

Practical Record

**On**

Software Engineering Laboratory (19PC2CS06)

**Submitted to**

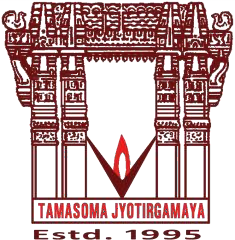
V N R V i g n a n a J y o t h i I n s t i t u t e o f E n g i n e e r i n g & T e c h n o l o g y

A n a u t o n o m o u s I n s t i t u t e – N A A C ‘ A + + ’

a n d N B A A c c r e d i t e d

## Bachelor of Technology in Computer Science and Engineering

(B.Tech II Year IV Sem)



***Submitted By***

VNR Vignana Jyothi Institute of Engineering & Technology Bachupally, Nizampet (S.O), Hyderabad–90



## VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

#### Bachupally(v), Hyderabad, Telangana, India.



***NAME*………… ……………………**

***DEPARTMENT OF*……………CSE………………………………….**

***ROLL NO*………………….…………………...………….**

***LABORATORY*: SOFTWARE ENGINEERING.**

### *CLASS*……………………………… ………………………………



VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

#### Bachupally(v), Hyderabad, Telangana, India



***CERTIFICATE***

Certified that this is the bonafide record of the practical work done during

the academic year 2022-2023… by the student

Name…………… ……………………………………………….

Hall Ticket No: …………..…………… Class…………………B…………………

In the Laboratory: …. SOFTWARE ENGINEERING……………………………………………

Department of …………………………CSE…………………………………………………………

Signature of the HoD Signature of the Staff Member

Date of Exam……………………….

Signature of the Examiners

Internal examiner External Examiner



Table of Contents

Software Development Lifecycle 5-8

Software requirements specification 9

SRS IEEE Template 9

[SRS for ATM 10-17](#_TOC_250006)

[Function Oriented Design 18](#_TOC_250005)

[Object Oriented Design 19](#_TOC_250004)

[Definition And Purpose of UML 20](#_TOC_250003)

Basic Building Blocks od UML 21

[UML Diagrams for ATM 22](#_TOC_250002)

Usecase 22

Class 23

Object 24

State chart 25

Activity 26

Sequence 27

Component 28

[Tools Of UML 29](#_TOC_250001)

[Open source 29-30](#_TOC_250000)

Licensed 32-33

**Software Development Lifecycle (SDLC)**

Software Development Life Cycle (SDLC) is a framework that defines the steps involved in the development of software at each phase. It covers the detailed plan for building, deploying and maintaining the software.

SDLC defines the complete cycle of development i.e. all the tasks involved in planning, creating, testing, and deploying a Software Product.

The 6 main phases in SDLC are:

1. Initial
2. Analysis
3. Design
4. Coding
5. Testing
6. Delivery and Maintenance

Here is the information regarding each phase, roles and responsibilities in that phase, the process involved and the proof documents.

#### INITIAL

##### Role:

Business Analyst (BA)

##### Process:

The Business Analyst visits the customer’s place and based on the questionnaire prepared , he will extract the requirements . Various elicitation methods are used to extract the requirements .

##### Proof Document:

The proof documents of the initial phase is **Business Design Document(BDD) or Functionality Requirement Specification (FRS)**

#### ANALYSIS

##### Role:

System Analyst (SA)

##### Process:

* Analysing the requirements by conducting feasibility study .
* We conduct tentative planning for schedule , budget and technology.

##### Proof Document:

The output document after the completion of analysis is **System Requirements Specification**(SRS). It contains all hardware and software details. A standard template for SRS is given by IEEE.

#### DESIGN

##### Role:

Chief Architect (CA) – HLD Technical Leads (TL) -LLD

##### Process:

The input document for this phase is SRS. Design process is carried out in two ways:

* **High Level Design** (HLD) – It is the design process in which how many modules that the project can be divided into can be assessed.
* **Low Level Design** (LLD) – It is another type of design process in which how many sub-modules a module can be divided into can be planned.

##### Proof Document :

The output document after the completion of Design is **Technical Design Document**(TDD).

It contains design information in the form of object diagrams, class diagrams , sequence diagram , flow charts done using UML tools.

#### CODING

##### Role:

Developer (or) Programmer

##### Process:

The input document for this phase is TDD.

The source code is developed based on the guidelines provided from quality standards. Some coding standards include:

* The function should follow comments
* Indentation
* The programs in all modules should be consistent and clear.

##### Proof Document :

The output document after the completion of Coding phase is **Source Code Document** (SCD).

#### TESTING

##### Role:

Test Engineer (TE)

##### Process:

The input document for this phase is SCD.

##### BDD Review:

The test engineer will get the information from BDD . This is called BDD review.

##### Preparation Of RR:

The Test Engineer will review the BDD. On understanding the functionality , if he gets some queries, all of them must be listed in a document known as Review Report(RR).

##### Sending RR to BA:

On sending the RR to BA , TE gets clarification from the BA. At this point the Test Engineer has 100% understanding of functionality.

##### Preparation Of Test Case Document:

* + All the testable features must be covered in the testing process.
  + A document has to be prepared with the checklist of all items that are to be tested. This document is known as Test Case Document.
  + Application must be stable irrespective of giving wrong or right input.
  + Testcase is defined as the perception or a case with which the probability of finding the defects is high.

##### Execution of test cases to test functionality:

* + Once the defects are uncovered , they will be listed in Defect Profile Document (DPD).
  + This document is sent to the author of SCD i.e. the developer.
  + Developer then gives the explanation and clarification.

##### Proof Document :

The output document after the completion of Testing phase is **Test Case Document (TCD)** and **Defect Profile Document (DPD).**

#### DELIVERY AND MAINTENANCE

##### Process:

* + Once the software testing phase is over and no bugs or errors left in the system then the final deployment process starts. Based on the feedback given by the project manager, the final software is released and checked for deployment issues if any.
  + Once the system is deployed, and customers start using the developed system, following 3 activities occur
    - **Bug fixing** - bugs are reported because of some scenarios which are not tested at all
    - **Upgrade** - Upgrading the application to the newer versions of the Software
    - **Enhancement** - Adding some new features into the existing software

The main focus of this SDLC phase is to ensure that needs continue to be met and that the system continues to perform as per the specification mentioned in the first phase.

# Software Requirements Specification (SRS) :

#### DEFINITION:

A software requirements specification (SRS) is a document that describes what the software will do and how it will be expected to perform. It also describes the functionality the product needs to fulfill all stakeholders (business, users) needs.

#### PURPOSE:

The purpose of this SRS document is to provide a detailed overview of our software product, its parameters, and goals. This document describes the project's target audience and its user interface, hardware, and software requirements. It lays out the functional and non-functional requirements of a system that is used for describing the user interactions.

**Template of SRS:**

##### Table of contents i

1. **Introduction** Error! Bookmark not defined.
   1. Purpose **Error! Bookmark not defined.**
   2. Document Conventions **Error! Bookmark not defined.**
   3. Intended Audience and Reading Suggestions **Error! Bookmark not defined.**
   4. Product Scope **Error! Bookmark not defined.**
   5. References **Error! Bookmark not defined.**
2. **Overall Description** Error! Bookmark not defined.
   1. Product Perspective **Error! Bookmark not defined.**
   2. Product Functions **Error! Bookmark not defined.**
   3. User Classes and Characteristics **Error! Bookmark not defined.**
   4. Operating Environment **Error! Bookmark not defined.**
   5. Design and Implementation Constraints **Error! Bookmark not defined.**
   6. User Documentation **Error! Bookmark not defined.**
   7. Assumptions and Dependencies **Error! Bookmark not defined.**
3. **External Interface Requirements** Error! Bookmark not defined.
   1. User Interfaces **Error! Bookmark not defined.**
   2. Hardware Interfaces **Error! Bookmark not defined.**
   3. Software Interfaces **Error! Bookmark not defined.**
   4. Communications Interfaces **Error! Bookmark not defined.**
4. **System Features** Error! Bookmark not defined.
   1. System Features one by one **Error! Bookmark not defined.**

##### Other Nonfunctional Requirements 5

* 1. Performance Requirements 5
  2. Security Requirements **Error! Bookmark not defined.**
  3. Seafety Requirements **Error! Bookmark not defined.**
  4. Software Quality Attributes **Error! Bookmark not defined.**
  5. Business Rules **Error! Bookmark not defined.**

##### Other Requirements 6

# SRS For ATM:

##### Introduction

The software ATM is to be developed for Automated Teller Machines (ATM). An automated teller machine (ATM) is computerized telecommunications device that provides a financial institution's customers a secure method of performing financial transactions, in a public space without the need for a human bank teller. Through ATM, customers interact with a user-friendly interface that enables them to access their bank accounts and perform various transactions.

##### Purpose

This document is intended for the following group of people:-

* Developers for the purpose of maintenance and new releases of the software.
* Documentation writers.
* Management of the bank.
* Testers.

##### Scope

This document applies to Automated Teller Machine software ATM. This software offers benefits such cash withdrawals, balance transfers, deposits, inquiries, credit card advances and other banking related operations for customers. It also allows the administrator to fix the tariffs and rules as and when required.

The software takes as input the login Id and the bank account number of the user for login. The outputs then comprise of an interactive display that lets the user select the desirable function that he wants to perform.

The software is expected to complete in duration of 4 months and the estimated cost is Rs. 20 lakhs.

##### Definitions, Acronyms, and Abbreviations

AC Alternate Current

AIMS ATM Information Management System

BMS Bank Management Software developed by KPM Bank

CDMA Code Division Multiple Access, a reliable data communication protocol CMS Card Management Software developed by KPM Bank

DES Data Encryption Standard

Dial-Up POS A message format for low cost communications

Electronic Journals For easier, safer information storage, related to modem.

MB Mega Bytes Ms Milliseconds.

Sec Seconds

SRS Software Requirements Specification.

TCP/IP Transmission Control Protocol/Internet Protocol. VGA Video Graphics Adaptor is a display standard.

##### References

The references for the above software are as follows:-

1. [www.google.co.in](http://www.google.co.in/)
2. [www.wikipedia.com](http://www.wikipedia.com/)

##### Overview

Section 1.0 discusses the purpose and scope of the software.

Section 2.0 describes the overall functionalities and constraints of the software and user characteristics. Section 3.0 and 4.0 details all the requirements needed to design the software.

##### The Overall Description

* 1. **Product Perspective**

The ATM is a single functional unit consisting of various subcomponents. This software allows the user to access their bank accounts remotely through an ATM. This software also allows to perform various other functions apart from just accessing his bank account such as mobile bill clearings etc. Some of its hardware components are cassettes, memory, drives, dispensers i.e. for receipts and cash, a card reader, printer, switches, a console, a telephone dialer port, a networking port and disks.

##### Product Functions

The major functions that ATM performs are described as follows:-

Language Selection:- The display provides user with a list of languages from which he can select any one in order to interact with the machine throughout that session. After the language selection the user is prompted with an option that whether he wants the selected language to be fixed for future use so that he is not offered with the language selection menu in future thus making the transaction a bit faster. User also has the freedom to switch to a different language mentioned in the list in between that session.

* Account Maintenance:- The various functions that a user can perform with his account are as follows:- Account Type:-The user has the freedom to select his account type to which all the transactions are made, i.e. he can select whether the account is current account or savings account etc.
* Withdrawal/Deposit: The software allows the user to select the kind of operation to be performed i.e. whether he wants to withdraw or deposit the money.
* Amount:- The amount to be withdrawn or deposited is then mentioned by the user.
* Money Deposition:- Money deposition shall be done with an envelope. After typing the amount to be deposited and verification of the same, the customer must insert the envelope in the depositary.
* Balance Transfer:- Balance transfer shall be facilitated between any two accounts linked to the card for example saving and checking account.
* Balance Enquiry:- Balance enquiry for any account linked to the card shall be facilitated.
* Billing:- Any transaction shall be recorded in the form of a receipt and the same would be dispensed to the customer. The billing procedures are handled by the billing module that enable user to choose whether he wants the printed statement of the transaction or just the updation in his account.
* Cancelling:- The customer shall abort a transaction with the press of a Cancel key. For example on entering a wrong depositing amount. In addition the user can also cancel the entire session by pressing the abort key and can start a fresh session all over again.

##### User Characteristics

There are different kinds of users that will be interacting with the system. The intended users of the software are as follows:-

User A: An inexperienced ATM customer. This user has little or no experience with electronic means of account management and is not a frequent user of the product. User A will find the product easy to use due to simple explanatory screens for each ATM function.

User B: An experienced customer. This user has used an ATM on several occasions before and does most of his account management through the ATM. There is only a little help session that too at the beginning of the session thus making the transaction procedure more faster.

Maintenance Personnel: A bank employee. This user is familiar with the functioning of the ATM. This user is in charge of storing cash into the ATM vault and repairing the ATM in case of malfunction. This user is presented with a different display when he logs in with the administrator’s password and is provided with options different from that of normal user. He has the authority to change or restrict various features provided by the software in situations of repairing.

##### Constraints

The major constraints that the project has are as follows:-

* The ATM must service at most one person at a time.
* The number of invalid pin entries attempted must not exceed three. After three unsuccessful login attempts, the card is seized/blocked and need to be unlocked by the bank.
* The minimum amount of money a user can withdraw is Rs 200/- and the maximum amount of money a user can withdraw in a session is Rs.10,000/- and the maximum amount he can withdraw in a day is Rs 30,000/-
* Before the transaction is carried out, a check is performed by the machine to ensure that a minimum amount of Rs 500/- is left in the user’s account after the withdrawal failing which the withdrawal is denied.
* The minimum amount a user can deposit is Rs 100/- and the maximum amount he can deposit is Rs 10,000/-. A user can select only that cellular operator for mobile bill clearings that is supported by the bank.
* The software requires a minimum memory of 20GB
* The database used should be Oracle7.0.

##### Assumptions and Dependencies

The requirements stated in the SRS could be affected by the following factors:

* One major dependency that the project might face is the changes that need to be incorporated with the changes in the bank policies regarding different services. As the policies changes the system needs to be updated with the same immediately. A delay in doing the same will result to tremendous loss to the bank. So this should be changed as and when required by the developer.
* Another constraint relating to the operating environment is that we are specific to Oracle Database.
* The project could be largely affected if some amount is withdrawn from the user’s account from the bank at the same time when someone is accessing that account through the ATM machine. Such a condition shall be taken care of.
* At this stage no quantitative measures are imposed on the software in terms of speed and memory although it is implied that all functions will be optimized with respect to speed and memory.

It is furthermore assumed that the scope of the package will increase considerably in the future.

##### System Features

* 1. **Functional requirements**

1. Remote Banking and Account Management

At the start, the user is provided with a log in screen and he is required to enter his PIN NO. and Account number which will be verified by the machine. In case of an unsuccessful attempt a user is asked again for his credentials but the maximum number of attempt given to the user is limited to 3 only, failing which his card is blocked and need to be unblocked by the bank.

After a log in, the user is presented with a list of language. The user can select any one in the list for interaction with the machine for the entire session.There is also a facility for the user to switch to any other language during that session.

After the language selection, the user is directed towards a main page that displays a set of option along with their brief description, enabling the user to understand their functioning. The user can select any of the listed option and can continue with the transaction.

At any moment if the user wants to abort the transaction, he is provided with an option to cancel it. Just by pressing the abort button he can cancel all the changes made so far and can begin with a new transaction. After the user is finished with his work, he is required to log out and then take his card out of the slot.

##### Validity Checks

In order to gain access to the system, the user is required to enter his/her correct user id/pin no and account no failing which his card may be blocked. The user can access only one account at a time and can enter only one account no. Also if the user is an administrator, he is required to enter his login id in order to access and change the facilities provided by the system.

##### Sequencing Information

The information about the users and their account should be entered into the database prior to any of the transactions and the backup be maintained for all account information

Error Handling/ Response to Abnormal Situations

If any of the above validation/sequencing flow does not hold true, appropriate error messages will be prompted to the user for doing the needful.

##### 4. External Interface Requirements

* + 1. User Interface Requirements

1. A login screen is provided in the beginning for entering the required username/pin no. and account number.
2. An unsuccessful login leads to a reattempt (maximum three) screen for again entering the same information. The successful login leads to a screen displaying a list of supported languages from which a user can select any one.
3. In case of administrator, a screen will be shown having options to reboot system, shut down system, block system, disable any service.
4. In case of reboot/ shut down, a screen is displayed to confirm the user’s will to reboot and also allow the user to take any backup if needed.
5. After the login, a screen with a number of options is then shown to the user. It contains all the options along with their brief description to enable the user to understand their functioning and select the proper option.
6. A screen will be provided for user to check his account balance.
7. A screen will be provided that displays the location of all other ATMs of same bank elsewhere in the city.
8. A screen will be provided for the user to perform various transactions in his account.

Other various user interface requirements that need to be fulfilled are as follows:-

The display screen shall be of 10" VGA color type. The display screen shall have 256 color resolution. The display screen shall also support touch screen facility. The speakers shall support Yamaha codecs. The keypad shall consist of 16 tactile keys. There shall be 8 tactile function keys. The keyboard will be weather resistant. The transaction receipt shall be 3.1" × 6". The statement receipt shall be 4.2" × 12". The deposit envelopes shall be 9" long and 4" wide.

##### Hardware Interface Requirements

 The ATM power supply shall have a 10/220 V AC manual switch. The ATM card should have the following physical dimensions:-

 The card reader shall be a magnetic stripe reader .

 The slot for a card in the card reader may include an extra indentation for the embossed area of the

card.

 There shall be a 40 column dot matrix receipt printer.

The statement dispenser shall be a maximum of 5" width and 0.5" thickness. The envelope depository shall be a maximum of 4.5" width, 10" length and 0.5" thickness. Screen resolution of at least 800X600-required for proper and complete viewing of screens.

##### Software Interface Requirements

In order to perform various different functions, this software needs to interact with various other softwares. So there are certain software interface requirements that need to be fulfilled which are listed as follows:-

The transaction management software used to manage the transaction and keep track of resources shall be BMS version 2.0. The card management software used to verify pin no and login shall be CMS version 3.0. Yamaha codecs 367/98 for active speakers. The database used to keep record of user accounts shall be Oracle version7.0.

##### Communication Interface Requirements

The machine needs to communicate with the main branch for each session for various functions such as login verification, account access etc. so the following are the various communication interface requirements that are needed to be fulfilled in order to run the software successfully:-

The system will employ dial-up POS with the central server for low cost communication. The communication protocol used shall be TCP/IP. Protocol used for data transfer shall be File Transfer Protocol.(FTP)

##### Other Non-functional Requirements

* 1. Performance Requirements

The following list provides a brief summary of the performance requirements for the software:

* + 1. Capacity

The ATM shall provide customers a 24 hour service.

* + 1. Dynamic requirements

The card verification time must not exceed 0.8 sec. under normal server workload and 1 sec. under peak server workload. The pin number verification time must not exceed 0.3 sec. under normal server workload and 0.5 sec. under peak server workload. Account balance display time must not exceed 3 sec. under normal server workload and 5 sec. under peak server workload. Account balance transfer time must not exceed 4 sec. under normal server workload and 6 sec. under peak server workload. Cash withdrawal transaction time must not exceed 4 sec. under normal server workload and 5 sec. under peak server workload. Deposit transaction time after insertion of the deposit envelope must not exceed 7 sec. under normal server workload and 8 sec. under peak server workload. Receipt printing time after must not exceed 3 sec. under normal server and peak server workload.

* + 1. Quality –

The primary objective is to produce quality software. As the quality of a piece of software is difficult to measure quantitatively, the following guidelines will be used when judging the quality of the software:

1. Consistency – All code will be consistent with respect to the style. (This is implied when adhering to the standard).
2. Test cases – All functionality will be thoroughly tested
   1. Software System Attributes
      1. Reliability

 The data communication protocol shall be such that it ensures reliability and quality of data and voice transmission in a mobile environment. The memory system shall be of non-volatile type.

* + 1. Availability

 The product will have a backup power supply in case of power failures. Any abnormal operations shall result in the shutting down of the system.

 After abnormal shutdown of the ATM, the system shall have to be manually restarted by a maintenance personnel.

 There should be no inconsistency introduced in the account during whose transaction the system is abnormally shut down.

* + 1. Security

 The system shall be compatible with AIMS security standards. The system shall have two levels of security i.e. ATM card and pin verification both authenticated by the CMS software. The Encryption standard used during pin transmission shall be triple DES.

 The password shall be 6-14 characters long. Passwords shall not contain name of customers as they are easy to be hacked. Passwords can contain digit, hyphen and underscore.

 User should be provided with only three attempts for login failing which his card needs to be blocked.

 There shall be a security camera installed near the ATM. There shall be a secured cash vault with a combination locking system. The product cabinet cover shall be manufactured using Fiber glass for security purposes.

* + 1. Maintainability

 The system components i.e. modem, memory, disk, drives shall be easily serviceable without requiring access to the vault.

 The system should have the mechanism of self-monitoring periodically in order to detect any fault.

 The system should inform the main branch automatically as soon as it detects any error. The kind of fault and the problem being encountered should also be mentioned by the system automatically.

# Function Oriented Design

**Function Oriented Design** is an approach to software design where the design is decomposed into a set of interacting units where each unit has a clearly defined function.

### DATA FLOW DIAGRAMS:

Data flow diagrams are used to graphically represent the flow of data in a business information system.

It describes the processes that are involved in a system to transfer data from the input to the file storage and reports generation.

It may be used as a communications tool between a systems analyst and any person who plays a part in the system that acts as the starting point for redesigning a system.

A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination.

### STRUCTURED CHART:

A structure chart (SC) in software engineering and organizational theory is a chart which shows the breakdown of a system to its lowest manageable levels.

They are used in structured programming to arrange program modules into a tree. Each module is represented by a box, which contains the module's name.

A structure chart illustrates the partitioning of a problem into subproblems and shows the hierarchical relationships among the parts.

# OBJECT ORIENTED DESIGN:

Object-oriented design is the process of planning a system of interacting objects for the purpose of solving a software problem. It is one approach to software design.

It is a technical approach for analyzing and designing an application, system, or business by applying object- oriented programming, as well as using visual modeling throughout the software development process to guide stakeholder communication and product quality.

### CLASS DIAGRAM:

The class diagram is the main building block of object-oriented modeling. It is used for general conceptual modeling of the structure of the application, and for detailed modeling translating the models into programming code. Class diagrams can also be used for data modeling.

The purpose of the class diagram can be summarized as −

* + - * Analysis and design of the static view of an application.
      * Describe responsibilities of a system.
      * Base for component and deployment diagrams.
      * Forward and reverse engineering.

### SEQUENCE DIAGRAM:

A sequence diagram is a type of interaction diagram because it describes how—and in what order—a group of objects works together.

It depicts the objects involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario.

# DEFINITION AND PURPOSE OF UML:

**Unified Modeling language** (UML) is a standardized modeling language enabling developers to specify, visualize, construct and document artifacts of a software system. Thus, UML makes these artifacts scalable, secure and robust in execution. UML is an important aspect involved in object-oriented software development. It uses graphic notation to create visual models of software systems.

#### Structure diagrams and their applications

Structuring diagrams show a view of a system that shows the structure of the objects, including their classifiers, relationships, attributes and operations

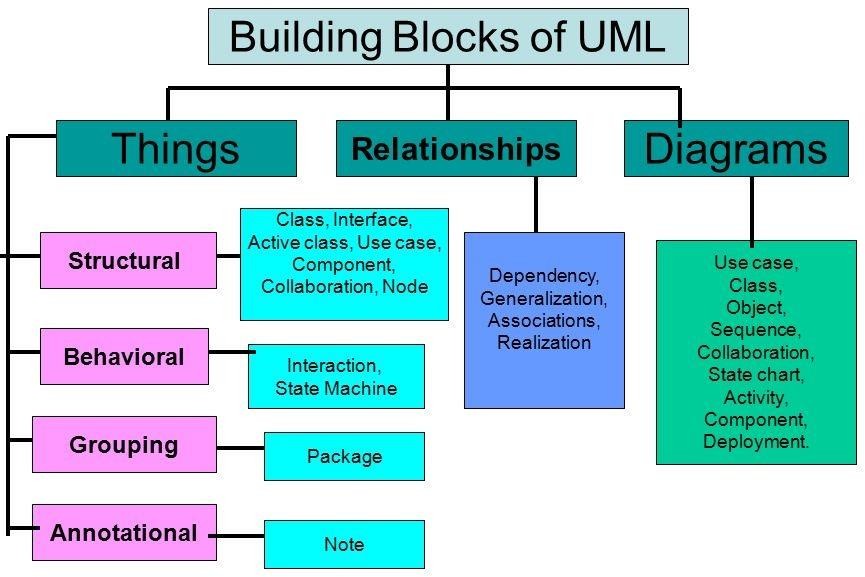
#### Behaviour diagrams and their applications

Behaviour diagrams are used to illustrate the behavior of a system, they are used extensively to describe the functionality of software systems.

#### Interaction diagrams and their applications

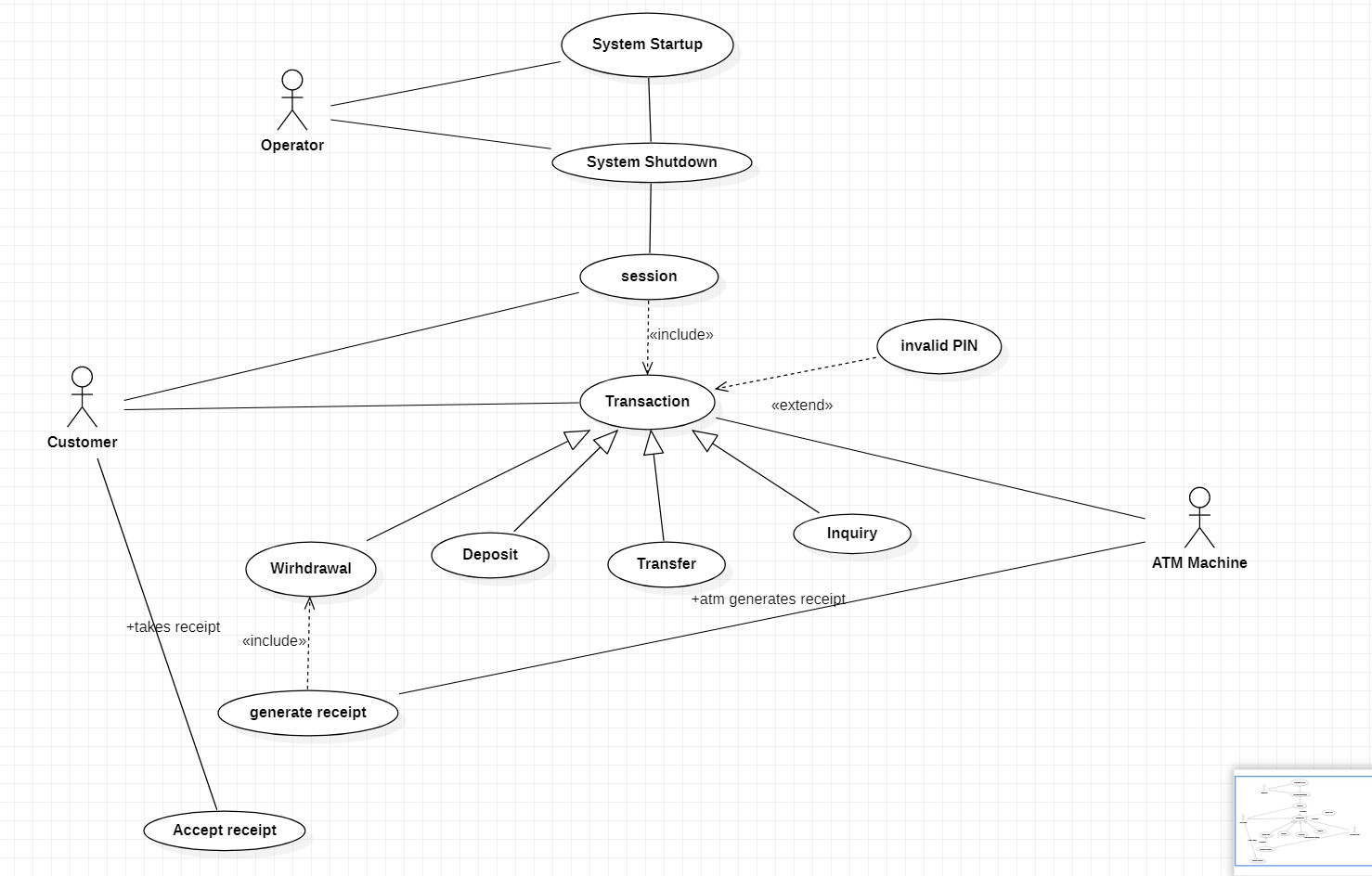
Interaction diagrams are subset of behaviour diagrams and emphasize the flow of control and data among the things in the system being modelled

# BUILDING BLOCKS OF UML:

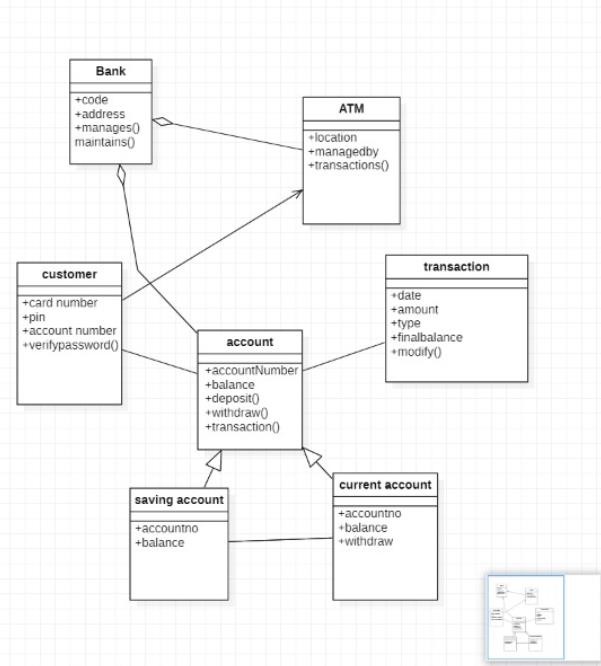


# UML DIAGRAMS FOR ATM:

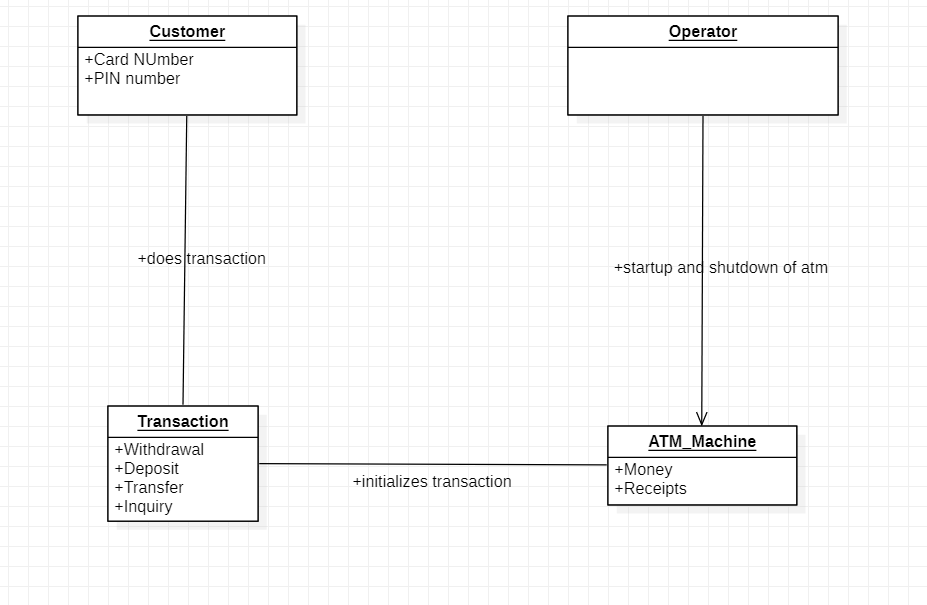
#### USECASE DIAGRAM



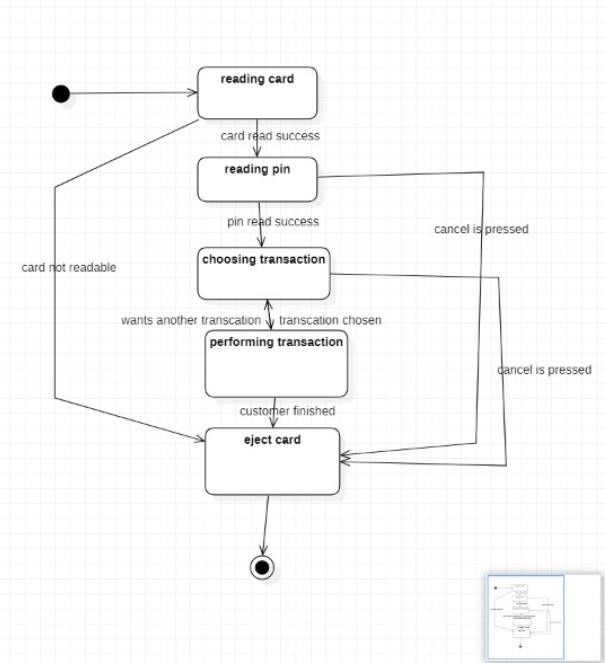
**CLASS DIAGRAM**



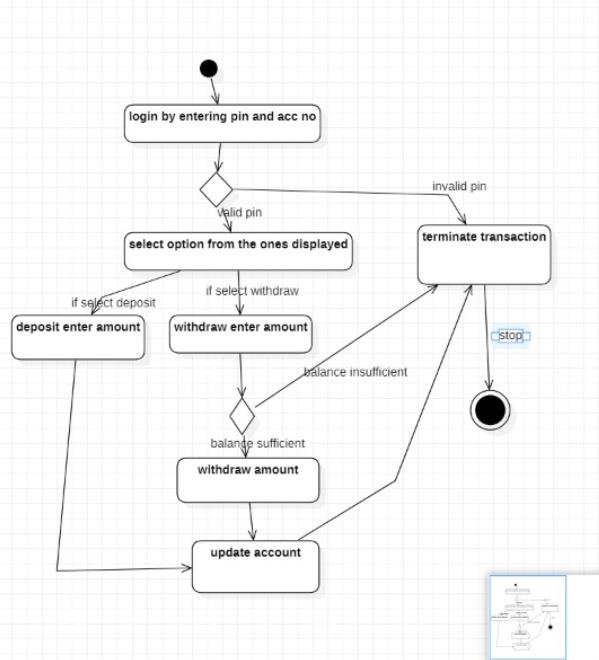
**OBJECT DIAGRAM**



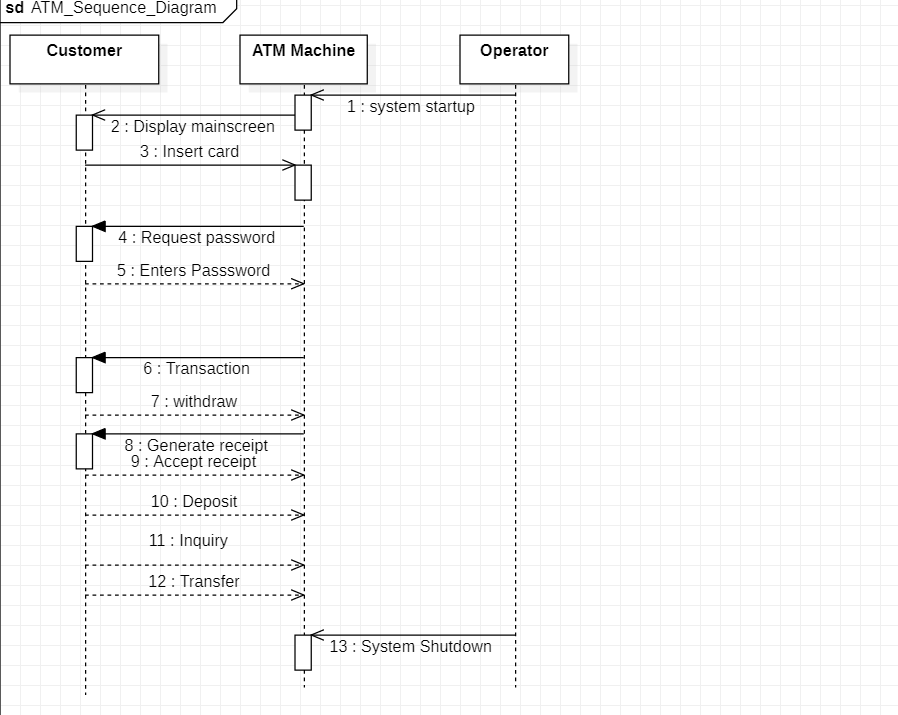
**STATE CHART DIAGRAM**



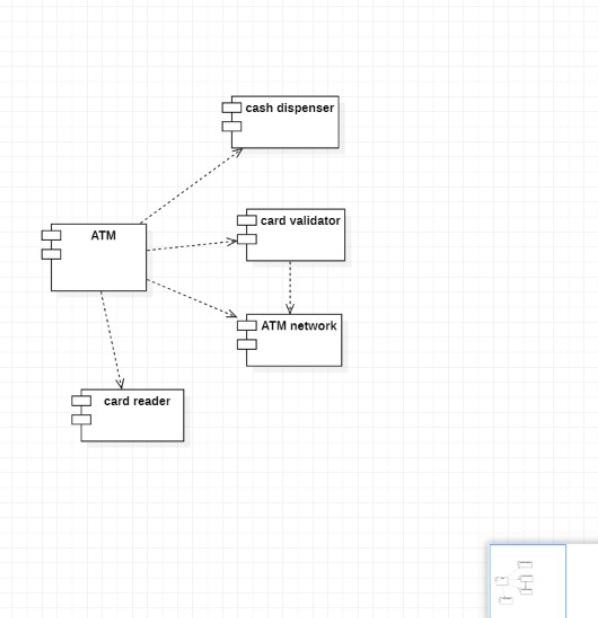
**ACTIVITY DIAGRAM**



**SEQUENCE DIAGRAM**



**COMPONENT DIAGRAM**



# TOOLS OF UML:

Since UML is a Unified Modelling Language, it is used to create meaningful, object-oriented models for a software application. It clearly represents the working of any hardware/ software system.

#### OPEN SOURCE:

##### STARUML:



StarUML is an open-source software modeling tool, which is provided by MKLab. It has come up with eleven different types of modeling diagrams.

FEATURES:

* It let you create Object, Use case, Deployment, Sequence, Collaboration, Activity, and Profile diagrams.
* It is a UML 2.x standard compliant.
* It offers multiplatform support (MacOS, Windows, and Linux).

##### UMBRELLO:

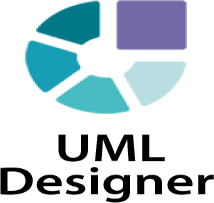


Umbrello is a Unified Modeling language tool, which is based on KDE technology. It supports both reverse engineering and code generation for [C++](https://www.javatpoint.com/cpp-tutorial) and [Java](https://www.javatpoint.com/java-tutorial).

Features:

* It implements both structural and behavioral diagrams.
* It imports C++ and can export up to a wider range of languages.

##### UML DESIGNER TOOL



The UML designer tool helps in modifying and envisioning UML2.5 models. It allows you to create all of the UML diagrams.

Features:

* + It provides transparency to work on DSL as well as UML models.
  + With the UML designer tool, the user can reuse the provided presentations.
  + It implements Component, Class, and Composite structure diagrams.
  + To start working with DSL, you can use UML legacy models.

##### ALTOVA:

UML Tools

Altova has provided UModel, which is another UML software modeling tool. It supports all types of 14 UML2 diagrams as well as SysML for the embedded systems. It also holds up for business process modeling for enterprise analysts. It generates visually designed software models by incorporating Java, C++, and C #or Visual Basic .NET.

Features:

* It provides a dedicated toolbar for an individual diagram.
* It offers unlimited undo/redo, which inspires to discover new ideas.
* In UML diagrams, you can easily add a hyperlink to any element.
* It also provides an intuitive color-coding, icons, customized alignment grid, and cascading styles for colors, fonts line size.

##### UMPLE:



Umple is an object-oriented and modeling language that textually supports state diagrams and class diagrams. It adapts JAVA, C++, and [PHP](https://www.javatpoint.com/php-tutorial), which results in more readable and short lines of code.

Features:

* It includes Singleton pattern, keys, immutability, mixins, and aspect-oriented code injection, which makes UML more understandable to the users.
* It enforces referential integrity by supporting UML multiplicity.

#### LICENSED TOOLS:

##### [Visual Paradigm](https://online.visual-paradigm.com/diagrams/features/uml-tool/):



Visual Paradigm is considered one of the best visual diagramming tools. If you want a platform that comes with a very basic, interactive and easy to use tool set for your UML diagramming needs, Visual Paradigm is most definitely one of the best options out there.

Visual Paradigm supports requirements management including user stories, use cases, SysML requirement diagrams and textual analysis.

##### [Lucidchart](https://www.lucidchart.com/pages/examples/uml_diagram_tool):



The platform is so interactive and versatile that majority of the developers and business professionals leverage their sources to get the UML diagrams sorted.

##### [Gliffy](https://www.gliffy.com/examples/uml-diagrams):



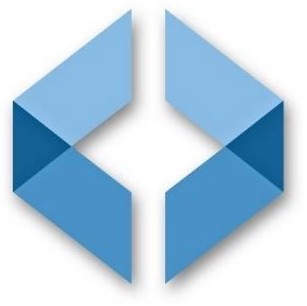
* It helps in easier and faster diagramming without any kinds of glitches as such.It helps in better visualization of the software in the graphical format. It allows you to collaborate with your team before the process of coding for the software starts.

##### [Creately](https://creately.com/lp/uml-diagram-tool/):



It has diagramming of the UML along with the collaboration option.

##### [SmartDraw](https://www.smartdraw.com/uml-diagram/uml-diagram-tool.htm):



It does come with a plethora of templates and extensions that you can integrated into your diagram for easy access of things. It does come with a very interactive user interface which is key when it comes to the UML diagramming.