Module-1

(fundamental)ss

Assignment Software Testing

1. What is SDLC ?

- SDLC Stands For “Software Development Life Cycle”.

To Ending.

- A Series of Steps Or Phases That Provides The Model Of Development.

- Simple Word “Entire Life Time of Software From Beginning

2. Write SDLC Phases With Basic Introduction ?

Phases Of SDLC:

1. Requirement Gathering:

- It Is The First Page Of SDLC In Which All The Necessary Information Is Collected From The Customer To Develop The Software As Per Their Expectation.

- Some Important Question Like: What Is The Need Of Software? , Who Will Be The End –User , What Is The Future Scope Of That Software Etc. Are Discussed.

- This Phase Gives A Clear Picture Of What We Are Gpoing To Build.

2. Planning:

- In This Phase Which Organization Discuss About The Cost &Benefits Of The Software.

- It Is Important Phase Because Profit From The Software Play An Important Role As If Cost Is Very High Then Company Many Face Loss.

- After The Planning Study, The Project May Be Accepted, Accepted With Modification Or Rejected

3. Design: 2

- In This Phase Which Architect Start Working On Logical Designing Of The Software.

- In This Phase SRS (System Requirement Specification) Document Is Created Which Contain All Logical Details Like. How The Software Will Look Like, Which Language Is Used, Database Design, Modular Design Etc.

- This Phase Is Provide A Prototype Of The Final Product.

4 .Implementation:

- When Designning Of The Of The Software Is Completed Then, A Group Of Developers Start Coding Of The Design Using A Programming Language

- A Number Of Developers Code The Modules & Then All Modules Are Arranged Together To Working Efficiently.

- It Is Longest Phase Of SDLC.

5. Testing:

- Once The Software Development Is Completed Then It Sent To The Testers, The Testing From Start Testing The Functionality Of The Entire System.

- In This Phase, The Software Is Checked F

- Whenever Bug Is Found, Then The Software Is Resent To The Coder To Fix & Then Overall Software Is Re-Tested.

- This Is Done To Verify That The Entire Application Works According To The Customer Requirement.

6. Deployment:

- After Overall Testing Of The Software And After Checking That Is Bug Free, Then The Software Is Launched & Available For The User To Use It.

• Even After Deployment Of The Software, If Any Bug Or Error Are Still Found Then The Software Is Re-Evaluated By The Maintenance Team It Is Re-Deployed With A New Version. 3

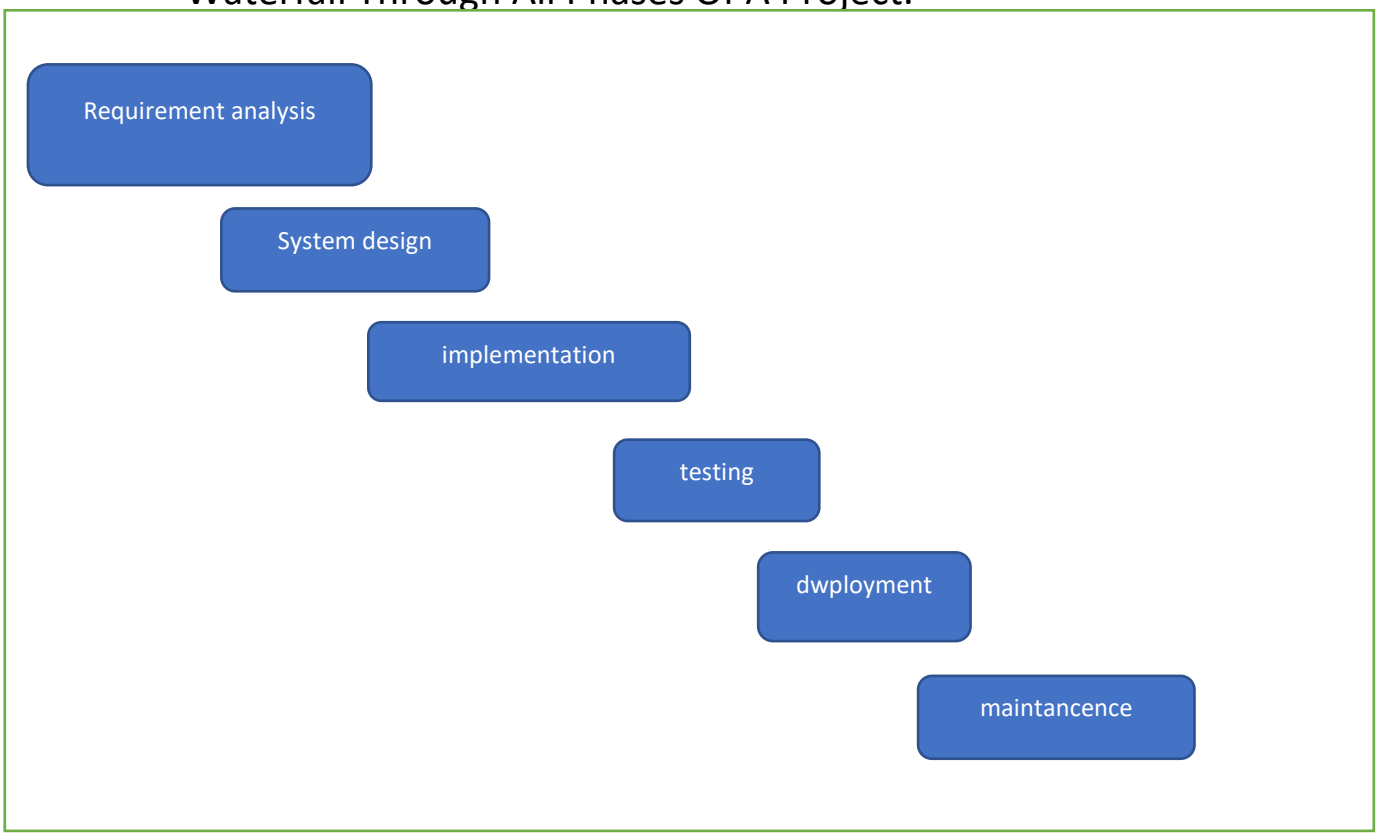
7. Maintenance:

• The Maintenance Team Look Over The Software Usage & User's Feedback.

• Maintenance Is Necessary To Eliminated Error In The System During Its Working Life & To Tune The Software. 1. Corrective Maintenance: Identify & Repairing Defects. 2. Adaptive Maintenance: Adaptive The Existing Solution To The New Platform. 3. Perfective Maintenance: Implementing New Requirement.

3. Explain Phase Of Waterfall Model?

• It Is A Sequential Development Process That Flow Like A Waterfall Through All Phases Of A Project.



Details Discussions With Clients And Clearly Visualize The Product. All Possible Requirements Of The System To Be Developed Are Captured In This Phase And Documented In A Requirement Specification Document.

- **Design:-**

The Requirement Specifications From First Phase Are Studied In This Phase And The System Design Is Prepared. This System Design Helps In Specifying Hardware And System Requirements And Helps In Defining The Overall System Architecture..

- **Implementation:-**

With Inputs From The System Design, The System Is First Developed In Small Programs Called Units, Which Are Integrated In The Next Phase. Each Unit Is Developed And Tested For Its Functionality, Which Is Referred To As Unit.

- **Testing:-**

All The Units Developed In The Implementation Phase Are Integrated Into A System After Testing Of Each Unit. Post Integration The Entire System Is Tested For Any Faults And Failures.

- **Maintenance:-**

There Are Some Issues Which Come Up In The Client Environment. To Fix Those Issues, Patches Are Released. Also To Enhance The Product Some Better Versions Are Released. Maintenance Is Done To Deliver These Changes In The Customer Environment.

4. What is oops ?

- The Basic Oriented Programming Is Based On Real World Entities Like Inheritance, Polymorphism,Etc.

- An Object –Based Programming Language Is One Which Easily Supports Object Orientation.

- An Object Is Like A Black Box And The Internal Details Are Hidden.

5 . write Basic Concept Of Oops?

- Object
- Class
- Encapsulation
- Inheritance
- Polymorphism
- Abstraction

6. What Is Object?

Object Gives The Permission To Access Functionality Of Class.

7. What Is class?

Class Is A Collection Of Data Member &Member Function.

8. What Is Encapsulation?

The Process Wrapping The Data In A Single Unit. To Secure The Data From Outside World.

9. What Is Inheritance ?

Making Class From An Existing Class.Deriving The Attribute Of Some Parent Class.

10. What Is Polymorphism?

One Name Multiple Form.

¬ Overloading-Same Name But Different Parameter(Must Be Use Inheritance)

1.Function Overloading

Rocky(Int A,Int B)
Rocky(Int A)

2.Constructor Overloading: Same Constructor Name But Different Parameter.

¬ Overriding: Same Name But Same Same Parameter(Must Use Inheritance)

Rocky(Int A)
Rocky(Int A)

11. What Is Abstraction?

Hiding Details And Showing Only Essential Information.

12. what is software testing?

Software Testing Is A Process Used To Identify The Correctness, Completeness, And Quality Of Developed Computer Software.

13.What Is SRS?

Software Requirement Specifications. SRS Is Complete Description Of An Applications Which Is To Be Developed Three Types Of SRS. Like FRS, BRS, FRD.

14. What Is Agile Methodology?

Agile SDLC Model Is Combinations Of Iterative And Incremental Process Model With Focus On Process Adaptability & Customer Satisfaction By Rapid Delivery Of Working Software Product.

15. Explain Working Of Methodology Of Agile Model And Also Write Pros And Cos?

Agile SDLC Model Is Combinations Of Iterative And Incremental Process Model With Focus On Process Adaptability & Customer Satisfaction By Rapid Delivery Of Working Software Product. Agile Methods Break The Product In To Small Incremental Builds. Each Iteration Typically Lasts From About One To Three Weeks.

Pros:

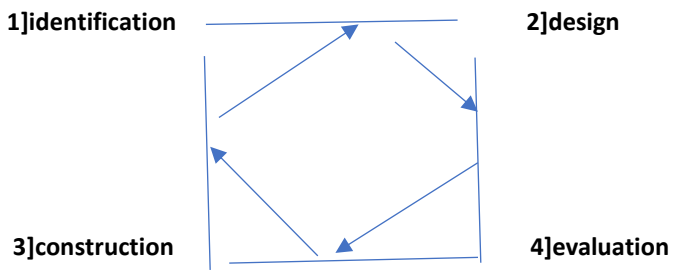
- Face To Face Communications With Client.
- Easy To Manage.
- Frequent Delivery
- Gives Flexibility To Developers.
- No Planning Required.

Cons:

- Not Suitable For Handling Complex Dependencies.
- Less Documentation.
- Maintenance Problem.

16. Write Phase Of Spiral Model ?

Spiral Model Is System Development Life Cycle Method Used For Risk Management That Combine The Iterative Development Process Model With Element Of The Waterfall Model.



Spiral model

Planning:-

Determination Of Objective, Alternative And Constrains.

Risk Analysis:-

Analysis Of Alternatives And Identification/Resolutions Of Risks. Risk:-Something That Will Delay Project Or Increase Its Cost.

Engineering:-

Development Of The 'Next Level' Product.
Customer Evualation:- Assessment Of The Result Of Engineering.

17. Write Agile Manifesto Principles?

- **Customer Satisfaction Through Early And Continuous Software Delivery–** Customers Are Happier When They Receive Working Software At Regular Intervals, Rather Than Waiting Extended Periods Of Time Between Releases

- **Accommodate Changing Requirements Throughout The Development Process –** The Ability To Avoid Delays When A Requirement Or Feature Request Changes.

- **Frequent Delivery Of Working Software –** Scrum Accommodates This Principle Since The Team Operates In Software Sprints Or Iterations That Ensure Regular Delivery Of Working Software.

- **Collaboration Between The Business Stakeholders And Developers Throughout The Project –** Better Decisions Are Made When The Business And Technical Team Are Aligned

- **Support, Trust, And Motivate The People Involved –** Motivated Teams Are More Likely To Deliver Their Best Work Than Unhappy Teams.

- **Enable Face-To-Face Interactions –** Communication Is More Successful When Development Teams Are Co-Located.

- **Working Software Is The Primary Measure Of Progress –** Delivering Functional Software To The Customer Is The Ultimate Factor That Measures Progress.

- **Agile Processes To Support A Consistent Development Pace –** Teams Establish A Repetable And Maintainable Speed At Which They Can Deliver Working Software, And They Repeat It With Each Release.

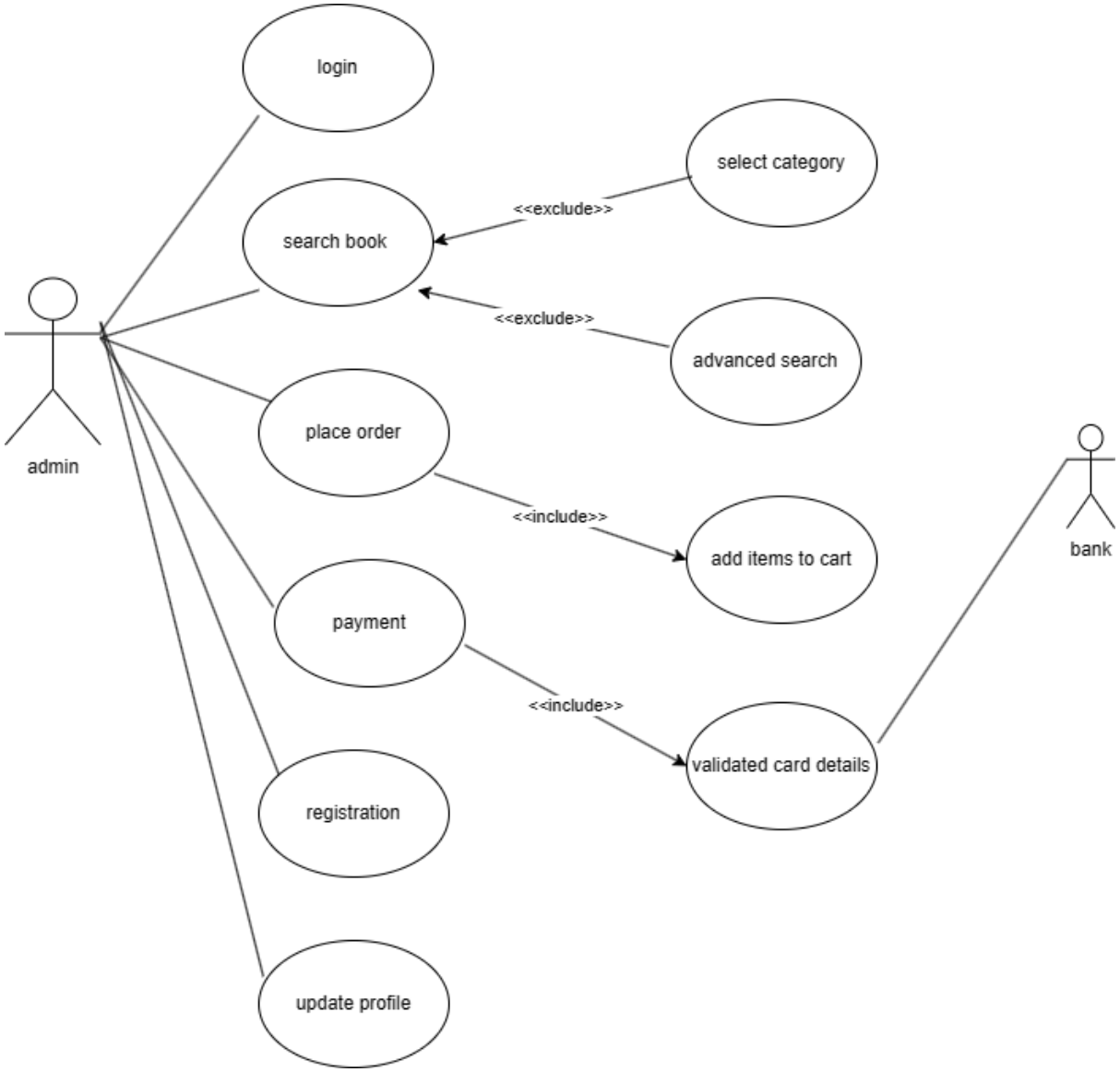
- **Attention To Technical Detail And Design Enhances Agility –** The Right Skills And Good Design Ensures The Team Can Maintain The Pace, Constantly Improve The Product, And Sustain Change.

- **Simplicity –** Develop Just Enough To Get The Job Done For Right Now.

- **Self-Organizing Teams Encourage Great Architectures, Requirements, And Designs –** Skilled And Motivated Team Members Who Have Decision-Making Power, Take Ownership, Communicate Regularly With Other Team Members, And Share Ideas That Deliver Quality Products.

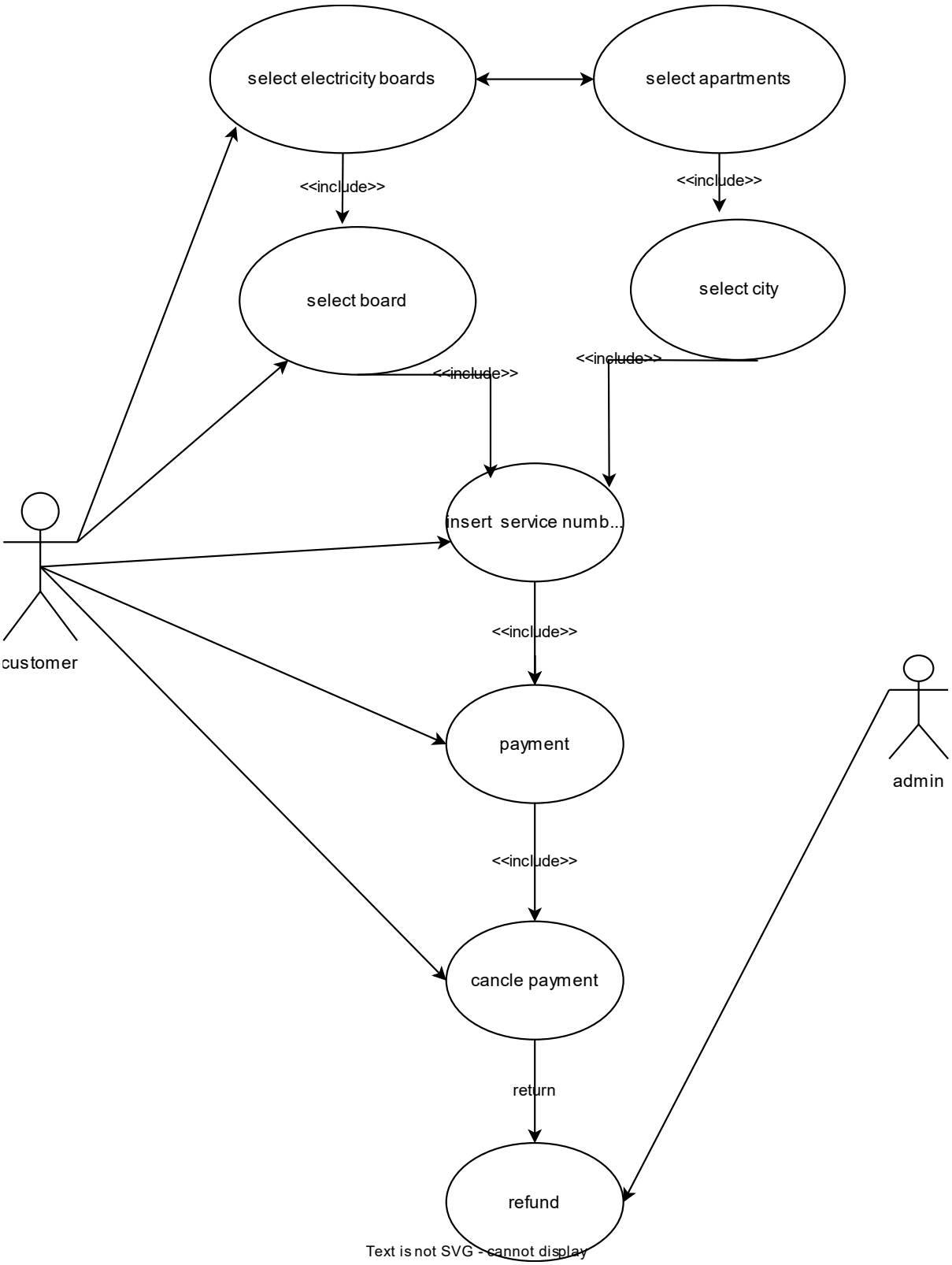
- **Regular Reflections On How To Become More Effective –** Self-Improvement, Process Improvement, Advancing Skills, And Techniques Help Team Members Work More Efficientl

18. Draw usecase on online book shopping?

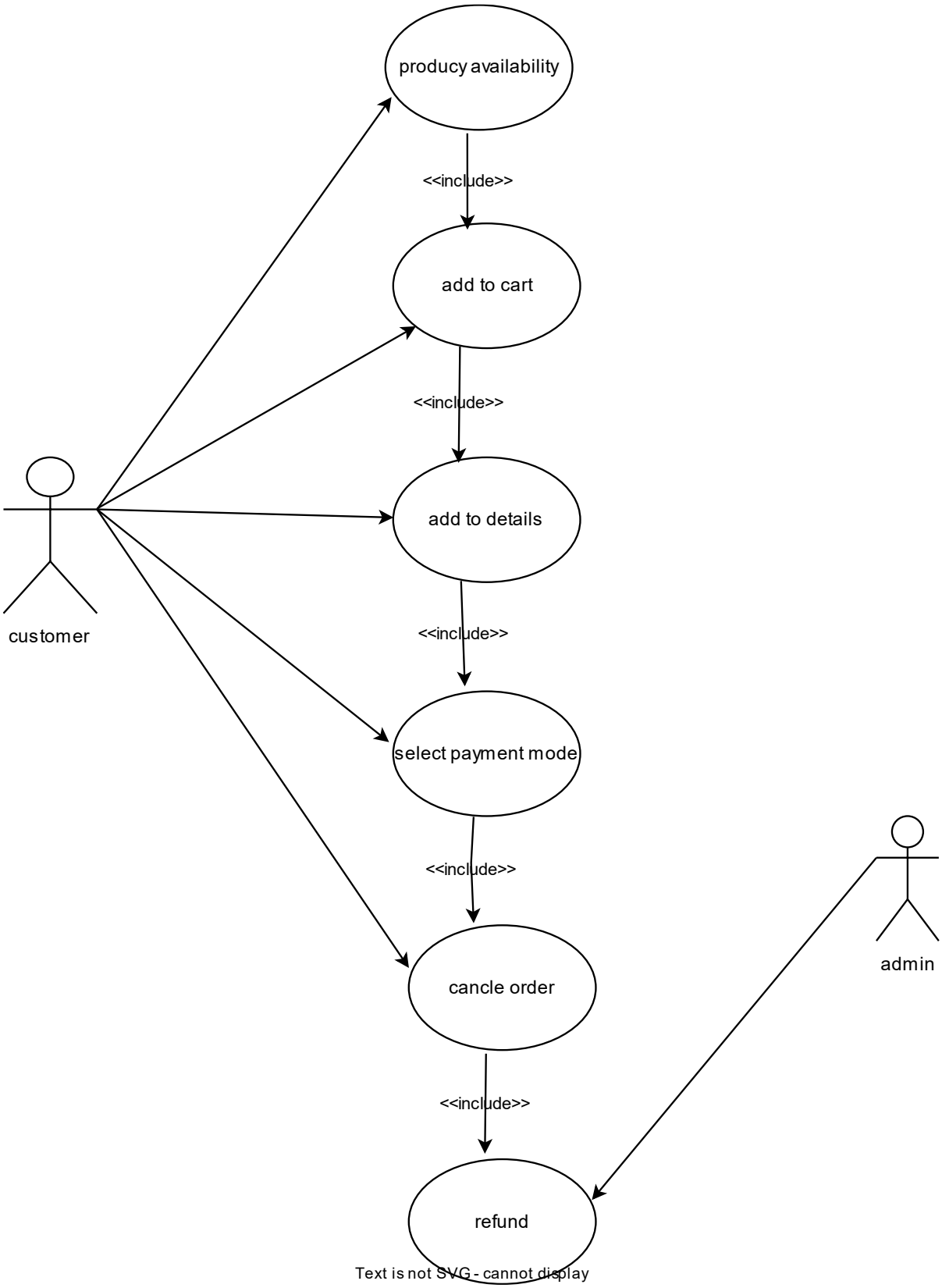


19 .draw usecase on online shopping product using COD?

20. Draw usecase on online bill payment system(paytm)?



21.draw usecase on online shopping using payment gateway ?


producy availability
<<include>>
add to cart
<<include>>
add to details
<<include>>
select payment mode
<<include>>
cancle order
<<include>>
refund
customer
admin
Text is not SVG - cannot display