Mini ATM System

Diploma in information communication Technology

Final Project Report

20.1P

School of Computing

National Institute of Business Management

Kandy

Mini ATM System

Final Project Report

Group Members

Pawani Maheshika	**KADICT201p – 005**
Dilini Ekanayake	**KADICT201p – 004**
Dilanjani Premarathna	**KADICT201P – 013**

Supervisor

Mrs.Inoka Abhayasinghe

Declaration

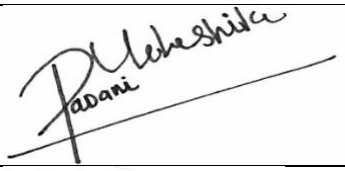
" I'm certify that this project does not incorporate without acknowledgement, any materials previously submitted for a Diploma in any institution and to the best of my knowledge and belief, it does not contain any material previously published or written by another person or my self except where due reference is made in the text. I also hereby give consent for my project report, if accepted, to be made available for Photocopying and interlibrary loans, and for the title and summary to be made available to outside organizations."

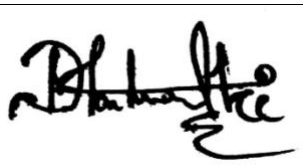
Supervisor : Mrs.Inoka Abhayasinghe

Date :

Signature : .....

Project : Mini ATM System

Name	**Index number**	**Signature**
Pawani Maheshika	Kadict201p-005	
Dilini Ekanayake	Kadict201p-004	

Dilanjani Premarathna	Kadict201p-013	

Acknowledgement

It is a great pleasure for us to knowledge all those who have contributed to words the conception origin and murturing of this report that is on system analysis the Mini ATM System.

The way can't walk it self. We have to walk on it for that we must have a guide.Maby guides have contributed to the successful completion of the project we would like to place on record our grateful thank to each on of them who helps us in this project report.

Our grateful thanks to go to our institute. Which provides us on opportunity as a project subject semester develop a report work skill in this system analysing.

Last but not the least we heartly thanks our teacher Mrs.Inoka Abhayasinghe

DETAIL OF CHAPTERS

3.0 System Requirements Study

3.1 User characteristics

3.2 Hardware and software Requirements

3.3 Constraints

4.0 System Analysis

4.1 study of current system

4.2 problem and weakness of current system

4.3 Requirements of new system

4.4 Feasibility study

4.5 Requirements validation

4.6 Functions of system

4.7 Data Modeling

4.7.1 flow chart or activity design

4.7.2 class diagram/ ER diagrams

4.7.3 system activity or object interaction diagram

4.7.4 data dictionary

4.8 functional and behavioral modeling

1.0 Introduction

1.1 Project Summary

An automated teller machine (ATM) or automatic banking machine (ABM) is a computerized
telecommunications device that provides the clients of a financial cashier, human clerk or bank teller. On most modern ATMs the customer is identified by inserting a plastic ATM card with a magnetic strip or a plastic smart card with a chip, that contains a unique card number and some security information such as an expiration date or CVVC(CVV) Authentication is provided by the customer entering a personal identification number(PIN).

1.2 Purpose

Using an ATM, customers can access their bank accounts In order to make cash withdrawals (or credit card advances) and check their account balances as well as purchase cellphone prepaid credit. If the currency being withdrawn from the ATM is different from that which the bank account is denominated in (eg. Withdrawing Japanese Yen from a bank account containing US Dollars), the money will be converted at a wholesale exchange rate. Thus ATMs often provide the best possible exchange rate for foreign travellers and are heavily used for purpose as well.

ATMs are known by various other names, including automated transaction machine, Automated transaction machine, automated banking machine, cashpoint (in Britain). Money machine , bank machine ,cash

machine, bole-in-the-wall. Banco mat (in various countries in Europe and Russia), Multibank (after a registered trade mark in Portugal), and any Time Money (in Sri Lanka)

1.3 Scope

The main purpose of the ATM division and information service is to provide the customers financial flexibility, worldwide acceptance and round-the clock convenience. Bank issues only VISA credit cards, the renowned credit card brand. cardholders can purchase good/service up to the credit limit and can reuse the credit facility upon repayment. Credit card is a safer substitute to cash and is the major mode of payment worldwide. standard chartered bank is the first to introduce the TAKA CREDIT CARD. The card is issued basically to a person's name and the specific person can use the card anywhere in Bangladesh. The business activity of premier Bank Credit Card section is to keep the records of all sales and customer's requests, the information of cardholders and reports them to necessary documents.

2.0 project Management

In this chapter we will discuss about project planning and scheduling. Our goal is to establish a pragmatic strategy for controlling, tracking and monitoring a complex technical project.

In project management following things must be done.

- Project planning and scheduling
- Risk Management
- Estimation

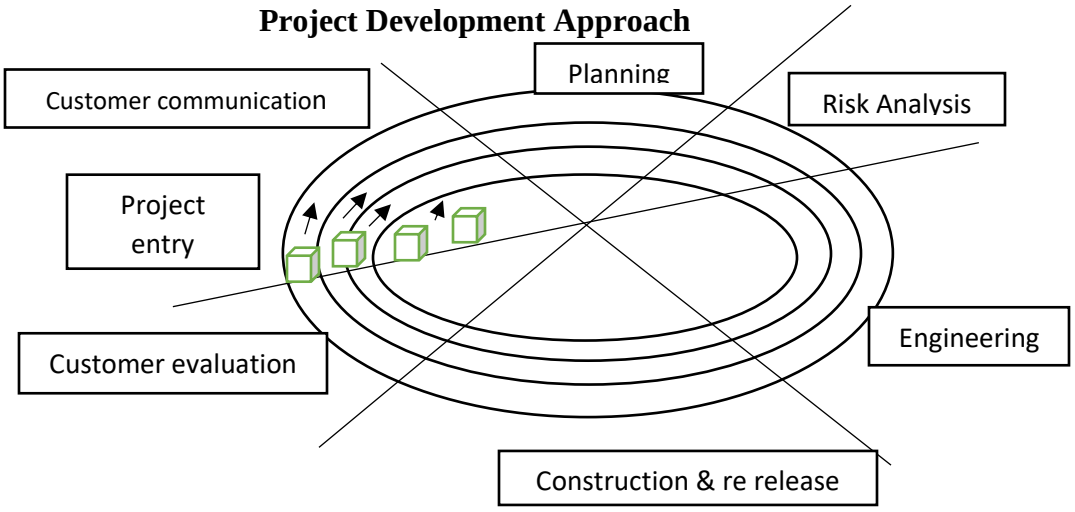
In project and the scheduling. Planning of the project is done. In scheduling different task are schedule according to the deadline of the project.

2.1.project planning and scheduling

Project planning must deals with the following things.

- **Project complexity :-** project complexity has a strong effect but is heavily influenced by past practitioner experience.
- **Project Size :-** As size increase the interdependency of elements also grow. Watch out for scope creep.
- **The degree of structural uncertainty :-** the degree to which requirements are solidified and the case of functional decomposition. The purpose of project planning is to ensure that the end result is completed on time. Within budget, and exhibits quality!

2.1.1 project development approach



The spiral model is an evolutionary software process model that couples the iterative nature of prototyping with the controlled and systematic aspects of the liner sequential model. It provides the potential for rapid development of incremental versions of the software. Using the spiral model, software is developed in series of incremental release.

The spiral model is divided into a number activities , also called task regions. There are between three and six task regions. Figure depicts a spiral model that contains six task regions.

- Customer communication – tasks required to establish effective communication between developer and customer.
- Planning – task required to define resources, timelines, and other project related information.
- Risk analysis – task required to assess both technical and management risks.
- Engineering- tasks required to build one or more representation of the application.
- Construction and release – task required to construct, test, install, and provide user support.
- Customer evolution – task required to obtain customer feedback based on evolution of the software representations created during the engineering stage and implemented during the installation stage.

Each of the regions is populated by a set of work tasks. Called a task set, that are adapted to the characteristics of the project to be undertaken. For small projects, the number of work tasks and their formality is low. For larger, more critical projects, each task region contains more work tasks that are defined to achieve a higher level of formality.

In our case ,we have to provide medium level of formality for making good project report. We will take decision about cost, schedule and number of iterations required to complete the software.

2.1.2.project plan

- **Stages of software Lifecycle**
- **Software Requirements Analysis**

This is the first stage of the project, which involves interaction with the customer to understand his/her needs, requirements, information, required functions, performance and interfacing in MLM software. For this purpose requirements analyst will arrange a meeting for gathering information and additional details for software development. After completing requirements can be computerized. The requirement is documented in the form of a software requirements specification(SRS) which is then presented to the customer for review.

- **Design**

Beginning once software requirements have been analyzed and specified software design is the first to three technical activities design, code generation, and test – that are required to build and verify the software.

Design is multi level process which defines following details:

- Data design
- Architecture design
- Interface design
- Component level design

- **Development**

The design must be translated into a machine-readable form. The coding step performs this task. In this stage, the developers will actually code the programs. The specifications arrived at the design stage for each and every function will be converted to code using tools that are finalized for the implementation of the software. At this stage the testing methodology to be adopted will be finalized. For each program test cases will be prepared and for each of these test cases. Test data will also be prepared. The actual

developers will do a first cur checking at this stage to see that the programs written by then are error free.

- **Testing**

In this stages the test group of the development team. using the cases and the test data already prepared will test the programs. Only after all the functions are tested singularly, an integrated testing will be performed to see that inter-function dependability is satisfied. Separate test cases and test data will be worked out for the integrated testing.

- **Accept Test**

This round of testing will be performed by the test group formed by the users of MLM software. This test group has to insure that the developed software is working as per their requirements. If some problems are found then is should be immediately communicated development group so that the problem can be looked into and hence rectified.

- **Data Creation**

For software, data is most important part. Data is information which is handled by software. So before coding software all master table data will have to be created.

- **Implementation**

Now the implementation of software is be done by programmers. All the requirements and information gathered by the analyst is now take actual image in form software. After making software it is uploaded in to the system so users, for whom developed, can use the software.

Once we examine that the project is feasible, we undertake project planning. The table below describes how we planned our project

2.1.3.schedule representation

#	Phases	Time period
1	Project summary	
2	Project plan	20/09- 25/09/2021
3	Risk Analysis	26/09/- 30/09/2021
4	Effort Estimation	01/10/- 05/10/2021
5	System Requirements study	06/10/- 15/10/2021
6	Feasibility Study	16/10/- 20/10/2021
7	Data and Function modeling	21/10/- 31/10/2021
8	Testing	21/12/- 31/12/2021

2.2Risk Management

2.2.1.Risk Identification

During the project plan we have consider all the proactive which we have think we will face during the project period. Here I have listed the risks which we have considered during the project plan:

- ❖ Possibility that the components are not available during the project period.
- ❖ Possibility that products purchased was not compatible.
- ❖ Possibility that the hardware resources are not available during the project period.
- ❖ Possibility that ethernet connection between robot controller and computer may not occur, to which we have to make the connection might not available.
- ❖ Possibility that software inter-compatibility may not be there.

2.2.2. Risk Analysis

Risk analysis is the important aspect of the project planning, whenever planning the software programmer always has to consider the risks of the projects which he might face in the future during designing the software.

Risk are of two types.

- Proactive Risk
- Reactive Risk

Risk Impact

Risk	**Effect**
Possibility of getting illness	Serious

Possibility of component not available	Catastrophic
Possibility of component not effective	Serious
Possibility of hardware resources are not available	Catastrophic
Possibility of robot controller not available	Serious
Possibility of communication between systems might not happen on time	Serious

Proactive risk assessment table

This are the proactive risks which we can consider during the project plan period so we can crop up with them easily and we can find the solution easily but we can find other proactive risks which we haven't considered in the project plan;

Risk	**Effect**
Possibility that because of some problem we have lost the backup of some days.	Serious
Possibility that because of finance problem in the company we won't get the fill resources	Catastrophic
Possibilities that project manager leave the project.	Serious

Reactive Risk Assessment Table

Proactive risks are most dangerous risk which we haven't considered during the project period and to cop up with them is not easy. These kinds of risks are risks, so programmer must have been active to cop up with them smartly.

Risk Estimation

Risk Estimation attempts to rate each risk in two ways;

- The likelihood or probability that risk is real and.
- The consequences of the problems associated which the risk should it occurs.
- The project planner along with other managers and technical staff, performs four risk projection steps.

  - ❖ Establish a scale rate that reflects the perceived likelihood of a risk.
  - ❖ Delineate the consequences of the risk.
  - ❖ Estimate the impact of the risk on the project and the product,
  - ❖ Note the overall accuracy of the risk projection so that there will be no misunderstanding.

The internet of these steps are to consider risks in a manner that leads to prioritization. No software team has resources to address every possible risk with the same degree of rigor. By prioritizing risks, the team can allocate resources where they will have the most impact.

2.2.3 Risk Planning

After estimating all the risks and risk effects, we will look how to manage this kind of risks;

With the kind of proactive risks we will prepare the plan for how to manage these risks, we will think about their options. If their kinds of risks

are facing and if they are reactive risks we have to manage them on the spot to avoid their dangerous effects:

Risk	Risk Management
Possibility of getting illness	we will provide some more time (approx, one week) to prepare the software
Possibility of component not available	We will check for other components, which have the same effects as those components
Possibility of component not effective	We will manage for the new components of that kind
Possibility of hardware resources are not available	We will manage for optional hardware while starting the project
Possibility of robot controller not available	We try to arrange a new controller

2.3. Estimation

2.3.1 Effort Estimation

Effort estimation methods are one of the important tools for project managers in controlling human resources of ongoing of future software projects, the estimations require historical project data including process and product metrics that characterize past projects.

Software cost and effort estimation will never be an exact science. Too many variables human, technical, environment, political, can affect the ultimate cost of software and effort applied to develop it. However software project estimation can be transformed from a black art to a series of systematic steps that provide estimate with acceptable risk. To achieve reliable cost and effort estimates, a number of options arise:

- **Software sizing**
  - Function point sizing

- Standard component sizing

❖ **Problem Based estimation**

LOC and FP data are used in two ways during software project Estimation:-

- As an estimation variable to size each element of the software and
- As baseline matrices collected from past projects and used in conjunction which estimation variables to develop cost and effort projections.

❖ **Schedules**

Obtain an early view of staffing requirements and contains and demonstrate the impact of changing deadlines, understaffing, and staff loading.

❖ **Quality**

Quantity the impact on defect rates of building to deadlines or reducing staff.

❖ **Risk**

Fine tune risk levels for all the major types of risk size, requirements, technology, maintenance, systems integration, and defects

3.0. System Requirements Study

3.1. user characteristics

There are 4 types of user dealing with the system.

User A? Administrator

Administrator: admin is having all the rights on the application.

User B? Employee

Employee: Employee of the company is one of the 4 users of this project

User C? Anonymous user

Anonymous user : anyway who visits website and any person applying for the posted job on the website.

User D? Client

Client: this is the registered user, who come to know about his/her project progress

3.2.Hardware and Software Requirement:

Hardware specification:

Processor :intel Dual based system

Processor Speed:1GHz to 2GHz

RAM :256MB to 512MB

Hard Disk :4GB to 30GB

Keyboard :104Keys

Software specification

Language :JDK 1.6

Database :Oracle 9i

Operating System:Windows NT/XP/Vista

RAM :512MB

3.3.Constraints:

General Constraints

1) This system will not take care of any virus problem that might occur on the computer with which it is installed. Avoiding the use of

pirated/illegal software and ensuring that floppies and other removable media are scanned for virus before use could minimize the possibility of viral infection.

2) Recovery of data after a system crash will be possible only if backups are taken at regular intervals.
3) Manual interfaces cannot be fully avoided. Documented proofs like dates etc, will have to verified by the concerned staff before entering it into the computerized system.

Hardware constraints

The performance of the system will be dependent on the machine conditions. The primary memory (RAM) and the secondary memory (Hard Disk Space) requirement of the system will be the same as that required by normal application and the operating system. And the space required storing the data. The space required to store the data would increase as more and more records are added to system.

Assumption and Dependencies

a. it is assumed that the user is familiar with the basic computer fundamentals.
b. Timely backup of data should be taken to avoid data loss in case of case of system crash.
c. Floppies and other removable media should be scanned for virus before use.
d. It is assumed that the maintenance of the database will be assigned to the authorized person only.
e. Only authorized persons will be allowed inside the system

4.0 System analysis

4.1 Study of current system

The OBS Administration falls short of controlling the employee's activities in analysing his/her strength and weakness .The decision for appraisal of assigns next project to the employee or to train him/her to chance the skills – where lies with proper projection. He is not provided with the detailed project information done or to be assigned based on Application / Verticals.

4.2 Problem and Weakness of current system

- Need of exam manual effort.
- It used to take much time to find any employee.
- Not very much accurate.
- Danger based losing the files in some cases.

4.3 Requirements of New system

Decision in assigning proper skilful hands for the project is an important issue in OBS Administrator should reports with the personal holding the necessary skills required for the project assignment. The decision in marketing analysis about the employee's skills in the prime important before booting in. The propose system of OBS module is the right software to be incorporated in to the automation of OBS software of helping the organization need with respect to skill human recourses.

The proposes system provides details general information about the employee along with educational, certification, skills and project details.

It enhances the OBS management in adding. Viewing and updating employees 'details and generated various report reading employee's skills and experience. Suggestion and Grievances posted by the employee are upheld of talking care of the necessary step in forwarding company's obligation.

Advantages of proposed system

- Very fast and accurate
- No need of any extra manual effort
- No fever of data loss
- Just need a little knowledge to operate the system
- Doesn't require any extra hardware device.
- At last very easy to find the employee.

4.4 FEASIBILITY STUDY:

Once the problem is clearly understood, the next step is to conduct feasibility study. Which is high – level capsule version of the entered system and design process. The objective is to determine whether or not the proposed system is feasible. The tOBSeeTests of feasibility have been carried out.

- Technical feasibility
- Economic feasibility
- Operational feasibility

TECHNICAL FEASIBILITY

In technical feasibility study, one has test whether the proposed system can be developed using existing technology or not. It is planned to implement a proposed system from java technology. It is evaded that the necessary hardware and software are available for implementation of the proposed system. Hence, the solution is technically feasible.

ECONOMICAL FEASIBILITY

As part of this, the costs and benefits associated with the proposed system compared and the project is economically feasible on the if tangible or intangible benefits outweigh costs. The system developments cost will be significant. So the proposed system is technically feasible.

It is a standard that ensure interoperability without stiffing competition and innovation among users, to the benefit of the public both in terms of cost and service quality. The proposed system is accountable to users. So the proposed system is operationally feasible.

4.5 Requirements Validation

Requirements validation examines the specification to ensure that all system requirements have been started unambiguously those inconstaninins error have been detected and the work products conform to the standard.

There are many requirements from user perspective and taken care while designing a system, are as follows:

- Dynamic nature of system i.e. System changes its working depending on situation.
- Component based definition of system. i.e. System Is divided in to smaller components which will work independently also there combined effort is results in to output of system
- Feasibility of system. i.e. System should work with grate ease with different type of documents.
- Flexible font – end design so that it can support functionality if all type of input
- Back – end should not a affect front – end or vice versa
- All database change should be done by font – end only
- Simplicity should be there in system design

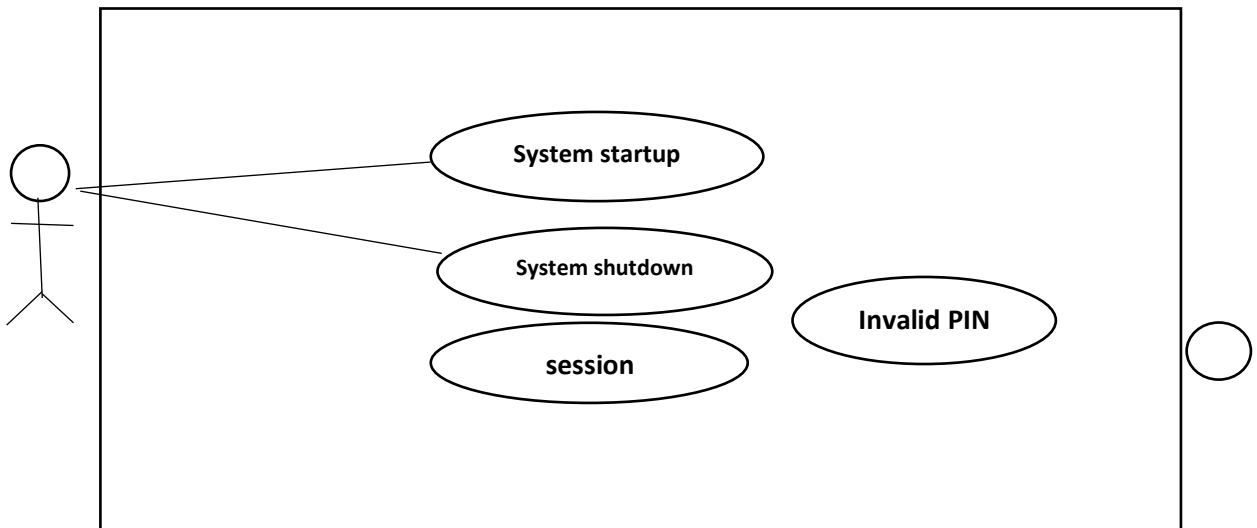
- User friendliness should be archive.
- System should be easily maintainable and adaptive.
- Design for such system is created in such a way that related information is kept in same tables
- Different information related to different component is stored in different tables
- To make the task of data entry essay virus combo boxes and list databases design should support the accommodation of new component information in way, such that it part of system.boxess are design so that user can have to just select the values from the given options.

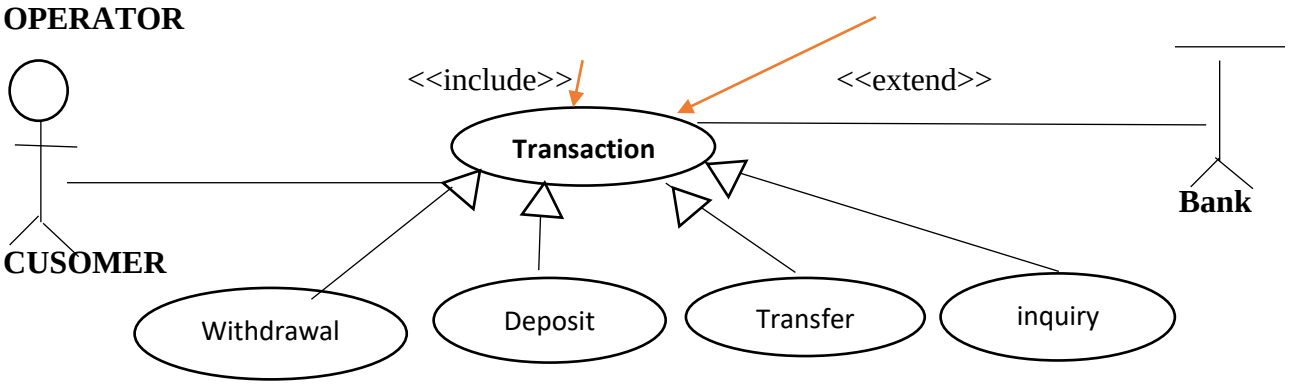
4.6 Foundation of system

Use Case:

Use case is a description of the set of sequins of actions that a system perform that yields an observable results of values to a particular things in a model.

ATM SYSTEM



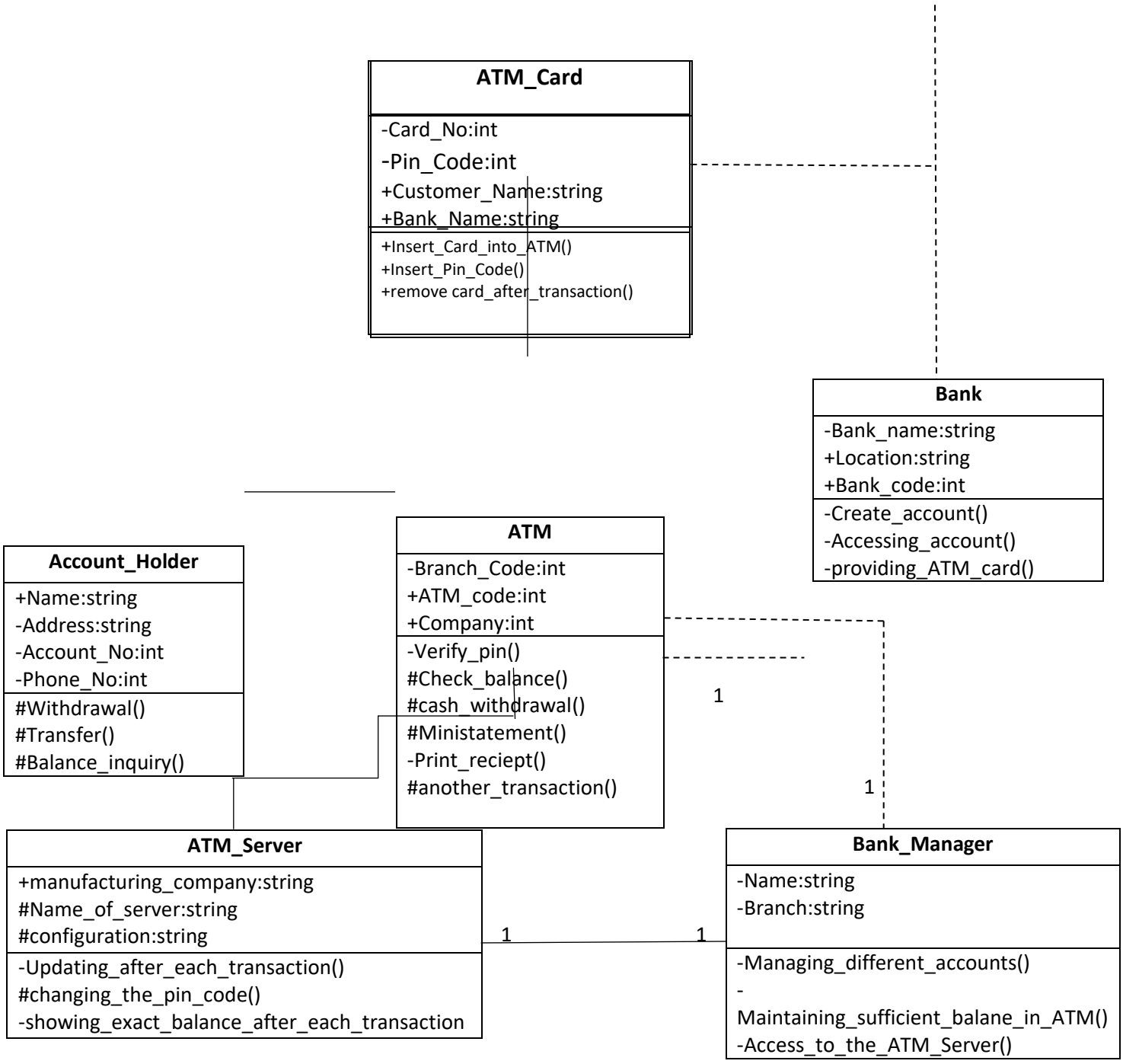


4.7 Data modelling

4.7.1 Class diagram / E-R Diagrams:

Class diagrams are the most common diagrams found in the molding object-oriented systems. A class diagram shows a set of classes,Interfaces, and collaboration and their relationships. Graphically, a class diagram is a collection of vertices and areas.

Class Diagram


ATM_Card
-Card_No:int
-Pin_Code:int
+Customer_Name:string
+Bank_Name:string
+Insert_Card_into_ATM()
+Insert_Pin_Code()
+remove card_after_transaction()
Bank
-Bank_name:string
+Location:string
+Bank_code:int
-Create_account()
-Accessing_account()
-providing_ATM_card()
Account_Holder
+Name:string
-Address:string
-Account_No:int
-Phone_No:int
#Withdrawal()
#Transfer()
#Balance_inquiry()
ATM
-Branch_Code:int
+ATM_code:int
+Company:int
-Verify_pin()
#Check_balance()
#cash_withdrawal()
#Ministatement()
-Print_reciept()
#another_transaction()
1
1
ATM_Server
+manufacturing_company:string
#Name_of_server:string
#configuration:string
-Updating_after_each_transaction()
#changing_the_pin_code()
-showing_exact_balance_after_each_transaction
1
1
Bank_Manager
-Name:string
-Branch:string
-Managing_different_accounts()
-
Maintaining_sufficient_balane_in_ATM()
-Access_to_the_ATM_Server()

❖ Class diagram is a graph that represents the relationship between the classes and represents their semantics.
  - Here ATM works as main class. All other clasess are related with thire class. ATM does following operation:
  - Verify_pin()
  - Check_balance()
  - Cash_withdrow()
  - Ministatment()
  - Pront_recept()
  - Another_transaction()

❖ ATM card related with ATM through many to many relationship. It does following operations:
  - Insert_Card_Into_ATM()
  - Insert_Pin_Code()
  - Remove_Card_After_Transacition()

❖ Account holder related with ATM through many to many relationship. It perform following operations:
  - Withdrawal()
  - Transfer()

❖ ATM server related with ATM by one or many to one or many relationship. It also performs some task shown as below:

-updating_afeter_each_transaction()
-changing_the_pin_code()

_showing_exact_balance_after_each transaction()

- Bank Manager associated with ATM through one or many to one relationship. ATM is not related with this class, but depended on this class but depended on this class. So there is a dependent relationship assigned to them. Bank manager does following tasks:

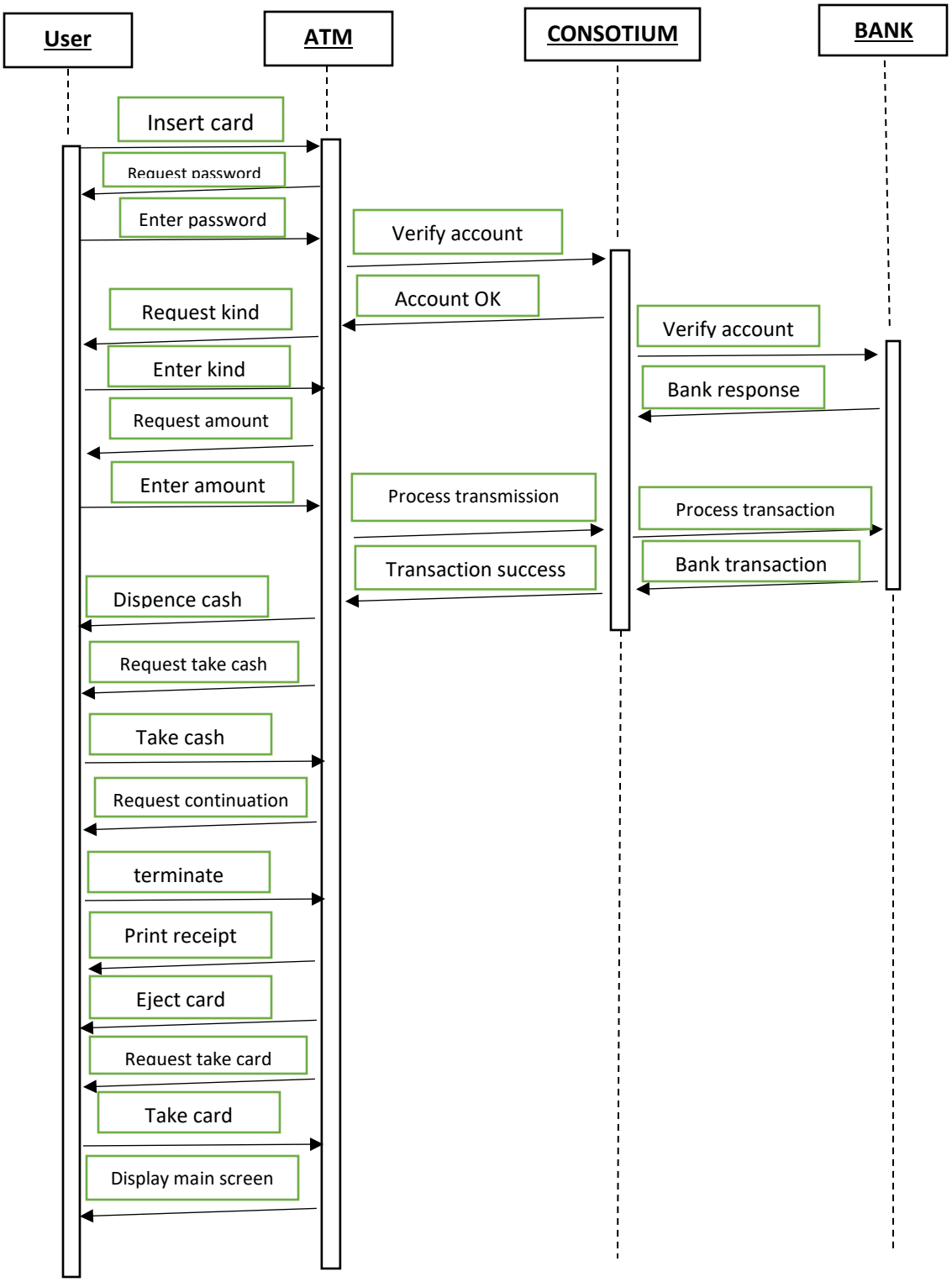
-managing_different_accoounts()
-maintaining_sufficient_balance_in_ATM()
-Access_to_the_ATM_server()

- Here a class Bank not related to any of the class, but some class are dependent on these classes which are shown as dependent relationships with it. Normally this class performs following operations:

-Crate_account()
-Accessing_account()
-providing_ATM_card()

4.7.2 System Activity or Object interaction Diagram:

An interaction diagram shows an interaction, consisting,of a set of objects and their relationships, including the messages that maybe dispatched among them. Interaction diagrams are used for modeling the dynamic aspects of the system.


User
ATM
CONSOTIUM
BANK
Insert card
Request password
Enter password
Verify account
Account OK
Request kind
Verify account
Enter kind
Bank response
Request amount
Enter amount
Process transmission
Process transaction
Transaction success
Bank transaction
Dispence cash
Request take cash
Take cash
Request continuation
terminate
Print receipt
Eject card
Request take card
Take card
Display main screen

4.7.3 Data Dictionary

If anybody wants to know how many characters in a data items, by what other names if it referents in the system, or where it is used in the system, they should be able the find the answer is the property developed data dictionary.

The data dictionary is developed during data flow analysis and assists the system development in determining the user requirements.

Importance of Data Dictionary

Analysis use data dictionary for five important reasons:

1. To manage the detail in large system.
2. To communicate a common meaning for all system elements.
3. To documents the feature of the system.
4. To facilities analyst of the details in order to evaluate characteristic and determine where system changes should be made.
5. To locate error and omissions in the system.

The following tables are used within the system;

Branch _Code	Integer	Code of particular Branch
ATM _ Code	Integer	Code of particular ATM
Company	String	Name of Manufacture of ATM

2. ATM _ Card

It provides information about ATM card, its holder and bank.

ATM _ Card		
Card _ No	Integer	No.particular card
PIN _ Code	Integer	Secrete code of card
Costumer _ Name	String	Name of Card Holder
Bank _ Name	String	Name of Bank

3. Account _ Holder

Account _ Holder		
Name	String	Name of Account owner
Address	String	Address of Account owner
Account _ No	Integer	Account no. of card holder
Phone _ No	Integer	Phone no. of Account owner

4. ATM _ Server

It has detail of ATM.As well as it provides interfaces between Bank and Account holder.

Manufacture _ Company	String	Name of manufacture company
Name _ Of _ Server	String	Server name to identify it
Configuration	String	Configuration set by Bank manager

5. Bank _Manager

It contains data of managers who set and manage the server of ATM.

Name	String	Name of Manager
Branch	String	Branch Name Where Manager do work

6. Bank

It Provide details of bank to get information about Account of a user.

Bank _ Name	String	Name of Bank
Location	String	Location of Bank
Bank _ Code	Integer	Code of bank to identify it

4.8 Functions and behavioral modeling

4.8.1 Context diagram

The top- level diagram is often called a "context diagram". It contains a single process, but it plays a very important role in studying current system. The context diagram defines the system that will be studied in the sense that it determines the boundaries. Anything that is not inside the process identified in the context diagram will not be part of this system

4.8.2 Data flow diagram (0 and 1 level)

A graphical tool used to describe and analyze the moment of the data toBSough a system manual or automated including the process store of data, and delays in the system. Data flow diagrams are the central tool and the basis from witch others component are developed. The

transformation of data from input to output toBSsougth process may be described logically and independently of the physical component associate with the system.

A graphical tool used to describe and analyze the moment of data through a system manual or automated including the process, Store of data, and dealys in the system.

Type of data flow diagram

Physical level DFD
Logical Level DFD

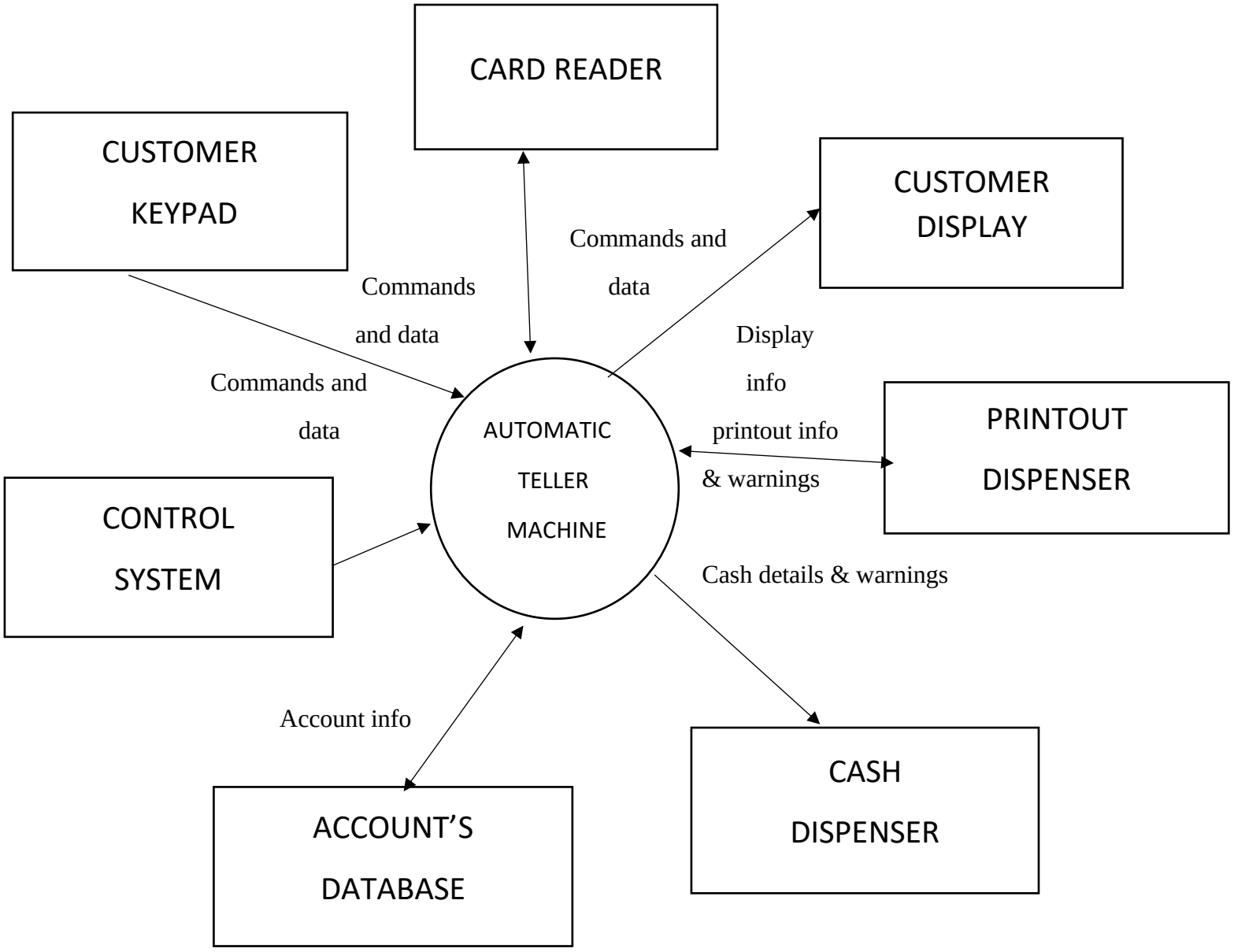
Physical level DFD

Structured analysis states that the current system is used ensure that the current system has been clearly understood physical DFDs shows actual devises, Department, And people ETC... Invoiced in the current system.

Logical Level DFD

Logical DFDs are the module of the proposed systems. They clearly should show the requirement witch the new system should be build. Leather during design activities this taken as the basics for drawing the systems structure charts.

Data Flow Diagram (Level -0)



This diagram shows the Automatic Teller System Software and the hardware that it interacts with. The arrows shows the direction of data following between the software and each hardware element.

Extensional entities

Control System

This system enabled and disables the customer interface and revivers customer request and system reports. A suitable control system would be personal computer linked to a central computer system with access to the accounts data based. The customer interface (keypad, Display etc…) is controlled by enable ad disable the card reader. Which is the customer entry - point to the system. Request for statements and toques book are posted to the control system. It also receives status reports follow printer – peeper and cache levels.

Account Data Based

This is a data base containing account numbers, balancers and other account information. Data is retrieved from the database when a customer requests a balance reports or a cache withdrawal. The database is updated after withdrawal.

Card reader

The card reader reserved the customer's cards and retrieves the PIN and account number stored on it. This information is transmitted to the software system witch unlabeled the customer keypad and national the PIN verification procedure when business is completed the card reader is instructed to return the card. If the customer enters and incorrect PIN, a fixed number of retires permitted, after which the card reader is instructed to confiscate the card.

Customer Keypad

The customer keypad allows a customer to enter a PIN number. Select options and enter cash values. The keypad is only enable even a card is detected in the card reader.

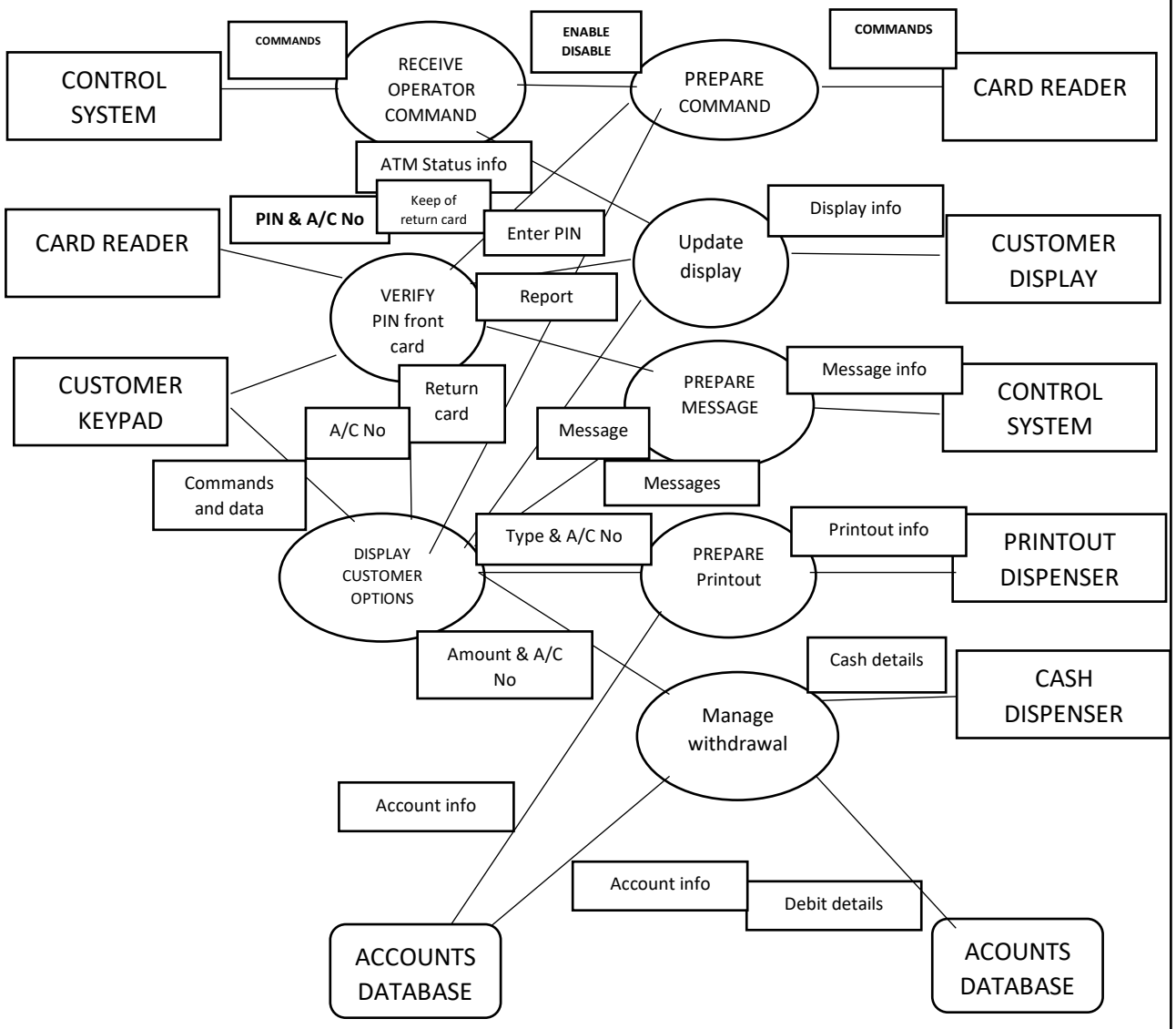
Customer Display

The customer display present massages, options and reposts to the customer. The display is active at all times.

Printout dispenser

This provides the costumer with a printer balance or recites. The printout dispenser report to the system if the paper level is low.

DATA FLOW DIAGRAM



This diagram shows data entry and leaving the system input data is received from the hardware elements of the left. Various type of data are processed by different part of the software system output is sent to the element of hardware on the right.

DFD Level 1 Process:

Interact With Operator

Is process deals with commands from the system Operator. These are the commands which enable or disable the customer interface by controlling the Card Reader. The operator may issue these commands from the another computer system or by using a switch on a control panel.

Interact With Customer

This process handles all interactions with the customers and operates only when a card is detected in the Card Reader. Input is receive initially from the Card Reader and then directly from the customer via the Customer Keypad. The customers receives output from the Customer Display,the printout Dispenser and Cash Dispenser . Customer interactions may also involve sending reports to the control system. The initial step of all customer interactions is to verify the customer's Pin number. After this menu of options is presented on the display which the customer selects by the pressing appropriate keys on the keypad these options lead to other displays and requests for further input. Some options require account details which are retrieved from the Accounts Database and may also involve updating the database. during the final stage of all customer interactions the Card Reader is instructed to the either return or confiscate the card.

Prepare Command

This process handles communication with Card Reader hardwar. The system requires that the card reader is able to receive the following commands:

ENABLE : Make the Card Reader ready to receive a card.

DISABLE : Prevents the Card Reader from accepting a card.

RETURN : Ejects a card from a Card Reader.

RETAIN : Confiscates an unauthorised card.

This Car Reader is enabled and disabled by a commands from system Operator. A card is returned or retained in response to interactions with the customer.

Update Display

This process deal with customer display screen when no card is in the card reader. The customer display shows General information such as "insert card" when a card is detected. The display is updating in the response to customer interactions. If the system is disabled by the system operator, the display is updated to indicate the system status.

The following is a list of screen which are shown on the customer display.

General information - Insert card and other messages.

PIN Verification - Enter PIN message.

Main Options - Some or all of the following ; Display Balance , Print balance, Cash with receipt, Cash without receipt, Order statement , Order chequebook, Return card.

Current Balance - Customer's account balance and cleared funds.

Balance Printed - Take your Balance message

Withdrawal Options - pre-defined cash amounts and other

amount.

Cash Amount - Enter Amount for cash withdrawal.

Cash Dispensed - Take You Cash Message.

Receipt Printed - Take Your Receipt Message.

Statement Ordered - Statement Ordered Message.

Chequebook Ordered - Chequebook Ordered Message.

Card Returned - Take Your Card Message.

Card Retained - Card Retained Message (for failed PIN Verification).

Prepare Message

This process prepares and transmits messages to the control systems. These messages can be requests from customers for statements and cheque books or report concerning the levels of printer - papers and cash.

Prepare Printouts

This process prepares and controls the use of the printout Dispenser to produce balance receipts.The customer's balance is retrieved from the Account Database(if required). if the printer- paper level becomes low option which involve printouts are disabled and a warning message is sent to the control system.

Manage Withdrawal

The process receives requests for withdrawal of specific amount from a certain account and operates the Cash Dispenser. Before proceeding, the customer's detail in the Accounts Database are checked. If the request exceeds the customer's balance (or agreed overdraft) the withdrawal is denied. The system uses a denomination selection algorithm based on the notes available and amount required. The Accounts Database is updated after each withdrawal. If the cash level becomes low, options providing cash withdrawals are disabled and warning message is sent to the control system.

4.8.3 Process Specification and Decision Table

01.Admin Login

If admin_login=true

Then give right to the access the application as well as admin functionalities

Else

Show error message

End if

02.User Login

If user_login=true

Then give right to the access the application

Else

Show error message

End if

03.Entry of details of respective forms to save in database

If Information entered = valid

Then save

Else

Show error message

End if

04.Entry of details to update records I The database

If Information updated = valid

Then do update in database

Else

Show error message

End if

05.Genaral Reports

If parameter for Report = valid

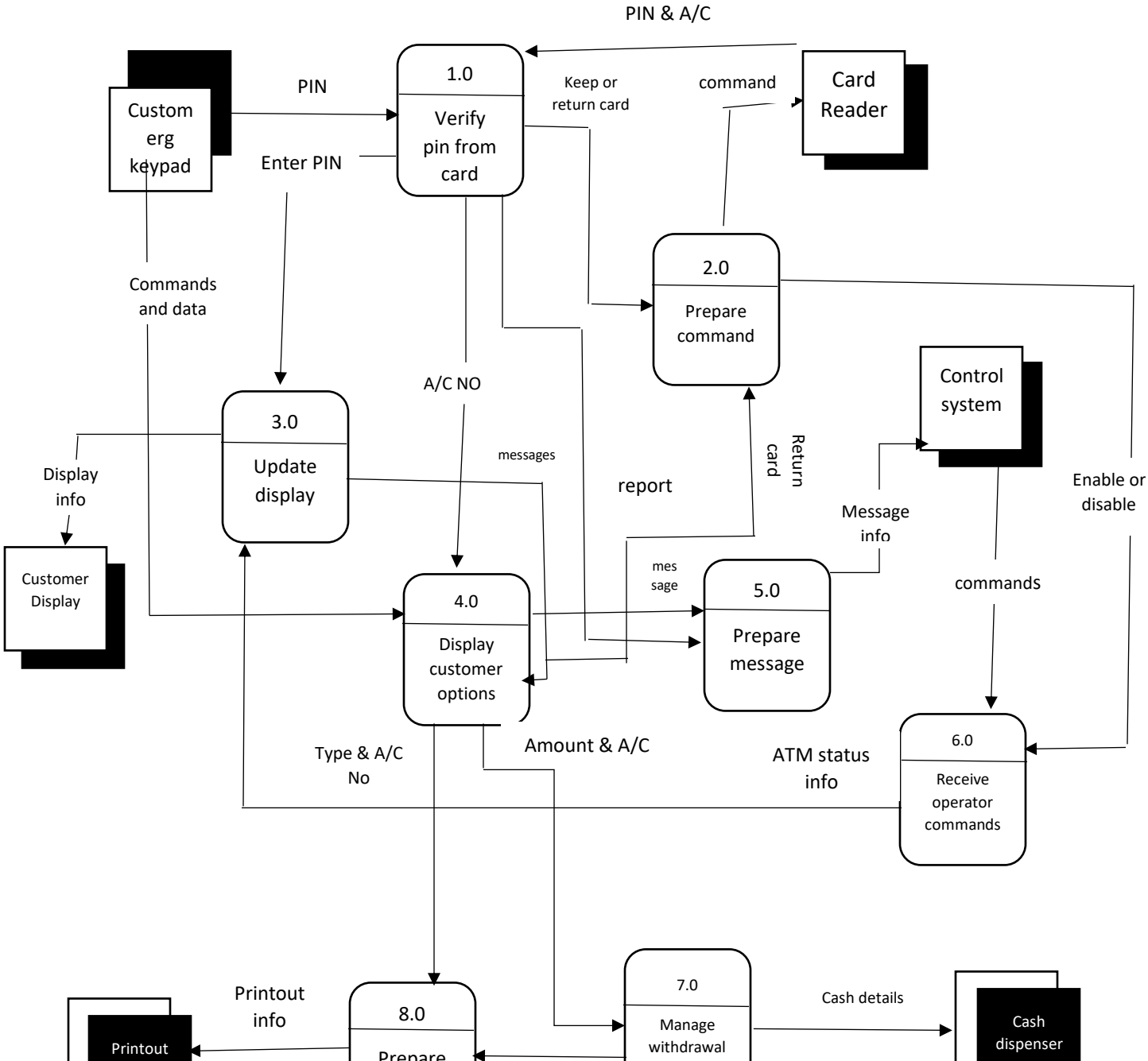
Then view the records

Else

Show error message

End if

4.8.4 Control Flow Diagram



5.0 Testing

5.1 Test plan:

Any system, to be successful, must be thoroughly tested, and well managed test plan should be prepared before actual testing is begin performed." Models" have been developed and need to be tested in a manner that can reduce occurring of defects as low as possible. Following are the activities we plan to test the system.

1. This system is indeed an evolutionary system so every unit of the system is continuously under testing phase.

2. One test activity "Basis Path Testing" that will try to cover all part in system. This activity identifies all part that provide different functionality of the system, and also other part to reach at the functionality.

3. Other testing activity is "Control Structure Testing "which will test each and every condition with positive and negative data combination.

4. This testing activity will also perform "Data Floe Testing" in which it will be tested how the data re following the system. And will also check whether the data entered from one produce, is reflected whenever it requires or not.

5. All conditioners will be tested with "Boundary Value Analysis", where different input will be given to test whether the system is functioning with boundary values or not.

6. Along with boundary value analysis the system is also tested with "Range Value Tested" where editable values will be tested with ranges of values.

7. A system is begging tested in "Unit Testing" manner at the completion of one unit that is tested totally with thoroughly with above mentioned testing activities.

8. The integration testing will also perform to ensure that the integrated unit is working properly with other units or not.

5.2 Testing Strategy

CONTENT TESTING :

Errors in project content can be as trivial app minor topographical error as incorrect information, improper organization of validation of intellectual property laws. Content testing attempt to uncover this and many other problems before the user encounter them.

Content Testing Objectives:

There are three types of objectives.

1) To uncover syntactic errors in text- based documents, graphical and other media.
2) To uncover semantic errors in any content object represented as navigation error.
3) To find errors in organization or structure of content that is presented to the end-user.

INTERFACE TESTING

Interface design model is reviewed to ensure that generic quality criteria established for user interfaces have been achieved and that application specific interface design issue has been properly addressed.

Interface testing strategy:

The overall strategy for interface testing is to

1. uncover error related to specific interface mechanisms
2. Errors in the way interface implement semantic of navigation, WebApp functionality of contents display to accomplish this strategy a number of objective must be achieved.

- Interface future are tested to ensure the design rules, aesthetic and related visual content available for the user without error.

- individual interface mechanisms are tested in manner that is analogous to unit testing for example, tests are designed to exercise all forms, client - side scripting dynamic XML.

Testing Interface Mechanisms

When a user interacts with a system, the interaction occurs through one or interface mechanisms.

Forms -:

At a microscope level, tests are performed to ensure that

1. Labels correctly identified fields within the form and that mandatory fields are identified visually for the user.
2. The server receives all information content within the form and their no data are lost in the transaction between client and the server.
3. Appropriate defaults are used when the user does not select from a pull-down menu or set of buttons.
4. Browsing functions don't corrupt data entered in the form.
5. Script that performs error checking on data entered work.
6. Properly and provide meaningful error message.

Client-side Scripting -:

Black box tests are conducted to uncover any error in processing as the script is executed. These tests are couple with forms testing because script input is often derived from data provides as part of forms processing.

Dynamic XML -:

Here the data transfer between the systems occurs in the form of XML file. The formatting of XML file should remain unaltered. Changes in the internal attribute structure can also lead to an error . Hence for this XML parsing functionalities are provided on the client side.

Application specific interface mechanisms-:

Test conforms to a checklists of functionality and features that are defined by the interface mechanisms.

☆ Boundary test minimum and maximum number of item that can be placed in to shopping cart.

☆ Test determine persistence of image capture contents.

☆ Test to determine whether the system can be record co-ordinate content at some future date.

USABILITY TESTING-:

Usability test may be designed by Project engineering team.

1. Define a set of Usability testing categories and identify goal for each.
2. Design test that will enable each goal to be evaluated.
3. Select participants who will conduct test.
4. Instrument participant's interaction with system while testing is conducted.
5. Develop a mechanism for assessing the Usability of the system.

The following test categories and objectives illustrate establish testing:

Interactivity

Are interaction mechanisms easy to use?

Layout

Are the navigation mechanism, content and function place in manner that allows the user to find them quickly?

Readability

Is the text well written and clear?

Aesthetics

Do layout colour, typeface ,and related characteristics lead to ease of use?

Display Characteristics

Does the system make optimal use of screen size and resolution?

Time Sensitivity

Can important features, functions and content to be use in a timely manner?

Accessibility

Is the system accessible to people who have Disabilities?

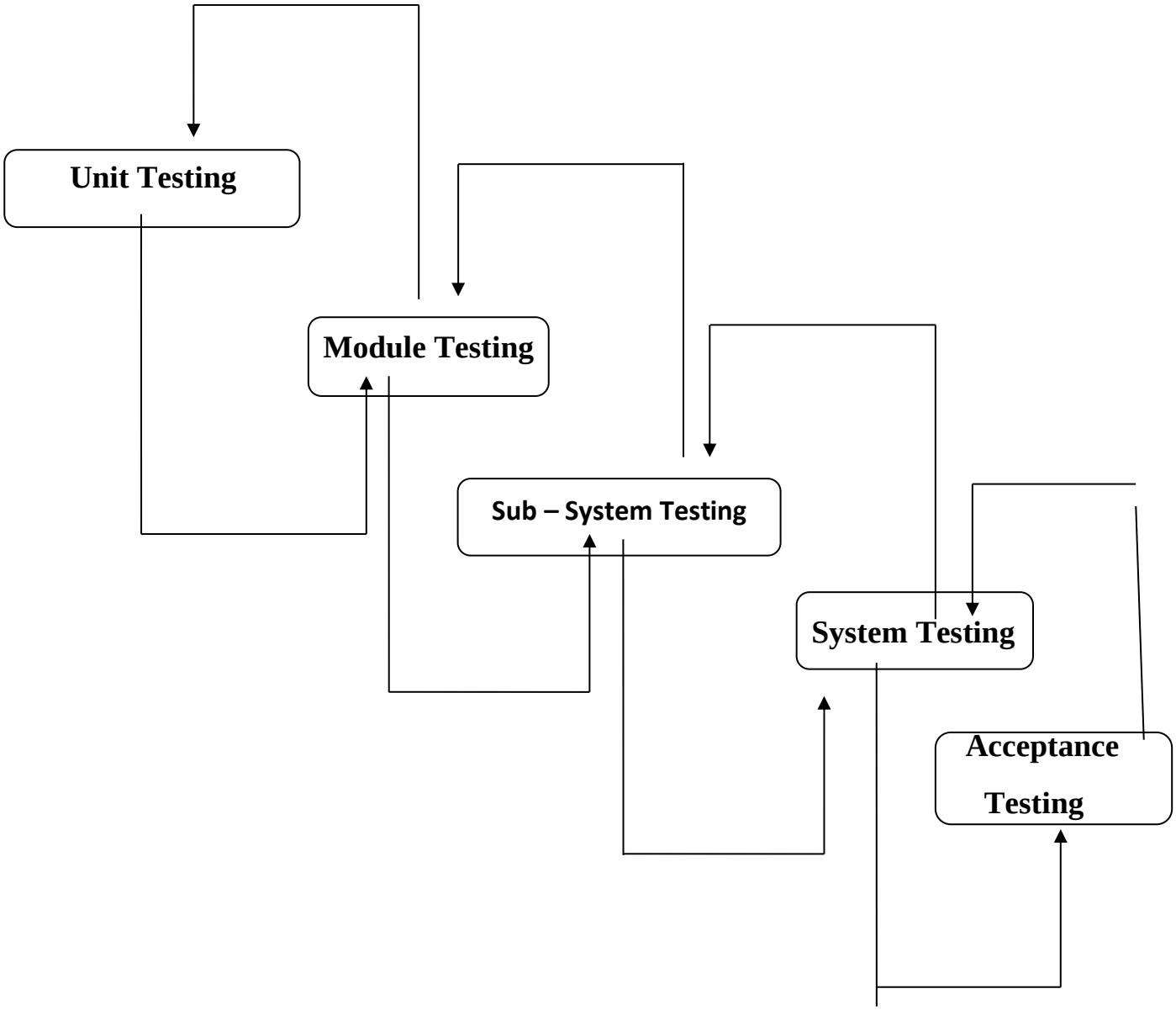
COMPATIBILITY TESTING

Project must operate within environment that differs from one another. Different computer, display device, OS, browser and network connection can have significant on system operation. The project team derives a series of compatibility, Validation tests, derived from existing interfaces tests, navigation tests, performance tests and security tests.

5.3Testing methods

Analysis and check system representation such as the requirement document, design diagrams and the program source code. They may be applied at all stages of the process.

Testing Method



There are different Models of testing. On the basis of testing methods there are two types of testing:

1. White-box testing

2. Black-box testing

1).WHITE-BOX TESTING

White-box testing sometimes called **glass-box testing,** is a test case design method that users the control structure of the procedural design to drive the test case.

■ Logical errors and incorrect assumption are inversely proportional to the probability that a program will be executed. Errors tend to creep into our work we design and implement function, condition or control that is out of the mainstream tends to be well understood.

■ We often believe that a logical path is not likely to be executed when in fact it may be executed on a regular basis. The logical flow of a program times counter intuitive.

2). BLACK-BOX TESTING

For our project periodically we have tested our software using black-boxtesting.Thinking as a client we have evaluated the software for its ea sy going andconvenience.

Unit Testing:

During the programming stages each and every form, modules and class treated unit has been put into the test data. Every module is tested independently. The steps are follows.

1.Manually code is tested like spelling checks, logic and errors.

2.Once the manual checking is over the complication has been done. Syntactical errors if any have to be corrected.

3.After the clean complication the program, some dummy data, as specification, has been used for testing of the module to see if it works as specified.

Integration Testing

After our individual's modules were tested out we go the integrated to create a complete system. This integration process involves building the system and testing the resultant system for problems that arise from component interaction.

Performance Testing

Performance testing is designed to test the runtime performance of the system within the context of the system. These tests were performed as module level as well as system level. Individual modules were tested for required performance.

Condition Testing

Performance testing is a test case design method that exercises the logical conditions.

Interface Testing

Interface sting is integral part of integration. We examined the code to be tested and explicitly list each call to an external component. In the system standards tests for GUIs have been performed, which are as follows:

• The position and related labels for all controls were checked.

• Validations for all inputs were done.

• Pull down controls was verified for proper functionality.

• Whether the non-editable text controls disabling and it was also verified that it doesn’t exceed the maximum allowed length.

5.4 Test Cases:

(1)Invalid Parameter Setting.

Test Case

If the cameras are calibrated in a wrong manner, then the values derived from the images would be incorrect.

Solution

Solution of this case is that we have to check the values from the .CSV file and see to that they are within the permissible range or not.

(2)Light source moves out of the camera plane.

Test Case

If the any user by mistake moves the light source out of the camera plane, the captured image would then not contain any co-ordinate information.

Solution

Solution of this case is that there is a condition kept when no co-ordinates are extracted pass on the value of the origin.

(3)If any of the cameras are not connected.

Test Case

If the cameras are disconnected from the computer the initialization of the process will not occur.

Solution

Solution of this case is that user has to be aware of all the hardware has been correctly connected to the system before initialization. User has to keep a checklist of all the system before initializing the system.

(4)If a proper base is not taught.

Test Case

If the user teaches an invalid base then there remain chances of the robot to collide with the surrounding.

Solution

Solution of this case is that user is not aware about the base, and then the user should execute the robot motion in T1 testing mode.

(5)Communication error between controller and computer.

Test Case

If user is unable to connect the computer to the controller, then user would not be able to pass on the co-ordinate values.

Solution

Solution of this case is that user has to first establish the connectivity by configuring the ApiConfig file, and check the connecting Ethernet cable. Successful connection can be known by using the PING command.

6.0 Limitation and Future Enhancement

Limitation

■ Although I have tried to add all the related features to this online Bus Reservation System but there are also some limitation.

■This system is stand alone system so data saved during different processes are stored in the machine in which that process was executed.

■So there is the problem of distributed database.

Future Enhancement

■As discussed the limitation of this system, we can implement this as client/server system. So all the data will be stored in the single machine, and for any purpose all the data will be retrieved from this central database.

■So there will be no human work require for the employee. There will be only one person required who will maintain this central database.

7.0 Conclusion

Back in 1969, Chemical Bank announced that a new form of banking was being launched. With that, customers were provided with plastic cards designed with a magnetic strip that could be used with a machine built into a wall. Gone were the days of having to stand inline for a teller or not having money on hand after normal banking hours. Almost everyone has heard of and used an ATM machine. Interestingly, some of people feel that ATM machines are the best thing to happen in the banking world while other people consider them a curse. The main complaint heard about ATM machines is that while they are convenient, they are expensive to use.

However, if we look at it from a banking perspective, business is business. Regardless of what we think of ATM machines, there is no doubt that they have changed the world and the way in which we do things. For example, think how many times we have been out somewhere only to discover we have no cash and we are out of checks, ah, but in the corner, there is an ATM machine. In the blink of an eye,

we swipe the card and now have cash on hand. In addition to pulling money out, the ATM machine also makes it convenient to deposit money, transfer money, and check balances. Best of all, to use an ATM machine, we do not have to go to the bank. We will find ATM machines at other banks, grocery stores, shopping malls, along the roadside, Buckingham Palace, airports, in casinos, and even on the South Rim of the Grand Canyon. For this reason, ATM machines are extremely helpful!

Reference

01.www.google.com

02.www.w3schools.com

03.www.youtube.com

THANK YOU