AKPAY

Description:

AKPAY is a digital transaction system that will work as a university-exclusive payment system. It is designed to simplify payments across campus by enabling all vendors in LUMS to accept payments using AKPAY. Each student, admin, and faculty member will have a transaction account linked to their SmartCard through which they can process transactions at any vendor within LUMS. This account can also be used to track transaction history, maintain a balance record, reactivate/deactivate the account, and recharge their account balance through its online system.

The system utilizes the C# (.NET) Framework for its front-end development alongside a Microsoft SQL Server backend. It will have functionalities based on both LINQ Queries and Stored Procedures, which will run on a Factory Design Pattern at runtime so that the user can choose how to run the program. In essence. AKPAY will offer a convenient and transparent transactional system exclusive to the campus database.

Problem Statement Motivation:

Having to carry multiple cards, being unable to pay directly to vendors in LUMS from mobile phones, and budgeting from your main bank account are problems that students and campus residents face on a regular basis. Additionally, systems across different eateries at LUMS vary from manual-only to hybrid payment systems. In order to standardize payments across campus, as well as help ease tracking and budget management across campus, we have planned this system. AKPAY will centralise payment systems across campus, making them cashless and secure for users holding valid LUMS emails. For vendors, this system will help organize data management, validate, and track transactions, while also providing a more consistent avenue for POS management across all vendors.

The potential Loyalty Points system that can be introduced with this can provide a two-way benefit to both the users and the vendors, as the users would get better offers, while the vendors would have an increase in the customer base as more people would focus on buying from within the university. The idea, tentatively focused on LUMS, can be extended to any university very easily as well. The main aim for our project is to standardize payments and make a secure, transparent, and easy-to-track unified payment system on campus.

System Objectives and Planning:

- The online application and the card allow a seamless system of transparent, secure transactions.
- Allows the user to switch dynamically during runtime between LINQ Queries and Stored Procedures.
- Will implement a clear database schema with entity relationships and constraints.
- The system focuses heavily on making security an essential feature that is implemented through verification processes preventing fraud transactions.
- The application has features for activating and deactivating the card, tracking the budget of money spent at each LUMS vendor with a clean visual design, and a student-to-student transaction system, too.
- In order to account for struggles to connect to the campus internet, we have implemented two different modes of payment: the card is used for scan/tap-n-go offline payments while the online application can be used for online payments in a similar fashion.
- It provides a standardized payment system to vendors across campus, thus making data tracking and analytics easy.
- It will help maintain records of transactions on a daily, weekly, or monthly basis for easier budgeting and financial management – caters to issues of monthly budgeting faced by students.

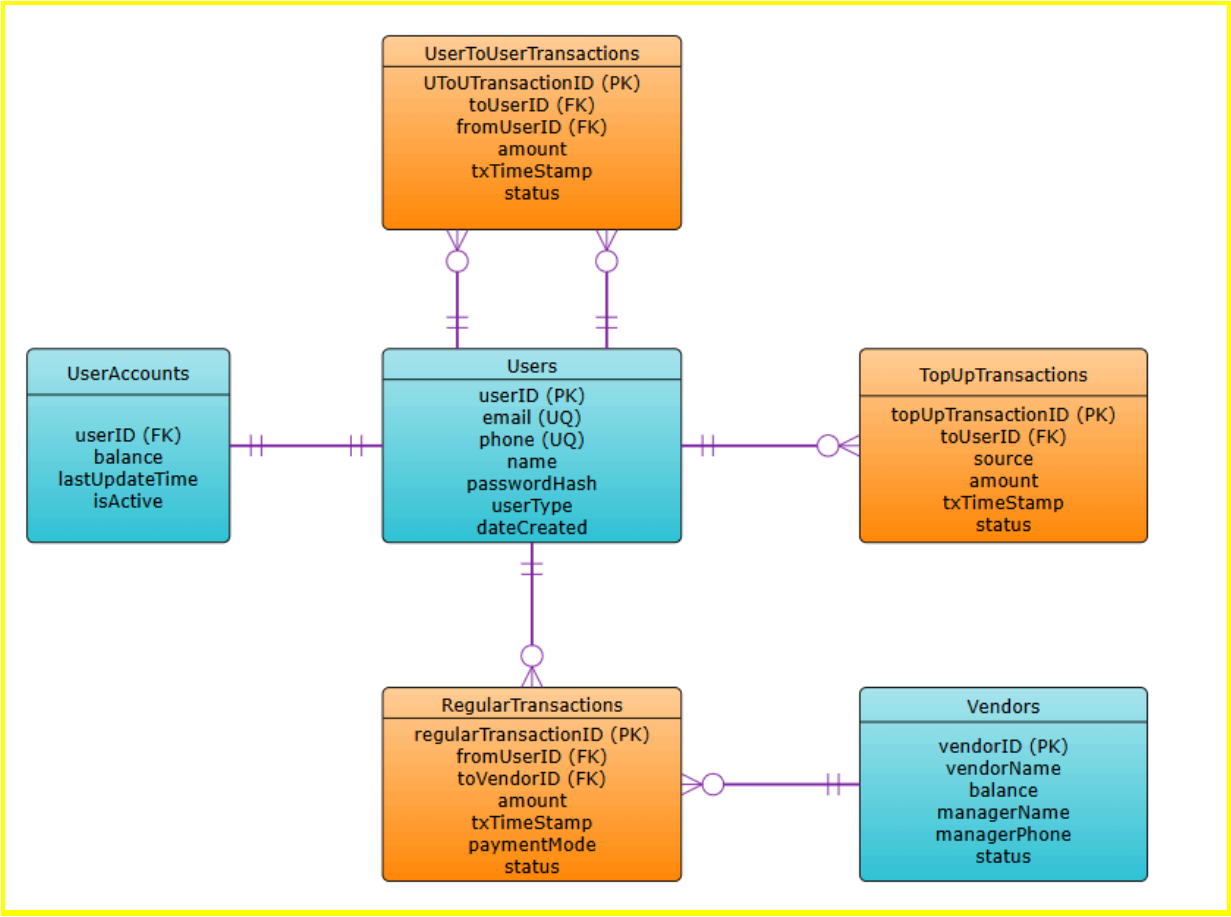
Main Features, Operations, and Outcome:

1. **Transaction Validation:** The transaction table will record the transactional amount, the sender, the receiver, the time and the date of the transaction. To ensure security within the system, this table will also implement a validation bit, which will track the transaction's progress and show whether it is rejected, processed, or pending. Invalid transactions will immediately be rejected, while those that fail to sync due to errors will handle data integrity through ensuring all transactions are atomic and no transaction occurs partially or is overwritten – transactions will switch to read-only once recorded. We are not deploying a caching system because of the small amount of vendors in our database, automatically allowing a faster retrieval of data.
2. **Online/Offline Methods:** In order to cater to our LUMS ecosystem, the database will ensure that if a user's mobile cannot connect to the Wi-Fi, they can still pay through the card-based system instead of the online application system. For offline transfers, we have assumed that all vendors, our application, and the university's servers are always running for a transaction to be processed.
3. **Data Visualisation:** For easier data tracking, our front-end display will visualise data for the user to track it better. It will include records of daily, weekly, or

monthly spending, and a ranking of vendors from most to least spent on. This will make tracking and budgeting more efficient and user-friendly.

4. **User-to-User Transfers:** In addition to our purchasing transaction system, students will be able to transfer money to one another as well. This will ensure a wider range of transactional networks to come under one umbrella of our application, making it a one-platform solution for all transactional queries within the LUMS ecosystem.

ER Diagram:



Names	**Roll numbers**
Ammar.	27100051
Muhammad Shahbaz Aziz Khan	27100049
Fajr Fatima	27020334
Muhammad Hassan Musa Gondal	27100456