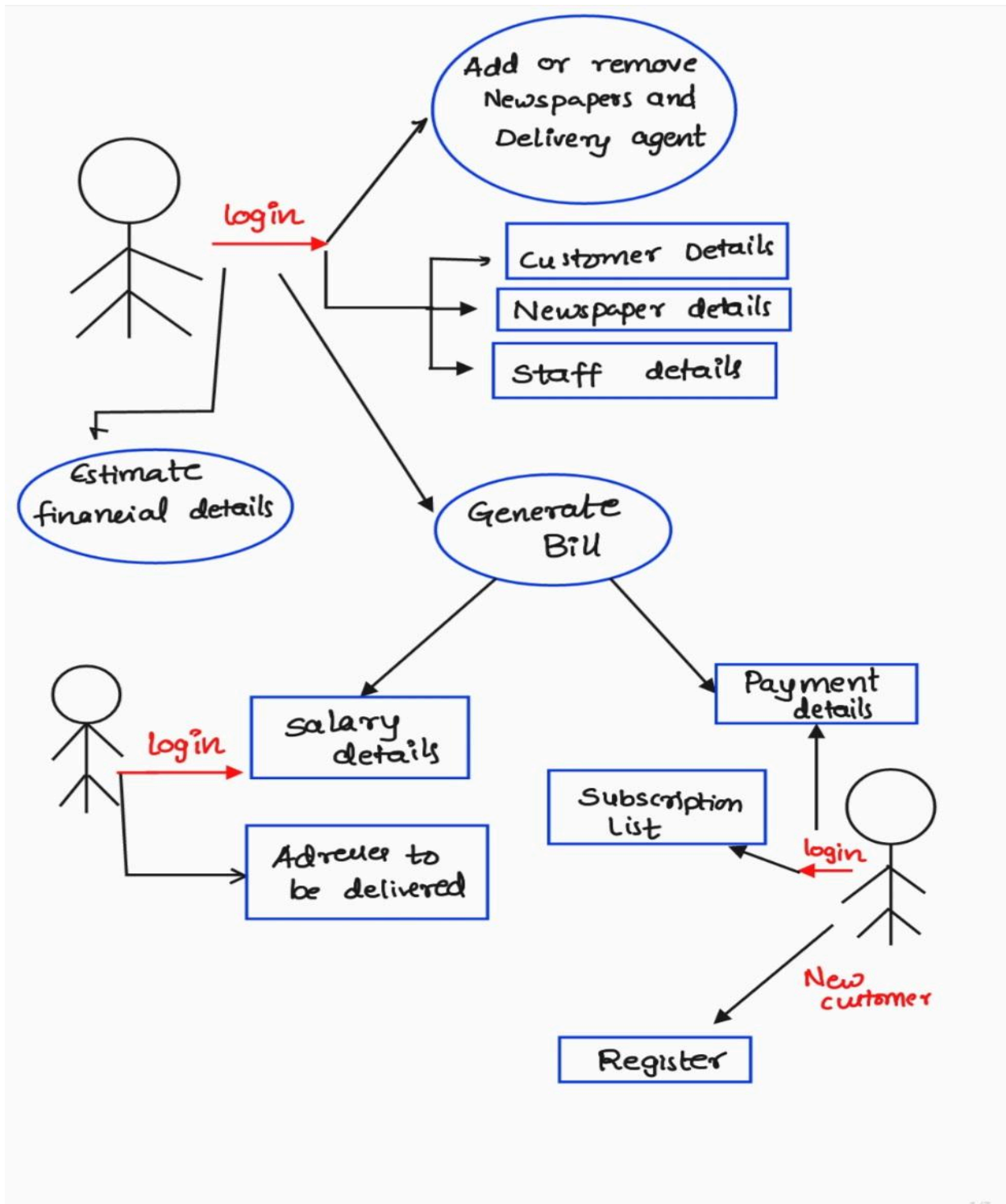● Adaptability: Software is understandable and any type of user can access it easily

5.5 Business Rules

1. The software will be free of cost for all kinds of stakeholders.

2. Only the Manager can add or remove delivery agents and newspapers.

3. Only the manager can generate the bill for the customer and delivery agent.

4. Only customers can change the subscription list of a customer.

6. Other requirements
Appendix A: Analysis method
Class Diagram

Class Diagram

User

+ User_id: integer

+ user_name: string

Manager

+ add Publication( )

+ add delivery agent( )

+ Bill generation( )

+ summary( )

+ due Remainder( )

Delivery Agent

+ Allocate Area( )

+ Salary details( )

Customers

+ customer id: int

+ customer name: string

+ address: string

+ phone_no: string

+ email id: int

+ subscription: string

+ Due: int

+ Amount: int

Publication

+ paper-name: string

+ Price: integer

+ paper_id: integer