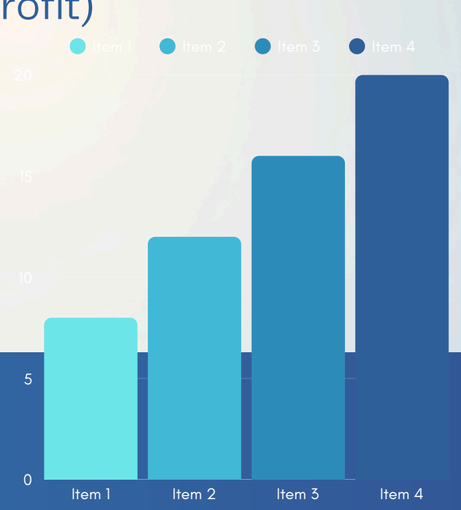
Project_Documentation.pdf

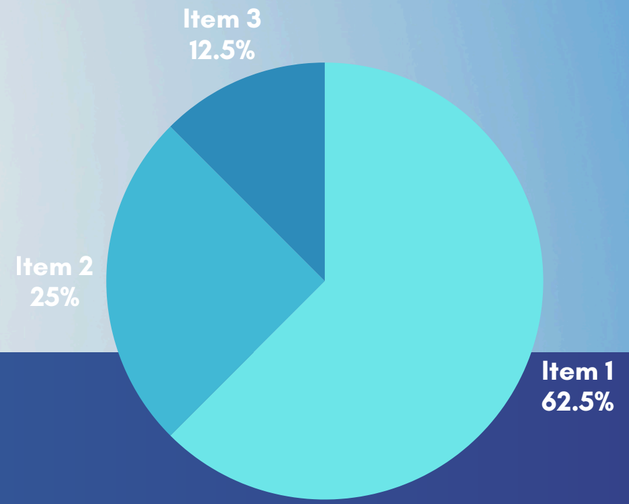
BUSINESS FINANCIAL MONITORING SYSTEM

3. Core Features and Capabilities

The system includes the following core functionalities:

- Manual income entry
- Manual expense entry with category classification
- OCR-based bill scanning and automatic amount extraction
- Real-time financial summary display (Income, Expenses, Net Profit)
- Budget monitoring with automated financial alerts
- Report export in CSV and Excel formats
- Financial data visualization (pie charts and line charts)
- Historical report archiving
- Search transactions by date
- Multi-business management within a single platform





Software management

4. Technical Architecture and Infrastructure

The system follows a modular three-layer architecture:

Frontend Layer
- Built using CustomTkinter
- Provides graphical user interface components
- Displays financial summaries and charts

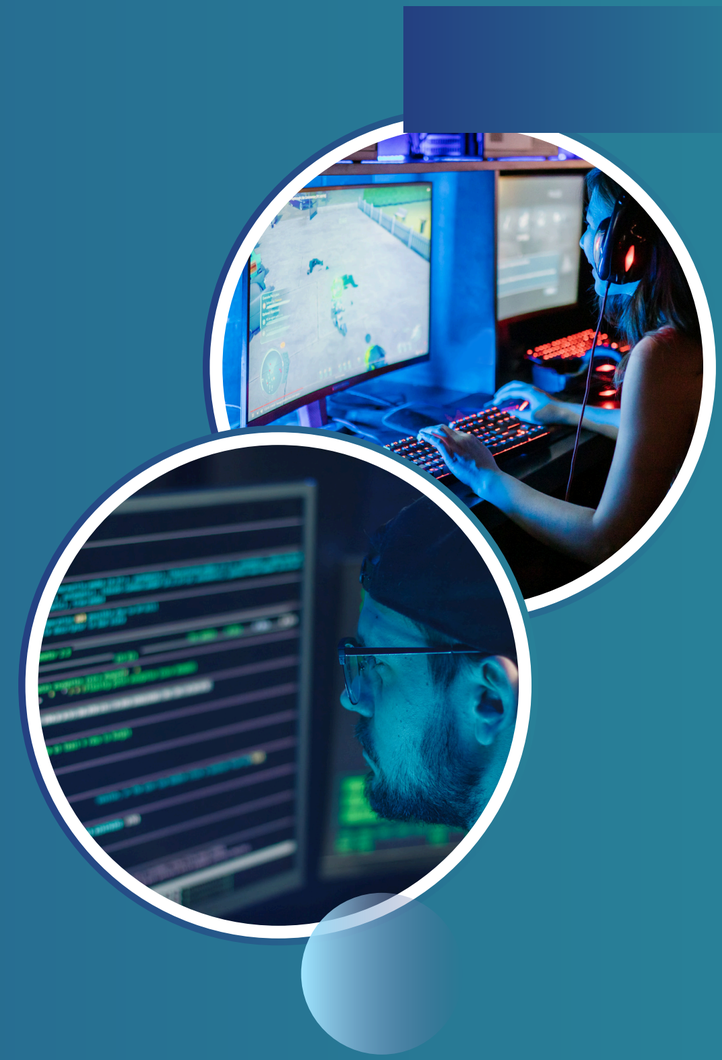
Backend Layer
- Developed in Python
- Handles business logic and financial calculations
- Manages OCR processing and report generation

Data Layer
- JSON files for storing business data
- CSV and Excel files for exported reports

The application is initiated through main.py, which connects all layers together.

5. Technology Stack

- Programming Language: Python 3.10+
- GUI Framework: CustomTkinter
- Data Visualization: Matplotlib
- OCR Engine: Tesseract (via pytesseract)
- Data Storage: JSON, CSV, Excel
- Libraries Used: Pillow, openpyxl



6. Security Architecture

The system implements basic security mechanisms, including:

- Username and password authentication
- Input validation for transaction entries
- Controlled access to financial data

Future improvements may include password encryption and database-level security controls.

7. Role and Authorization Management Model

The system supports a role-based access concept:

- Administrator: Full access to all system functionalities
- Accountant: Can add income and expense records
- Viewer: Can only view reports and analytics

Although currently implemented with a single administrator account, the architecture allows future expansion to multi-user environments.

8. Clearance Level System

The system proposes a hierarchical clearance model:
- Level 1: Read-only access
- Level 2: Data entry and modification rights
- Level 3: Full administrative control

Each user can be assigned a clearance level to ensure data integrity and operational control.

9. User Experience and Interface

The system provides:

- A modern and intuitive graphical interface
- Sidebar navigation for business selection
- Real-time financial dashboard
- Interactive data visualization charts
- Clear buttons for all major operations

The design prioritizes simplicity, usability, and efficiency.

10. Installation and Deployment

To deploy the system:

1. Install Python 3.10 or higher
2. Install required dependencies using:
   pip install –r requirements.txt

3. Install Tesseract OCR engine
4. Run the application using:
   python main.py

The system can operate on Windows, macOS, and Linux environments.

11. Integration and Scalability

The system supports:

- Management of multiple businesses
- Expandable architecture for database integration
- Potential API connectivity for accounting systems
- Flexible modular design for feature expansion

This structure ensures long-term scalability.

12. Competitive Analysis and Differentiation

Compared to traditional financial tracking methods, this system offers:

- Automated bill processing through OCR
- Visual financial analytics
- Multi-business management
- Exportable professional reports

These features differentiate it from basic manual bookkeeping systems.

13. Licensing and Commercial Model

The system may be distributed under:

- Free Trial License (limited features)
- Commercial License (full access)

The commercial model can include subscription-based or one-time licensing options.

14. Technical Requirements

MINIMUM SYSTEM REQUIREMENTS INCLUDE:

- PYTHON 3.10 OR HIGHER
- INSTALLED TESSERACT OCR ENGINE
- REQUIRED PYTHON LIBRARIES
- WINDOWS 10+, MACOS, OR LINUX
- MINIMUM 4GB RAM RECOMMENDED
- SUFFICIENT STORAGE FOR REPORT GENERATION