Sprawozdanie z całości

Julia Ołowniuk

1. Project topic

Security systems in a bank

2. Topic description

Let's start with what security systems in a bank are. They are integrated computer systems that provide protection for customer data, as well as bank systems and infrastructure from threats. They are crucial for the functioning of banks because financial security is one of the most important aspects for customers and financial institutions.

Before introducing the security system, the company operated on traditional security methods such as physical protection, surveillance cameras, locks, and alarms. Physical protection involved hiring security guards who monitored entrances and exits from buildings and guarded valuable documents. Surveillance cameras were placed in different locations in the company, and the recorded image was archived and viewed by security personnel. Locks and alarms protected doors and windows from intrusion. Additionally, to ensure the security of transactions, bank employees had to verify the identity of customers using documents and apply various procedures such as monitoring transactions, reporting suspicious activities, etc. These procedures were time consuming and required significant effort.

These methods were effective to some extent but did not provide complete protection against modern threats such as hacking attacks or data theft. Physical protection was insufficient because it did not protect against cyber threats. Surveillance cameras and locks were easy to bypass by outsiders, increasing the risk of data or money theft. Therefore, I decided to introduce security systems to the bank. Of course, security systems also require monitoring, but they work much faster and often have more capabilities.

One of the most important improvements achieved through the security system is the automation of many procedures, which allows for time and cost savings. For example, the security system can automatically verify the customer's identity using technologies, allowing for faster and more effective customer identity verification. In addition, the security system can monitor transactions and detect irregularities such as suspicious transactions or attempts at intrusion, allowing for quick response and minimizing damage.

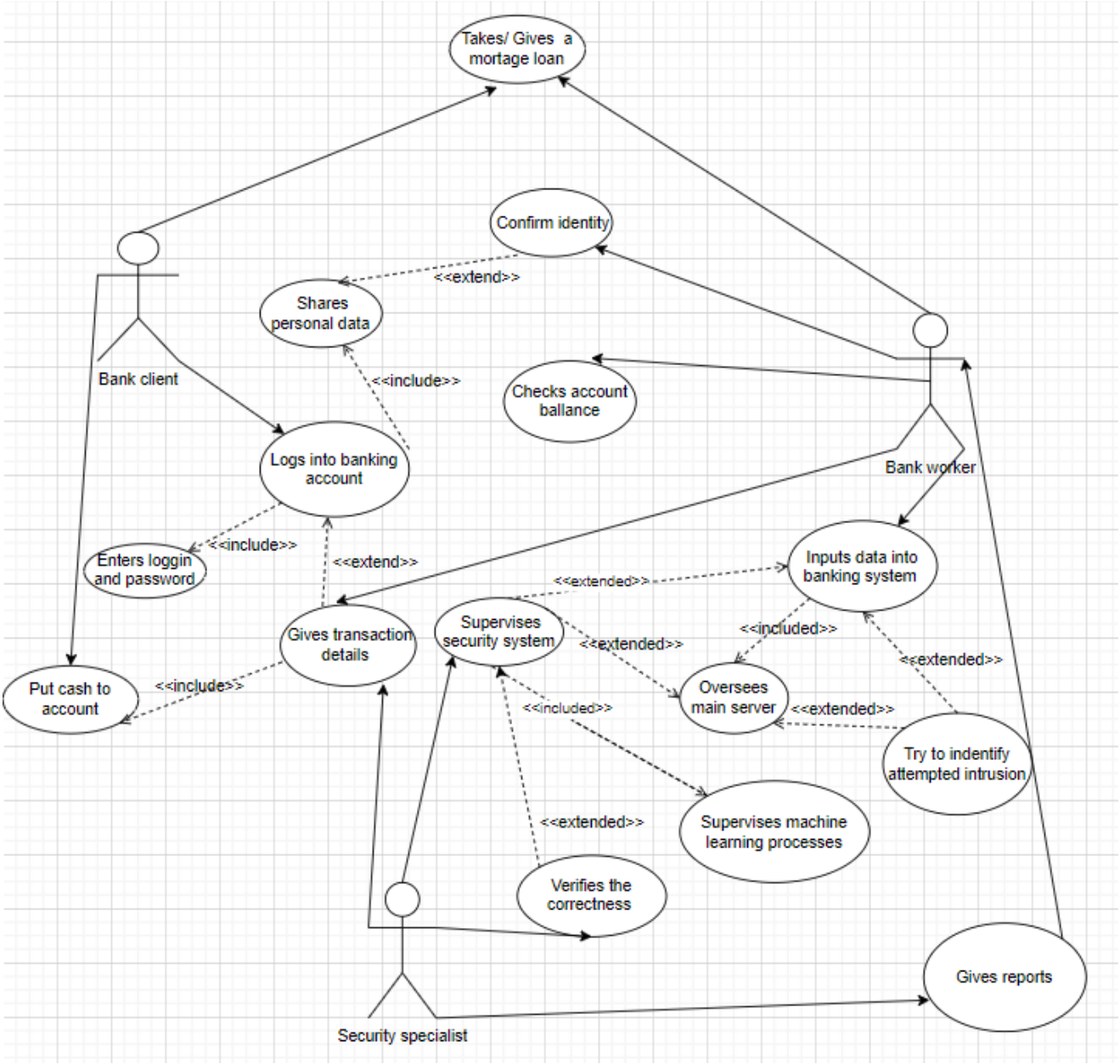
Another important improvement is increasing the security of customer data. Security systems in banks provide protection for customer data against unauthorized access and data theft. This ensures that customers can be confident that their data is safe and protected.

Finally, security systems in banks allow for faster and more effective resolution of security related issues such as hacking attacks, attempts to steal data, or financial fraud. This minimizes risk.

In summary, security systems in banks are crucial for ensuring financial security and protecting customer data. Improvements achieved through these systems include primarily the automation of many procedures and increasing the security of customer data. The introduction of a security system in a bank allows for faster and more effective operation, which translates into greater customer trust and minimizing financial risk. Thanks to security systems, banks can operate faster and more effectively, allowing for better results.

3. Use case diagram

The use case diagram illustrates the interactions between actors (users) and the system, showcasing the functionalities of the system.



4. Usage scenarios

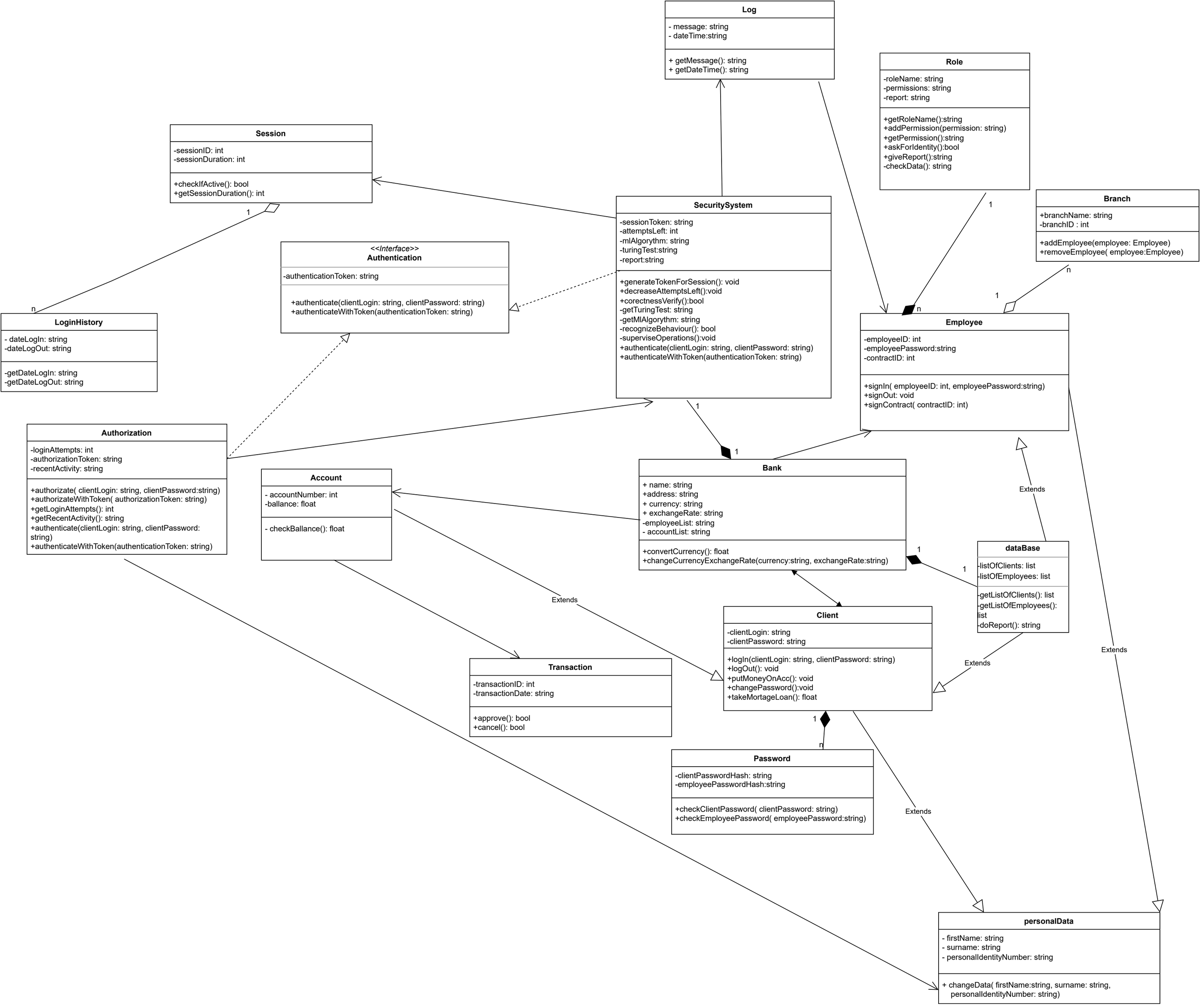
Description of individual interactions between the user of the system and the system itself. Using it presents in a clear and understandable manner how the various system functions are utilized in practice by the users.

No.	Scenario name	Bank client	Bank worker	Security specialist	System
1.	Client login	enters their login and password	not applicable	if the system fails to recognize the customer, despite their presence in the database, it will be fixed.	conducts the authorization process and positively or negatively confirms the client's access to their account
2.	Cash deposit	brings cash to the bank branch	accepts the deposit and enters the transaction details into the banking system if system does not work	verifies the correctness of the entered data	conducts the transaction authorization process and records it in the banking system
3.	Changing the client's password	logs into the internet banking system and changes the password	if the password is changed too many times within a short period of time, the client is contacted for identity verification	verifies if the system has not made a mistake	conducts the authorization process and records the password change in the banking system
4.	Application for a mortgage loan	submits a mortgage loan application at a bank branch	enters the application details into the banking system and conducts a preliminary client verification	monitors for errors in the contract entered by the system	the authorization process for the application and records it in the banking system.
5.	Money transfer order	enters recipient information and transfer amount	confirms the transaction	detects any attempted intrusions	monitors the customer's activity
6.	Checking transaction history	selects the "transaction history" option	not applicable	supervises to ensure no errors are displayed	authorizes access to customer data

7.	Monitoring client activity in the system	may be asked to confirm their identity.	receives a notification in case of any detected threats and contacts with client	uses monitoring tools to detect suspicious customer activity in the system and generate protocols	it helps with machine learning if there are any suspicious activities
8.	Detecting attempted intrusion	may be asked to confirm their identity	informs the customer about suspicious activity and asks for identity confirmation	analyses reports and takes appropriate action	detects unauthorized access attempts
9.	Viewing account balance	logs into the banking system, selects the "account balance" option, and receives information about their account balance	logs into the banking system, selects the client's account, and displays information about the account balance to them	ensures the confidentiality of the information	monitors the client's activity thanks to machine learning, detects any attempts of unauthorized access to account balance information, authorizes access to the client's data
10.	Managing client data	logs into the banking system, selects the "client data" option, and updates their personal or contact information	logs into the banking system, selects the client's account, and updates their personal or contact information	oversees whether any unauthorized changes have been made to the client's data	authorizes access to the client's data and protects it from unauthorized access or changes

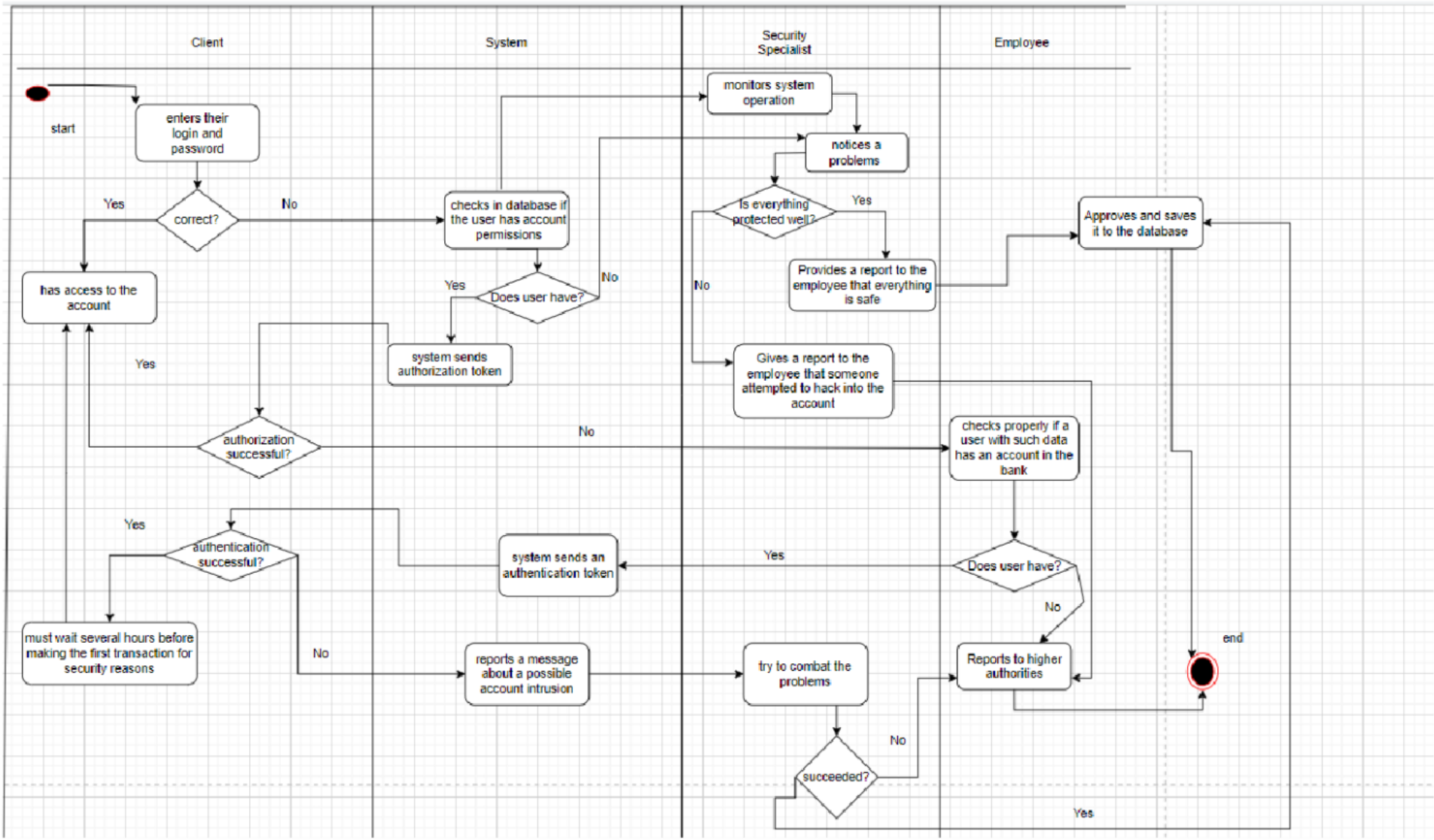
5. The class diagram

   Class diagram depicting the relationship and dependencies between classes in the system.


Log
- message: string
- dateTime:string
+ getMessage(): string
+ getDateTime(): string
Role
-roleName: string
-permissions: string
-report: string
+getRoleName():string
+addPermission(permission: string)
+getPermission():string
+askForIdentity():bool
+giveReport():string
-checkData(): string
Session
-sessionID: int
-sessionDuration: int
+checkIfActive(): bool
+getSessionDuration(): int
SecuritySystem
-sessionToken: string
-attemptsLeft: int
-mlAlgorythm: string
-turingTest:string
-report:string
+generateTokenForSession(): void
+decreaseAttemptsLeft():void
+corectnessVerify():bool
-getTuringTest: string
-getMlAlgorythm: string
-recognizeBehaviour(): bool
-superviseOperations():void
+authenticate(clientLogin: string, clientPassword: string)
+authenticateWithToken(authenticationToken: string)
Branch
+branchName: string
-branchID : int
+addEmployee(employee: Employee)
+removeEmployee(employee:Employee)
<<Interface>>
Authentication
-authenticationToken: string
+authenticate(clientLogin: string, clientPassword: string)
+authenticateWithToken(authenticationToken: string)
LoginHistory
- dateLogIn: string
-dateLogOut: string
-getDateLogIn: string
-getDateLogOut: string
Employee
-employeeID: int
-employeePassword:string
-contractID: int
+signIn(employeeID: int, employeePassword:string)
+signOut: void
+signContract(contractID: int)
Authorization
-loginAttempts: int
-authorizationToken: string
-recentActivity: string
+authorize(clientLogin: string, clientPassword:string)
+authorizateWithToken(authorizationToken: string)
+getLoginAttempts(): int
+getRecentActivity(): string
+authenticate(clientLogin: string, clientPassword: string)
+authenticateWithToken(authenticationToken: string)
Account
- accountNumber: int
-ballance: float
- checkBallance(): float
Bank
+ name: string
+address: string
+ currency: string
+ exchangeRate: string
-employeeList: string
- accountList: string
+convertCurrency(): float
+changeCurrencyExchangeRate(currency:string, exchangeRate:string)
dataBase
-listOfClients: list
-listOfEmployees: list
-getListOfClients(): list
-getListOfEmployees(): list
-doReport(): string
Client
-clientLogin: string
-clientPassword: string
+logIn(clientLogin: string, clientPassword: string)
+logOut(): void
+putMoneyOnAcc(): void
+changePassword():void
+takeMortageLoan(): float
Transaction
-transactionID: int
-transactionDate: string
+approve(): bool
+cancel(): bool
Password
-clientPasswordHash: string
-employeePasswordHash:string
+checkClientPassword(clientPassword: string)
+checkEmployeePassword(employeePassword:string)
personalData
- firstName: string
- surname: string
- personalIdentityNumber: string
+ changeData(firstName:string, surname: string, personalIdentityNumber: string)
Extends
Extends
Extends
Extends
Extends
1
n
1
n
1
n
1
1
1
1
1
n

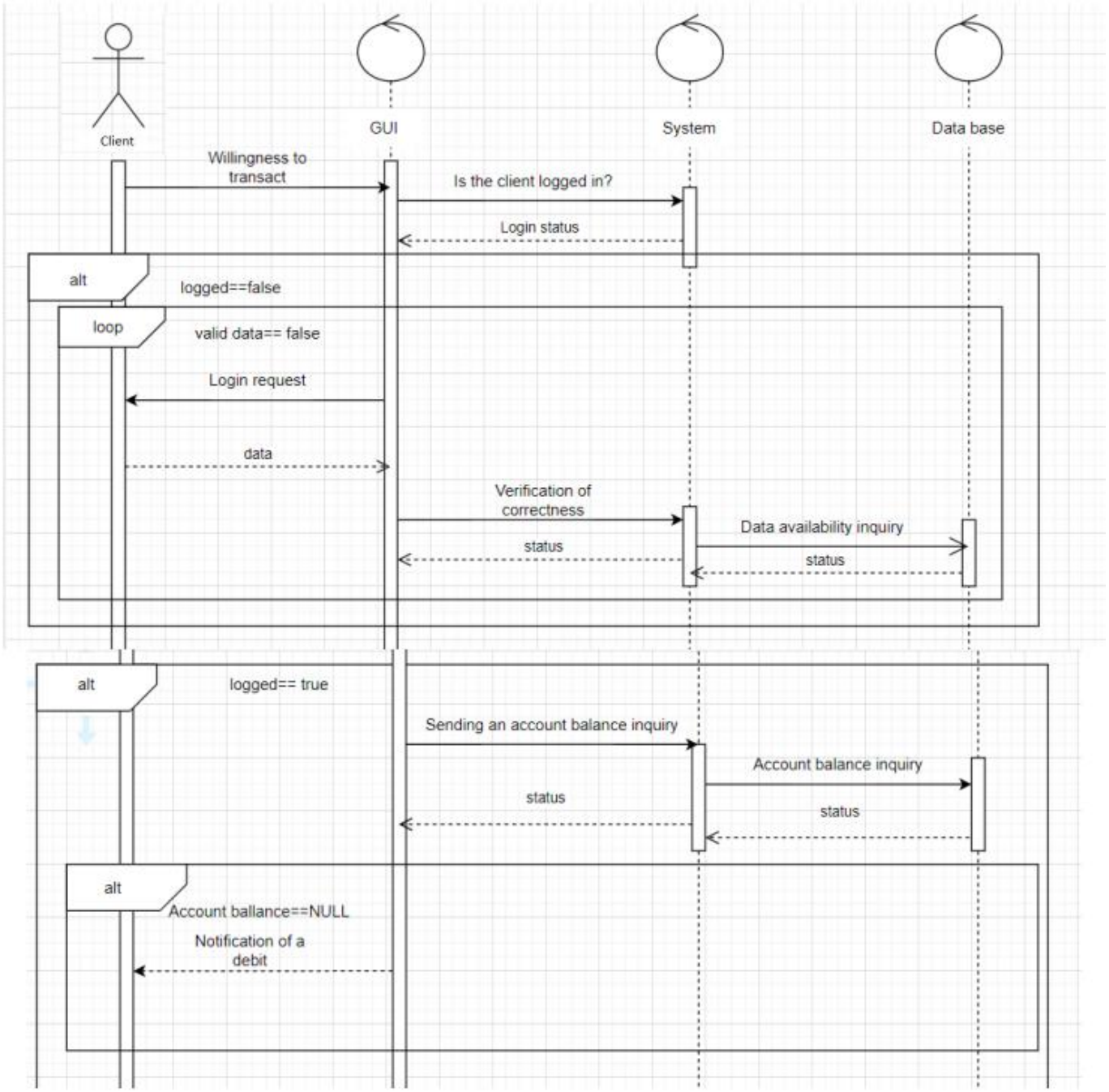
6. Activity diagram

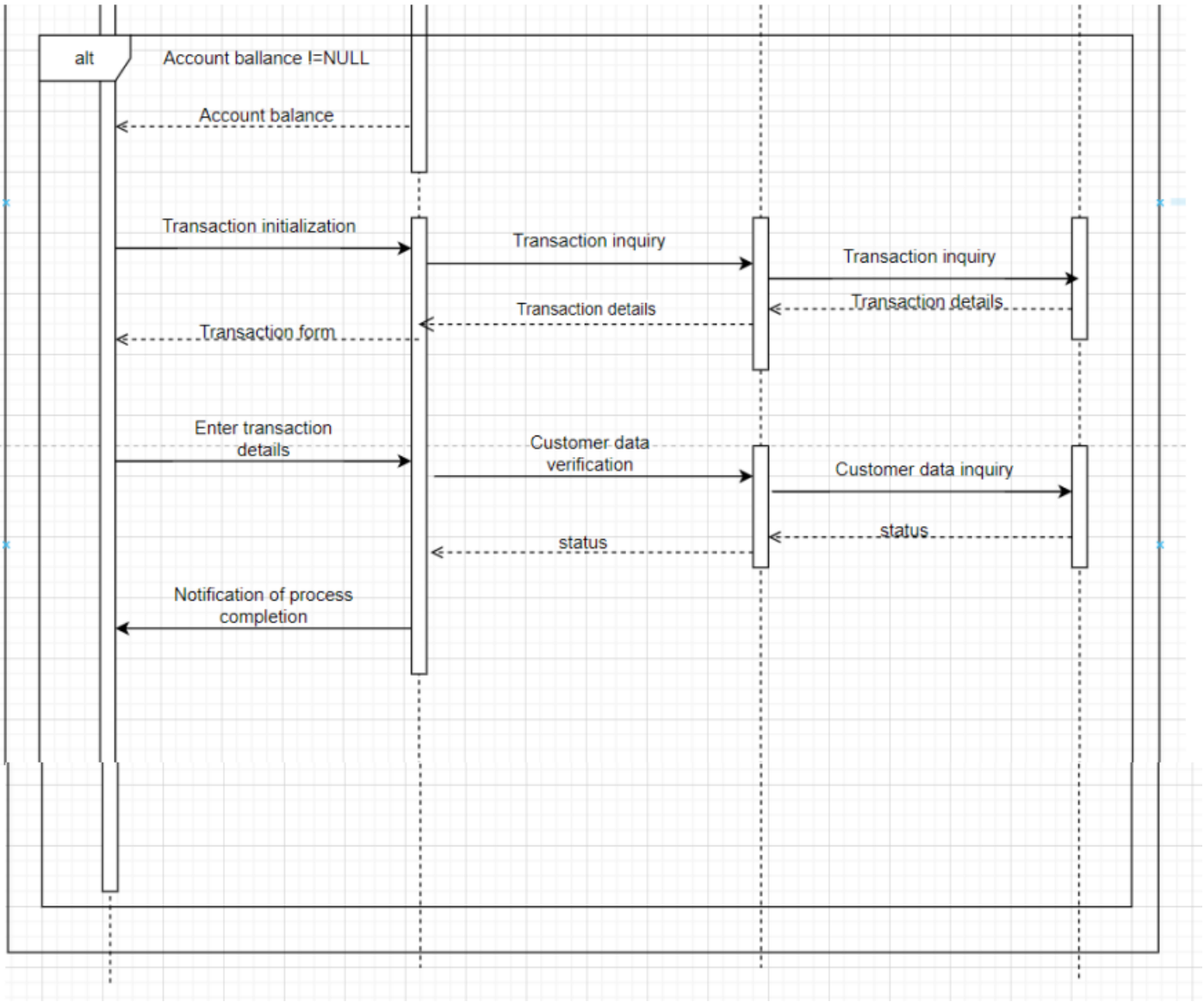
Diagram illustrating the main program flow.



7. Sequence Diagram “Transaction”

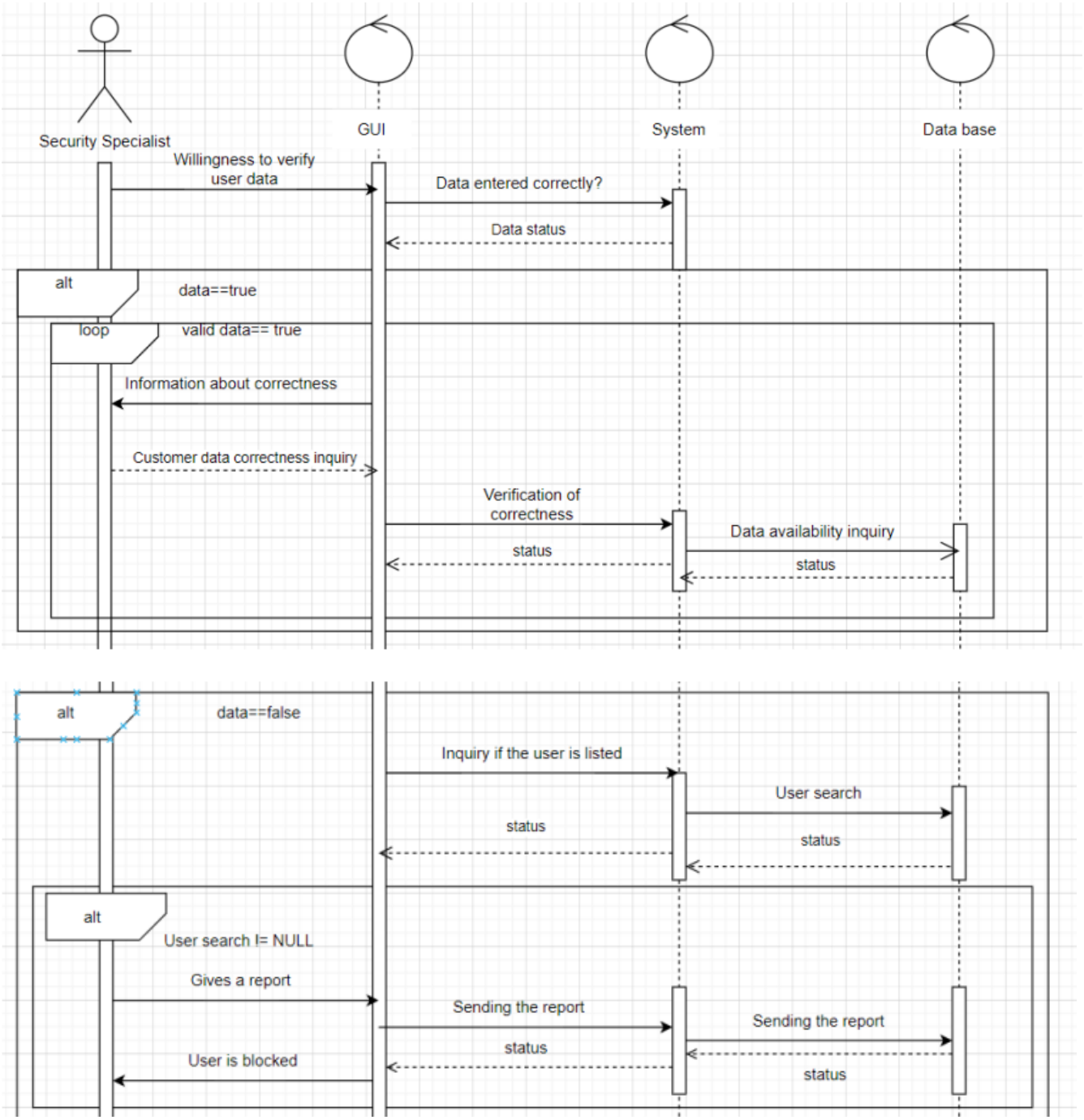
   Diagram presenting the sequence of steps in a transaction.

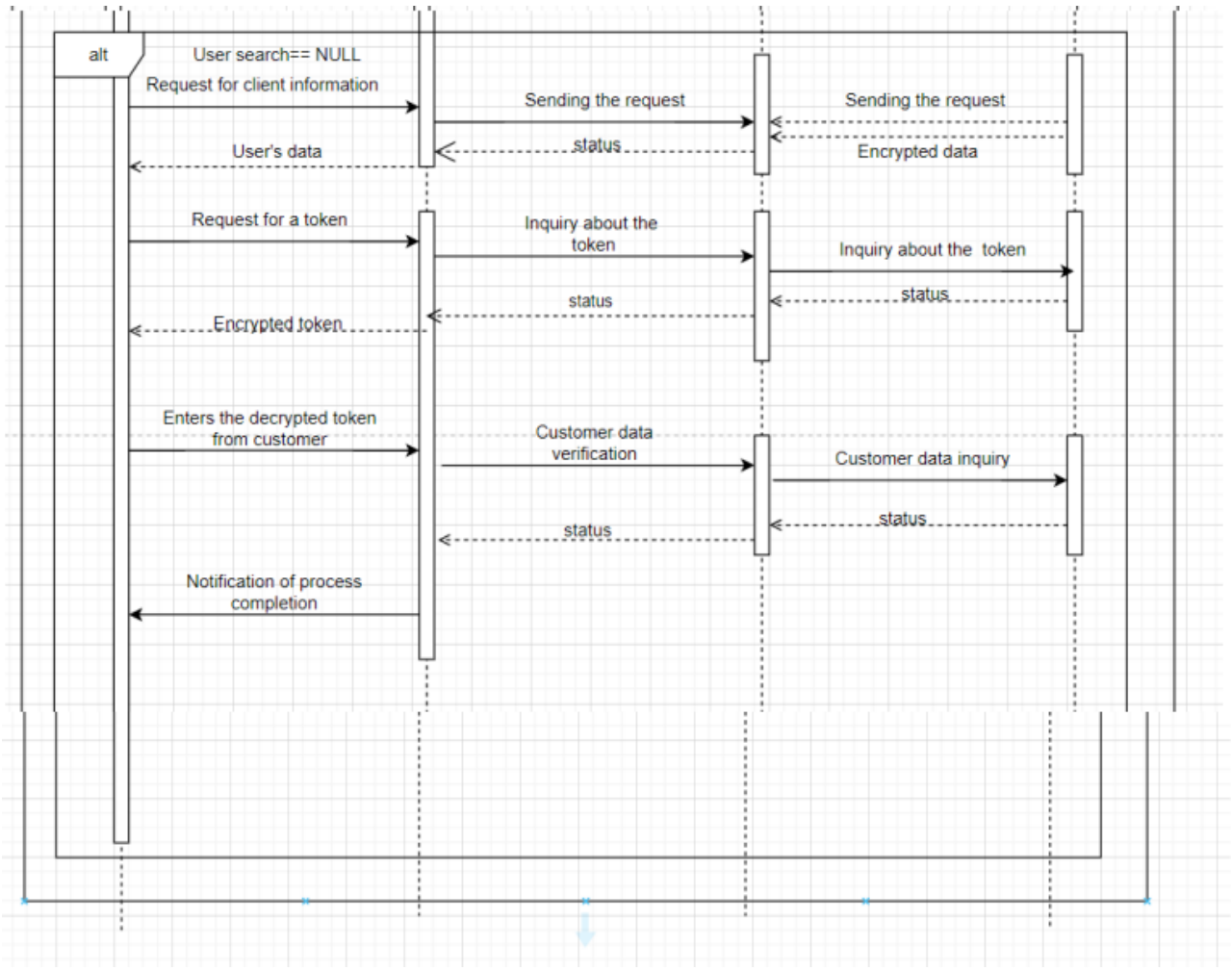



alt
Account ballance !=NULL
Account balance
Transaction initialization
Transaction inquiry
Transaction inquiry
Transaction details
Transaction details
Transaction form
Enter transaction details
Customer data verification
Customer data inquiry
status
status
Notification of process completion

8. Sequence Diagram „Verification with token”

Diagram illustrating the verification process using a token.

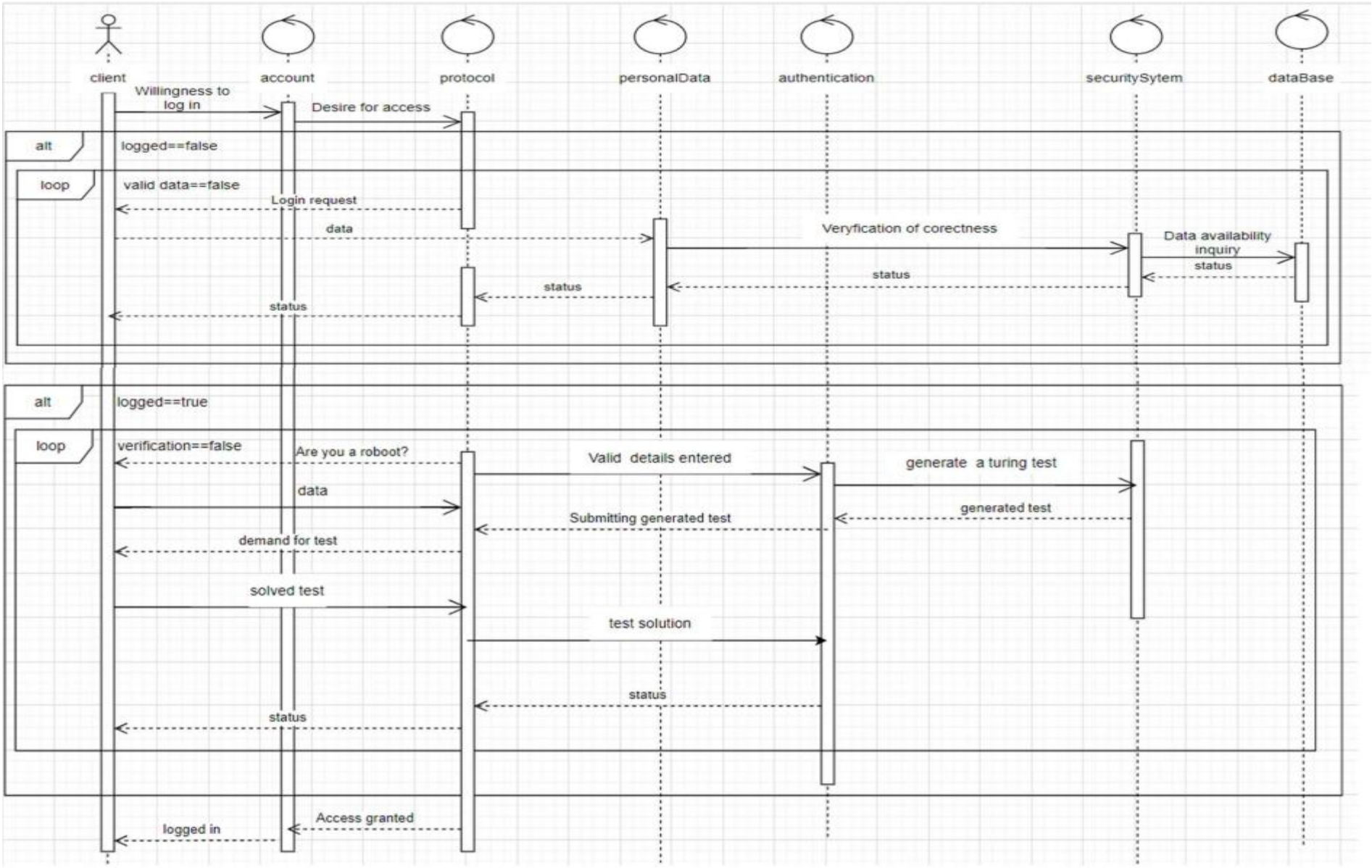



alt
User search== NULL
Request for client information
Sending the request
Sending the request
status
Encrypted data
User's data
Request for a token
Inquiry about the token
Inquiry about the token
status
status
Encrypted token
Enters the decrypted token from customer
Customer data verification
Customer data inquiry
status
status
Notification of process completion

9. Sequence Diagram “Verification of whether the client is not a robot”

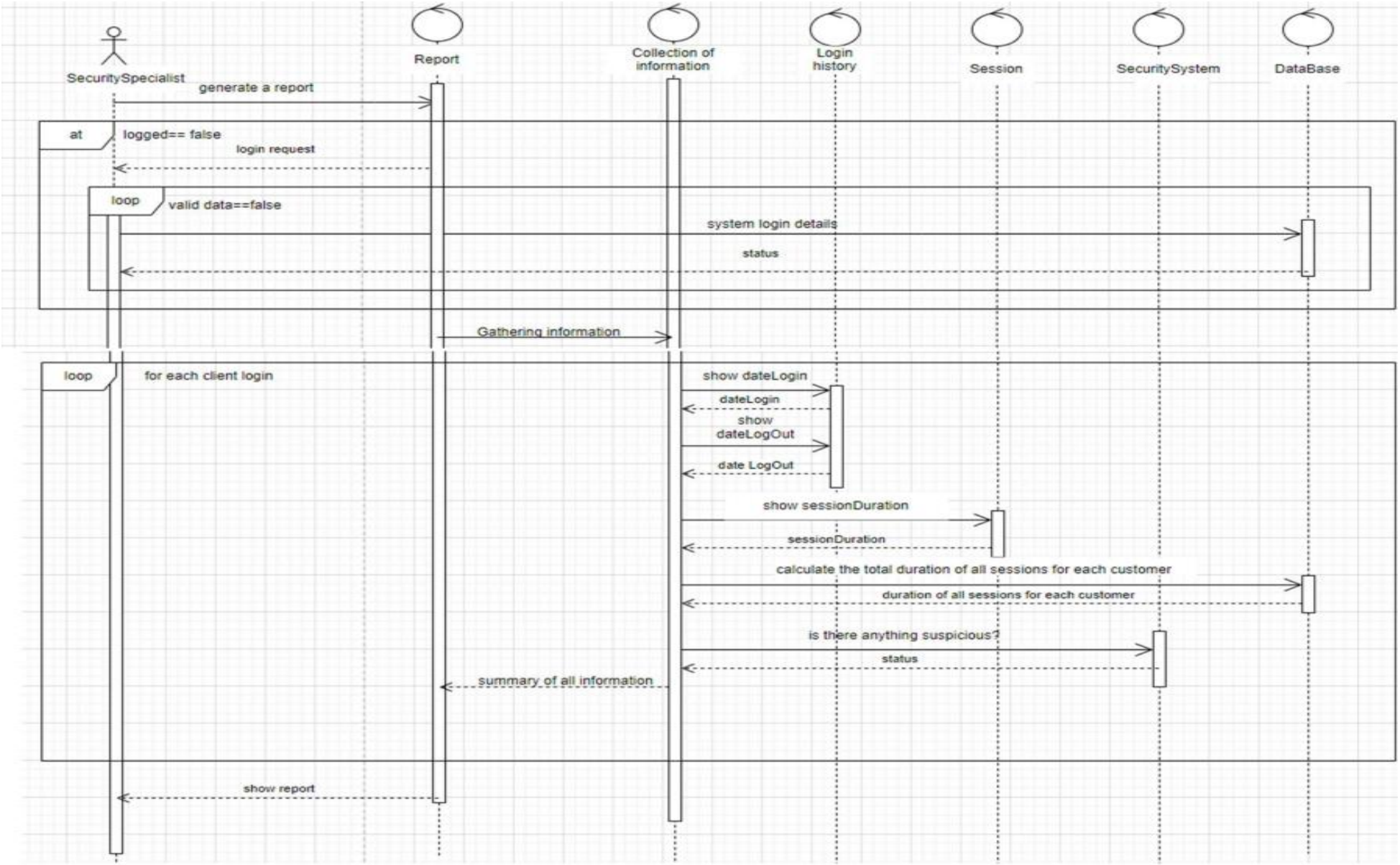
The sequence diagram describing the generation of protocol whether the user trying to log into the bank is not a robot thanks to Turing’s tests.

It includes relationship between classes: account, protocol, personalData, security system and data base.



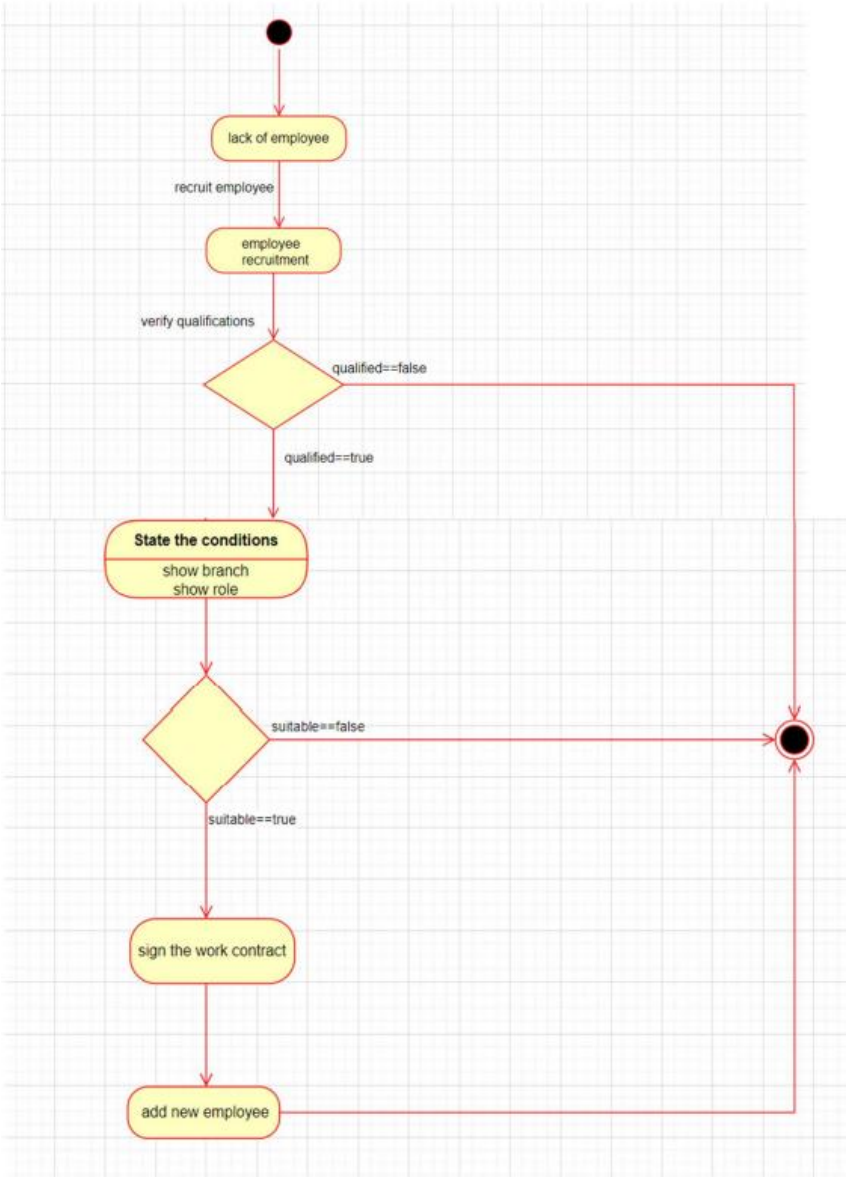
10. Sequence Diagram “Client’s all activity history report”

It is a sequence diagram describing the generation of a client’s all activity history report. It includes relationships between classes: session, login history, security system, data base, report, collection of information.



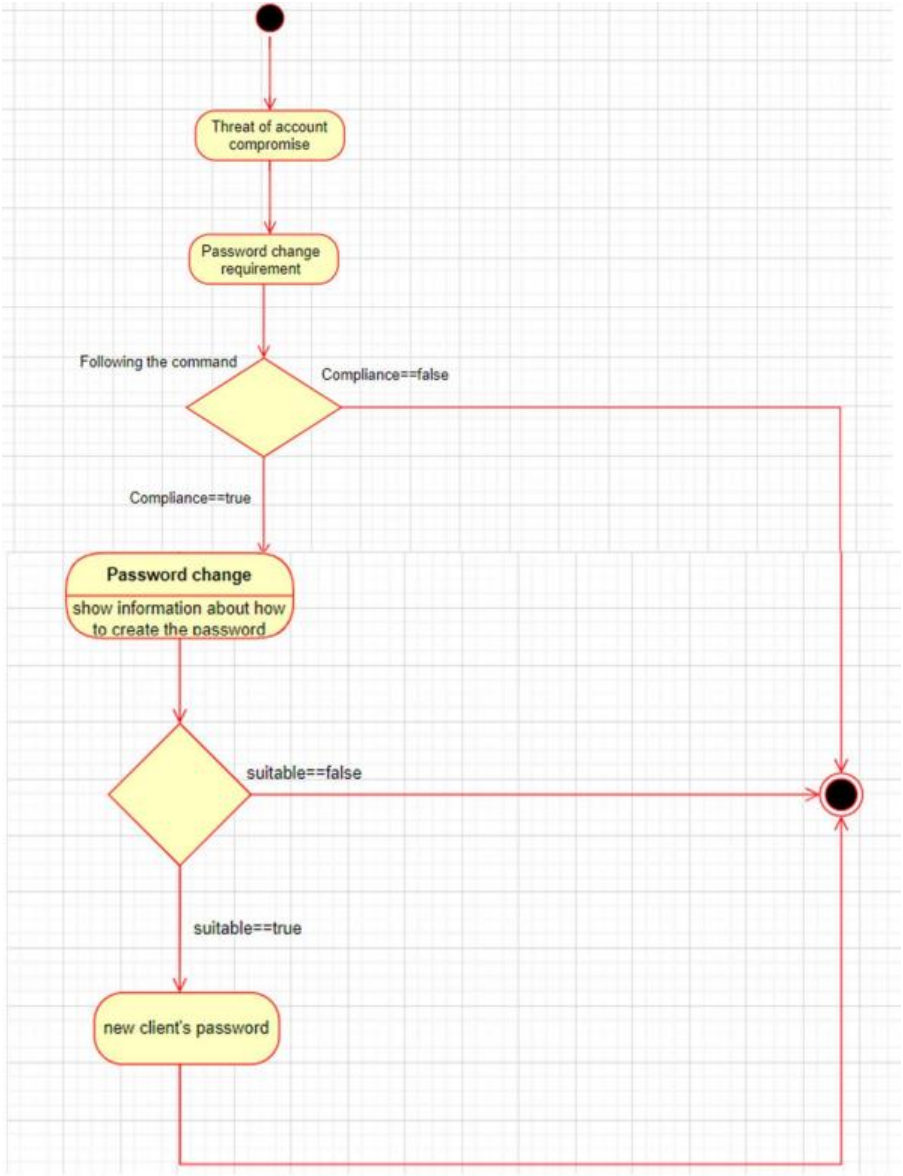
11. State Diagram " Adding a new employee".

In the diagram, state transitions are shown from the absence of an employee to adding a new employee.



12. State Diagram “ Password change by the client”

The state diagram illustrating the password change by the client, depicted from threat of account compromise to the new password. Password change is required for security reasons.



13. Graphical User Interface Test Scenarios

Test name	User actions/ input data	Expected outcomes	Test result
Client login	Click the "Login" button Enter your username: Julie16 Enter your password: 1235 Click the "Log in" button Perform the Turing test Click the "Save" button	The client Julie16 has been logged in	Successfully completed
Invalid client's password	Click the "Login" button Enter your username: Julie16 Enter your password:!125 Click the "Log in" button Confirm the message "Incorrect password." Select the option "Verify identity with SMS token." Enter the received SMS token: 111 Click "Log me in with SMS token."	The client Julie16 has been logged in thanks to SMS token	Successfully completed
Changing the password of the logged-in customer	Click the "Account Settings" button Select "Security Settings." Choose "Change Password." Enter the old password :1235 Enter the new password : 1234 Enter the new password again: 1234 Click the "Save" button Click "Log me out from other logged-in devices."	The password has been changed from 1235 to 1234	Successfully completed
Changing the password of an unlogged-in customer with token	Click the "Login" button Click the "Forgot Password" button Enter the phone number associated with the account: 123456789 Enter the received SMS token: 234 Enter the new password: 2345 Enter the new password again: 2345 Click the "Save" button Click the "Log me out from other logged-in devices" button	The password has been changed to 2345 thanks to SMS token	Successfully completed
Performing a transaction by a logged-in customer	Check the "Account Balance." Click on "Perform Transaction." Enter the amount: 30 Select the currency: PLN Enter the account number: PL 69 9439 0007 8478 5014 9432 2978. Enter the sender's information: Julia Ołowniuk Enter the recipient's information: Jan Kowalski Click the "Perform Transaction" button. Enter the PIN to confirm the transaction: 1367 Click "Ok"	The amount of 30 PLN has been sent from Julia Ołowniuk to Jan Kowalski	Successfully completed

Obtaining a low-value mortgage loan by a logged-in customer	Click the "Mortgage Loan" button Select the amount: 100 Choose the currency: PLN Select the repayment period: 30 days Choose identity verification confirmation: authentication token. Please enter the PESEL as authentication token:01020304056 Click "Safe". Perform the Turing test Click the "Save" button Click "Take out a mortgage loan". Click "Save".	The amount of 100 PLN has been transferred to the account of Julia Olowniuk as a mortgage loan	Successfully completed
Displaying login dates for client	Searching for customers with the surname "Olowniuk". Selecting a client from the list. Clicking on "Login Dates".	The list of login dates for the client with the surname Olowniuk	Successfully completed
Generate the protocol of log checking	Click on "Check logs." Click on "Return log message." Click on " Log date and time" Click on "Generate protocol"	Generated log checking protocol	Successfully completed
Authentication for identity client's verification	Click on "Authenticate the client" Choose client: Julia Olowniuk Send authentication token to : Julia Olowniuk Confirm the confirmation Adding to the database: identity verification confirmed Click on "Save changes"	The customer Julia Olowniuk has been verified	Successfully completed
Authorization process to grant client new permissions.	Click on "Authorize the client" Choose client: Julia Olowniuk Send authorization token to : Julia Olowniuk Confirm the confirmation Click on "Managing client permissions" Choose "Julia Olowniuk" Edit client's permissions "Add permissions to the client" Click on "Save changes"	Client Julia Olowniuk is being granted new permissions	Successfully completed