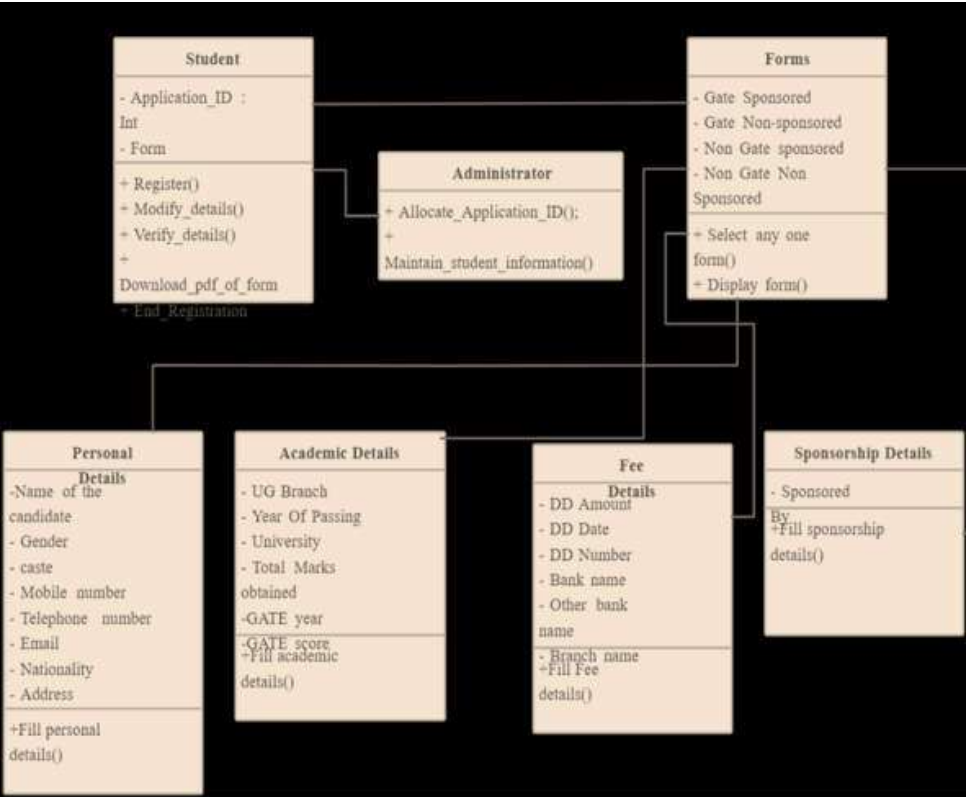
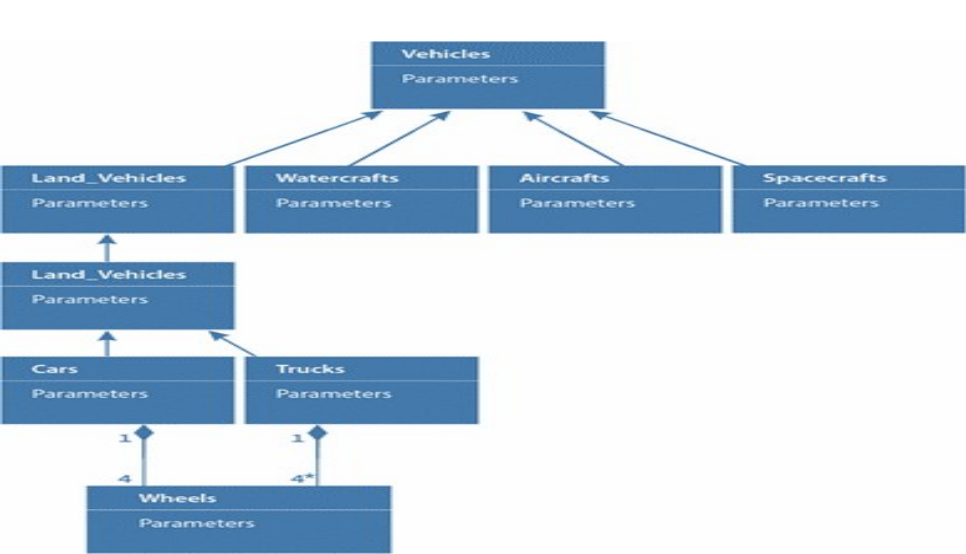


Class diagram for Admission System

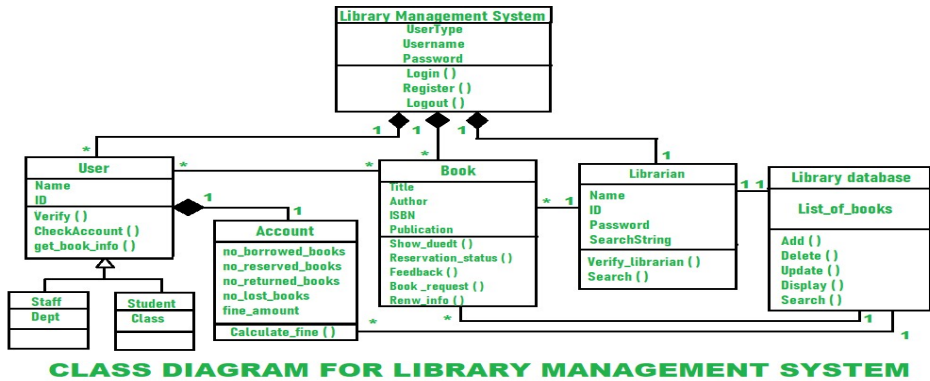


Class diagram for Vehicle domain

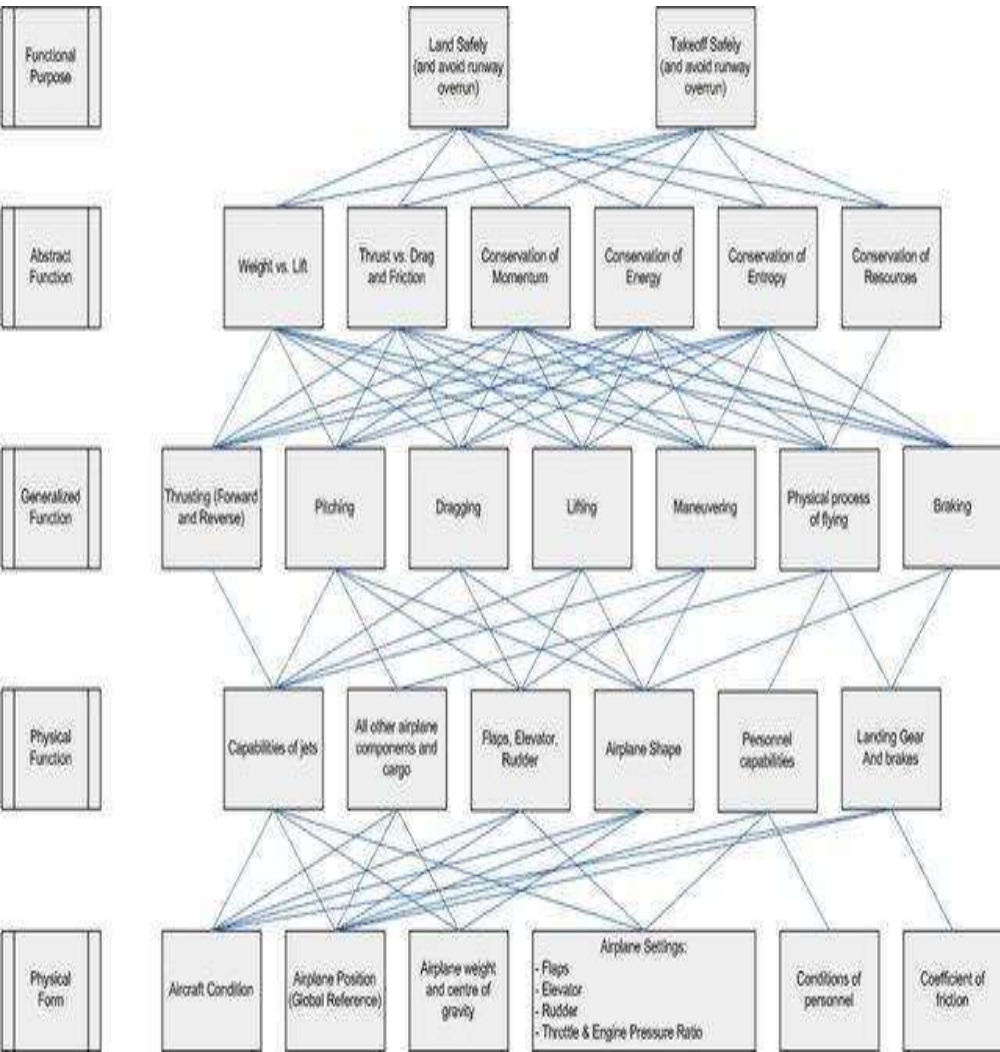


Class Diagram for Library Management System simply describes structure of Library Management System class, attributes, methods or operations, relationship among objects.

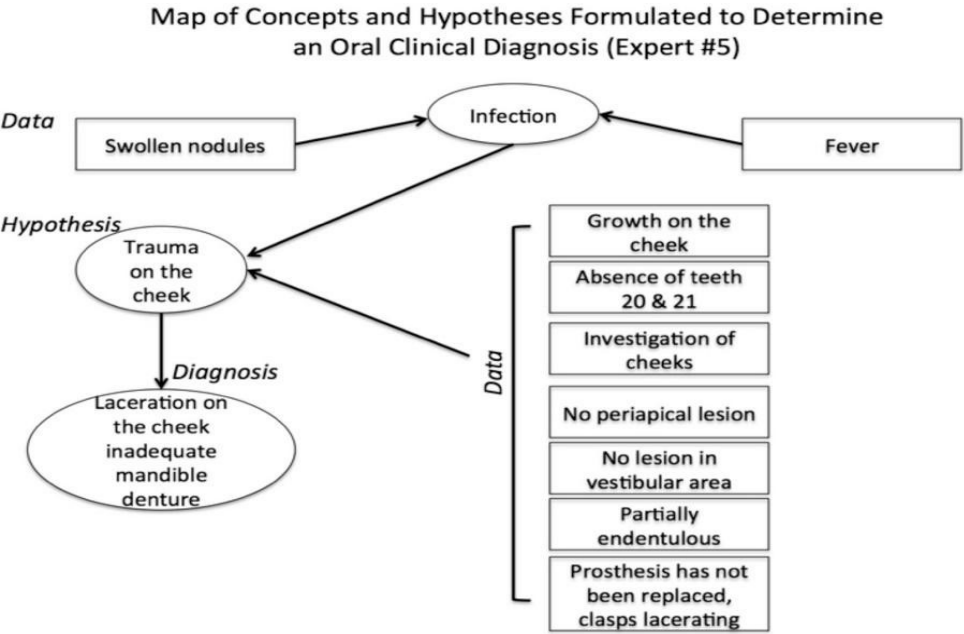
Class diagram for Library Management System



Work Domain Analysis : Aircraft Domain



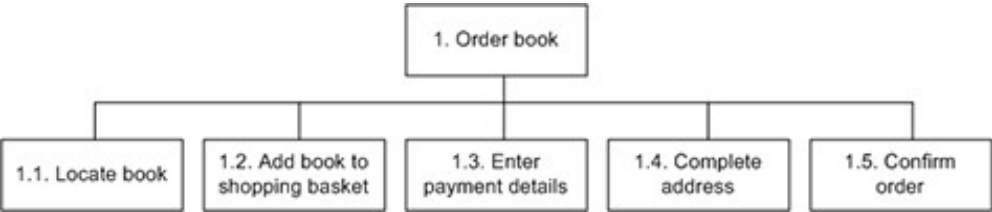
Cognitive Task Analysis using flow chart to determine an oral clinical diagnosis



Cite:-Journal of Dental Education- Wiley Online Library

b) **Hierarchical Task Analysis** Hierarchical task analysis (HTA) also referred as hierarchical decomposition is a widely used type of task analysis where a high-level task is decomposed into a hierarchy of subtasks.

Figure below —Hierarchical task analysis for ordering a book



In this hierarchical task analysis, task is broken into subtasks, expressing the relationships between the parent task and its subtasks through a numbering scheme. This hierarchical task analysis is very coarse from a user experience standpoint. It does *not* communicate anything about what is happening at the level of a user’s interaction with the system. However, it *does* give a clear understanding of the task’s high-level steps. A more complete task analysis would ultimately get down to the level of user interactions. To illustrate, Subtask 1.4, “Complete address,” would break down as follows:

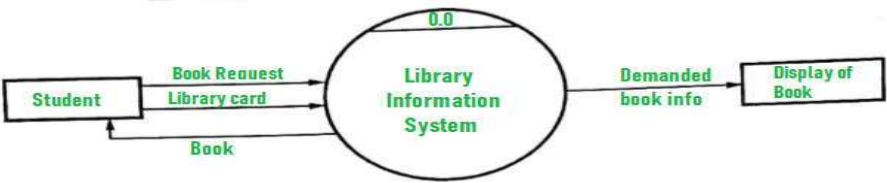
Notation

	Yourdon DeMarco	Gane & Sarson	SSADM	Yourdon and Coad
External Entity				
Process				
Data Store				
Data Flow				

Example of Data Flow Diagram for Library Management System

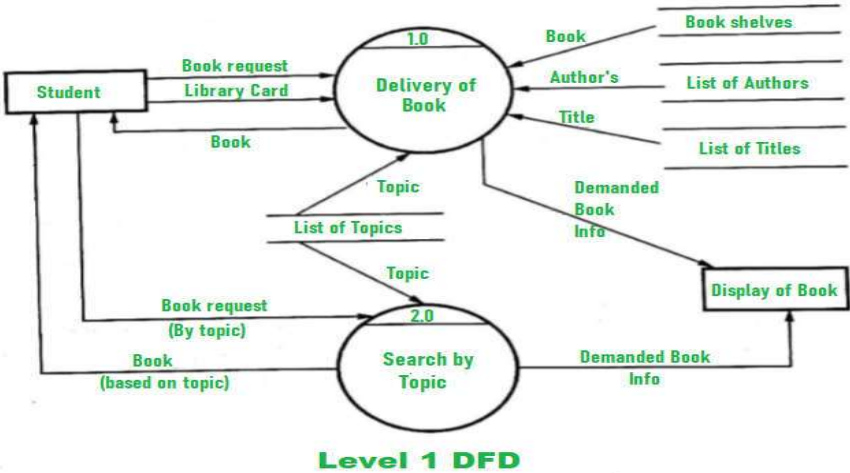
- Student entity will request for book , provide his library card and after authentication the library system will provide him with the book.

i. Level 0 DFD –

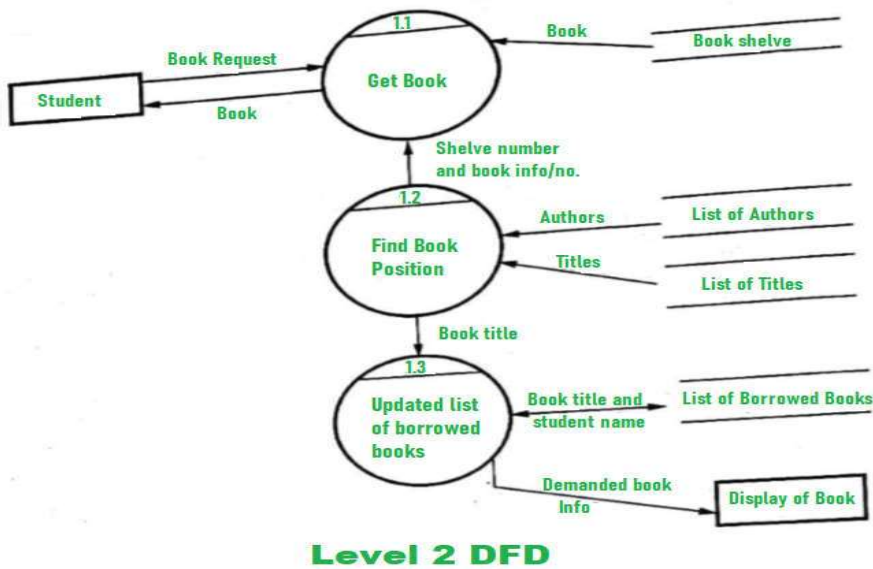


ii. Level 1 DFD –

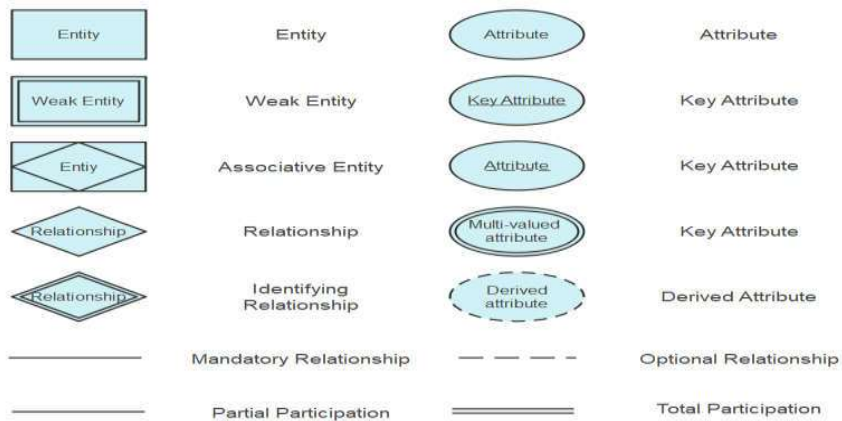
Book request process including delivery of book and book search process is being demonstrated here. **Data store** is used to represent this type of information.



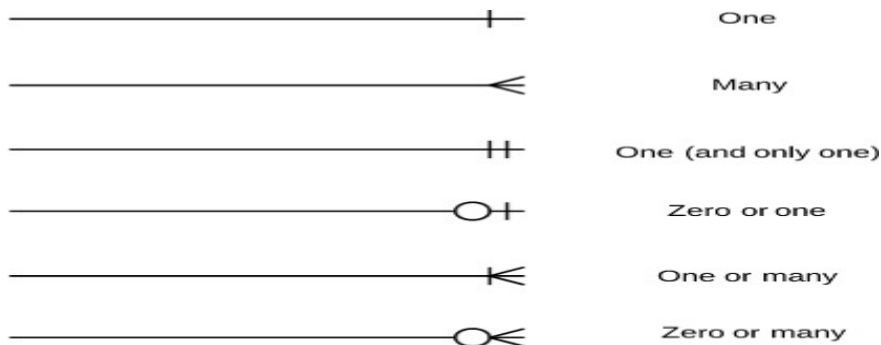
iii. Level 2 DFD – Detailing of Level 1 DFD



3) **Entity Relationship Diagram** : An entity relationship diagram (ERD) is a graphical diagram that depicts the relationships among people, objects, places, concepts or events within an information technology (IT) system.

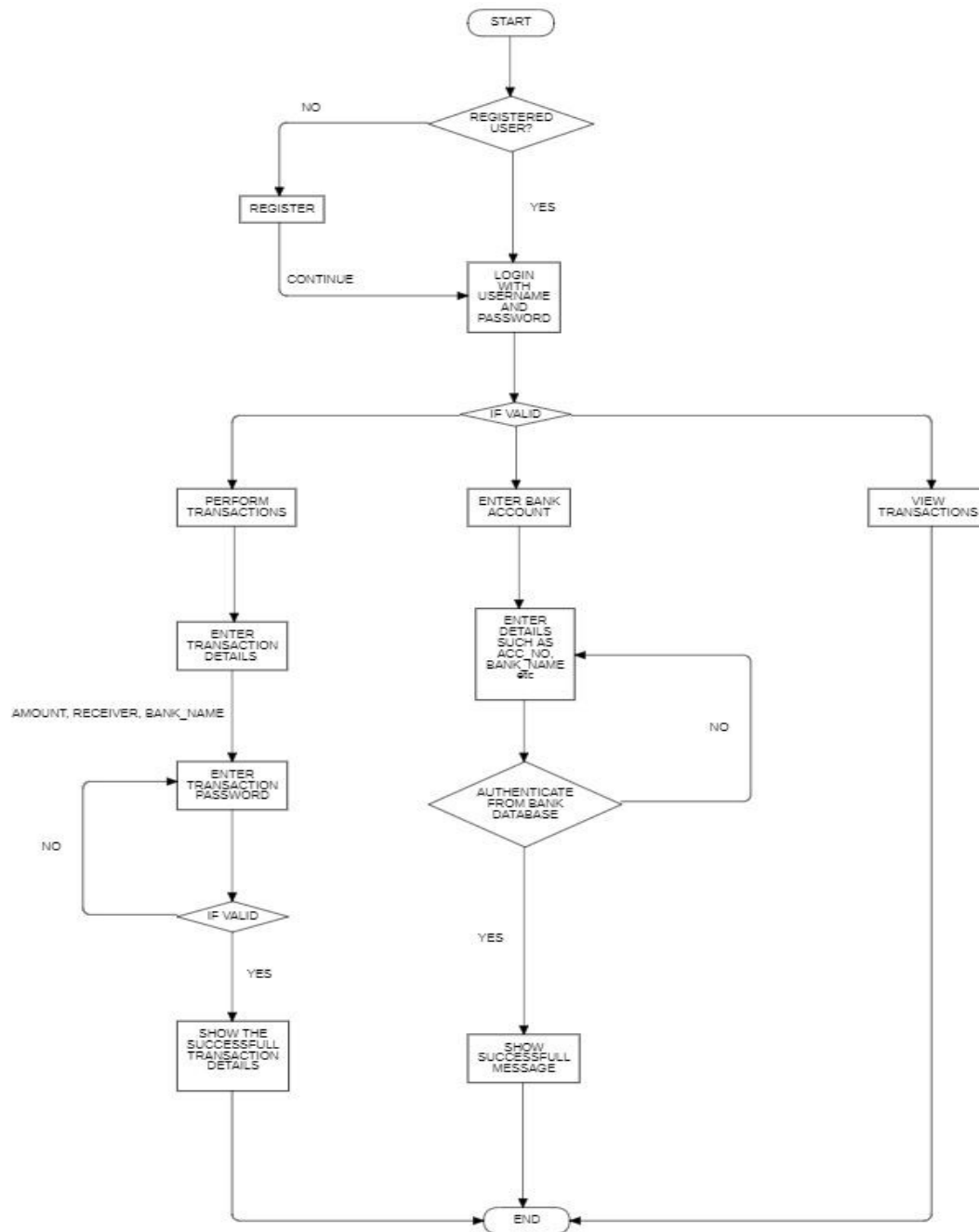


Cardinality in Entity Relationship Diagram



Flowchart:

Requirement gathering



Conclusion: We have successfully created and submitted the project proposal.

Self Learning Topics: Understand the user, types of users, requirement gathering techniques, contextual enquiry Include the business purpose and user needs.



The “specific application domain” can be for organic, semidetached or embedded systems which are systems that differ in terms of its complexity. For example Domain can range from College, banking to avionics, multimedia video games to applications within an MRI device. The goal of domain analysis is to find or create those classes that are broadly applicable, so that they may be reused.

Domain analysis is an ongoing umbrella software engineering activity that is not connected to any one software project. In a way, the role of a domain analyst is like the role of a master tool smith in a heavy manufacturing environment. The job of the tool smith is to design and build tools that may be used by many people doing similar but not necessarily the same jobs. The role of the domain analyst is to design and build reusable components that may be used by many people working on similar but not necessarily the same applications.

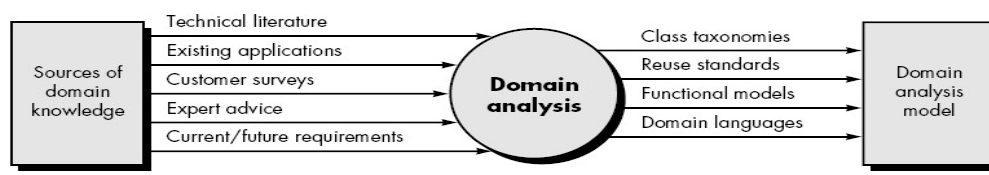
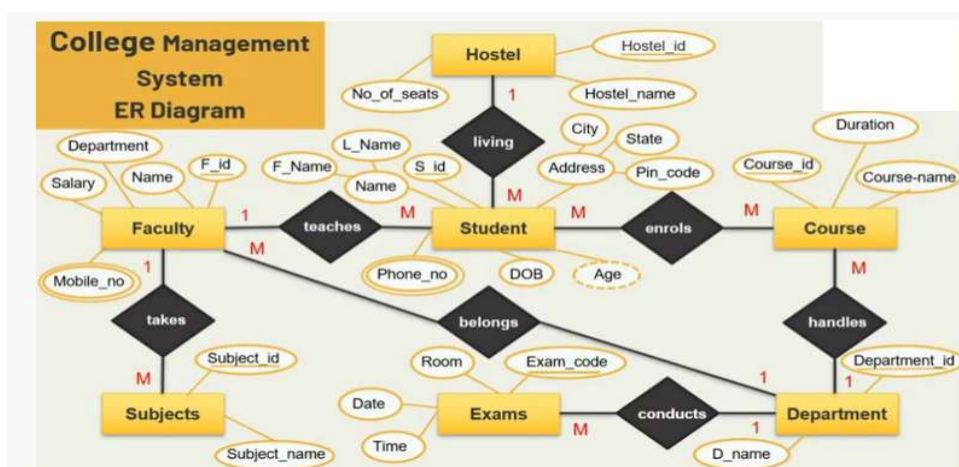


Figure illustrates key inputs and outputs for the domain analysis process.

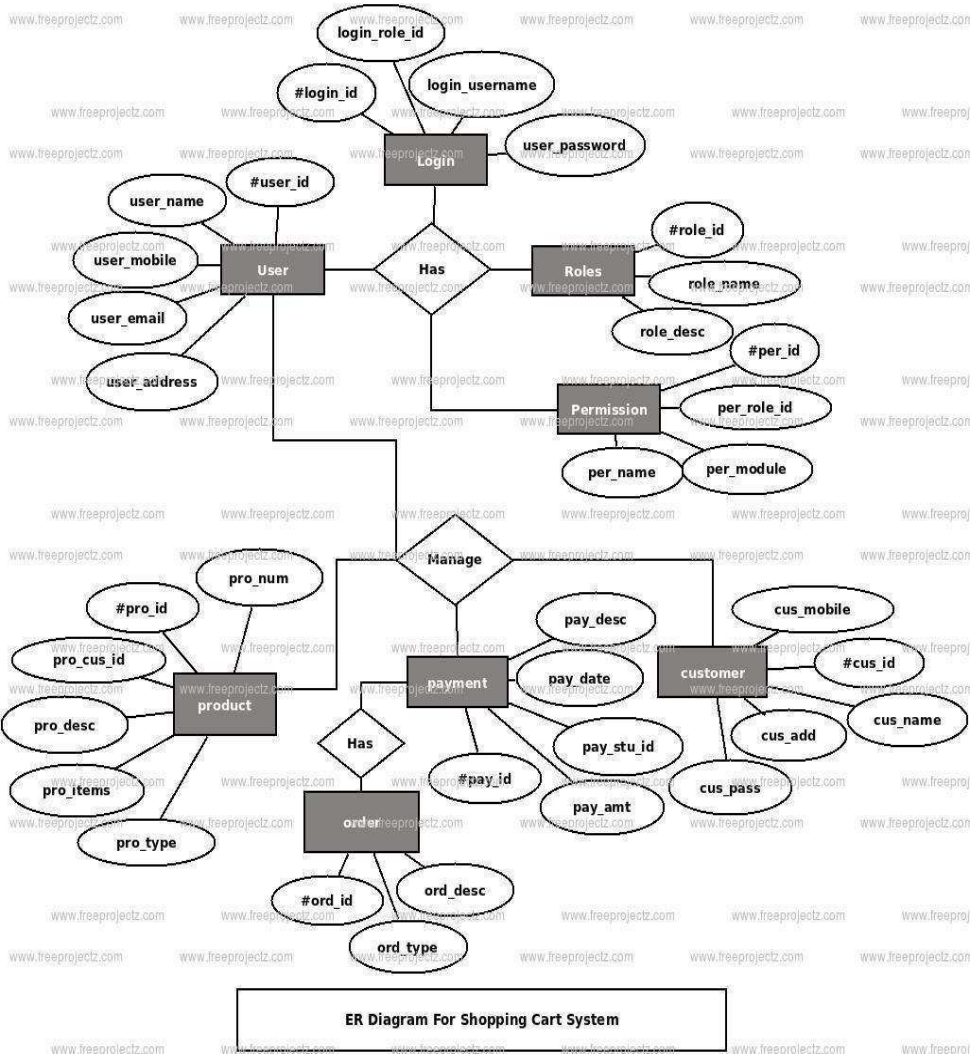
Domain Analysis methodologies include Unified modelling language diagrams Class diagram, Component diagram, Package diagram, etc. and Entity relationship diagrams

Examples

Entity Relationship Diagram for College Management System for understanding the college domain

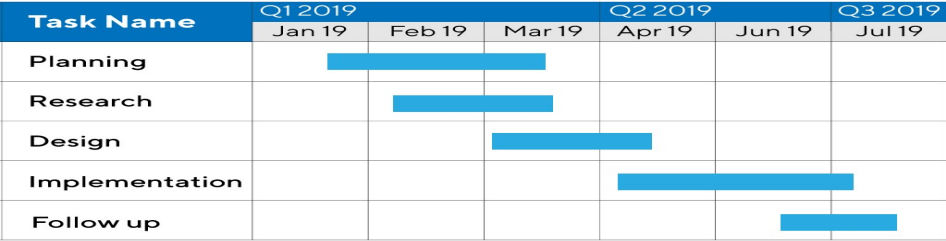


Example of Entity Relationship Diagram (ERD) for Shopping Cart System




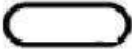





4) **Gantt Chart:** A [Gantt chart](#) is a bar chart that displays a detailed schedule of tasks defining linear schedule and task dependencies for a project.

Gantt Chart

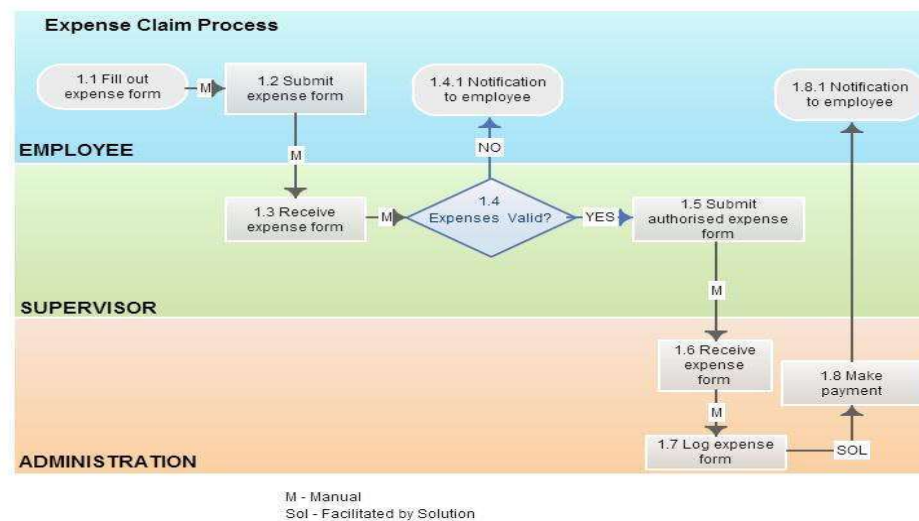


5) **Petri Nets :** . It is a technique for the description and analysis of concurrent /distributed systems. It is also known as a place/transition net. It is a mathematical model that describes the distributed system. It is a

Flow chart Notations

Symbol	Symbol Name	Description
	Flow Lines	Used to connect symbols
	Terminal	Used to start, pause or halt in the program logic
	Input/output	Represents the information entering or leaving the system
	Processing	Represents arithmetic and logical instructions
	Decision	Represents a decision to be made
	Connector	Used to Join different flow lines
	Sub function	used to call function

Example of Business process model using Flow chart to depict the organisation Claim Process.

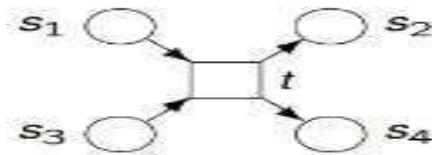




2) Data flow Diagram: -

Data flow diagram is to visualize the working processes within the organization viz manual or automated. It specifies the relationship between processes, data stores and external entities in business information system.

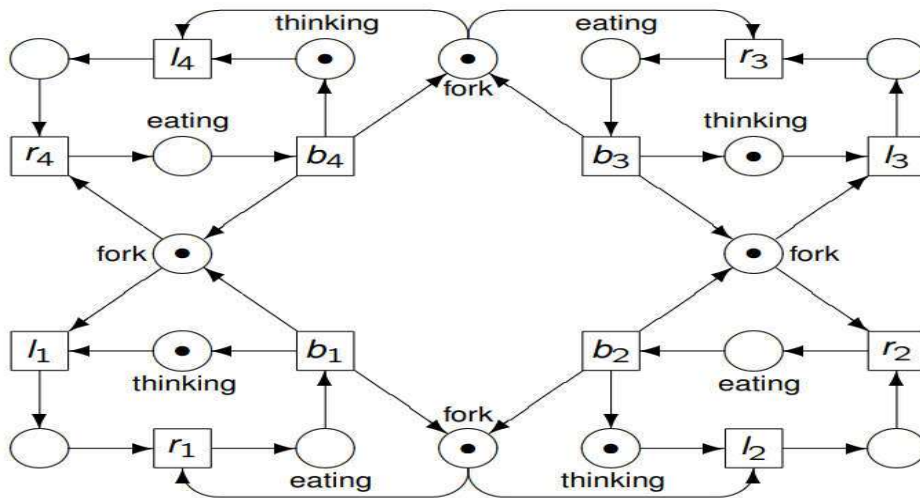
class of discrete event dynamic system .The basic idea is to describe state changes in a system with transitions.

Analysis

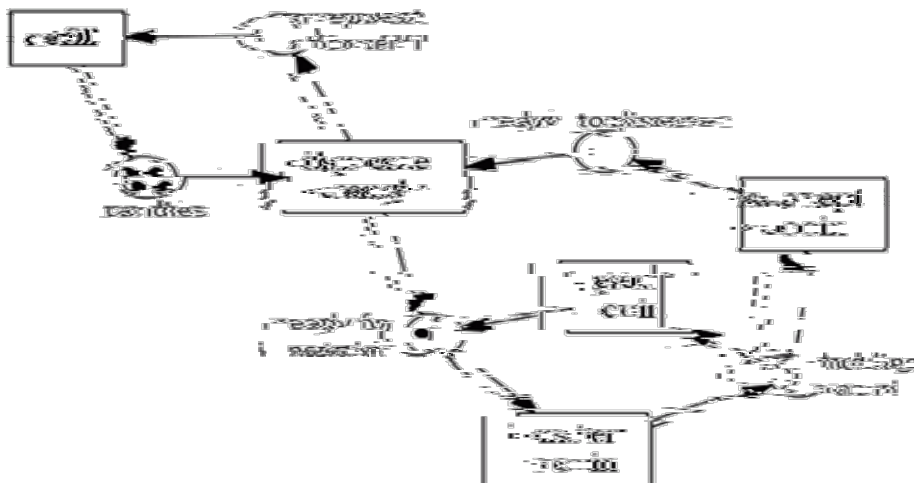


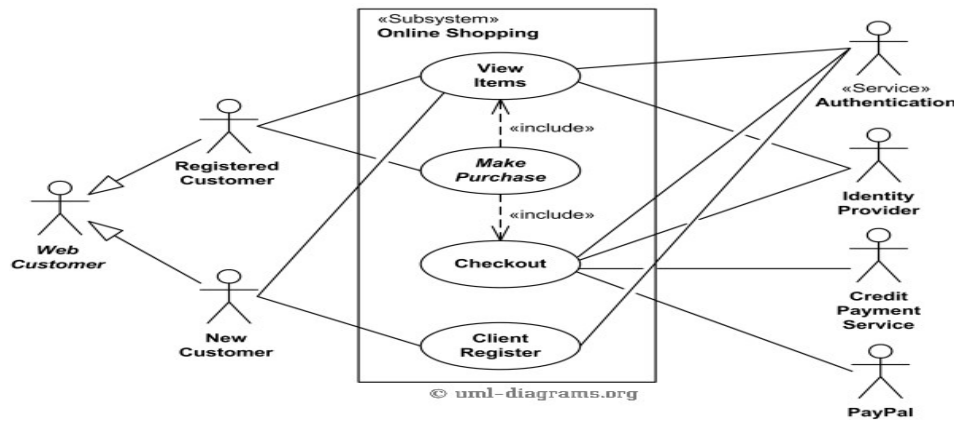
Petri nets contain **Places**  and **Transitions**  that may be connected by directed arcs. *Places symbolize states, conditions, or resources that need to be met/be available before an action can be carried out. Transitions symbolize actions*

Example: Petri Net for Dinning Philosopher Problem



Petri Net diagram for Candy dispensing machine



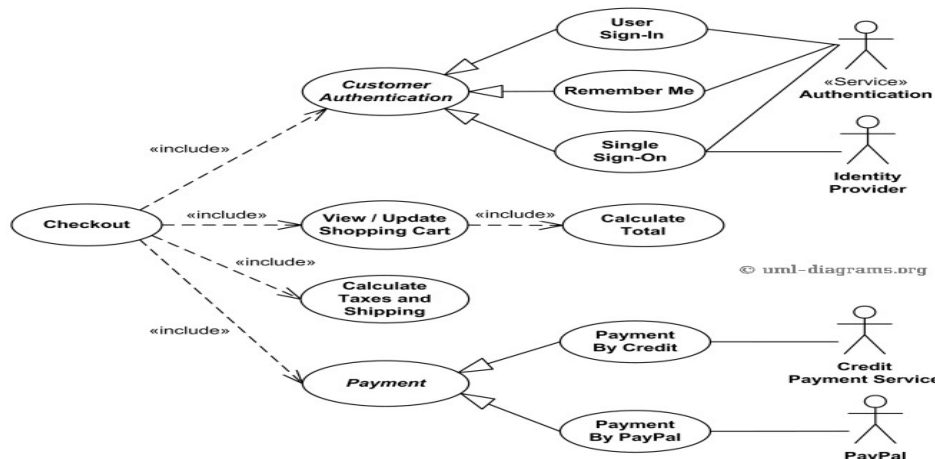


Online shopping UML use case diagram example - top level use cases.

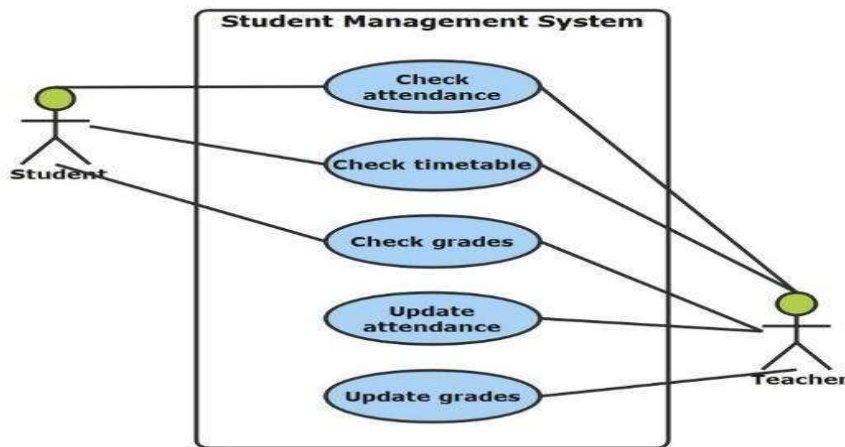
View Items use case is extended by several optional use cases - customer may search for items, browse catalog, view items recommended for him/her, add items to shopping cart or wish list. All these use cases are extending use cases because they provide some optional functions allowing customer to find item. **Customer Authentication** use case is included in **View Recommended Items** and **Add to Wish List** because both require customer to be authenticated. At the same time, item could be added to the shopping cart without user authentication.

Checkout use case includes several required uses cases. Customer on web should be authenticated. It could be done through user login page, user authentication cookie ("Remember me") or Single Sign-On (SSO). Web site authentication service is used in all these use cases, while SSO also requires participation of external identity provider.

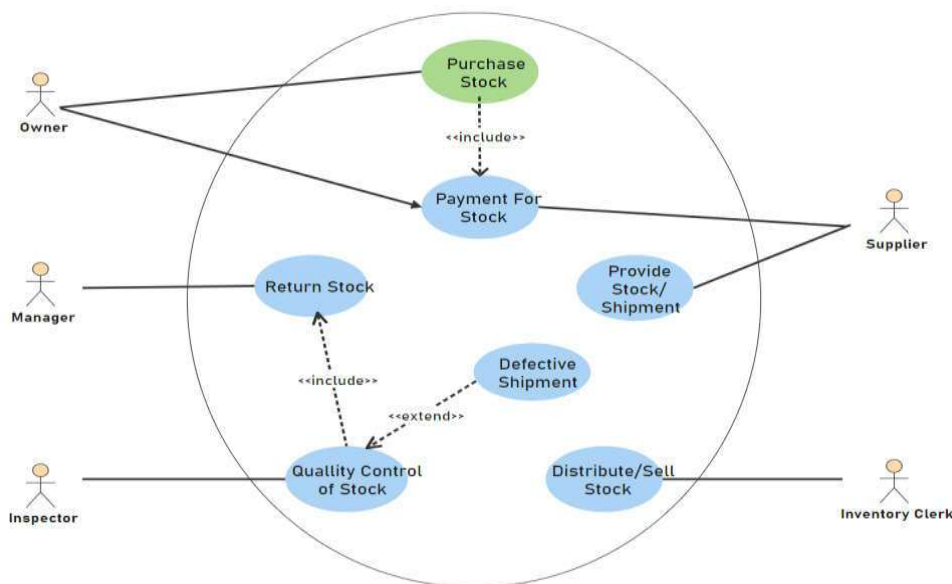
Checkout use case also includes **Payment** use case which could be done either by using credit card and external credit payment service or with PayPal.



A teacher can check attendance, check timetable, check test scores, update attendance and update the score. The student, on the other hand, can check attendance, timetable, and test score. The interactions of the student and the teacher are what sum up the student management use case diagram example.



USE CASE SCENARIO EXAMPLE 5- INVENTORY MANAGEMENT SYSTEM



This use case diagram example shows the various interaction by the inventory management UML case diagram. The main elements in these inventory management systems are the owner, manager, inspector, supplier, and inventory clerk.

The other use cases includes purchase stock, payment for stock, return stock, quality control of stock, defective shipment, distribute/sell stock and provide stock shipment. The use case has several interactions and builds up the inventory management system.