ILLINOIS TECH | Discover. Create. Solve.

IIT BANK_INSTANT PAY

ITMD-510 Object oriented application development

Master of information technology and management

Names: Boga Supriya (**A20528526**)

Naga Satya Silpa Annadevara(**A20517818**)

Professor: James papademas

IN SUBMISSION OF FINAL PROJECT

Table of Contents

Abstract

This abstract talks about Instant Pay, a service that processes payments and allows customers to get their money in real-time or almost real-time. We created an application such that the users can send money instantaneously to other people or companies using the service, which is offered by financial institutions like banks or payment processors, through mobile apps, online banking platforms, or payment gateways that facilitate real-time transactions. As the need for quicker and more convenient payment solutions has developed, Instant Pay's popularity has surged dramatically in recent years. Payroll processing, bill payments, P2P payments, and e-commerce are just a few of the various transactions that use the service.

Improved cash flow, cheaper payment processing, and more customer satisfaction are some advantages of Instant Pay. In general, Instant Pay has become a practical and effective payment method that satisfies the expanding need for real-time payments. The inventory control system is comparable to this project. This system is used by Eclipse to log in, and when necessary for

security, it will create a user for you.

Among this project's most important aspects are:

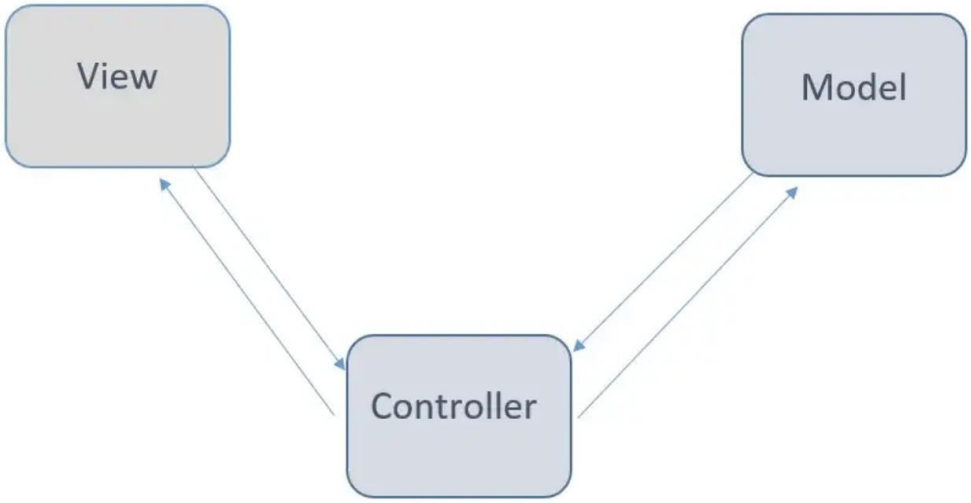
- Login
- New User Registration (Create user and sign up)
- Regular User
- Admin User
- Role based access control
- CRUD operations to add and remove items
- MVC architecture used for code efficiency

MVC Architecture

Model, view, and controller are the three components that make up the Model-View-Controller (MVC) architectural pattern. These parts are all designed to deal with particular facets of application development. MVC's status as an architectural pattern suggests that it has total control over the application's architecture. It may be misleading to refer to it as a design pattern, despite the fact that design patterns are usually used to manage certain technical issues, whereas architecture patterns deal with architectural issues and have an impact on the entire design of our program.
It is made up of three main components:

- Model
- View
- Controller

and each of them has specific responsibilities.



Model :

The Model component represents every bit of user-related data-related logic. Any additional business logic-related data or information that is transferred between the View and Controller components may fall under this category. For instance, a Customer object might get customer information from the database, modify it, and then update the information again or use it to render data.

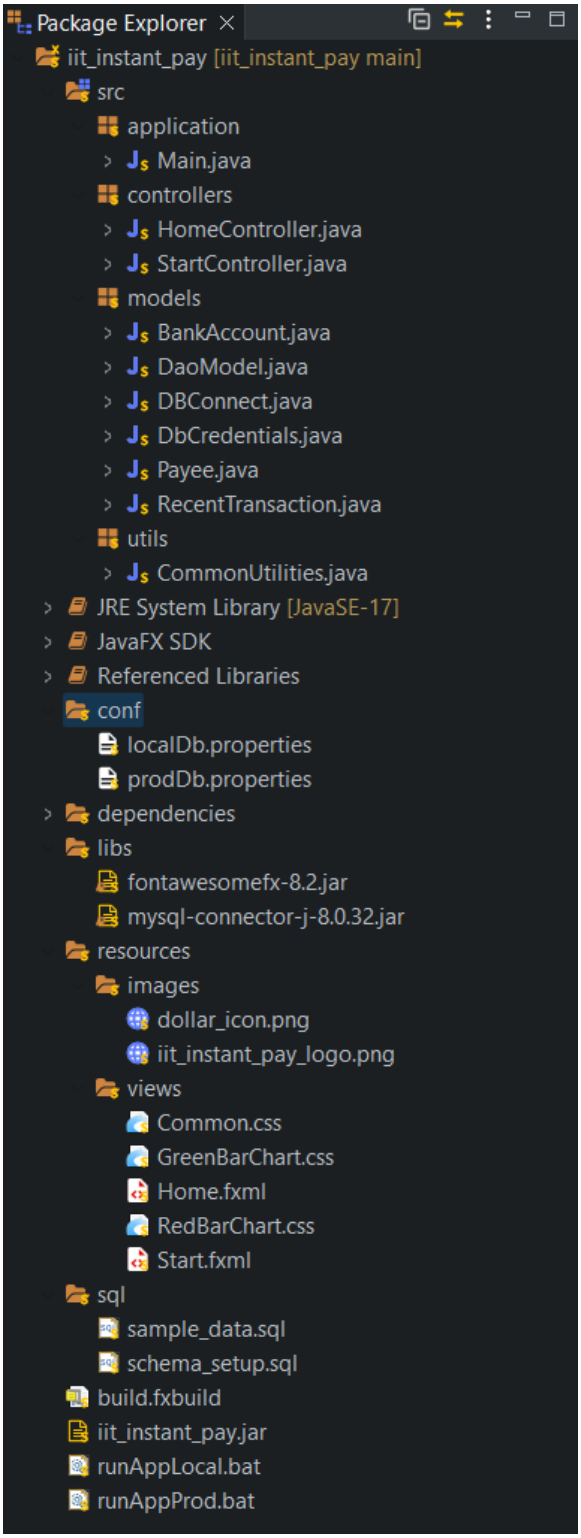
View :

Data representation falls under the purview of the view component. The user interface (UI) is actually made for the user by it. Therefore, all that comes to mind when you think of the view component in web applications is the HTML/CSS part. The view only interacts with the controller since the model component obtains the data it needs for views indirectly, through the controller.

Controller :

The controller, which acts as an intermediary and provides the connection between the views and the model, is known as the "main man" because of this. The controller simply needs to guide the model—it doesn't have to worry about handling data logic. Data from the model is processed before being sent to the view, where the user is given instructions on how to represent it. Views and models are unable to communicate directly.

Project Architecture:



- **src:** Contains Java source code.
    - **src/application:** Application entry point.
    - **src/controllers:** Controllers for tying together views and models.
    - **src/models:** Models for storing / retrieving data and accessing Database
    - **src/utils:** Common utilities
- **conf**: Configurations for database connection.
- **dependencies**: External frameworks for running application
- **libs**: External libraries for running application
- **resources**:
    - **views**: View definitions for UI
    - **images**: Image files used in UI
- **sql**: SQL scripts for initiating database schema and populating with test data.

Application Functionality:

Regular User:

- All users registered using the registration form are considered regular users / clients.
- When a regular user logs in, they are taken to the their homepage with two tabs:
    - Recent Transactions
    - Payees
- Users account id and current balance are shown on the left hand panel.
- “Recent Transactions” tab shows the recent debits / credits from the account.
- From the “Payees” tab, users can link other accounts as Payess and make payments to them.

- Users can use the “Reload” and “Sign out” buttons on the left hand pane to reload the information (to see new transactions etc) and logoff respectively.

Admin User:

- Admin users will have access to two tabs:
  - Analytics
  - Admin
- From the “Analytics” tab, admin users can use the visualizations to identify key metrics like the total cash reserves at the bank, largest accounts, users with most activity.
- From “Admin” tab, you can lookup accounts using account id and:
  - Update account details like name, email, balance or
  - Delete payees

Transaction:

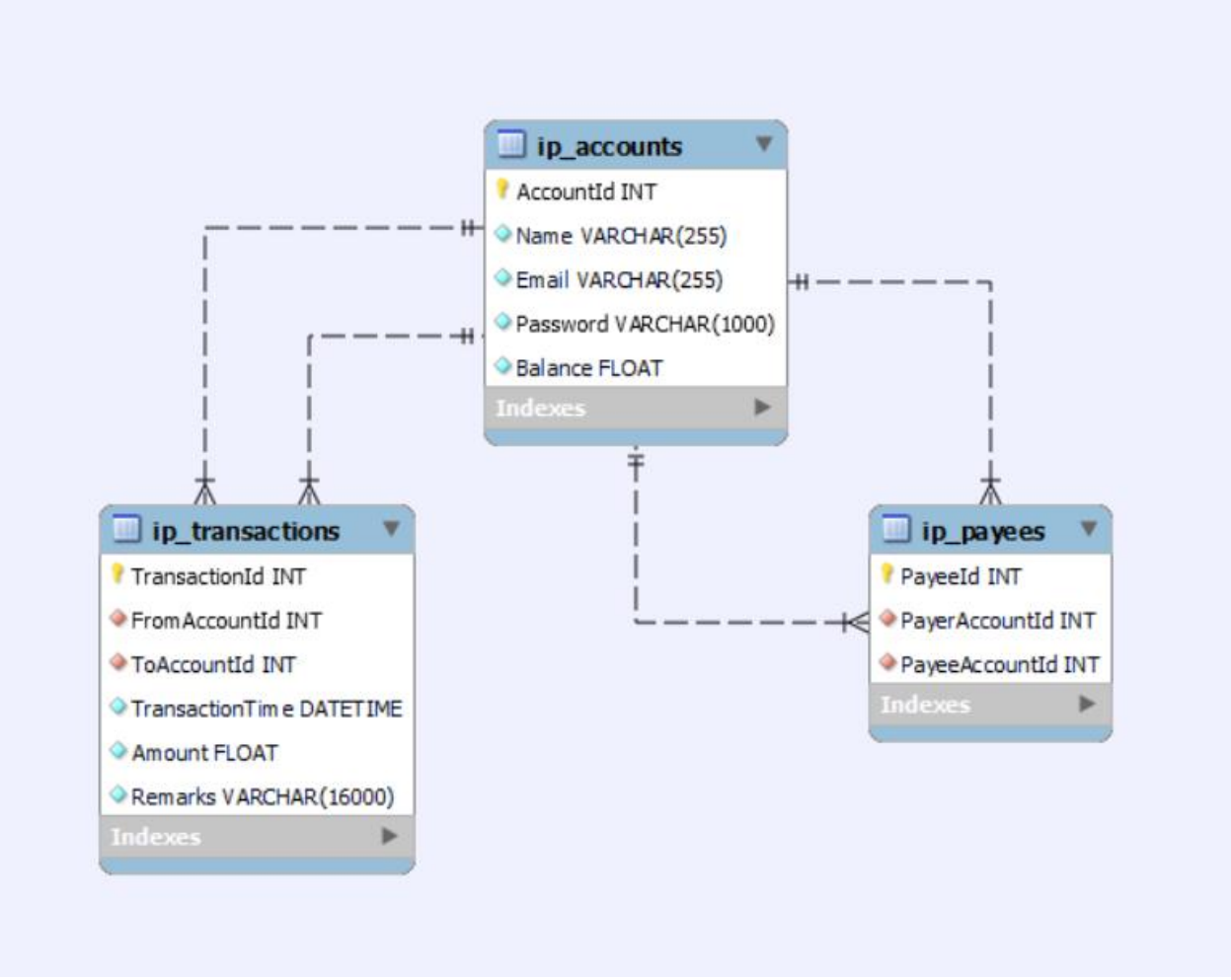
- When a user selects a payee and initiates a transaction, the information is passed to a Database Stored Procedure to actually perform the required database operations.
- All the required DML commands within the stored procedure are wrapped in a transaction to ensure that all DML commands are run or none of them are run. This will avoid scenarios where some DMLs are executed without others, resulting in money getting deducted from one account without actually getting deposited into the other.

Password Protection:

- Storing plain text passwords in a database is considered a security vulnerability, because anyone with access to table data will be able to easily access the passwords.

- To enhance the security of the application, passwords are encrypted using the SHA256 algorithm.
- When a user enters password at the login screen, the password is hashed during runtime and validated against the encrypted password stored in the database.
- Since SHA256 is a one-way algorithm, even if someone gets access to encrypted password, they will not be able to reverse engineer to get the real password.

Entity Relationship Diagram:



Running the Application

- **Compatibility:**
    - Project was created and tested using Java 17 on Windows 10.
    - Because the included JavaFX dependencies are Windows OS specific, the application is only compatible with Windows.
- In order to run the application, clone the Github repo: https://github.com/Shilpa206/iit_instant_pay
- Run the following batch script: *iit_instant_pay/iit_instant_pay/**runAppProd.bat***
- The batch script executes the JAR file after setting:
    - The class and module paths include JavaFX and other libraries like MySQL Connector.
    - Environment variables needed during run time.
- Application will dynamically pick the database credentials from conf/ directory path, which is set as environment variable in batch file.
- **Database:**
    - Before running the application the first time, you will need to set up the required database schema.
    - From the clone repo, go to **sql/** directory and execute the commands in **schema_setup.sql** on your MySQL database.
    - You can execute **sample_data.sql** to insert test data.

IMPORTANCE OF GIT

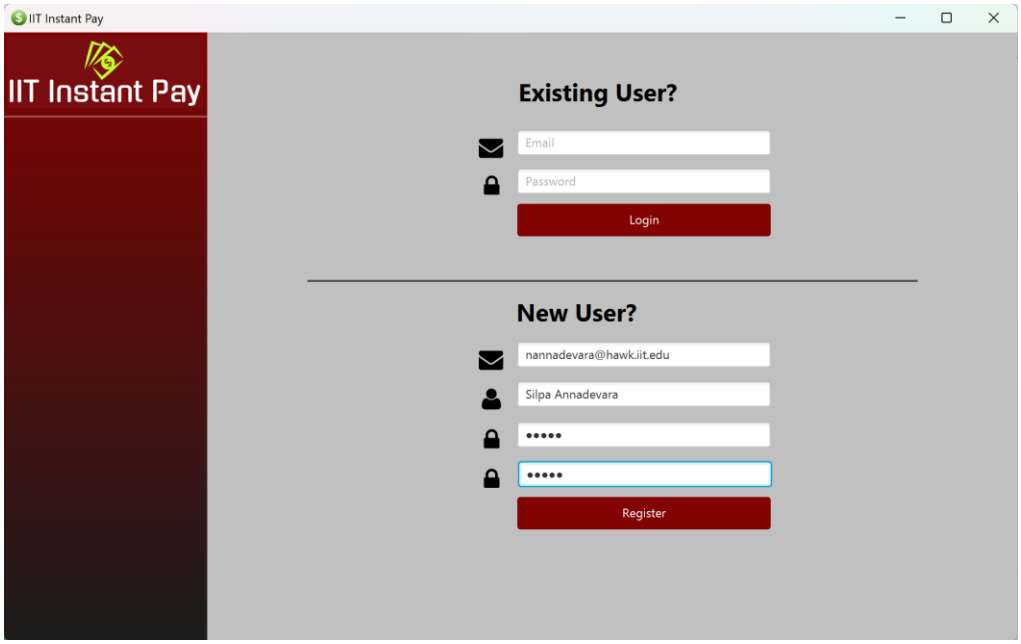
The name Git refers to the most well-known version control system. Git keeps track of the changes you make to files so that you may review previous revisions if necessary and have a record of what has been done. Git also makes collaboration easier by allowing several contributors' updates to be combined into a single repository. So whether you write code that only you will see or work in a team, Git will be useful for you.

Your computer keeps a copy of each of your files as well as their history. The files' revision histories can also be saved on internet servers (like GitHub or Bitbucket). If you have a central site where you can publish your modifications and download changes from others, you can collaborate with other developers more easily. Even two persons can work on separate parts of the same document simultaneously.

Git can automatically combine the changes, so you can integrate them without losing each other's work!

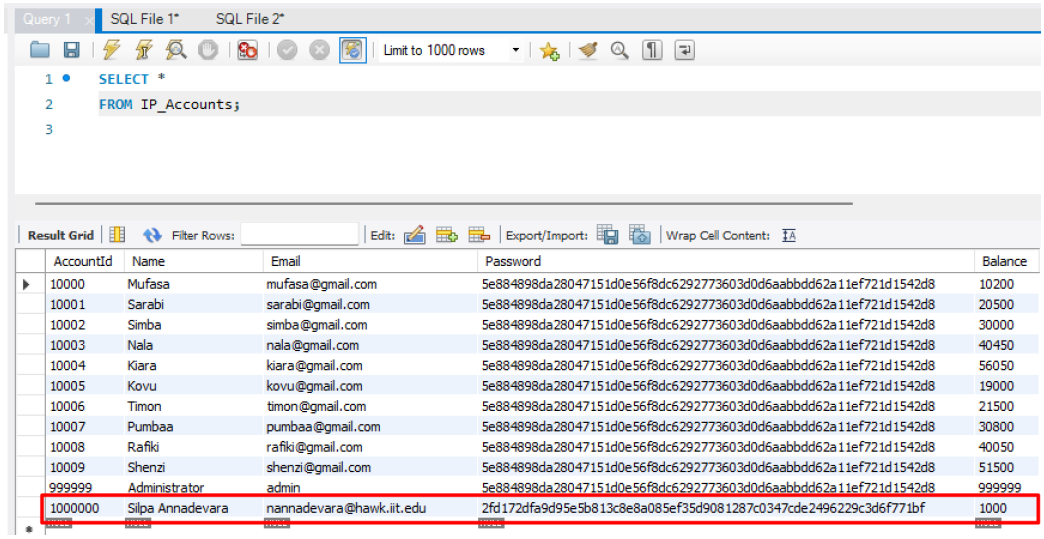
PHOTOS OF APP RUNNING

1) **New user registration**



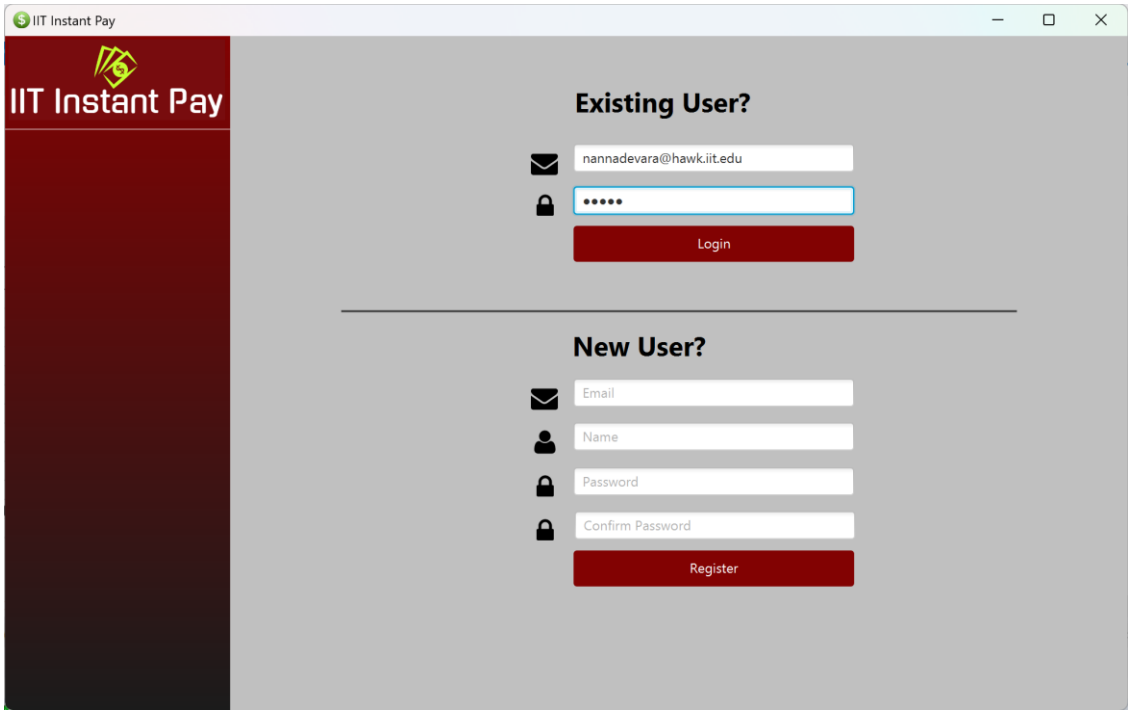
2) **New user details in DB:**

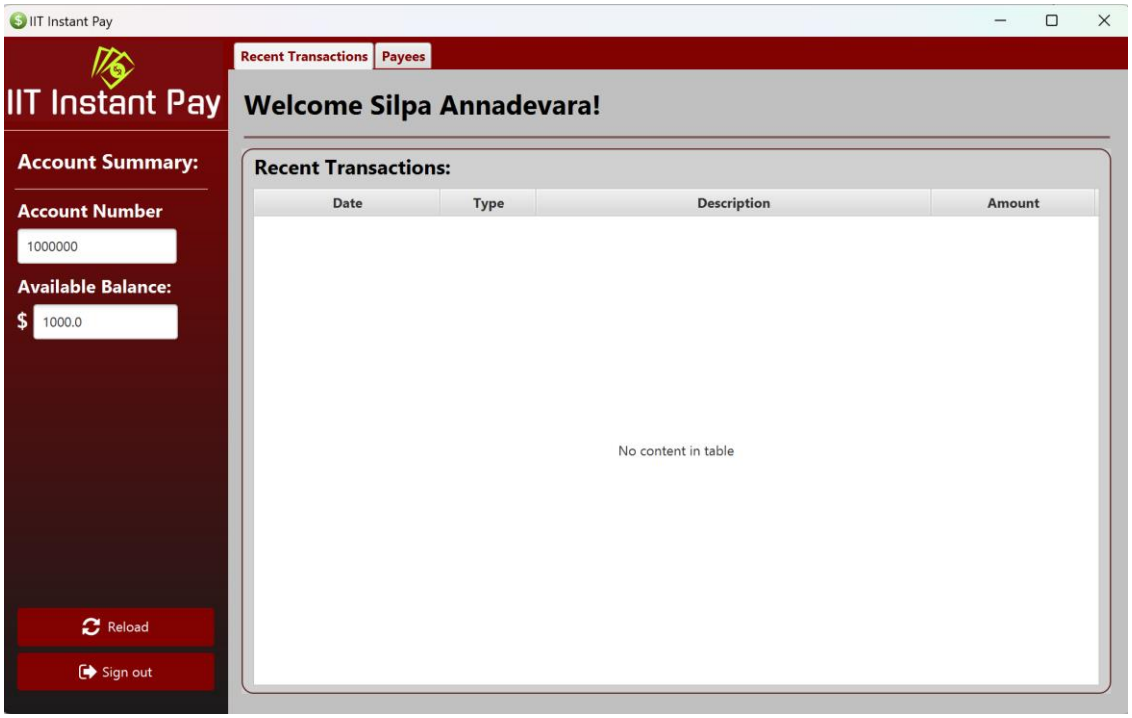
Note: User password is encrypted using SHA256

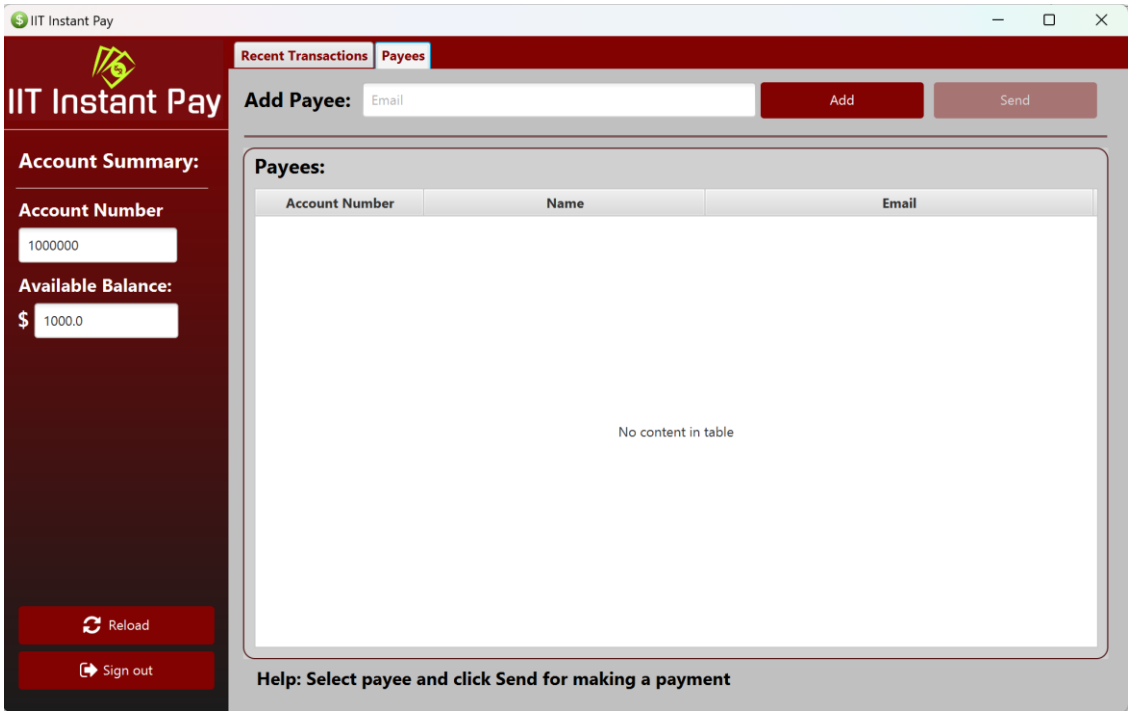


AccountId	Name	Email	Password	Balance
10000	Mufasa	mufasa@gmail.com	5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8	10200
10001	Sarabi	sarabi@gmail.com	5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8	20500
10002	Simba	simba@gmail.com	5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8	30000
10003	Nala	nala@gmail.com	5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8	40450
10004	Kiara	kiara@gmail.com	5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8	56050
10005	Kovu	kovu@gmail.com	5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8	19000
10006	Timon	timon@gmail.com	5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8	21500
10007	Pumbaa	pumbaa@gmail.com	5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8	30800
10008	Rafiki	rafiki@gmail.com	5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8	40050
10009	Shenzi	shenzi@gmail.com	5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8	51500
999999	Administrator	admin	5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8	999999
1000000	Silpa Annadevara	nannadevara@hawk.iit.edu	2fd172dfa9d95e5b813c8e8a085ef35d9081287c0347cde2496229c3d6f771bf	1000

3) **Showing a successful login for a regular user:**

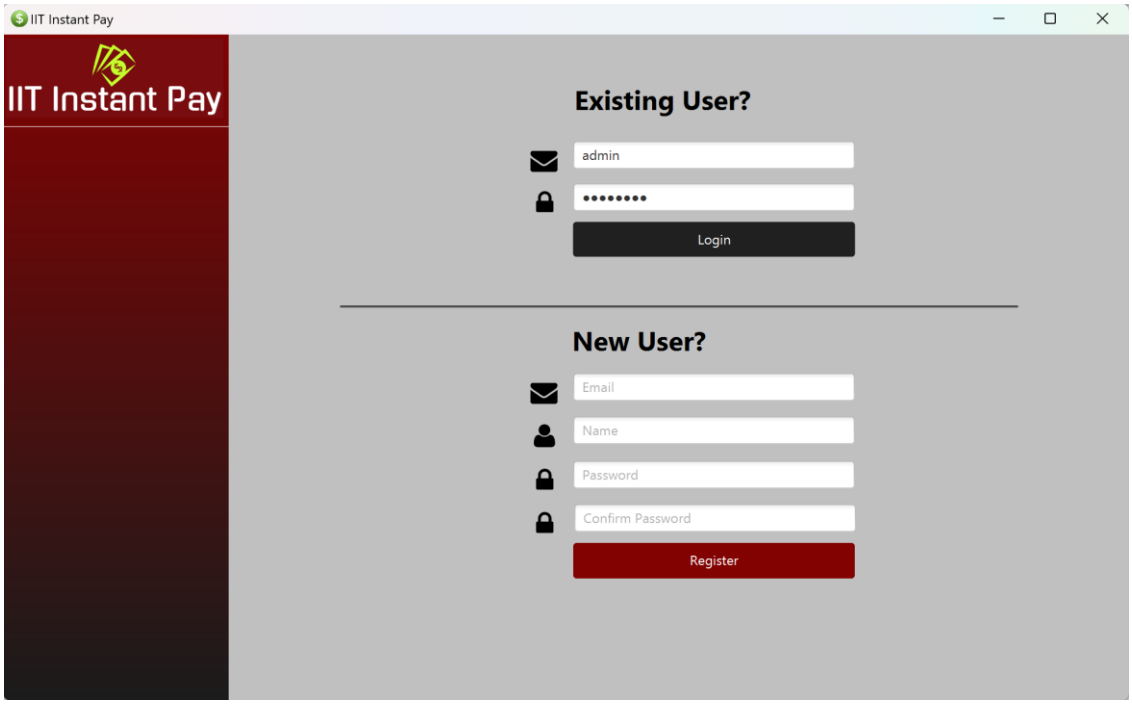




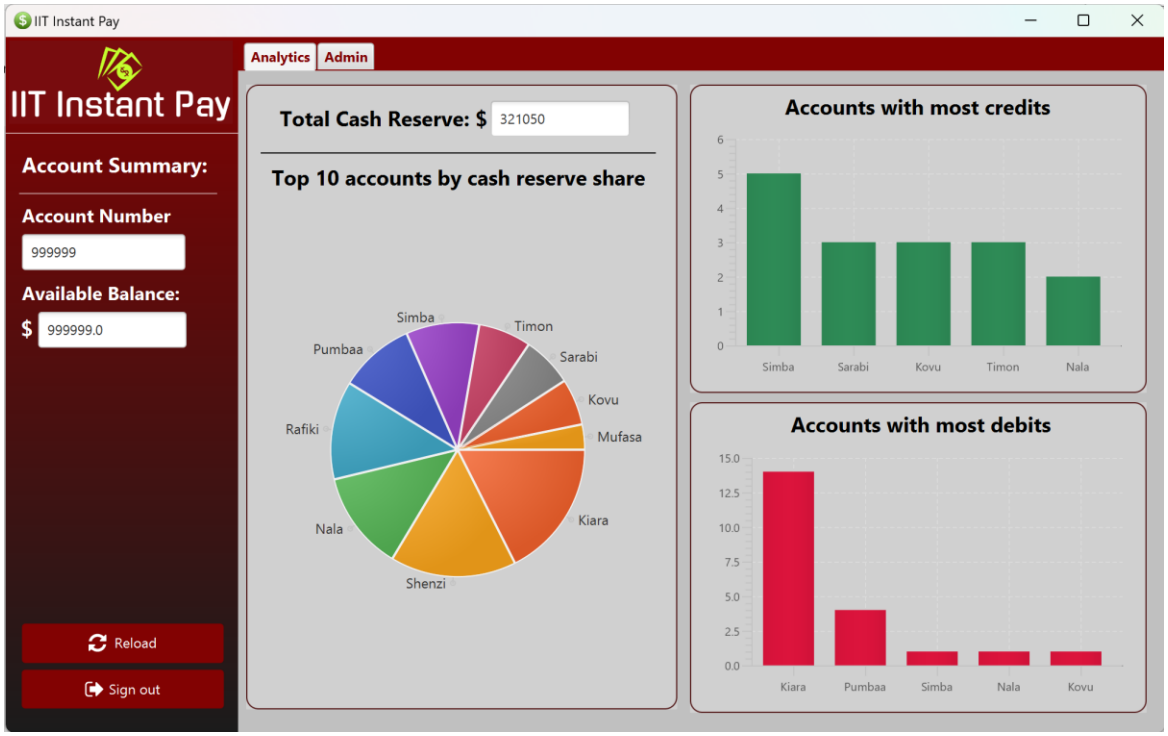


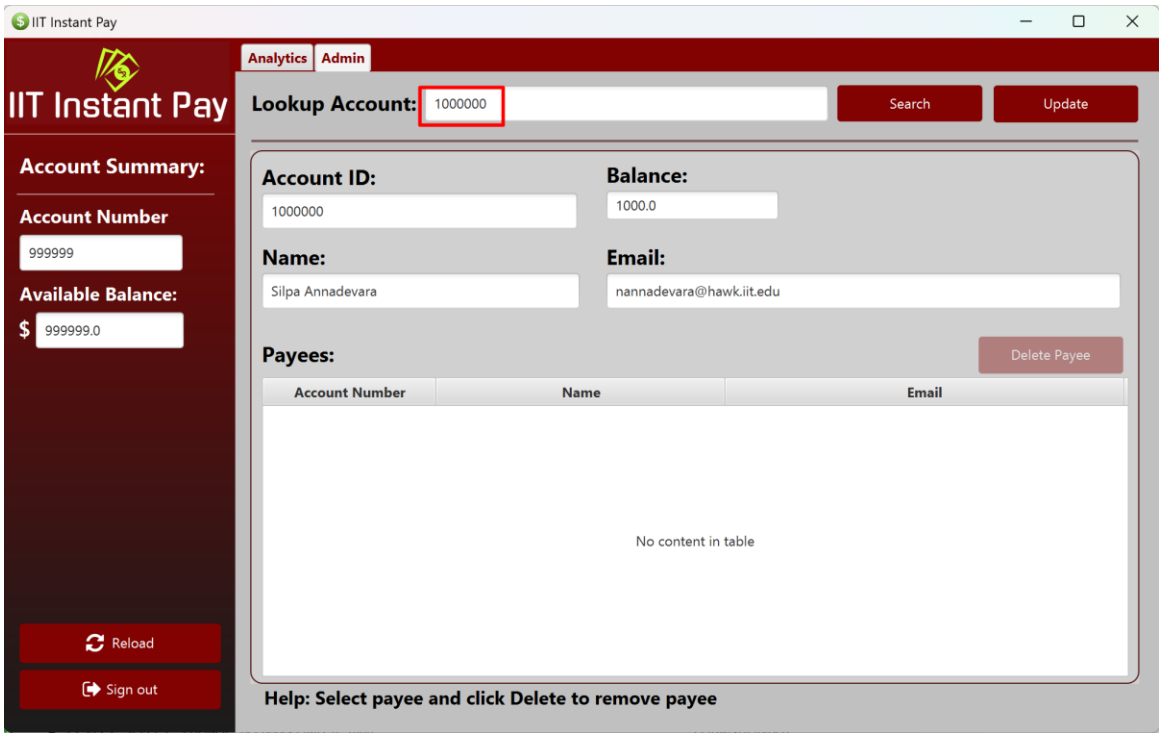
4) **Show a successful login for an <u>admin:</u>**

<u>Admin Credentials:</u> admin / password:



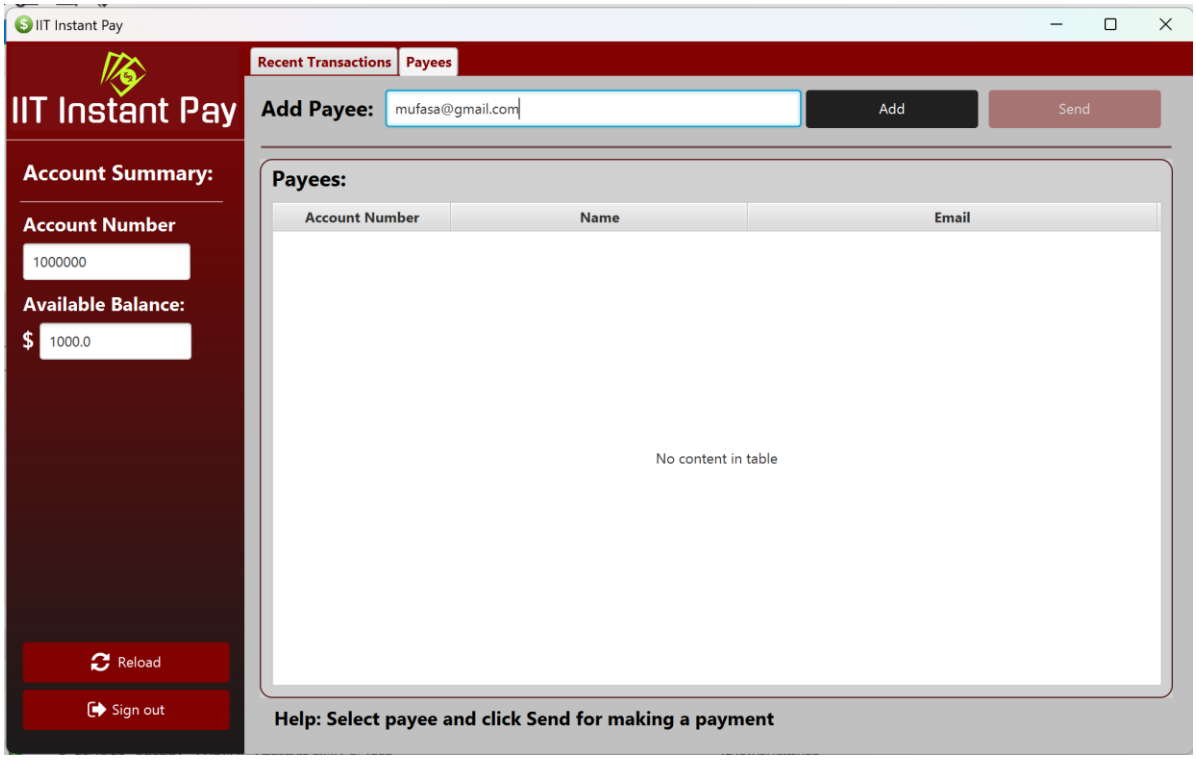
Admin will have two tabs: Analytics and Admin.
Analytics tab will visualizations showing the status of the system.
Admin tab can be used to search for bank account (using account id) and update (account details) / delete (payees) information.

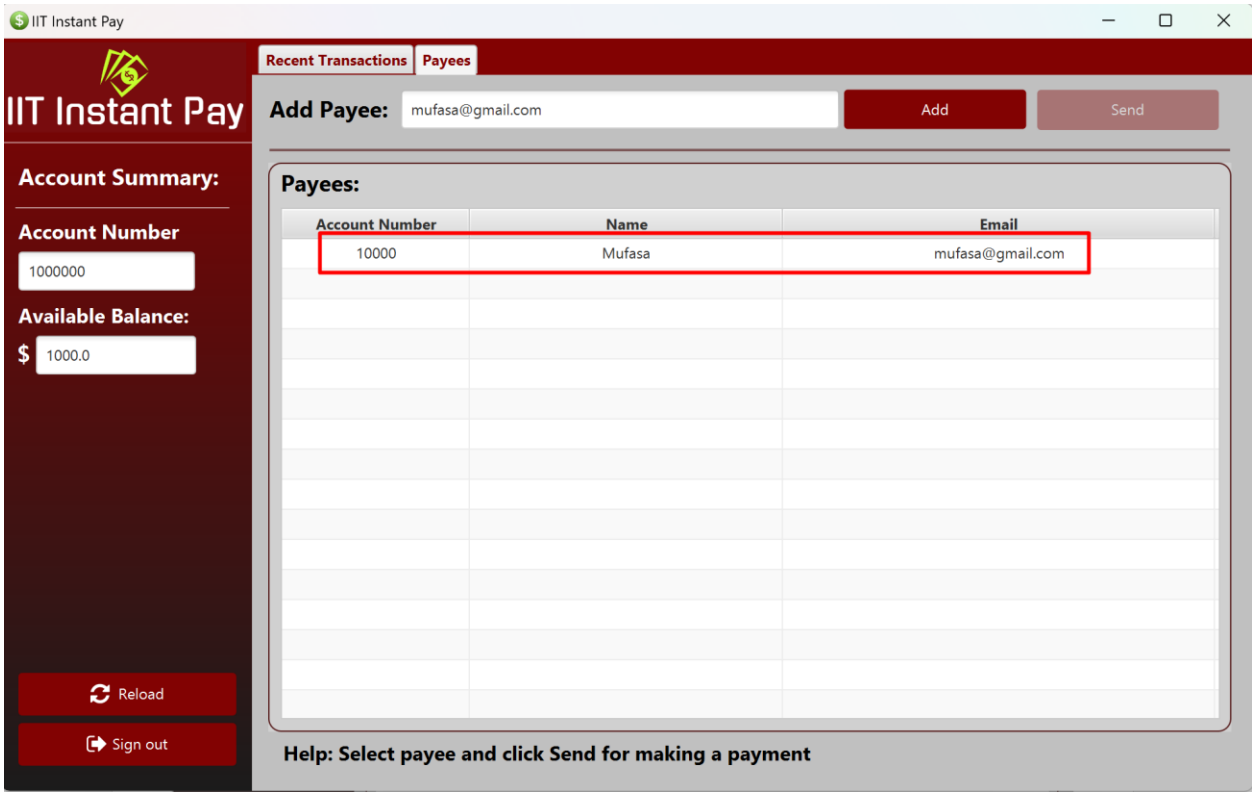




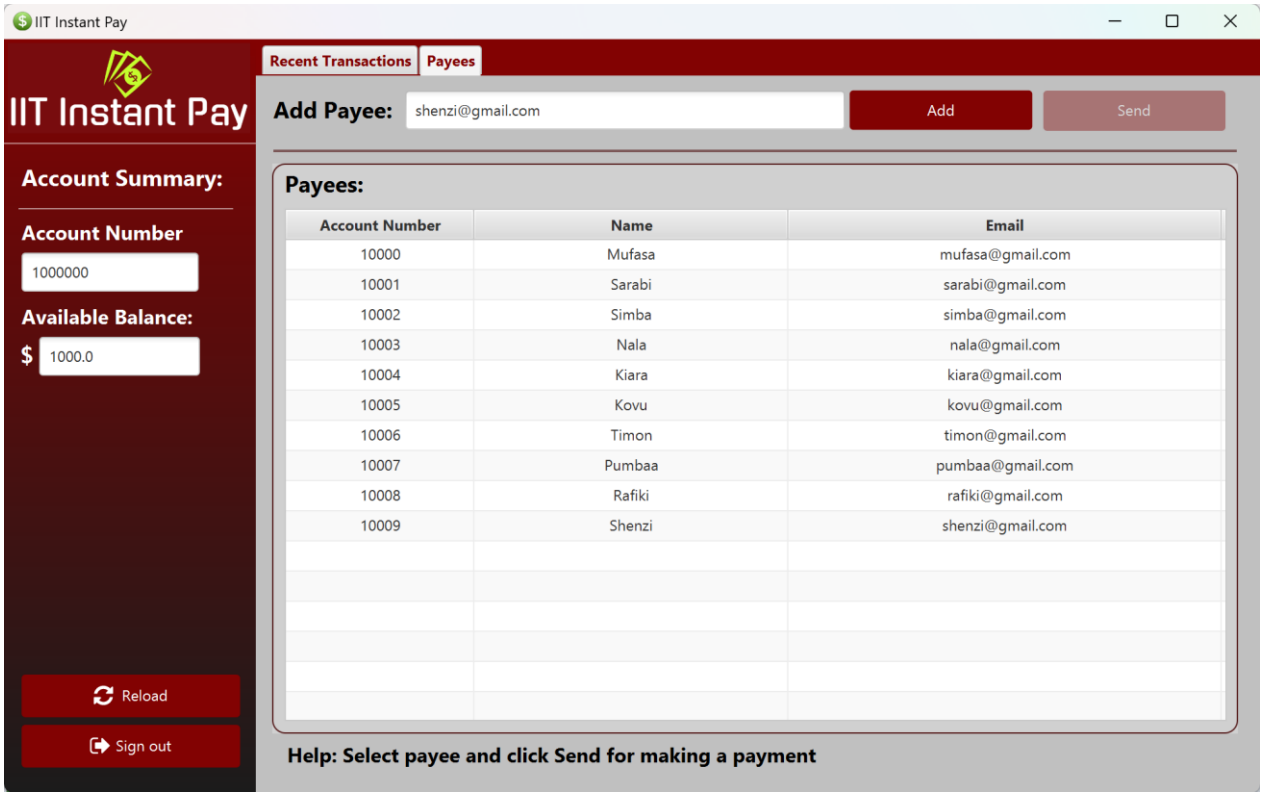
5) **Show a result of all 10 records added to your database <u>from the database itself</u>.**

Adding records to database by adding Payees to an account.
Adding Payee:

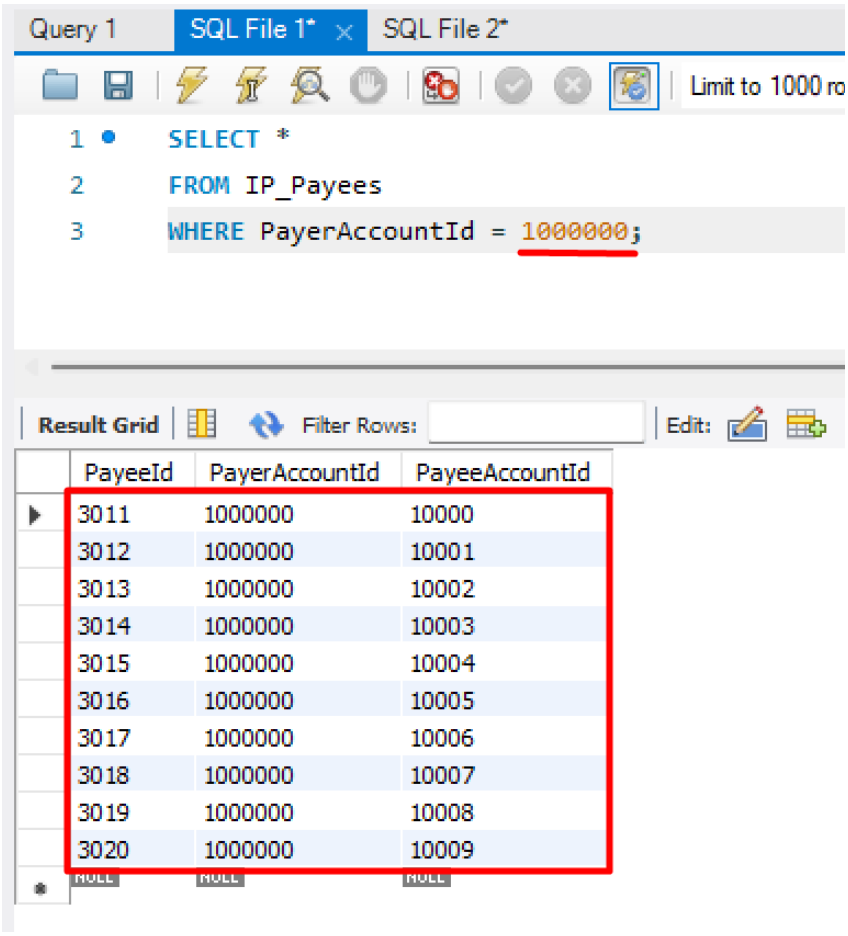




6) **After adding 9 more payees:**

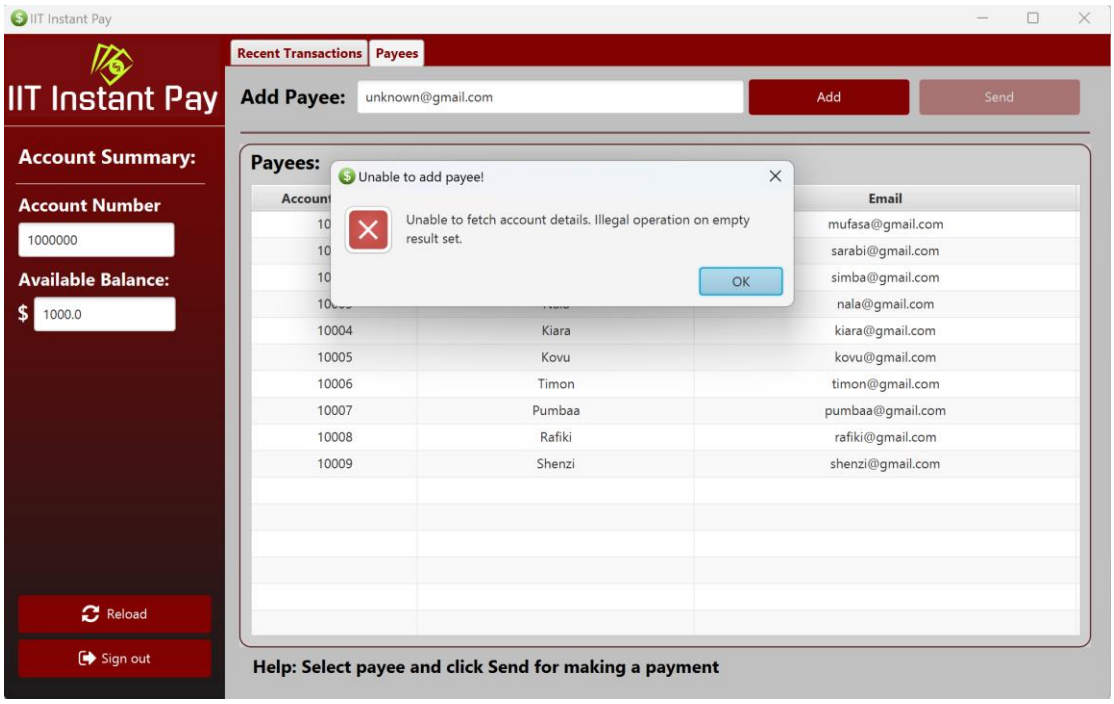


7) **Database Table**:

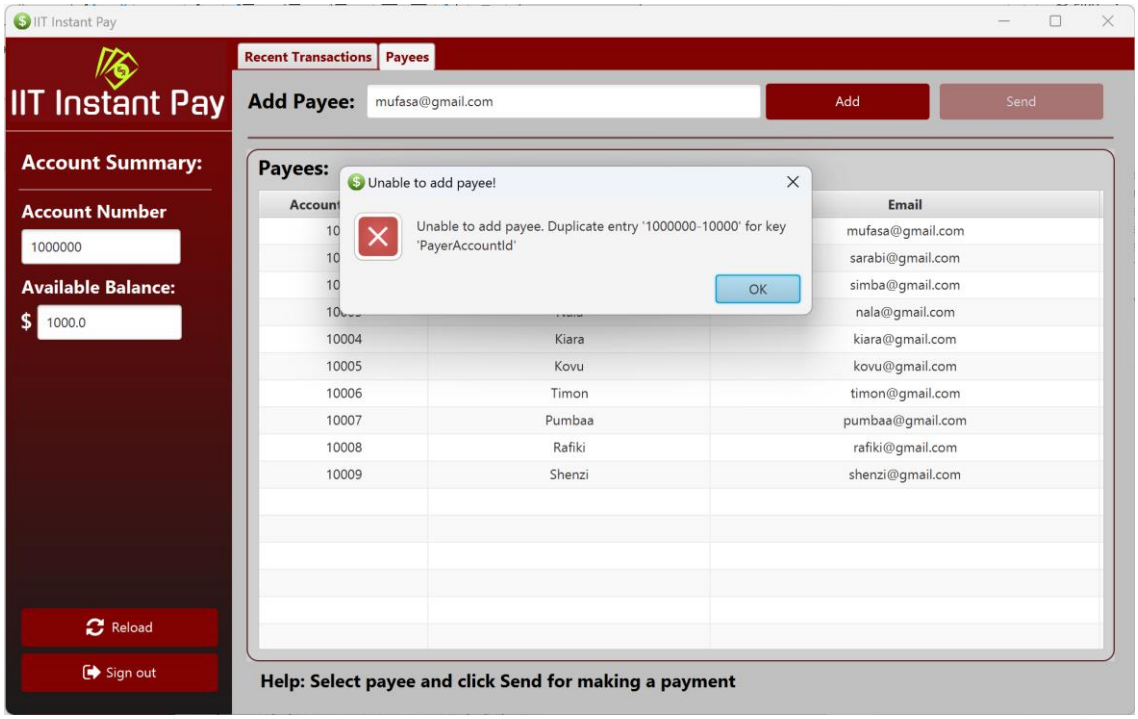


8) **Error Handling**:

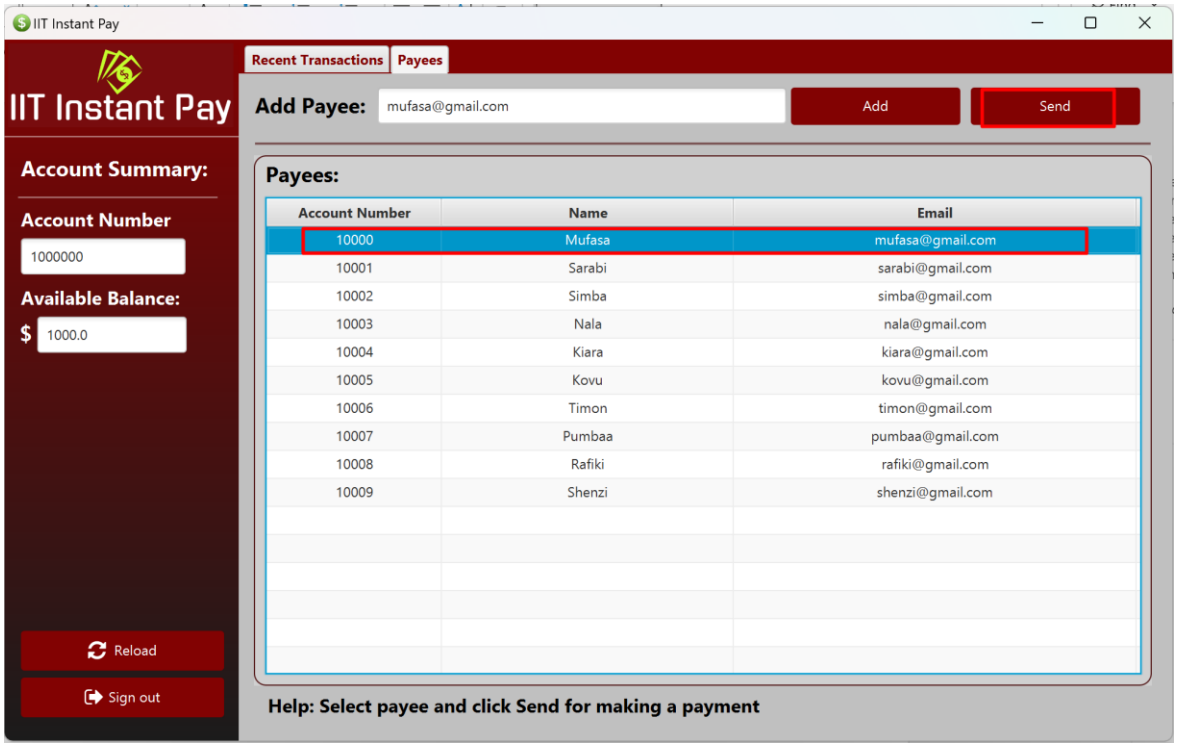
When adding unknown user:

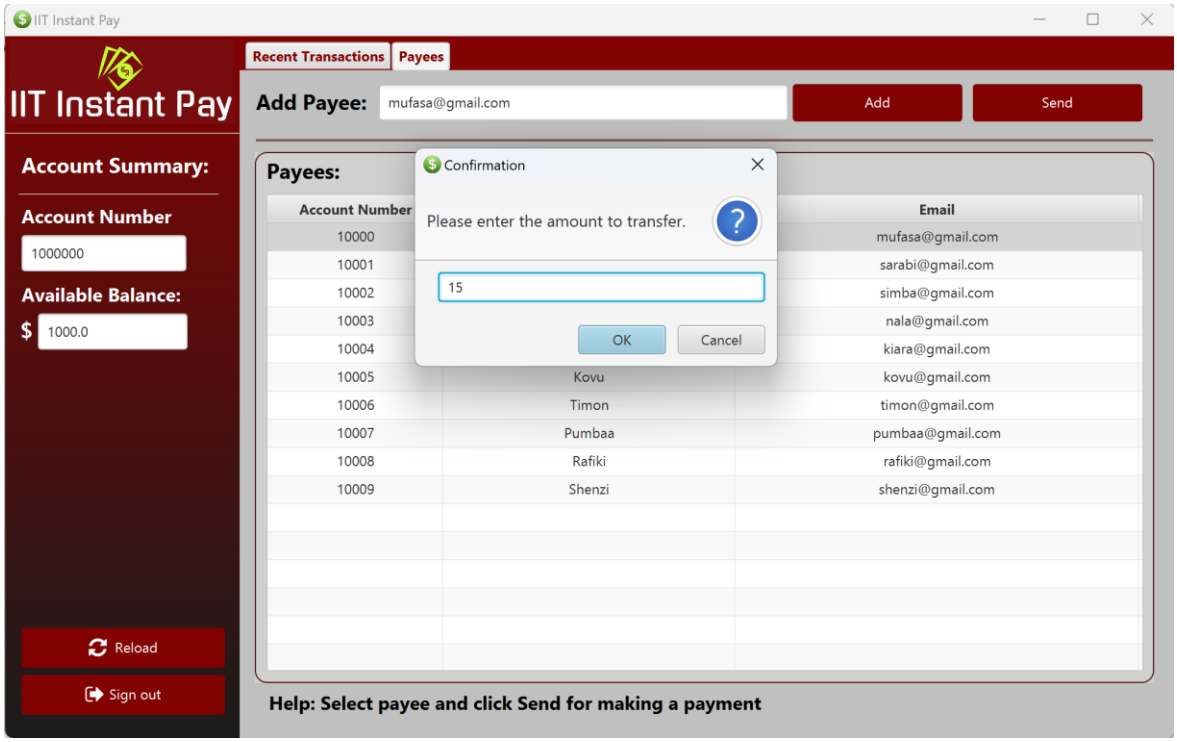


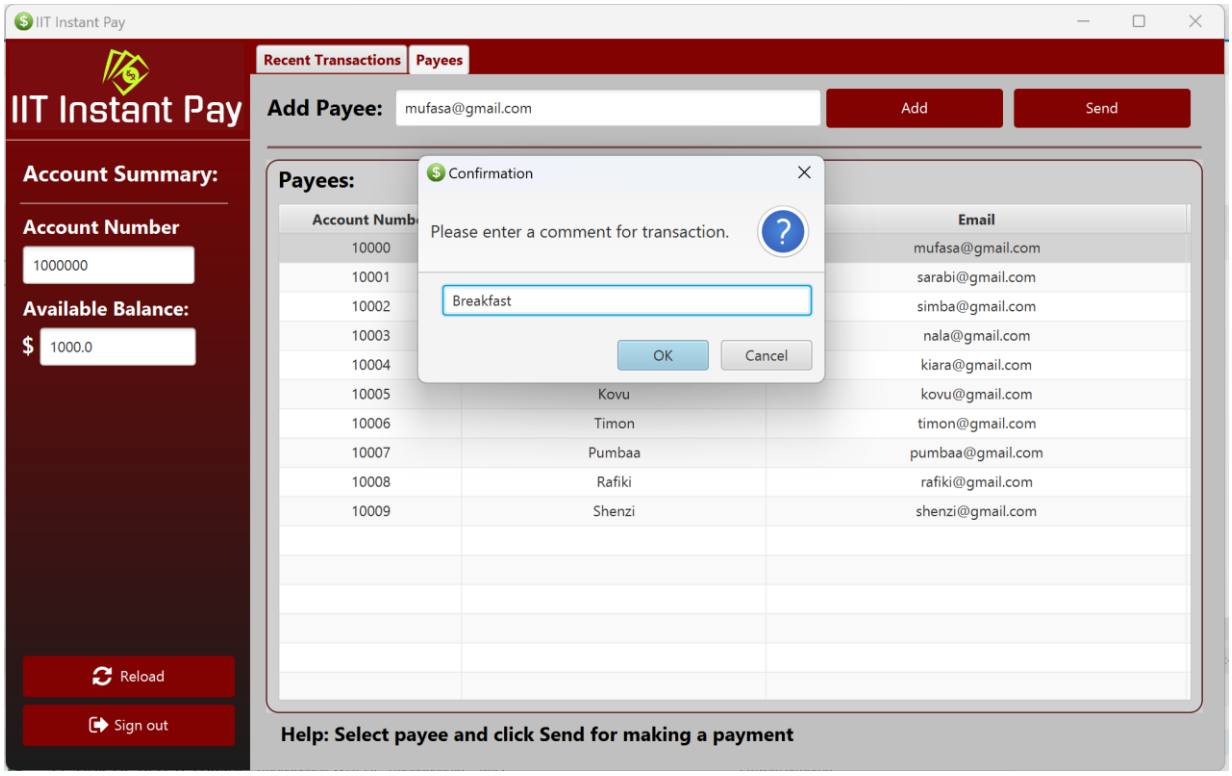
9) **When adding duplicate payee:**

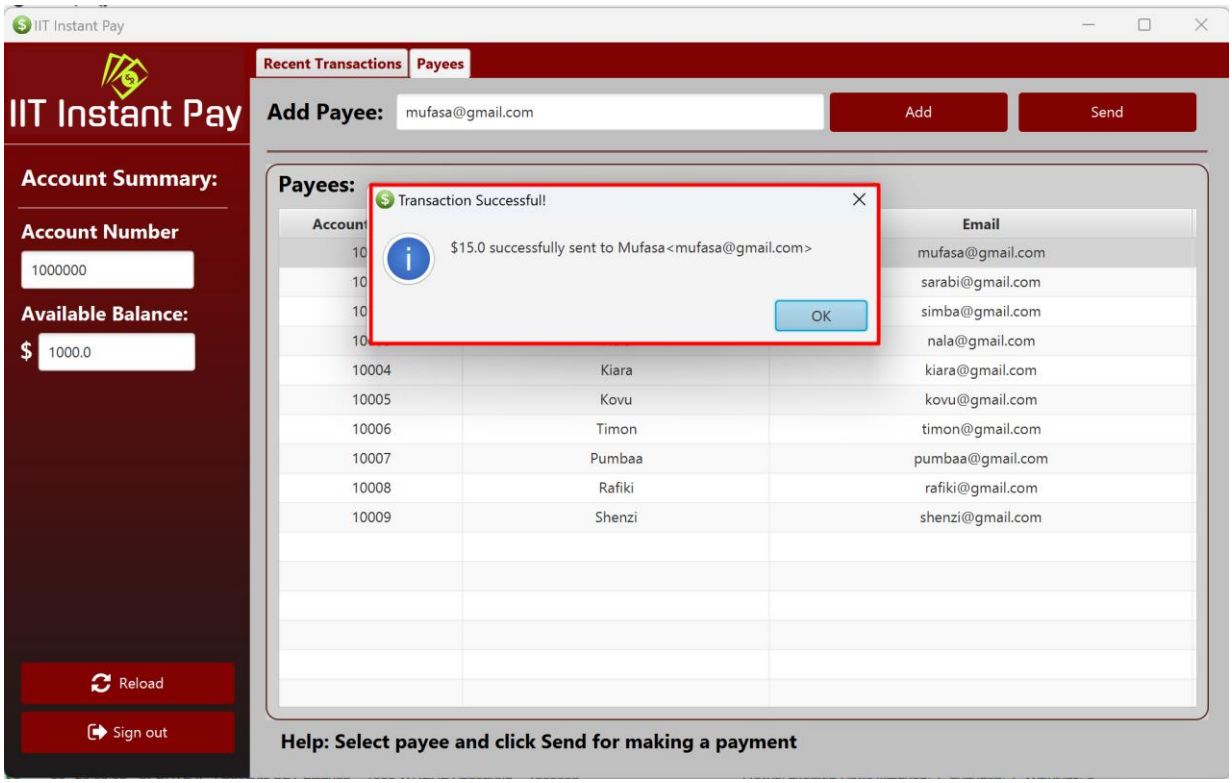


10)Sending money to a payee:



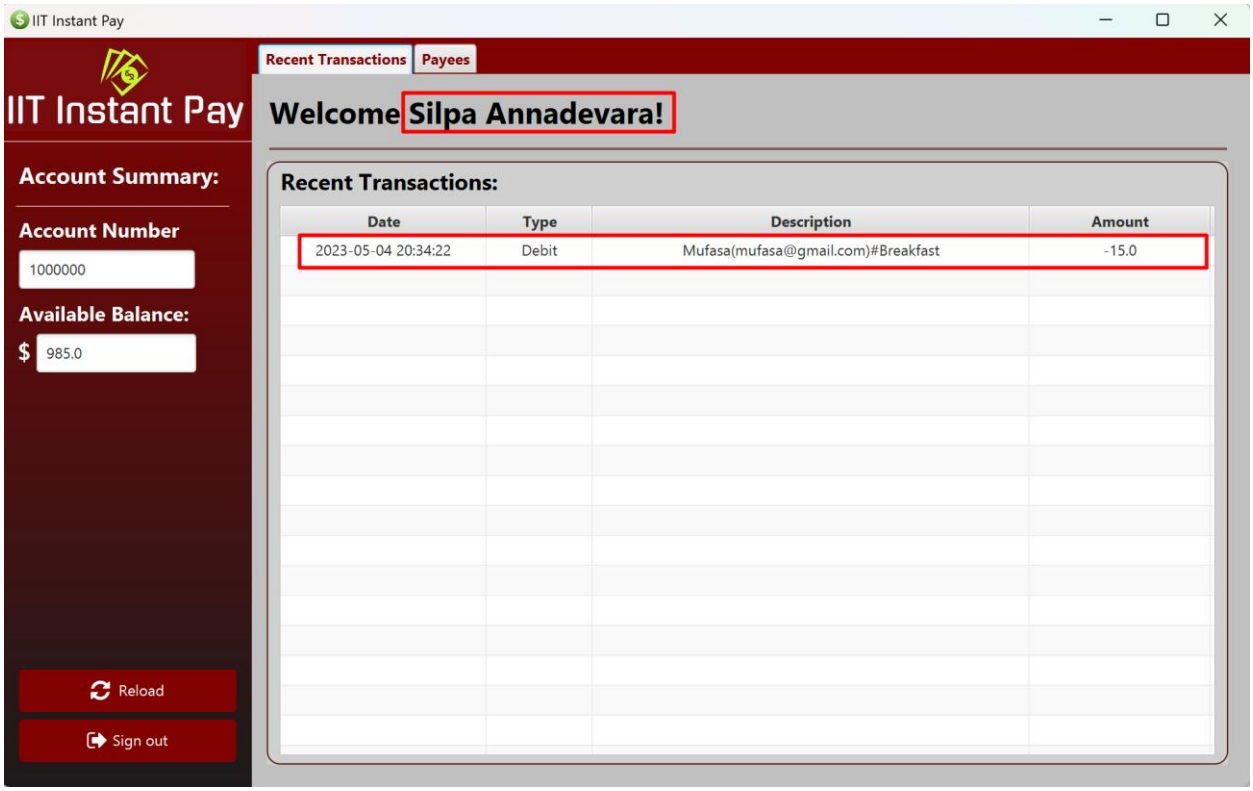



IIT Instant Pay
Recent Transactions
Payees
IIT Instant Pay
Add Payee:
mufasa@gmail.com
Add
Send
Account Summary:
Account Number
1000000
Available Balance:
$
1000.0
Payees:
Confirmation
Please enter a comment for transaction.
Breakfast
OK
Cancel
Account Numb
Email
10000
mufasa@gmail.com
10001
sarabi@gmail.com
10002
simba@gmail.com
10003
nala@gmail.com
10004
kiara@gmail.com
10005
Kovu
kovu@gmail.com
10006
Timon
timon@gmail.com
10007
Pumbaa
pumbaa@gmail.com
10008
Rafiki
rafiki@gmail.com
10009
Shenzi
shenzi@gmail.com
Reload
Sign out
Help: Select payee and click Send for making a payment

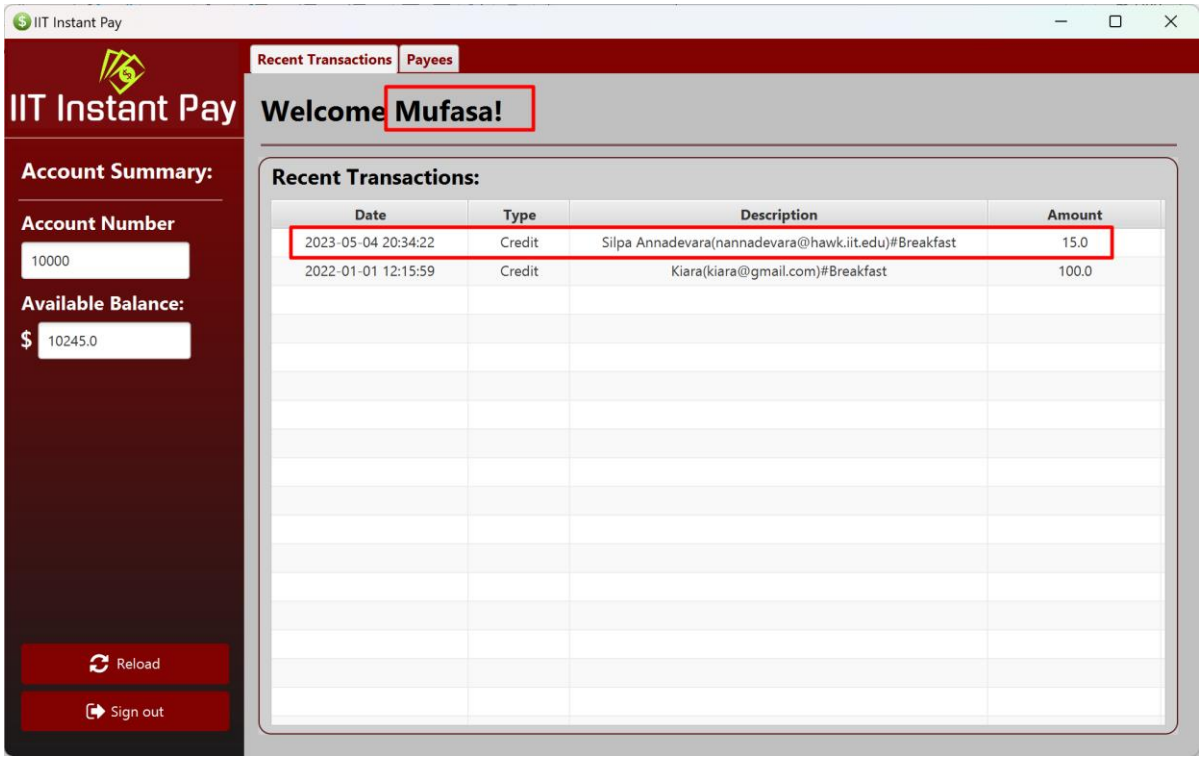

IIT Instant Pay
Recent Transactions
Payees
IIT Instant Pay
Add Payee:
mufasa@gmail.com
Add
Send
Account Summary:
Account Number
1000000
Available Balance:
$
1000.0
Payees:
Transaction Successful!
$15.0 successfully sent to Mufasa<mufasa@gmail.com>
OK
Email
mufasa@gmail.com
sarabi@gmail.com
simba@gmail.com
nala@gmail.com
10004
Kiara
kiara@gmail.com
10005
Kovu
kovu@gmail.com
10006
Timon
timon@gmail.com
10007
Pumbaa
pumbaa@gmail.com
10008
Rafiki
rafiki@gmail.com
10009
Shenzi
shenzi@gmail.com
Reload
Sign out
Help: Select payee and click Send for making a payment

11)Transaction will show up under recent transaction of both payer and payee as debit and credit respectively:

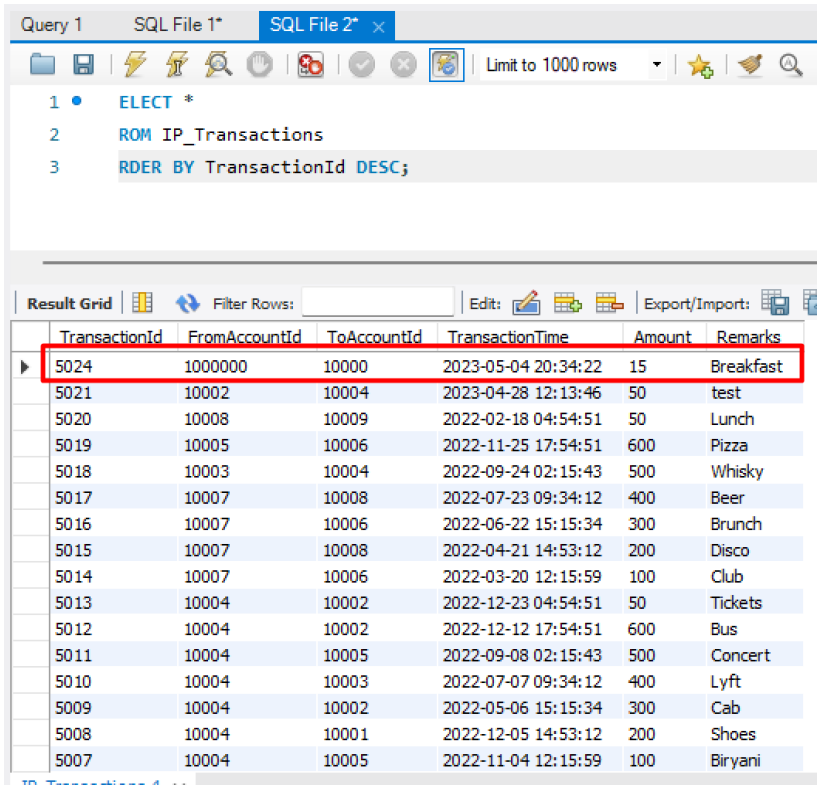
Payer:



Payee:

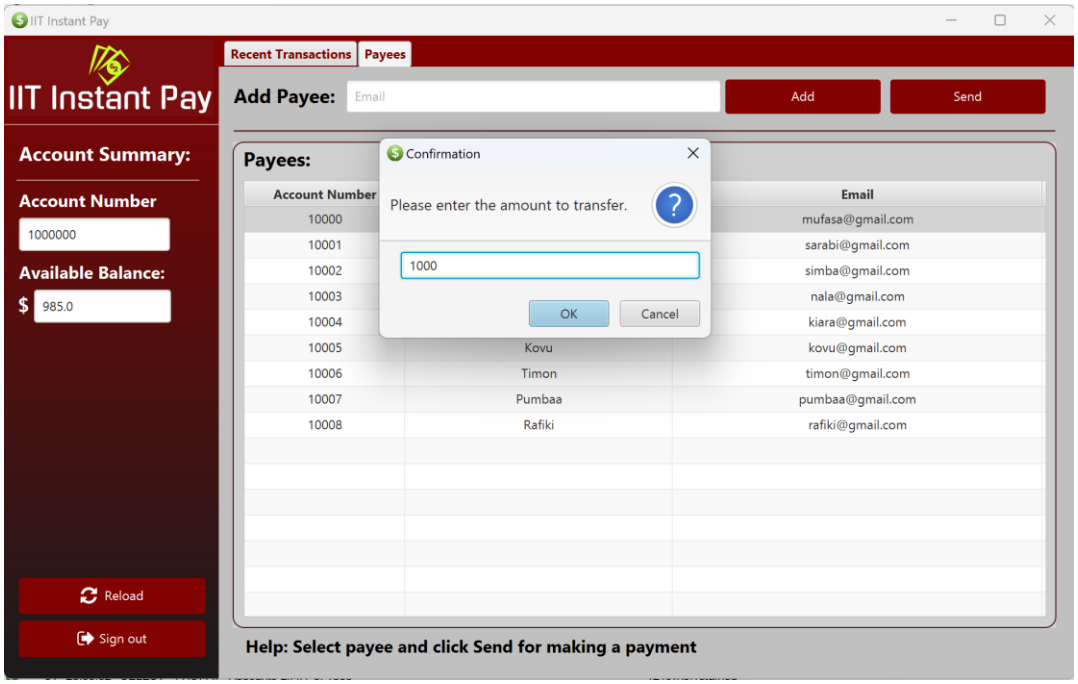


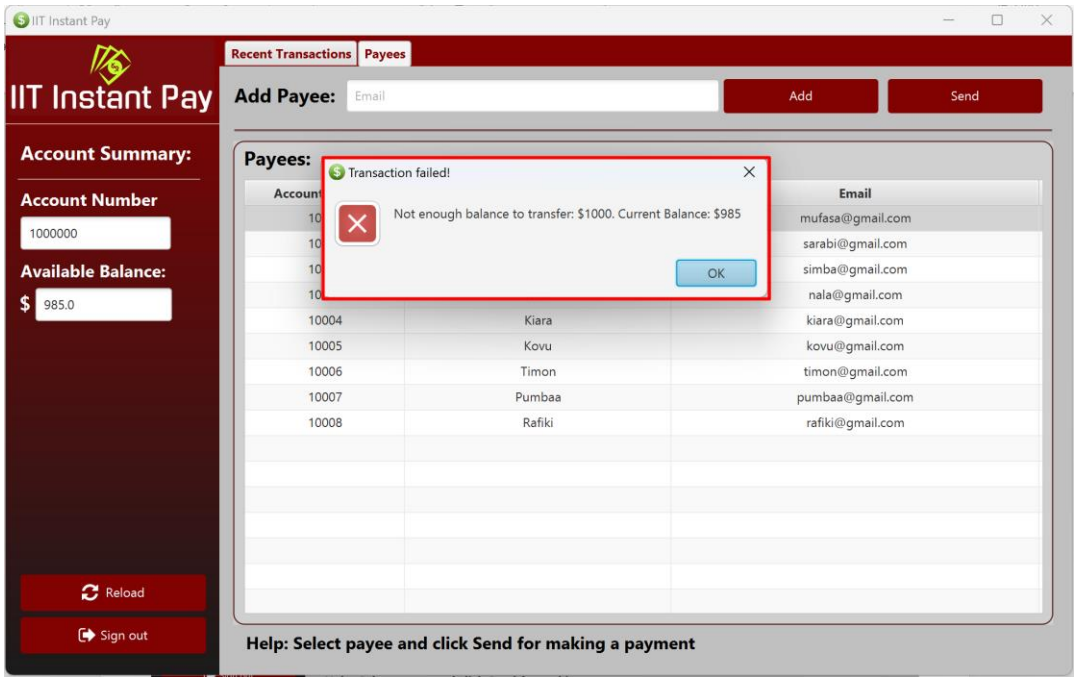
12) Database Table for transactions:



TransactionId	FromAccountId	ToAccountId	TransactionTime	Amount	Remarks
5024	1000000	10000	2023-05-04 20:34:22	15	Breakfast
5021	10002	10004	2023-04-28 12:13:46	50	test
5020	10008	10009	2022-02-18 04:54:51	50	Lunch
5019	10005	10006	2022-11-25 17:54:51	600	Pizza
5018	10003	10004	2022-09-24 02:15:43	500	Whisky
5017	10007	10008	2022-07-23 09:34:12	400	Beer
5016	10007	10006	2022-06-22 15:15:34	300	Brunch
5015	10007	10008	2022-04-21 14:53:12	200	Disco
5014	10007	10006	2022-03-20 12:15:59	100	Club
5013	10004	10002	2022-12-23 04:54:51	50	Tickets
5012	10004	10002	2022-12-12 17:54:51	600	Bus
5011	10004	10005	2022-09-08 02:15:43	500	Concert
5010	10004	10003	2022-07-07 09:34:12	400	Lyft
5009	10004	10002	2022-05-06 15:15:34	300	Cab
5008	10004	10001	2022-12-05 14:53:12	200	Shoes
5007	10004	10005	2022-11-04 12:15:59	100	Biryani

13) Trying to send money you don't have:





14) Show some update performed by the <u>admin</u> for the first record entered into the db:

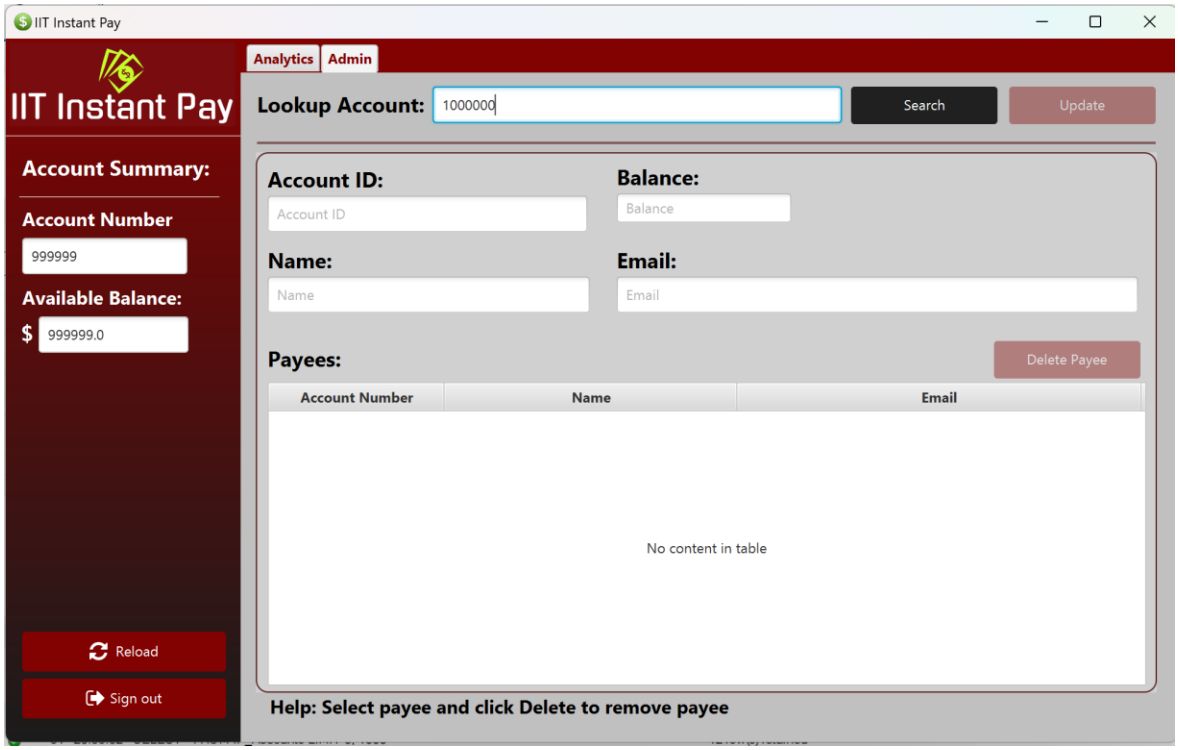
Only admin user can update or delete record.
Admin can update account details like email address, name, balance.

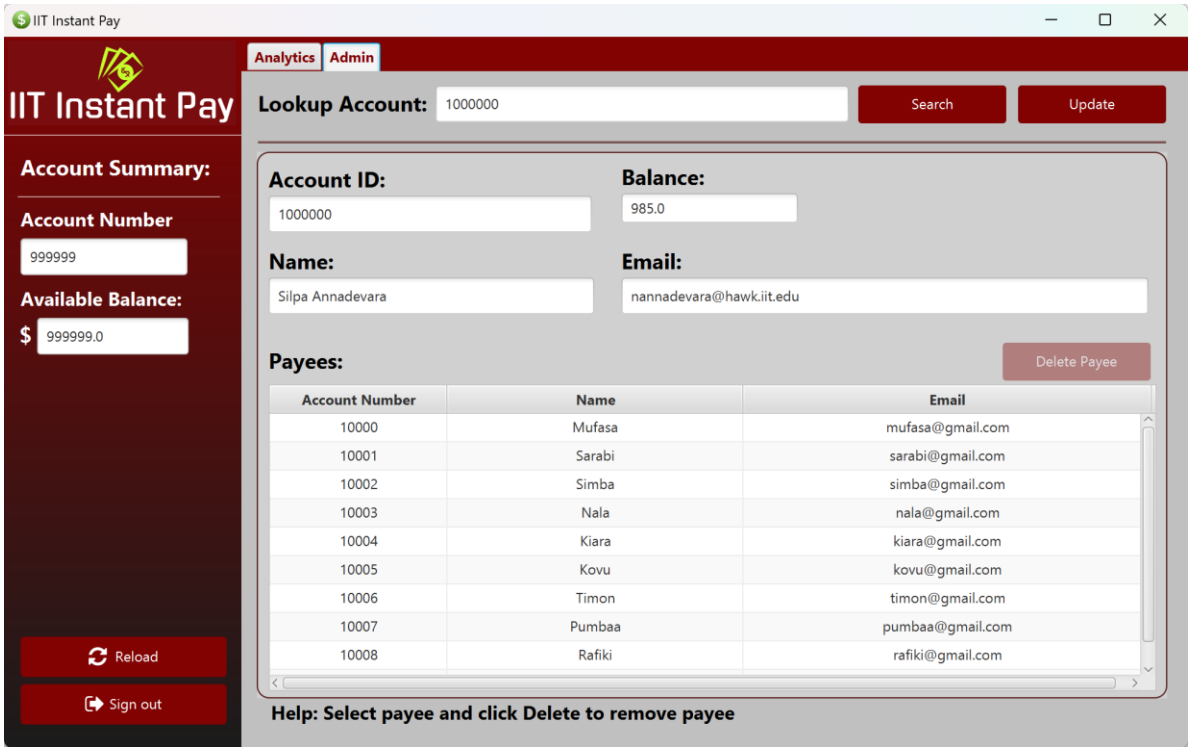
Admin can delete payees of any user.

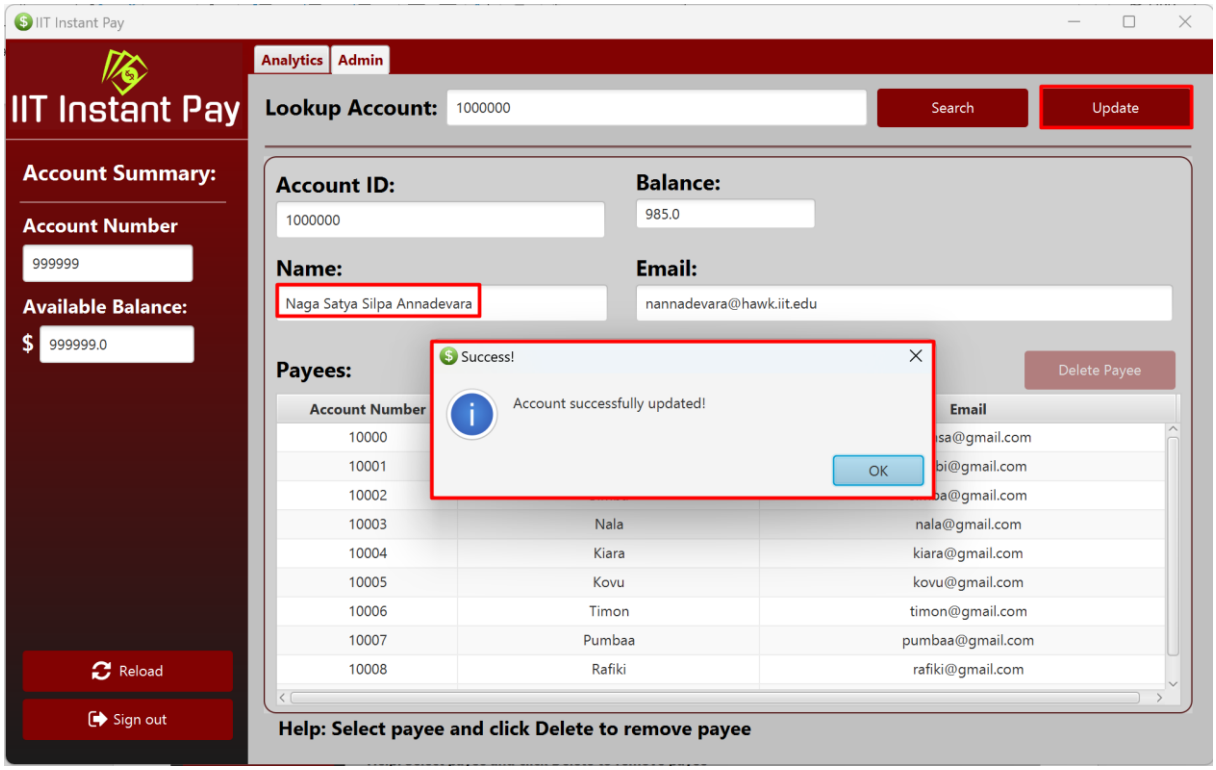
15) Update Account Information:

After logging in as admin.

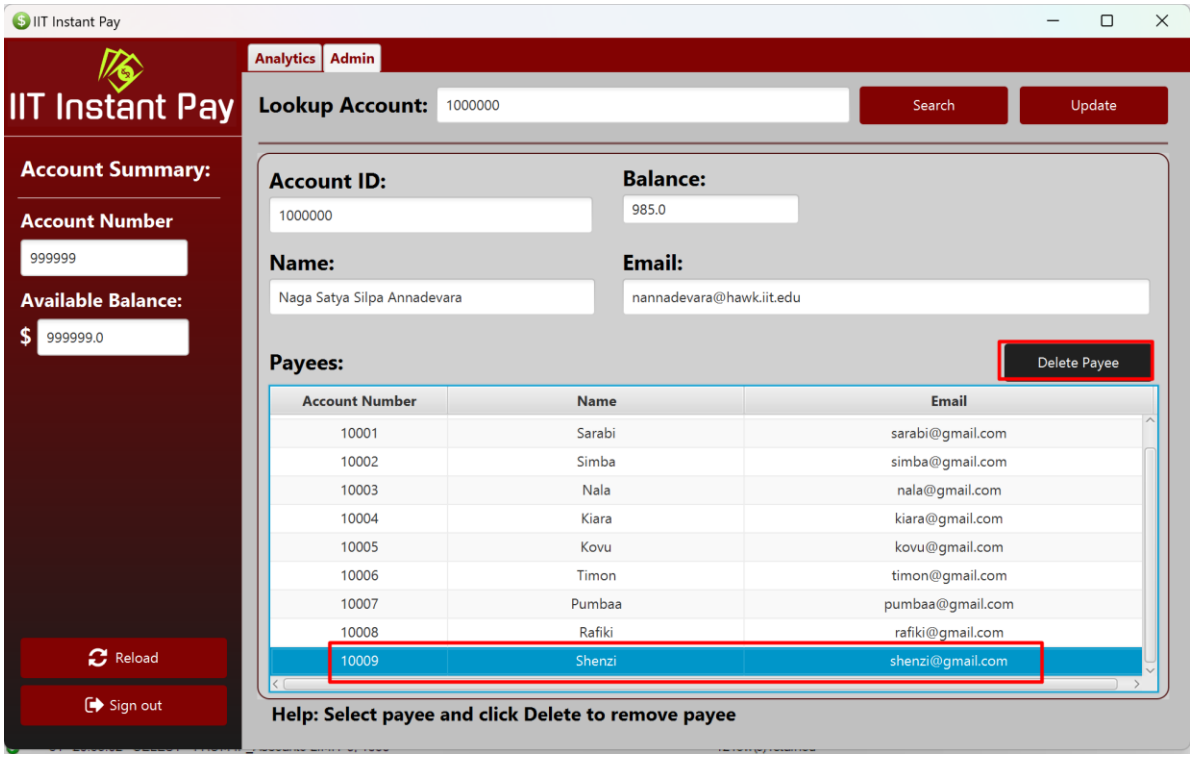
Lookup account:

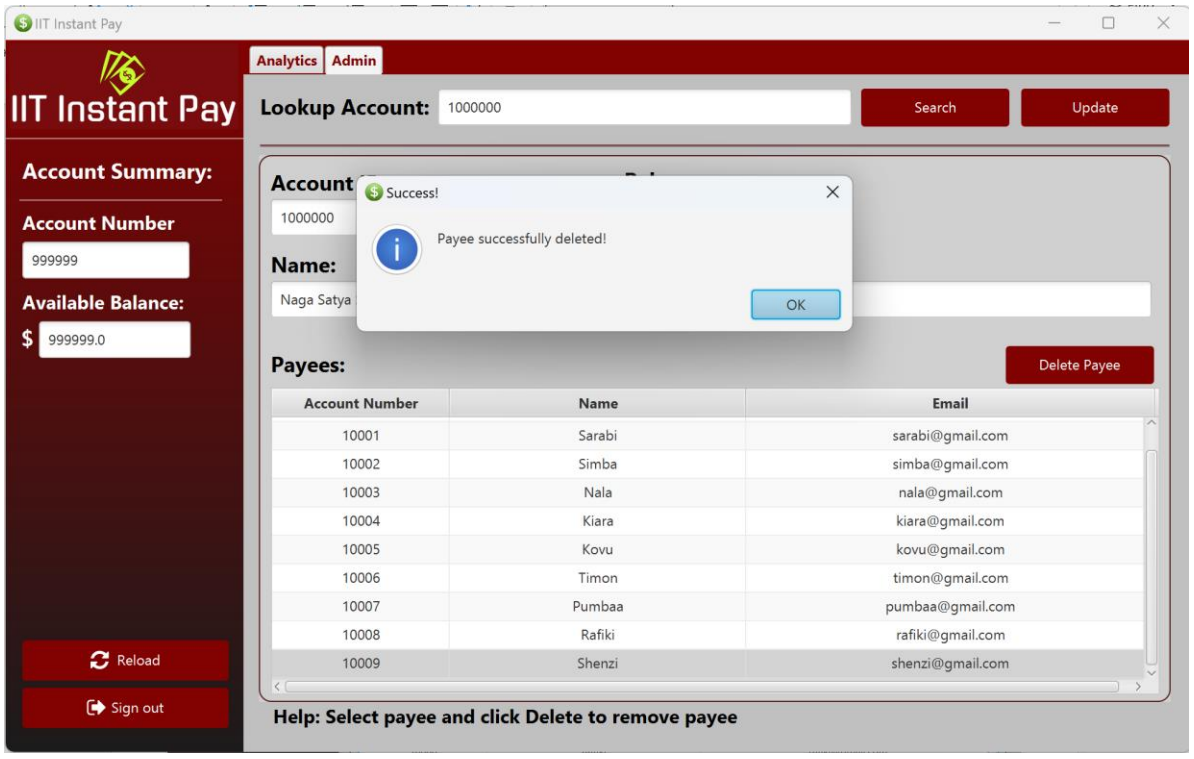


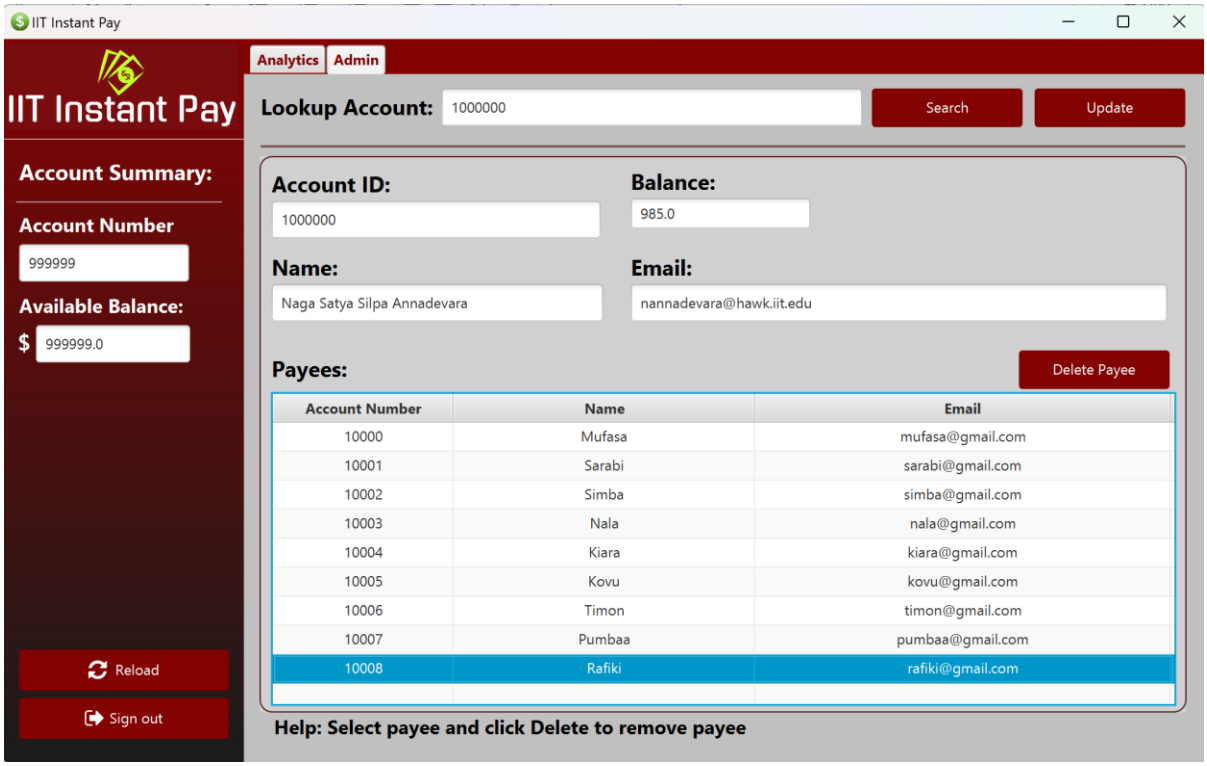

IIT Instant Pay
Analytics
Admin
IIT Instant Pay
Lookup Account:
1000000
Search
Update
Account Summary:
Account Number
999999
Available Balance:
$
999999.0
Account ID:
1000000
Balance:
985.0
Name:
Silpa Annadevara
Email:
nannadevara@hawk.iit.edu
Payees:
Delete Payee
Account Number
Name
Email
10000
Mufasa
mufasa@gmail.com
10001
Sarabi
sarabi@gmail.com
10002
Simba
simba@gmail.com
10003
Nala
nala@gmail.com
10004
Kiara
kiara@gmail.com
10005
Kovu
kovu@gmail.com
10006
Timon
timon@gmail.com
10007
Pumbaa
pumbaa@gmail.com
10008
Rafiki
rafiki@gmail.com
Reload
Sign out
Help: Select payee and click Delete to remove payee


IIT Instant Pay
Analytics
Admin
IIT Instant Pay
Lookup Account:
1000000
Search
Update
Account Summary:
Account Number
999999
Available Balance:
$
999999.0
Account ID:
1000000
Balance:
985.0
Name:
Naga Satya Silpa Annadevara
Email:
nannadevara@hawk.iit.edu
Payees:
Delete Payee
Success!
Account successfully updated!
OK
Account Number
Email
10000
10001
10002
10003
Nala
nala@gmail.com
10004
Kiara
kiara@gmail.com
10005
Kovu
kovu@gmail.com
10006
Timon
timon@gmail.com
10007
Pumbaa
pumbaa@gmail.com
10008
Rafiki
rafiki@gmail.com
Reload
Sign out
Help: Select payee and click Delete to remove payee

16) Delete Payee:

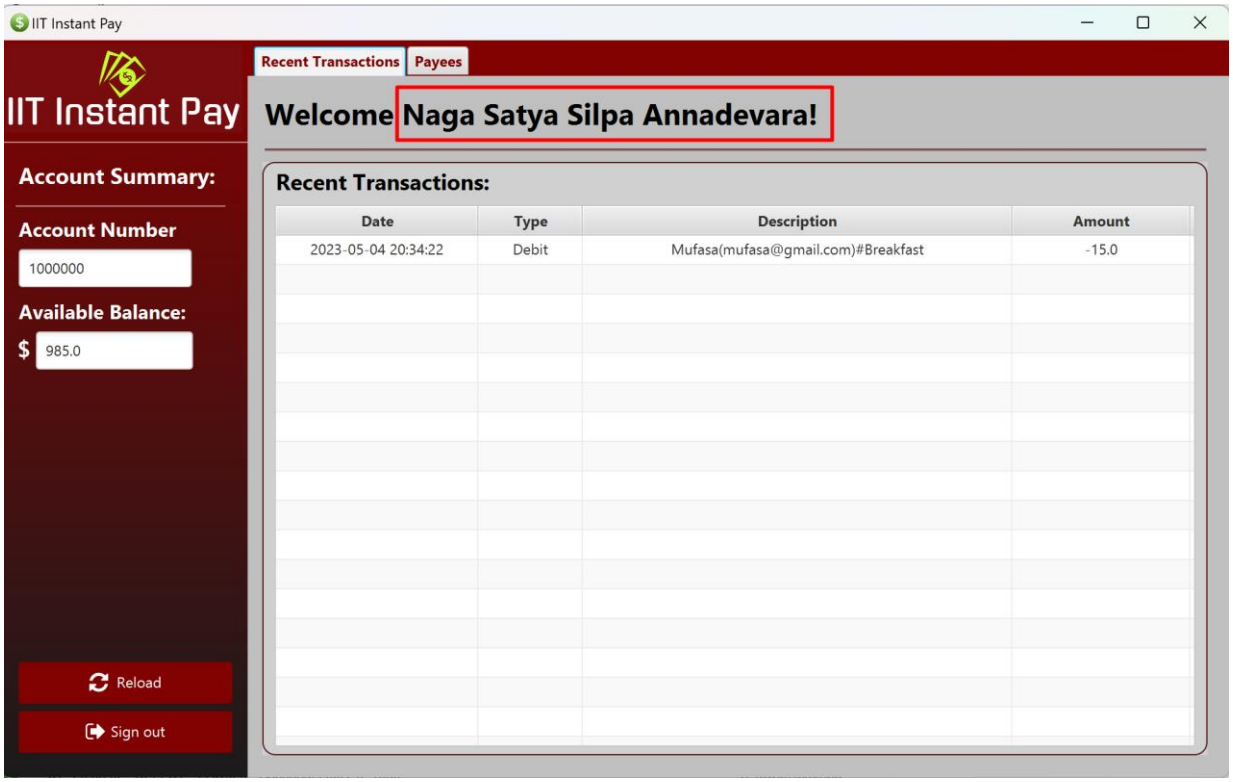


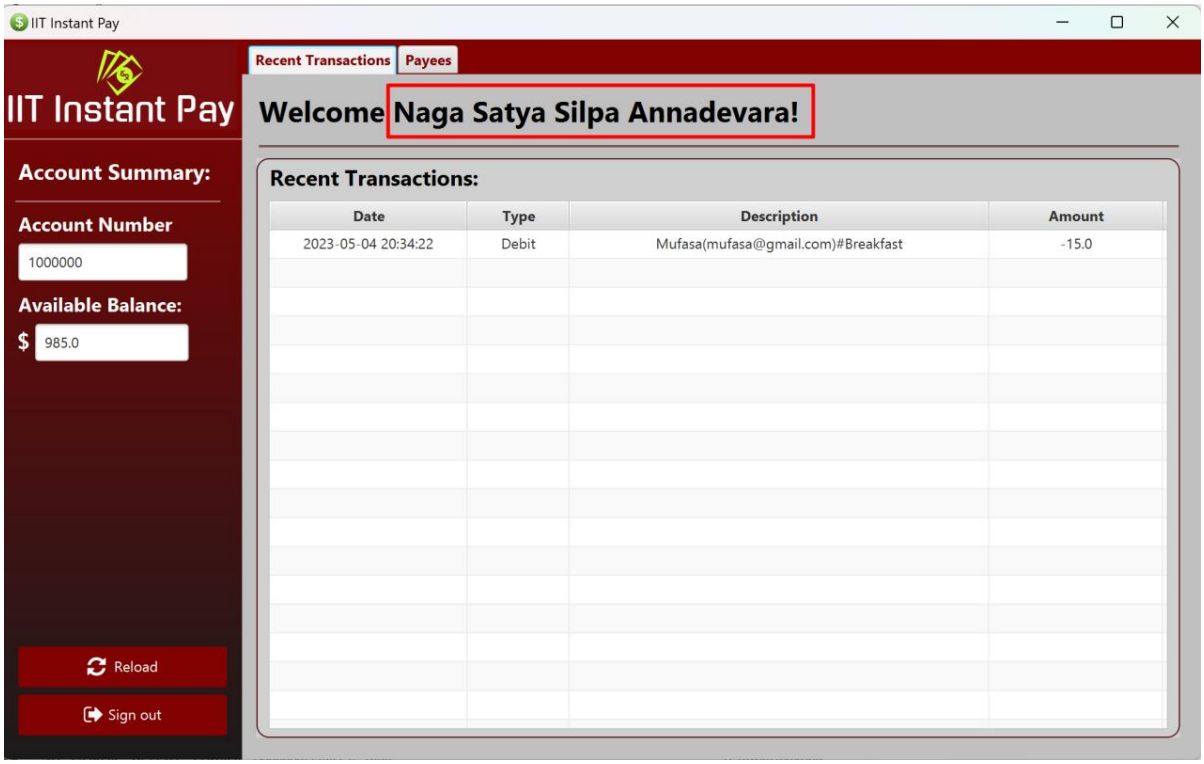


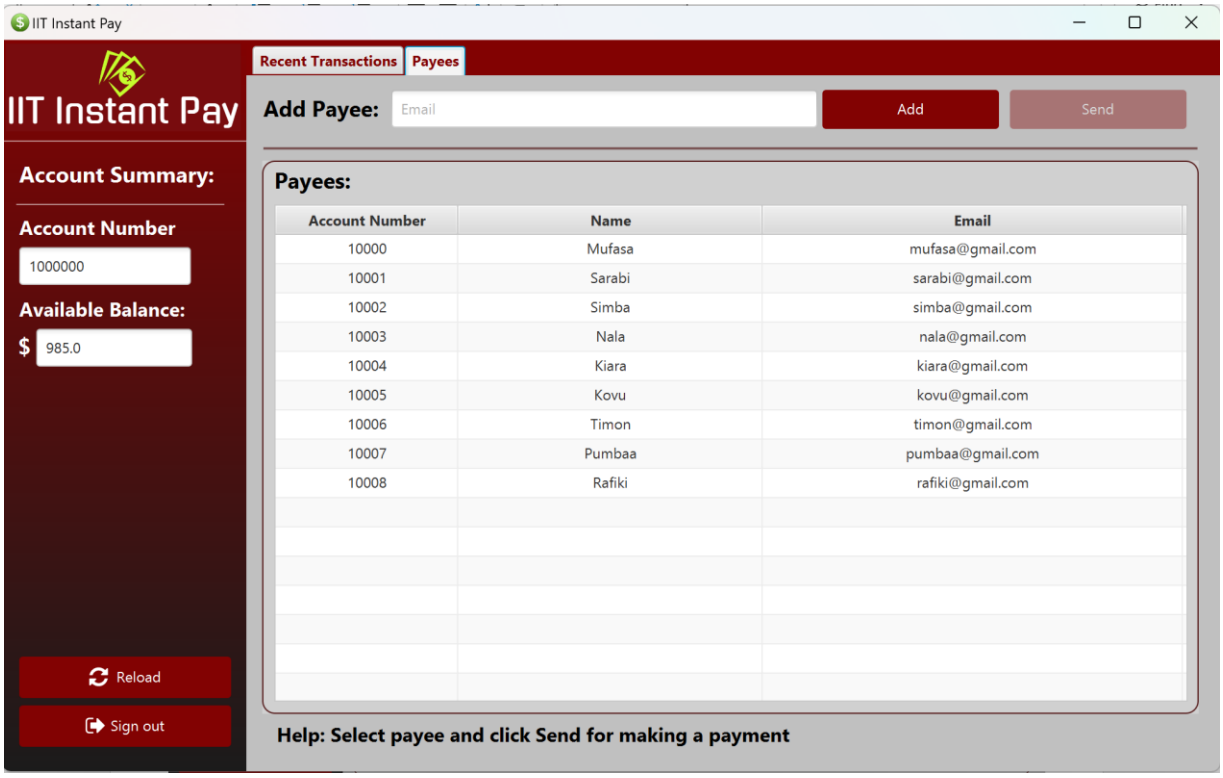


17) Show a snapshot of your remaining records in some columnar view.

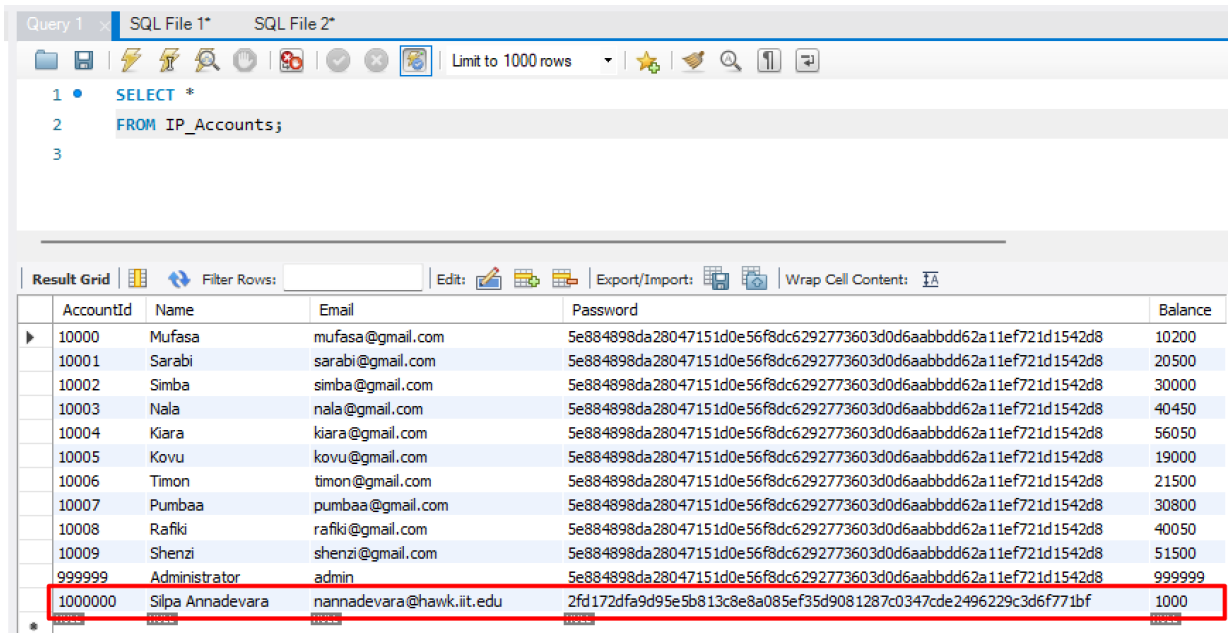
Logging back in using Silpa account to verify changes:


IIT Instant Pay
IIT Instant Pay
Account Summary:
Account Number
1000000
Available Balance:
$ 985.0
Reload
Sign out
Recent Transactions
Payees
Welcome Naga Satya Silpa Annadevara!
Recent Transactions:
Date
Type
Description
Amount
2023-05-04 20:34:22
Debit
Mufasa(mufasa@gmail.com)#Breakfast
-15.0


IIT Instant Pay
IIT Instant Pay
Account Summary:
Account Number
1000000
Available Balance:
$ 985.0
Reload
Sign out
Recent Transactions
Payees
Welcome Naga Satya Silpa Annadevara!
Recent Transactions:
Date
Type
Description
Amount
2023-05-04 20:34:22
Debit
Mufasa(mufasa@gmail.com)#Breakfast
-15.0



18) Show a snapshot of your ending user table (i.e., the user login credentials) <u>from the database itself</u>.



AccountId	Name	Email	Password	Balance
10000	Mufasa	mufasa@gmail.com	5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8	10200
10001	Sarabi	sarabi@gmail.com	5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8	20500
10002	Simba	simba@gmail.com	5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8	30000
10003	Nala	nala@gmail.com	5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8	40450
10004	Kiara	kiara@gmail.com	5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8	56050
10005	Kovu	kovu@gmail.com	5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8	19000
10006	Timon	timon@gmail.com	5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8	21500
10007	Pumbaa	pumbaa@gmail.com	5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8	30800
10008	Rafiki	rafiki@gmail.com	5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8	40050
10009	Shenzi	shenzi@gmail.com	5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8	51500
999999	Administrator	admin	5e884898da28047151d0e56f8dc6292773603d0d6aabbdd62a11ef721d1542d8	999999
1000000	Silpa Annadevara	nannadevara@hawk.iit.edu	2fd172dfa9d95e5b813c8e8a085ef35d9081287c0347cde2496229c3d6f771bf	1000