SDK Bank System Documentation

**Overview**

The SDK Bank System is a Streamlit-based banking application that provides various financial services including account management, money transfers, bill payments, and currency conversion.

**System Architecture**

Main Components:

1. **Main Application (**wep\_page(streamlit run).py**)**
   * Streamlit web interface
   * User interaction and navigation
   * Integration of all banking services
2. **Core Banking System (**sdk\_bank.py**)**
   * Account management (create, login)
   * Balance operations (deposit, withdraw)
   * Transaction logging
3. **Transaction Logger (**transaction\_logger.py**)**
   * Records all financial transactions
   * Provides transaction history
4. **Services:**
   * clicker.py: Peer-to-peer money transfers
   * efawateer.py: Bill payment system (electricity, water, telecom)
   * currency\_converter.py: Currency conversion tool

**Key Features**

1. User Authentication

* Account creation with security question
* Login via username or phone number
* Password recovery system

2. Banking Operations

* Deposit/withdraw funds
* Check account balance
* View transaction history

3. Financial Services

* **Click Transfer**: Send money to other users via phone number
* **EFAWATEER**: Pay utility bills (electricity, water, telecom)
* **Currency Converter**: Convert JOD to 20+ currencies

**Technical Details**

Session Management

* Uses Streamlit's session\_state to maintain:
  + User authentication state
  + Multi-step forms (for bill payments)
  + Bank instance persistence

**Data Storage**

* Transactions are logged to transactions.txt
* Account data is stored in memory (lists/dictionaries)

**Security Notes**

* Passwords are stored as integers (consider enhancing security)
* Uses security questions for password recovery
* All sensitive input fields use password masking

**Important Implementation Notes**

1. **Background Image**
   * Currently uses a hardcoded path (C:\Users\Ahmad\Desktop\VS code\Final Project\background\_image.jpg)
   * Should be modified for deployment
2. **Transaction Logging**
   * Uses a text-based system with custom parsing
   * Logs are stored in C:\Users\Ahmad\Desktop\VS code\Final Project\transactions.txt
3. **Currency Conversion**
   * Rates are hardcoded in currency\_converter.py
   * Supports 20+ currencies with fixed exchange rates
4. **Bill Payment System**
   * Electricity: Calculates based on device power usage
   * Water: Simple volume-based calculation
   * Telecom: Call minutes + data usage

***For Developers***

Extending the System

1. To add new services:
   * Create a new module similar to clicker.py or efawateer.py
   * Import and initialize in the main app
2. To modify transaction logging:
   * Edit transaction\_logger.py
   * Current format uses text blocks with separators
3. To change UI:
   * Modify the Streamlit components in wep\_page(streamlit run).py
   * Uses HTML/CSS for advanced styling

**Known Limitations**

* All data is in-memory (not persistent between runs)
* No database integration
* Basic security implementation (passwords as integers)
* Hardcoded file paths

**Example Usage Flow**

1. User creates account
2. Logs in with username/phone
3. Accesses dashboard:
   * Deposits funds
   * Pays electricity bill
   * Sends money to another user
   * Checks transaction history
4. Logs out