Problem Statement

Insurance companies face challenges in efficiently processing insurance applications and generating accurate quotations. The current manual process is time-consuming, error-prone, and lacks consistency. Key issues include:

- Manual data entry from various document formats (PDFs, images)
- Inconsistent interpretation of handwritten information
- Time-consuming calculation of premiums based on complex rating guides
- Delays in providing quotations to potential clients
- Increased operational costs due to manual processing

Solution Statement

We propose an automated document processing and quotation system that leverages modern technologies to streamline the application process. The solution includes:

- A user-friendly web interface for document upload
- Automated document processing using OCR and machine learning
- Intelligent information extraction from various document formats
- Automated premium calculation based on predefined rating guides
- Instant generation of professional PDF quotations

Expected Output

The system is expected to deliver the following outputs:

For Users:

- A responsive web interface for easy document upload
- Real-time feedback on upload status and processing progress
- Instant display of extracted information for verification
- Ability to download a professional PDF quotation

For Insurance Companies:

- Extracted information from uploaded documents
- Calculated premium based on the extracted information
- Generated PDF quotation containing all relevant details
- Reduced processing time from days to minutes
- Increased accuracy in data extraction and premium calculation
- Improved consistency in quotation generation
- Significant reduction in operational costs

System Performance:

- Processing time of less than 2 minutes per application
- OCR accuracy of at least 95% for typed text
- Handwriting recognition accuracy of at least 90% for digits
- Ability to handle concurrent requests from multiple users.

System Workflow

User

Uploads documents · Downloads quotation · Displays results

React Frontend

Sends files · Requests PDF · Returns data · Serves PDF

Flask Backend

Saves files → File System
Processes images → OpenCV
Extracts text → Tesseract OCR
Recognizes handwriting → Keras Model
Extracts information → Regex Processing
Calculates premium → Custom Rating Logic
Generates PDF → ReportLab