Star UML Introduction

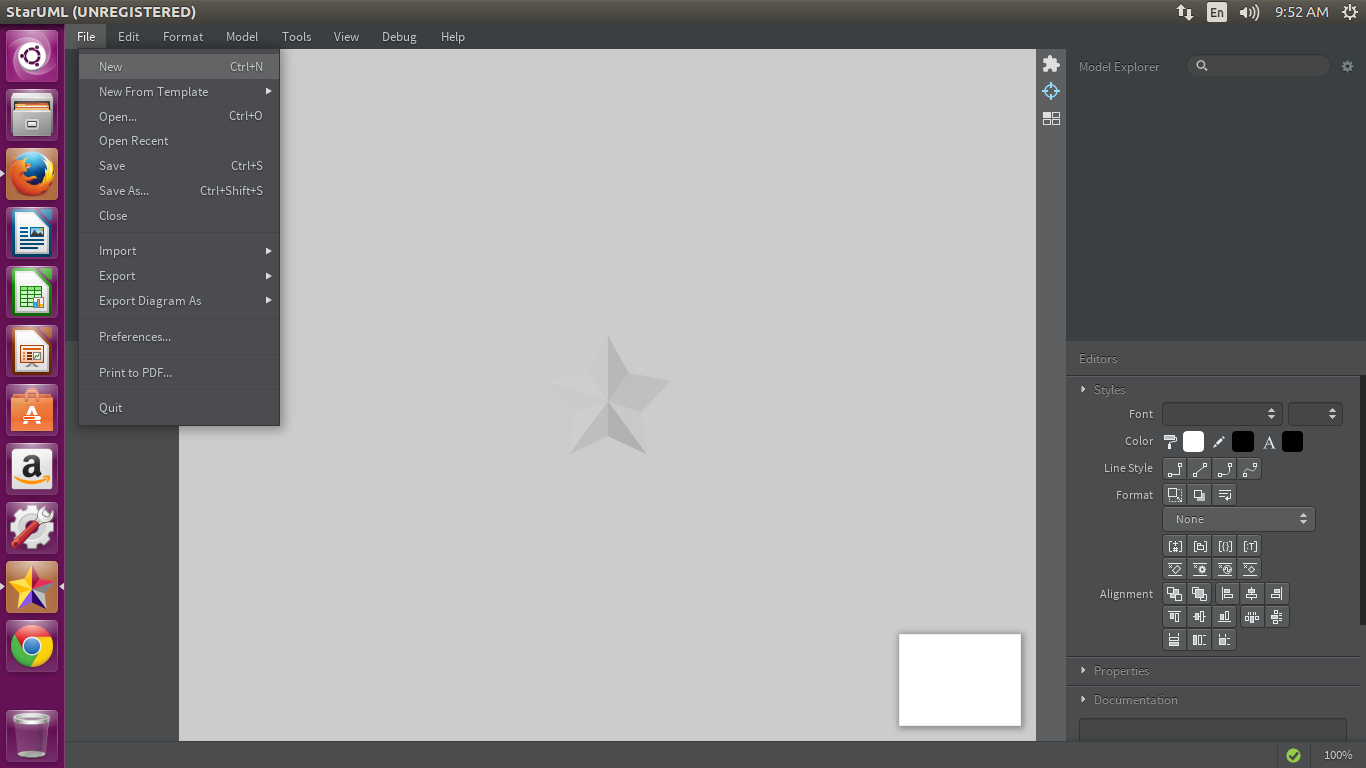
### [Download - StarUML](http://staruml.io/download)

staruml.io/download

Name of the software:Star UML

Version:-v2.8.0(http://staruml.io/download)

after downloading how to open the software



five windows in the interface of star uml

1) Working diagram Window- in this window the diagrams which we are able to draw

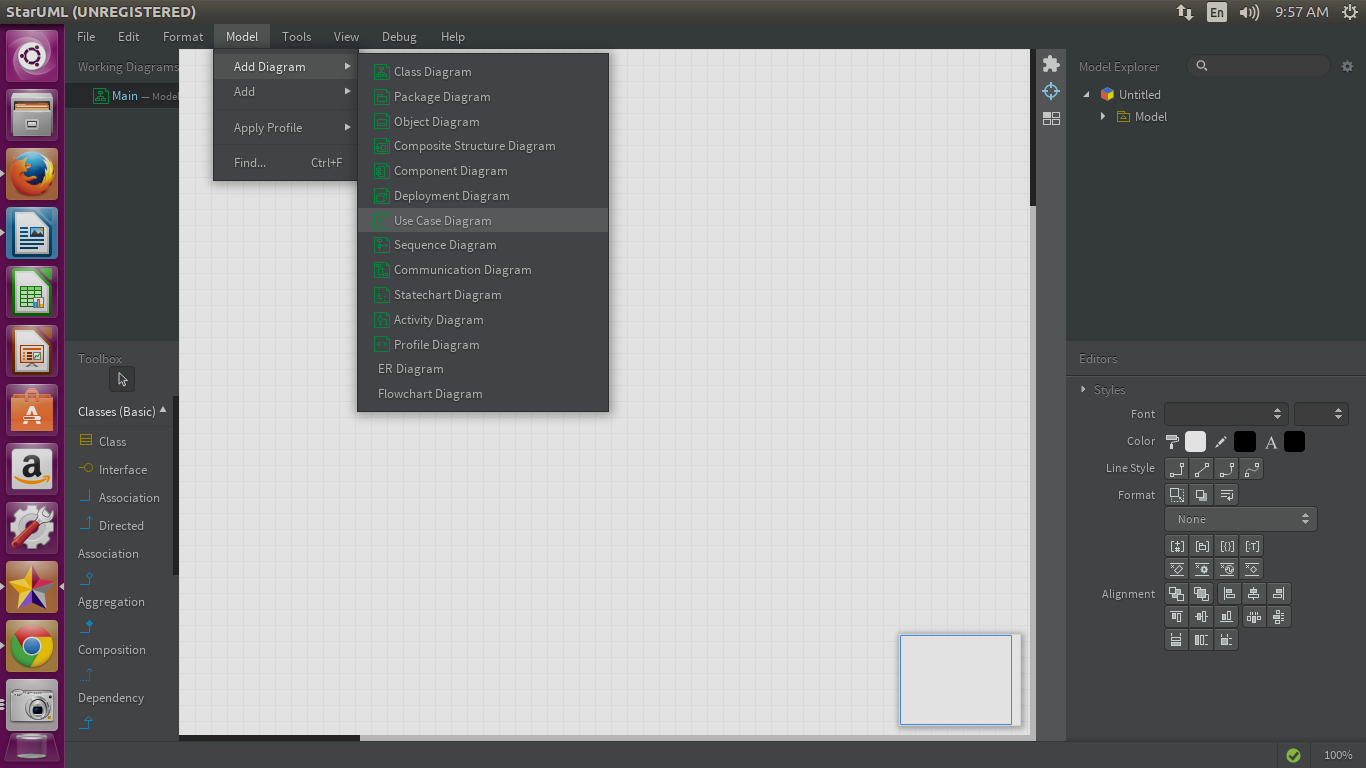
2) Tool box window- in this window the tools are displayed for the concerned diagrams

3) Main window- it is the location where we draw the diagrams

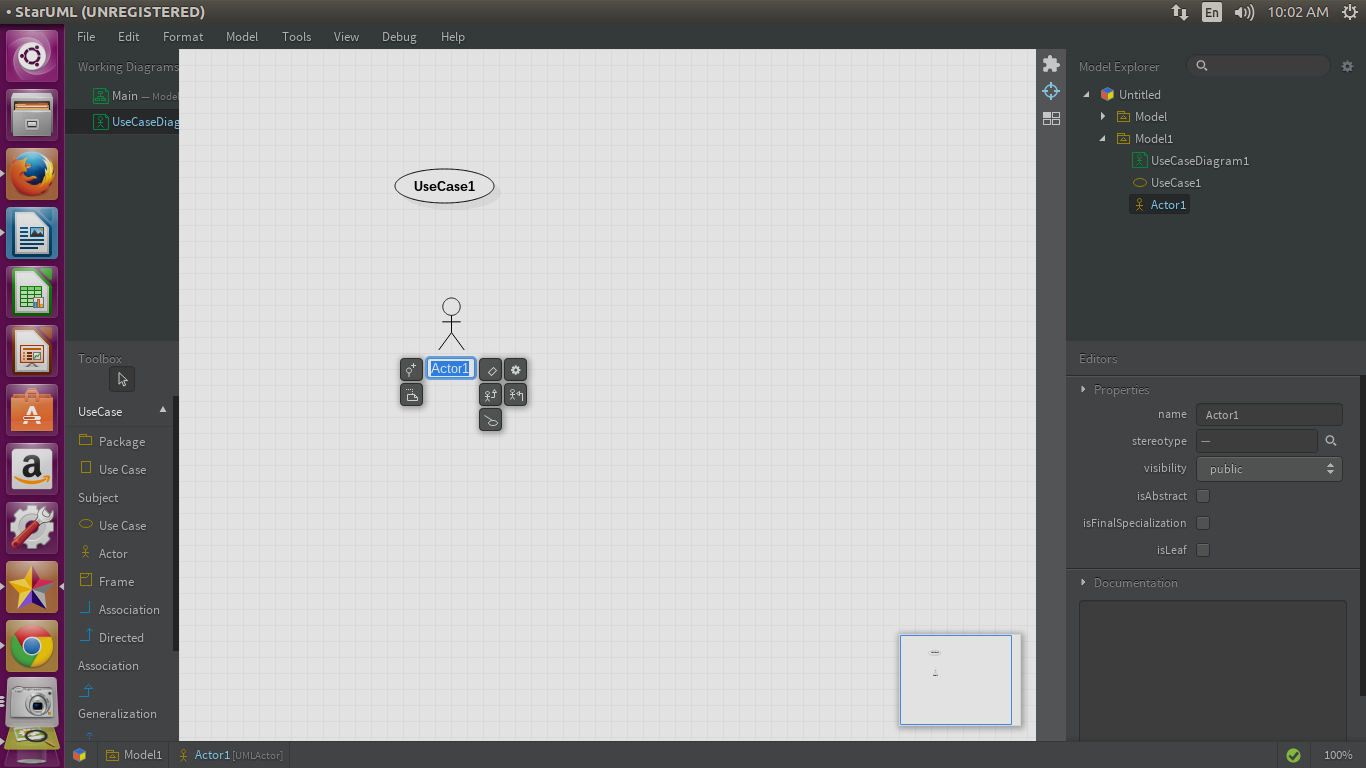
4) model explorer window- all the tools with their names are displayed

5) stlyes window- the apperance of each tool like color, font style etc are used from this window

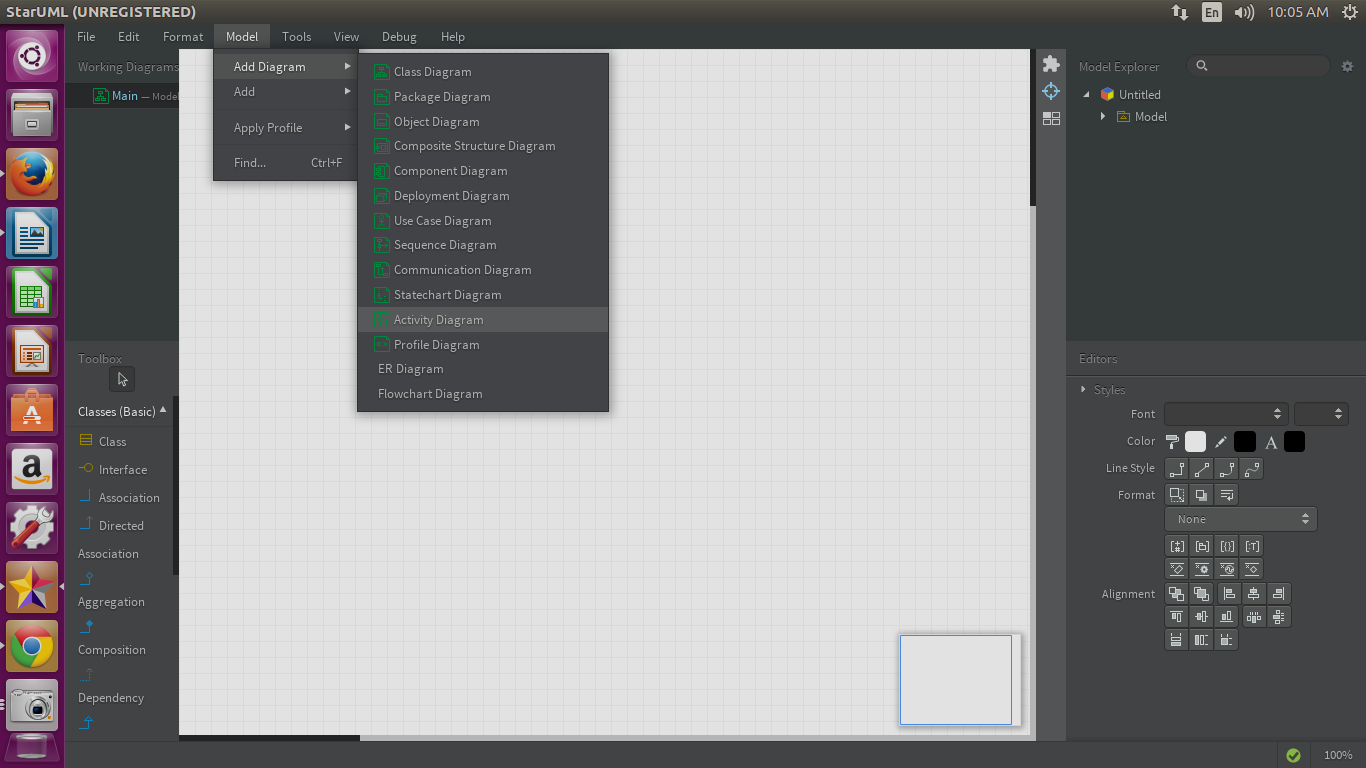
To draw the use case diagram the procedure is



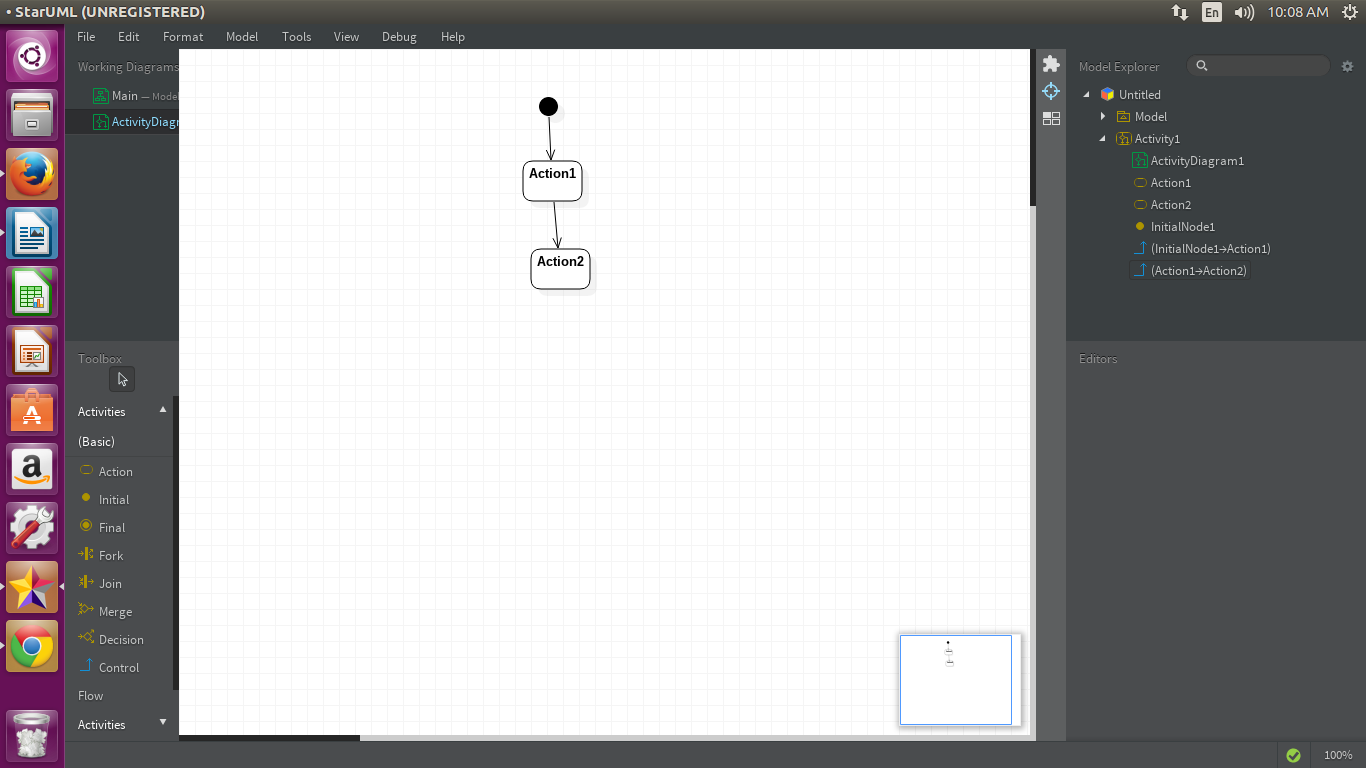
when we select the usecase diagram and placing the necessary tools that are required for drawing use case diagram



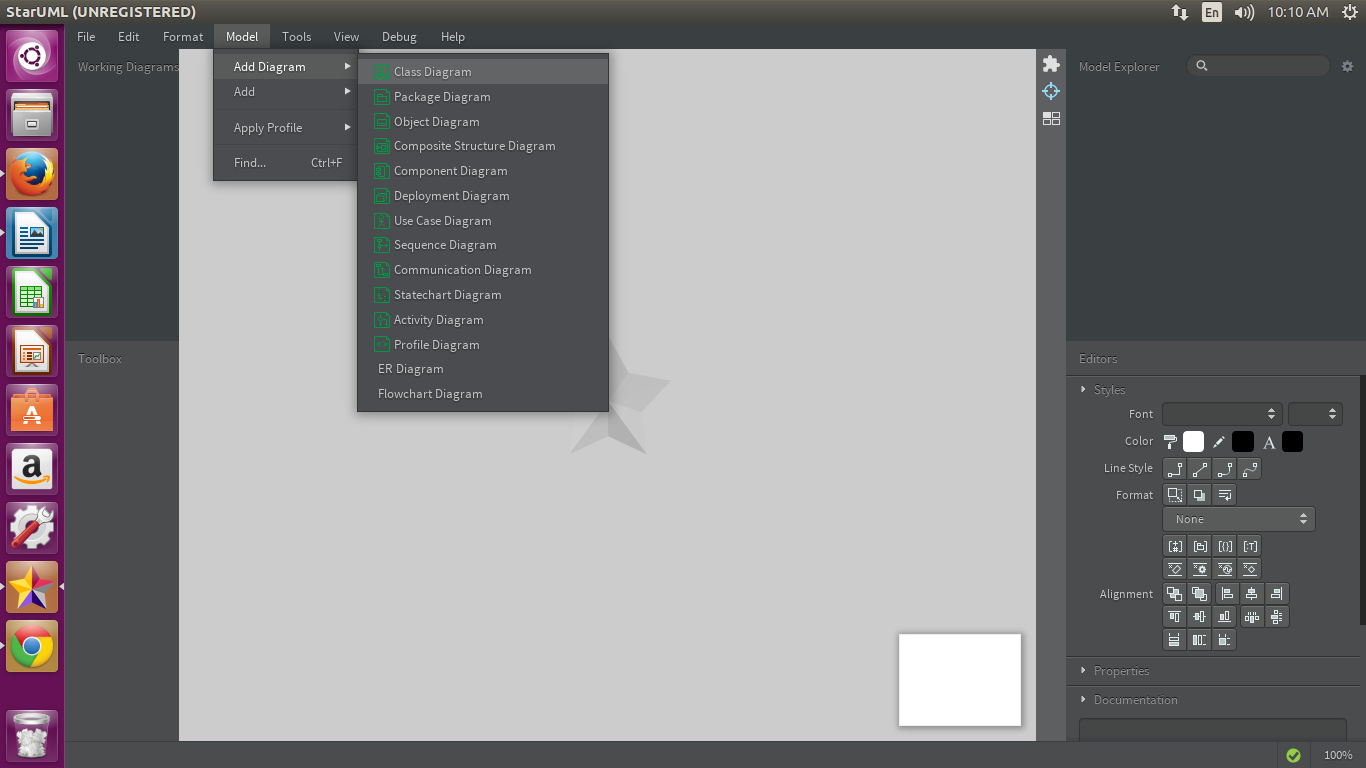
To draw activity diagram the procedure is

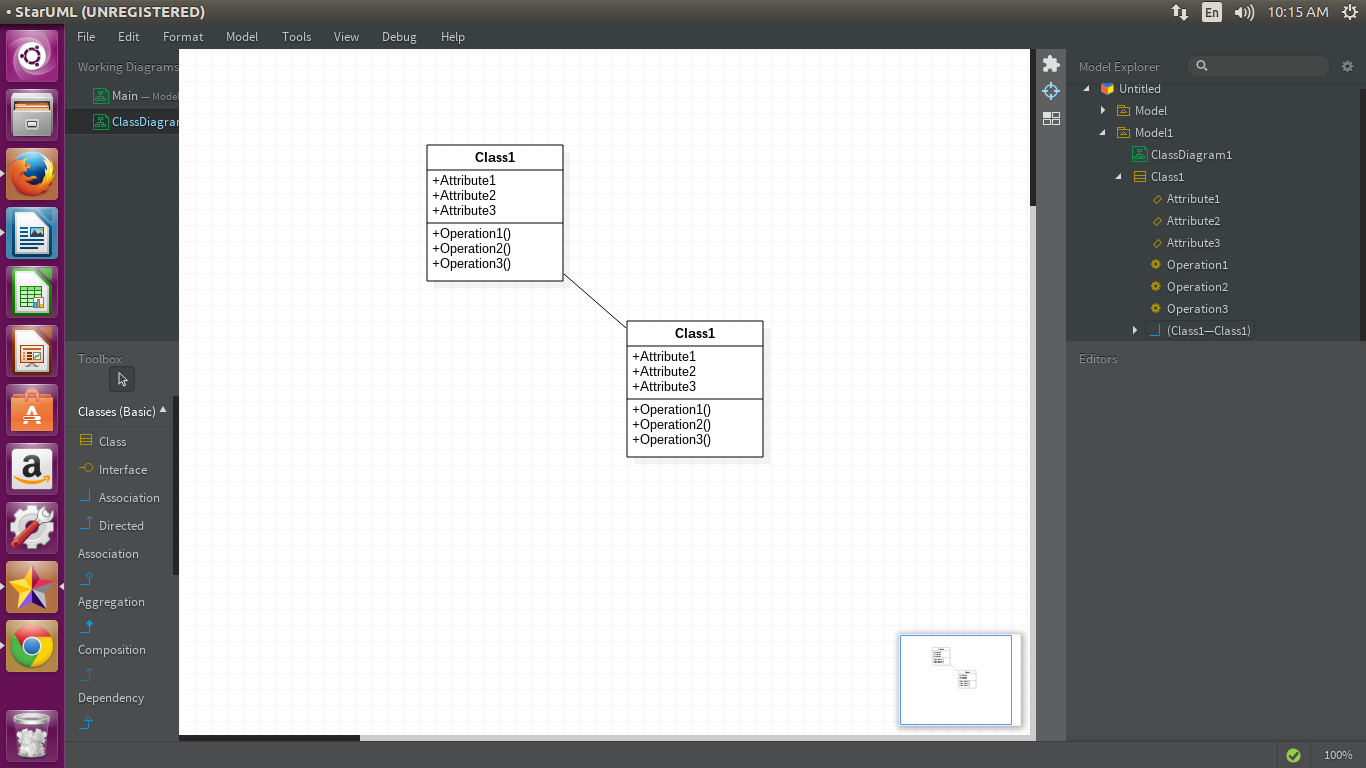


when we select the actvity diagram and placing the necessary tools that are required for drawing diagram

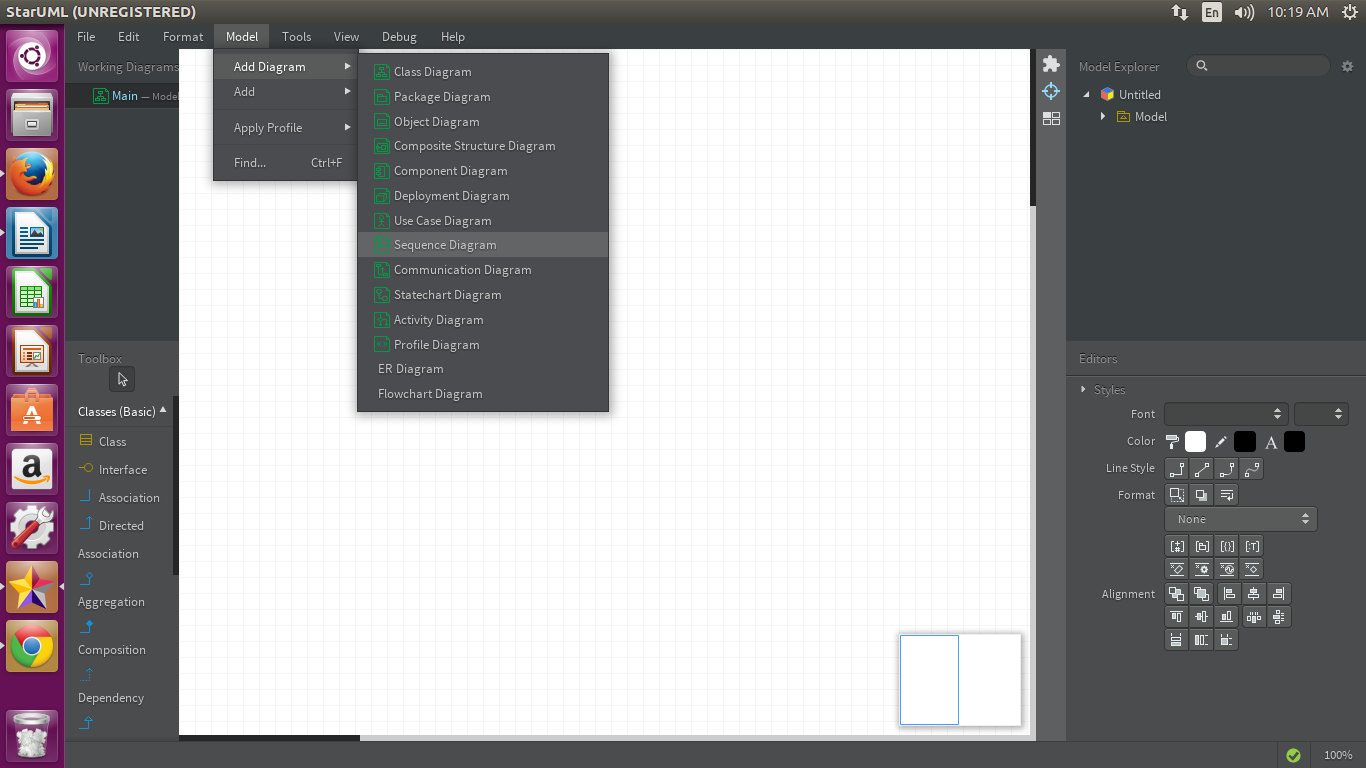


To draw class diagram the procedure is

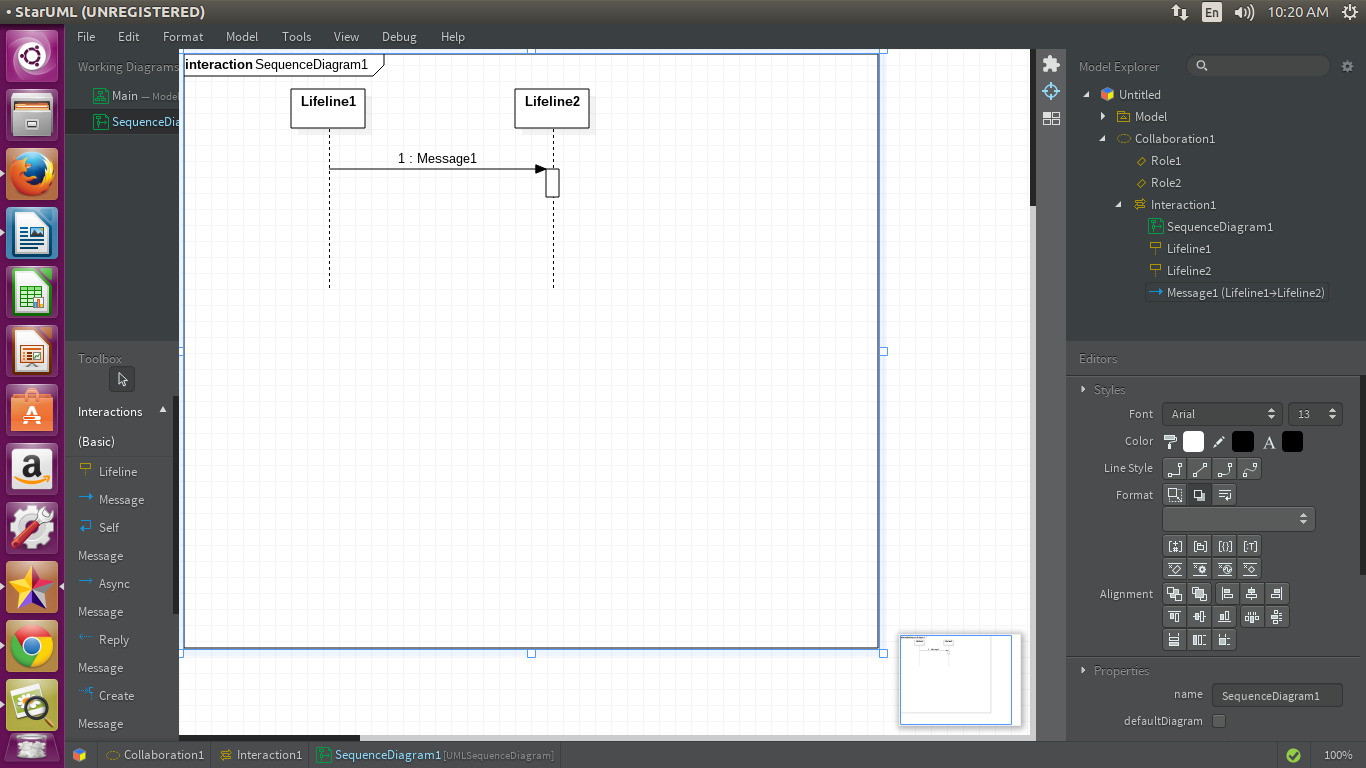
when we select the class diagram and placing the necessary tools that are required for drawing diagram



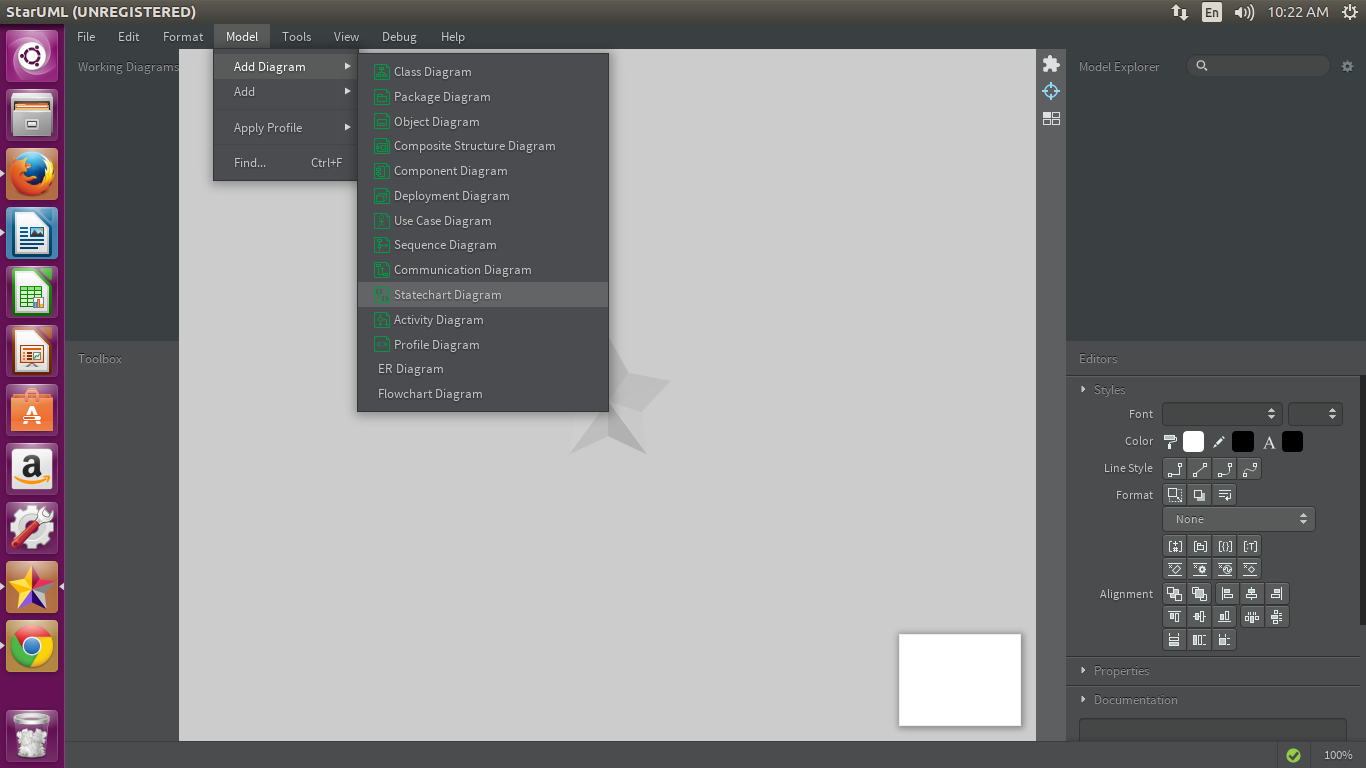
To draw sequence diagram the procedure is



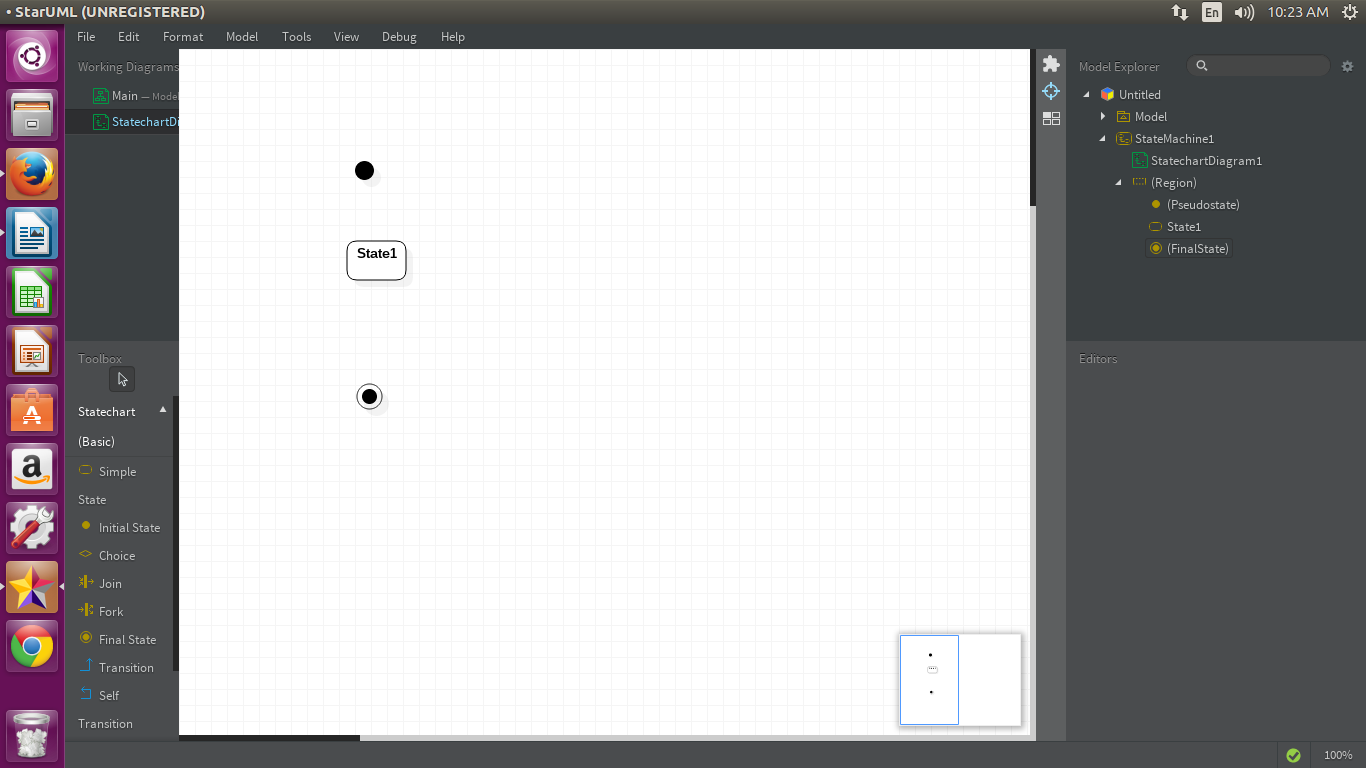
when we select the sequence diagram and placing the necessary tools that are required for drawing diagram



To draw state chart diagram the procedure is



when we select the statechart diagram and placing the necessary tools that are required for drawing diagram



Library Management Case Study

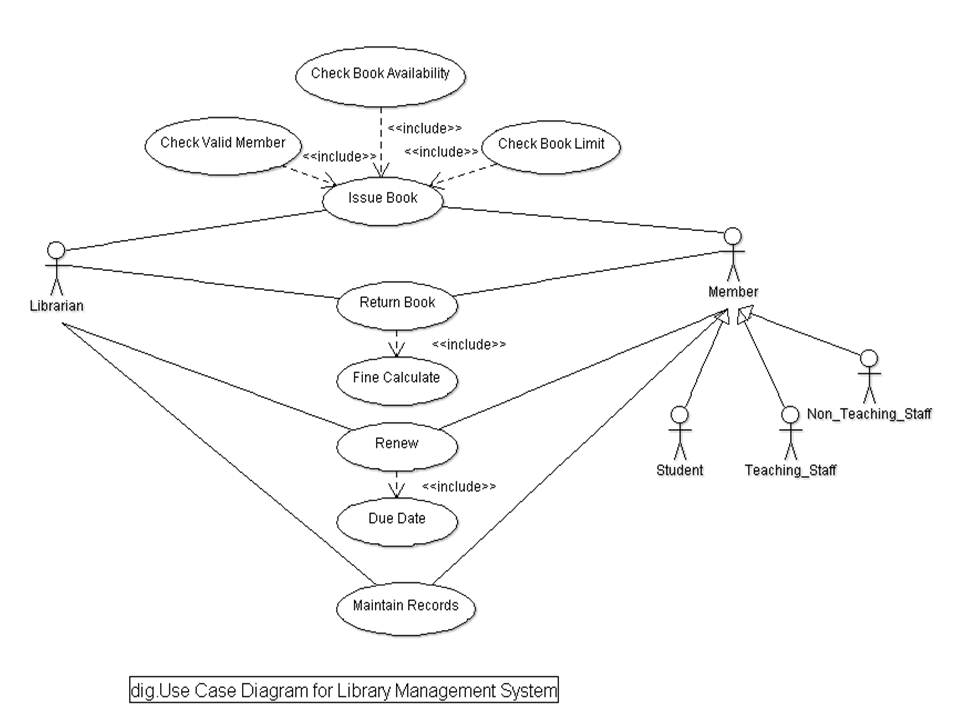
The library management system project is related to the storage of information regarding the library. Library is the place with the huge collection of books. It is place from where the students and the faculties issue the books for their reference purposes. But the maintenance of keeping the records of issuing and borrowing is difficult if you use a normal book as a registry. To make this task easier, the library management system will be very useful. It helps in maintaining the information regarding the issuing and borrowing of books by the students and the faculties. The library management system case study gives the case study of the library management system

There will be many answers for these case studies, These are some possible answers

Use case diagram:

**Procedure**

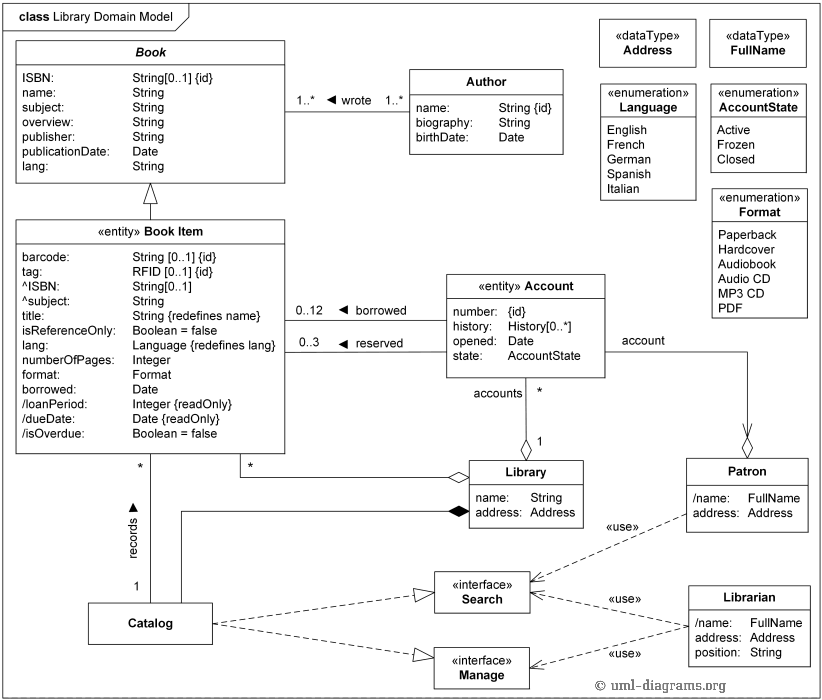
* Click on tools and create an actor namely ‘student’, ‘librarian’, ‘supplier’ using createactor option.
* Click on toolsand create nine use cases namely ‘submit id card’, ’verify id card’, ‘grant permission’, ‘request to issue book’, ‘issue book’, ‘return book’, ‘search for database’ and ‘check the availability’, ‘issue the book if available’ using createuse case option.
* Establish an association relationship between actor and usecase unidirectional by clicking on *tools* and *association* option from student actor to submit id card use case.
* Establish an association relationship between actor and usecase unidirectional by clicking on *tools* and *association* option from student actor to verify id card usecase.
* Establish an association relationship between actor and usecase unidirectional by clicking on *tools* and *association* option from student actor to request to issue book usecase.
* Establish an association relationship between actor and usecase unidirectional by clicking on *tools* and *association* option from student actor to issue book usecase.
* Establish an association relationship between actor and usecase unidirectional by clicking on *tools* and *association* option from student actor to return book usecase.
* Establish an association relationship between actor and usecase unidirectional by clicking on *tools* and *association* option from student actor to issue the book usecase.
* Establish an association relationship between actor and usecase unidirectional by clicking on *tools* and *association* option from supplier actor to check the book availability usecase.
* Establish an association relationship between actor and usecase unidirectional by clicking on *tools* and *association* option from supplier actor to issue the book is available usecase.
* Establish an association relationship between actor and usecase unidirectional by clicking on *tools* and *association* option from librarian actor to issue book usecase.
* Establish an association relationship between actor and usecase unidirectional by clicking on *tools* and *association* option from librarian actor to verify id card usecase.
* Establish an association relationship between actor and usecase unidirectional by clicking on *tools* and *association* option from librarian actor to grant permission usecase.
* Establish an association relationship between actor and usecase unidirectional by clicking on *tools* and *association* option from grant permission usecase to student actor.



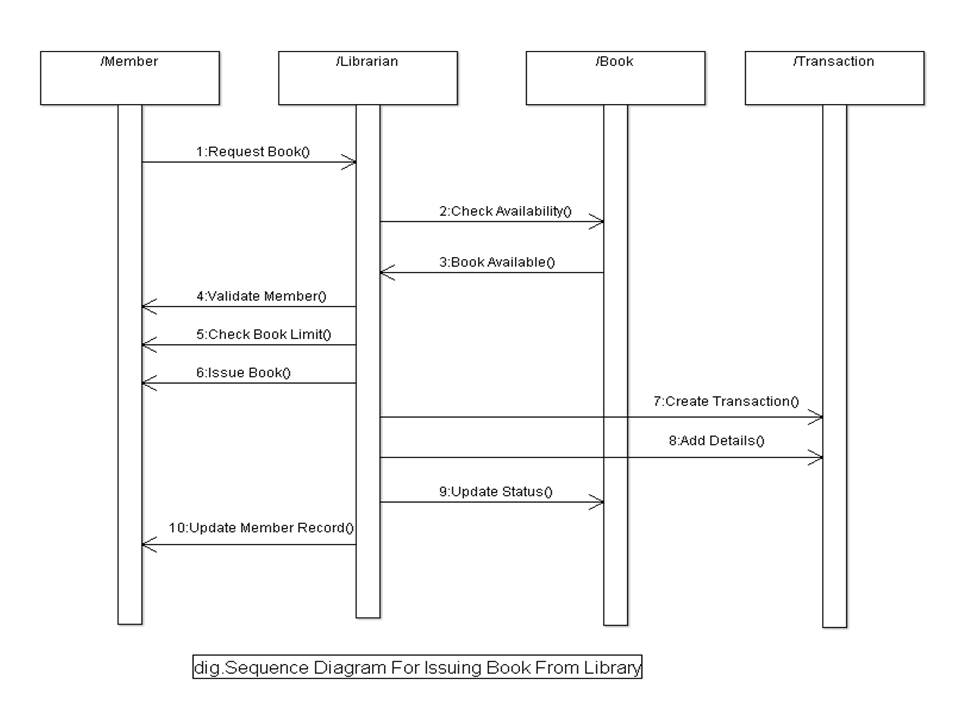
**Procedure**

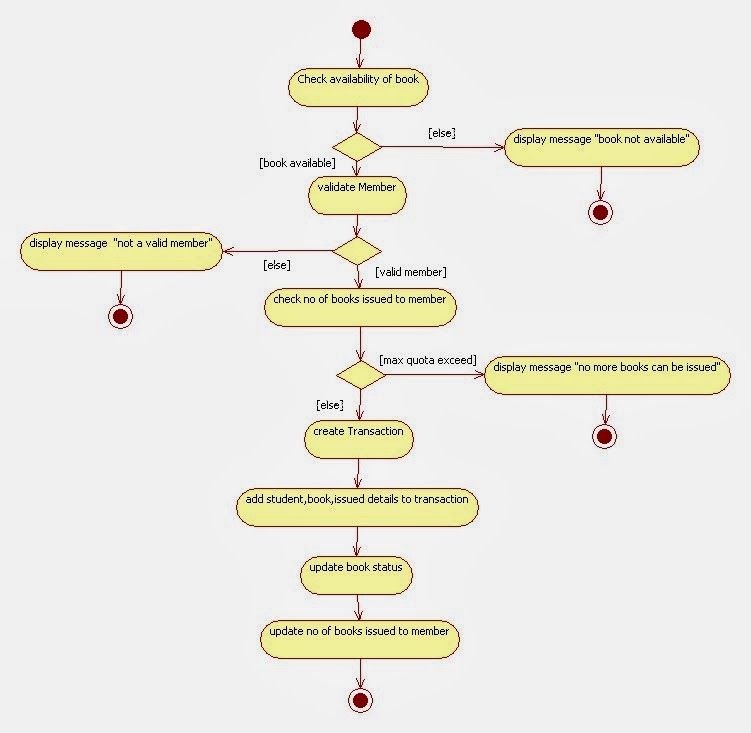
* Click on toolsand create eight classes namely ‘catalog’, ’librarian’, ’book’, ‘member’, ‘student’, ‘faculty’, ‘reference book’ and ‘issuable book’ using createclass option.
* Insert the attributes of catalog class i.e. name, created by and number of books as string type by opening the specifications window of catalog class
* Insert the operations to be performed in the Catalog class by clicking *operations* in its specification window. The operations are receiving add book with no return type, remove book having return type as double with no arguments.
* Insert the operations to be performed in the Librarian class by clicking *operations* in its specification window. The operations are add new book with no return type, update books having return type as double with no arguments.
* Similarly insert the operations of book class. The operations include add to catalog having return type as double with no arguments.
* Similarly insert the operations of member class. The operations include issue book, return book having return type as double with no arguments.
* Similarly insert the operations of student class. The operations include return book with no return type, check out book having return type as double with no arguments.
* Similarly insert the operations of faculty class. The operations include check out book with no return type with no arguments.
* Establish a *generalization* between member and student, member and faculty classes by clicking on *tools* and *generalization* option.
* Establish a *generalization* between book and reference book, book and issuable book classes by clicking on *tools* and *generalization* option.
* Establish an association relationship between the two classes unidirectional by clicking on *tools* and *association* option from catalog class to librarian class.
* Set multiplicity for cashier and shopper classes as ‘1..n’ by using properties.
* Name the association as ‘uses’ by opening its specification.
* Establish an association relationship between the two classes unidirectional by clicking on *tools* and *association* option from book class to member class.
* Set multiplicity for cashier and shopper classes as ‘1..n’ by using properties.
* Name the association as ‘uses’ by opening its specification.
* Establish an association relationship between the two classes unidirectional by clicking on *tools* and *association* option from member to librarian class.
* Set multiplicity for cashier and shopper classes as ‘1..n’ by using properties.
* Name the association as ‘gets updated by’ by opening its specification.
* Establish an association relationship between the two classes unidirectional by clicking on *tools* and *aggregation* option from book to catalog class.
* Set multiplicity for cashier and online debit card processing system classes as ‘1..n’ by using properties.
* Name the aggregation as ‘consists of’ by opening its specification

The class diagram is obtained.

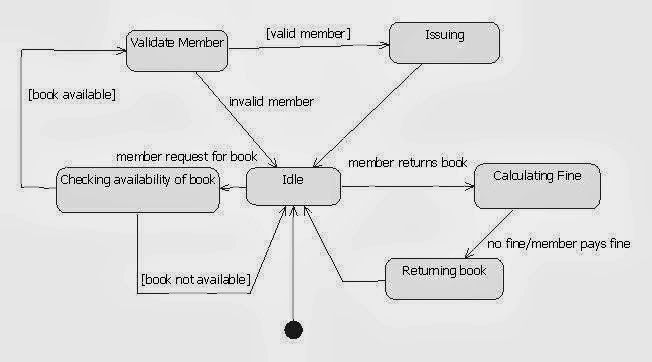


**Class diagram for the LMS**





**ACTIVITY DIAGRAM FOR LMS**



STATE CHART DIAGRAM FOR LMS