 

HINDUSTAN AERONAUTICS LIMITED KORAPUT DIVISION, ODISHA

Project Report On

PHARMACY APPLICATION:(MEDIFLOW)

Submitted By:

Name: ADITYA DAS, ABHILASH DAS, G.HARIESH REDDY

Course: Bachelor in Technology in Computer Science and Engineering

Year & Semester: 3rd year 6th semester and 2ND Year 4thSemester

College Name: GIET UNIVERSITY AND NIST UNIVERSITY

Under The Guidance of

Mr RAHUL KUMAR

SENIOR MANAGER(IT)

Submitted To

TRAINING AND DEVELOPMENT INSTITUTE

HAL, KORAPUT

ACKNOWLEDGEMENT

I take this opportunity to express my sincere gratitude to Hindustan Aeronautics Limited, Sukhoi Engine Division, Koraput, Odisha having accorded approval to undertake this technical report in their organization.

I acknowledge my sincere thanks to Mr RAHUL KUMAR, SENIOR MANAGER(IT), for his cooperation and guidance during my vocational training.

Last but not the least, I would like to express my sincere and humble gratitude to all the staff of IT Dept. for their cooperation and help at various stages to make this project a success.

**ADITYA DAS, VT no:25-127  
ABHILASH DAS, VT no:25-128  
G.HARIESH REDDY, VT no.:25-131**

CERTIFICATE

This is to certify that ADITYA DAS from GIET UNIVERSITY, has successfully undergone one-month vocational training in HAL, SUKHOI Engine Division, Koraput, Odisha from 2nd JUNE to 30th JUNE.

He has taken keen interest in learning and completing the vocational training in HAL. During his training peiod, he was familiarized with various activities of the IT department.

I found him to be sincere & hardworking and his performance was very good during the training period. His character and conduct were satisfactory.

I wish him all the success in his future endeavors.

**Mr RAHUL KUMAR, Mr UTTAM CHANDRA,**

**SENIOR MANAGER(IT) IT MANAGER**

CERTIFICATE

This is to certify that ABHILASH DAS from NIST UNIVERSITY, has successfully undergone one-month vocational training in HAL, SUKHOI Engine Division, Koraput, Odisha from 2nd JUNE to 30th JUNE

He has taken keen interest in learning and completing the vocational training in HAL. During his training period, he was familiarized with various activities of the IT department.

I found him to be sincere & hardworking and his performance was very good during the training period. His character and conduct were satisfactory.

I wish him all the success in his future endeavors.

**Mr RAHUL KUMAR, Mr UTTAM CHANDRA,**

**SENIOR MANAGER(IT) IT MANAGER**

CERTIFICATE

This is to certify that G.HARIESH REDDY from NIST UNIVERSITY, has successfully undergone one-month vocational training in HAL, SUKHOI Engine Division, Koraput, Odisha from 2nd JUNE to 30thJUNE

He has taken keen interest in learing and completing the vocational training in HAL. During his training peiod, he was familiarized with various activities of the IT department.

I found him to be sincere & hardworking and his performance was very good during the training period. His character and conduct were satisfactory.

I wish him all the success in his future endeavors.

**Mr RAHUL KUMAR, Mr UTTAM CHANDRA,**

**SENIOR MANAGER (IT) IT MANAGER**

CONTENTS

* History of Hal
* MAIN.PY
* STYLE.CSS
* LOGIN.HTML
* REGISTER.HTML
* DASHBOARD\_STORE.HTML
* ADD MEDICINE.HTML
* VIEW MEDICINE.HTML
* MY ORDERS.HTML
* SEARCH MEDICINES.HTML
* DASHBOARD\_PUBLIC.HTML
* PROJECT DESCRIPTION AND RESULTS

HISTORY OF HAL

Hindustan Aeronautics Limited (HAL) was established in 1940 in Bangalore by visionary industrialist Walchand Hirachand with support from the Kingdom of Mysore and the British Government. Initially named Hindustan Aircraft Limited, it began by producing licensed aircraft for the British during World War II.

After India gained independence, the Government of India took over HAL in 1947, and it played a key role in developing India’s aerospace industry. Over the decades, HAL grew into a premier defense and aerospace company, producing indigenous aircraft like HF-24 Marut (India’s first jet fighter), and collaborating internationally to manufacture aircraft such as the MiG-21, Su-30MKI, and Hawk trainer jets.

In the 21st century, HAL became instrumental in developing advanced indigenous aircraft like the Light Combat Aircraft (LCA) Tejas, Advanced Light Helicopter (ALH) Dhruv, and Light Utility Helicopter (LUH). It also supports ISRO in satellite launches and space missions.

By 2025, HAL has transformed into a global aerospace player, contributing significantly to India’s defense self-reliance (Atmanirbhar Bharat) and exporting defense equipment. With ongoing projects like AMCA (fifth-generation fighter) and partnerships in UAVs and space systems, HAL continues to be the backbone of India's aerospace ambitions.

An agreement was signed in August, 1962 with the Soviet Union for manufacture of MIG-21 E7FL Air craft under license the Aero engine factory at Koraput(ODISHA), the Air frame Factory at Nasik(MAHARASHTA),and the Avionics Factory at Hyderabad(Andhra Pradesh) have been set up to meet this requirement on the name of Aeronautics India Limited which was formed on April 1964 and new company under the name of Hindustan Aeronautics Limited was formed.

The government sanction for the first phase of construct of the Aero Engine factory at Sunebeda (Koraput) was accorded March 1964 and the factory started manufacture of R11F2-Series-III engines for fitment on MIG21FL Aircraft from 1969 onwards. The first engines of imported category manufactured in December 1968 and various categories of engines were produced during the subsequent years. The first raw material engine was produced in February 1971. The production programmes for the factory also include manufacture of forging and casting required for MIG-Aircraft.

To meet the Air force requirement for improved fight interceptor aircraft, an agreement was signed with USSR in August 1976 for manufacturing MiG-21BIS Aircraft. The power plant of this aircraft is the R25 turbojet engine. The government approval for setting up capital facilities was accorded in October 1977. The first engine of imported category delivered to HAL Nasik Division in the year 1978-79. The FI raw material engine was delivered during January 1983.

With signing of the inter-governmental agreement for manufacture of MiG 27M Aircraft on 19th March 1982, this Division would be involved in the manufacture of R-29B series of engine from the year 1984-85.

In order to attain self-sufficiency and to avoid difficulties regarding supply of Raw Material & other layout items from USSR, it was decided to provide indigenous supply of spares manufacturing for Overhaul/maintenance of the fleet.

The Government approval for undertaking the tax received during 1977-78 and the indigenization plan was formed to tackle

ARS and first moving spares.

Metallic material.

Non-metallic material.

Ready-made articles.

HISTORY GROWTH AND OBJECTIVES OF THE EARLY YEARS

The origins of the present-day Hindustan Aeronautics Limited can be traced to the erstwhile Hindustan Aeronautics Limited, which was set up in December 1940 by a foreign industrialist the late Seth Walchand Hirachand in association with the government of the princely state of Mysore. The register of joint tock companies of Mysore State registered the company on 23rd December 1940 as private limited company.

The company was registered on 23rd December 1940 as private limited company by the register of joint stock companies of Mysore State. The company commenced its operations with the aim manufacturing the Harlow Trainer, Curtiess Haw Fighter the Vul Bomber in collaboration with the Inter Continentals Aircraft company USA. The first flight handed over to the government of India in August 1941. Aten seat glider designed by Dr. V.Ghatage was also test flown in August 1941.

There are various divisions of HAL located in different places in India, which are:-

1. Bangalore Division (Karnataka)

2. Barrackpore Division (West Bengal)

3. Nasik Division (Maharashtra)

4. Hyderabad Division (Telangana)

5. Lucknow Division (Uttar Pradesh)

6. Kanpur Division (Uttar Pradesh)

7. Korwa Division (Uttar Pradesh)

8. Koraput Division (Odisha)

CORPORATE OBJECTIVES

The objectives for which the company is established are out in the memorandum of Association of HAL. There are clauses which inter alia provide for design, development, manufacture, repair and overhaul of aircraft, engines, related materials, components and equipment.

Manufacture of Missiles and Weapons, to take or otherwise acquire and hold shares in other company having objective altogether or in part similar to those of HAL etc.

In April 1971, the board of directors of HAL appointed committee known as review committee of HAL to review the functioning of the company and make its recommendations.

Committee formulated a statement of basic objective adoption by HAL, as detailed below:

BASIC OBJECTIVES

To serve as an instrument of the national policy to achieve self- reliance in the design, development and production aircraft and aeronautical equipment to meet the counter changing and growing needs, with special emphasis on millet requirement.

In fulfilment of these objectives, the company shall regard its fundamentally responsible for design and development, rely however, upon such relevant facilities as are available in to national institutions, but always holding itself basic a responsible for the growth and furtherance of the counter aeronautics capability.

To so conducts its business economically and efficiently it can contribute its due share to the national effort achieving a self- reliant and self- generating economy.

Towards this end, to develop and maintain an organization which will readily respond to and adopt the changing matrix of soc techno-economic relationship and where in a climate of grow professional competence, self- discipline, mutual understand deep commitment and a sense of belonging will be fostered each employee will be encouraged to grow in accordance with potential for the furtherance of the organization.The recommendations of the review were subsequently approved by the board of Directors in September 1972 and forwarded to government, who informed the company in May 1973 that they may adopt these objectives. These objectives remain essential unchanged to date except that they have been amplified in 1983 through the medium of MD's dated 14th 1983 emphasizing the following:

That our products are of the highest quality and reliability. That our products are fully supported after sale to customer. That capacity utilization is optimized, restoring which is necessary to

diversification and export. Greater thrust towards indigenization of materials and product to improve self-reliance.

**MAJOR COMPLEXES**

**HAL has five main complexes in India: -**

1. Bangalore Complex

2. MIG Complex

3. Accessories Complex

4. Design Complex

5. Helicopter Complex

**BANGALORE COMPLEX**

(A) Aircraft Division - Manufacturing Jaguar Aircrafts

(B) Engine Division - Manufacturing Jaguar Engines

(C) Helicopter Division - Manufacturing Helicopters

(D) Forge & Foundry Division - Manufacturing high precision casting and forging

(E) Overhaul Division - Overhaul of Jaguar and other engines

(F) Space Division - Manufacturing and launching of pads and common satellites

(G) Servicing Division - For common services to all divisions

**2. MIG COMPLEX**

(A) Nasik Division - Manufacturing and overhaul of airframes

(B) Koraput Division - Manufacturing and overhaul of MIG engines and SU-30 engine overhaul and manufacturing

3. ACCESSORIES COMPLEX

(A) Hyderabad Division - Manufacturing of electronics and navigational equipment

(B) Kanpur Division - Manufacturing of passenger aircraft and gliders

(C) Lucknow Division - Manufacturing of hydraulic pumps, fuel pumps and stator generator

(D) Korwa Division - Manufacturing of advanced navigational equipment

4. DESIGN COMPLEX

(A) Bangalore Division - Modification of any component or unit of an engine

**5. HELICOPTER COMPLEX**

(A)Barrackpore Division - Manufacturing and overhaul of helicopter components

**KORAPUT ENGINE DIVISION**

Engine Division - Koraput, a unit of HAL's vast network, was set up in April 1964 to manufacture R11-f2 turbo jet engine of MiG-21 FL aircraft under license from the erstwhile SSR. Subsequently, the division took up, under various license agreements, manufacture of R-11 series engines of MIG-21 FL and MiG-21M aircraft, R-25 series engines of MiG-21 BI's aircraft and R-29B engines for Mig-27M aircraft. Simultaneously, facilities were set up for overhaul of R-11 and R-25 series engines, which started in 1971 and 1983 respectively. Starting from 1997-98 overhaul of RD - 33 Engines of MiG-29 aircraft were also conducted.

The various departments present in Engine Division are:

1. Forge

2. Foundry

3. Tool room

4. Small parts & fuel

5. Sheet metal & welding

6. Blades

7. Electroplating

8. Heat treatment

9. Compressor

10.Turbine

11.CNC

12.Assembly

13.Overhaul

14.Gear

15.Test house

16.Maintenance

SUKHOI ENGINE DIVISON

Sukhoi Engine Division has 6 workshops, called as Hangar.

Each hangar has different work for engine.

1.HANGER 101

* Compressor Shop-611
* Turbine Shop-610
* Blade Shop-615

2.HANGER 102

* CNC Shop-616
* Gear Casing Shop-609

3.HANGER 103

* Heat Treatment Shop-603
* Electroplating Shop-604

4. HANGER-104

* Assembly-620
* Jet Nozzle Assembly-620A
* Fuel&Small Parts-614
* Rig Room-614A

5.HANGER-105

* Sheet Metal Shop-612
* Special Equipment Shop

6.HANGER-106

* Aggregate Overhaul Shop-613

|  |  |  |  |
| --- | --- | --- | --- |
| SL.NO | AIRCRAFT | ENGINE | INDIGENOUS NAME |
| 1 | MIG-21FL | R11-F20 | BADAL |
| 2 | MIG-21M/MF | R11-F2S/F2SK | TRISHUL |
| 3 | MIG-21BIS | R-25 | VIKARM |
| 4 | MIG-23MF | R-29 | RAKSHAK |
| 5 | MIG-23BN | R-29B | VIJAY |
| 6 | MIG-25 | R-29B | GARUD |
| 7 | MIG-27M | R-29B | BAHADUR |
| 8 | MIG-29 | RD-33 | VAJ |
| 9 | SUKHOI 30MKI | AL-31FP | FLANKER-H |
| 10 | GNAT | ORPHEUS-701 | AJEET |
| 11 | HF-24 | ORPHEUS-703 | MARUT |
| 12 | HJT-16 | VIPER-11 | KIRAN |
| 13 | JAGUAR | ADOUR MK-803 | SHAMSHOR |
| 14 | MIRAGE-2000 | M-53 | VAJRA |
| 15 | LCA MK1A | GE F404-IN20 | TEJAS |
| 16 | HS-748 | DART-531 | CHITRA |
| 17 | ALLOUTEE | ARTOUSTE-IIIB | CHEETAH |

**MAIN.PY**

from flask import Flask, render\_template, request, redirect, url\_for, session, g

import sqlite3

app = Flask(\_\_name\_\_)

app.secret\_key = '001Ad002x1ew2341aditya'

DATABASE = 'app.db'

# Database functionality

def get\_db():

    if 'db' not in g:

        g.db = sqlite3.connect(DATABASE)

        g.db.row\_factory = sqlite3.Row

    return g.db

@app.teardown\_appcontext

def close\_db(exception):

    db = g.pop('db', None)

    if db is not None:

        db.close()

def init\_db():

    with app.app\_context():

        db = get\_db()

        db.executescript('''

        CREATE TABLE IF NOT EXISTS users (

            id INTEGER PRIMARY KEY AUTOINCREMENT,

            email TEXT UNIQUE NOT NULL,

            password TEXT NOT NULL,

            role TEXT NOT NULL

        );

        CREATE TABLE IF NOT EXISTS medicines (

            id INTEGER PRIMARY KEY AUTOINCREMENT,

            store\_id INTEGER,

            name TEXT,

            code TEXT,

            quantity INTEGER,

            category TEXT,

            FOREIGN KEY (store\_id) REFERENCES users(id)

        );

        CREATE TABLE IF NOT EXISTS orders (

            id INTEGER PRIMARY KEY AUTOINCREMENT,

            user\_id INTEGER,

            store\_id INTEGER,

            medicine\_code TEXT,

            quantity\_requested INTEGER,

            status TEXT,

            FOREIGN KEY (user\_id) REFERENCES users(id),

            FOREIGN KEY (store\_id) REFERENCES users(id)

        );

        ''')

        db.commit()

@app.route('/')

def home():

    return redirect(url\_for('login'))

@app.route('/register', methods=['GET', 'POST'])

def register():

    if request.method == 'POST':

        try:

            email = request.form['email'].strip()

            password = request.form['password'].strip()

            role = request.form['role'].strip()

            db = get\_db()

            db.execute('INSERT INTO users (email, password, role) VALUES (?, ?, ?)', (email, password, role))

            db.commit()

            return redirect(url\_for('login'))

        except sqlite3.IntegrityError:

            return "User already exists"

        except Exception as e:

            return f"Error: {e}"

    return render\_template('register.html')

@app.route('/login', methods=['GET', 'POST'])

def login():

    if request.method == 'POST':

        try:

            email = request.form['email'].strip()

            password = request.form['password'].strip()

            db = get\_db()

            user = db.execute('SELECT \* FROM users WHERE email = ? AND password = ?', (email, password)).fetchone()

            if user:

                session['user\_id'] = user['id']

                session['role'] = user['role']

                return redirect(url\_for('dashboard\_store' if user['role'] == 'store' else 'dashboard\_public'))

            else:

                return "Invalid Login credentials."

        except Exception as e:

            return f"Login error: {e}"

    return render\_template('login.html')

@app.route('/logout')

def logout():

    session.clear()

    return redirect(url\_for('login'))

@app.route('/dashboard/store')

def dashboard\_store():

    if session.get('role') != 'store':

        return redirect(url\_for('login'))

    return render\_template('dashboard\_store.html')

@app.route('/add\_medicine', methods=['GET', 'POST'])

def add\_medicine():

    if session.get('role') != 'store':

        return redirect(url\_for('login'))

    if request.method == 'POST':

        name = request.form['name'].strip()

        code = request.form['code'].strip()

        quantity = int(request.form['quantity'])

        category = request.form['category'].strip()

        store\_id = session['user\_id']

        db = get\_db()

        existing = db.execute(

            'SELECT \* FROM medicines WHERE store\_id = ? AND name = ? AND code = ?',

            (store\_id, name, code)).fetchone()

        if existing:

            new\_quantity = existing['quantity'] + quantity

            db.execute('UPDATE medicines SET quantity = ? WHERE id = ?', (new\_quantity, existing['id']))

        else:

            db.execute(

                'INSERT INTO medicines (store\_id, name, code, quantity, category) VALUES (?, ?, ?, ?, ?)',

                (store\_id, name, code, quantity, category))

        db.commit()

        return redirect(url\_for('view\_medicines'))

    return render\_template('add\_medicine.html')

@app.route('/view\_medicines')

def view\_medicines():

    if session.get('role') != 'store':

        return redirect(url\_for('login'))

    db = get\_db()

    store\_id = session['user\_id']

    meds = db.execute('SELECT \* FROM medicines WHERE store\_id = ?', (store\_id,)).fetchall()

    return render\_template('view\_medicines.html', medicines=meds)

@app.route('/view\_orders', methods=['GET', 'POST'])

def view\_orders():

    if session.get('role') != 'store':

        return redirect(url\_for('login'))

    db = get\_db()

    store\_id = session['user\_id']

    if request.method == 'POST':

        order\_id = request.form['order\_id']

        action = request.form['action']

        order = db.execute('SELECT \* FROM orders WHERE id = ?', (order\_id,)).fetchone()

        medicine = db.execute(

            'SELECT \* FROM medicines WHERE code = ? AND store\_id = ?',

            (order['medicine\_code'], store\_id)).fetchone()

        if action == 'accept':

            if medicine and medicine['quantity'] >= order['quantity\_requested']:

                db.execute('UPDATE orders SET status = ? WHERE id = ?', ('Accepted', order\_id))

                db.execute('UPDATE medicines SET quantity = quantity - ? WHERE code = ? AND store\_id = ?',

                           (order['quantity\_requested'], order['medicine\_code'], store\_id))

            else:

                return "Not enough stock to accept this order."

        elif action == 'reject':

            db.execute('UPDATE orders SET status = ? WHERE id = ?', ('Rejected', order\_id))

        db.commit()

    orders = db.execute('''

        SELECT o.\*, u.email FROM orders o

        JOIN users u ON o.user\_id = u.id

        WHERE o.store\_id = ?

    ''', (store\_id,)).fetchall()

    return render\_template('view\_orders.html', orders=orders)

@app.route('/dashboard/public')

def dashboard\_public():

    if session.get('role') != 'public':

        return redirect(url\_for('login'))

    return render\_template('dashboard\_public.html')

@app.route('/search', methods=['GET', 'POST'])

def search():

    results = []

    if session.get('role') != 'public':

        return redirect(url\_for('login'))

    if request.method == 'POST':

        keyword = request.form['keyword']

        db = get\_db()

        results = db.execute('''

            SELECT m.\*, u.email as store\_email FROM medicines m

            JOIN users u ON m.store\_id = u.id

            WHERE m.name LIKE ? OR m.code LIKE ?

        ''', ('%' + keyword + '%', '%' + keyword + '%')).fetchall()

    return render\_template('search.html', results=results)

@app.route('/request\_order', methods=['POST'])

def request\_order():

    if session.get('role') != 'public':

        return redirect(url\_for('login'))

    code = request.form['code']

    store\_id = int(request.form['store\_id'])

    quantity = int(request.form['quantity'])

    user\_id = session['user\_id']

    db = get\_db()

    db.execute(

        'INSERT INTO orders (user\_id, store\_id, medicine\_code, quantity\_requested, status) VALUES (?, ?, ?, ?, ?)',

        (user\_id, store\_id, code, quantity, 'Pending'))

    db.commit()

    return redirect(url\_for('dashboard\_public'))

@app.route('/my\_orders')

def my\_orders():

    if session.get('role') != 'public':

        return redirect(url\_for('login'))

    db = get\_db()

    user\_id = session['user\_id']

    orders = db.execute('SELECT \* FROM orders WHERE user\_id = ?', (user\_id,)).fetchall()

    return render\_template('my\_orders.html', orders=orders)

if \_\_name\_\_ == '\_\_main\_\_':

    with app.app\_context():

        init\_db()

    app.run(host='0.0.0.0', port=5000, debug=True)

**STYLE.CSS**

/\* General \*/

body {

margin: 0;

padding: 0;

font-family: "Segoe UI", Roboto, Helvetica, Arial, sans-serif;

background: linear-gradient(to bottom right, #f0fdf4, #d1fae5);

display: flex;

justify-content: center;

align-items: center;

min-height: 100vh;

}

.container {

background-color: #ffffff;

padding: 2rem;

border-radius: 1rem;

box-shadow: 0 10px 30px rgba(0, 0, 0, 0.1);

width: 100%;

max-width: 700px;

margin: 2rem;

}

/\* Headings \*/

h2 {

text-align: center;

color: #059669;

font-size: 2rem;

margin-bottom: 1.5rem;

}

h3 {

font-size: 1.2rem;

color: #2563eb;

margin-top: 1.5rem;

margin-bottom: 0.5rem;

}

/\* Inputs & Selects \*/

input[type="text"],

input[type="email"],

input[type="password"],

input[type="number"],

select {

width: 100%;

padding: 0.75rem;

margin-top: 0.25rem;

margin-bottom: 1rem;

font-size: 1rem;

border: 1px solid #d1d5db;

border-radius: 0.5rem;

box-sizing: border-box;

transition: border 0.3s, box-shadow 0.3s;

}

input:focus,

select:focus {

border-color: #10b981;

box-shadow: 0 0 0 3px rgba(16, 185, 129, 0.3);

outline: none;

}

/\* Buttons \*/

button {

background-color: #10b981;

color: #fff;

border: none;

padding: 0.7rem 1.2rem;

font-size: 1rem;

font-weight: 500;

border-radius: 0.5rem;

cursor: pointer;

transition: background 0.3s ease;

}

button:hover {

background-color: #059669;

}

button.danger {

background-color: #ef4444;

}

button.danger:hover {

background-color: #dc2626;

}

/\* Cards / Lists \*/

ul {

list-style: none;

padding: 0;

}

.card {

background: #f9fafb;

padding: 1rem;

border-radius: 0.75rem;

box-shadow: 0 2px 10px rgba(0, 0, 0, 0.05);

margin-bottom: 1rem;

}

/\* Status badges \*/

.badge {

display: inline-block;

padding: 0.35em 0.8em;

font-size: 0.85rem;

font-weight: 600;

border-radius: 9999px;

margin-top: 0.3rem;

}

.badge.pending {

background-color: #fef3c7;

color: #92400e;

}

.badge.approved {

background-color: #d1fae5;

color: #065f46;

}

.badge.rejected {

background-color: #fecaca;

color: #991b1b;

}

/\* Link style \*/

a {

color: #2563eb;

text-decoration: none;

font-weight: 500;

}

a:hover {

text-decoration: underline;

}

/\* Forms & layout tweaks \*/

form {

display: flex;

flex-direction: column;

gap: 0.5rem;

}

/\* Responsive \*/

@media (max-width: 600px) {

.container {

padding: 1rem;

}

h2 {

font-size: 1.5rem;

}

form {

gap: 0.75rem;

}

}

LOGIN.HTML

<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <title>Login</title>

  <link rel="stylesheet" href="{{ url\_for('static', filename='style.css') }}">

</head>

<body>

  <div class="container">

    <h2>Login</h2>

    <form method="post">

      <label>Email:</label><br>

      <input type="email" name="email" required><br>

      <label>Password:</label><br>

      <input type="password" name="password" required><br><br>

      <button type="submit">Login</button>

    </form>

    <p style="margin-top: 1rem;">Don't have an account?

      <a href="{{ url\_for('register') }}">Register</a>

    </p>

  </div>

</body>

</html>

REGISTER.HTML

<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <title>Register</title>

  <link rel="stylesheet" href="{{ url\_for('static', filename='style.css') }}">

</head>

<body>

  <div class="container">

    <h2>Register</h2>

    <form method="post">

      <label>Email:</label><br>

      <input type="email" name="email" required><br>

      <label>Password:</label><br>

      <input type="password" name="password" required><br>

      <label>User:</label><br>

      <select name="role" required>

        <option value="store">Medical Store</option>

        <option value="public">Public User</option>

      </select><br><br>

      <button type="submit">Register</button>

    </form>

    <p style="margin-top: 1rem;">Already have an account?

      <a href="{{ url\_for('login') }}">Login</a>

    </p>

  </div>

</body>

</html>

DASHBOARD\_STORE.HTML

<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <title>Public User Dashboard</title>

  <link rel="stylesheet" href="{{ url\_for('static', filename='style.css') }}">

</head>

<body>

  <div class="container">

    <h2>Public User Dashboard</h2>

    <ul>

      <li><a href="/search">🔍 Search for Medicines</a></li>

      <li><a href="/my\_orders">📋 My Orders</a></li>

      <li><a href="/logout">🚪 Logout</a></li>

    </ul>

  </div>

</body>

</html>

ADD MEDICINE.HTML

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Add Medicine</title>

<link rel="stylesheet" href="{{ url\_for('static', filename='style.css') }}">

</head>

<body>

<div class="container">

<h2>Add Medicine</h2>

<form method="post">

<label>Name:</label><br>

<input name="name" placeholder="Medicine Name" required><br>

<label>Code:</label><br>

<input name="code" placeholder="Medicine Code" required><br>

<label>Quantity:</label><br>

<input name="quantity" type="number" placeholder="Quantity" required><br>

<label>Category:</label><br>

<select name="category" required>

<option value="Allopathic">Allopathic</option>

<option value="Ayurvedic">Ayurvedic</option>

<option value="Homeopathic">Homeopathic</option>

</select><br><br>

<button type="submit">Add Medicine</button>

</form>

<p style="margin-top: 1rem;">

<a href="/dashboard/store">← Back to Dashboard</a>

</p>

</div>

</body>

</html>

VIEW MEDICINE.HTML

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Your Medicines</title>

<link rel="stylesheet" href="{{ url\_for('static', filename='style.css') }}">

</head>

<body>

<div class="container">

<h2>Your Medicines</h2>

<ul>

{% for m in medicines %}

<li class="card">

<strong>{{ m.name }}</strong> ({{ m.code }})<br>

Quantity: {{ m.quantity }}<br>

Category: <span class="badge">{{ m.category }}</span>

</li>

{% endfor %}

</ul>

<p style="margin-top: 1rem;">

<a href="/dashboard/store">← Back to Dashboard</a>

</p>

</div>

</body>

</html>

MY ORDERS.HTML

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Order Requests</title>

<link rel="stylesheet" href="{{ url\_for('static', filename='style.css') }}">

</head>

<body>

<div class="container">

<h2>Order Requests</h2>

{% for order in orders %}

<div class="card">

<p><strong>User:</strong> {{ order['email'] }}</p>

<p><strong>Medicine Code:</strong> {{ order['medicine\_code'] }}</p>

<p><strong>Quantity:</strong> {{ order['quantity\_requested'] }}</p>

<p>

<strong>Status:</strong>

<span class="badge

{% if order['status'] == 'Pending' %} pending

{% elif order['status'] == 'Approved' %} approved

{% elif order['status'] == 'Rejected' %} rejected

{% endif %}

">{{ order['status'] }}</span>

</p>

{% if order['status'] == 'Pending' %}

<form method="post" style="margin-top: 1rem; display: flex; gap: 0.5rem;">

<input type="hidden" name="order\_id" value="{{ order['id'] }}">

<button type="submit" name="action" value="accept">✅ Accept</button>

<button type="submit" name="action" value="reject" style="background: #dc2626;">❌ Reject</button>

</form>

{% endif %}

</div>

{% endfor %}

<p style="margin-top: 1rem;">

<a href="/dashboard/store">← Back to Dashboard</a>

</p>

</div>

</body>

</html>

SEARCH.HTML

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Search Medicines</title>

<link rel="stylesheet" href="{{ url\_for('static', filename='style.css') }}">

</head>

<body>

<div class="container">

<h2>Search Medicines</h2>

<form method="post">

<input type="text" name="keyword" placeholder="Enter medicine name or code" required>

<button type="submit">Search</button>

</form>

{% if results %}

<h3>Results:</h3>

<ul>

{% for med in results %}

<li class="card">

<p><strong>{{ med.name }}</strong> (Code: {{ med.code }}) - Qty: {{ med.quantity }}</p>

<p>Category: {{ med.category }} | Store: {{ med.store\_email }}</p>

<form action="/request\_order" method="POST" style="margin-top: 0.5rem;">

<input type="hidden" name="code" value="{{ med.code }}">

<input type="hidden" name="store\_id" value="{{ med.store\_id }}">

<label>Quantity:</label>

<input type="number" name="quantity" min="1" max="{{ med.quantity }}" required>

<button type="submit">🛒 Order</button>

</form>

</li>

{% endfor %}

</ul>

{% endif %}

<p style="margin-top: 1rem;">

<a href="/dashboard/public">← Back to Dashboard</a>

</p>

</div>

</body>

</html>

DASHBOARD\_PUBLIC.HTML

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>My Orders</title>

<link rel="stylesheet" href="{{ url\_for('static', filename='style.css') }}">

</head>

<body>

<div class="container">

<h2>My Orders</h2>

<ul>

{% for order in orders %}

<li class="card">

<p><strong>Medicine Code:</strong> {{ order['medicine\_code'] }}</p>

<p><strong>Quantity:</strong> {{ order['quantity\_requested'] }}</p>

<p>

<strong>Status:</strong>

<span class="badge

{% if order['status'] == 'Pending' %} pending

{% elif order['status'] == 'Approved' %} approved

{% elif order['status'] == 'Rejected' %} rejected

{% endif %}

">

{{ order['status'] }}

</span>

</p>

</li>

{% endfor %}

</ul>

<p style="margin-top: 1rem;">

<a href="/dashboard/public">← Back to Dashboard</a>

</p>

</div>

</body>

</html>

PROJECT DESCRIPTION AND RESULTS

**📦 Pharmacy Application(MEDIFLOW) – Project Description**

The **Pharmacy Application(MEDIFLOW)** is a full-stack web application built using **Flask** (backend) and **SQLite** (database). It is designed to streamline the process of **managing medicine inventory**, **handling customer orders**, and ensuring efficient communication between **medical store users** (sellers) and **public users** (buyers).

This system supports two user roles:

* 🏪 **Medical Store Users** – Can add, update, and manage medicines, view and respond to customer orders.
* 👤 **Public Users** – Can search for available medicines, place orders, and track order status.

Key features include:

* Secure login/registration for both user types
* Inventory management (add, view, update stock)
* Medicine search and ordering system
* Order status tracking (Pending, Approved, Rejected)
* Clean, user-friendly dashboard UI

This system helps **pharmacies** and **healthcare platforms** maintain accurate inventory, reduce manual paperwork, and improve service quality for end users.

Technology Stack:

* Frontend: HTML, CSS
* Backend: Python with Flask
* Database: SQLite (lightweight and embedded)

**RESULTS:**

 



