

MASTER OF COMPUTER APPLICATIONS
II Year I Semester
INTERNET TECHNOLOGIES

MC310

Unit I:

HTML: Common tags- List, Tables, Images, Forms, Frames and IFrames, Cascading Style Sheets, Introduction to Java Scripts, Dynamic HTML with Java Script.

XML: Defining XML tags, their attributes and values, Document Type Definition, XML Schemas, Document Object Model, DOM and SAX Parsers, XHTML

Unit II:

Introduction to PHP: Declaring variables, data types, arrays, strings, operators, expressions, control structures, functions, Reading data from web form controls like text boxes, radio buttons, lists etc., Handling File Uploads, Connecting to database (MySQL as reference), executing simple queries, handling results, Handling sessions and cookies

Unit III:

Object Oriented Programming with PHP: Creating classes and objects, public, private and protected access, constructor and destructor, Inheritance, Overriding and overloading methods, calling base class methods, static members, interfaces, abstract classes

File Handling: File operations like opening, closing, reading, writing, appending, deleting etc. on text and binary files, listing directories

XML Processing: creating an XML file, using simple XML functions, extracting elements and attributes, modifying XML elements and attributes, adding and deleting elements and attributes, parsing XML file with parser functions. Simple Ajax implementation with PHP

Unit IV:

Introduction to Servlets: Lifecycle of a Servlet, The Servlet API, The javax.servlet Package, Reading Servlet parameters, Reading Initialization parameters, Handling Http Request & Responses, Using Cookies and Sessions

Introduction to JSP: The Anatomy of a JSP Page, JSP Processing, Declarations, Directives, Expressions, Code Snippets, Implicit objects, Using Beans in JSP Pages, Using Cookies-Session Tracking, Security Issues, Database Access: Using JDBC to access Database from JSPs and Servlets

Unit V:

Introduction to MVC architecture, Anatomy of a simple struts application, struts-config.xml file, Presentation layer with JSP, Struts Controller class, JSP bean, html and logic tag libraries, ActionForms, DynaActionForm, Actions, Forwarding, Error Handling, Database Connection Pooling, validation framework and examples for simple data types, Internationalization

TEXT BOOKS:

1. Web Programming, building internet applications, Chris Bates 2nd edition,

WILEY Dreamtech (UNIT 1)

2. The Complete Reference PHP – Steven Holzner, Tata McGraw-Hill (Unit 2/3)
3. Java Server Pages – Hans Bergsten, SPD O'Reilly (UNITs 3,4,5)
4. The World of Scripting Languages , David Barron, Wiley Publications.
5. Professional Jakarta Struts – James Goodwill, Richard Hightower, Wrox Publishers.

REFERENCE BOOKS.

1. Programming world wide web, R.W. Sebesta, Fourth edition, Pearson.
2. Core SERVLETS AND JAVASERVER PAGES VOLUME 1: CORE TECHNOLOGIES , Marty Hall and Barry Brown Pearson
3. Internet and World Wide Web – How to program , Dietel and Nieto, Pearson.
4. Jakarta Struts Cookbook – Bill Siggelkow, S.P.D O'Reilly.
5. Professional Java Server Programming, S. Allamaraju and others Apress(dreamtech).
6. Java Server Programming, Ivan Bayross and others The X Team, SPD
7. Web Warrior Guide to Web Programming-Bai/Ekedaw-Thomas
8. Beginning Web Programming-Jon Duckett WROX.
9. Java Server Pages, Pekowsky, Pearson.
10. Java Script, D. Flanagan, O'Reilly, SPD.

MASTER OF COMPUTER APPLICATIONS

II Year I Semester

MC320

DATA WAREHOUSING AND DATA MINING

Objectives:

- To understand data mining concepts.
- To learn about various data preprocessing techniques.
- To learn about data warehousing.
- To learn about various data mining functionalities such as association rule mining, clustering, classification and outlier analysis.

UNIT I

Introduction: Fundamentals of data mining, Data Mining Functionalities, Classification of Data Mining systems, Data Mining Task Primitives, Integration of a Data Mining System with a Database or a Data Warehouse System, Issues in Data Mining.

Data Preprocessing: Need for Preprocessing the Data, Data Cleaning, Data Integration and Transformation, Data Reduction, Discretization and Concept Hierarchy Generation.

UNIT II

Data Warehouse and OLAP Technology for Data Mining: Data Warehouse, Multidimensional Data Model, Data Warehouse Architecture, Data Warehouse Implementation, Usage of Data Warehousing Online Analytical Processing and Mining

Data Cube Computation: Efficient Methods for simple Data Cube Computation (Full Cube, Iceberg Cube, Closed Cube and Shell Cube), Discovery Driven exploration of data cubes, Attribute-Oriented Induction for data characterization and its implementation

UNIT III

Mining Frequent Patterns, Associations and Correlations: Basic Concepts, The Apriori algorithm for finding frequent itemsets using candidate generation, Generating association rules from frequent itemsets, Mining frequent itemsets without candidate generation, Mining various kinds of Association Rules, Correlation Analysis

UNIT IV

Classification and Prediction: Description and comparison of classification and prediction, preparing data for Classification and Prediction

Classification by Decision Tree Induction, Bayesian Classification, Rule-Based Classification, Classification by Backpropagation

Prediction, linear and non-linear regression, evaluating accuracy of a Classifier or a Predictor

UNIT V

Cluster Analysis: Types of Data in Cluster Analysis, A Categorization of Major Clustering Methods, k-means and k-medoids methods, CLARANS, Agglomerative and divisive hierarchical clustering, chameleon dynamic modeling, DBSCAN, Grid based clustering method: STING, Conceptual Clustering, Constraint-Based Cluster Analysis, Outlier Analysis.

TEXT BOOKS:

1. Data Mining – Concepts and Techniques - Jiawei Han, Micheline Kamber and Jian Pei, 3rd edition, Morgan Kaufmann Publishers, ELSEVIER.
2. Introduction to Data Mining – Pang-Ning Tan, Michael Steinbach and Vipin Kumar, Pearson education.

REFERENCES:

1. Data Warehousing in the Real World – Sam Aanhory & Dennis Murray Pearson Edn Asia.
2. Insight into Data Mining, K.P.Soman, S.Diwakar, V.Ajay, PHI, 2008.
3. Data Warehousing Fundamentals – Paulraj Ponnaiah Wiley student Edition
4. The Data Warehouse Life cycle Tool kit – Ralph Kimball Wiley student edition
5. Building the Data Warehouse By William H Inmon, John Wiley & Sons Inc, 2005.
6. Data Mining Introductory and advanced topics –Margaret H Dunham, Pearson education
7. Data Mining Techniques – Arun K Pujari, 2nd edition, Universities Press.
8. Data Mining, V.Pudi and P.Radha Krishna, Oxford University Press.
9. Data Mining: Methods and Techniques, A.B.M Shawkat Ali and S.A.Wasimi, Cengage Learning.
10. Data Warehouse 2.0, The Architecture for the next generation of Data Warehousing, W.H.Inmon, D. Strauss, G.Neushloss, Elsevier, Distributed by SPD.

MASTER OF COMPUTER APPLICATIONS

II Year I Semester

MC330

SOFTWARE ENGINEERING

UNIT I

Introduction to Software Engineering: The evolving role of software, Changing Nature of Software

A Generic view of process: Software engineering- A layered technology, a process framework, The Capability Maturity Model Integration (CMMI), Process patterns, process assessment, personal and team process models.

Process models: The waterfall model, Incremental process models, Evolutionary process models, Specialized process models, The Unified process.

UNIT II

Software Requirements: Functional and non-functional requirements, User requirements, System requirements, Interface specification, the software requirements document.

Requirements engineering process: Feasibility studies, Requirements elicitation and analysis, Requirements validation, Requirements management.

UNIT III

System models: Context Models, Behavioral models, Data models, Object models, structured methods.

Design Engineering: Design process and Design quality, Design concepts, the design model, pattern based software design.

Creating an architectural design: software architecture, Data design, Architectural styles and patterns, Architectural Design, mapping data flow into a software architecture.

Testing Strategies: A strategic approach to software testing, test strategies for conventional software, Black-Box and White-Box testing, Validation testing, System testing, the art of Debugging.

UNIT IV

Introduction to UML: Importance of modeling, object oriented modeling, conceptual model of the UML, Architecture,

Basic Structural Modeling: Classes, Relationships, common Mechanisms, and diagrams. Advanced Structural Modeling: Advanced classes, advanced relationships, Interfaces, Types and Roles, Packages. Class & Object Diagrams: Terms, concepts, modeling techniques for Class & Object Diagrams.

UNIT V

Basic Behavioral Modeling-I: Interactions, Interaction diagrams.

Basic Behavioral Modeling-II: Use cases, Use case Diagrams, Activity Diagrams.

TEXT BOOKS: Software Engineering

1. Software Engineering :A practitioner's Approach, Roger S Pressman, sixth edition.
McGrawHill International Edition, 2005
2. Software Engineering, Ian Sommerville, seventh edition, Pearson education, 2004.
3. Grady Booch, James Rumbaugh, Ivar Jacobson : The Unified Modeling Language User Guide, Pearson Education 2nd Edition
4. Hans-Erik Eriksson, Magnus Penker, Brian Lyons, David Fado: UML 2 Toolkit, WILEY-Dreamtech India Pvt. Ltd.

REFERENCE BOOKS:

1. Software Engineering, A Precise Approach, Pankaj Jalote, Wiley India, 2010.
2. Software Engineering : A Primer, Waman S Jawadekar, Tata McGraw-Hill, 2008
3. Fundamentals of Software Engineering, Rajib Mall, PHI, 2005
4. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press.
5. Software Engineering1: Abstraction and modeling, Diner Bjorner, Springer International edition, 2006.
6. Software Engineering2: Specification of systems and languages, Diner Bjorner, Springer International edition, 2006.
7. Software Engineering Foundations, Yingxu Wang, Auerbach Publications, 2008.
8. Software Engineering Principles and Practice, Hans Van Vliet, 3rd edition, John Wiley & Sons Ltd.
9. Software Engineering 3: Domains, Requirements, and Software Design, D. Bjorner, Springer International Edition.
- 10. Introduction to Software Engineering, R.J. Leach, CRC Press.**
- 11. Meilir Page-Jones: Fundamentals of Object Oriented Design in UML, Pearson Education.**
12. Pascal Roques: Modeling Software Systems Using UML2, WILEY-Dreamtech India Pvt. Ltd.
13. Atul Kahate: Object Oriented Analysis & Design, The McGraw-Hill Companies.
14. Mark Priestley: Practical Object-Oriented Design with UML, TMH.
15. Applying UML and Patterns: An introduction to Object – Oriented Analysis and Design and Unified Process, Craig Larman, Pearson Education.
16. Object-Oriented Analysis and Design with the Unified Process By John W. Satzinger, Robert B Jackson and Stephen D Burd, Cengage Learning.
17. UML and C++, R.C. Lee, and W.M. Tepfenhart, PHI.
18. Object Oriented Analysis, Design and Implementation, B. Dathan, S. Ramnath, Universities Press.
19. OODesign with UML and Java, K. Barclay, J. Savage, Elsevier.
20. Learning UML 2.0, Russ Miles and Kim Hamilton, O'Reilly, SPD.

MC342

MASTER OF COMPUTER APPLICATIONS
II Year I Semester
ARTIFICIAL INTELLIGENCE
(ELECTIVE – 1)

Objectives:

- To learn the difference between optimal reasoning Vs human like reasoning
- To understand the notions of state space representation, exhaustive search, heuristic search along with the time and space complexities
- To learn different knowledge representation techniques
- To understand the applications of AI: namely Game Playing, Theorem Proving, Expert Systems, Machine Learning and Natural Language Processing

UNIT-I

Introduction: What is AI? Foundations of AI, History of AI, Agents and environments, The nature of the Environment, Problem solving Agents, Problem Formulation, Search Strategies

UNIT-II

Knowledge and Reasoning: Knowledge-based Agents, Representation, Reasoning and Logic, Propositional logic, First-order logic, Using First-order logic, Inference in First-order logic, forward and Backward Chaining

UNIT-III

Learning: Learning from observations, Forms of Learning, Inductive Learning, Learning decision trees, why learning works, Learning in Neural and Belief networks

UNIT-IV

Practical Natural Language Processing: Practical applications, Efficient parsing, Scaling up the lexicon, Scaling up the Grammar, Ambiguity, Perception, Image formation, Image processing operations for Early vision, Speech recognition and Speech Synthesis

UNIT-V

Robotics: Introduction, Tasks, parts, effectors, Sensors, Architectures, Configuration spaces, Navigation and motion planning, Introduction to AI based programming Tools

TEXT BOOKS

1. Stuart Russell, Peter Norvig: “Artificial Intelligence: A Modern Approach”, 2nd Edition, Pearson Education, 2007

REFERENCES

1. Artificial Neural Networks B. Yagna Narayana, PHI
2. Artificial Intelligence , 2nd Edition, E.Rich and K.Knight (TMH).
3. Artificial Intelligence and Expert Systems – Patterson PHI.
4. Expert Systems: Principles and Programming- Fourth Edn, Giarrantana/ Riley, Thomson.
5. PROLOG Programming for Artificial Intelligence. Ivan Bratka- Third Edition – Pearson Education.
6. Neural Networks Simon Haykin PHI

MASTER OF COMPUTER APPLICATIONS
II Year I Semester
INFORMATION SECURITY
(ELECTIVE – 2)

MC353

Objectives:

- Understand the basic categories of threats to computers and networks
- Understand various cryptographic algorithms.
- Describe public-key cryptosystem.
- Describe the enhancements made to IPv4 by IPSec
- Understand Intrusions and intrusion detection
- Discuss the fundamental ideas of public-key cryptography.
- Generate and distribute a PGP key pair and use the PGP package to send an encrypted e-mail message.
- Discuss Web security and Firewalls

UNIT – I

Attacks on Computers and Computer Security: Introduction, The need for security, Security approaches, Principles of security, Types of Security attacks, Security services, Security Mechanisms, A model for Network Security **Cryptography: Concepts and Techniques:** Introduction, plain text and cipher text, substitution techniques, transposition techniques, encryption and decryption, symmetric and asymmetric key cryptography, steganography, key range and key size, possible types of attacks.

UNIT – II

Symmetric key Ciphers: Block Cipher principles & Algorithms(DES, AES,Blowfish), Differential and Linear Cryptanalysis, Block cipher modes of operation, Stream ciphers, RC4,Location and placement of encryption function, Key distribution **Asymmetric key Ciphers:** Principles of public key cryptosystems, Algorithms(RSA, Diffie-Hellman,ECC), Key Distribution

UNIT – III

Message Authentication Algorithms and Hash Functions: Authentication requirements, Functions, Message authentication codes, Hash Functions, Secure hash algorithm, Whirlpool, HMAC, CMAC, Digital signatures, knapsack algorithm.

UNIT – IV

E-Mail Security: Pretty Good Privacy, S/MIME **IP Security:** IP Security overview, IP Security architecture, Authentication Header, Encapsulating security payload, combining security associations, key management

UNIT – V

Web Security: Web security considerations, Secure Socket Layer and Transport Layer Security, Secure electronic transaction **Intruders, Virus and Firewalls:** Intruders, Intrusion detection, password management, Virus and related threats, Countermeasures, Firewall design principles, Types of firewalls

Case Studies on Cryptography and security: Secure Inter-branch Payment Transactions, Cross site Scripting Vulnerability, Virtual Elections.

TEXT BOOKS:

1. Cryptography and Network Security : William Stallings, Pearson Education, 5th Edition
2. Cryptography and Network Security: Atul Kahate, Mc Graw Hill, 2nd Edition.
3. Network Security and Cryptography: Bernard Menezes, CENGAGE Learning

REFERENCES:

1. Cryptography and Network Security: C K Shyamala, N Harini, Dr T R Padmanabhan, Wiley India, 1st Edition.
2. Cryptography and Network Security : Forouzan Mukhopadhyay, Mc Graw Hill, 2nd Edition
3. Information Security, Principles and Practice : Mark Stamp, Wiley India.
4. Principles of Computer Security: WM.Arthur Conklin, Greg White, TMH
5. Introduction to Network Security: Neal Krawetz, CENGAGE Learning.
6. Principles of Information security by Michael E Whitman and Herbert J.Mattord.

MASTER OF COMPUTER APPLICATIONS
II Year I Semester
INTERNET TECHNOLOGIES LAB

MC360

List of Sample Problems

1. Develop static pages (using Only HTML) of an online Book store. The pages should resemble: www.amazon.com The website should consist the following pages.
 Home page, Registration and user Login
 User Profile Page, Books catalog
 Shopping Cart, Payment By credit card
 Order Conformation
2. Validate the Registration, user login, user profile and payment by credit card pages using JavaScript.
3. Create and save an XML document at the server, which contains 10 users information. Write a program, which takes User Id as an input and returns the user details by taking the user information from the XML document.
4. Bean Assignments
 - a. Create a JavaBean which gives the exchange value of INR(Indian Rupees) into equivalent American/Canadian/Australian Dollar value.
 - b. Create a simple Bean with a label - which is the count of number of clicks. Than create a BeanInfo class such that only the “count” property is visible in the Property Window.
 - c. Create two Beans-a)Keypad .b)DisplayPad .After that integrate the two Beans to make it work as a Calculator.
 - d. Create two Beans Traffic Light(Implemented as a Label with only three background colours-Red,Green,Yellow) and Automobile(Implemented as a TextBox which states its state/movement). The state of the Automobile should depend on the following Light Transition Table.

Light Transition	Automobile State
Red ---> Yellow	Ready
Yellow ---> Green	Move
Green --> Red	Stopped
5. Install TOMCAT web server. Convert the static web pages of assignments 2 into dynamic web pages using Servlets and cookies. Hint: Users information (user id, password, credit card number) would be stored in web.xml. Each user should have a separate Shopping Cart.
6. Redo the previous task using JSP by converting the static web pages of assignments 2 into dynamic web pages. Create a database with user information and books information. The books catalogue should be dynamically loaded from the database. Follow the MVC architecture while doing the website.
7. Implement the “Hello World!” program using JSP Struts Framework.
- 8.Redo the problem 5 using PHP.

Additional Assignment Problems for the IT Lab.:
--

Write an HTML page including any required Javascript that takes a number from one text field in

the range of 0 to 999 and shows it in another text field in words. If the number is out of range, it should show “out of range” and if it is not a number, it should show “not a number” message in the result box.
Write a java swing application that takes a text file name as input and counts the characters, words and lines in the file. Words are separated with white space characters and lines are separated with new line character.
Write a simple calculator servlet that takes two numbers and an operator (+, -, /, * and %) from an HTML page and returns the result page with the operation performed on the operands. It should check in a database if the same expression is already computed and if so, just return the value from database. Use MySQL or PostgreSQL.(Do the same problem using PHP)
Write an HTML page that contains a list of 5 countries. When the user selects a country, its capital should be printed next to the list. Add CSS to customize the properties of the font of the capital (color, bold and font size).
Write a servlet that takes name and age from an HTML page. If the age is less than 18, it should send a page with “Hello <name>, you are not authorized to visit this site” message, where <name> should be replaced with the entered name. Otherwise it should send “Welcome <name> to this site” message. (Do the same problem using PHP)
Write a calculator program in HTML that performs basic arithmetic operations (+, -, /, * and %). Use CSS to change the foreground and background color of the values, buttons and result display area separately. Validate the input strings using JavaScript regular expressions. Handle any special cases like division with zero reasonably. The screen may look similar to the following:
<div> <div>Value 1</div> <div>Operator</div> <div>Value 2</div> <div>=</div> <div>Result</div> </div> <div> <input type="text"/> <input type="text" value="+"/> <input type="text"/> <input type="text" value="="/> <input type="text"/> </div>
Write a Java program that creates a calculator GUI, as shown in figure. Extra components may be added for convenience:

The Color Scheme may be Black on White or Blue on Yellow (selectable) and accordingly all components colors must be changed. The values can be either entered or increased or decreased by a step of 10. The operators are +, -, / and * (selectable). Once any change

takes place, the result must be automatically computed by the program.

Write a Java Application that will read an XML file that contains personal information (Name, Mobile Number, age and place). It reads the information using SAX parser. After reading the information, it shows two input Text Fields in a window, one for tag name and the other for value. Once these two values are given, it should list all the records in the XML file that match the value of the given field in a text area (result box). For example, if the two text boxes are entered with “name” and “ABCD” then it should show all the records for which name is “ABCD”? An Illustration is given below that takes a mobile number and lists all the records that have the same mobile number.

Consider the following web application for implementation:

The user is first served a login page which takes user's name and password. After submitting the details the server checks these values against the data from a database and takes the following decisions.

If name and password matches, serves a welcome page with user's full name.

If name matches and password doesn't match, then serves “password mismatch” page

If name is not found in the database, serves a registration page, where users full name, present user name (used to login) and password are collected. Implement this application using:

1. Pure JSP
2. Pure Servlets

3. Struts Framework
4. PHP

Implement a simple arithmetic calculator with +, -, /, *, % and = operations using Struts Framework. The number of times the calculator is used should be displayed at the bottom (use session variable).

MASTER OF COMPUTER APPLICATIONS
II Year I Semester
MC380 SOFTWARE ENGINEERING LAB

Objectives:

The student should be able to

understand how to analyze a problem

create requirement specifications.

design the software as a solution to the problem.

represent the design using Unified Modeling Language.

Create the views like use case view, logical view, component view, Deployment view,

Design Database

Do forward and Reverse Engineering, and Generation of documentation of the project.

2. Student has to take up another case study of his/her own interest and do the same whatever mentioned in first problem. Some of the ideas regarding case studies are given in reference books which were mentioned in theory syllabus can be referred for some idea.

Problems on Software Engineering:

Consider Bank ATM as example, write the SRS and design the system for Cash withdrawal, Balance inquiry and PIN change. Write possible test cases for the applications.

Consider a bus reservation application and develop the SRS and design the system for new reservation and cancellation (Ignore cash payment part). Write the test cases for this application.

Consider a library automation application and develop the SRS and design the system for add new book, issue a book, return a book and search for author/book title. Write the test cases for this application.

Problems on UML:**Unified Modeling Language Lab:**

Students are divided into batches of 5 each and each batch has to draw the following diagrams using UML for an ATM system whose description is given below.

UML diagrams to be developed are:

1. Use Case Diagram.
2. Class Diagram.
3. Sequence Diagram.

4. Collaboration Diagram.
5. State Diagram
6. Activity Diagram.
7. Component Diagram
8. Deployment Diagram.
9. Test Design.

Description for an ATM System

The software to be designed will control a simulated automated teller machine (ATM) having a magnetic stripe reader for reading an ATM card, a customer console (keyboard and display) for interaction with the customer, a slot for depositing envelopes, a dispenser for cash (in multiples of Rs. 100, Rs. 500 and Rs. 1000), a printer for printing customer receipts, and a key-operated switch to allow an operator to start or stop the machine. The ATM will communicate with the bank's computer over an appropriate communication link. (The software on the latter is not part of the requirements for this problem.)

The ATM will service one customer at a time. A customer will be required to insert an ATM card and enter a personal identification number (PIN) - both of which will be sent to the bank for validation as part of each transaction. The customer will then be able to perform one or more transactions. The card will be retained in the machine until the customer indicates that he/she desires no further transactions, at which point it will be returned - except as noted below.

The ATM must be able to provide the following services to the customer:

1. A customer must be able to make a cash withdrawal from any suitable account linked to the card, in multiples of Rs. 100 or Rs. 500 or Rs. 1000. Approval must be obtained from the bank before cash is dispensed.
2. A customer must be able to make a deposit to any account linked to the card, consisting of cash and/or checks in an envelope. The customer will enter the amount of the deposit into the ATM, subject to manual verification when the envelope is removed from the machine by an operator. Approval must be obtained from the bank before physically accepting the envelope.
3. A customer must be able to make a transfer of money between any two accounts linked to the card.
4. A customer must be able to make a balance inquiry of any account linked to the card.
5. A customer must be able to abort a transaction in progress by pressing the Cancel key instead of responding to a request from the machine.

The ATM will communicate each transaction to the bank and obtain verification that it was allowed by the bank. Ordinarily, a transaction will be considered complete by the bank once it has been approved. In the case of a deposit, a second message will be sent to the bank indicating that the customer has deposited the envelope. (If the customer fails to deposit the envelope within the timeout period, or presses cancel instead, no second message will be sent to the bank and the deposit will not be credited to the customer.)

If the bank determines that the customer's PIN is invalid, the customer will be required to re-enter the PIN before a transaction can proceed. If the customer is unable to successfully enter the PIN after three tries, the card will be permanently retained by the machine, and the customer will have to contact the bank to get it back.

If a transaction fails for any reason other than an invalid PIN, the ATM will display an explanation of the problem, and will then ask the customer whether he/she wants to do another transaction.

The ATM will provide the customer with a printed receipt for each successful transaction

The ATM will have a key-operated switch that will allow an operator to start and stop the servicing of customers. After turning the switch to the "on" position, the operator will be required to verify and enter the total cash on hand. The machine can only be turned off when it is not servicing a customer. When the switch is moved to the "off" position, the machine will shut down, so that the operator may remove deposit envelopes and reload the machine with cash, blank receipts, etc.

REFERENCE BOOKS

- 1.Learning UML 2.0,Russ Miles and Kim Hamilton,O'Reilly,SPD.
- 2.Mastering UML with Rational Rose,W.Boggs&M.Boggs,Wiley India.