PDF Generation Logic (enhanced_invoice.py)

1. Executive Summary

Implements the programmatic generation of PDF invoices using the ReportLab library.

2. Code Logic & Functionality

1. **Canvas Setup**: Creates a PDF canvas in memory using `io.BytesIO`.
2. **Layout Design**: Defines the physical layout of elements (Logo, Table, Footer) using absolute coordinates or flowables.
3. **Content Rendering**: Iterates through booking data to populate the itemized table.
4. **Output**: Returns the raw PDF bytes to be sent to the user's browser.

3. Key Concepts & Definitions

- **ReportLab**: An open-source library for creating PDFs in Python.
- **Flowable**: An element (paragraph, table) that can 'flow' across pages.
- **Buffer**: A temporary memory space used to store the PDF before saving.

4. Location Details

Path: `enhanced_invoice.py` **Type**: .PY File

5. Source Code Preview (Snippet)

Running typical software analysis on this file:

```
""" Enhanced PDF Invoice Generator with Professional Branding Includes school logo, QR code, better layout, and
terms """ from reportlab.lib.pagesizes import letter from reportlab.lib import colors from reportlab.platypus
import SimpleDocTemplate, Table, TableStyle, Paragraph, Spacer, Image from reportlab.lib.styles import
getSampleStyleSheet, ParagraphStyle from reportlab.lib.units import inch from reportlab.graphics.shapes import
Drawing, Rect from reportlab.graphics.barcode.qr import QrCodeWidget from reportlab.graphics import renderPDF
from io import BytesIO from datetime import datetime def create_school_logo(width=200): """Create a simple
SVG-style logo using ReportLab graphics""" from reportlab.graphics.shapes import Drawing, Circle, String, Polygon
d = Drawing(width, 80) # School icon (simplified building) d.add(Polygon(points=[40,60, 60,40, 80,60],
fillColor=colors.HexColor('#002E5D'))) d.add(Rect(45, 30, 30, 30, fillColor=colors.HexColor('#0056A3')))
d.add(Rect(50, 35, 8, 10, fillColor=colors.white)) d.add(Rect(67, 35, 8, 10, fillColor=colors.white)) # Text
d.add(String(100, 50, 'GREENWOOD', fontSize=18, fontName='Helvetica-Bold', fillColor=colors.HexColor('#002E5D')))
d.add(String(100, 35, 'International School', fontSize=10, fontName='Helvetica',
fillColor=colors.HexColor('#0DA49F'))) return d def create_qr_code(booking_id): """Create QR code for invoice
verification""" qr_code = QrCodeWidget(f'https://greenwood.edu.uk/invoice/{booking_id}') d = Drawing(80, 80)
d.add(qr_code) return d def generate_professional_invoice(booking): """Generate enhanced PDF invoice with
branding""" buffer = BytesIO() doc = SimpleDocTemplate( buffer, pagesize=letter, topMargin=0.75*inch,
bottomMargin=0.75*inch, leftMargin=0.75*inch, rightMargin=0.75*inch ) elements = [] styles =
getSampleStyleSheet() # Custom Styles header_style = ParagraphStyle( 'HeaderStyle', parent=styles['Heading1'],
fontSize=32, textColor=colors.HexColor('#002E5D'), spaceAfter=5, fontName='Helvetica-Bold' ) subtitle_style =
ParagraphStyle( 'SubtitleStyle', parent=styles['Normal'], fontSize=11, textColor=colors.HexColor('#0DA49F'),
spaceAfter=20, fontName='Helvetica-Bold' ) section_heading = ParagraphStyle( 'SectionHeading',
parent=styles['Heading2'], fontSize=14, textColor=colors.HexColor('#002E5D'), spaceAfter=10, spaceBefore=15,
fontName='Helvetica-Bold', borderPadding=5, backColor=colors.HexColor('#F8F9FA') ) body_text = ParagraphStyle(
'BodyText', parent=styles['Normal'], fontSize=10, textColor=colors.HexColor('#333333') ) # === Header with Logo
=== logo_table = Table([[create_school_logo(), '']], colWidths=[250, 200]) logo_table.setStyle(TableStyle([
('VALIGN', (0, 0), (-1, -1), 'TOP'), ])) elements.append(logo_table) elements.append(Spacer(1, 20)) # === INVOICE
Title === elements.append(Paragraph("INVOICE", header_style)) elements.append(Paragraph("Activity Booking
Confirmation", subtitle_style)) elements.append(Spacer(1, 10)) # === Colored Header Bar === header_bar =
Table([['']], colWidths=[450]) header_bar.setStyle(TableStyle([ ('BACKGROUND', (0, 0), (-1, -1),
colors.HexColor('#0DA49F')), ('LINEABOVE', (0, 0), (-1, -1), 3, colors.HexColor('#002E5D')), ]))
elements.append(header_bar) elements.append(Spacer(1, 20)) # === Invoice Info and QR Code === invoice_info_data =
[ ['Invoice Number:', f'GRN-{booking.id:06d}', '', create_qr_code(booking.id)], ['Invoice Date:',
datetime.now().strftime('%d %B %Y'), '', ''], ['Payment Status:', 'PAID ✓', '', ''], ['Transaction ID:',
f'TXN-{booking.id}-{datetime.now().strftime("%Y%m%d")}', '', 'Scan for verification'] ] invoice_info_table =
Table(invoice_info_data, colWidths=[120, 200, 20, 110]) invoice_info_table.setStyle(TableStyle([ ('FONTNAME', (0,
0), (0, -1), 'Helvetica-Bold'), ('FONTSIZE', (0, 0), (-1, -1), 10), ('TEXTCOLOR', (0, 0), (0, -1),
colors.HexColor('#555555')), ('VALIGN', (0, 0), (-1, -1), 'TOP'), ('SPAN', (3, 0), (3, 2)), ('ALIGN', (3, 0), (3,
3), 'CENTER'), ('FONTSIZE', (3, 3), (3, 3), 7), ('TEXTCOLOR', (3, 3), (3, 3), colors.grey), ('BOTTOMPADDING', (0,
0), (-1, -1), 6), ])) elements.append(invoice_info_table) elements.append(Spacer(1, 25)) # === Bill To Section
=== elements.append(Paragraph("Bill To", section_heading)) bill_to_data = [ ['Parent/Guardian:',
booking.parent.full_name], ['Email:', booking.parent.email], ['Phone:', booking.parent.phone or 'N/A'],
['Student:', f'{booking.child.name} (Year {booking.child.grade})'] ] ... [Code Truncated for Documentation
Readability - See Source File for Complete Logic] ...
```