COS 301 Mini Project Phase 2 Group 10
Final Count Down

Thomas Honiball 15348751 Khodani Tshisimba 11034531
Christiaan Nel 16323883 Fiwa Lekhulani 15187145
Kearabiloe Nare 10068083 Keorapetse Shiko 12231992

February 2019

Contents

1 Introduction

1.1 Purpose

The purpose of this document is to define the software requirements and design specifications of a New Generation ATM (New-Gen ATM). The main idea behind this New-Gen ATM is to reduce the currently large traffic in queues caused by cash based operations at FNB ATMs. This document takes an in-depth look at how the previously established subsystems will be defined and implemented as well as how they will interact with one another.

1.2 Scope

This document will provide a layout of the architectural design of the system as a whole. This will include the main subsystems as well as other subsystems that form part of the New Generation (New-Gen) ATM. Non-functional requirements are also declared and described to allow a higher quality of usability and general appeal to the user.
User characteristics and how each user interacts with a subsystem is defined. A subsystem will be broken down into it's definition and implementation (domain model), functional requirements and interaction between use case and requirement. one of the subsystems will be used to display a model between an actor and system and finally the system as a whole will be viewed in light of the use cases and subsystems in the form of a traceability matrix.

1.3 Definitions, Acronyms, and Abbreviations

Terms	Definition
ATM	Automated Teller Machine
FNB	First National Bank
NFC	Near-Field Communication
UI	User Interface
R&A	Report and Analysis
App	Application (Mobile)
CRUD	Create, Read, Update, Delete

References

[1] *Criteria For Bank Account:* 2019 Available from https://www.banking.org.za/consumer-information/conventional-banking/criteria-for-a-bank-account [2019]

[2] *FIC act* https://www.fic.gov.za/Documents/FIC%20Act%20with%202017%20amendments%20(1)%20(1).p [2017]

[3] *SEBok* https://www.sebokwiki.org/wiki/System_Requirements

1.4 Overview

1. **Architectural Design** - This will be used to showcase the architectural style of the system and contains details about the technologies and protocols which will be used for the implementation of the system.

2. **Non-functional Requirements** - This will include the main criteria that can be used to judge the operations of the system with a particular focus on quality attributes, constraints and technological requirements.

3. **User Characteristics** - Lists all the intended users and explains for what purpose each type of user would use the system by referring to the specific subsystem.

4. **Subsystems** - This will include the subsystem's domain model(which includes business concepts, attributes and relations), use case diagram, functional requirements and subsystem tracebility matrix.

5. **Actor-System Interaction Model** - The authentication subsystem will specify preconditions and postconditions and will state which exception will be thrown for each precondition if the condition is not met.

6. **System Tracebility Matrix** - A matrix with use case numbers as rows and subsystems as columns.

2 Architectural Design

The New-Gen ATM system functions using Client Server, Object Persistence and Event Driven architecture to accomplish the business goals. Object Persistence in the form of a centralised SQL database which stores the user and transaction details, ensures that user data persists regardless of device or time between uses. The mobile device and ATM both report to a centralised server to ensure that data remains constant and secure as well as keeping the client side environment as compact as possible, preferring to keep data on the server rather than the client. The client application uses an Event Driven system to wait for client input or the interaction with an ATM before starting operation. This creates a responsive client system which can focus on ease of use.

2.1 Deployment Model

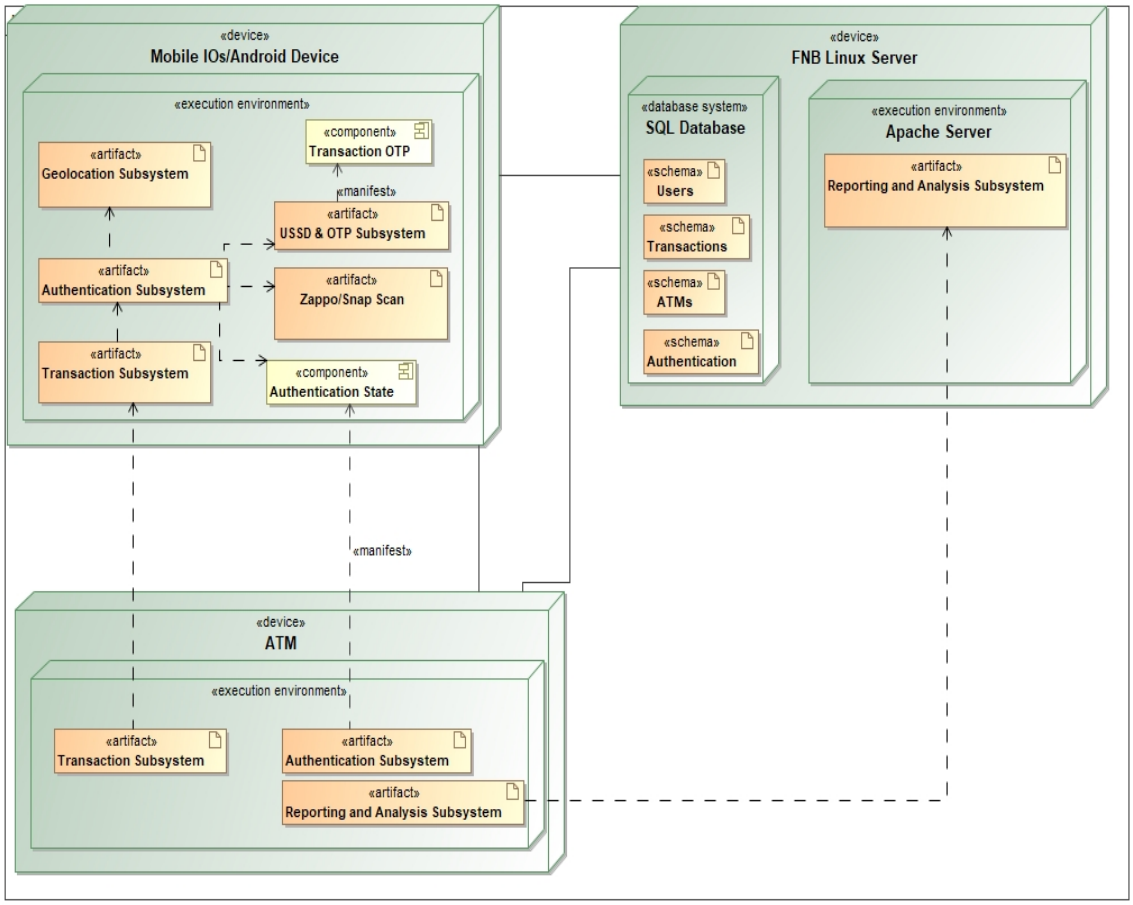

Figure 1: System Deployment Model

3 Non-functional Requirements

3.1 Quality Attributes

Performance:

This quality requirement refers to the how well the system works and the speed at which operations can be done. This is depended on the hardware(type of phone) being used in the ATMs as well as the connection speed between the ATM and the back end system. The performance can be measured in terms of

time, elements of the system to be measured would include the speed at which the app/ transactions take place.

Reliability:

This quality requirement refers to the ability of the system to work as expected and maintain how it works over the course of its expected life time. It is essential that when a transaction is being processed, that if there is a failure the system needs to alert the user and cancel the transaction on both mediums. We will check that the software can perform failure-free operation for a specified amount of time across a range of devices.

Scalability:

This quality requirement refers to the how expandable the system will be, if it is able to be implemented in ares which are densely populated such as cities as well as less populated areas such as rural towns. This also extends to how the system will work in areas with perfect signal opposed to limited signal areas. It is important to note that not all users will be using devices that have NFC or have data always ready to be used, consideration and research into the use of QR codes and making use of USSD codes must be done. To test this quality requirement we will test the performance(failure/success rate) of the systems in the various specified areas which have been named above as well as the amount of people able to make use of the system.

Security:

This quality requirement refers to the safety of the users confidential information, ensuring that it cannot be intercepted and misused to the determent of the user. The main idea of the system will be to make use an NFC for the first part of the verification. Two other methods to ensure validation will take place, one being the use of tokens and OTP will be sent to the user. This will be quantified by the amount of reports issued by users of user profiles getting logged into as well as any issues with regards to the validation system(how many people have the ability to make use of tokens, OTP and NFC).

Flexibility:

This quality requirement refers to the ability of the system to work as it is intended but have the ability to work as a standard ATM does in the event that a problem occurs with the implemented system. The back end system that will be implemented needs to be made backwards compatible to ensure this flexibility. The system should only allow for the use of an ATM in a standard fashion if the new system is having issues. Testing needs to be done to ensure that the system will work on both the new system and old system. The time

it takes to switch from the new system to the old system in the event that an error occurs.

Maintainability:

This quality requirement refers to the ability of the system to be repaired/maintained both easily and timely. The ease at which a system can be maintained relies on the implementation of the system which needs to allow for quick access, comprehension of the problem through alerts which results in timely repairs/maintenance. The system software must be able to alert/notify technicians of problems, the alert should contain a detailed and relevant error message The system software must keep a detailed list containing what is needed for the next maintenance i.e. printing paper,ink etc. Thus the measure of quality would be the time taken to repair/maintain the system. The system software must be able to alert/notify technicians of problems, the alert should contain a detailed and relevant error message The system software must keep a detailed list containing what is needed for the next maintenance i.e. printing paper,ink etc.

Auditability/Monitorability:

This quality requirement refers to the ability of the system to be examined or scrutinized using information stored by the system regarding changes, problems and everyday usage. Auditability/Monitorability is completely reliant on well structured and detailed logs/records that contain relevant information regarding the specific issue. The system must keep detailed records of daily events/transactions Thus the measure of quality would be how consistent the logging of events would be.

Intergratebility:

This quality requirement refers to the ability of the system to be merged or combined into/with another system or subsystem. It is imperative in modern times that systems can be used in conjunction with one another. The system must be implemented in a modular manner so that changes can be pinpointed within certain subsystems to prevent changes in other unrelated subsystems Thus the measure of quality would be time-consuming and expensive it would be to adopt a certain system so that it would function when being integrated with/in another system.

Cost:

This quality requirement refers to the overall costs associated with the system. These costs manifest in a multitude of aspects the largest being: initial capital in terms of implementation (labour costs, hardware costs, software costs), the cost of needed to ensure that the system is in working order(maintenance and repairs) and future plans for upgrading the system. However, the overall cost would be

lower if the number of errors would be less due to the obvious reduction in repair cost. A lower amount of errors in the system through good implementation using practices such as test-driven development Thus the measure of quality increasing the mean time to failure it which in less overall monetary value needed in the system. The lower amount of errors in the system through good implementation using practices such as test-driven development

Usability/Interaction:

This quality requirement refers to how easily the system can be learned and effectively used. The system components which is in direct contact with human interaction should be designed such that it is easy to understand and operate. This includes both internal(in terms of the ATM software) and external (in terms of the built-in keypad/keyboard). The system must provide clear tips/instructions to assist in specific problems(this could take place in the form of helpful notifications/pop-ups with also offer language choices) .The measure of quality could be quantified according to user scores given to the system after it's used by a user in the form of usability tests which measure performance based on the success rate at performing various tasks.

3.2 Constraints

3.2.1 Schedule Constraints - The project specification imposes a six-week limitation to provide a prototype that has 3 basic functionalities and at least 2-factor authentication with a backend that has CRUD. Given the different software requirements specifications received from phase one. The more innovative features such as conversion of currencies in the application will be abandoned due to the tight schedule.

3.2.2 Design Constraints

3.2.3 Policies and Regulations

Remote Access Permissions Tech support staff will require access to a client's Next-Gen ATM application remotely. Granted the need in access is important, it will be a violation of a client's privacy by imposing access to the device. Awareness of the access needs to be given to the user and the user needs to give special permission, through authentication, for the access to be granted

Privacy Infringement The introduction of a new system that collects data may spark fear ("Surveys since the 1970s show that loss of privacy is associated with the quantity of personal information collected, and that fear of privacy infringements constantly increases with the integration of computers in everyday life") []

3.2.4 Physical Constraints

3.3 Technology Requirements (Technical)

4 User Characteristics

Intended users and which subsystem the respective user interacts with:

4.1. Client - Uses the UI to interact with their accounts. This includes utilising the authentication subsystem to enable access to the transaction subsystem which includes features to withdraw, deposit, query bank balance or print mini statements. The client can also use geo-location subsystem to find the nearest usable ATM within the current location.

4.2. Analyst - Mainly use the Report and analysis subsystem to monitor ATM utilization by clients.

4.3. Auditors - Use the report and analysis subsystem to audit transactions that have occurred(mainly when, who and how much money is being transacted)[2]

4.4. System Administrator - Use backend to maintain, updated and monitor the performance of the entire system

5 Use Cases

UC.X *Use Case Name*
Description

UC.1 Client request transactions
Client request a list of transactions that they want to fulfill before going to an ATM or while waiting in a queue.

UC.2 Client requests to login
Client presents authentication token in order to login to their account

UC.3 Client requests to confirm request list
Client presents authentication token/OTP to confirm their request list

UC.4 Client requests Authentication Assistance
Tech Support helps client after multiple failed authentication attempts

UC.5 Client checks balance
A client wishes to check the balance on their bank account

UC.6 Client Withdrawal
A client wants to draw money from their bank account

UC.7 Client Deposit
A client wants to deposit money into their bank account

UC.8 Client generates mini-statement
A client wants to print a mini-statement to check on their transactions

UC.9 Client looks for ATM
A client wants to find the nearest functional ATM to process their transaction

UC.10 Client goods purchase
A client wants to purchase goods in a store using money on their account

UC.11 Client reports faulty ATM
A client reports a faulty ATM that appears to be available on App

UC.12 System Administrator identifies ATMs for Repair
The System Administrator wants to know which ATMs require repairs

UC.13 System Administrator requests usage report
System Administrators requires a compiled usage report of the NextGenATM App

UC.14 Auditor checks transactions
The Auditor wants to check that the transactions being processed by the system all balance with money allocated to the transaction

UC.15 Analyst checks ATM traffic
The Analyst wants to find out how many clients use a particular ATM over a span of time

UC.16 Client approves purchase
A Client wants to be able to accept or decline a purchase to avoid fraudulent purchases

UC.17 Tech Support remote access
Tech Support requires remote access to a client's app (permission required) to assist client with transaction

UC.18 Client token/OTP generation
A Client and Authentication service token/OTP generation shall be synchronised (smartphone devices only)

6 Subsystem 1 (Authentication)

6.1 Domain Model

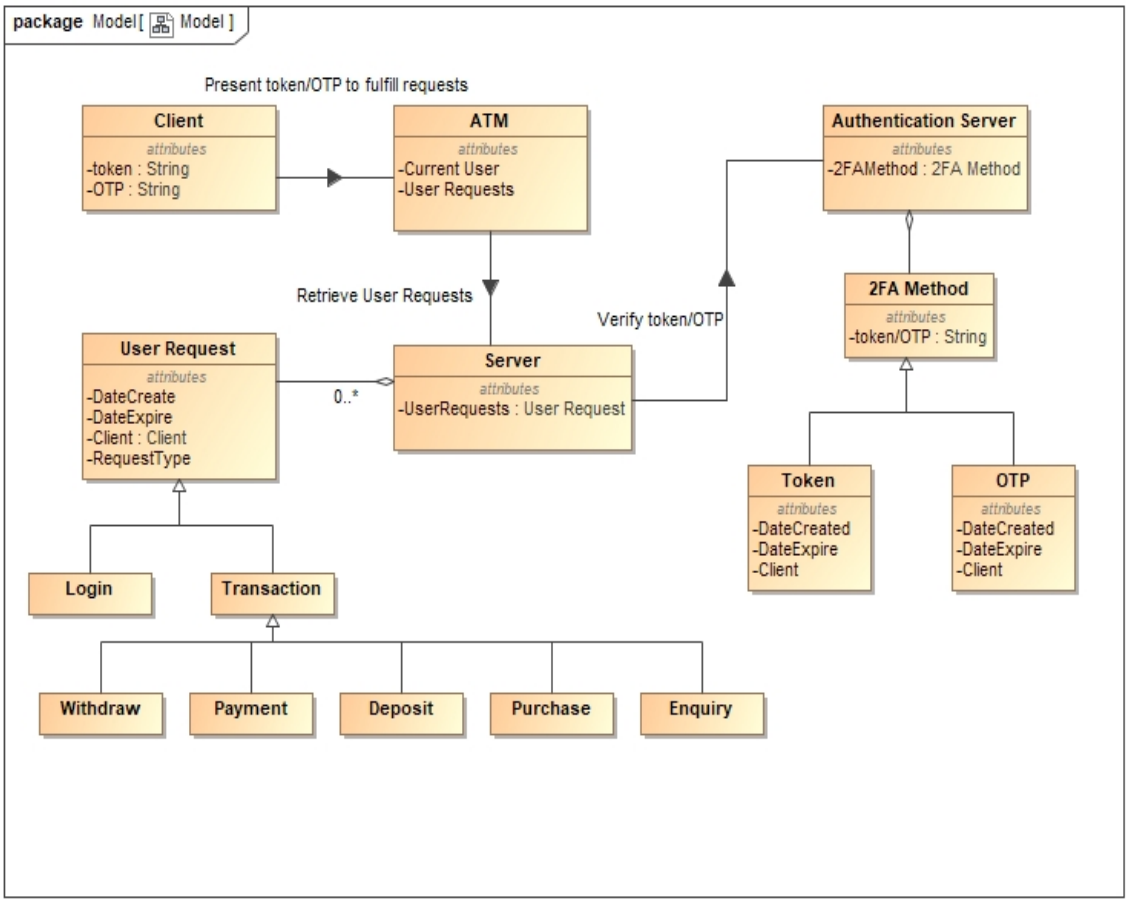

Figure 2: System Deployment Model

6.2 Functional Requirements

R1.1 The Authentication Subsystem shall allow authentication through non-smartphone devices

R1.1.1 The Authentication Subsystem shall authenticate requests given an OTP.

R1.1.2 The Authentication Subsystem shall accept requests if a valid OTP was given.

R1.1.3 The Authentication Subsystem shall decline requests if an invalid OTP was given.

R1.1.4 The Authentication Subsystem shall decline requests if an expired OTP was given.

R1.2 The Authentication Subsystem shall allow authentication through smartphone devices.

R1.2.1 The Authentication Subsystem shall authenticate requests given a NFC device.

R1.2.2 The Authentication Subsystem shall authenticate requests given an OTP.

R1.2.3 The Authentication Subsystem shall allow valid tokens/OTPs.

R1.2.4 The Authentication Subsystem shall decline invalid or expired tokens/OTPs.

R1.2.5 The Authentication Subsystem token/OTP generation shall be synchronised with online application users.

R1.2.6 The Authentication Subsystem token/OTP generation shall be synchronised with offline application users.

6.3 Subsystem Traceability Matrix

	UC.2	UC.3	UC.4
R1.1.1		X	
R1.1.2		X	
R1.1.3		X	
R1.1.4		X	
R1.2.1	X	X	
R1.2.2		X	
R1.2.3	X	X	
R1.2.4	X	X	
R1.2.5			X
R1.2.6			X

7 Subsystem 2 (Transactions)

7.1 Domain Model

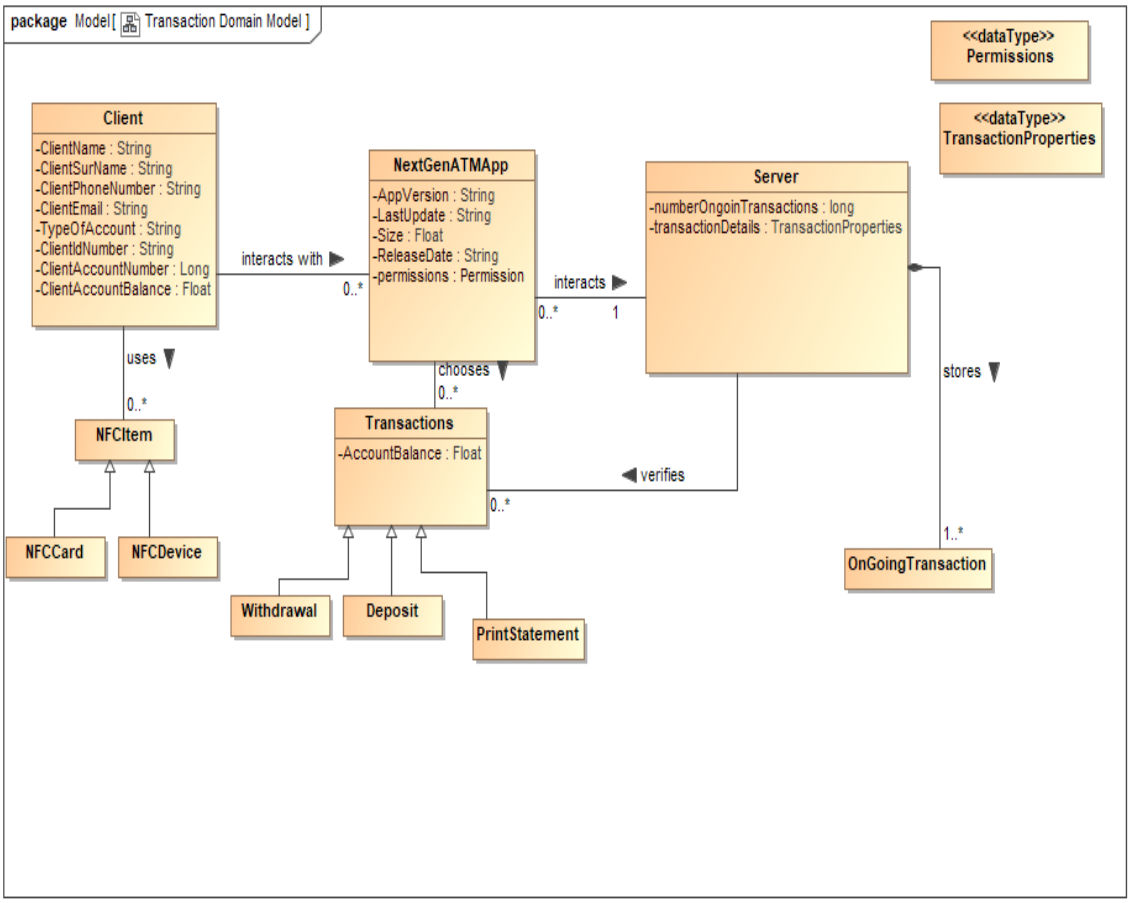

Figure 3: Transaction Domain Model[1]

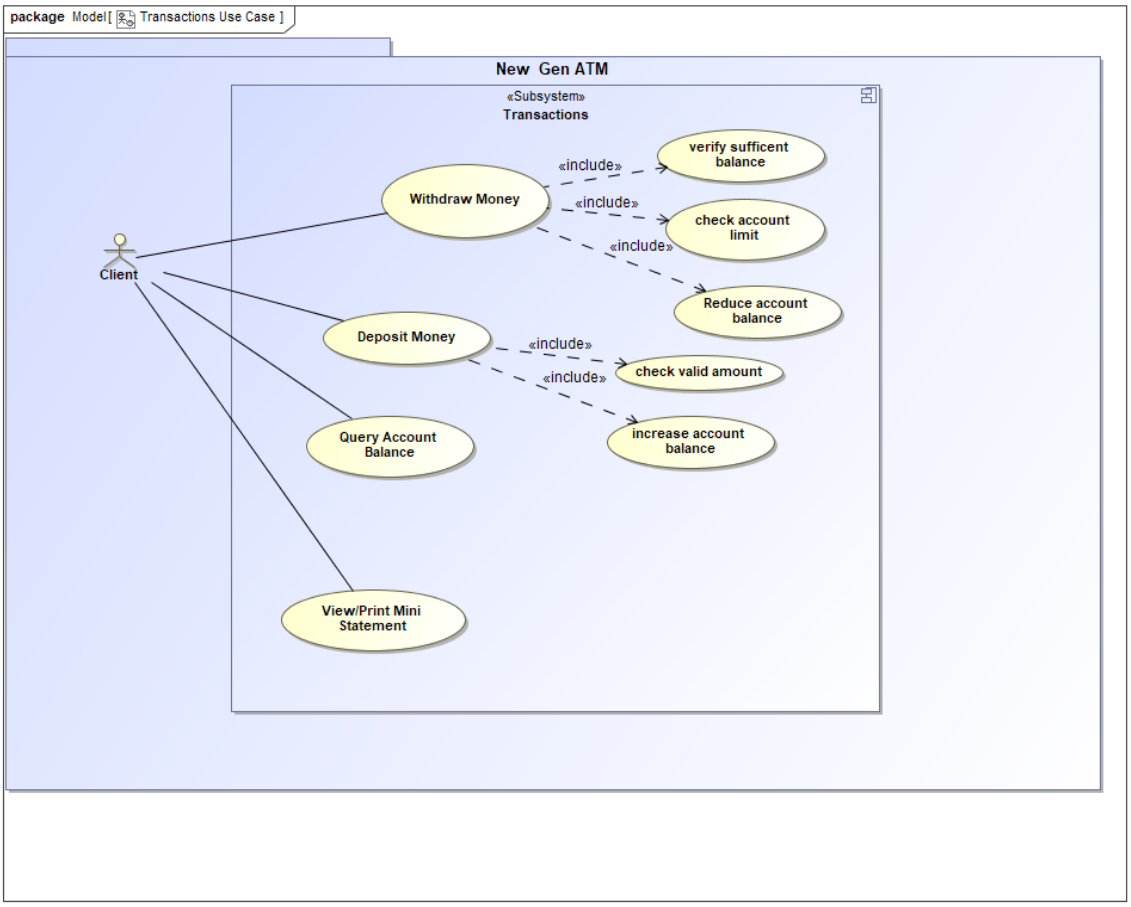

Figure 4: Transaction Use Case

7.2 Functional Requirements

R2 The Transaction Subsystem shall support four categories of transactions

- **R2.1** The Transaction Subsystem shall support the withdrawal of money
- **R2.2** The Transaction Subsystem shall support the deposit of cash
- **R2.3** The Transaction Subsystem shall support the querying/viewing of bank account balance
- **R2.4** The Transaction Subsystem shall support the printing of mini statement
- **R2.5** The Transaction Subsystem shall provide a list of transactions available

7.3 Subsystem Traceability Matrix

		UC.5	UC.6	UC.7	UC.8
R2.1			X		
R2.2				X	
R2.3		X			
R2.4		X			X
R2.5	X				

8 Subsystem 3 (Geo-location)

8.1 Domain Model

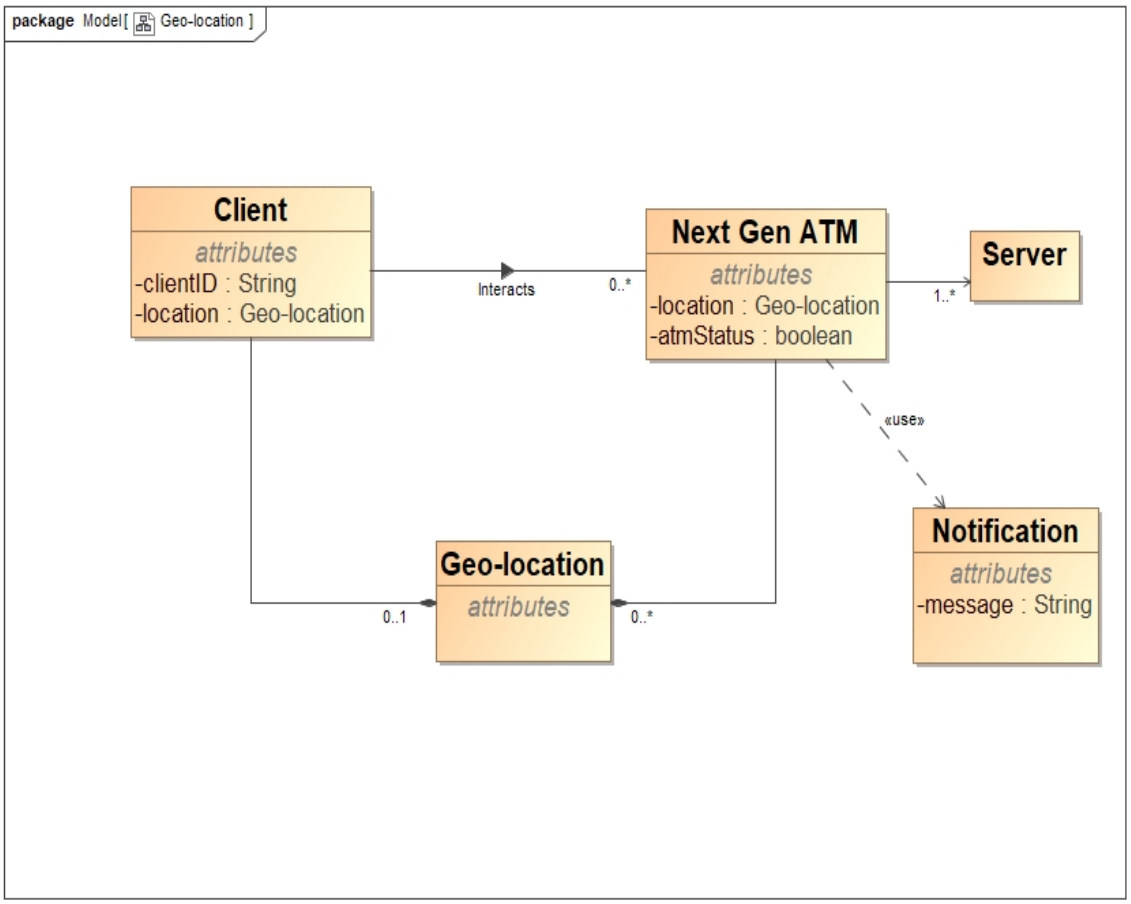

Figure 5: Geo-location Domain Model

8.2 Functional Requirements

R3.1 The Geo-location subsystem should provide the location of each of the New Gen ATM's.

R3.1.1 The subsystems should show the users the available New Gen ATM's in the area

R3.2 The Geo-location subsystem should notify users about its availability.

R3.2.1 The subsystems should notify the users when the system is offline.

3.2.2 The subsystems should show the nearest Next Gen ATM in the area.

8.3 Subsystem Traceability Matrix

	UC.9	UC.12
R3.1.1	X	
R3.2.1		X
R3.2.2	X	

9 Subsystem 4 (Mobile Subsystem)

9.1 Domain Model

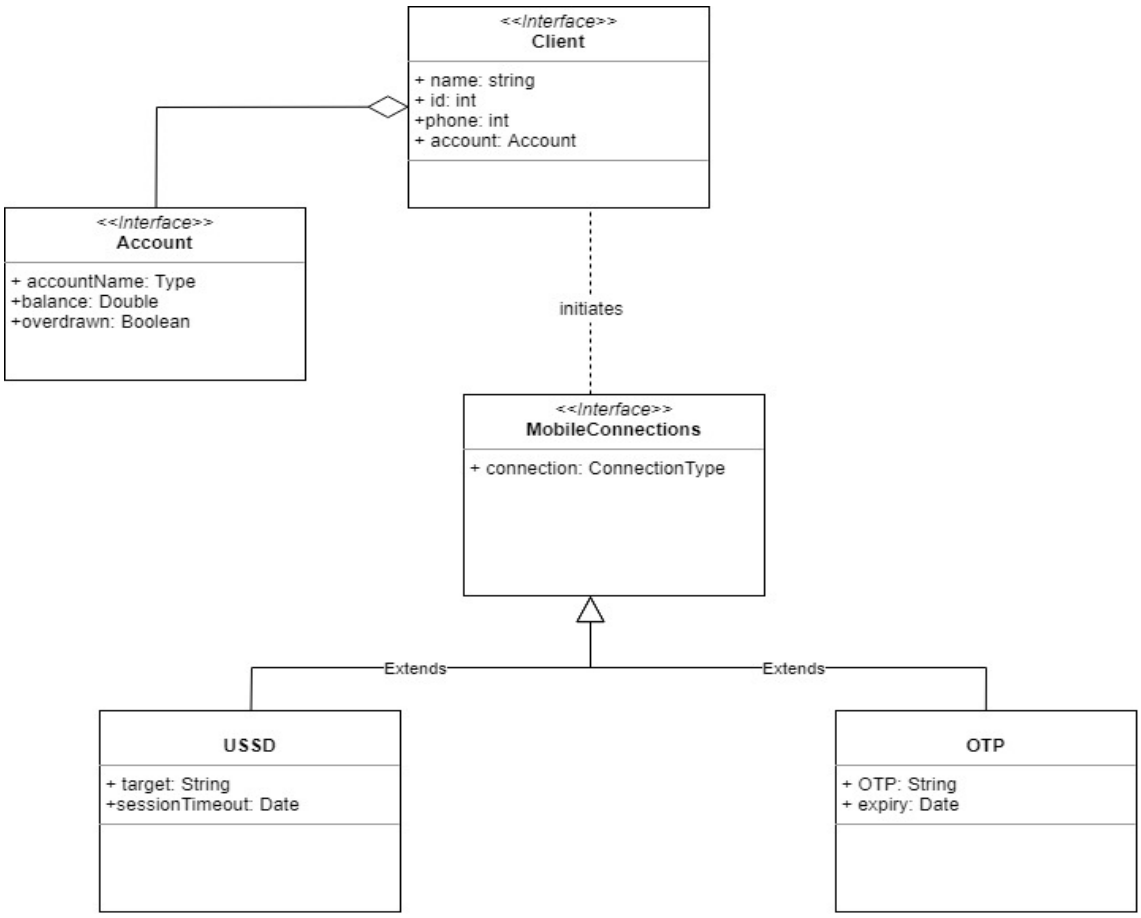

Figure 6: Mobile Domain Model

9.2 Functional Requirements

R4.1 The mobile subsystem shall allow users to initiate USSD connections with the whole system to allow for transactions.

R4.2 The mobile subsystem shall enable the system to generate and send One Time Pins to Users

R4.3 The mobile subsystem shall allow the system itself to end USSD sessions.

R4.4 The mobile subsystem shall allow users to send back OTP's sent from the system in order to allow for authentication.

9.3 Subsystem Traceability Matrix

	UC.2	UC.3	UC.4
R4.1	X	X	X
R4.2	X	X	X
R4.3	X	X	X
R4.4	X	X	X

10 Subsystem 5 (Report and Analysis)

10.1 Domain Model

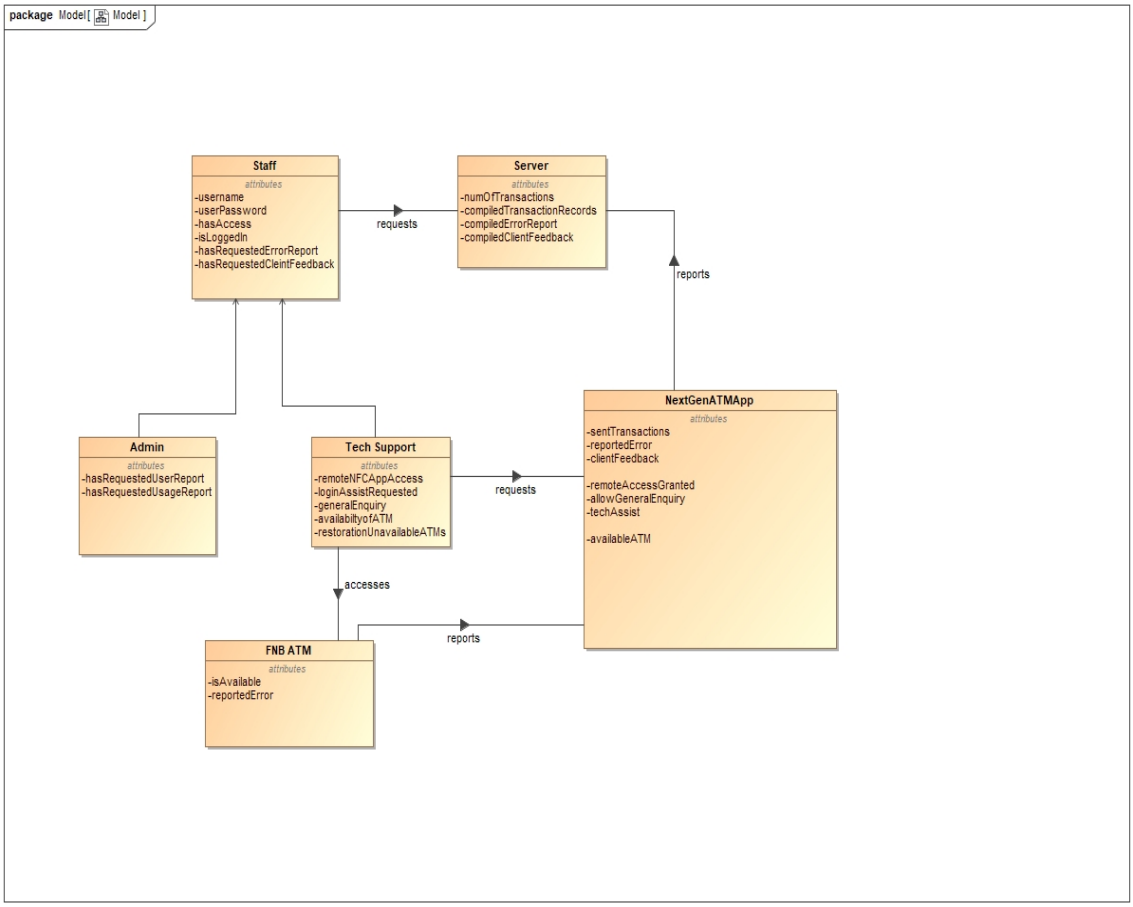

Figure 7: Mobile Domain Model

10.2 Functional Requirements

R5.1 Report and Analysis shall allow staff member access

R5.1.1 Admin staff can only access user feedback and app reports

R5.1.2 Tech support reviews error report

R5.1.3 Tech support can access App remotely

R5.2 FNB ATM should report to the NextGenApp

R5.2.1 App should know whether an ATM is available

R5.2.2 App will know available ATM within it's location

R5.3 Tech support as remote access to FNB ATM

R5.3.1 Tech support attempt to restore ATM remotely

R5.3.2 Tech support can view available ATMs

R5.3.3 Tech support can view ATM's transaction lists

10.3 Subsystem Traceability Matrix

	UC.9	UC.11	UC. 12	UC.13	UC.14	UC.15	UC.17
R5.1.1			X	X	X		
R5.1.2		X					
R5.1.3							X
R5.2.1	X	X					
R5.2.2	X	X					
R5.3.1							X
R5.3.2			X				X
R5.3.3						X	

11 Subsystem 6 (QR Payment)

11.1 Domain Model

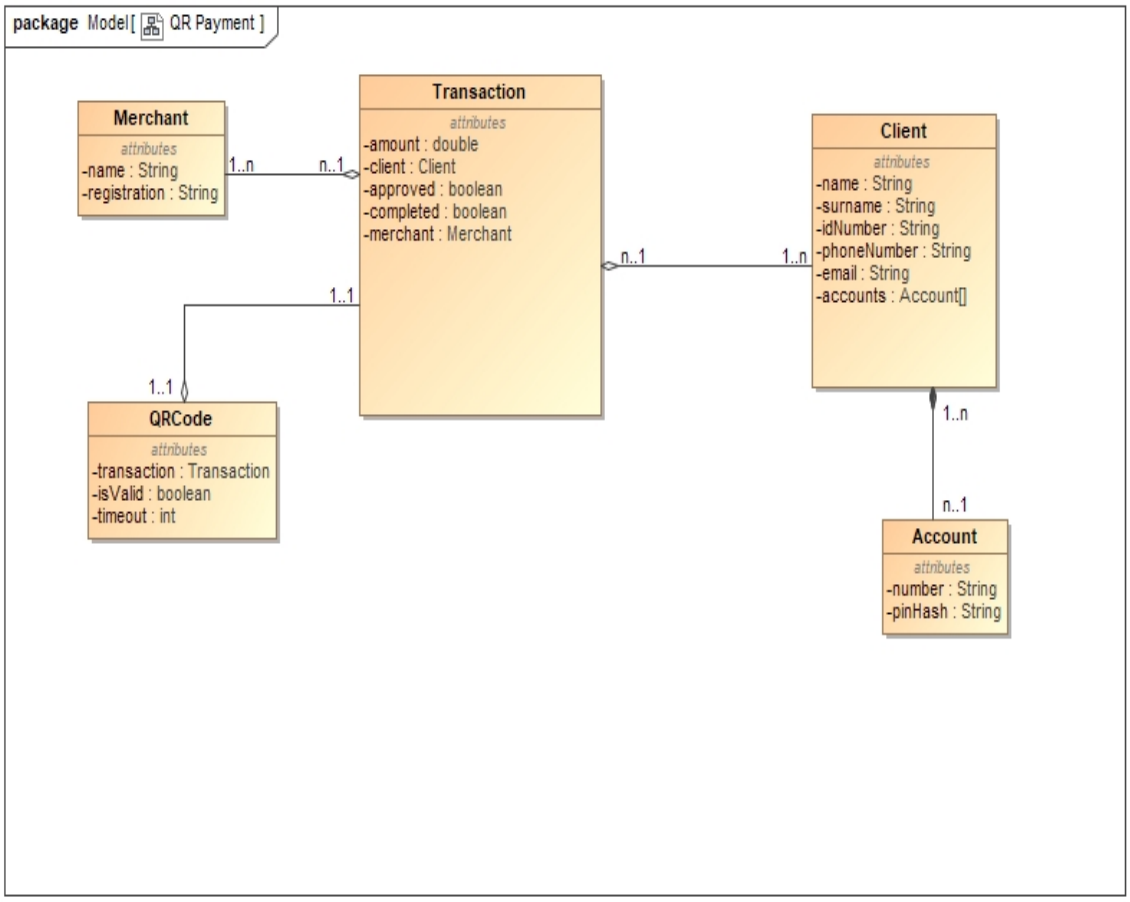

Figure 8: QR Payment Domain Model

11.2 Functional Requirements

R6.1 The QR Payment subsystem shall allow users to generate a QR code for payment.

R6.2 The QR Payment subsystem shall accept a merchant transaction authorized by QR code

R6.3 The QR Payment subsystem shall allow users to approve a merchant transaction authorised by QR code

R6.4 The QR Payment subsystem shall block a merchant transaction declined or not approved by the user

11.3 Subsystem Traceability Matrix

	UC.10	UC.16
R6.1	X	
R6.2	X	
R6.3		X
R6.4		X

12 Actor-System Interaction Model

12.1 Words and their meaning in this context

Token Token generated by the Mobile Application through smartphones and output through their build-in NFC tag. For security reasons this can only be used when logged into the Mobile Application and the token should expire and a new one should be generated every few seconds. The Token can be used to log into the ATM, or confirm transaction request.

OTP One Time PIN, generated by the Mobile Application through smartphones. If a user does not have a smartphone, the OTP won't be replaced after it is sent via SMS and will expire after a few minutes. If a user does have a smartphone, the OTP will be viewable from in-app, and will generate a new OTP when it expires. The OTP can only be used to confirm transaction requests.

Actor-System Interaction Model for Use Case 3. Note the Client interacts with the ATM System to confirm their requests, after that the ATM System ask the Authentication System if the token/OTP is valid. The Authentication System either accept or decline the token/OTP and notifies the ATM System. The response received by the ATM System is then given to the Client.

Precondition	
Valid token/OTP - throws invalid token/OTP.	
Actor: User	**System:** ATM System
	0. Show Users Requests
1. Present token/OTP as confirmation to fulfill Requests	2. Authenticate user given the token/OTP
4. User sees confirmation and details of fulfilled Requests	3. Fulfill User Requests
Postcondition	
User Requests have been fulfilled.	

Precondition	
Valid token/OTP - throws invalid token/OTP.	
Actor: ATM System	**System:** Authentication System
1. Present Users token/OTP to confirm validity	2. Authenticate user given the token/OTP
3. Show confirmation and details of fulfilled Requests	

Postcondition
User Requests have been fulfilled.

13 System Traceability Matrix

	Authentication	**Transactions**	**Geo-location**	**Mobile**	**R&A**	**QRPayment**
UC.1		X				
UC.2	X			X		
UC.3	X			X		
UC.4				X		
UC.5		X				
UC.6		X				
UC.7		X				
UC.8		X				
UC.9			X		X	
UC.10						X
UC.11					X	
UC.12			X		X	
UC.13					X	
UC.14					X	
UC.15					X	
UC.16						X
UC.17					X	
UC.18	X					