

USE CASE DIAGRAM

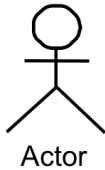
১। USE CASE DIAGRAM কি ? কেন ব্যবহার করা হয় ?

- ⇒ বিভিন্নভাবে সিস্টেমের সাথে যোগাযোগ স্থাপনের মাধ্যমই বলতে পারি **USE CASE**.
- ⇒ আরেকটু সহজ করে বলি । **USE CASE** সাধারণত আপনার পুরো সিস্টেমকে সারাংশ করে দেয় ছোট একটা DIAGRAM এর মাধ্যমে । যেন আপনার সিস্টেমের যেই ইউজার (যাদেরকে আমরা ACTORS বলি) একে অপরের সাথে অথবা সিস্টেমের সাথে সংযোগ স্থাপন করে কিছু SYMBOL ব্যবহার করে প্রকাশ করে ।
- ⇒ কীভাবে সিস্টেমের সাথে আমাদের যোগাযোগ হচ্ছে এবং সিস্টেমের বাহিরে ও ভিতরের কাজগুলো কীভাবে হচ্ছে তা আমরা জানতে এবং বুঝতে পারি এই **USE CASE DIAGRAM** ব্যবহার করে ।

কি লাভ হবে / পাবো USE CASE DIAGRAM করে ?

- ⇒ পুরো সিস্টেমের সাথে ইউজারের সম্পর্ক একটা চিত্রের মাধ্যমে উপস্থাপন করতে পারবে।
- ⇒ একটা সিস্টেমের কাজের প্রয়োজনীয়তা উল্লেখ করা।

USE CASE DIAGRAM করতে হলে আমাদের কিছু SYMBOL এর সাথে পরিচিত হতে হবেঃ



ACTOR:

ACTOR এর চিত্রটা সবসময় এমনই হবে এবং এইভাবেই প্রকাশ করতে হবে।

এই **ACTOR** এর কাজ হলো যে পুরো সিস্টেমের সাথে যোগাযোগ রাখে। একজন **ACTOR** একটা মানুষ হতে পারে, একটা সংগঠন হতে পারে, অথবা এমন কেউ হতে পারে/ হবে যে আপনার APPLICATION/SYSTEM এর সাথে যোগাযোগ করবে।

(যে role play করবে সেই একটা **ACTOR**। এখানে একাধিক ACTOR থাকতে পারে)



Use Case

USE CASE:

Use Case এর চিত্রটা সবসময় এমনই দেখতে (ওভাল শেইপ) হবে এবং এইভাবেই প্রকাশ করতে হবে।

সিস্টেমটা আসলে কি কি করতে পারে বা সিস্টেমটার **capability** কোন কোন কাজের জন্য সেটাকেই আমরা বলি ওই সিস্টেমটার জন্য **Use Case** .



Connector/ Link

Connector:

একটার সাথে আরেকটার কানেকশন তৈরী করে এই চিহ্নটার মাধ্যমে।

উপরে আমরা ACTOR পেলাম এবং USE CASE ও পেলাম এখন তাদের একে অপরের সাথে CONNECT/ সংযোগ করবো এই চিহ্ন দিয়ে



Generalization

GENERALIZATION :

Generalization এই চিহ্নের মাধ্যমে প্রকাশ করা হয়।

প্রথমেই বলে নেই এটা অনেকে ব্যবহার করতে চায় না আবার অনেকে ব্যবহার করে তাই যখন যেখানে ব্যবহার করবেন তার আগে জিজ্ঞাসা করে নিবেন এটা ব্যবহার করতে পারবেন কিনা তাহলে আর কোনো সমস্যা হবে না।

GENERALIZATION:

Generalization মানে হচ্ছে আপনার কাজটাকে আরো **General** করে দেওয়া ।

একটা উদাহরন দেইঃ

ধরেন পরবর্তী সেমিস্টারে আপনি কিছু সাবজেক্ট নিবেন সেটা আমাকে বললেন । কিন্তু কোন ধরনের সাবজেক্টগুলো নিবেন সেটা কিন্তু বলেননি । যেমনঃ এভাবে আরো **General** করে বলতে পারেন “ **Departmental and Non-Departmental** ” .

এইষে আপনি এভাবে প্রকাশ করে আরো **General** করে দিলেন এটাই হলো **Generalization** .

আরেকটা উদাহরন দেইঃ

তোমাকে বললাম এখানে কিছু পাখি দেখা যাচ্ছে । যদি এটাকে আরো সুন্দর করে বলি এখানে ২টা টিয়া পাখি এবং ৩টা ময়না পাখি দেখা যাচ্ছে । তাহলে ব্যপারটা কি হলো আমি আরো **General** করে দিলাম । কি কি পাখি আছে সেটাও বলে দিলাম। এটাই হলো **Generalization**

RELATIONSHIP :

Relationship মানে হলো একে অপরের সাথে সম্পর্ক স্থাপন করা ।

এই চিহ্ন দিয়েই **Relationship** বুঝানো হয় ।

Relationship আবার দুই ধরনের।

যেমন:

একটা হলো <<include>>

আরেকটি হলো <<extend>>

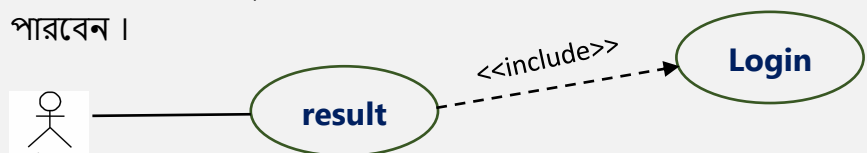
<<include>> RELATIONSHIP

(যে যার উপর নির্ভরশীল তার দিকে চিহ্ন -> দিতে হবে)

<<include>> মানে হচ্ছে আপনি একটা কাজ করতে চাচ্ছেন কিন্তু আপনি ঐ কাজটি অন্য আরেকটা কাজ করা ছাড়া করতে পারবেন না/ পারছেন না । তার মানে ঐ কাজটা আরেকটা কাজের উপর নির্ভরশীল । এটাই হচ্ছে <<include>> রিলেশনশীপ ।

ধরেন, আপনি আপনার ভার্চুয়াল পোর্টাল থেকে রেজাল্ট দেখতে চাচ্ছেন তাহলে আপনাকে প্রথমে কি করতে হবে >>>???

LOGIN করতে হবে, তারপর আপনি পোর্টালের ভিতরের যাবতীয় কাজ করতে পারবেন ।



Relationship

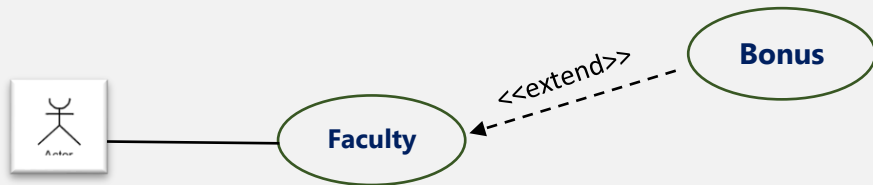
RELATIONSHIP :

<<extend>> RELATIONSHIP

<<extend>> হচ্ছে <<include>> এর অপজিট/ উল্টা ।

<<include>> এ আপনাকে একটা কাজ করে আসতে হতো কিন্তু <<extend>> এ আপনাকে সেটা করতে হবে না । এটা দরকার হতেও পারে আবার নাও হতে পারে ।

মনে করেন আপনি সেমিস্টার ফাইনাল পরীক্ষা দিয়েছেন। এখন আপনি যদি রেগুলার ক্লাস করে থাকেন এবং ভালো রেজাল্ট করেন এবং অল্পের (১/২) জন্য যদি গ্রেড মিস হয়ে যায় তখন আপনার ফেকাল্টির দয়া মায়া হলে আপনাকে ১/২ মার্কস বোনাস দিয়ে গ্রেড বাড়াতেও পারে আবার নাও বাড়াতে পারে । এটাই হচ্ছে **<<extend>> Relationship**. এখানে বোনাস হচ্ছে ফেকাল্টির উপর নির্ভরশীল তাই চিহ্নটা ফেকাল্টির দিকেই আছে ।



USE CASE DIAGRAM করতে কিছু জিনিস জানতেই হবেঃ

১। USE CASE DIAGRAM, গল্প (সিনারিও) আকারে থাকতে পারে অথবা সরাসরি বলতে পারে যেকোনো একটা সিস্টেমের উপর ডায়াগ্রাম তৈরী করো। এতে করে সবার ডায়াগ্রামই আলাদা হবে/ হতে পারে এতে চিন্তার কিছু নেই।

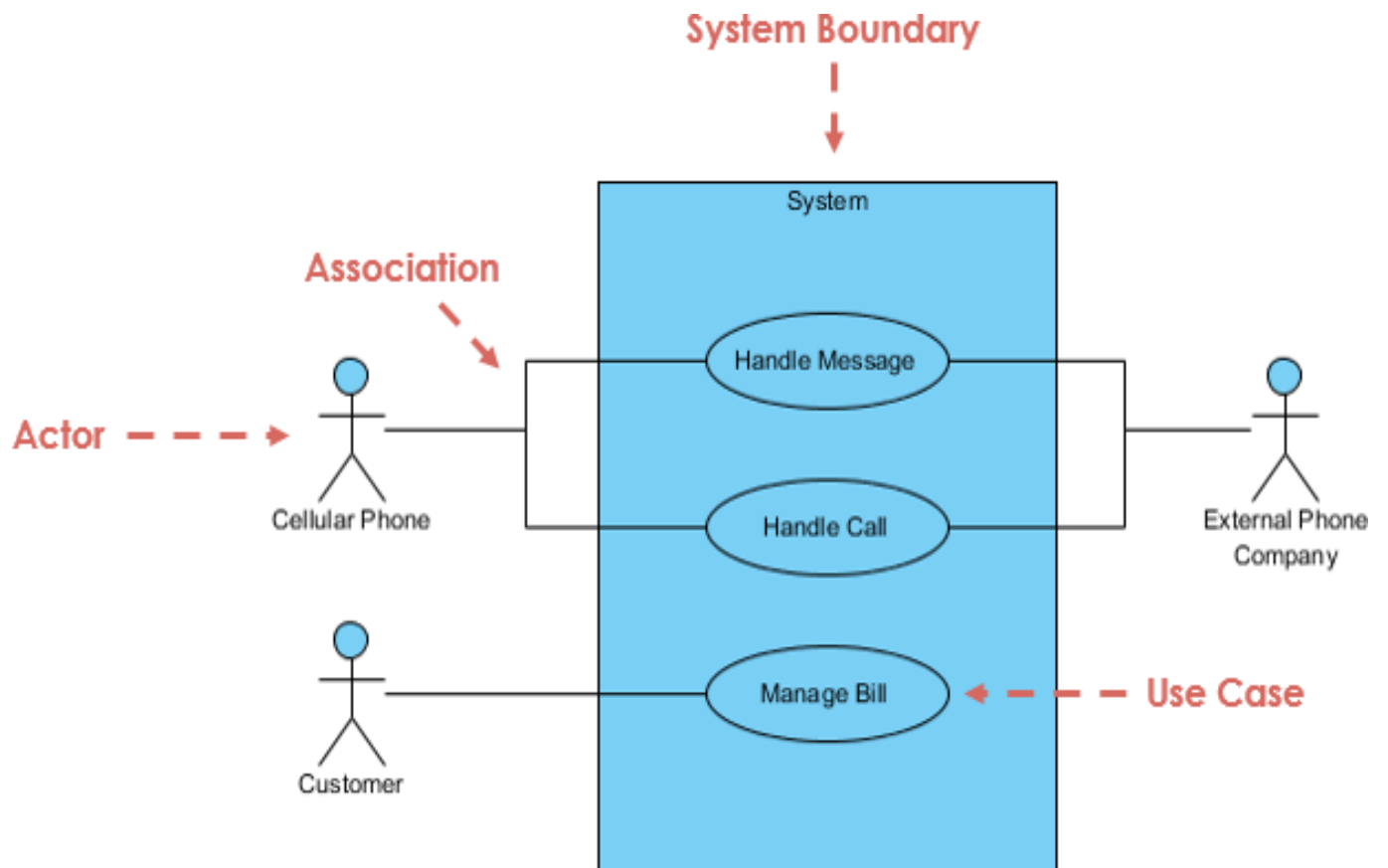
২। আপনি যখন USE CASE DIAGRAM লিখবেন অবশ্যই চেস্টা করবেন USE CASE এর নামগুলোকে VERB দিয়ে শুরু করতে। এটা বেস্ট কেইস। এবং এমনও হতে পারে এই VERB ব্যবহার না করার কারনে আপনার পুরো ডায়াগ্রামই কেটে দিতে পারে।

৩। আপনাকে একটা ব্যাপারে খেয়াল রাখতে হবে ACTOR গুলো যেনো সবসময় SINGULAR হয় PLURAL করা যাবে না কিন্তু। এবং একটা ACTOR এর সাথে আরেকটা ACTOR এর কোনো কানেকশন দিতে পারবেন না।

৪। অনেকেই একটা ভুল করেন সেটা হলো USE CASE এর ভিতর একটা অথবা দুইটা লাইন লিখে দেন। এটা কখনোই করা যাবে না। চেস্টা করবেন USE CASE কে সর্বোচ্চ তিনটা শব্দের মধ্যে রাখার জন্য।

৫। আপনি যখন USE CASE করতে যাবেন সেখানে একাধিক ACTOR থাকতেই পারে, তাই চেস্টা করবেন অন্তত দুইটা ACTOR নিয়ে কাজ করতে তাহলে দেখতে সুন্দর হবে। হিজিবিজি করা যাবে না। প্রথমে USE CASE গুলো লিখবেন সব, তারপর ACTOR এর সাথে সংযোগ করে দিবেন।

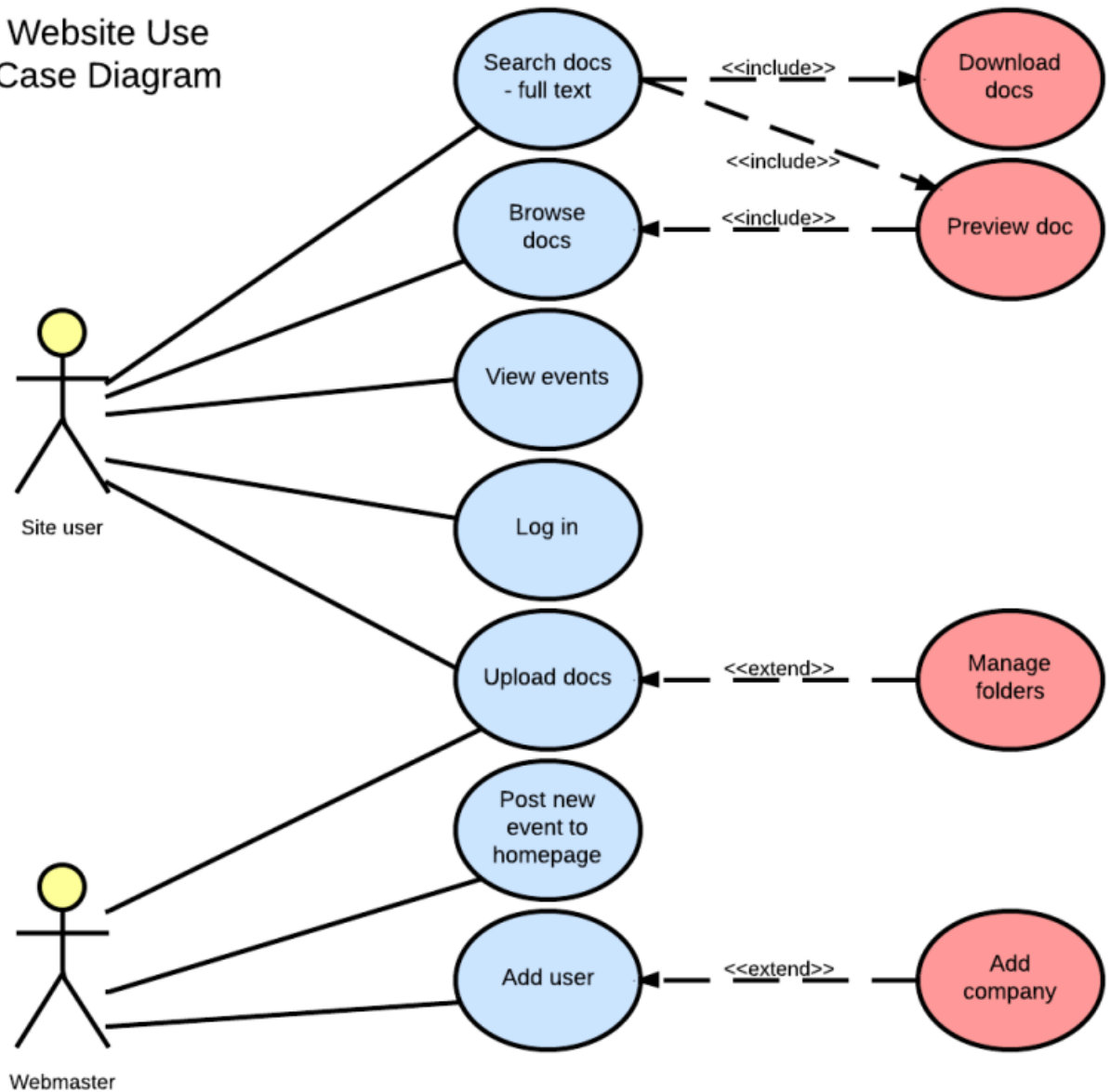
কিছু উদাহরণ দেখি তাহলে অনেকটাই বুঝতে পারবো:



চিত্রঃ একটা USE CASE DIAGRAM এর মডেল

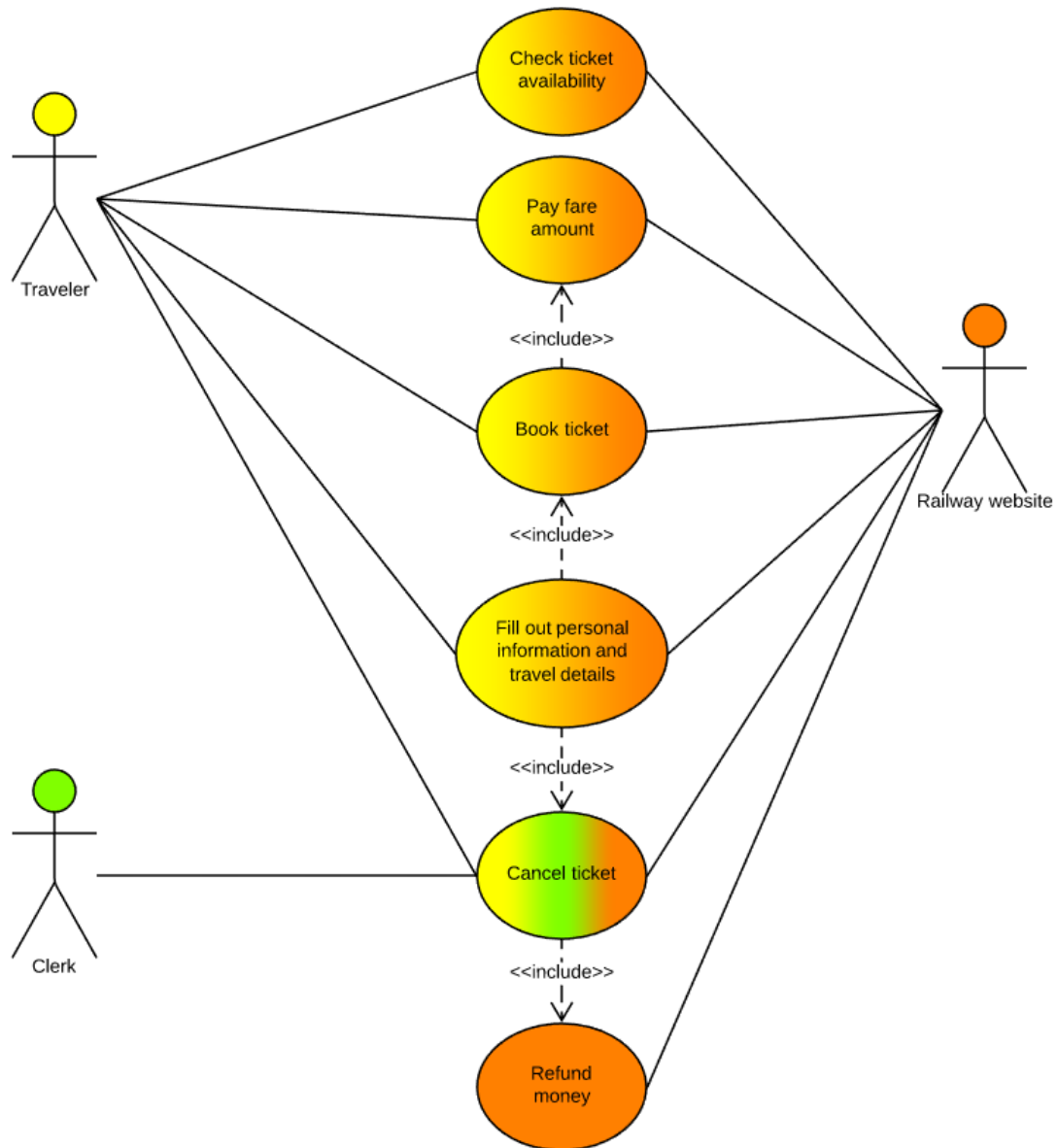
NO. 1

Website Use
Case Diagram



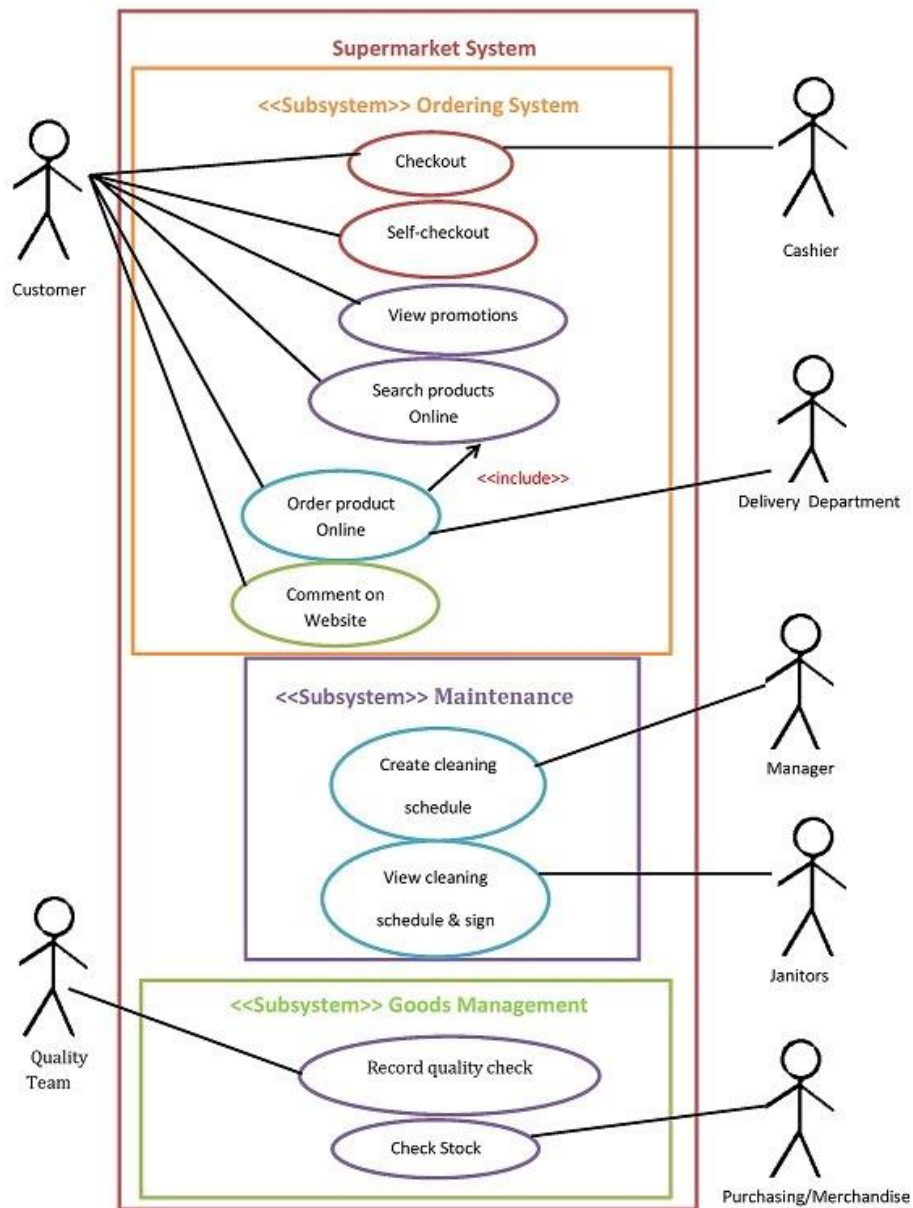
NO. 2

Railway reservation use case diagram example



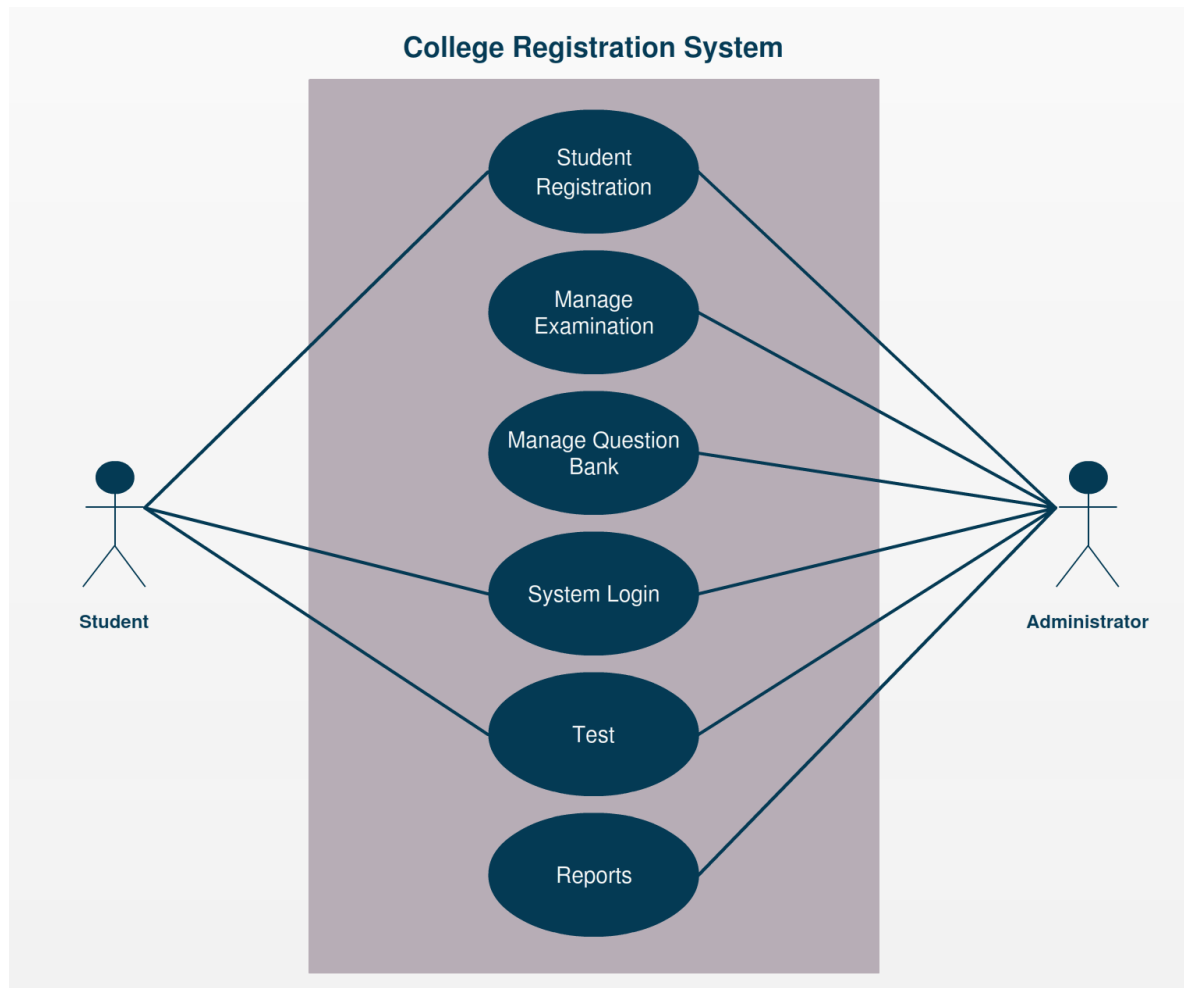
NO. 3

SUPER MARKET SYSTEM



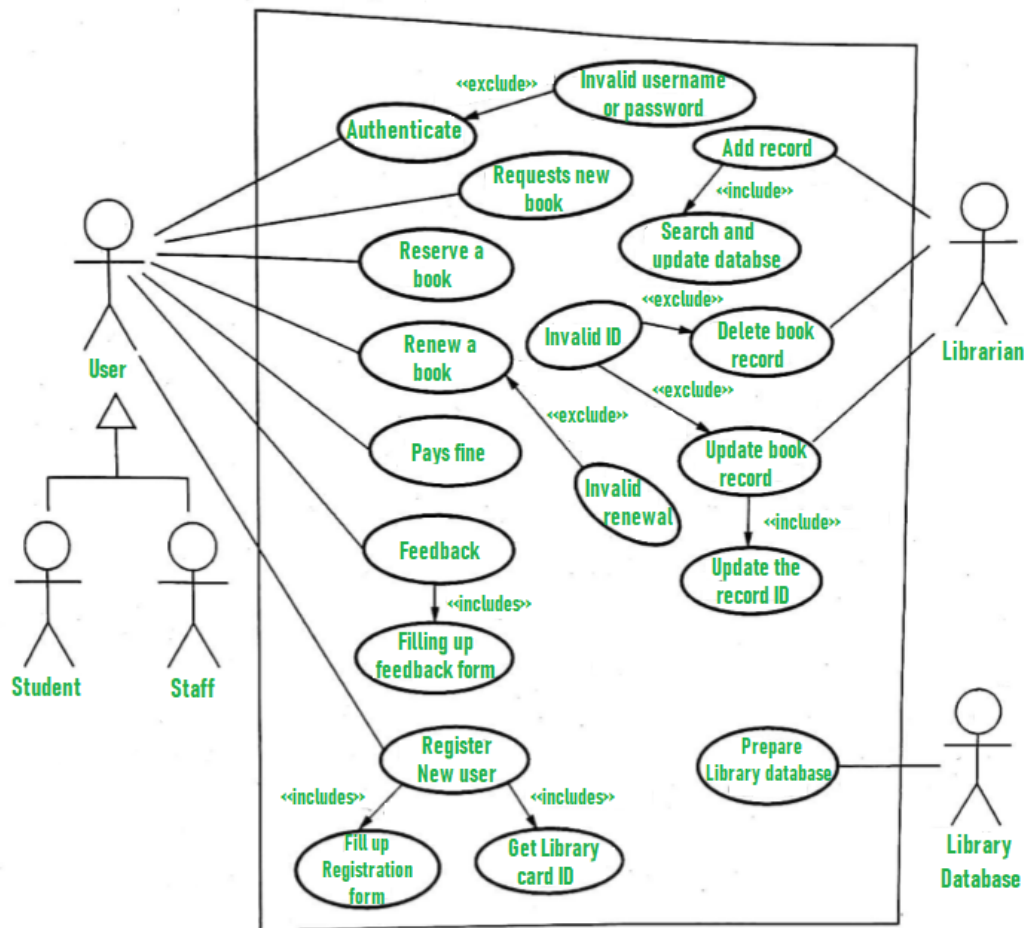
NO. 4

COLLEGE REGISTRATION SYSTEM



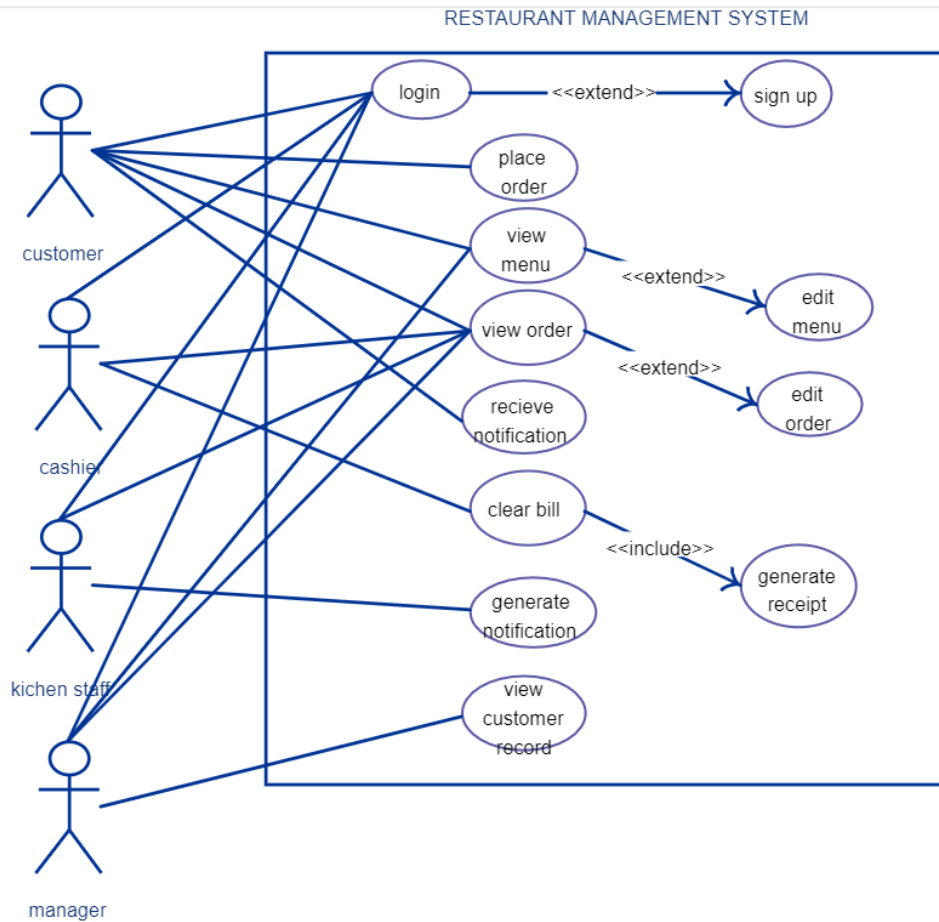
NO. 5

Use Case Diagram for Library Management System



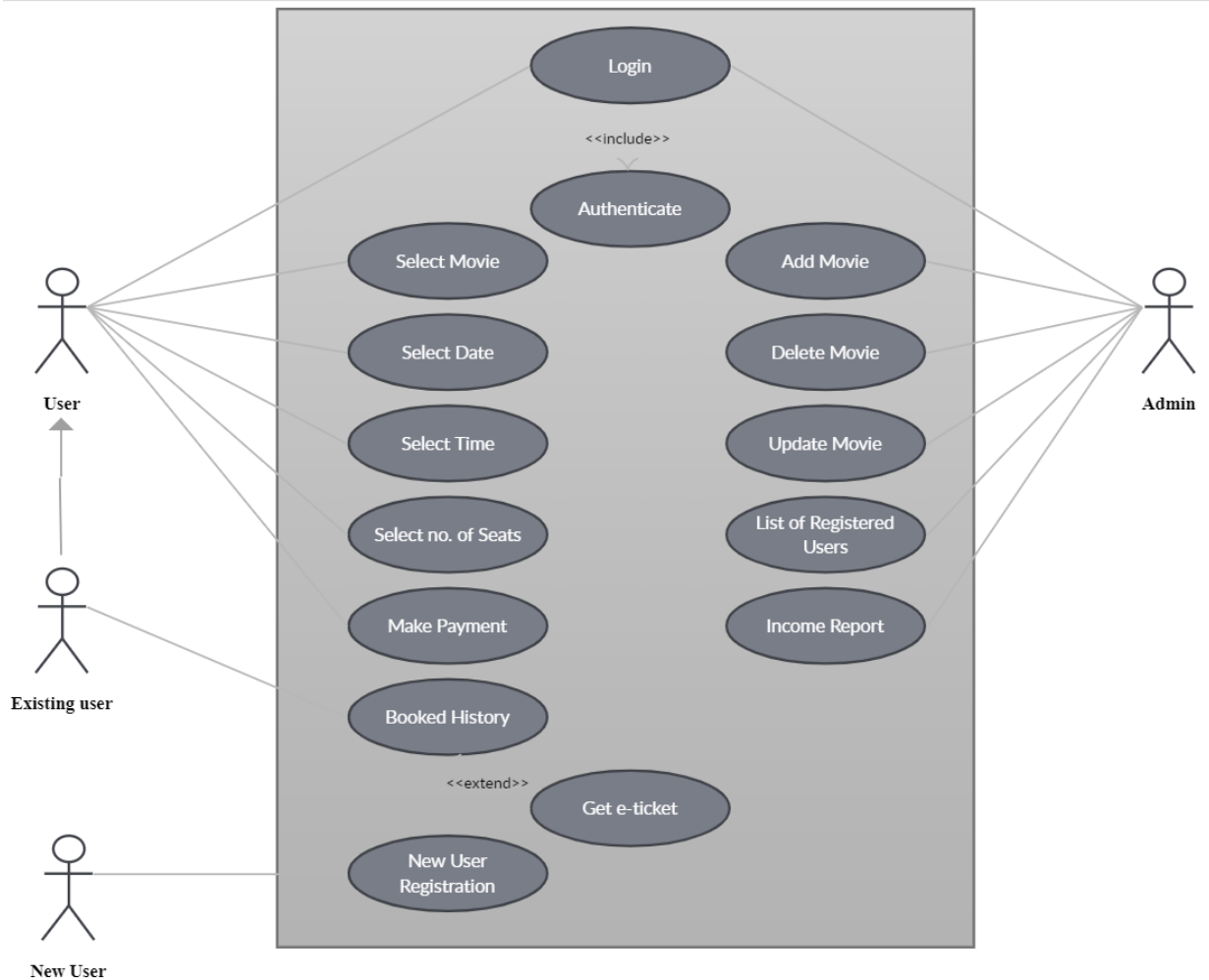
NO. 6

RESTURENT MANAGEMENT SYSTEM



NO. 7

ONLINE MOVIE TICKET BOOKING SYSTEM



NO. 8

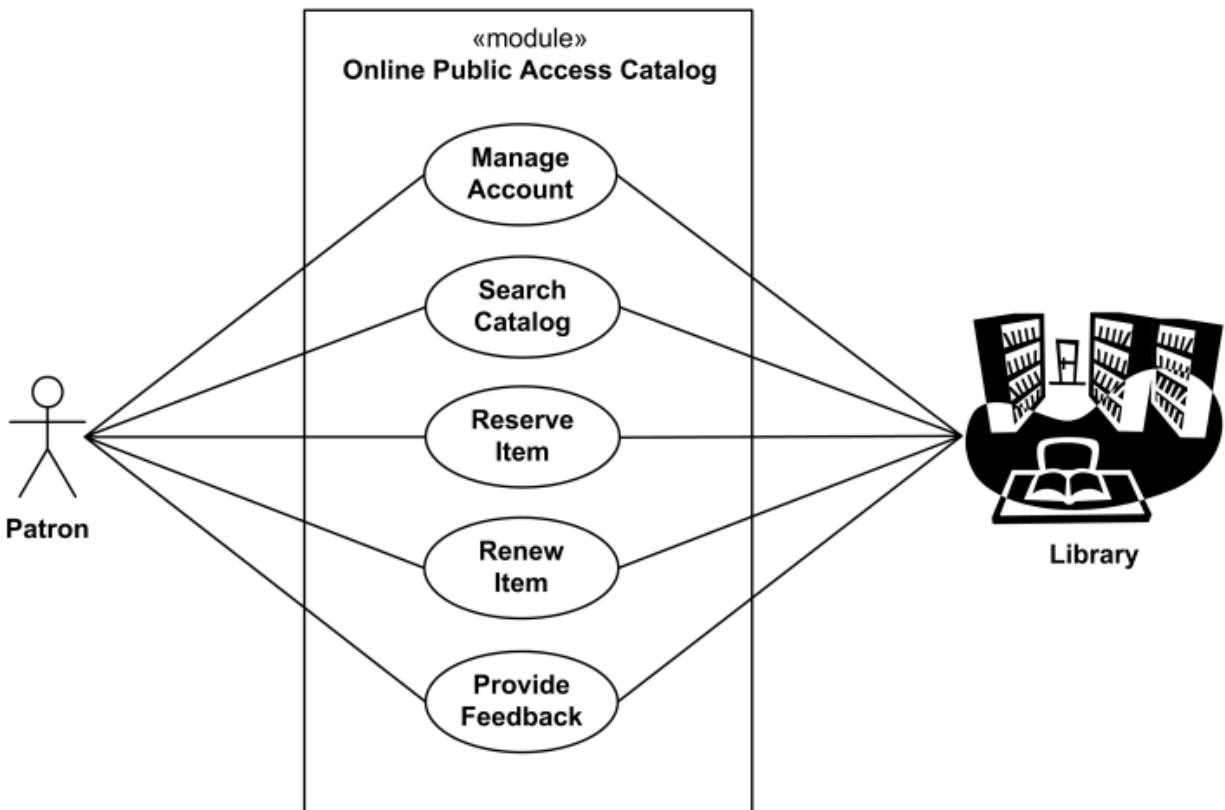
e-Library online public access catalog (OPAC)

Purpose: *List top level use cases for e-Library online public access catalog.*

**** An Online Public Access Catalog (OPAC)** is an e-Library website which is part of **Integrated Library System (ILS)**, also known as a **Library Management System (LMS)**, and managed by a library or group of libraries. ******

Summary:

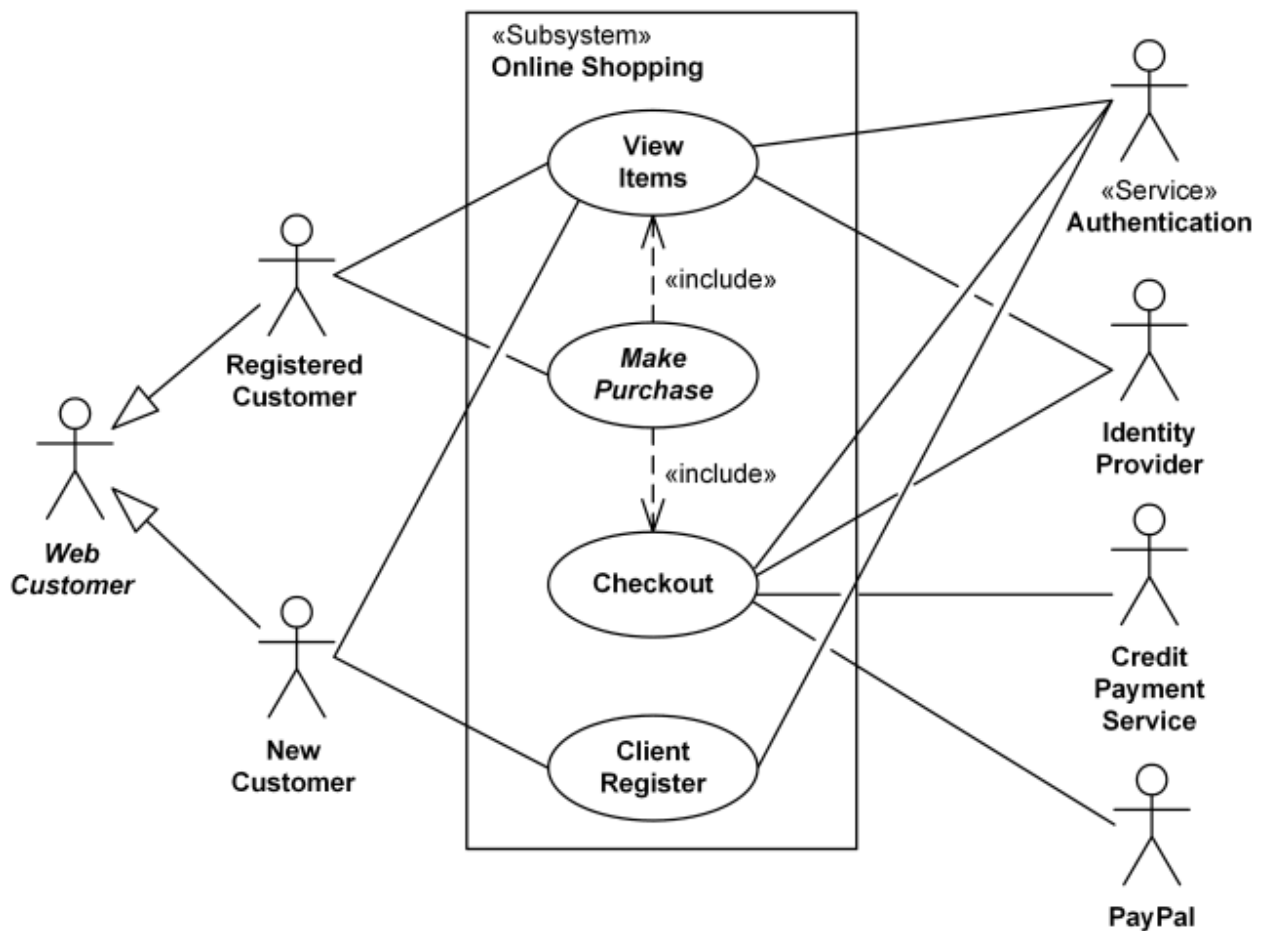
Patrons of a library can search library catalog online to locate various resources - books, periodicals, audio and visual materials, or other items under control of the library. Patrons may reserve or renew item, provide feedback, and manage their account.



NO. 9

Web Customer actor uses some web site to make purchases online. Top level **use cases** are **View Items**, **Make Purchase** and **Client Register**. View Items use case could be used by customer as top level use case if customer only wants to find and see some products. This use case could also be used as a part of Make Purchase use case. Client Register use case allows customer to register on the web site, for example to get some coupons or be invited to private sales. Note, that **Checkout** use case is **included use case** not available by itself - checkout is part of making purchase.

Except for the **Web Customer** actor there are several other actors which will be described below with detailed use cases.

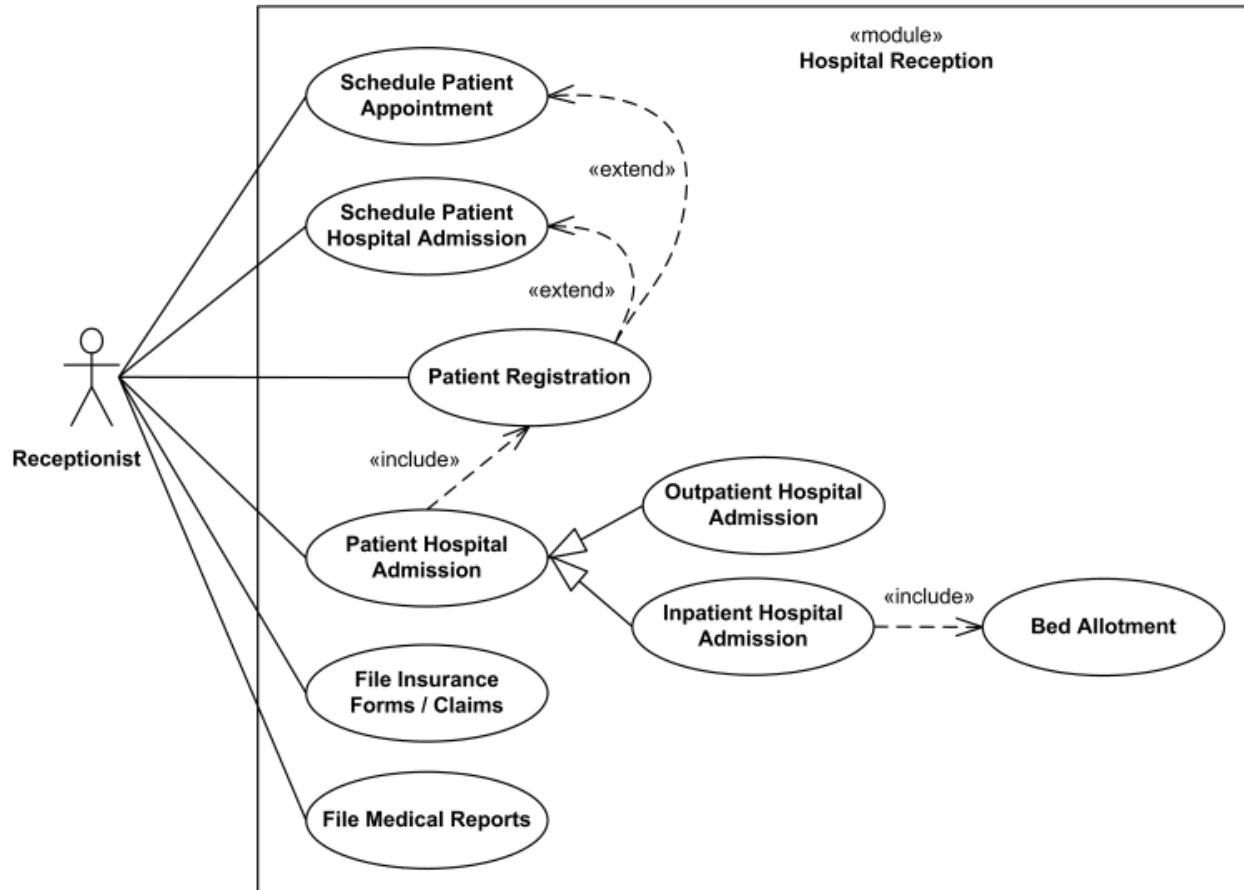


NO. 10

Hospital Management System is a large system including several subsystems or modules providing variety of functions. UML use case diagram example below shows actor and use cases for a hospital's reception.

Purpose: *Describe major services (functionality) provided by a hospital's reception.*

Hospital Reception subsystem or module supports some of the many job duties of hospital receptionist. Receptionist schedules patient's appointments and admission to the hospital, collects information from patient upon patient's arrival and/or by phone. For the patient that will stay in the hospital ("inpatient") she or he should have a bed allotted in a ward. Receptionists might also receive patient's payments, record them in a database and provide receipts, file insurance claims and medical reports.



REFERENCE :

1. <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-use-case-diagram/>
2. <https://creatly.com/blog/diagrams/use-case-diagram-tutorial/>
3. <https://www.uml-diagrams.org/use-case-diagrams-examples.html>
4. EVERY DIAGRAM COLLECT FROM GOOGLE

[বিঃদ্রঃ মানুষ ভুল ত্রুটির উর্ধে নয় । তাই কোনো ভুল থাকলে জানাবেন ।]