1. **INTRODUCTION**

**1.1 Purpose**

The main purpose that banks have been serving since their inception is keeping our money safe for us. While keeping our money safe, they also let us earn a certain amount of interest on the money deposited with them. Traditional banks have been doing this, and internet banks continue the same function. The only difference is in the way the transactions are made.

We all know about internet banking but few of us actually understand about the history of internet banking and how it all came out. Knowing history of internet banking can be incredibly useful, especially since it will allow us to have more respect for the little things that we take for granted internet banking has been around for quite a few years now, but has really only become prominent over the past year or so in particular. Internet banking offers an array of different advantages to the user, including account balances and history including year-to date information, the ability to transfer money from one account to another and to payees for bill payments, check history, reorders, and stop payments, check credit card balances and statements, complete online loan applications, secure interactive messaging with staff and much more.

Internet banking basically allows you to be able to do everything that you can do it all right from the convenience of your own home.

The aim of this project is to develop a secured online banking system with the following objectives:

* Create a banking system that is easily accessible by customers from the comfort of their homes, offices etc.
* Reduce the flow of human traffic and long queues at banks.
* Reduce the time wasted in going to banks to stay on queues.
* Promote efficient and effective banking for the banks by focusing on those services that still require physical presence at the banking hall.

**1.2 Document Conventions.**

🡪HTTP : Hyper Text Transfer Protocol.

🡪HTML : Hyper Text Markup Language.

🡪TCP/IP: Transmission Control Protocol/Internet Protocol.

**1.3 Intended Audience & Reading Suggestions.**

The different types of readers are:-

(a). Customers.

🡪 Project Scope.

🡪 Security Available.

(b). Employers.

(c). Developers.

🡪 Project Scope.

🡪 Use Case Module.

(d). Project Manager.

🡪 System Features.

🡪 Hardware Requirement.

🡪 Software Requirement.

🡪 Interface Requirement.

**1.4 Definations, Acronyms & Abbreviation.**

🡪Account Details.

🡪ATM Request Form.

🡪New Account Opening Form

Abbreviation Used:-

PAN – Permanent Account Number.

DD - Demand Draft

SB - Saving Bank

CA - Current Account.

ATM – Automatic Teller Machine.

**1.5 Scope.**

The Scope of this project is limited to the activities of the operation units of the banking system which include opening of account, deposit of funds, withdrawal of funds & transfer.

🡪 Any bank can use this application to provide better service to their customers.

🡪 Customers can access his/her all accounts present in various branches of the same bank at one click.   
🡪 Bank can publish various upcoming plans for customers through this application.  
🡪 Manager can access all accounts present in the bank through this application.  
🡪 Reduction in work load of all employees will possible through this application as transaction rights are provided online to customer.  
🡪It can be extended for global communication between all banks in the world.

**1.6 References.**

🡪 References from following website:

[www.bankofbaroda.com](http://www.bankofbaroda.com)

[www.youtube.com](http://www.youtube.com)

🡪 Reference from our text book “Software Engineering” of Genius Publication.

**2. OVERALL DISCRIPTION.**

**2.1 Product Perspective.**

The client will have client interface in which he can interact with the banking system. It is a web based interface which will be the web page of the banking application. Starting a page is displayed asking the type of customer he is whether ordinary or a corporate customer. Then the page is redirected to login page where the user can enter the login details. If the login particulars are valid then the user is taken to a home page where he has the entire transaction list that he can perform with the bank. All the above activities come under the client interface. The administrator will have an administrative interface which is a GUI so that he can view the entire system. He will also have a login page where he can enter the login particulars so that he can perform all his actions. This administrative interface provides different environment such that he can maintain database & provide backups for the information in the database. He can register the users by providing them with username, password & by creating account in the database. He can view the cheque book request & perform action to issue the cheque books to the clients.

**2.2 Product Features.**

The Internet banking system consists of following modules :

1) Login Process

This module allows valid customers to access the functionalities provided by the bank.

2) Balance Enquiry

🡪This module maintains the balance details of a particular account.

3) Update Profile

🡪This module allows the customer to update profile of their account.

4) Funds Transfer

🡪This module allows the customers to transfer funds from one account to another within the same bank.

5) Change of Password

🡪This module allows customers to change their password.

6) Mini Statements

This module allows customers to view their transaction details.

2.2 User Classes & Characteristic

**Customers :** The normal users will have an account of fixed or savings and should have a minimum balance of Rs. 500. He can transfer funds to another account of the same bank & may view his monthly or annual statements.

**Industrialists, Entrepreneur, Organizations academicians:** These users will have all the three accounts & should have a minimum balance of 20,000 Rs. He can view the statements of his organization or industry.

**2.3 Operating Environment.**

**Server Side**

Hard Drive 🡪 More than 160 GB.

RAM 🡪 More than 1 GB.

Processor 🡪 Pentium 4 or Higher.

Client Side

Hard Drive 🡪 More than 160 GB.

RAM 🡪 More than 1 GB.

Processor 🡪 Pentium 4 or Higher.

**2.5 Design & Implementation Constraint.**

•

🡪 This system works only on a single server.

•

🡪 This is designed in ASP.Net.

🡪 Language used is C#.

•

🡪 Limited to HTTP/HTTPS protocols.

**2.6 User Documantation.**

A registered user can have following facilities:

🡪 Accounts and accounts status.

🡪 The balance enquiry.

🡪 The fund transfer standards.

🡪 Cheque Book Request.

🡪 Password Changing.

**2.7 Assumption & Dependencies.**

**Assumption:**

The details of customers such as username, password, account type and their corresponding authority details should be manually entered by the administrator before using this system.

•

Every user should be comfortable of working with computer and net browsing.

He should be aware of the banking system.

He must have basic knowledge of English too.

**3. EXTERNAL INTERFACE REQUIREMENT.**

**3.1 User Interface**

There are four different ways for a user to interact with the system:

**Viewers:**

Many unknown persons or un-authenticated persons visit the Bank official site via internet. They collect the information and search what are the schemes are available in the bank web page. Those viewers or visitors became the customer of the bank.

**New User:**

Who all visited that Bank webpage or heard about the bank those persons getting ready to start account in bank. They register the bank application form, submit and start account in the nearest bank.

**Existing User:**

The Existing user is the most typical user of the Online Banking system. Each Users have their own account and registered or authorized login access. The Existing user can login in online to their account perform the operation of deposit, withdrawn, transfer, balance queries and transactions. All the operation of the banking do in online it helpful for user because save time and efficient process.

**Administrator:**

Admin is master user of the system because they are main role of the system. Admin grant and maintain the database of the existing user and grant the permissions to users. It over rules all other users

**3.2 Hardware Interface**

*Client Side:*

**User on Internet** :  Web Browser, Operating System (any)

**Application Server** : WAS

**Data Base Server** : DB2

**Network** : Internet

**Development Tools** :  ASP.Net, HTML,OS(Windows).

*Server Side:*

Processor : Pentium IV or higher.

RAM : 1 GB or more.

Disk Space : More than 160 GB.

**3.3 Software Interface.**

**User on Internet** :  Web Browser, Operating System (any)

**Application Server** : WAS

**Data Base Server** : DB2

**Network** : Internet

**Development Tools** :  ASP.Net, HTML, OS(Windows),

**3.4 Communication Interface.**

APNA Bank website i:e [www.apnabank.com](http://www.apnabank.com) offers all banking facilities anywhere any time through internet facility. It also offers Mobile Banking facility which is a SMS based service.

🡪 Client on Internet will be using HTTP/HTTPS protocol.

🡪 Client on Intranet will be using TCP/IP protocol

🡪 A Web Browser such as IE 6.0 or equivalent

**4. Other Non-functional Requirement.**

**4.1 Performance Requirement.**

System can withstand even though many no. of customers request the desired service. Access is given to only valid users of bank who requires the services such as balance enquiry, update profile, funds transfer, mini statements, and request for stop payments and for cheque book

It is available during whole week for all 24 hours.

**4.2 Safety Requirement.**

By incorporating a secure database and proven DB2 UDB into the system, reliable performance and integrity of data is ensured. There must be a power backup for server system. Since the product is of 24x7 availability there should be power backup for server which provides the information . Every day the data should be backup even when the operation of an user is not successful i.e., while performing the operation power failure occurs then data should be backup.

**4.3 Security Requirement**

Sensitive data is protected from unwanted access by user’s appropriate technology and implementing strict user-access criteria.

Facility of unique user number and Password in such a way that unauthorized user cannot log in. Operational rights for each user/terminal can be defined. Thus, a user can have access to specific terminals and specific options only.

Online Banking uses the SSL (Secure Socket Layer) protocol for transferring data. SSL is encryption that creates a secure environment for the information being transferred between customer browser and Bank. Online Banking uses 128-bit digital certificate from VeriSign for encryption of the Secure Sockets Layer (SSL) session. SSL is the industry standard for encrypted communication and ensures that customer's interaction with the Bank over the Internet is secure. Secure Socket Layer (SSL) protects data in three key ways:

**Authentication:-** ensures that you are communicating with the correct server. This prevents another computer from impersonating Bank.

**Encryption :-** scrambles transferred data.

**Data integrity :-** verifies that the information sent by customer to Bank wasn't altered during the transfer. The system detects if data was added or deleted after customer sent the message. If any tampering has occurred, the connection is dropped

**4.4 Software Quality Attributes.**

**4.4.1. Usability.**

The users of the system are members and the administrators who maintain the system. The members are assumed to have basic knowledge of the computers and Internet browsing. The administrators of the system to have more knowledge of the internals of the system and is able to rectify the small problems that may arise due to disk crashes, power failures and other catastrophes to maintain the system. The proper  user interface, user’s manual, online  help and

the guide to use and maintain the system must be sufficient to educate the users on how to use the system without any problems.

**4.4.2 Reliability.**

The system is safety critical. If it moves out of normal operation mode, the requirement to drop to the next lower floor and open its doors is given priority. This emergency behaviour shall not occur without reason.

The system has to be very reliable due to the importance of data and the damages incorrect or incomplete data can do.

**4.4.3 Availability.**

When in normal operating conditions, request by a user for an servicer shall be handled within 1second. Immediate feedback of the systems activities shall be communicated to the user by link  page clicked. At peek system load, individual users at either the server in the security office, at the links or inside the banking system shall not experience any delay in the service response to their commands longer than 1 second.

The system is available 100% for the user and is used 24 hrs. A day and 365 days a year. The system shall be operational 24 hours a day and 7 days a week.

**4.4.4 Security.**

There shall be no security mechanisms in place to keep unwanted users out of the system. However, all users of the system shall not be able to perform actions or request actions from the Banking system, which will cause harm to any person or damage to the system or its environment.

**4.4.5 Maintainability.**

There shall be design documents describing the internal works of the software. There shall bean access on the control panel and servers for the purpose of upgrading the software or flashing any firmware.

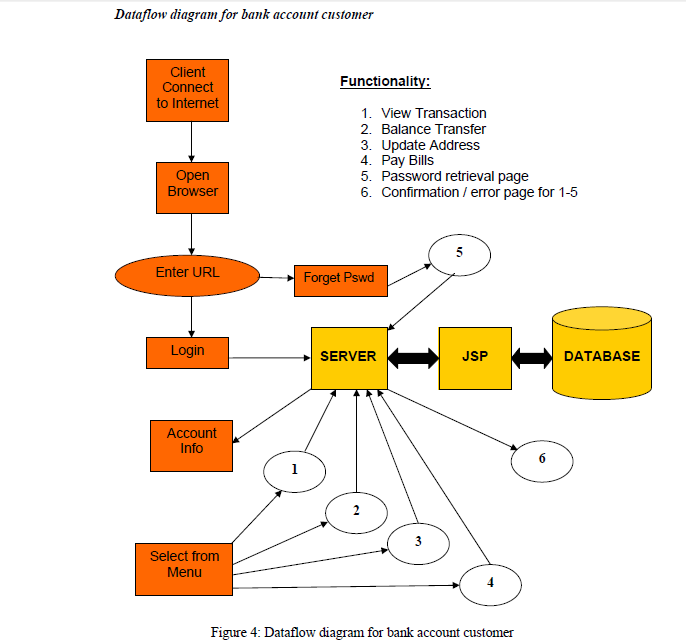
**4.4.6 Portability.**

There are no portability requirements. Requirement Organization: All requirements shall be organized according to object. First general requirements for all service types shall be described. Following are sections for each service type and their special requirements. Last are requirements related to other objects like the users view pages and any other.

**5. System Design.**

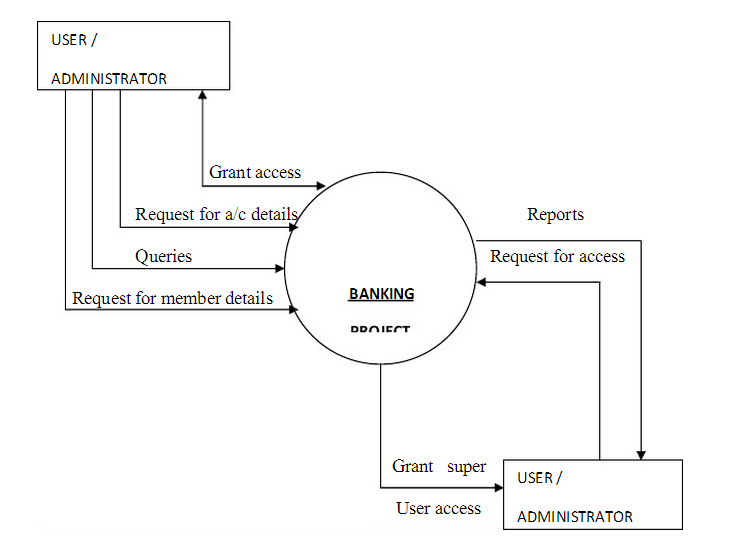
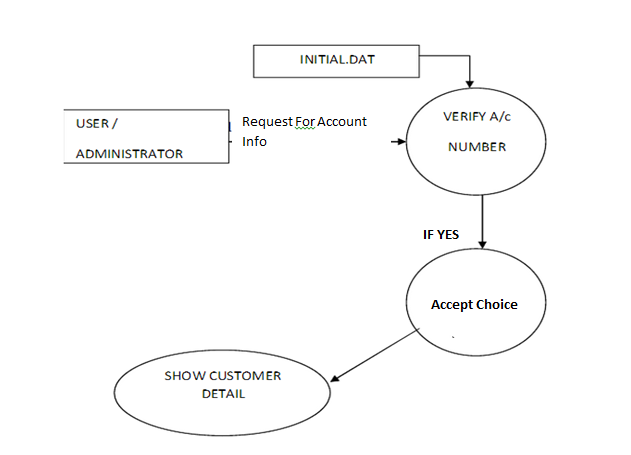
**5.1 Process Description(Data Flow Diagram).**

A **data flow diagram** (**DFD**) is a graphical representation of the "flow" of data through an information system modeling its *process* aspects. Often they are a preliminary step used to create an overview of the system which can later be elaborated.



ASP

“COMMON DFD OF BANK SYSTEM”



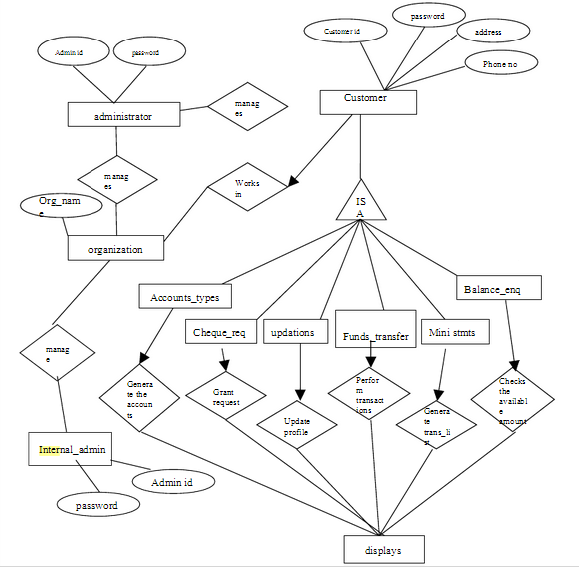
BANK SYSTEM

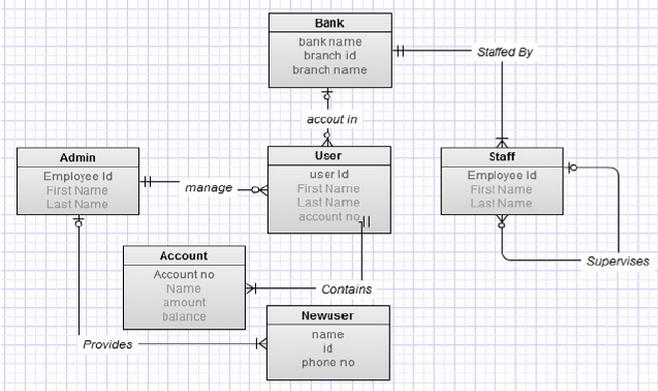
**“ZERO LEVEL DFD”**

**“1ST LEVEL DFD”**

**5.2 Relation Between Entities(Entity Relationship Diagram).**

An entity-relationship diagram is a data modelling technique that creates a graphical representation of the entities, and the relationships between entities, within an information system.

****



**“Entity Relationship Diagram”**

**6. SNAPSHOTS**

**6.1 HOME :-**

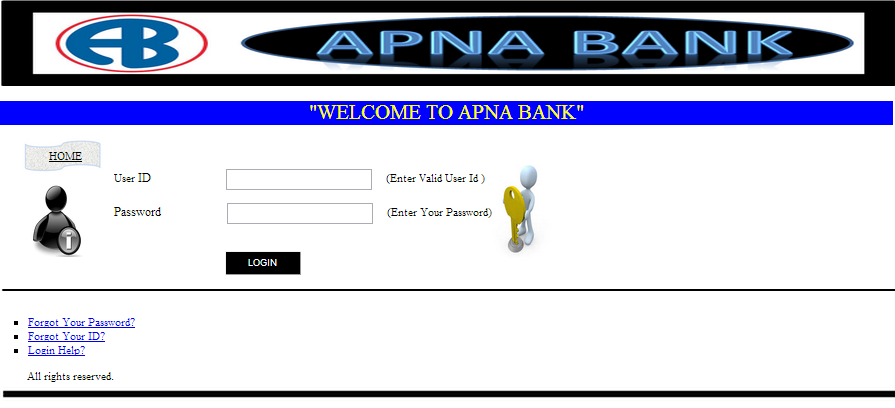
This page tell u about the bank. It tells about the various facilities available on this bank website so that user can choose his desired requirement & can go through it for results.



**6.2 LOGIN**

This page provides a form to login to the Internet banking website.

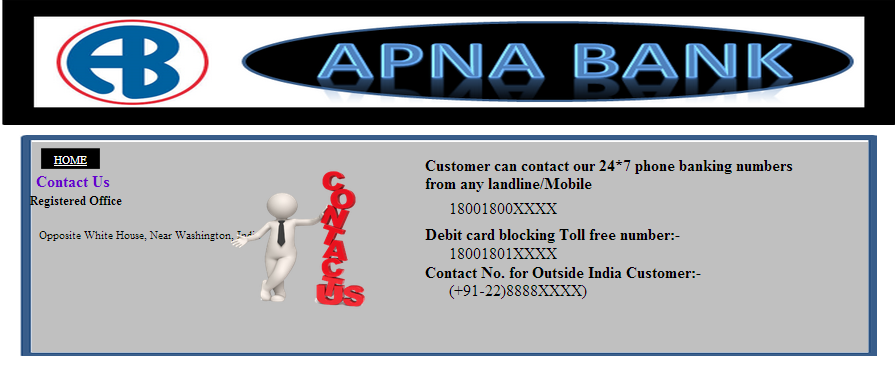
A registered user can login with a valid User ID and password.



**6.3 CONTACT US:**

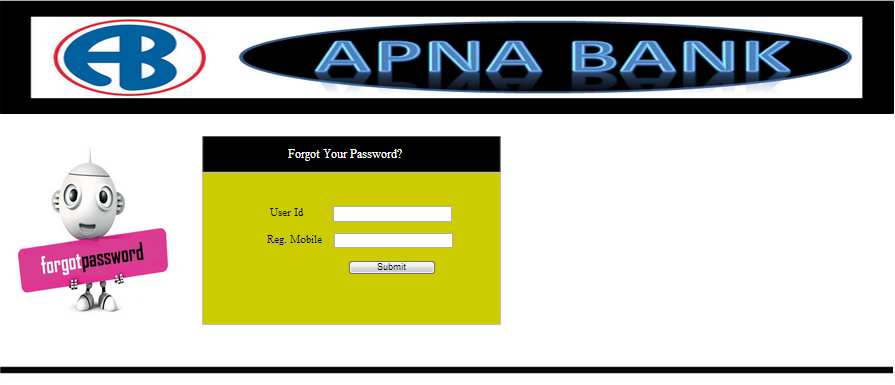
This page let u know about the contact details of the bank.

It let u know about the bank address & also about the banking customer care numbers.

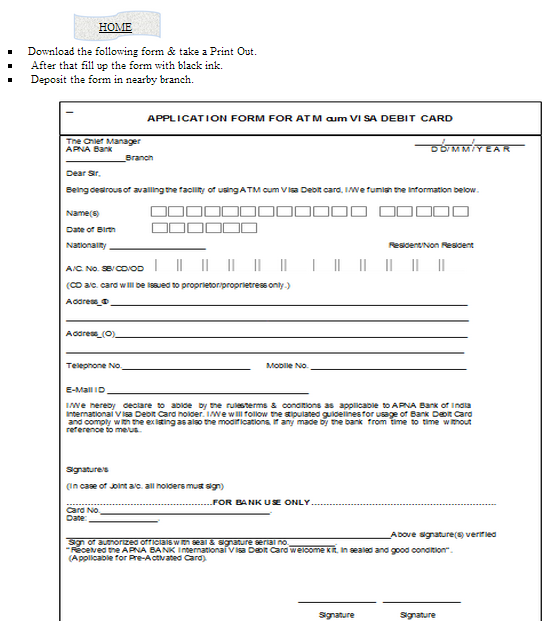


**6.4 FORGOT PASSWORD :**

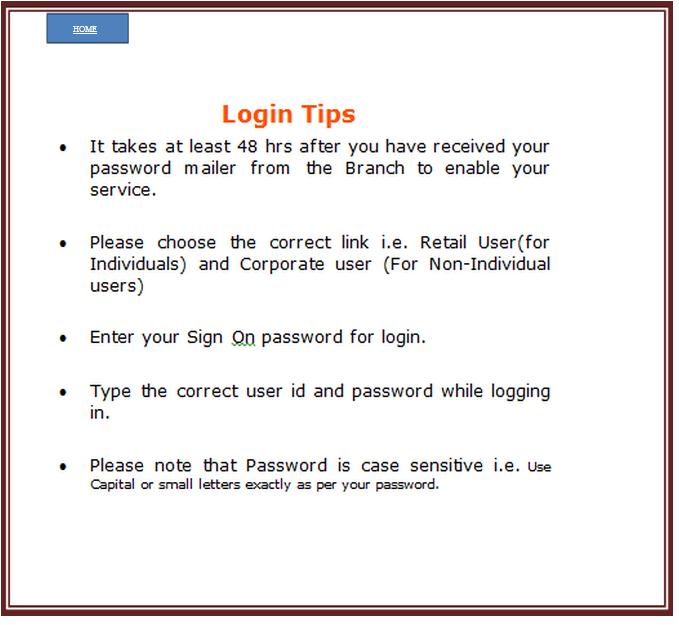
If you have forgot the password of our net banking account, then this page will help you to recover our password using your registered mobile number.



6.5 ATM application page:

If you want your own atm card then this page is required to be filled.

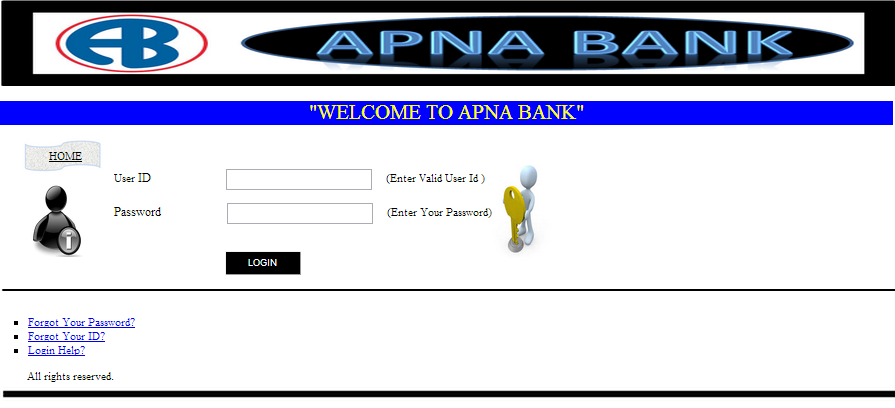
**6.6 Help Page:**

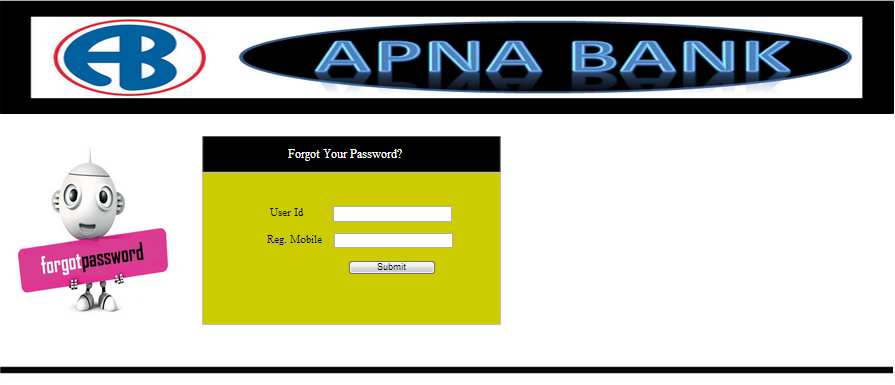
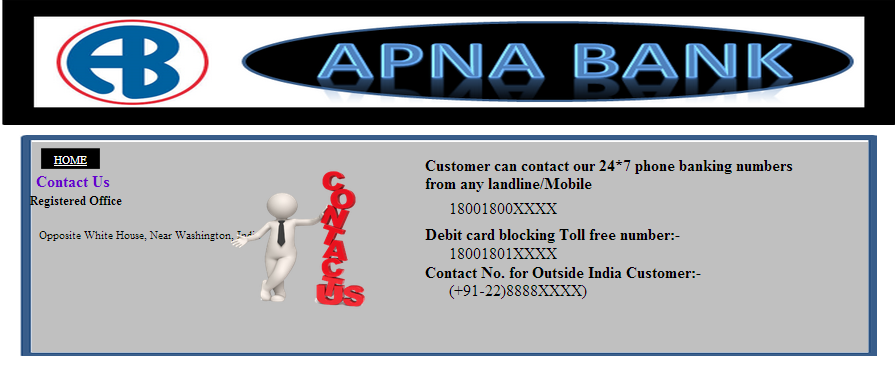
If any customer is having any problem with his/her account login then this page let them help about it.

**6.7 TRANSACTION** :

This page tells the customer about their transaction, profile, account details etc.



**6.8 All Snapshots**





**7. CODING.**

**🡪 Coding for login page.**

using System;

using System.Data;

using System.Configuration;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Web.UI.HtmlControls;

public partial class \_Default : System.Web.UI.Page

{

conclass c1 = new conclass();

protected void Page\_Load(object sender, EventArgs e)

{

lbl\_msg.Text = "";

}

protected void btn\_submit\_Click(object sender, EventArgs e)

{

String str;

str = " SELECT COUNT(\*) FROM dbo.Admin\_Login WHERE ";

str += " LoginID= '" + txt\_login.Text + "' AND";

str += " Passwd= '" + txt\_passwd.Text + "'";

if (Convert.ToInt32(c1.selectScalar(str)) > 0)

{

Session.Add("LoginID", txt\_login.Text);

Response.Redirect("~/administrator/homeadmin.aspx");

}

else

{

lbl\_msg.Text = "Invalid User.... ";

}

}

protected void WUC\_topmenu1\_Load(object sender, EventArgs e)

{

}

protected void txt\_login\_DataBinding(object sender, EventArgs e)

{

}

protected void WUC\_footer1\_Load(object sender, EventArgs e)

{

}

}

🡪Coding for Database Connectivity.

using System;

using System.Data;

using System.Configuration;

using System.Web;

using System.Data.SqlClient;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Web.UI.HtmlControls;

/// <summary>

/// Summary description for conclass

/// </summary>

public class conclass

{

String conStr = "SERVER=AKSHAY;DATABASE=ONLINE BANKING SYSTEM; trusted\_connection=yes; ";

public conclass()

{

//

// TODO: Add constructor logic here

//

}

SqlConnection con = new SqlConnection();

SqlCommand cmd = new SqlCommand();

SqlDataAdapter adpt = new SqlDataAdapter();

public DataSet selectDS(String sql)

{

DataSet ds = new DataSet();

con.ConnectionString = conStr;

con.Open();

cmd.CommandText = sql;

cmd.Connection = con;

adpt.SelectCommand = cmd;

adpt.Fill(ds);

con.Close();

return ds;

}

public SqlDataReader selectRD(String sql)

{

SqlDataReader RD;

con.ConnectionString = conStr;

con.Open();

cmd.CommandText = sql;

cmd.Connection = con;

RD = cmd.ExecuteReader();

return RD;

}

public String selectScalar(String str)

{

string v;

con.ConnectionString = conStr;

con.Open();

cmd.CommandText = str;

cmd.Connection = con;

v = Convert.ToString(cmd.ExecuteScalar());

con.Close();

return v;

}

public void executeQry(String sql)

{

con.ConnectionString = conStr;

con.Open();

cmd.CommandText = sql;

cmd.Connection = con;

cmd.ExecuteNonQuery();

con.Close();

}

}

**8. Software Testing.**

Software Testing is an empirical investigation conducted to provide stakeholders with information about the quality of the product or service under test, with respect to the context in which it is intended to operate.

Software testing, depending on the testing method employed, can be implemented at any time in the development process, however the most test effort is employed after the requirements have been defined and coding process has been completed.

**Testing Methods:-**

* Black Box Testing.
* White Box Testing.

**Black Box Testing.**

Black box testing treats the software as a "blackbox," without any knowledge of internal implementation. Black box testing methods include: equivalence partitioning, boundary value analysis, all-pairs testing, fuzz testing, model-based testing, traceability matrix, exploratory testing and specification-based testing.

**White Box Testing.**

White box testing, by contrast to black box testing, is when the tester has access to the internal data structures and algorithms (and the code that implement these)

Types of white box testing:-

**🡪 Api testing -**

Testing of the application using Public and Private APIs.

Code coverage - creating tests to satisfy some criteria of code coverage. For example, the test designer can create tests to cause all statements in the program to be executed at least once.

**🡪 Fault injection methods.**

**🡪 Mutation testing methods.**

**🡪 Static testing** –

White box testing includes all static testing.

**9. ADVANTAGES & DISADVANTAGES.**

**Advantages**

🡪Opening & closing of accountes

🡪Make the payments  of  merchandise  transaction

through Debit & Credit cards.

🡪 It gives reliefs to their customer from carrying heavy cash.

🡪 Enables prompt & speedy operation to clients.

🡪 It saves lot of time to their customers &convenient to access.

🡪 It brings efficiency in CRM(Customer relationship management).

**Disadvantages**

🡪 Customer may have to face risky transaction & fraud.

🡪 Failure of power supply cause to break down of system.

🡪 Loss of heavy income at times of settlement of higher magnitude.

🡪 Cost involved in trainning staff may not be profitable

specially in times of attrition.

🡪 Sometimes may leads to threats from outside.

**10. CONCLUSION.**

Thus reaching to the conclusion of our project we observe that Traditional  banks offer many services to their customers, including accepting customer’s   money, deposits, providing

various  banking  services to  customers,  and making

loans  to  individuals  and  companies.  Compared  with

traditional channels of offering banking services through physical branches, e-banking uses the Internet to deliver traditional banking services to their customers,such as opening accounts, transferring funds, and electronic bill payment.

E-banking services are delivered to customers through the Internet and the web using Hypertext Markup Language (HTML). In order to use the e-banking services, customers  need

Internet access and web browser software.

Multimedia  information in HTML format from

online banks can be displayed in web browsers. The heart of the e-banking application is the computer system,

which includes web servers, database management

systems, and web application programs that can generate dynamic HTML pages.

The range of e-banking services is likely to increase in the future. Some banks plan to introduce electronic money and electronic checks. Electronic money can be stored in computers or smart cards and consumers can use the electronic money to purchase small value items over the Internet.