Student Enrollment System

Payment Module Implementation Report

Date: January 16, 2026

Module: Real UPI Payment Integration

1. Executive Summary

This report documents the successful implementation of the Real UPI Payment Module. The system now supports dynamic course fees and generates scannable QR codes for direct payments via PhonePe, Google Pay, and Paytm.

Key Features Delivered:

- **Dynamic Course Fees:** Admins can set specific fees for each course.
- **Real UPI QR Code:** Generates valid `upi://` QR codes for scanning.
- **Deep Linking:** Mobile buttons to open PhonePe/GPay directly.
- **Robust Fallback:** Uses Online QR API if local libraries fail.
- **Transaction Simulation:** 'I Have Paid' confirmation flow.

2. Technical Implementation

A. Database Schema Changes

Modified the schema to support fees and payment status.

Table	Change	Description
courses	Added `fee` (FLOAT)	Stores course price (Default: 500.0).
enrollments	Status Update	Added 'pending_payment' status logic.

B. Infrastructure & Packages

New Python libraries installed to support QR generation.

Package	Purpose	Installation Command
qrcode[pil]	Generates QR code images locally.	pip install qrcode[pil]
pillow	Image processing dependency.	pip install pillow

3. Payment Workflow Logic

Step 1: Enrollment

User clicks 'Enroll'. System creates an Enrollment record with `status='pending_payment'`.

Step 2: Payment Page

User is redirected to `/student/payment/`. The system:

- Fetches the Course Fee.
- Generates a UPI Deep Link: `upi://pay?pa=iranna4@ptyes&...`
- Generates a QR Code image (Locally or via fallback API).
- Displays 'Pay via PhonePe/GPay' buttons.

Step 3: Confirmation

User pays via App -> Clicks 'I Have Paid'.

System updates status to 'enrolled', sends confirmation email with Invoice/Rules PDF.

Configuration: Changing the UPI ID

To change the recipient bank account, edit `routes/student_routes.py`:

```
# Inside routes/student_routes.py

# Using User's Provided UPI ID
admin_upi_id = "iranna4@ptyes"  # <-- CHANGE THIS VAULE

qr_b64, upi_url = generate_upi_qr(
    upi_id=admin_upi_id,
    ...
)
```

4. Session Command Log

Chronological list of key actions taken during this session.

Action Type	Details
Prompt	User: 'make Real UPI Payment Integration'
Coding	Added `fee` column to models & database.
Coding	Created `payment.html` & `confirmation.html`.
Command	python -m pip install qrcode pillow
Debugging	Fixed `BuildError` in auth.login (Blueprints).
Debugging	Fixed `No module named qrcode` crash.
Debugging	Fixed 'Invalid UPI ID' error (Updated to 'iranna4@ptyes').
Verification	Tested QR Code generation and display.

End of Report