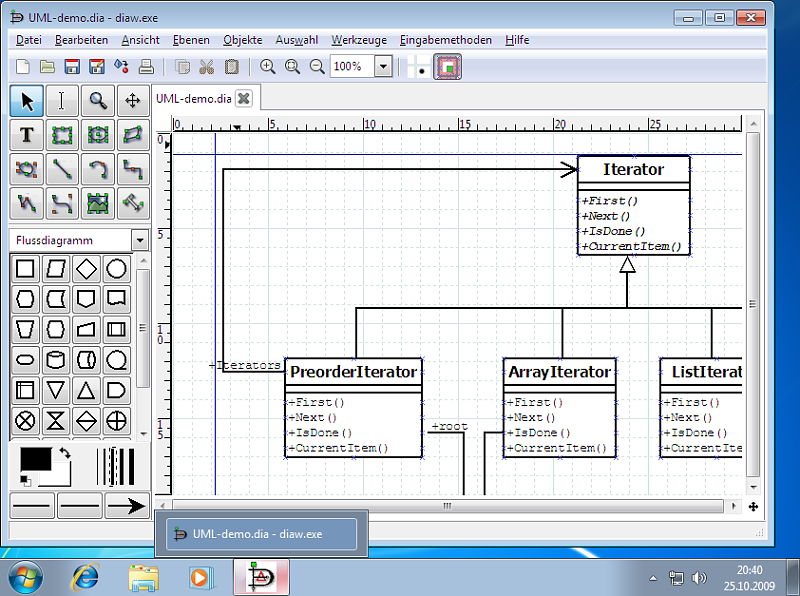
**1] Study andprepare a case study ofVisual Modelling UMLtools like Dia Software,Argo UML, VisualParadigmTool,containingintroductionpart,GUIinterface,exploremenubar,toolbar,status bar, browser window, diagram toolbar and diagram window.**

1. **DiaSoftware:-**



Dia Software is an open-source diagram creation tool that facilitates visual modeling using the Unified ModelingLanguage(UML).Developed toprovideasimpleandintuitiveplatform,Diaiswidelyutilized for creating a variety of diagrams, including flowcharts, entity-relationship diagrams, and UMLdiagrams. This case study delves into the key aspects of Dia Software, ranging from its graphical user interface to specific features like the menubar, toolbar, and diagram creation tools.

# GUIInterface

Dia Software boasts a clean and minimalist graphical user interface, ensuring ease of use for both beginnersand experiencedusers.Theinterfaceisdesignedtobeintuitive,with astraightforward layoutthat allows users to focus on the diagram creation process.

# Menubar

Dia'smenubarfollowsaconventionalstructure,offeringessentialoptionsforuserstomanagetheir diagrams and design elements. It contains File, Edit,View, Tools, Objects, Help.

# Toolbar

Dia's toolbar is strategically designed to offer quick access to frequently used tools during the diagram creationprocess.Thetoolbarincludesiconsforselectingobjects,drawingshapes,connectingelements,and aligning objects.

# StatusBar

Dia's status bar is a simple yet informative element at the bottom of the interface. It provides basic informationabout thecurrent stateofthediagram, suchasthetypeofobject beinghovered overorthe coordinates of the cursor.

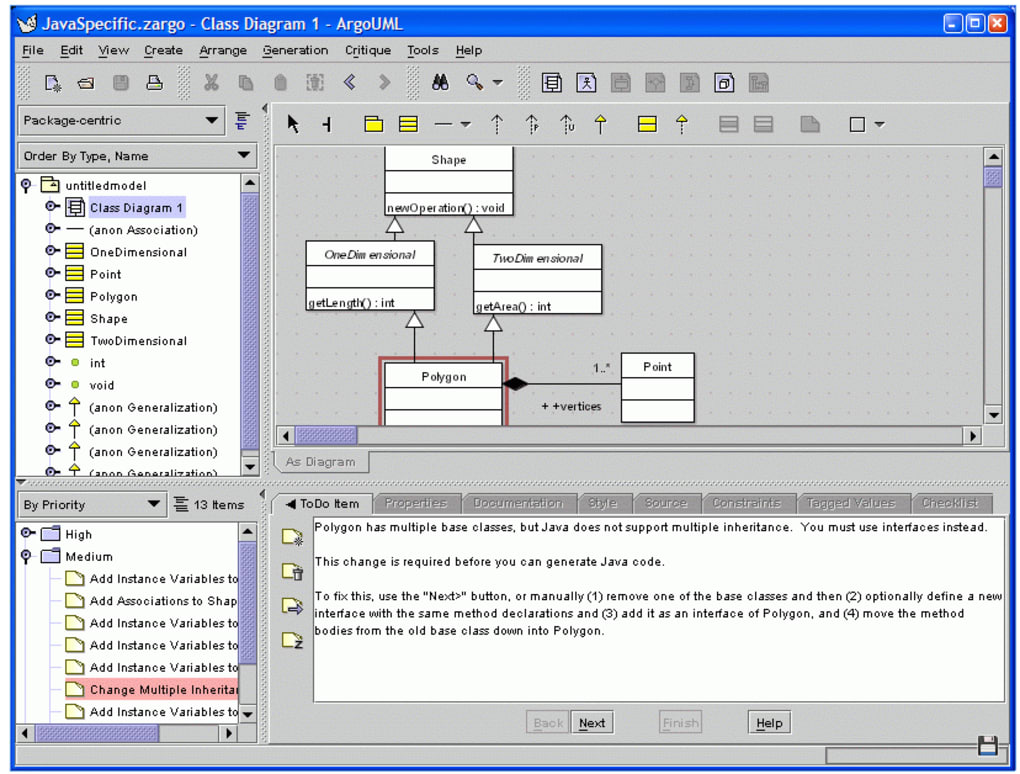
# BrowserWindow

Diafeaturesa straightforward objectbrowserthataidsusersin managingelementswithin thediagram.The object browser provides a hierarchical view of the diagram's components, allowing for easy navigation and organization.

# DiagramToolbarandDiagramWindow

Dia's diagram toolbar is designed tocater to the specific needs of UMLdiagram creation. It includes tools fordrawingclasses, relationships,usecasediagrams,andmore.Thediagramwindowservesasthecanvas for users to visually represent their system models, offering an uncluttered space for arranging and connecting elements.

# ArgoUML:-



ArgoUMLstandsasapowerfulandopen-sourceUnifiedModelingLanguage(UML)tool, providing software engineers with a comprehensive platform for designing and visualizing complex systems.

Developed tocatertothediverseneedsofsoftwaremodeling,ArgoUMLisknown foritsrobustfeatureset and user-friendly interface.

# GUIInterface:

Argo UMLpresents users with a traditional yet effective graphical user interface, balancing familiarity withfunctionality.TheinterfaceisorganizedtoprovideastructuredworkspaceforUMLdiagramcreation, making it suitable for both novice and experienced users.

# Menubar:

ArgoUML'smenubaristhoughtfullyorganized, offeringa rangeofoptionstomanagetheUMLmodeling process, Key Components include File, Edit, Project,View, Window.

# Toolbar:

Argo UML's toolbar isadynamiccomponent that adaptsto theselected diagram type. It provides quick accesstoessentialtoolsforcreatingandmodifyingUMLelements. Iconsforclasscreation,relationship establishment, and various diagram-specific actions contribute to a streamlined and efficient modeling process.

# StatusBar:

ThestatusbarinArgoUMLservesasaninformative element,providinguserswith essentialdetailsabout the ongoing modeling activities.

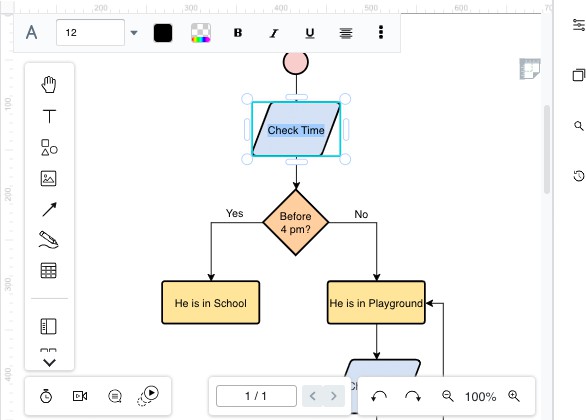
# BrowserWindow

ArgoUMLincorporatesaclassexplorerasitsbrowserwindow,offeringahierarchicalviewoftheproject's classes and relationships. This feature aids in efficient navigation, allowing users to manage and organize elements within the project.

# DiagramToolbarandDiagramWindow:

ArgoUML'sdiagramtoolbarisdynamic,adaptingtotheselected UMLdiagramtype. It providesusers with specialized tools for creating and modifying elements specific to the chosen diagram.

# VisualParadigmTool

****

* + VisualParadigmToolstandsasaversatileandfeature-richUnifiedModelingLanguage(UML)tool, designed to meet the complex demands of modern software engineering. Renowned for its advanced capabilities, Visual ParadigmTool offers an extensive suite of tools for system modeling, design, and documentation.

# GUIInterface

VisualParadigmToolboastsamodernandintuitivegraphicaluserinterface,reflectingitscommitmentto providing a comprehensive yet user-friendly environment for UMLmodeling. The interface is designed to accommodate a variety of users, from beginners to professionals.

# Menubar:

VisualParadigmTool'smenubarisexpansive, offeringawiderangeofoptionstomanagetheUML modeling process, Keycomponents include File, Edit, Diagram, Tools , Window.

# Toolbar:

VisualParadigmTool'stoolbarisrobustandcustomizable,offeringavastarrayoftoolsforcreatingand modifying UML elements.

# StatusBar:

ThestatusbarinVisualParadigmTooliscomprehensive,providinguserswithdetailedinformationabout the ongoing modeling activities. It displays contextual information, such as mouse coordinates, object properties, and other relevant details.

# BrowserWindow:

Visual ParadigmTool features a project browser as itsbrowser window, offering a comprehensive overviewoftheentireproject structure.Thisincludesahierarchicalviewof diagrams,classes,and other project elements.

# DiagramToolbarandDiagramWindow:

* + VisualParadigmTool's diagramtoolbaris extensiveanddynamic, adaptingtotheselected UMLdiagram type. Itprovidesuserswithspecializedtoolsforcreatingandmodifyingelementsspecifictoeach diagram.

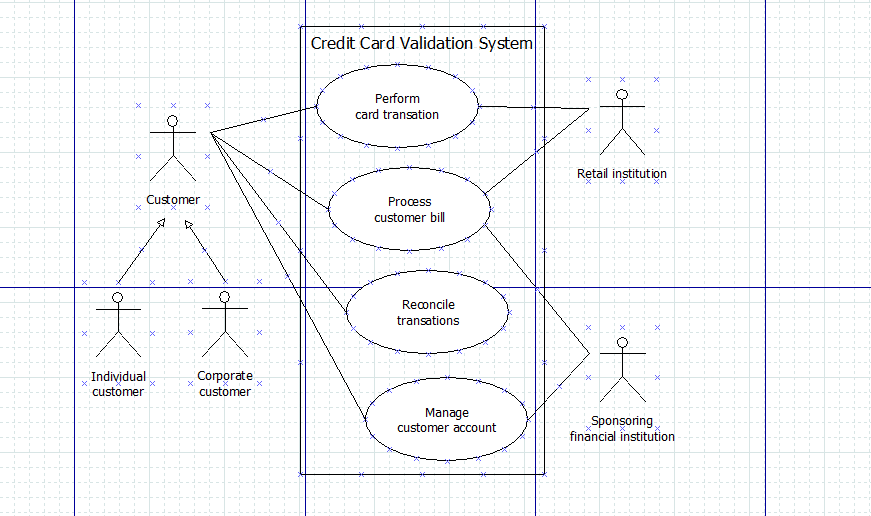
**Practical-2**

# PrepareUseCasediagramtomodelthebehaviourofCellularPhoneNetworksystem.

****

**Practical-3**

# PrepareUseCasediagramshowsthecontextofacreditcardvalidationsystem.



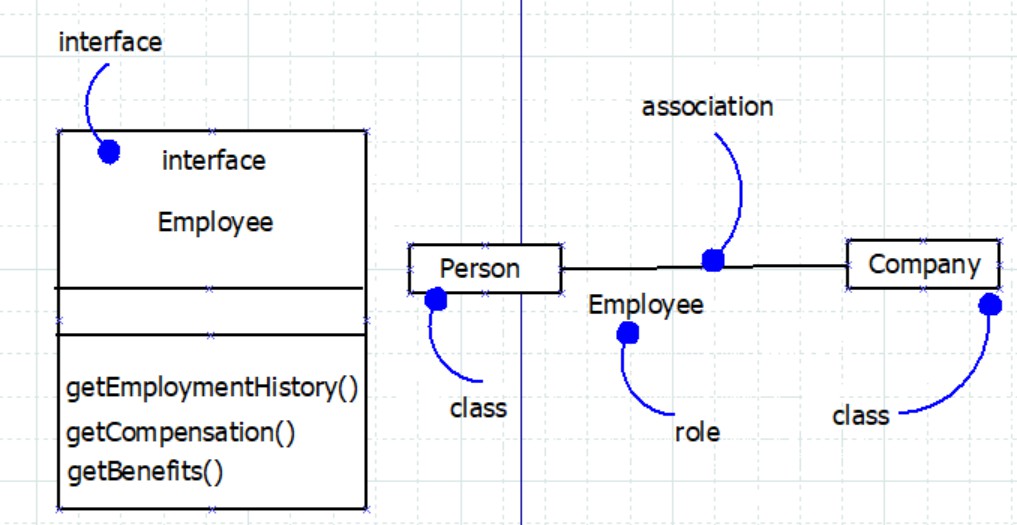
**Practical-4**

**Drawa diagramforshowinginterfaceanditsrealizationrelationship.Also drawdiagramfor showing Types and Roles.**

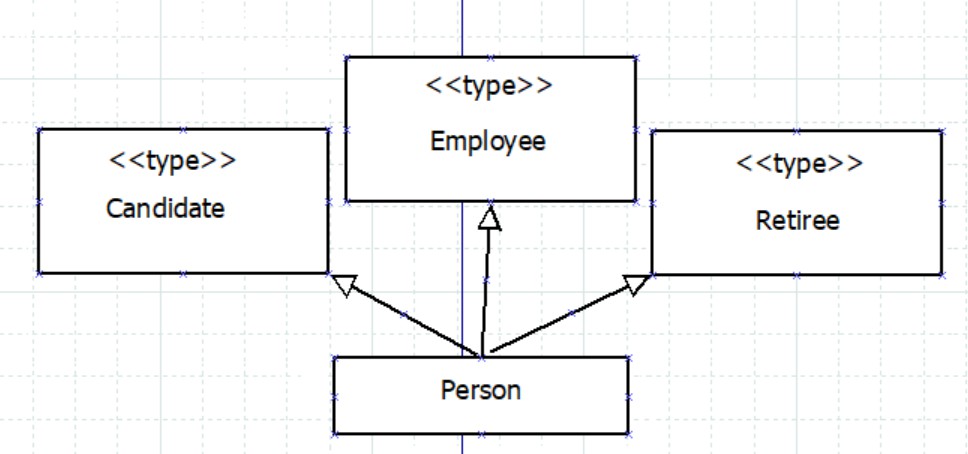
**Interfaces**

****

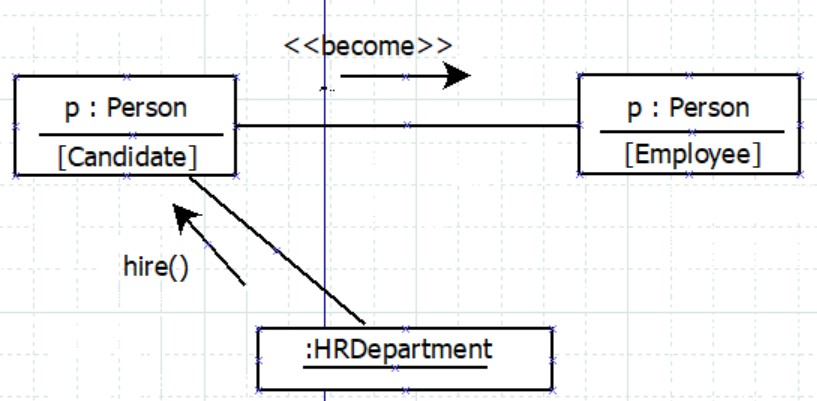
**Roles**

****

* **ModelingStaticTypes**

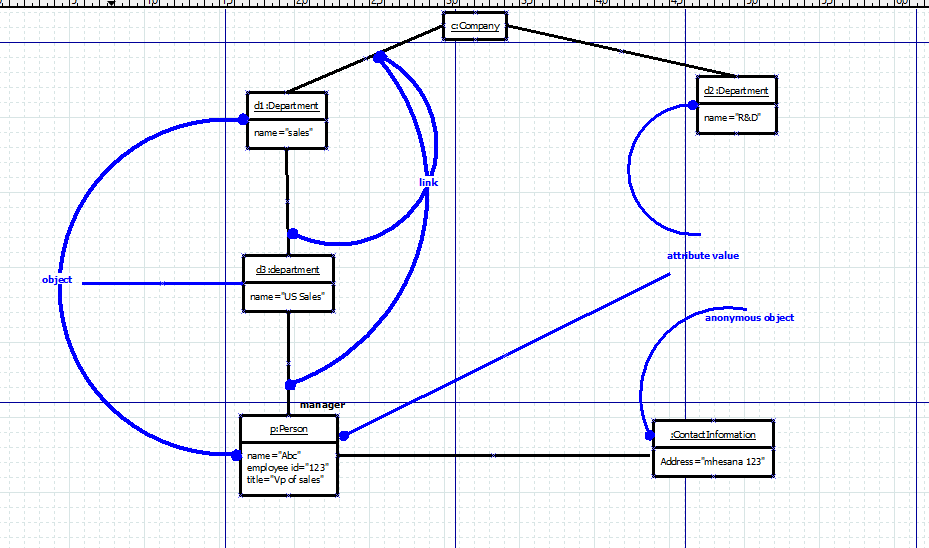
****

* **ModelingDynamicTypes**

****

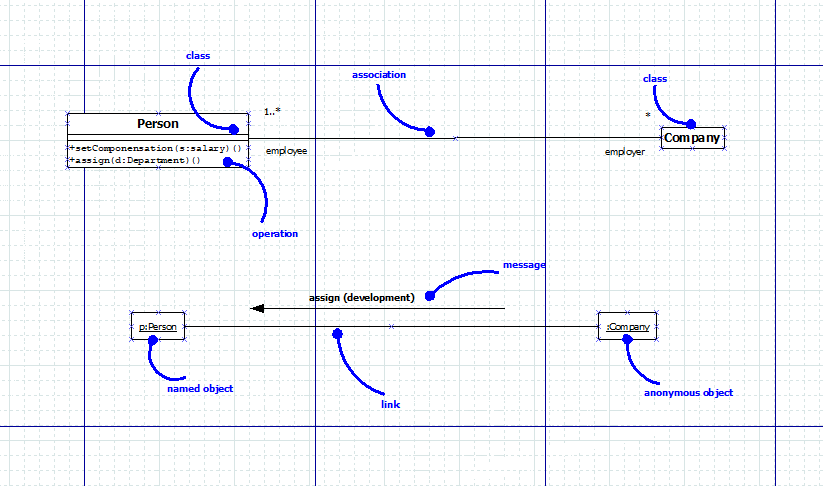
**Practical-5**

# DrawObjectDiagramwithasuitable example.

****

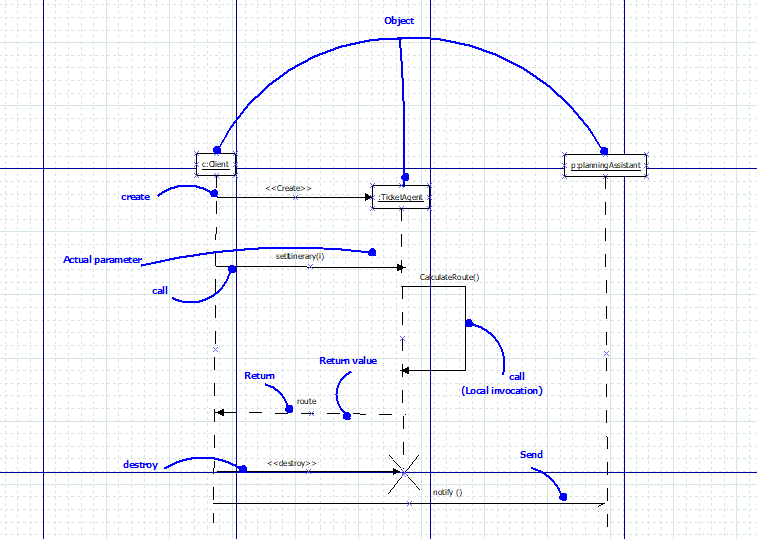
**Practical-6**

# DrawaDiagramwithasuitableexampletoshowlinkandassociation.

****

**Practical-7**

# Drawa Diagramwitha suitableexampletoshowoperationslikeCall,Return,Send,Createand Destroy.

****

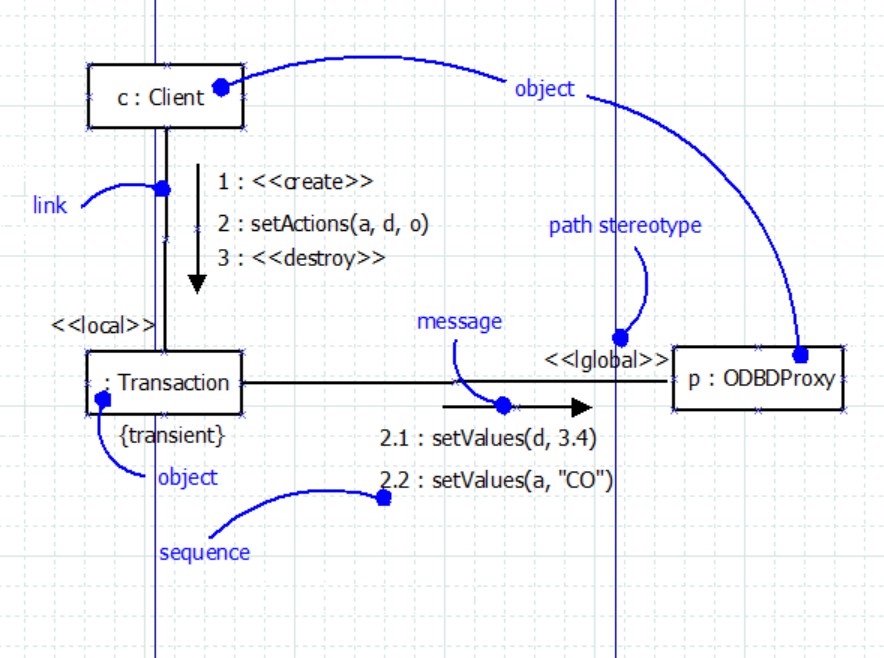
**Practical-8**

# DrawaSequencediagramshowingobject,linkand message.

****

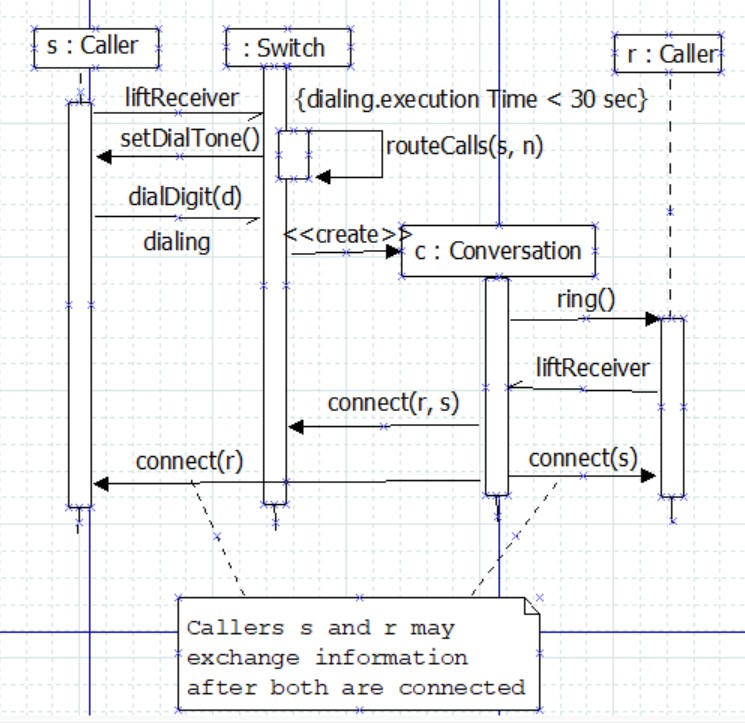
**Practical-9**

# Drawacollaborationdiagramfororganisingobjectparticipationininteraction.

****

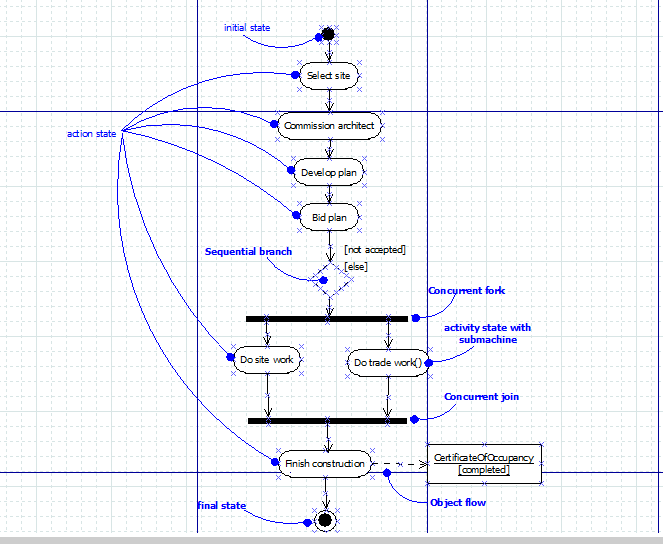
**Practical-10**

# Drawa sequence diagramthatspecifiestheflowof control involvedininitiatingasimple, two-party phone call by Time ordering.

****

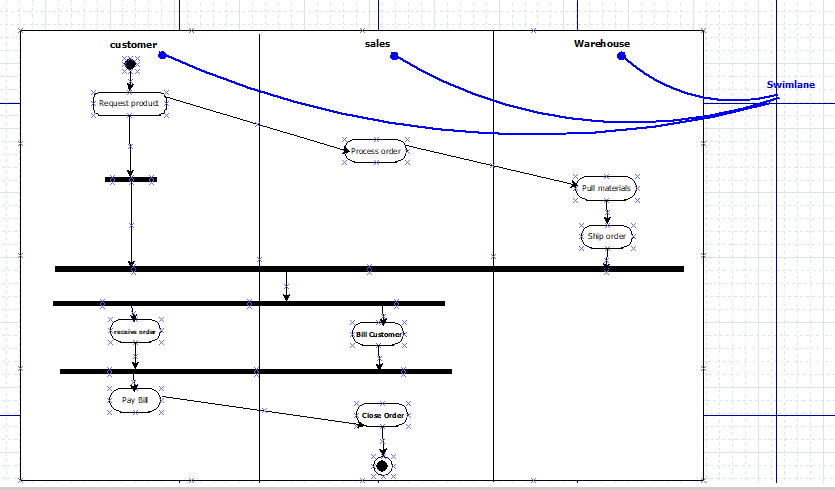
**Practical-11**

# Prepareactivitydiagram(showingactionstate,transitions,fork,join,objectflow,branch,guard expression) to focus first on the activities that take place among objects over the time.

****

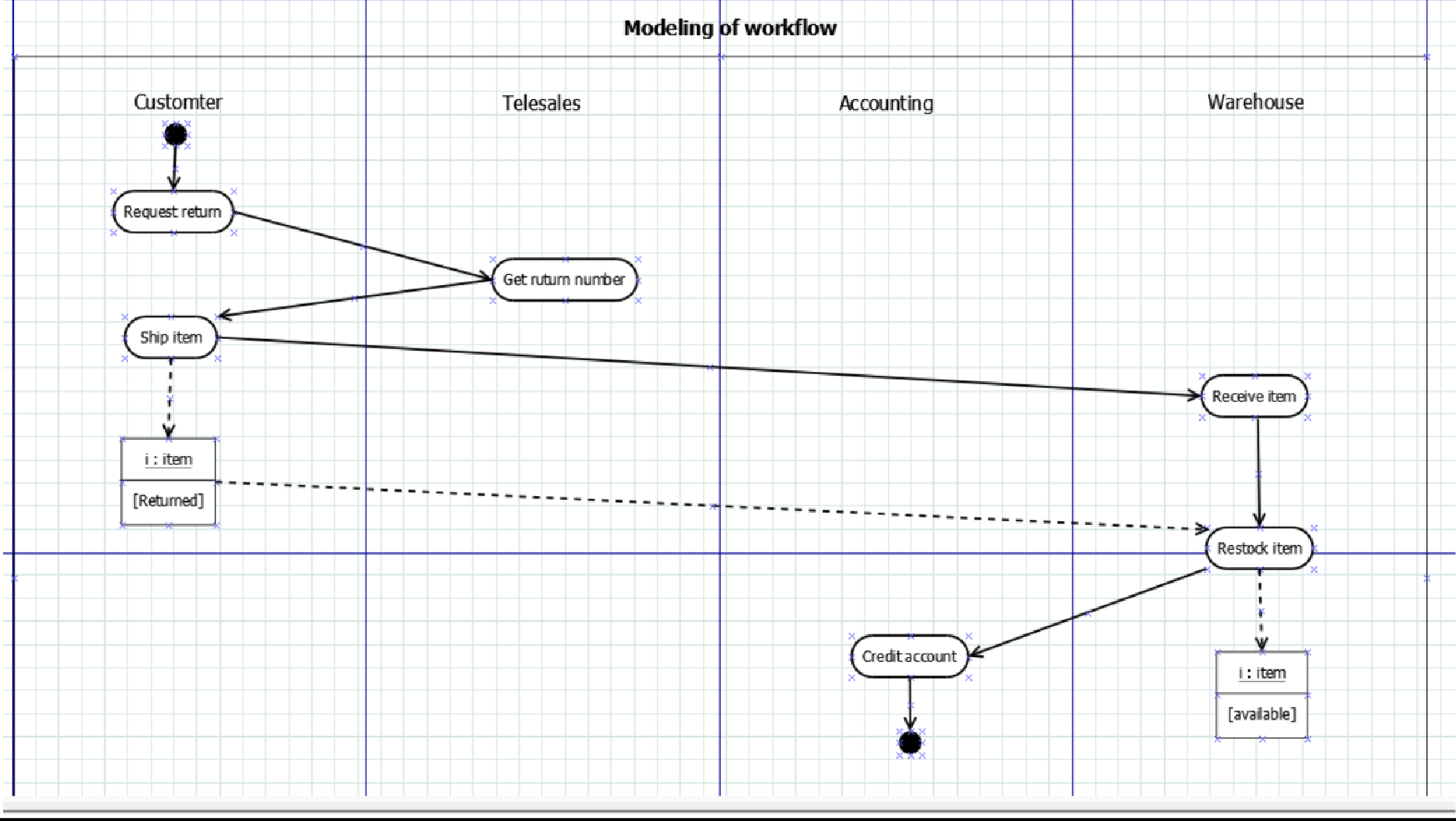
**Practical-12**

# DrawaSwimlanediagramspecifyinglocusofactivities.

****

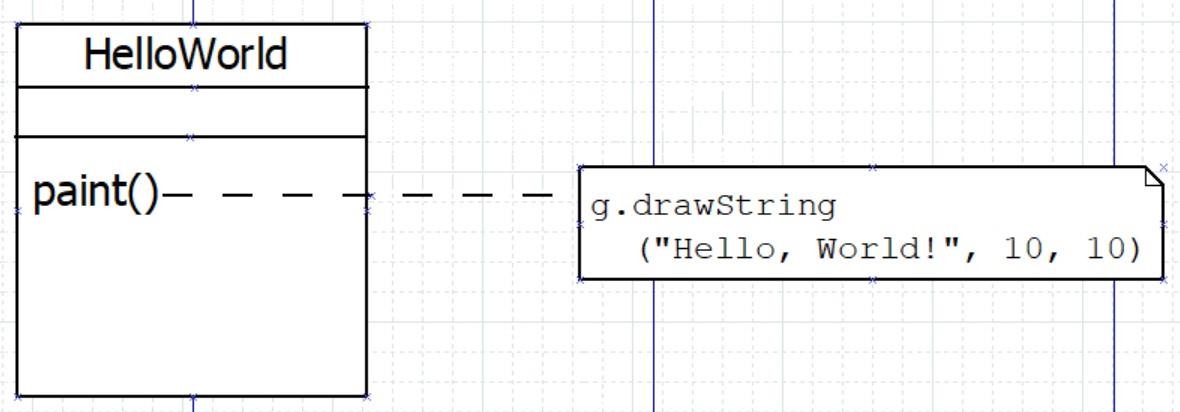
**Practical-13**

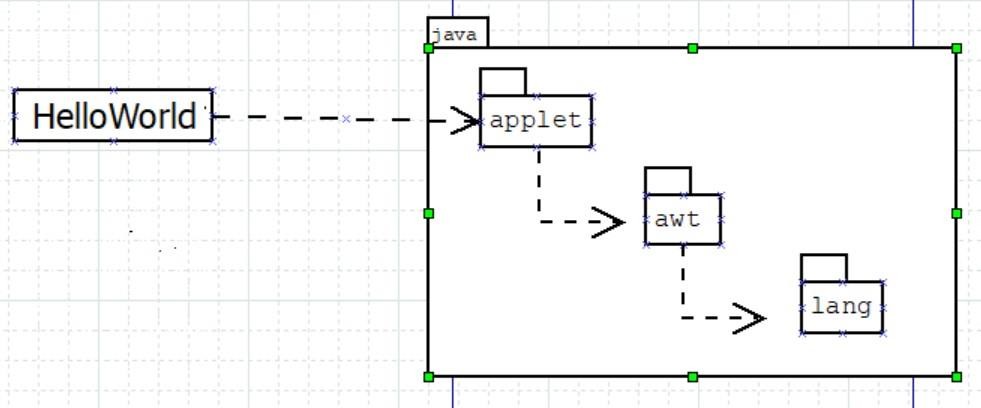
# Drawactivity diagramshowingaretailbusinessspecifies theworkflowinvolvedwhena customer returns an item from a mail order.

****

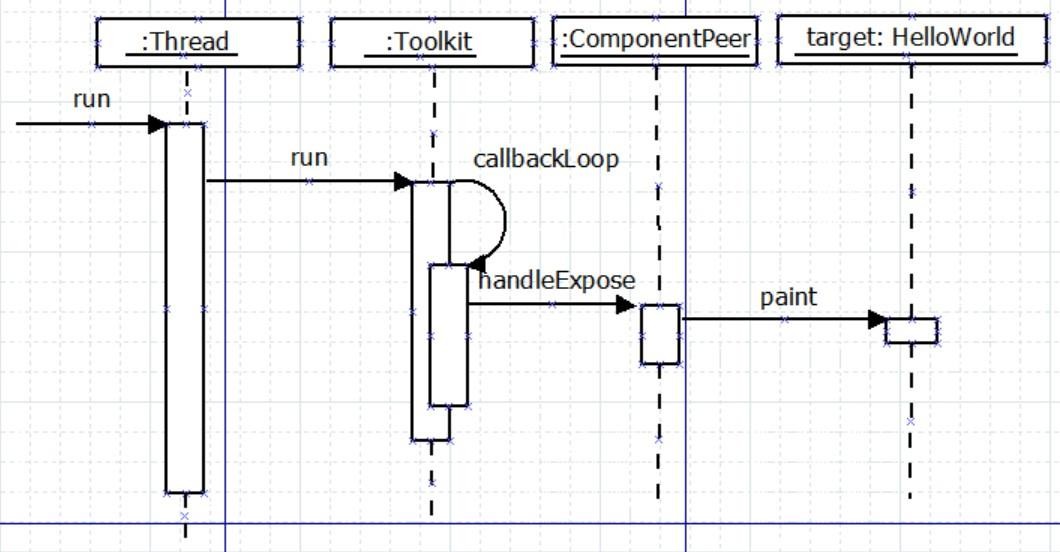
**Practical-14**

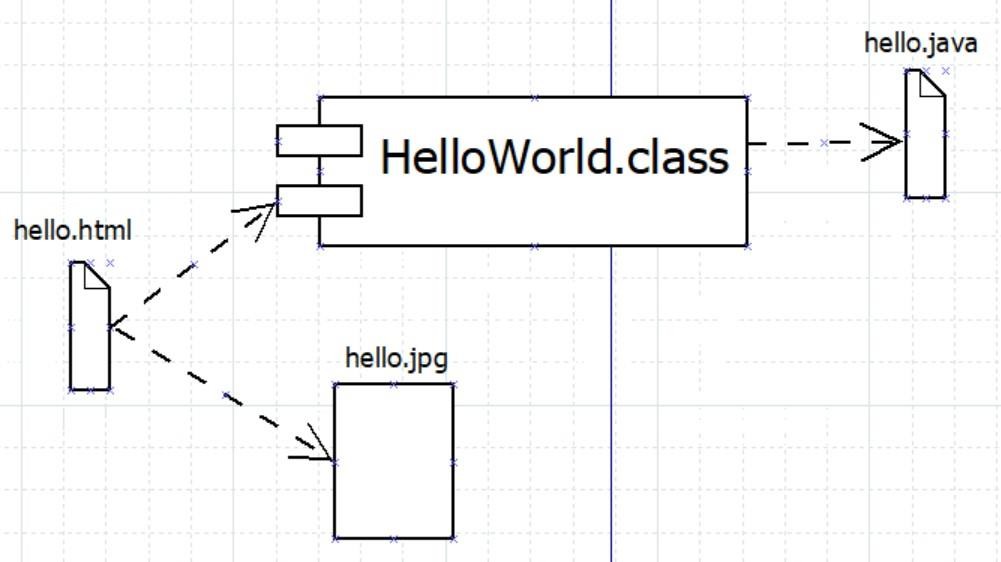
**Performforwardengineeringinjava.(Modeltocodeconversion)toshowAbstraction,package, sequence diagram and component diagram.**

* **Abstraction:**
* **Package:**

****

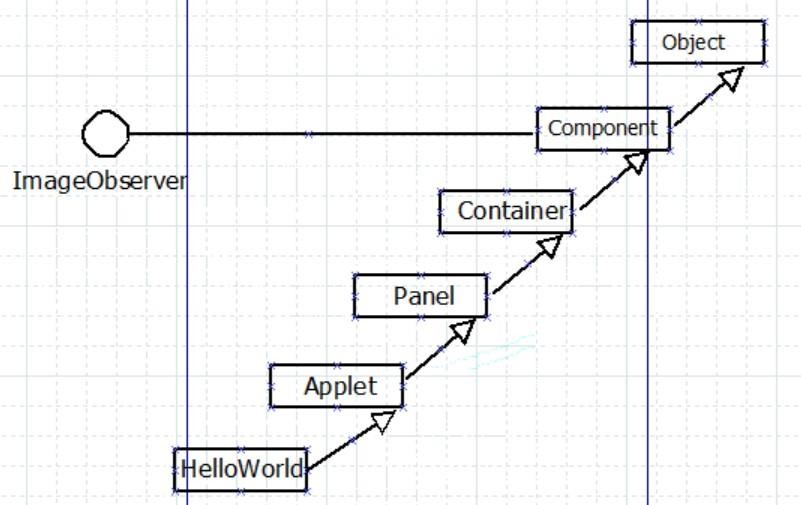
* **Sequence Diagram:**



* **Component Diagram:**

**Practical – 15**

**Perform reverse engineering in java.(Code to Model conversion).**

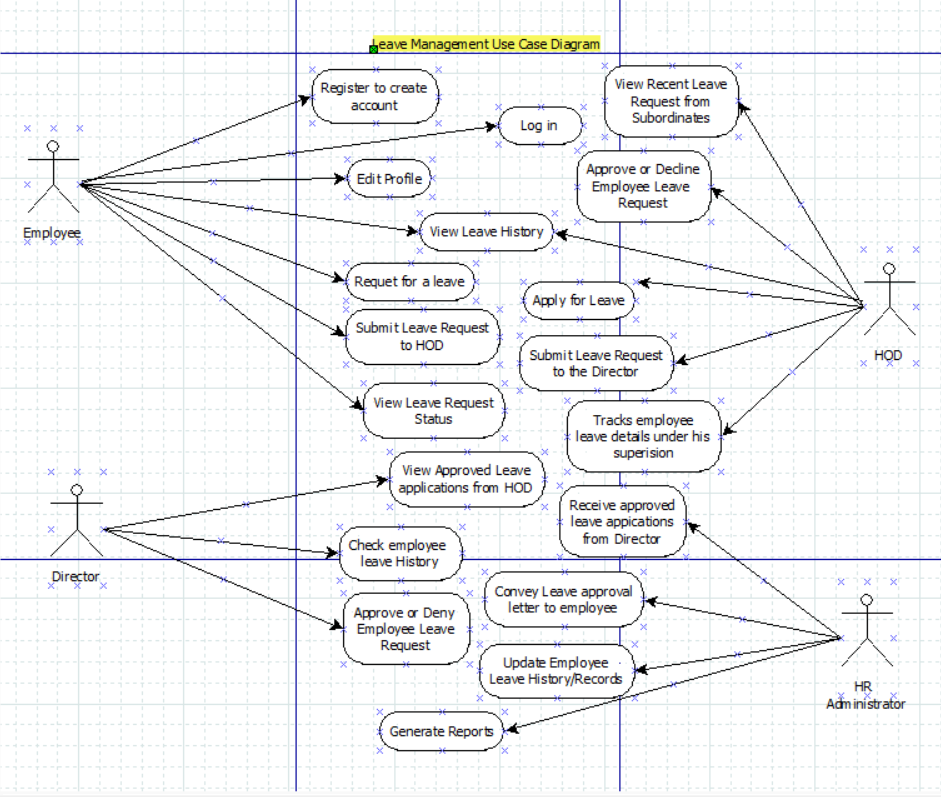
****

**Practical:-16**

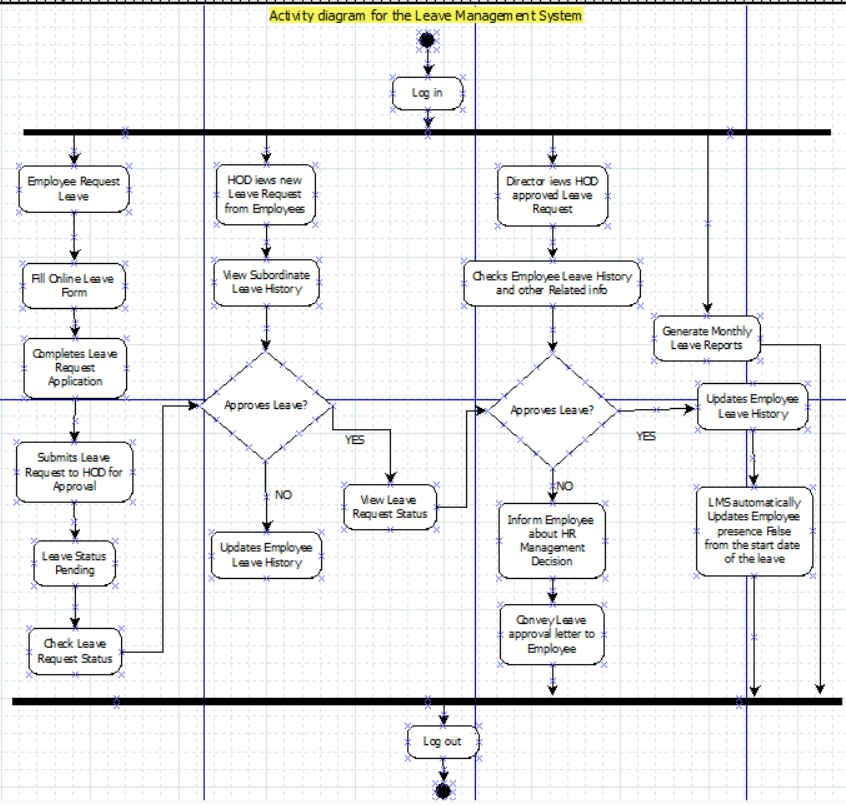
**Draw Use Case , Class Diagram, Sequence Diagram, an Activity diagram for Leave Management System.**

**Write Objective, Module Summary and design Data Dictionary of System.**

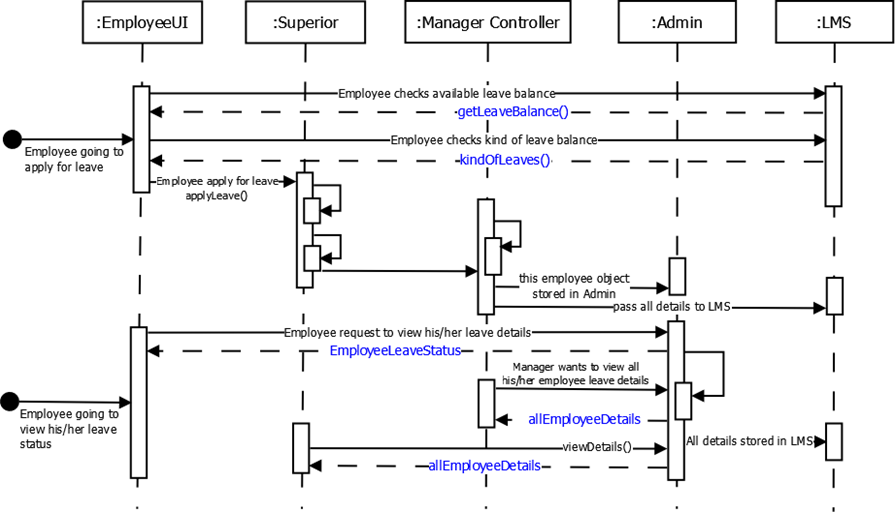
**Use case diagram**



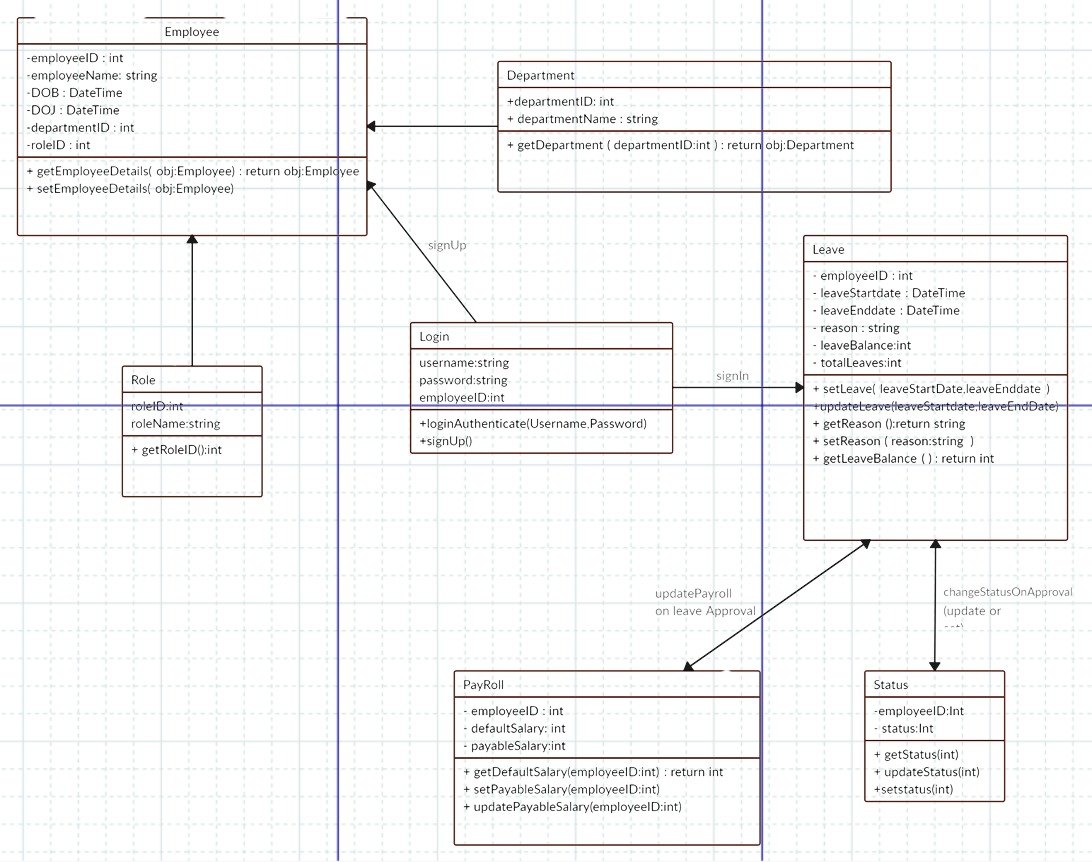
Activity Diagram for the Leave Management System



Sequence Diagramfor the Leave Management System



Class Diagram of Leave Management System



**Objective:** The objective of the Leave Management System (LMS) is to automate and streamline the process of managing employee leaves within an organization. This system aims to simplify leave requests, approvals, tracking, and reporting, thereby enhancing efficiency, transparency, and accuracy in managing employee attendance and leaves.

**Module Summary:**

1. **User Authentication Module:** This module handles user authentication and authorization, allowing employees and administrators to securely access the system with appropriate permissions.
2. **Leave Request Module:** Employees can submit leave requests through this module, specifying the type of leave (e.g., sick leave, vacation, maternity/paternity leave), duration, and any additional details. They can also view the status of their requests.
3. **Leave Approval Module:** Managers and administrators can review and approve/deny leave requests submitted by employees. This module provides them with an interface to manage leave requests efficiently, ensuring adequate coverage and compliance with organizational policies.
4. **Leave Balances Module:** This module maintains records of employees' leave balances, including accrued, taken, and available leaves. It calculates leave entitlements based on organizational policies and updates balances accordingly.
5. **Leave Calendar Module:** A visual representation of employees' leave schedules, allowing managers and team members to view who is on leave on any given day. This module helps in resource planning and scheduling tasks effectively.
6. **Reporting Module:** Generates comprehensive reports on leave usage, trends, and patterns. It provides insights to management for decision-making, such as identifying peak leave periods, analyzing absenteeism rates, and ensuring compliance with labor regulations.

**Data Dictionary:**

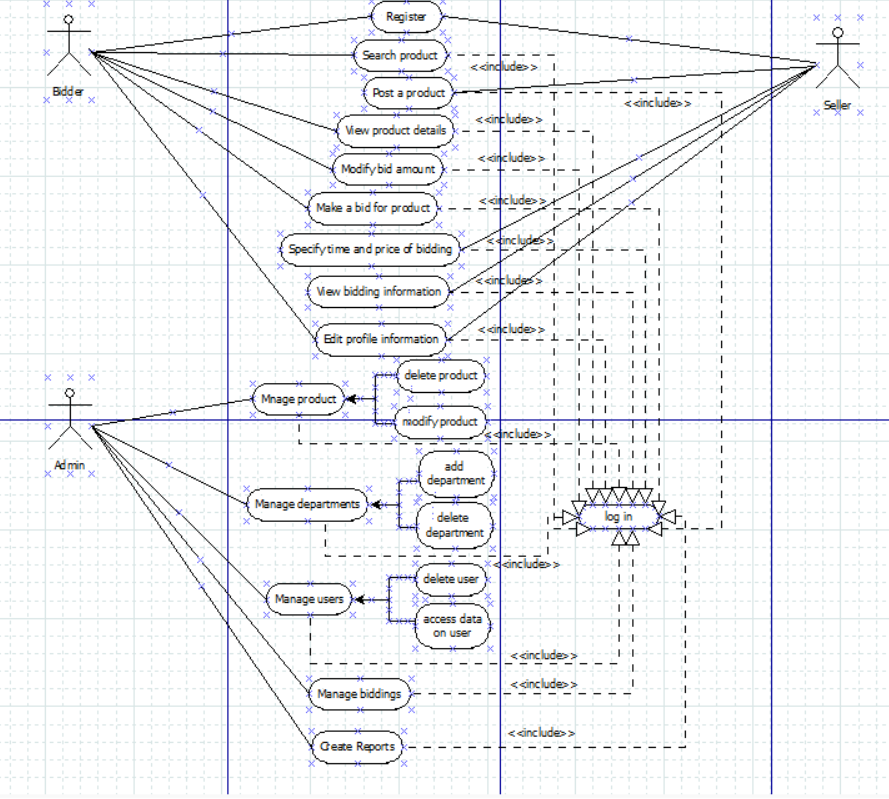
1. **User Table:**
   * UserID (Primary Key)
   * Username
   * Password (Hashed)
   * Role (Employee, Manager, Administrator)
   * Department
   * Email
   * Phone
2. **Leave Request Table:**
   * RequestID (Primary Key)
   * UserID (Foreign Key)
   * LeaveType (Sick Leave, Vacation, etc.)
   * StartDate
   * EndDate
   * Reason
   * Status (Pending, Approved, Denied)
   * ManagerID (Foreign Key, for approval workflow)
3. **Leave Balance Table:**
   * BalanceID (Primary Key)
   * UserID (Foreign Key)
   * Year
   * Total Leave Entitlement
   * LeaveTaken
   * LeaveAccrued
   * LeaveAvailable
4. **Leave Approval Table:**
   * ApprovalID (Primary Key)
   * RequestID (Foreign Key)
   * ManagerID (Foreign Key)
   * ApprovalStatus (Approved, Denied)
   * ApprovalDate
   * Comments
5. **Leave Calendar Table:**
   * CalendarID (Primary Key)
   * LeaveDate
   * UserID (Foreign Key)
   * LeaveType
   * Description
6. **Report Table:**
   * ReportID (Primary Key)
   * Report Name
   * Generated By
   * Generated Date
   * Report Content

**Practical:-17**

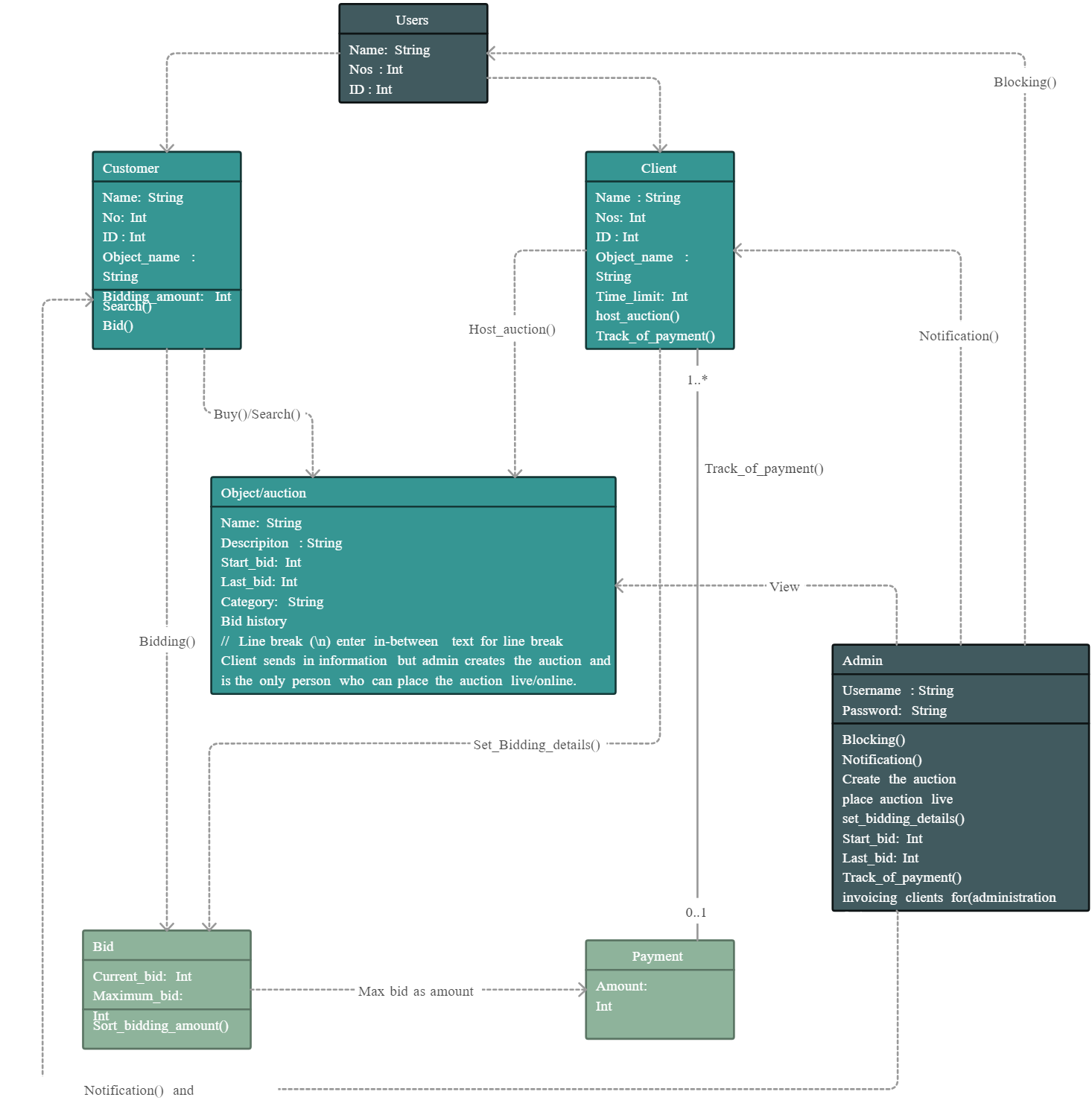
**Draw Use Case , Class Diagram, Sequence Diagram, an activity diagram for e-Auction –wherein selling, buying and service offered take place.**

**Write Objective, Module Summary and design Data Dictionary of System.**

USE CASE DIAGRAM OF E-AUCTION



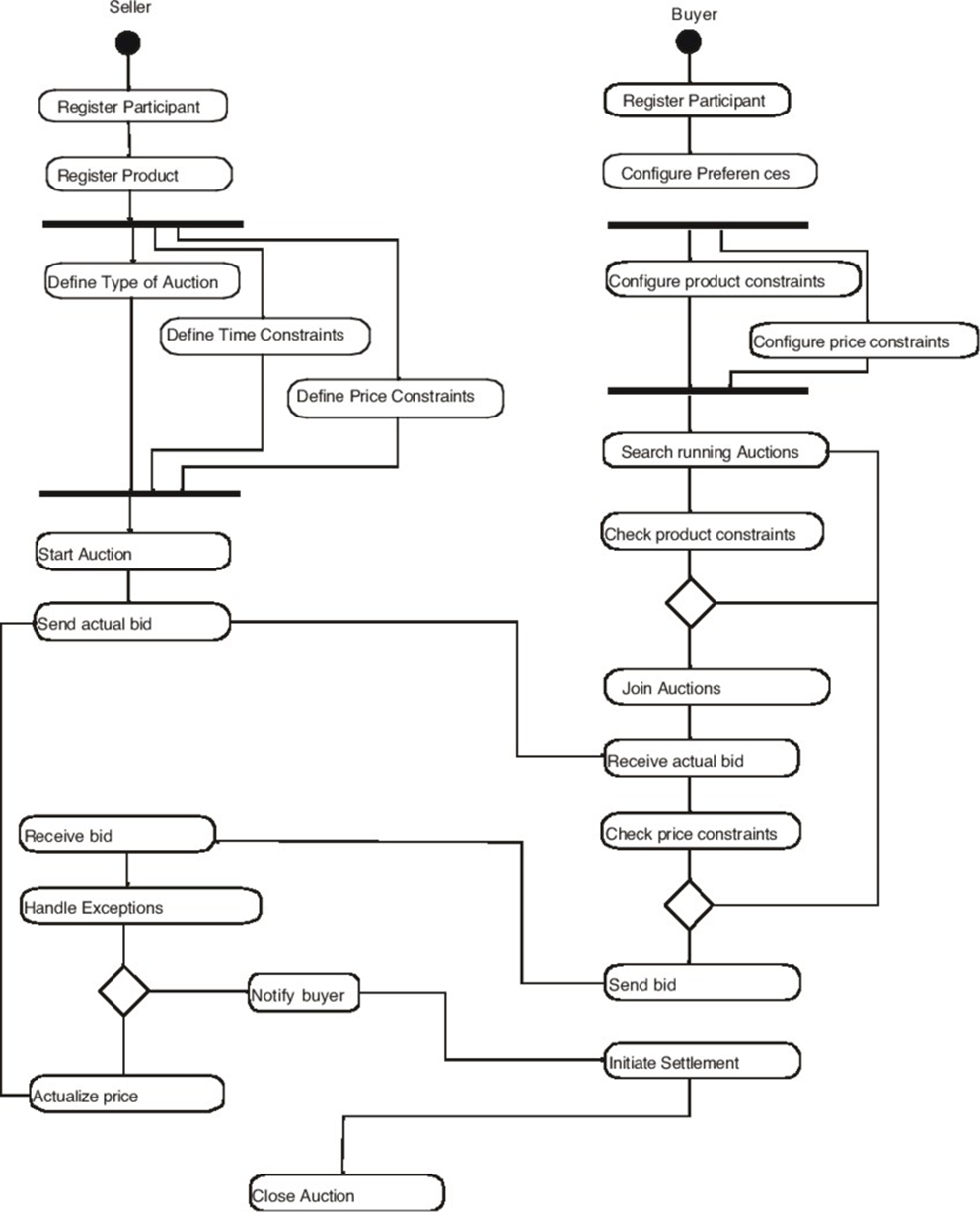
class diagram of e-auction system



sequence diagram of e-auction system



Activity diagram of e-auction system



**Objective:** The objective of the e-Auction system is to provide a digital platform for facilitating online auctions where users can buy, sell, or offer services in a transparent and efficient manner. This system aims to streamline the auction process, eliminate geographical barriers, attract a wider audience, and ensure fair competition among participants.

**Module Summary:**

1. **User Management Module:** This module handles user registration, authentication, and authorization. Users can create accounts, log in securely, and manage their profiles. Administrators have additional privileges for managing user accounts and permissions.
2. **Auction Listing Module:** Sellers can create listings for items or services they wish to auction. They provide details such as item description, images, starting bid price, reserve price (optional), and auction duration. Listings can be categorized for easy navigation.
3. **Bidding Module:** Registered users can place bids on active auction listings. They can view current highest bids, bid history, and receive notifications when they are outbid. The bidding process is automated, and bids are updated in real-time.
4. **Payment and Checkout Module:** Upon the conclusion of an auction, the winning bidder is notified, and the payment process is initiated. This module facilitates secure payment transactions, integrates with payment gateways, and provides options for shipping and delivery.
5. **Auction Management Module:** Administrators have access to tools for managing auctions, including monitoring listings, resolving disputes, enforcing auction rules, and handling reported issues. They ensure the integrity and smooth operation of the platform.
6. **Reporting and Analytics Module:** Generates reports on auction performance, user activity, sales trends, and revenue generation. It provides insights to administrators for strategic decision-making, identifying popular categories, and optimizing platform functionality.

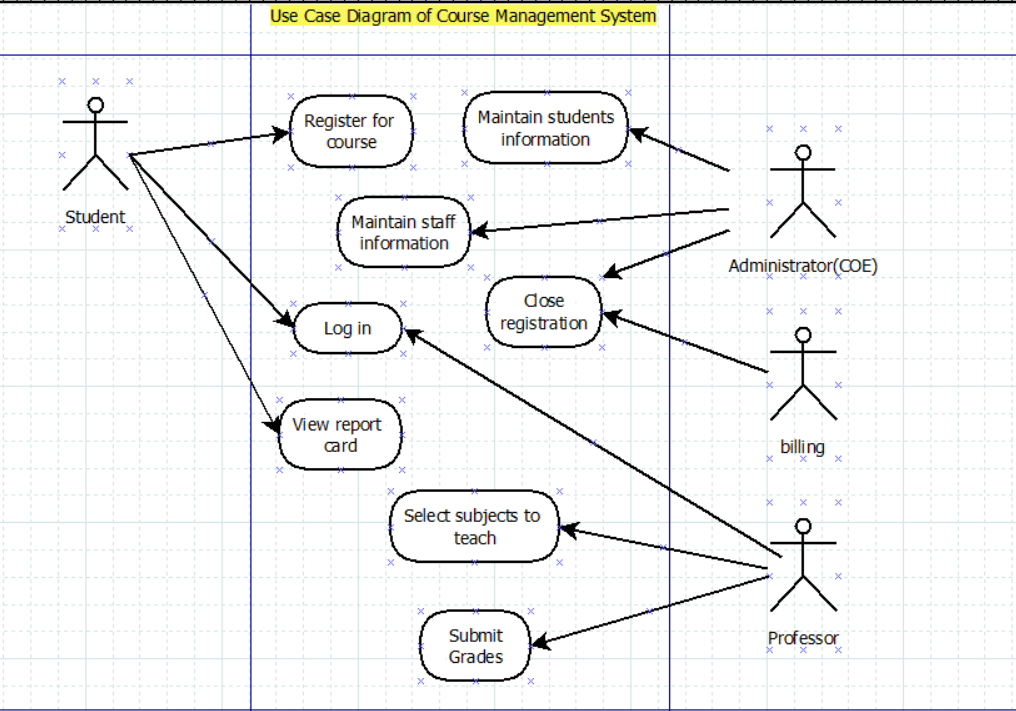
**Data Dictionary:**

1. **User Table:**
   * UserID (Primary Key)
   * Username
   * Password (Hashed)
   * Email
   * Phone
   * Address
   * UserType (Buyer, Seller, Administrator)
2. **Auction Listing Table:**
   * ListingID (Primary Key)
   * SellerID (Foreign Key)
   * Title
   * Description
   * Category
   * StartingBidPrice
   * ReservePrice
   * AuctionDuration
   * StartTime
   * EndTime
   * Status (Active, Completed)
3. **Bid Table:**
   * BidID (Primary Key)
   * ListingID (Foreign Key)
   * BidderID (Foreign Key)
   * BidAmount
   * BidTime
4. **Transaction Table:**
   * TransactionID (Primary Key)
   * BuyerID (Foreign Key)
   * SellerID (Foreign Key)
   * ListingID (Foreign Key)
   * TransactionAmount
   * TransactionDate
   * PaymentStatus (Pending, Completed)
   * DeliveryStatus (Pending, Shipped, Delivered)
5. **Auction Category Table:**
   * CategoryID (Primary Key)
   * CategoryName
6. **Report Table:**
   * ReportID (Primary Key)
   * ReportName
   * GeneratedBy
   * GeneratedDate
   * ReportContent

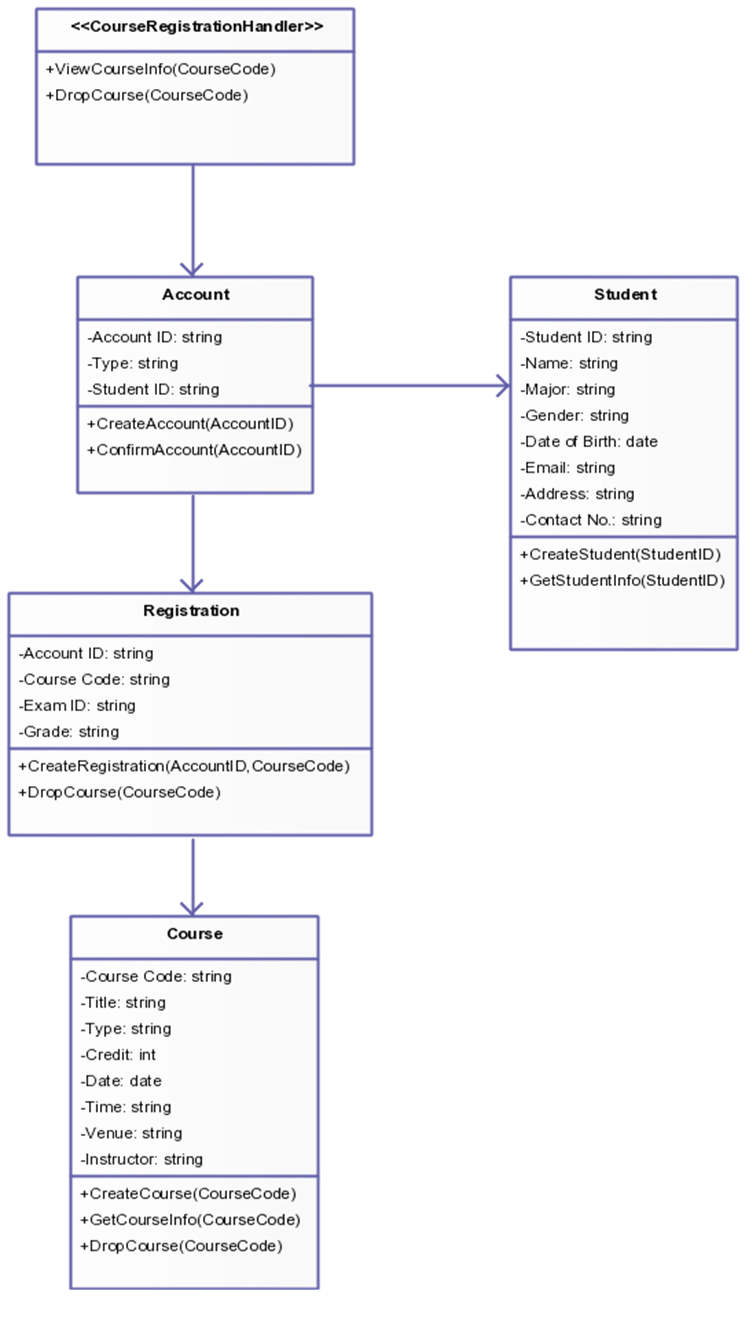
**Practical:-18**

**Draw Use Case , Class Diagram, Sequence Diagram, an activity diagram for Course Management System (CMS).**

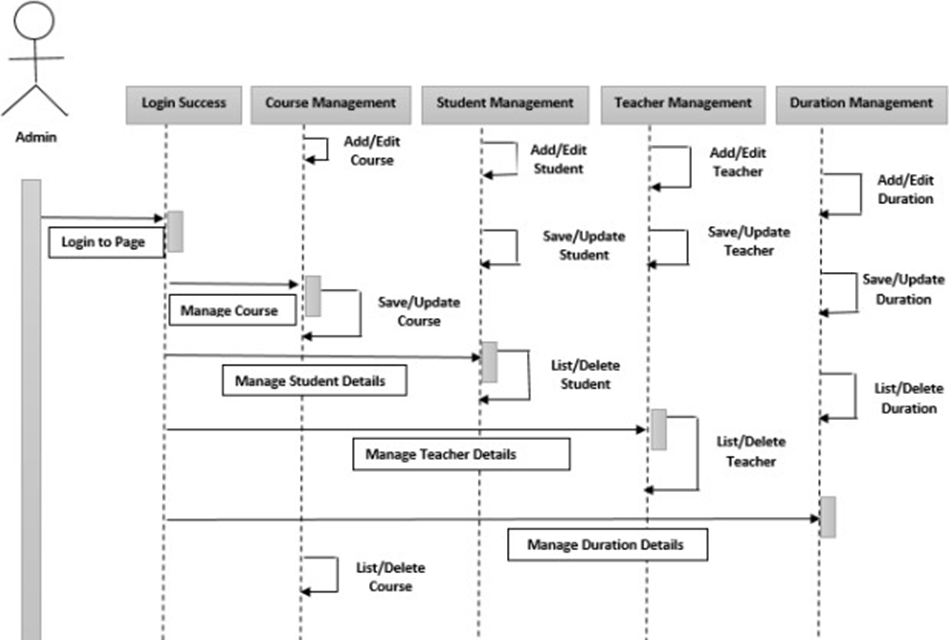
**Write Objective, Module Summary and design Data Dictionary of System.**



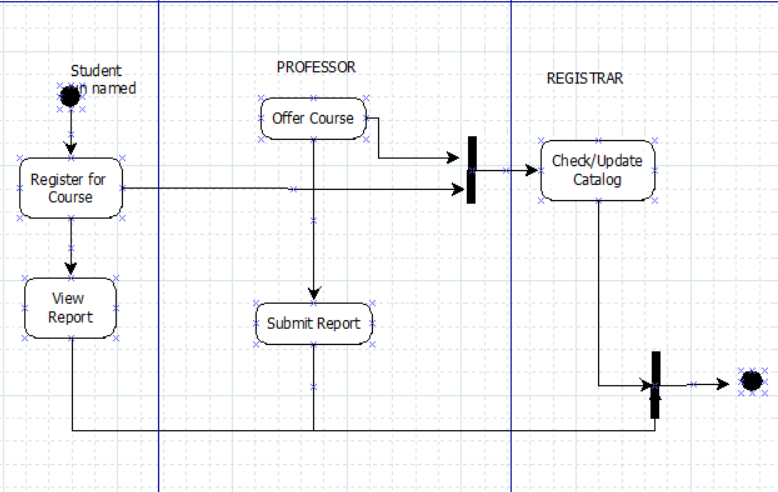
Class Diagram



Sequence Diagram of Course Management System



Activity Diagram of Course Management System



**Objective:** The objective of the Course Management System (CMS) is to provide an efficient and user-friendly platform for managing various aspects of educational courses within an institution. This system aims to streamline course administration, enhance communication between instructors and students, facilitate curriculum management, and track academic progress effectively.

**Module Summary:**

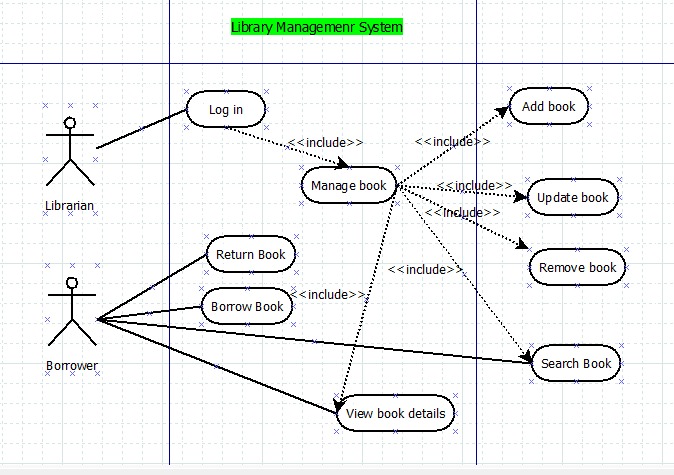
1. **User Management Module:** This module handles user authentication, registration, and profile management. It allows administrators, instructors, and students to securely access the system with appropriate permissions. Administrators have additional privileges for managing user accounts and roles.
2. **Course Creation and Management Module:** Instructors can create and manage course offerings within this module. They can define course details such as title, description, schedule, prerequisites, and enrollment capacity. They can also upload course materials, assignments, and resources for students.
3. **Enrollment Module:** Students can browse available courses, view course details, and enroll in courses of interest. This module facilitates the enrollment process, checks for prerequisites, and updates enrollment status. Administrators can manage enrollment capacities and waitlists.
4. **Gradebook Module:** Instructors can record and manage students' grades and academic performance within this module. They can create grade categories, enter grades for assignments, quizzes, and exams, calculate overall course grades, and provide feedback to students.
5. **Communication Module:** Facilitates communication between instructors and students through announcements, messages, and discussion forums. It provides a platform for sharing important course-related information, clarifying doubts, and fostering collaboration among participants.
6. **Reporting and Analytics Module:** Generates reports on course enrollment, student performance, attendance, and other metrics. Administrators can gain insights into course popularity, student engagement, and overall academic performance, facilitating data-driven decision-making.

**Data Dictionary:**

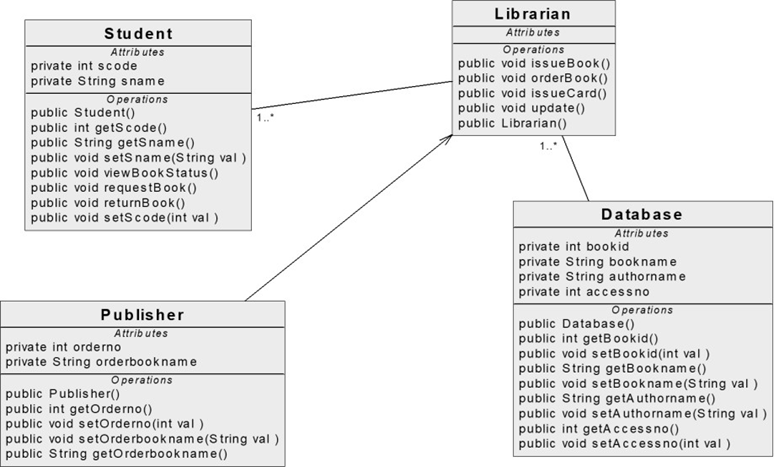
1. **User Table:**
   * UserID (Primary Key)
   * Username
   * Password (Hashed)
   * Email
   * Phone
   * UserType (Administrator, Instructor, Student)
2. **Course Table:**
   * CourseID (Primary Key)
   * Title
   * Description
   * InstructorID (Foreign Key)
   * Schedule
   * Prerequisites
   * EnrollmentCapacity
   * Status (Active, Inactive)
3. **Enrollment Table:**
   * EnrollmentID (Primary Key)
   * StudentID (Foreign Key)
   * CourseID (Foreign Key)
   * EnrollmentDate
   * Status (Enrolled, Waitlisted, Completed)
4. **Gradebook Table:**
   * GradeID (Primary Key)
   * StudentID (Foreign Key)
   * CourseID (Foreign Key)
   * AssignmentID (Foreign Key)
   * Grade
   * Comments
   * SubmissionDate
5. **Assignment Table:**
   * AssignmentID (Primary Key)
   * CourseID (Foreign Key)
   * Title
   * Description
   * DueDate
   * MaxPoints
6. **Communication Table:**
   * MessageID (Primary Key)
   * SenderID (Foreign Key)
   * ReceiverID (Foreign Key)
   * Subject
   * MessageContent
   * Timestamp
7. **Report Table:**
   * ReportID (Primary Key)
   * ReportName
   * GeneratedBy
   * GeneratedDate
   * ReportContent

**Practical:- 19**

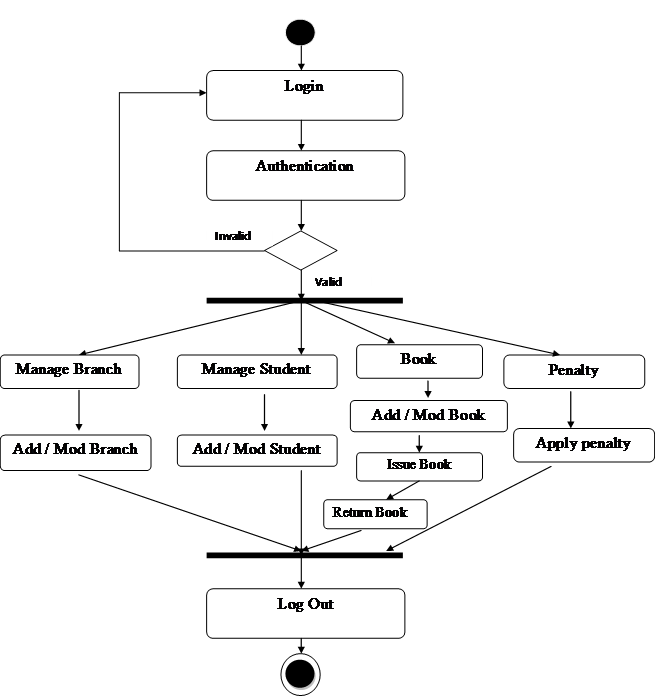
**Draw use case diagram, class diagram and an activity diagram for Library Management System. Write Objective, Module Summary and design Data Dictionary of System.**



Class Diagram



Activity Diagram



**Objective:** The objective of the Library Management System (LMS) is to automate and streamline the processes involved in managing library resources, including books, journals, multimedia materials, and patron information. This system aims to improve accessibility to library resources, enhance patron experience, optimize resource utilization, and facilitate efficient library operations.

**Module Summary:**

1. **Catalog Management Module:** This module allows librarians to catalog and manage library resources. Librarians can add new items to the library collection, update existing entries with metadata such as title, author, publication date, ISBN, and assign unique identifiers. They can also classify items based on subject, genre, or other criteria for easy retrieval.
2. **Patron Management Module:** Facilitates the management of patron information, including registration, membership renewal, and account management. Patrons can create accounts, borrow materials, renew items, and pay fines if applicable. Librarians can update patron records, track borrowing history, and manage membership privileges.
3. **Circulation Module:** Handles the circulation of library materials, including check-in, check-out, and renewal processes. Librarians can scan items and patron IDs to record transactions, manage due dates, and handle reservations and holds. This module also tracks item availability and notifies patrons of overdue items or holds.
4. **Acquisition and Ordering Module:** Manages the acquisition and ordering of new library materials. Librarians can create purchase orders, track orders from vendors, receive and process shipments, and update inventory records accordingly. This module helps maintain an up-to-date collection and facilitates budget management.
5. **Reporting and Analytics Module:** Generates reports on library usage, circulation statistics, collection inventory, and financial transactions. Librarians can analyze trends, identify popular items, assess resource demand, and make informed decisions regarding collection development and resource allocation.
6. **Interlibrary Loan Module:** Facilitates interlibrary loan services, allowing patrons to request materials from other libraries within a network or consortium. Librarians can process loan requests, coordinate resource sharing, and track borrowed items. This module expands access to materials beyond the library's own collection.

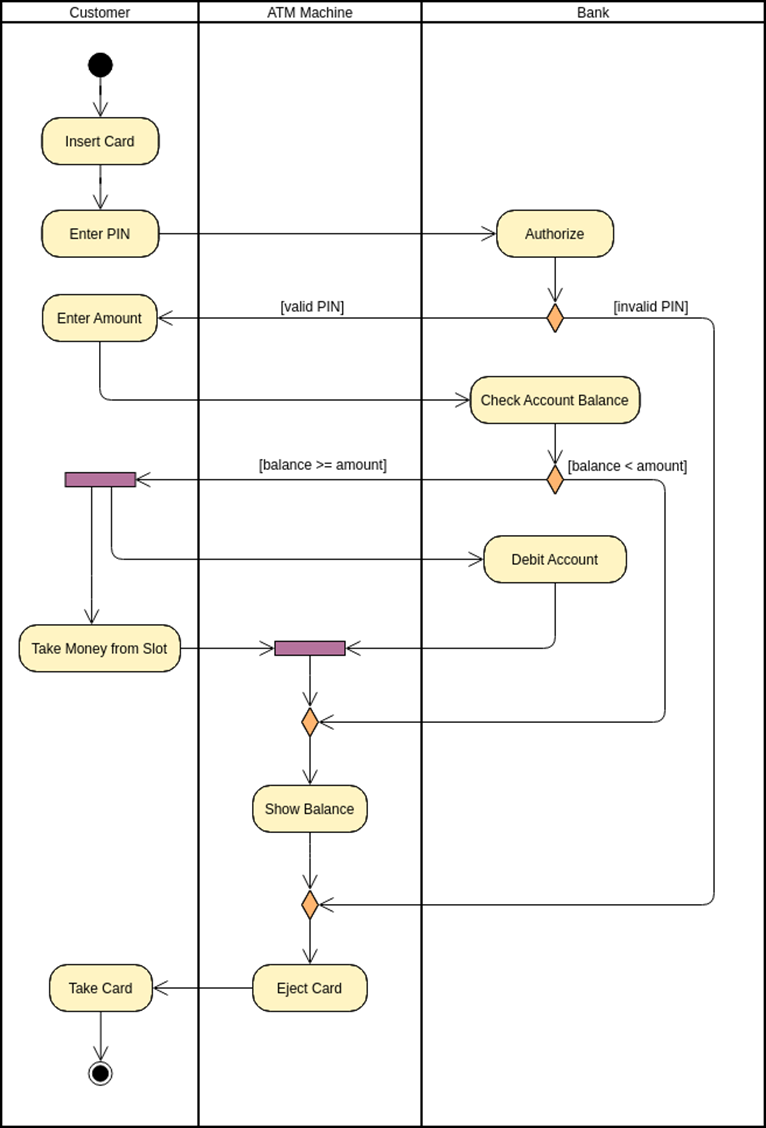
**Data Dictionary:**

1. **Book Table:**
   * BookID (Primary Key)
   * Title
   * Author
   * ISBN
   * Publisher
   * PublicationDate
   * Genre
   * AvailabilityStatus
2. **Patron Table:**
   * PatronID (Primary Key)
   * FirstName
   * LastName
   * Email
   * Phone
   * Address
   * MembershipStatus
3. **Transaction Table:**
   * TransactionID (Primary Key)
   * PatronID (Foreign Key)
   * BookID (Foreign Key)
   * TransactionType (Check-out, Check-in, Renewal)
   * TransactionDate
   * DueDate
   * ReturnDate
   * FineAmount
4. **Order Table:**
   * OrderID (Primary Key)
   * VendorID (Foreign Key)
   * OrderDate
   * ExpectedDeliveryDate
   * ActualDeliveryDate
   * TotalCost
5. **Vendor Table:**
   * VendorID (Primary Key)
   * VendorName
   * ContactPerson
   * Email
   * Phone
   * Address
6. **Report Table:**
   * ReportID (Primary Key)
   * ReportName
   * GeneratedBy
   * GeneratedDate
   * ReportContent

**Practical:-20**

**Prepare Activity diagram for banking system showing fork and join.**

**Write Objective, Module Summary and design Data Dictionary of System.**



**Objective:** The objective of the Banking System is to provide a robust platform for managing banking operations, including account management, transactions, and customer services. This system aims to facilitate secure and efficient banking services, ensure accuracy in financial transactions, and enhance customer satisfaction.

**Module Summary:**

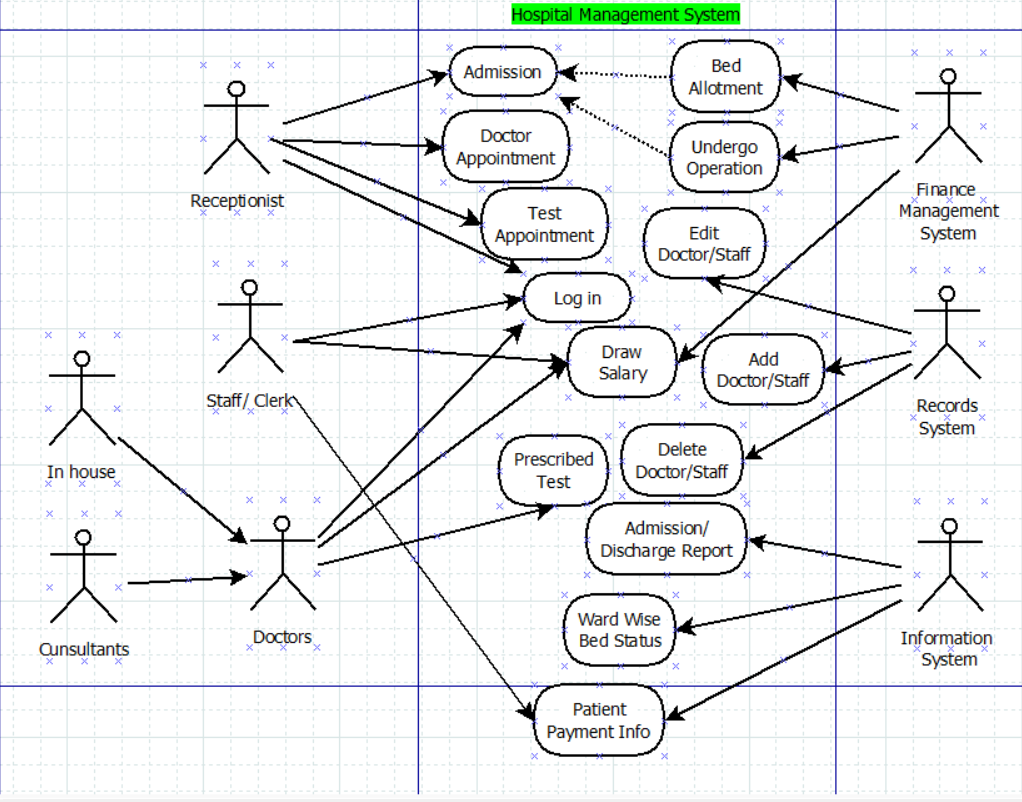
1. **Account Management Module:** This module handles account creation, modification, and deletion. It allows customers to open various types of accounts (e.g., savings, checking, fixed deposit) and manage account details such as account holders' information, account balance, and account status (active or inactive).
2. **Transaction Processing Module:** Facilitates the processing of various types of transactions, including deposits, withdrawals, transfers, and bill payments. This module ensures the integrity and security of financial transactions, verifies transaction details, and updates account balances accordingly.
3. **Customer Service Module:** Provides customer support services, including inquiries, account inquiries, and dispute resolution. Customers can seek assistance regarding their accounts, transactions, or general banking services through multiple channels such as phone, email, or in-person interactions.
4. **Loan Management Module:** Manages loan products offered by the bank, including loan application, approval, disbursement, and repayment. This module calculates loan eligibility, generates loan agreements, tracks repayment schedules, and manages overdue accounts.
5. **Risk Management Module:** Monitors and manages risks associated with banking operations, including credit risk, market risk, and operational risk. This module assesses risk exposure, implements risk mitigation strategies, and ensures compliance with regulatory requirements.
6. **Reporting and Analytics Module:** Generates reports and analytics on various aspects of banking operations, including account balances, transaction volumes, loan portfolios, and profitability. These reports provide insights for decision-making, performance evaluation, and regulatory compliance.

**Data Dictionary:**

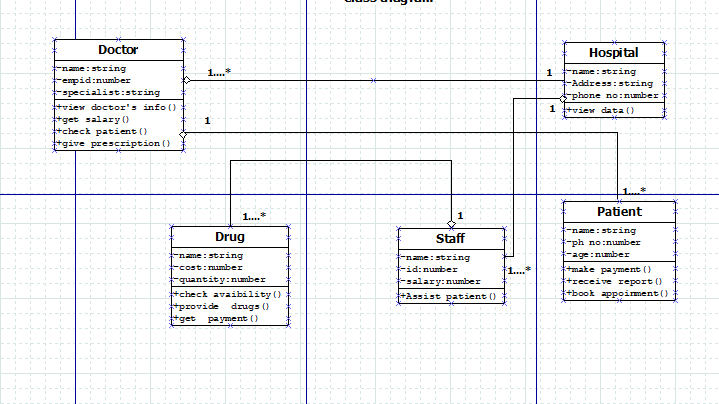
1. **Account Table:**
   * AccountID (Primary Key)
   * AccountType
   * AccountNumber
   * AccountHolderName
   * Balance
   * Status (Active, Inactive)
   * OpenDate
2. **Transaction Table:**
   * TransactionID (Primary Key)
   * AccountID (Foreign Key)
   * TransactionType (Deposit, Withdrawal, Transfer, Payment)
   * Amount
   * TransactionDate
   * Description
3. **Customer Table:**
   * CustomerID (Primary Key)
   * FirstName
   * LastName
   * Address
   * Phone
   * Email
4. **Loan Table:**
   * LoanID (Primary Key)
   * CustomerID (Foreign Key)
   * LoanType
   * LoanAmount
   * InterestRate
   * LoanTerm
   * DisbursementDate
   * Status (Approved, Disbursed, Repaid)
5. **Risk Table:**
   * RiskID (Primary Key)
   * RiskType
   * Description
   * MitigationStrategy
6. **Report Table:**
   * ReportID (Primary Key)
   * ReportName
   * GeneratedBy
   * GeneratedDate
   * ReportContent

**Practical-21**

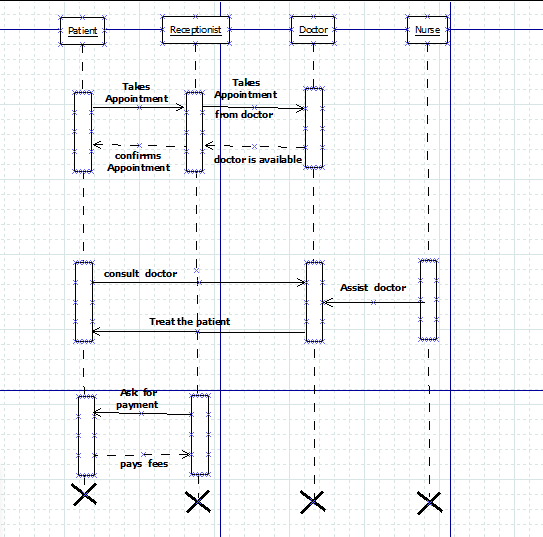
**Draw use case diagram, sequence diagram, class diagram, activity diagram and object Diagram for Hospital Management System. Write Objective, Module Summary and design Data Dictionary of System.**



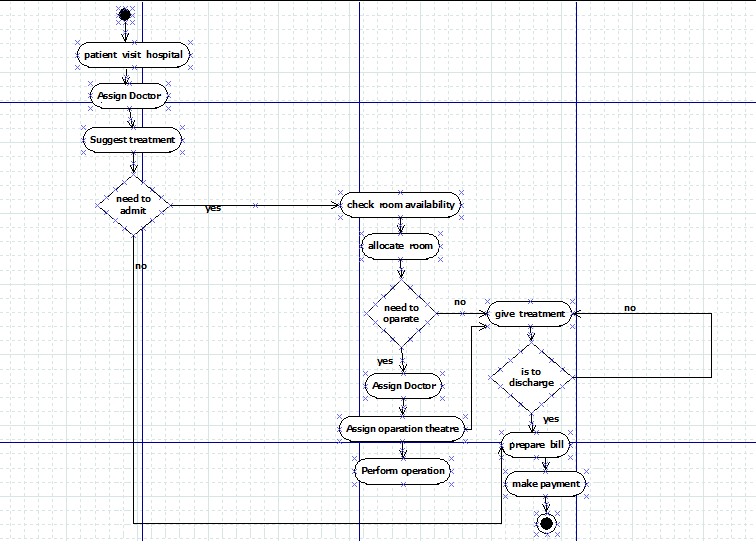
**Class Diagram**



Sequence Diagram



Activity Diagram



**Objective:** The objective of the Hospital Management System (HMS) is to automate and streamline various administrative and operational processes within a hospital or healthcare facility. This system aims to enhance efficiency, accuracy, and quality of patient care delivery, optimize resource utilization, and improve overall hospital management.

**Module Summary:**

1. **Patient Management Module:** This module handles patient registration, admission, and discharge processes. It maintains patient demographics, medical history, insurance information, and assigns unique identifiers for tracking patient records. It also manages bed allocation and room assignments based on availability and medical requirements.
2. **Appointment Scheduling Module:** Facilitates appointment scheduling for outpatient services, consultations, diagnostic tests, and procedures. Patients can request appointments through various channels (e.g., online portal, phone, in-person), and staff can schedule appointments based on availability of doctors, resources, and facilities.
3. **Medical Records Management Module:** Manages electronic health records (EHR) of patients, including clinical notes, test results, prescriptions, and treatment plans. This module ensures secure storage, retrieval, and sharing of medical information among healthcare providers while adhering to patient privacy regulations.
4. **Billing and Payment Module:** Handles billing processes, including generating invoices, processing insurance claims, and collecting payments for healthcare services rendered. It integrates with insurance systems to verify coverage and eligibility, calculates charges, and generates billing statements for patients.
5. **Inventory Management Module:** Manages inventory of medical supplies, pharmaceuticals, and equipment. It tracks stock levels, performs inventory audits, generates purchase orders, and manages suppliers. This module ensures availability of essential supplies and prevents stockouts or overstocking.
6. **Staff Management Module:** Handles staff scheduling, payroll processing, and performance evaluation. It maintains employee records, tracks work hours, calculates salaries, and manages leave requests. This module also facilitates training and professional development for staff members.

**Data Dictionary:**

1. **Patient Table:**
   * PatientID (Primary Key)
   * FirstName
   * LastName
   * DateOfBirth
   * Gender
   * Address
   * Phone
   * InsuranceDetails
   * AdmissionDate
   * DischargeDate
   * BedNumber
   * RoomNumber
2. **Appointment Table:**
   * AppointmentID (Primary Key)
   * PatientID (Foreign Key)
   * DoctorID (Foreign Key)
   * AppointmentDate
   * AppointmentTime
   * AppointmentType
   * Status (Scheduled, Completed, Cancelled)
3. **Medical Record Table:**
   * RecordID (Primary Key)
   * PatientID (Foreign Key)
   * DoctorID (Foreign Key)
   * Date
   * MedicalNotes
   * TestResults
   * Diagnosis
   * TreatmentPlan
   * Prescription
4. **Billing Table:**
   * BillID (Primary Key)
   * PatientID (Foreign Key)
   * Date
   * TotalAmount
   * InsuranceCoverage
   * PaymentStatus (Pending, Paid)
5. **Inventory Table:**
   * ItemID (Primary Key)
   * ItemName
   * Description
   * Quantity
   * UnitPrice
   * ExpiryDate
   * SupplierID (Foreign Key)
6. **Staff Table:**
   * StaffID (Primary Key)
   * FirstName
   * LastName
   * DateOfBirth
   * Gender
   * Position
   * Department