###### A

Mini Project On

#### AUTOMATED MEDICINE REPOSITORY

(Submitted in partial fulfillment of the requirements for the award of Degree)

BACHELOR OF TECHNOLOGY

In

###### COMPUTER SCIENCE AND ENGINEERING

By

###### AKUNURI ARUN DEEPAK(197R1A05J2)

Under the Guidance of

**J. NARASIMHARAO**

(Associate Professor)



##### DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

**CMR TECHNICAL CAMPUS UGC AUTONOMOUS**

(Accredited by NAAC, NBA, Permanently Affiliated to JNTUH, Approved by AICTE, New Delhi) Recognized Under Section 2(f) & 12(B) of the UGCAct.1956, Kandlakoya (V), Medchal Road, Hyderabad-501401.

**2019-2023**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**



#### CERTIFICATE

This is to certify that the project entitled **“AUTOMATED MEDICINE REPOSITORY”** being submitted by **AKUNURI ARUN DEEPAK (197R1A05J2)** in

partial fulfillment of the requirements for the award of the degree of B.Tech in Computer Science and Engineering to the Jawaharlal Nehru Technological University Hyderabad, is a record of bonafide work carried out by them under our guidance and supervision during the year 2022-23.

The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree or diploma.

**J. Narasimharao Dr. A. Raji Reddy**

(Associate Professor) DIRECTOR

INTERNAL GUIDE

**Dr. K. Srujan Raju EXTERNAL EXAMINER**

HOD

**Submitted for viva voice Examination held on**

##### ACKNOWLEDGEMENT

Apart from the efforts of us, the success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project.

We take this opportunity to express my profound gratitude and deep regard to my guide **J. Narasimharao,** Associate Professor for his exemplary guidance, monitoring and constant encouragement throughout the project work. The blessing, help and guidance given by him shall carry us a long way in the journey of life on which we are about to embark.

We also take this opportunity to express a deep sense of gratitude to the Project Review Committee (PRC) **Dr. Punyaban Patel, Ms. Shilpa, Dr. M . Subha Mastan Rao & J. Narasimharao** for their cordial support, valuable information and guidance, which helped us in completing this task through various stages.

We are also thankful to **Dr. K. Srujan Raju,** Head, Department of Computer Science and Engineering for providing encouragement and support for completing this project successfully.

We are obliged to **Dr. A. Raji Reddy,** Director for being cooperative throughout the course of this project. We also express our sincere gratitude to Sri. **Ch. Gopal Reddy,** Chairman for providing excellent infrastructure and a nice atmosphere throughout the course of this project.

The guidance and support received from all the members of **CMR Technical Campus** who contributed to the completion of the project. We are grateful for their constant support and help.

Finally, we would like to take this opportunity to thank our family for their constant encouragement, without which this assignment would not be completed. We sincerely acknowledge and thank all those who gave support directly and indirectly in the completion of this project.

A.ARUN DEEPAK (197R1A05J2)

##### ABSTRACT

The aim of the project is to improve self-dependent health care for patients who need to depend on themselves for their daily health needs through technology.

With many patients and householders taking regular medicine which doesn’t require any formal prescription, this solution will be helpful to those people.

The main working of project has a cloud service provider through which we deploy our application to run the result. The project involves the usage of cloud based API automation through Django framework which is embedded to the cloud infrastructure API. This allows in easy-to-go format of working of the messaging system delivered via mobile notification service provided by the AWS cloud infrastructure.

##### FIGURE NO FIGURE NAME PAGE NO

###### Figure 3.1 AMR Solution Architecture 7

###### Figure 3.2 Use Case Diagram 8

Figure 3.3 Class Diagram 10

Figure 3.4 Sequence Diagram 11

Figure 3.5 Activity Diagram 12

|  |  |  |
| --- | --- | --- |
| **SCREENSHOT NO.** | **SCREENSHOT NAME** | **PAGE NO**. |
| Screenshot 5.1 | Medicine Repository Main Page | 16 |
| Screenshot 5.2 | SNS Log | 16 |
| Screenshot 5.3 | Message Output SNS | 17 |
|  |  |  |
|  |  |  |

ABSTRACT i

LIST OF FIGURES ii

LIST OF SCREENSHOTS iii

1. [INTRODUCTION 1](#_TOC_250034)
   1. [PROJECT SCOPE 1](#_TOC_250033)
   2. [PROJECT PURPOSE 1](#_TOC_250032)
   3. [PROJECT FEATURES 1](#_TOC_250031)
2. [SYSTEM ANALYSIS 2](#_TOC_250030)
   1. [PROBLEM DEFINITION 2](#_TOC_250029)
   2. [EXISTING SYSTEM 2](#_TOC_250028)
      1. LIMITATIONS OF THE EXISTING SYSTEM 3
   3. [PROPOSED SYSTEM 3](#_TOC_250027)
      1. ADVANTAGES OF PROPOSED SYSTEM 3
   4. [FEASIBILITY STUDY 4](#_TOC_250026)
      1. [ECONOMIC FEASIBILITY 4](#_TOC_250025)
      2. [TECHNICAL FEASIBILITY 5](#_TOC_250024)
      3. SOCIAL FEASIBILITY 5
   5. [HARDWARE & SOFTWARE REQUIREMENTS 5](#_TOC_250023)
      1. [HARDWARE REQUIREMENTS 5](#_TOC_250022)
      2. [SOFTWARE REQUIREMENTS 6](#_TOC_250021)
3. [ARCHITECTURE 7](#_TOC_250020)
   1. [PROJECT ARCHITECTURE 7](#_TOC_250019)
   2. [DESCRIPTION 7](#_TOC_250018)
   3. [USE CASE DIAGRAM 8](#_TOC_250017)
   4. [CLASS DIAGRAM 9](#_TOC_250016)
   5. [SEQUENCE DIAGRAM 10](#_TOC_250015)
   6. [ACTIVITY DIAGRAM 11](#_TOC_250014)
4. [IMPLEMENTATION 12](#_TOC_250013)
   1. [SAMPLE CODE 12](#_TOC_250012)
5. SCREENSHOTS 16
6. [TESTING 19](#_TOC_250011)
   1. [INTRODUCTION TO TESTING 19](#_TOC_250010)
   2. [TYPES OF TESTING 19](#_TOC_250009)
      1. [UNIT TESTING 19](#_TOC_250008)
      2. [INTEGRATION TESTING 20](#_TOC_250007)
      3. [FUNCTIONAL TESTING 20](#_TOC_250006)
   3. [TEST CASES 21](#_TOC_250005)
      1. [CLASSIFICATION 21](#_TOC_250004)
7. CONCLUSION & FUTURE SCOPE 22
   1. [PROJECT CONCLUSION 22](#_TOC_250003)
   2. [FUTURE SCOPE 22](#_TOC_250002)
8. REFERENCES 23
   1. [REFERENCES 23](#_TOC_250001)
   2. [GITHUB LINK 23](#_TOC_250000)

# INTRODUCTION

#### INTRODUCTION

##### PROJECT SCOPE

This project is titled “Automated Medicine Repository”. Traditional healthcare repository and inventory methods based on manual and API based servicing have caused a major problem to both healthcare companies and on the personal customers. By choosing to automate the entire inventory wherein the data can be fetched from a DB or a file residing on a local or cloud server without having to interfere manually.

##### PROJECT PURPOSE

This project has been developed to identify cloud based API –automation methods that can perform large scale data manipulation Cloud computing technology is making crucial advances in solving problems that have restricted the best attempts of the business solutions community for many years. It has proven to be excellent at designing scalable solutions to meet challenging business requirements without having to worry about the upfront capital expenses.

##### PROJECT FEATURES

The main features of this project are that this solution performs an automation on the target server to generate requested information. It addresses the problem of data integration and API scalability with a service-oriented-architecture. It has mainly beaten records in IT Infrastructure, Cloud Automation, DevOps based Development and many other real world scenarios. There are many such IT tools such as Hybrid-cloud infrastructure, MuleSoft API for Automation etc., .

## SYSTEM ANALYSIS

##### SYSTEM ANALYSIS

**SYSTEM ANALYSIS**

System Analysis is the important phase in the system development process. The System is studied to the minute details and analyzed. The system analyst plays an important role of an interrogator and dwells deep into the working of the present system. In analysis, a detailed study of these operations performed by the system and their relationships within and outside the system is done. A key question considered here is, “what must be done to solve the problem?” The system is viewed as a whole and the inputs to the system are identified. Once analysis is completed the analyst has a firm understanding of what is to be done.

##### PROBLEM DEFINITION

A general statement of medicine repository can be formulated as the given database of information which needs to be informed at regular intervals with accurate data and no misinformation needs to be given to patients or healthcare workers relying on this software.

.

##### EXISTING SYSTEM

Initially, medicine repository systems focused on manual data extraction and verification, although developments in automation have made significant advancements. Data Automation offers the advantage of using a scalable API to extract complex information. However, data extraction poses some serious problems: need to refine the data results to obtain relevant data at the exact time intervals.

* + 1. DISADVANTAGES OF EXISTING SYSTEM

Following are the disadvantages of existing system:

* Fault tolerant
* Low bandwidth
* High Latency
* No data duplication on a backup system allowing for BCP-DR
* Increase in upfront capital expenses rather than operational expenses.

##### PROPOSED SYSTEM

This project presents a system that allows the medicine information to be notified to the patients or healthcare professionals about the medicine information, it’s generic composition, date of expiry, number of doses prescribed and number of doses taken or to be taken. The entire information is present on a database server located on the cloud. This data is then taken through an API and then forwarded to the end user through SMS available through the cloud service provider. This allows in automating the entire solution through cloud with the help of service-oriented-architecture model wherein we use minimal cloud resources.

* + 1. ADVANTAGES OF THE PROPOSED SYSTEM
* Cloud based system ensuring zero pint of failure.
* Eliminating manual touch points in the system
* Automating the messaging architecture.
* HIPAA compliant solution.

##### FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and a business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. Three key considerations involved in the feasibility analysis:

* Economic Feasibility
* Technical Feasibility
* Social Feasibility
  + 1. ECONOMIC FEASIBILITY

The developing system must be justified by cost and benefit. Criteria to ensure that effort is concentrated on a project, which will give best, return at the earliest. One of the factors, which affect the development of a new system, is the cost it would require.

The following are some of the important financial questions asked during preliminary investigation:

* + - * The costs conduct a full system investigation.
      * The cost of the hardware and software.
      * The benefits in the form of reduced costs or fewer costly errors.

Since the system is developed as part of project work, there is no manual cost to spend for the proposed system. Also all the resources are already available, it give an indication that the system is economically possible for development.

* + 1. TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

* + 1. BEHAVIORAL FEASIBILITY

This includes the following questions:

* + - * Is there sufficient support for the users?
      * Will the proposed system cause harm?

The project would be beneficial because it satisfies the objectives when developed and installed. All behavioral aspects are considered carefully and conclude that the project is behaviorally feasible

##### HARDWARE & SOFTWARE REQUIREMENTS

* + 1. HARDWARE REQUIREMENTS:

Hardware interfaces specify the logical characteristics of each interface between the software product and the hardware components of the system. The following are some hardware requirements.

* + - * Processor : Intel Dual Core I5 and above
      * Hard disk : 8GB and above
      * RAM : 8GB and above
      * Input devices: Keyboard, mouse.

##### SOFTWARE REQUIREMENTS:

Software Requirements specifies the logical characteristics of each interface and software components of the system. The following are some software requirements,

* Operating system : Windows 8 and above
* Languages : Python, Html, CSS
* Tools : Python IDEL3.7 version, Anaconda - Jupyter, Spyder
* Framework: Django v4.1.3
* AWS- LightSail, RDS(MySQL), API Gateway, Route53.

## ARCHITECTURE

##### 3. ARCHITECTURE

##### AMR SOLUTION ARCHITECTURE

This project architecture shows the procedure followed for classification, starting from input to final output.

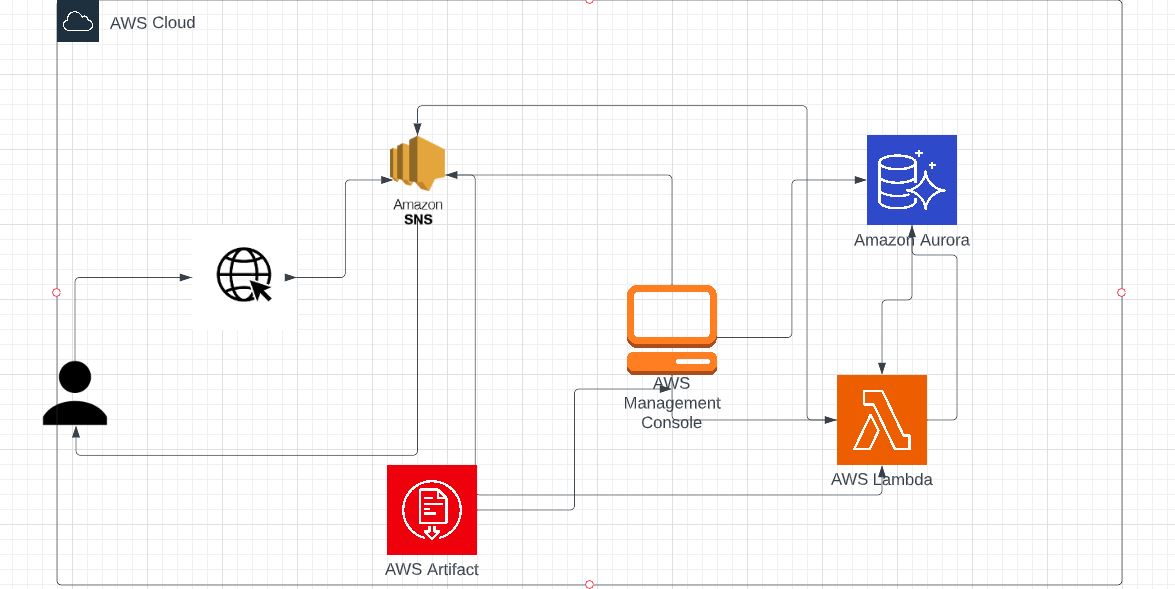


Figure 3.1: AMR SOLUTION ARCHITECTURE

3.2 DESCRIPTION

This project is totally based upon automating the data extraction process used in enterprise grade system. The system is built to automate messaging system as part of the medicine inventory solution and then produce a SMS notification for every Medicine details delivered. The model is built using Django framework. The data extraction is built using python based database API which are embedded to the cloud API gateway which enables it’s scalability for business requirements. Every API call is securely monitored to ensure HIPAA compliance throughout the system making it reliable for information and data sensitive countries like Europe and United Kingdom.

3.3 USE CASE DIAGRAM

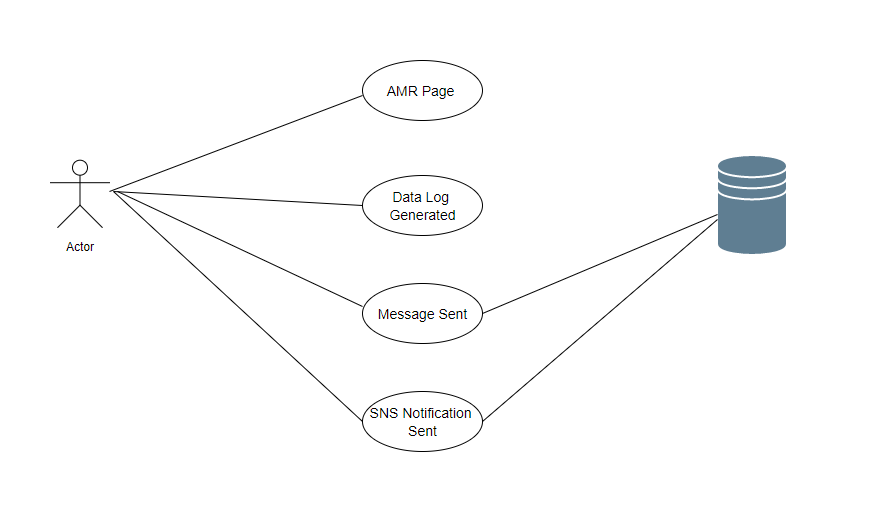
A use case diagram is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well. The use cases are represented by either circles or ellipses. The actors are often shown as stick figures. 

Figure 3.2: Use Case Diagram for Automated Medicine Repository

##### 3.4 CLASS DIAGRAM

Class diagram is a type of static structure diagram that describes the structure of a system by showing the system’s classes, their attributes, operations (or methods), and the relationships among objects.

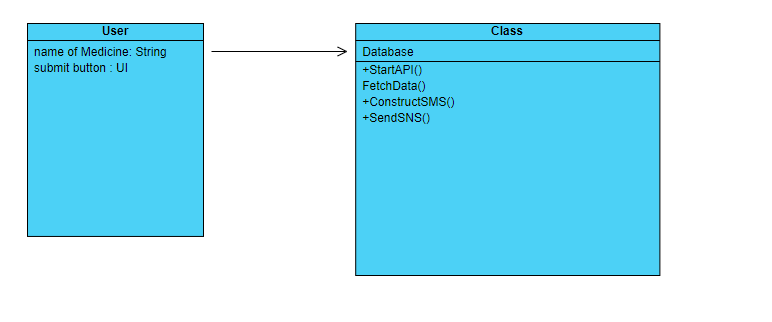


Figure 3.3: Class Diagram for Automated Medicine Repository

3.5 SEQUENCE DIAGRAM

A sequence diagram is a type of interaction diagram because it describes how—and in what order—a group of objects works together. These diagrams are used by software developers and business professionals to understand requirements for a new system or to document an existing process.

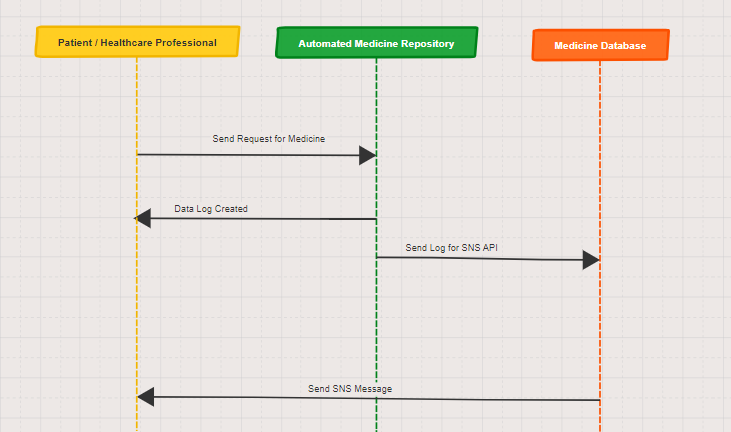


Figure 3.4: Sequence Diagram for Automated Medicine Repository

**3.6 ACTIVITY DIAGRAM**

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes (i.e., workflows), as well as the data flows intersecting with the related activities. Although activity diagrams primarily show the overall flow of control, they can also include elements showing the flow of data between activities through one (or) more data stores.

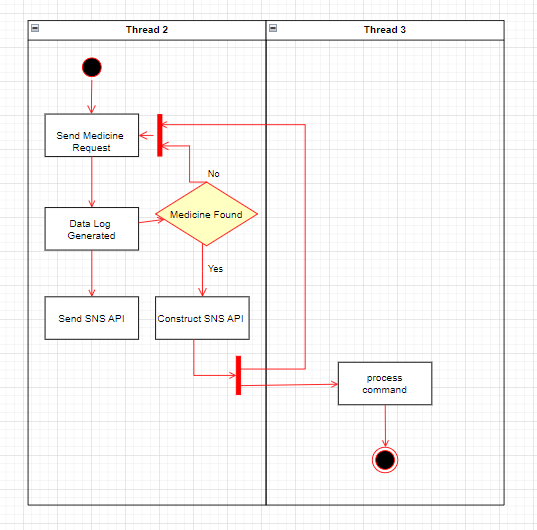


Figure 3.5: Activity Diagram for Automated Medicine Repository

**4. IMPLEMENTATION**

##### 4.1 SAMPLE CODE

data = Medicine.objects.all()

med = {

"medicine\_number": data

}

return render\_to\_response("login/profile.html", med)

// in html file :

{% for student in student\_number %}

{{ medicine.number}}

{{ medicine.name}}

{% endfor %}

# -\*- coding: utf-8 -\*-

"""

Created on Thu Nov 3 10:20:01 2022

@author: Akunuri Arun Deepak

"""

INSTALLED\_APPS = [

'django.contrib.admin',

'django.contrib.auth',

'django.contrib.contenttypes',

'django.contrib.sessions',

'django.contrib.messages',

'django.contrib.staticfiles',

# films app:

'films.apps.FilmsConfig',

# add this:

'django\_extensions',

]

from films.models import Film, Genre

import csv

def run():

with open('Medicines.csv') as file:

reader = csv.reader(file)

next(reader) # Advance past the header

Film.objects.all().delete()

Genre.objects.all().delete()

for row in reader:

print(row)

genre, \_ = Genre.objects.get\_or\_create(name=row[-1])

film = Film(title=row[0],

year=row[2],

genre=genre)

film.save()

// SNS API

import boto3

client = boto3.client('sns')

response = client.add\_permission(

TopicArn='string',

Label='string',

AWSAccountId=[

'string',

],

ActionName=[

'string',

]

)

</div>

<form action="LogPage.html">

<label for="fname">Medicine Name</label><br>

<input type="text" id="med\_name" name="Medcine Name"><br>

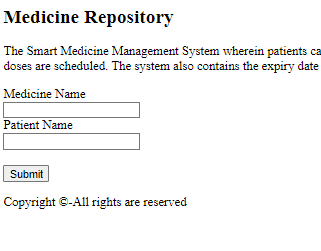
<label for="Patient Name">Patient Name</label><br>

<input type="text" id="pat\_name" name="Patient\_Name"><br><br>

<input type="submit" value="Submit">

</form>

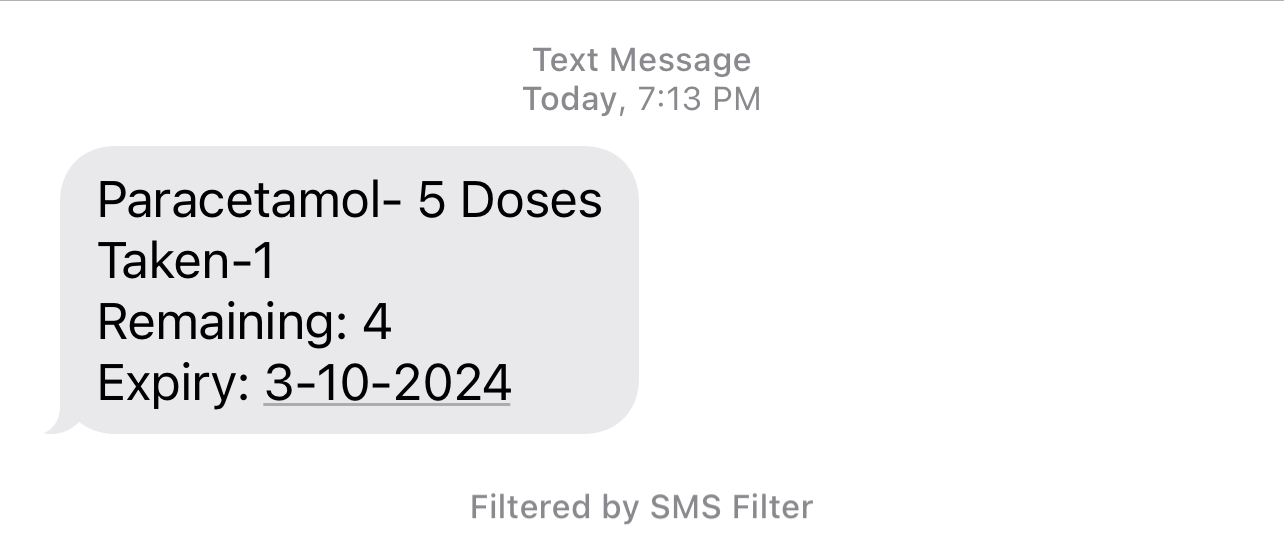
**5. RESULTS**



Screenshot 5.1: AMR Main Page



Screenshot 5.2: Message Log Sent to Server



Screenshot 5.3: Medicine Output Generated

## 6. TESTING

#### 6. TESTING

##### INTRODUCTION TO TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, subassemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

##### TYPES OF TESTING

* + 1. UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .It is done after the completion of an individual unit before integration. This is a structural testing that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

* + 1. INTEGRATION TESTING

Integration tests are designed to test integrated software components to determine if they actually run as one program. Integration tests demonstrate that although the components were individually satisfactory, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

* + 1. FUNCTIONAL TESTING

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input: Identified classes of valid input must be accepted.

Invalid Input

: Identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked. Organization and preparation of functional tests is focused on requirements, key functions, or special test cases.

##### TEST CASES

* + 1. CLASSIFICATION

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test case ID | Test case name | Purpose | Input | Output |
| 1 | Medicine Information | To provide exact Output of the Medicine | The user gives the input in the form of a text | An output is SNS using Messaging service. |
| 2 | Medicine Automation | To automate ensuring zero touch points | The patient sets the number of doses to be taken | An output is SNS notifications generated to deliver medicine information. |

**7. CONCLUSION**

##### 7. CONCLUSION & FUTURE SCOPE

##### PROJECT CONCLUSION

The potential benefit of using medicine repository is that any large inventory of medicine can be managed by using automation software which has automation capabilities ensuring zero touch point technique eliminating use of manual data extraction and manipulation. This project has zero infrastructure costs and minimal downtime ensuring resiliency and low latency for remote geophysical locations. Timely upgrade and patches without affecting data transfer allows the users to be confident about the software data.

##### FUTURE SCOPE

The potential idea behind the idea of the project is the additional feature, “automation though cloud”. This feature allows users to implement the software and allow the software to run directly on the cloud incurring less resources and cost, eliminating touch points to manually extract data. The data can be delivered to the phone allowing more easy access to the data while on the go.

### 8. BIBLIOGRAPHY

##### 8. BIBLIOGRAPHY

##### REFERENCES

[1] N. Yadav and S. P. Panda, "A Path Forward for Automation in Robotic Process Automation Projects: Potential Process Selection Strategies," *2022 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COM-IT-CON)*, 2022, pp. 801-805, doi: 10.1109/COM-IT-CON54601.2022.9850739.

[2] M. Pau, M. Mirz, J. Dinkelbach, P. Mckeever, F. Ponci and A. Monti, "A Service Oriented Architecture for the Digitalization and Automation of Distribution Grids," in IEEE Access, vol. 10, pp. 37050-37063, 2022, doi: 10.1109/ACCESS.2022.3164393.

[12] M. Pau, M. Mirz, J. Dinkelbach, P. Mckeever, F. Ponci and A. Monti, "A Service Oriented Architecture for the Digitalization and Automation of Distribution Grids," in IEEE Access, vol. 10, pp. 37050-37063, 2022, doi: 10.1109/ACCESS.2022.3164393.

##### GITHUB LINK

<https://github.com/Arun200207/Automated-Medicne-Repository.git>