

LLD 7 : How to approach design problems

and



Image uploading

→ FB

→ Insta

→ WhatsApp

→ Reels

IS & Memory

Postgres

Postgres
Redis
Memcached
Replica II

Curfit

→ Single Database →

(9:10)

Facebook

Cassandra
Tome

Gaming
New S Feed

Messages

Grocery
Notes
Web

My SQL

Image Uploading

User Service

Scalur Academy



forever

Topics

⇒ Query Pattern

SOR: Write

SOR: Read

⇒ Cassandra

→ How to approach LLD Problem

→ Amazon, Do an LLD design of a Pen
(Object Oriented)

LLD Interviews

(60 Mins)

fi Money



→ FSE working code is
not expected → test cases
trace ref

→ ACK to code an individual
class / interface ↗ sleep
↗ attribute

→ 30 min

1.5 hrs

30 min

going to ask
q

SDES

→ handle concurrent

→ Database

map > —

↑

...

10/10

Mod. -

Gathering Req ✓
Free Code ✗

LLD ✓
UML ✗

Tips and Techniques

- Gather Req, 100%
→ Clarify Req, 5-8 min
- Use Case Diagram, 5 min
- Class Diagram, 3 min
- Scheme Design, 3 min

→ In Seq
Order in
exact same
way

→ Please lead the
interview - Candidate
should lead the
interview

→ 50*

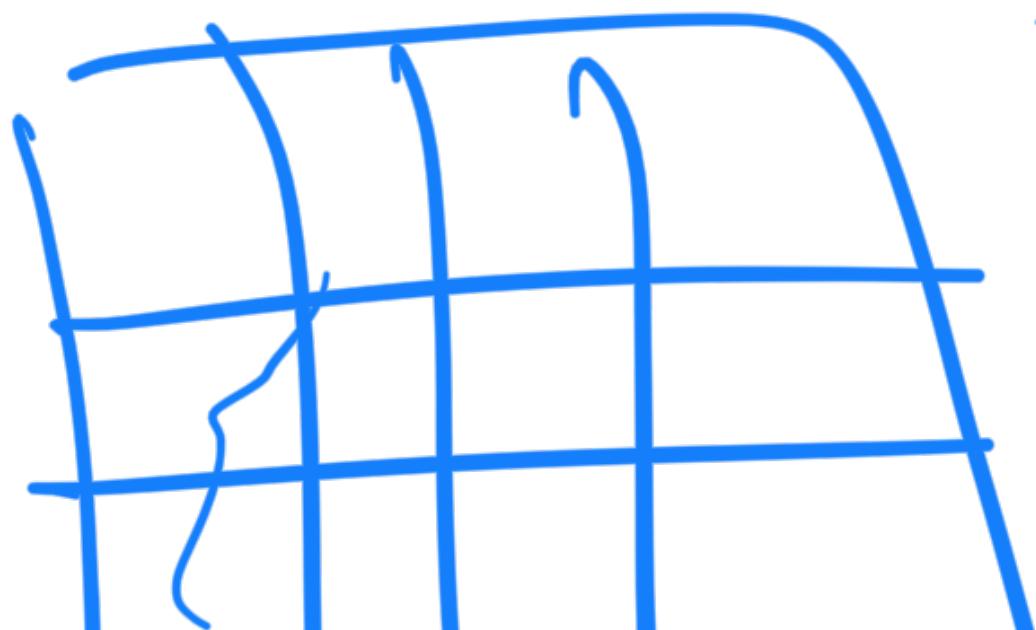
→ 25 mins

→ Code



Gathering Fig

→ Have a non ambiguous visualisation
of the working product



→ Sugget tangential

days - be creative

→ Since all my are gathered, ideally

~~1 2 3 4~~ "test them all"

ASSUME

Assume

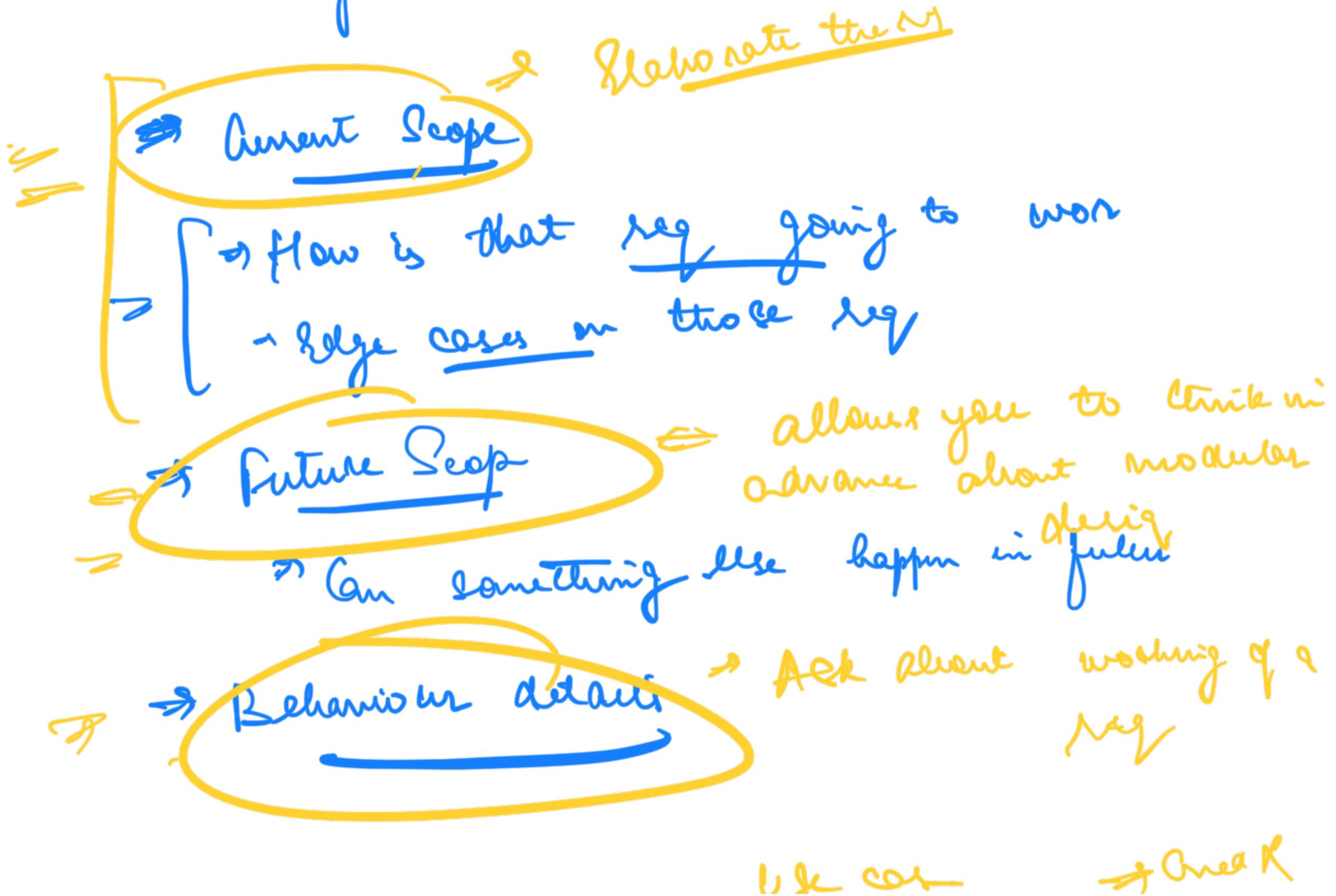
feel

Clarify the req

function in that

will have no impact

Don't ask any ↗
in your design



Use Case Diagram



→ proper semantics



Class Diagram

.. + ... Tool in the area

Step 0: find all the entities in memory

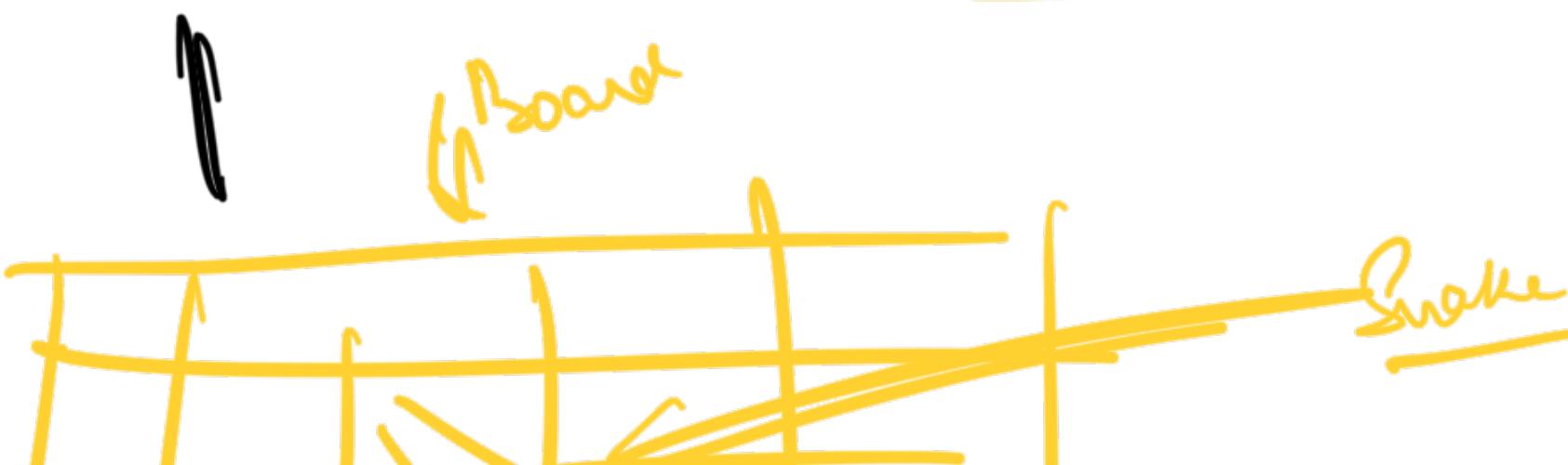


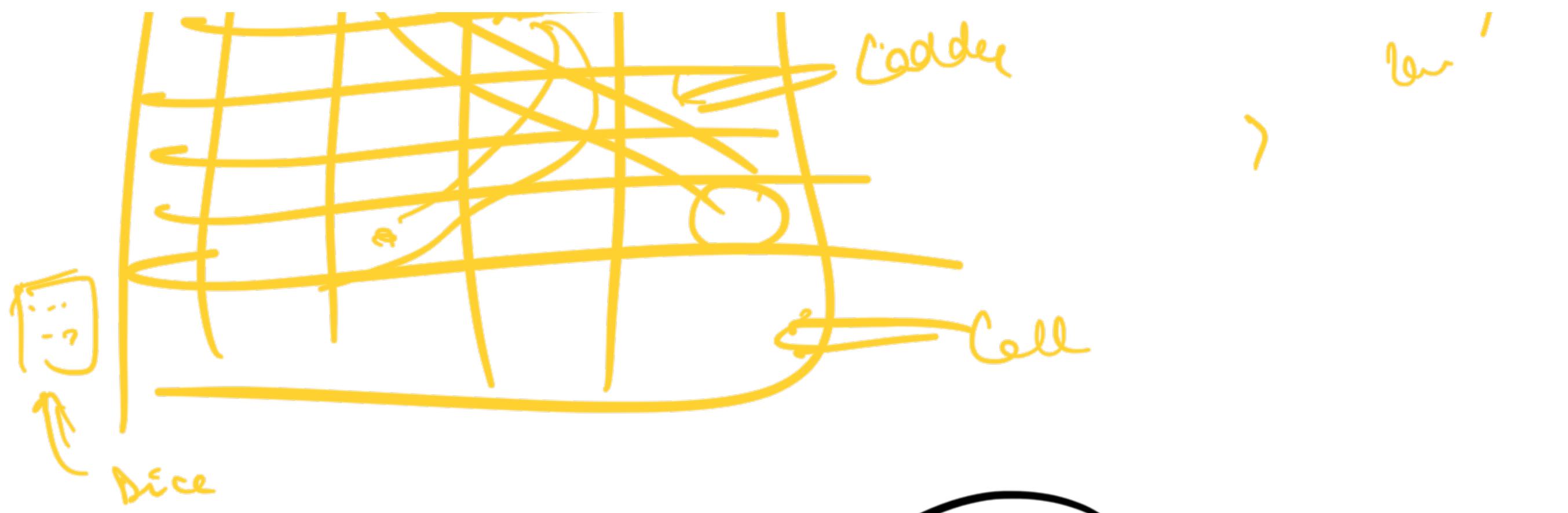
Step 1

: As Only a subset of those entities will become clear



C, Visualization

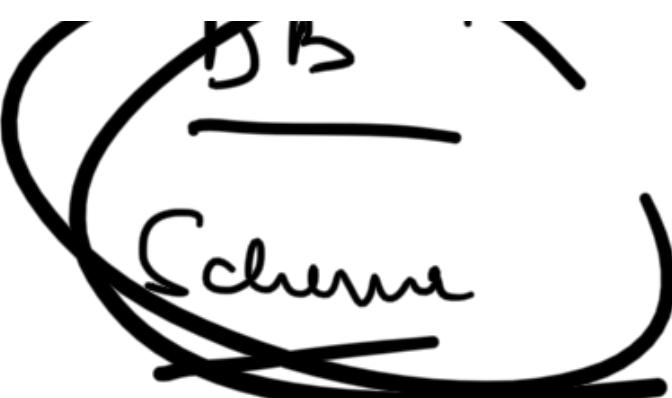




→ Only classes of Models (entities) are expected in the environment

User Controller, User Sensor, User

→ Don't do anything that is
not needed.



Talk about strategy / adaptor DP





Scheme Design

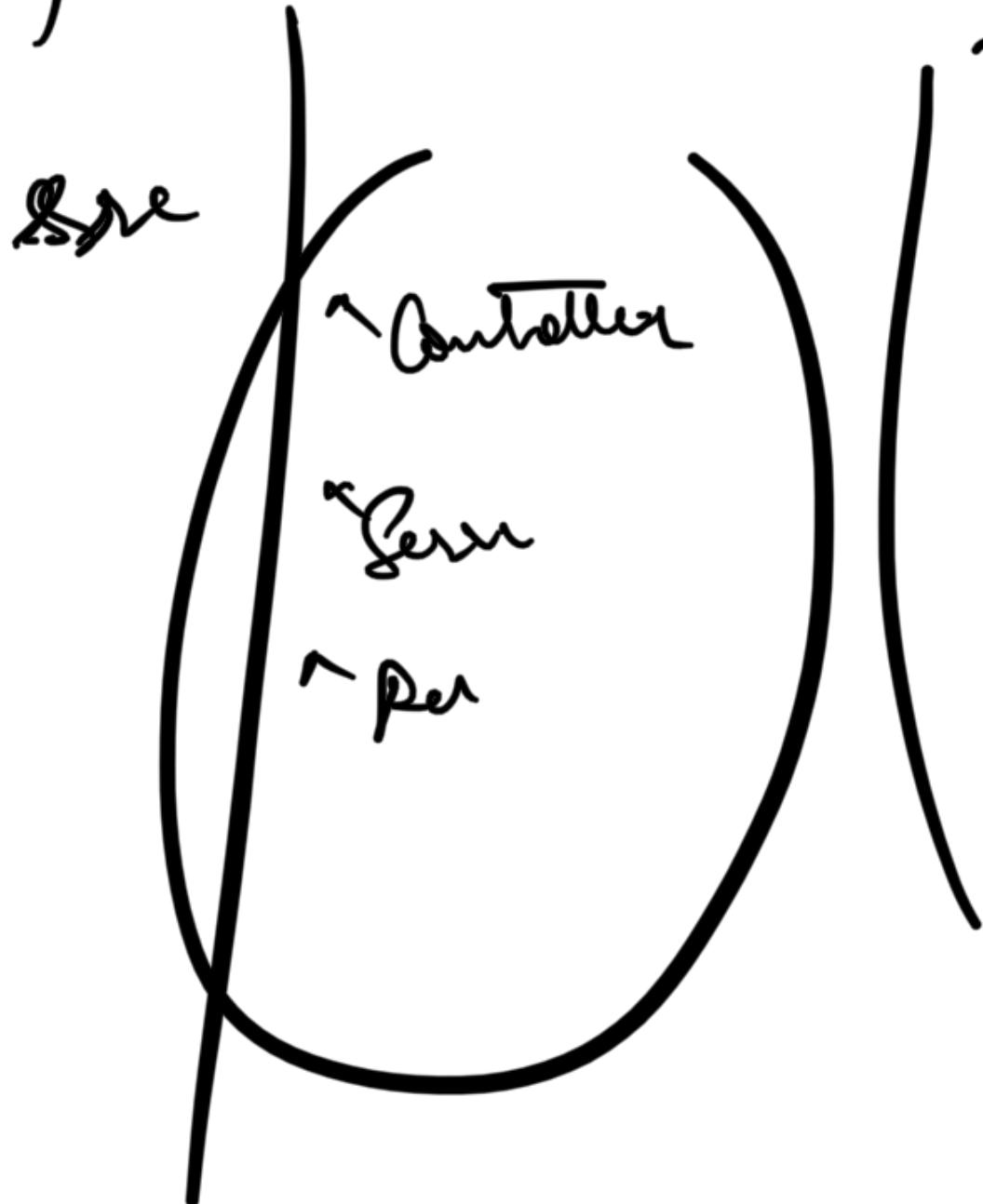
- Find the correct cardinality
- How are you up rel²



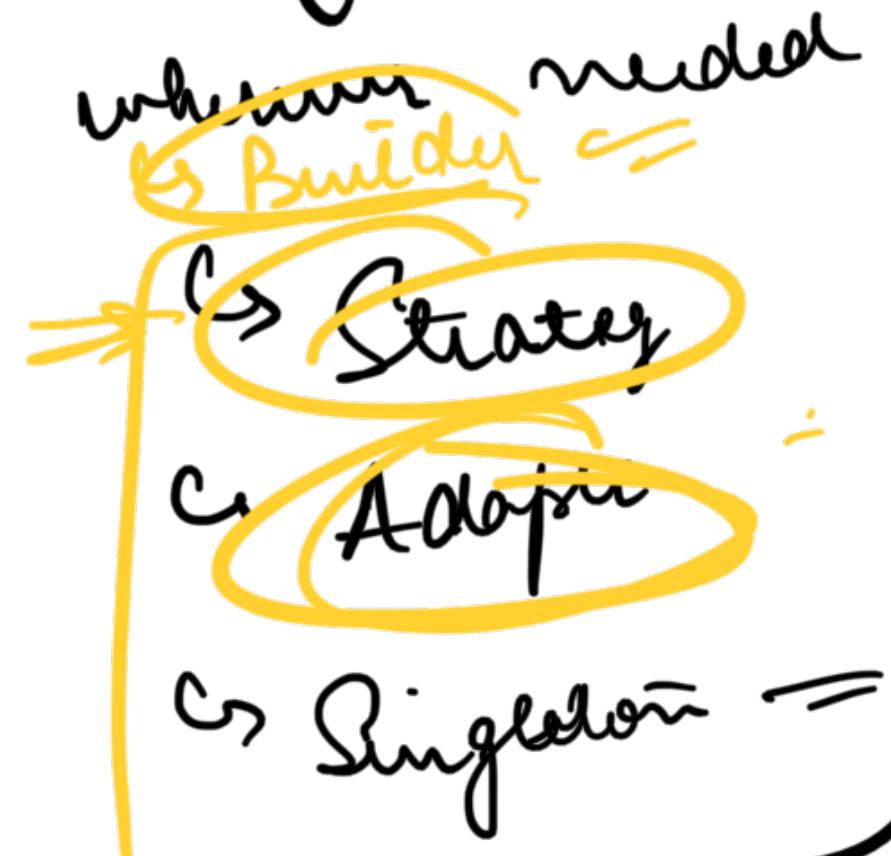
Code

2

① Project Structure Design



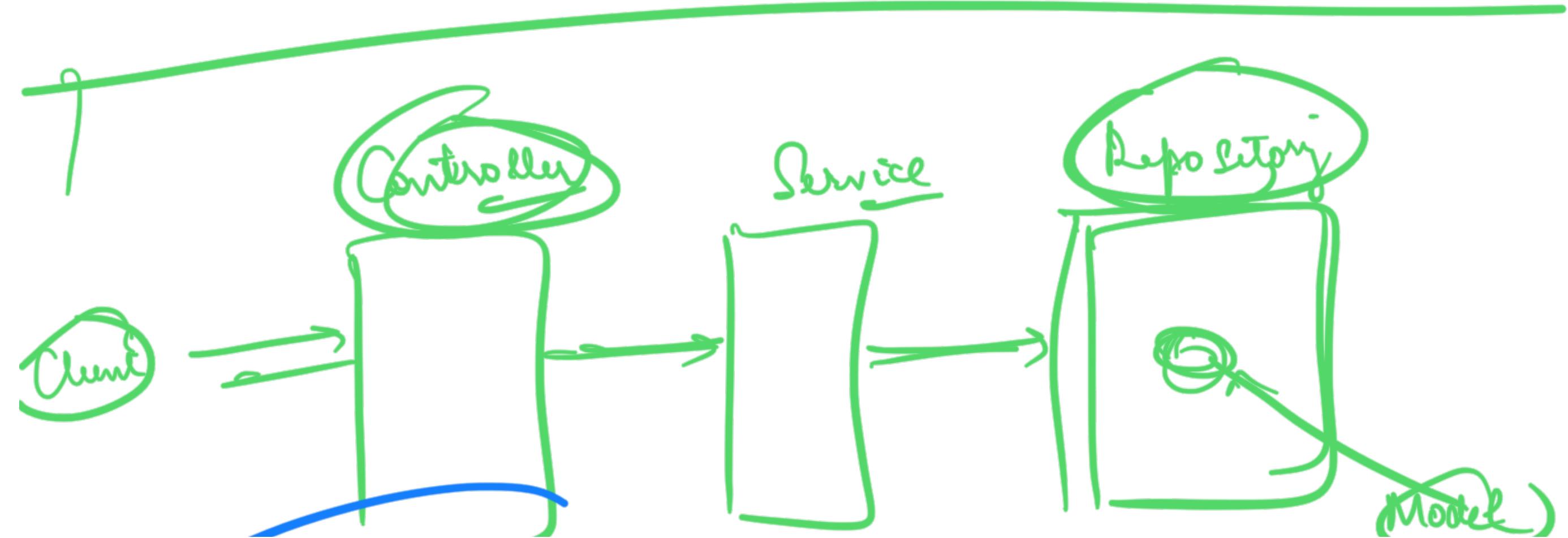
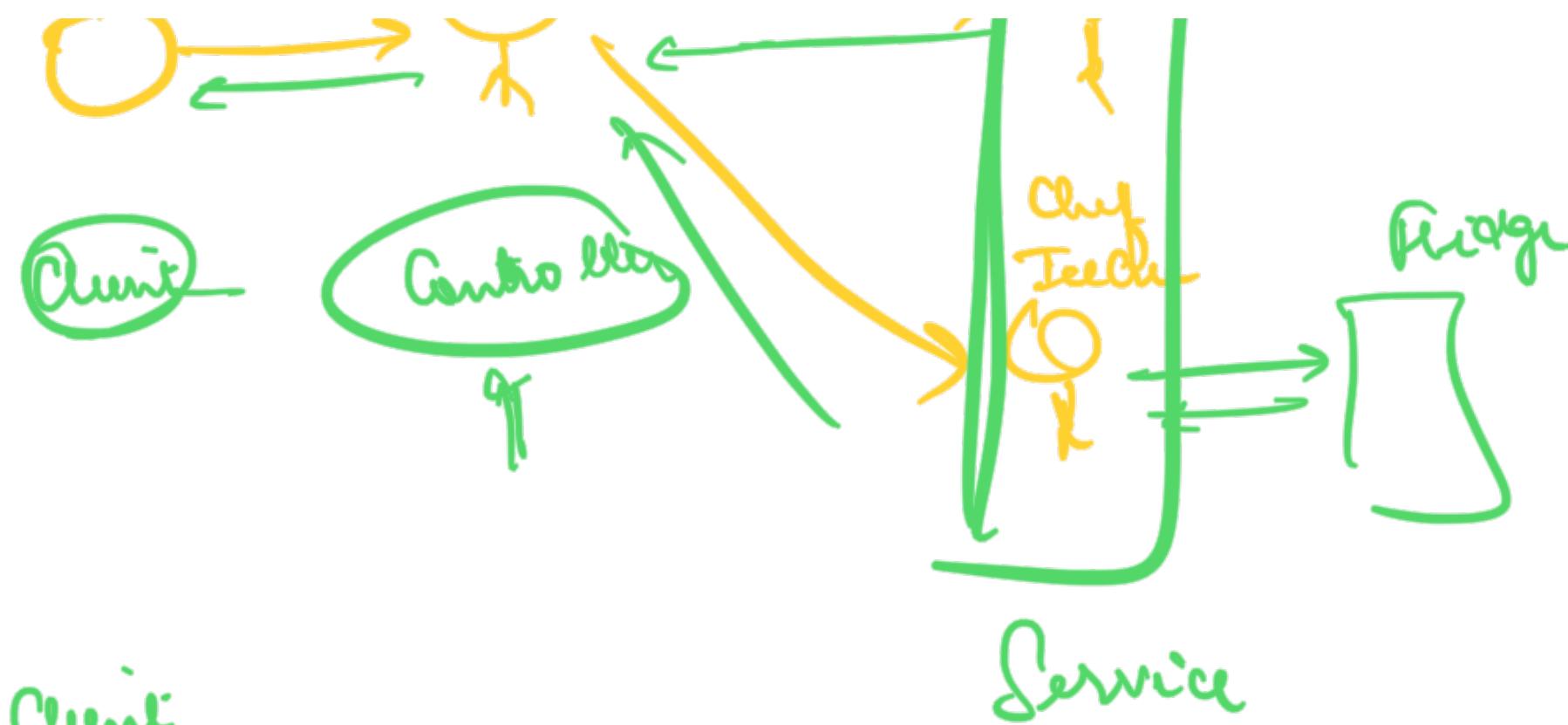
② Following Design Pattern



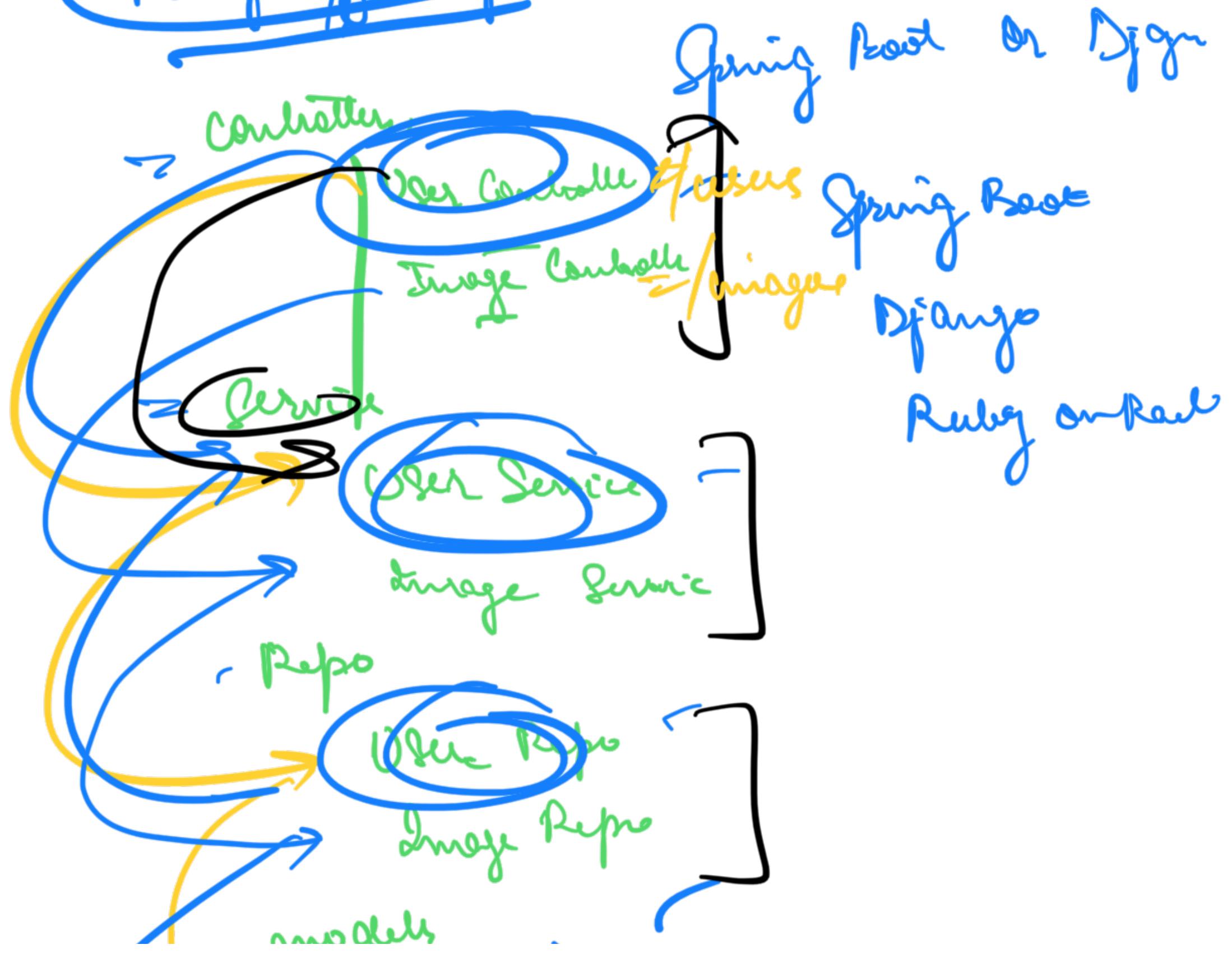
③ Code to interfaces

Project Structure

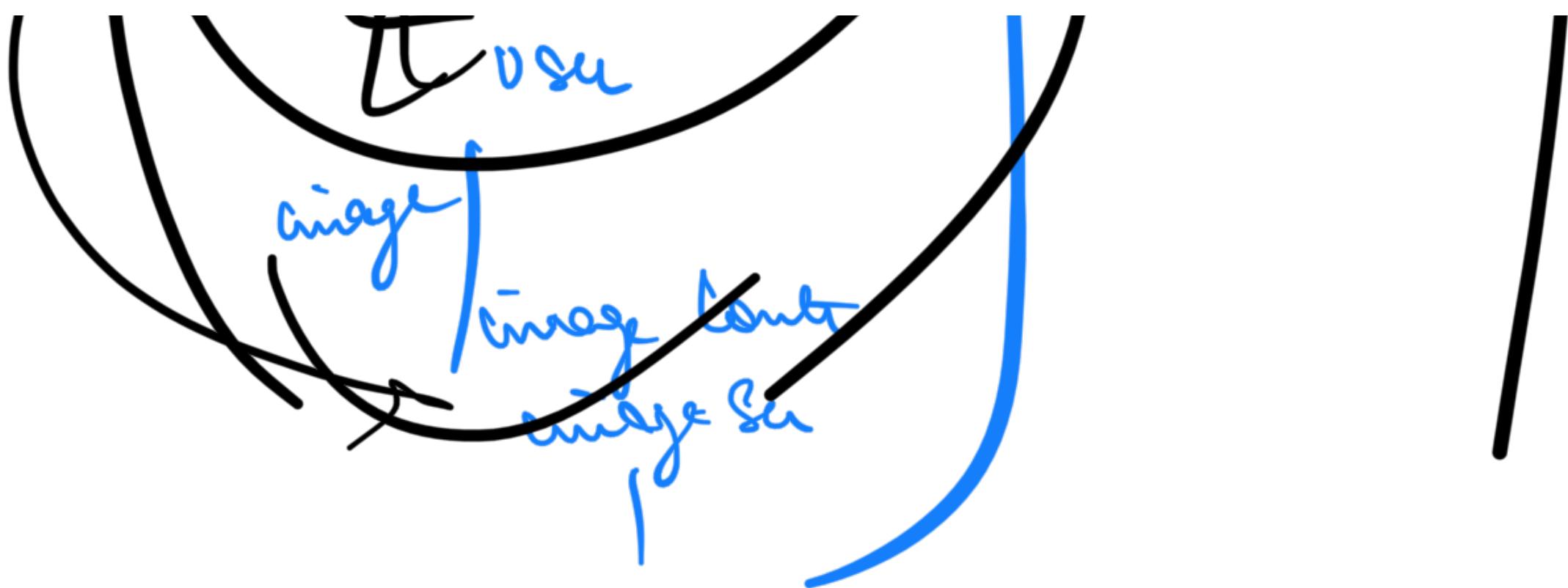




Package by layer







P

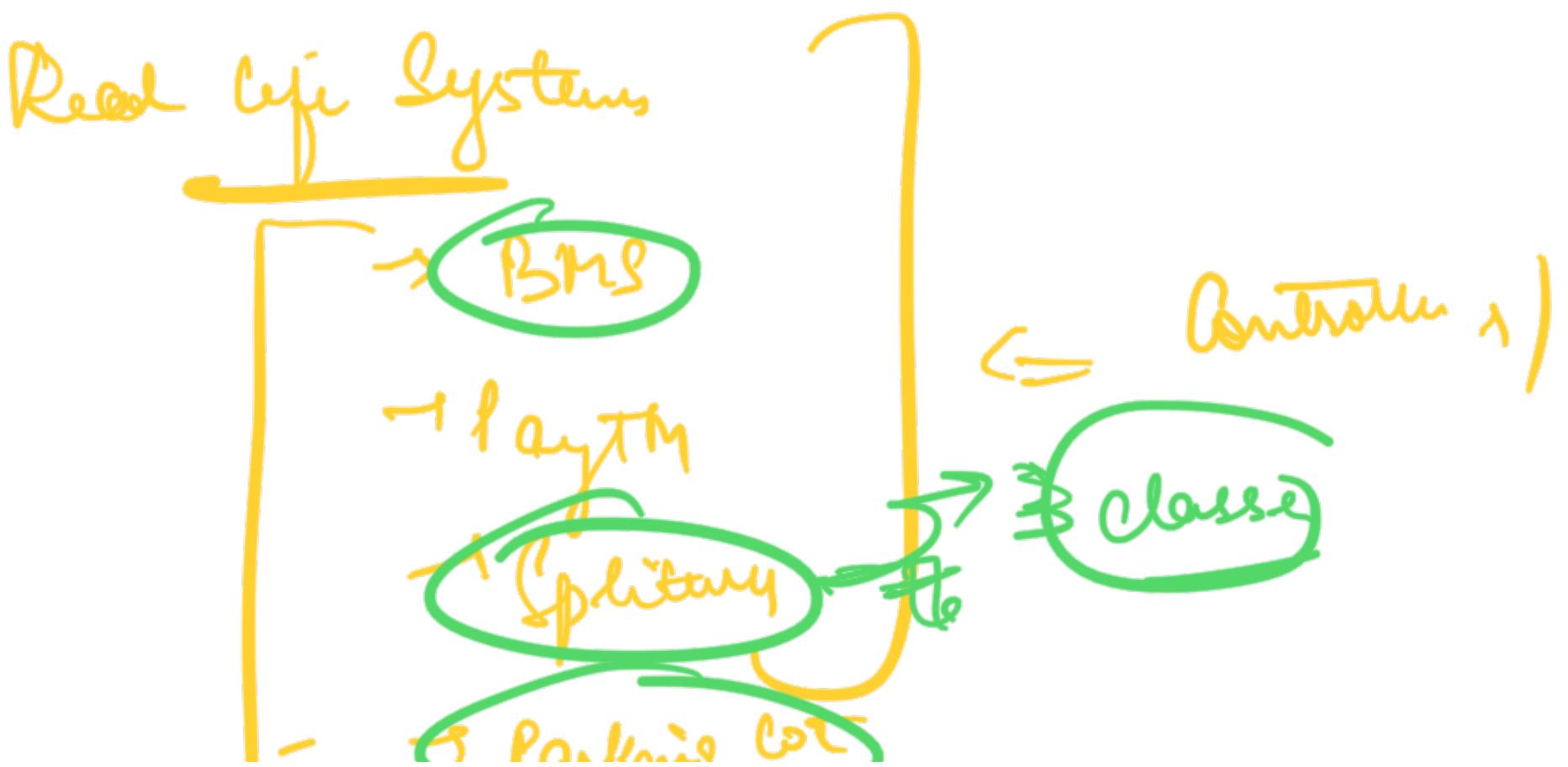
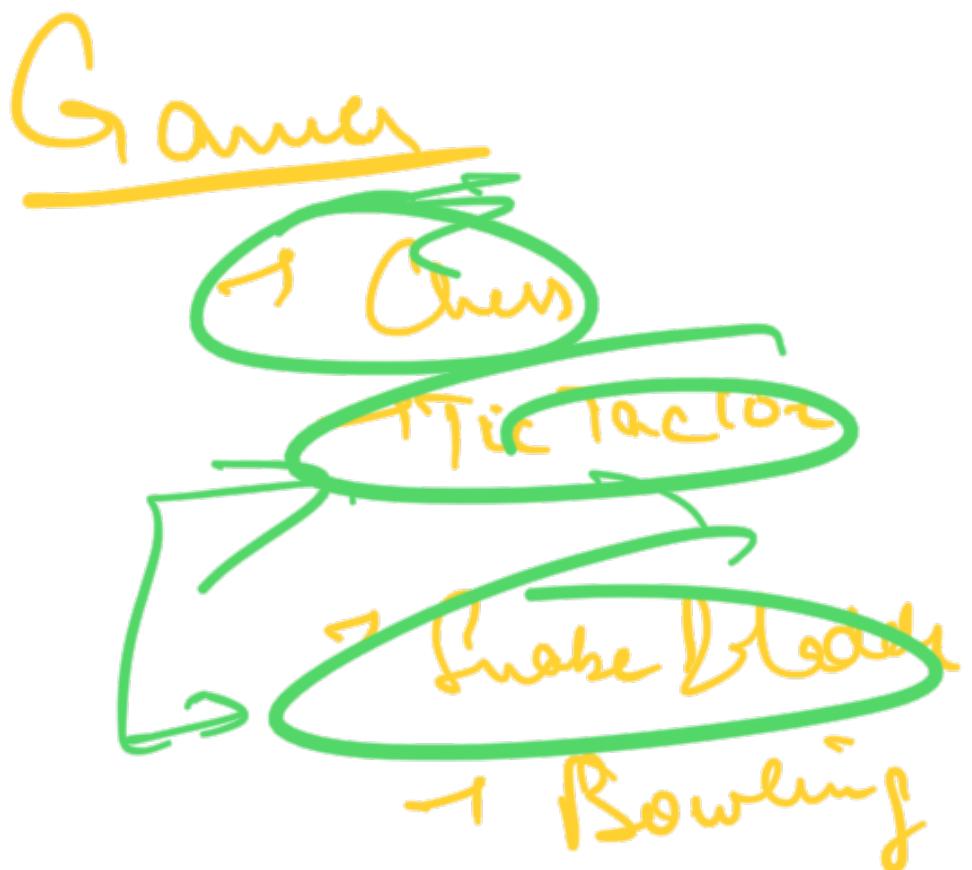
Type of lens problems



→ Notebook

... now ...

- HTML structure





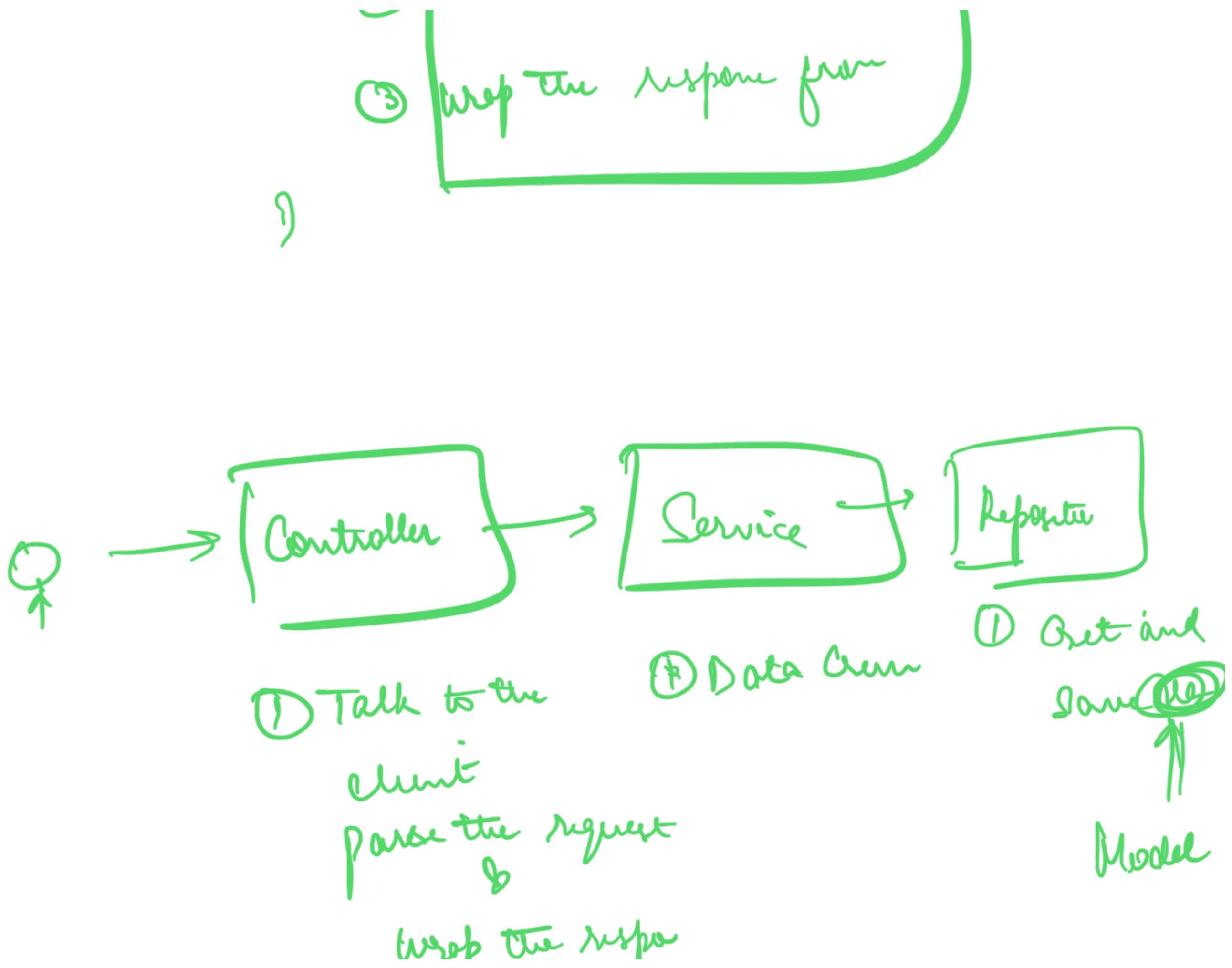
Eng Problems



→ Dust Cache



- ① parse the req from
- ② take necessary action





DUMPS