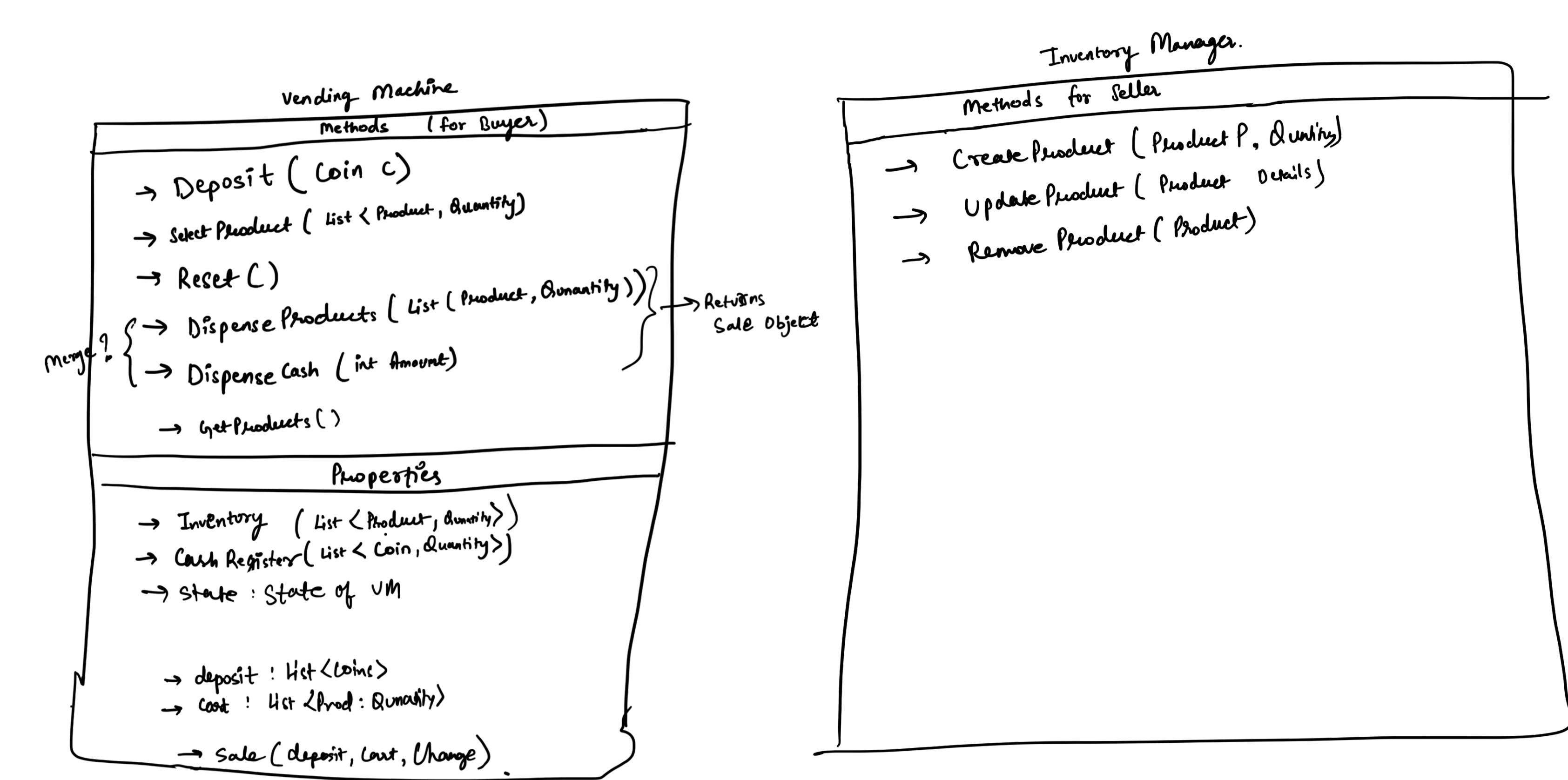
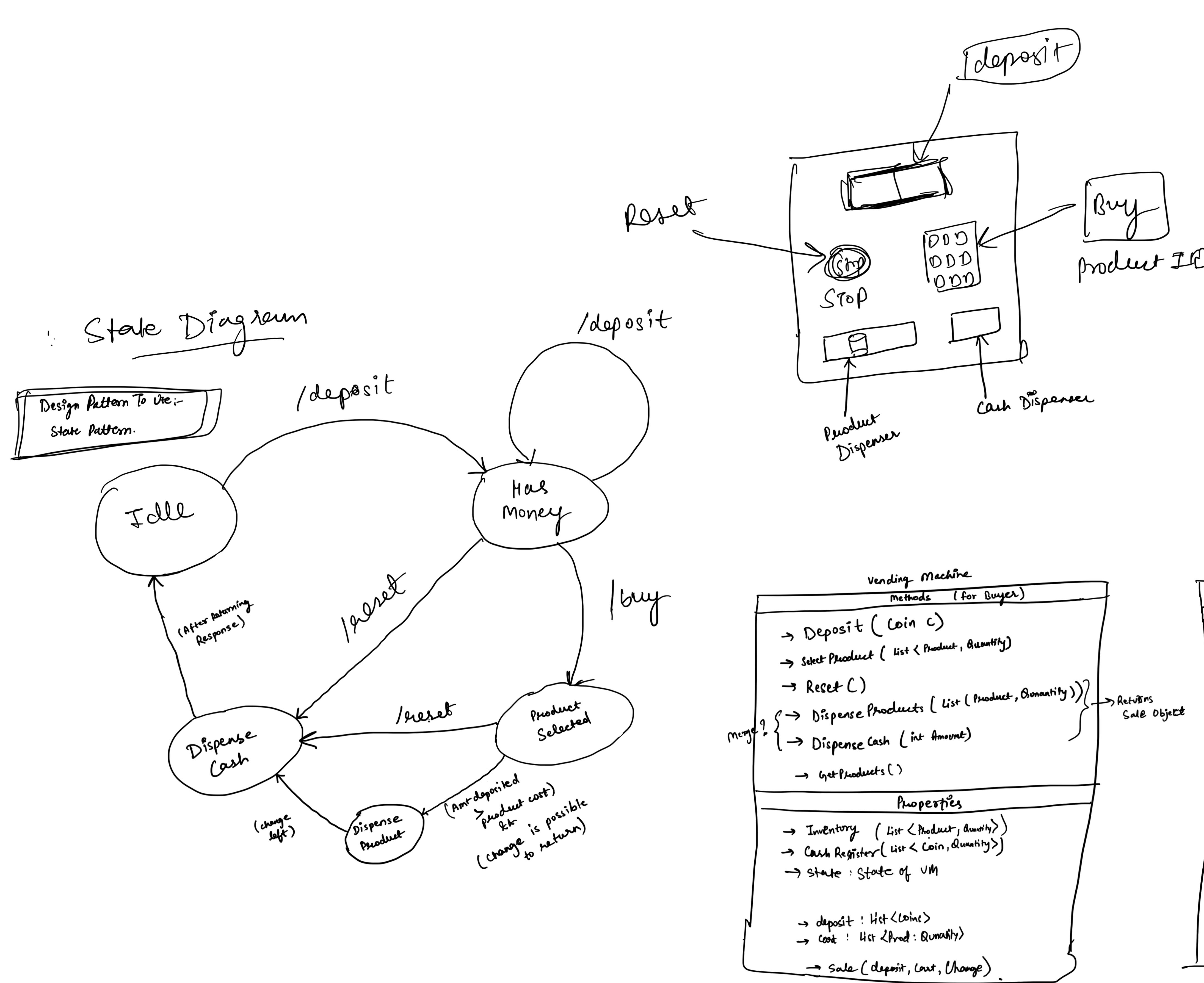


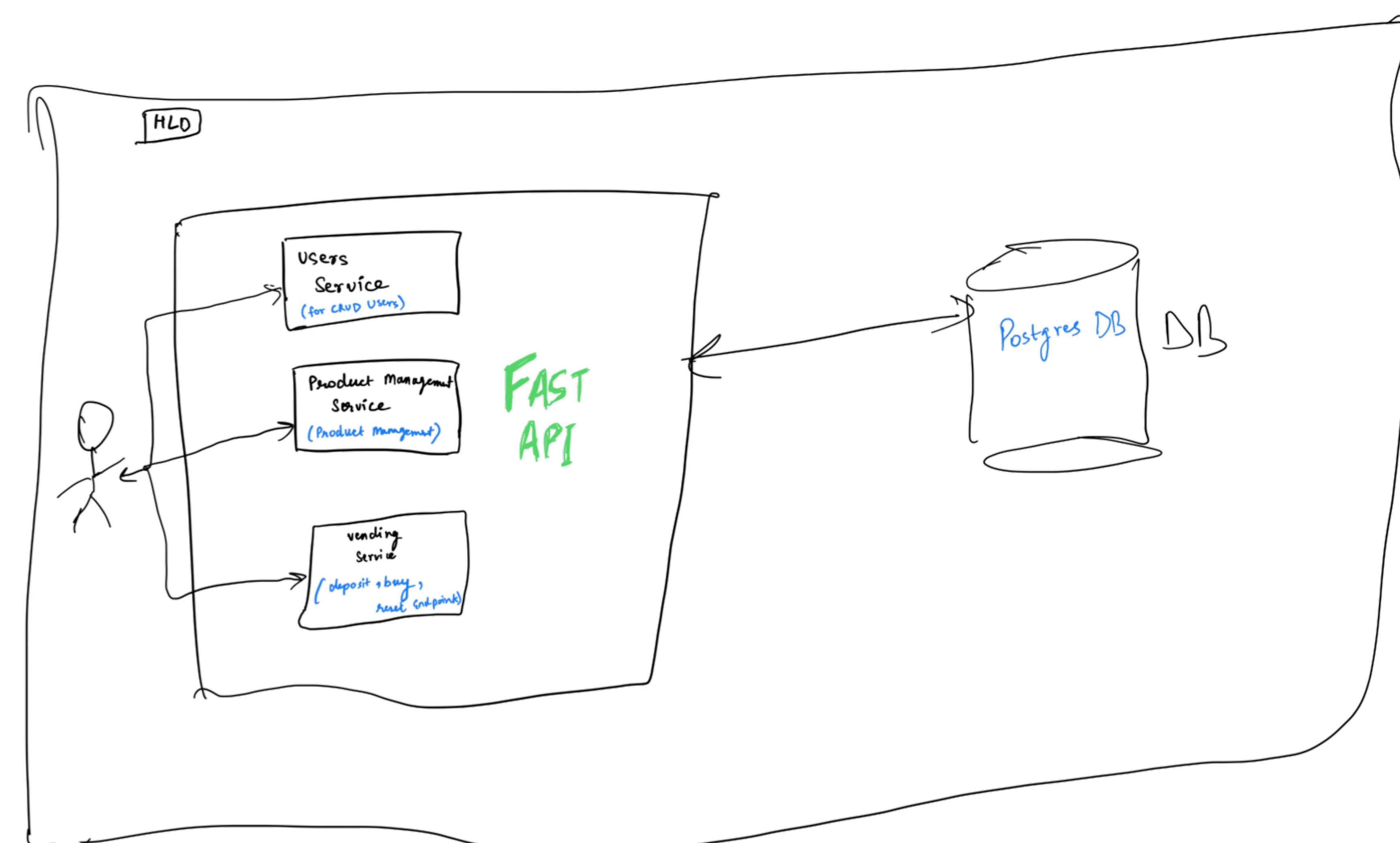
Steps:-

- Requirements
- Capacity Estimate?
- HLD
- DB Design
- API Design
- Components
- Scalability?



Goal

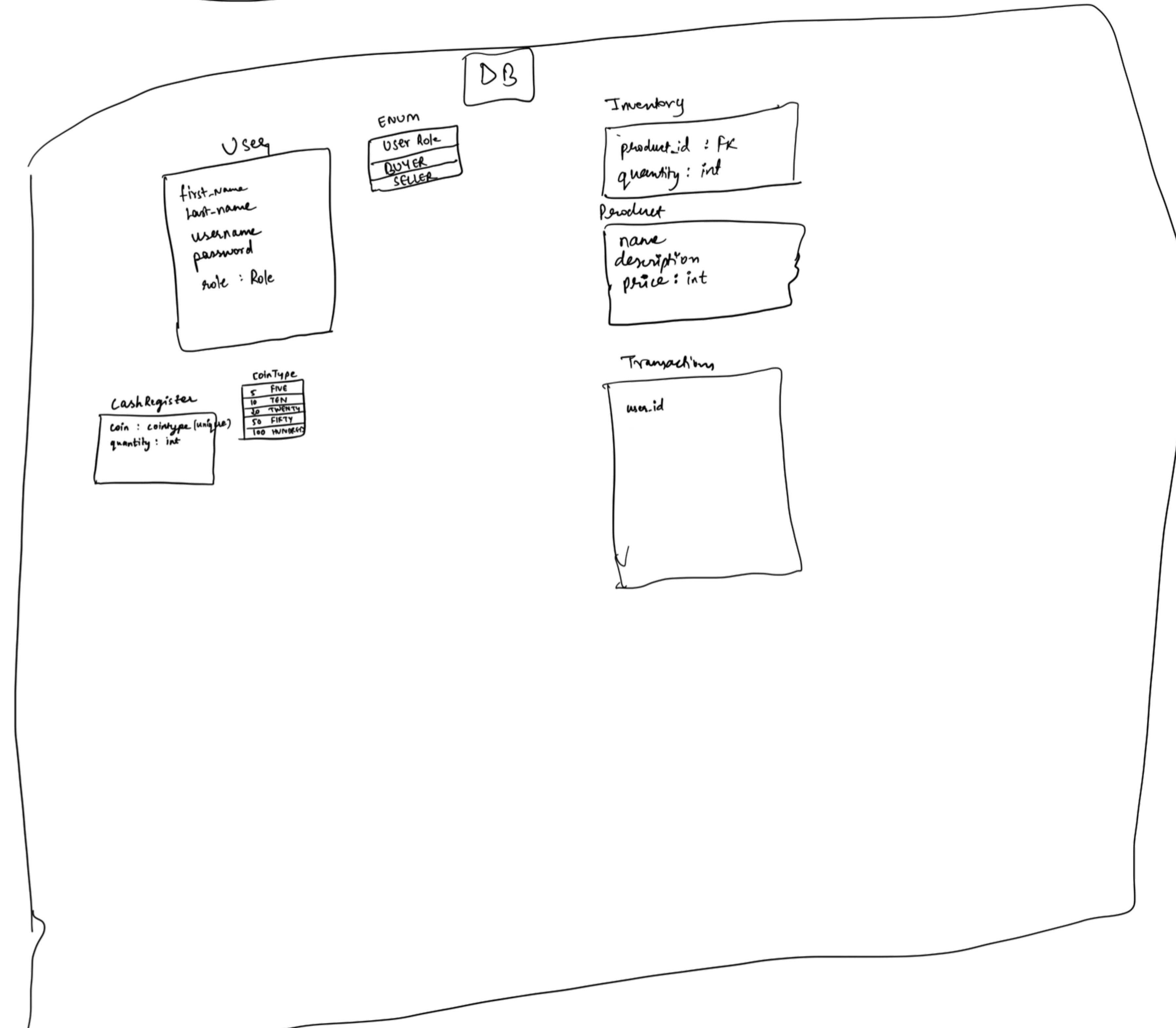
- ① User Roles
 - Buyer: Can deposit coins to make purchases
 - Seller: Can add, update, remove products
- ② Coins :- 5, 10, 20, 50, 100 cent coins



Requirements:-

→ functional:-

- CRUD for users
 - Create User → Public
 - Update User } self auth.
 - Remove User }
 - Get User
- CRUD for Product
 - Get products !- Public
 - Create } Product created by User (seller)
 - Update }
 - Delete }
- /deposit : for Buyer Users } Auth
 - Add Coins (5, 10, 20, 50 and 100)
- /buy : for Buyer users } Auth
 - product Id + Amount (Input)
 - Output :
 - Spent:Amount
 - products
 - Change
- /reset :- Reset the deposit



→ Non functional Requirements :-

- Only one user can use the machine at a time. How to enforce this?
- Deposit, Buy and Reset Endpoints can only be called by one user at a time.
 - Lock the machine?
 - How to make sure no two different users can use these endpoints at a time?
- Will the user need to be logged in to use the apis?