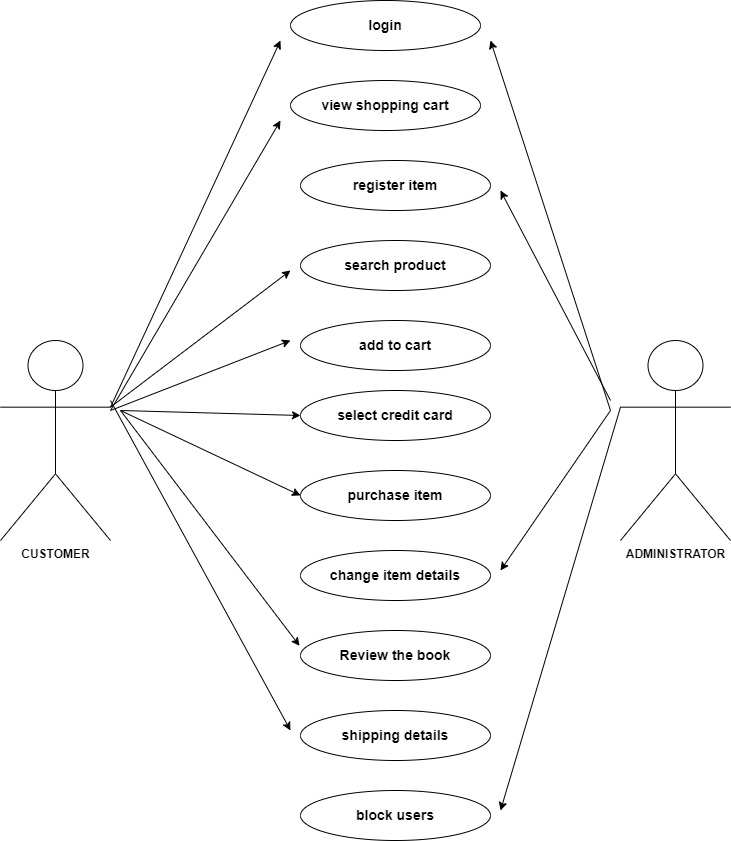
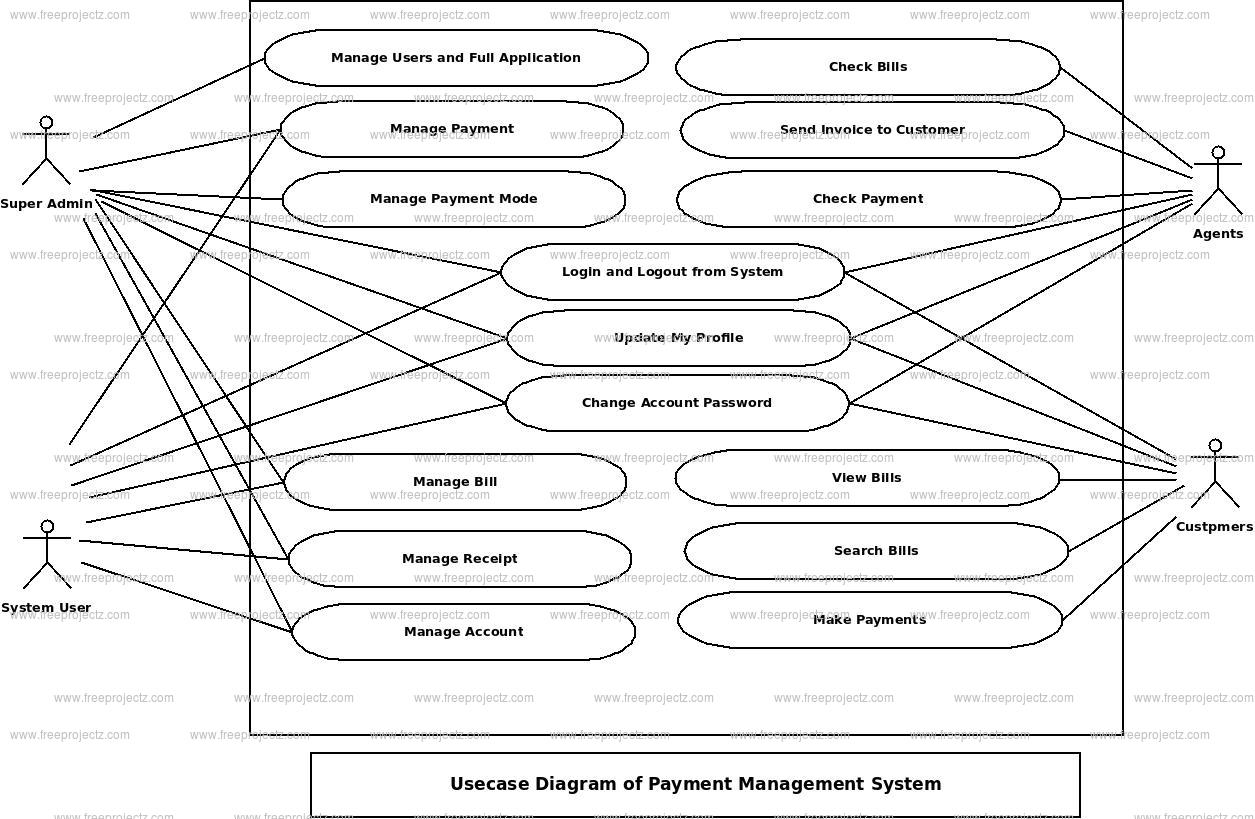
1. SDLC (software development life cycle) is a structure imposed on the development of a software product that defines the process for planning, implementation, testing, documentation,

deployment, and ongoing maintenance and support. There are a number of different development models.

1. Software Testing is a process used to identify the correctness, completeness, and quality of developed computer software.
2. Agile methodology is a combination of iterative and incremental process models with focus on process adaptability and customer satisfaction by rapid delivery of working software product.
3. SRS (software requirement specification) is a complete description of the behavior of the system to be developed. It includes a set of use cases that describe all of the interactions that the users will have with the software.
4. OOPS (Object oriented program) Identifying objects and assigning responsibilities to these objects. Objects communicate to other objects by sending messages. Messages are received by the methods of an object.
5. Object-oriented programming has a web of interacting objects, each house-keeping its own state. Objects of a program interact by sending messages to each other. An object is like a black box and the internal details are hidden.
6. A class represents an abstraction of the object and abstracts the properties and behavior of that object. Class can be considered as the blueprint or definition or a template for an object and describes the properties and behavior of that object, but without any actual existence.
7. Encapsulation is the practice of including in an object everything it needs hidden from other objects. The internal state is usually not accessible by other objects. Encapsulation is placing the data and the functions that work on that data in the same place.
8. Inheritance means that one class inherits the characteristics of another class. This is also called a “is a” relationship. Inheritance is the process of forming a new class from an existing class that is from the existing class called as base class, new class is formed called as derived class. Inheritance describes the relationship between two classes.
9. Polymorphism means “having many forms”. It allows different objects to respond to the same message in different ways, the response specific to the type of the object.





1. <1> Requirement collection – Establish customer needs

<2> Analysis – Model and specify the requirements - what

<3> Design – Model and specify a solution – why

<4> Implementation – Construct a solution in software

<5> Testing – Validate the solution against the requirements

<6> Maintenance – Repair defects and adapt the solution to the new requirements

1. <1> Requirement analysis - The requirements phase of waterfall project management identifies what the project should do. This phase involves identifying and describing the project’s risks, assumptions, dependencies, quality metrics, costs, and timeline.

<2> Design - The design phase solidifies and documents all your decisions. In this case, you develop solutions that can solve the project’s requirements.

<3> Implementation - The implementation phase executes your project plan and design to produce the desired product. If your company develops software, you will spend this phase coding the software functionalities.

<4> Testing - Testing verifies that the product developed in the implementation phase fulfills the entire project’s requirements.

<5> Maintenance - The maintenance phase extends beyond the five stages of project management into the project’s lifetime. This phase involves making minor modifications to improve the product developed during implementation and performing other routine maintenance tasks.

1. <1> Initial Requirements

<2> Risk

<3> Evolving System

<4> Alpha Demo

1. <1> Individuals and interactions over processes and tools.

<2> Working software over comprehensive documentation.

<3> Customer collaboration over contract negotiation.

<4> Responding to change over following a plan.

1. Agile model working methodology - The Agile methodology is a way to manage a project by breaking it up into several phases. It involves constant collaboration with stakeholders and continuous improvement at every stage. Once the work begins, teams cycle through a process of planning, executing, and evaluating. Continuous collaboration is vital, both with team members and project stakeholders.

Pros. - Is a very realistic approach to software development.

Promotes teamwork and cross training.

Functionality can be developed rapidly and demonstrated.

Resource requirements are minimum.

Suitable for fixed or changing requirements.

Delivers early partial working solutions.

Good model for environments that change steadily.

Minimal rules, documentation easily employed.

Enables concurrent development and delivery within an overall planned context.

Little or no planning required Easy to manage Gives flexibility to developers.

Cons. - Not suitable for handling complex dependencies.

More risk of sustainability, maintainability and extensibility.

An overall plan, an agile leader and agile PM practice is a must without which it will not

work.

Strict delivery management dictates the scope, functionality to be delivered, and

adjustments to meet the deadlines.

Depends heavily on customer interaction, so if customer is not clear, team can be driven

in the wrong direction.

There is very high individual dependency, since there is minimum documentation

generated.

Transfer of technology to new team members may be quite challenging due to lack of

documentation.





