Here is an outline of the Solution which comprises of a mobile App that allows registered users to order medications and integrates with the existing legacy system.

**Components:**

**Mobile App**: **Utibu Mobile Application (Developed with Flutter):**

Developed using Flutter for cross-platform compatibility (Android and iOS).

User interface is designed to allow users to:

* Login with secure authentication.
* Browse a medication catalogue.
* Place medication orders.

**Improvements that can be made to the current state of the App**

1. Register new with secure authentication.
2. View their medication history and upcoming refills.
3. Browse a medication catalogue downloadable from the server.
4. Place medication orders with quantity selection.
5. Develop functionality in the mobile app to confirm orders and provide feedback to the user.
6. Choose payment method (immediate or on pickup).
7. Track order status (confirmed, processing, ready for pickup).
8. View their medication statements.
9. Enable communication channels between customers and pharmacists/administrators for inquiries or support.

**N/B**- *Time was a constraint and I did not manage to implement some of the above functions within the required deadline.*

**Technology Stack:**

**Mobile App**: Flutter (development)

**User Authentication**: should be developed utilising Firebase.

**API Server:** Python (Flask or Django framework).

Deploy on a cloud platform (AWS, Google Cloud, etc.) with a public IP address.

The API Server should act as a mediator between the mobile app and the legacy system and perform the following functions.

* Handles user authentication and authorization.
* Receives medication orders from the app.
* Validates orders based on medication availability.
* If valid, forwards the order details to the legacy system integration.
* Sends confirmation or rejection response back to the mobile app.
* Provides access to user statements upon request.

**Legacy System Integration:**

To Integrate the Mobile App with the Legacy System, a Python script using libraries for database access (such as SQLAlchemy) can be written. The Script should accomplish the following:

1. Runs on a machine within the health facility with access to the legacy database.
2. Listens for incoming medication orders from the API server.
3. Connects to the Microsoft SQL Server database.
4. Checks stock availability in the database.
5. If stock is sufficient, creates a new order record in the database.
6. Updates inventory levels in the database.
7. Informs the API server of order success or failure.

The middleware script acts as a bridge, translating API messages into database operations within the legacy system.

**Communication Channels:**

Make use of email APIs, SMS gateways, or push notifications for communication.

**Benefits of this solution when implemented:**

1. Improved patient experience with remote medication ordering.
2. Reduced wait times at the pharmacy.
3. Increased efficiency for pharmacy staff.
4. Real-time stock management.
5. Existing legacy system remains operational.

**Additional Considerations:**

* Secure data transmission using encryption (HTTPS) and user authentication.
* Implement offline functionality to allow users to browse medications and see past orders even without internet connection.
* Push notifications for order status updates.
* Integration with a payment gateway for online payments.

**Note:**

Sign-In Button in the login page has been modified to call the Introduction page for demonstration purposes.

Emulator Version Used – Pixel 6 API 34 running on Android 14.0