Solid Principles:  
<https://medium.com/backticks-tildes/the-s-o-l-i-d-principles-in-pictures-b34ce2f1e898>  
<https://blog.algomaster.io/p/solid-principles-explained-with-code>

Desing Patterns:

(i)Strategy Pattern:   
<https://github.com/ashishps1/awesome-low-level-design/tree/main/design-patterns/java/strategy>  
[https://refactoring.guru/design-patterns/strategy](https://refactoring.guru/design-patterns/strategy%09)  
(ii)Observer Pattern:

<https://github.com/ashishps1/awesome-low-level-design/tree/main/design-patterns/java/observer>  
<https://refactoring.guru/design-patterns/observer>

(iii)Decorator Pattern:

<https://github.com/ashishps1/awesome-low-level-design/tree/main/design-patterns/java/decorator>

<https://refactoring.guru/design-patterns/decorator>

(iv)Factory & Abstract Factory Pattern:

<https://github.com/ashishps1/awesome-low-level-design/tree/main/design-patterns/java/factory>

<https://refactoring.guru/design-patterns/factory-method>

<https://refactoring.guru/design-patterns/abstract-factory>

(v) Chain of Responsibility:

<https://refactoring.guru/design-patterns/chain-of-responsibility>

(vi) State Design Pattern:  
<https://refactoring.guru/design-patterns/state>

(vii) Proxy Design Pattern:

<https://refactoring.guru/design-patterns/proxy>  
  
Problems:

(i)tic-tac-toe: <https://gitlab.com/shrayansh8/interviewcodingpractise/-/blob/main/src/LowLevelDesign/LLDTicTacToe/Main.java>

𝐅𝐫𝐞𝐬𝐡𝐞𝐫 𝐋𝐞𝐯𝐞𝐥 𝐒𝐲𝐬𝐭𝐞𝐦 𝐃𝐞𝐬𝐢𝐠𝐧  
  
1. Design a simple URL shortening service.  
2. Design a basic chat application.  
3. Design a file storage system.  
4. Design a simple social media platform.  
5. Design a simple search engine.  
6. Design a simple e-commerce website.  
7. Design a basic ride-sharing system.  
8. Design a basic video streaming service.  
9. Design a simple recommendation system.  
10. Design a basic food delivery app.  
11. Design a parking lot management system.  
12. Design a simple music streaming service.  
13. Design a basic online ticket booking system.  
14. Design a simple note-taking application.  
15. Design a weather forecasting system.  
16. Design a basic email service.  
17. Design a file synchronization system.  
18. Design a simple calendar application.  
19. Design a basic online quiz platform.  
20. Design a user authentication system.  
  
𝐄𝐱𝐩𝐞𝐫𝐢𝐞𝐧𝐜𝐞𝐝 𝐋𝐞𝐯𝐞𝐥 𝐒𝐲𝐬𝐭𝐞𝐦 𝐃𝐞𝐬𝐢𝐠𝐧  
  
1. Design a URL shortening service like [bit.ly](http://bit.ly/).  
2. Design a distributed key-value store like Redis.  
3. Design a scalable social network like Facebook.  
4. Design a scalable recommendation system like Netflix.  
5. Design a distributed file system like Hadoop's HDFS.  
6. Design a real-time messaging system like WhatsApp.  
7. Design a web crawler like Google.  
8. Design a distributed cache like Memcached.  
9. Design a content delivery network (CDN) like Cloudflare.  
10. Design a scalable search engine like Google.  
11. Design a ride-sharing system like Uber.  
12. Design a video streaming service like YouTube.  
13. Design an online food delivery system like Zomato.  
14. Design a collaborative document editing system like Google Docs.  
15. Design an e-commerce platform like Amazon.  
16. Design a recommendation system for an online marketplace.  
17. Design a fault-tolerant distributed database system.  
18. Design a scalable event-driven system like Twitter.  
19. Design a scalable photo-sharing platform like Instagram.  
20. Design a distributed task scheduling system.