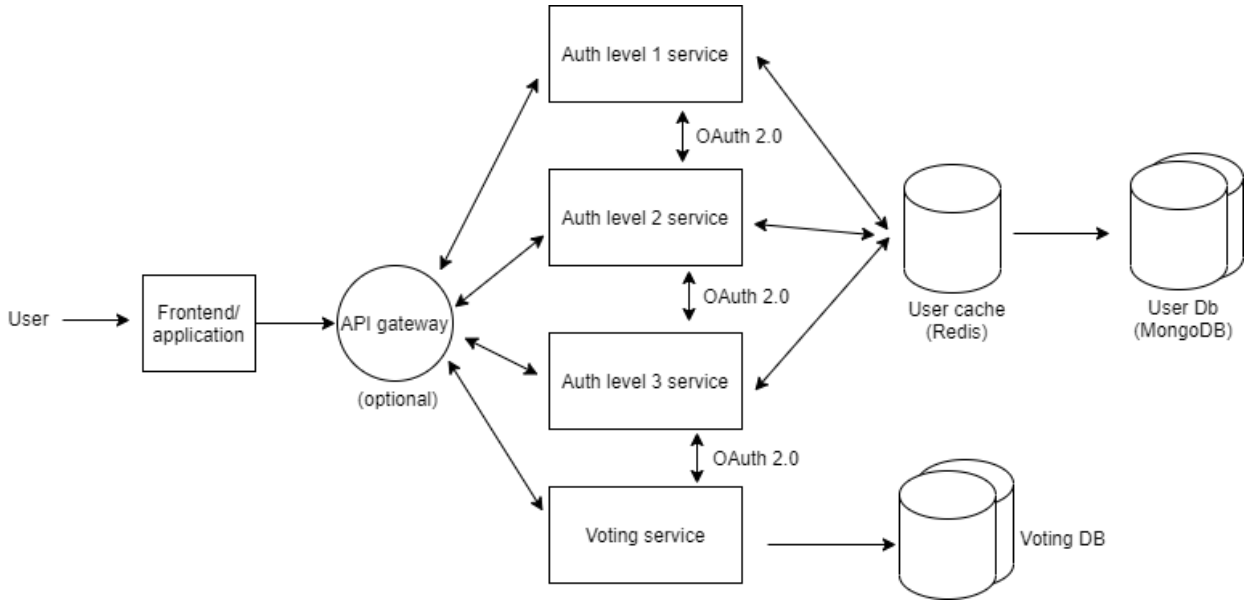
System Design



1. **Frontend** - This is the user interface through which the users will interact with the system.
2. **API Gateway** - It provides a single entry point and acts as a router, routing the requests from the frontend to appropriate service at the backend. Thus, API gateway provides an abstraction layer between the UI and the backend. However, it can cause problems like single point of failure and performance overheads (because of increase in network calls)
3. **User authentication***
    a. **Auth level 1 service** - This service will authenticate users based on their Aadhar number, password and captcha (to prevent DDoS attacks)
    b. **Auth level 2 service** - Responsibilities of this service are generating, encrypting and communicating OTP's over registered Email ID, decrpyting and verifying the same. Adding OTP layer will ensure that even if someone gets a users password by some malpractices he/she cannot access the account without access to the email ID.

Additional features - Encryption of OTP's wil prevent shoulder surfing and phising as well

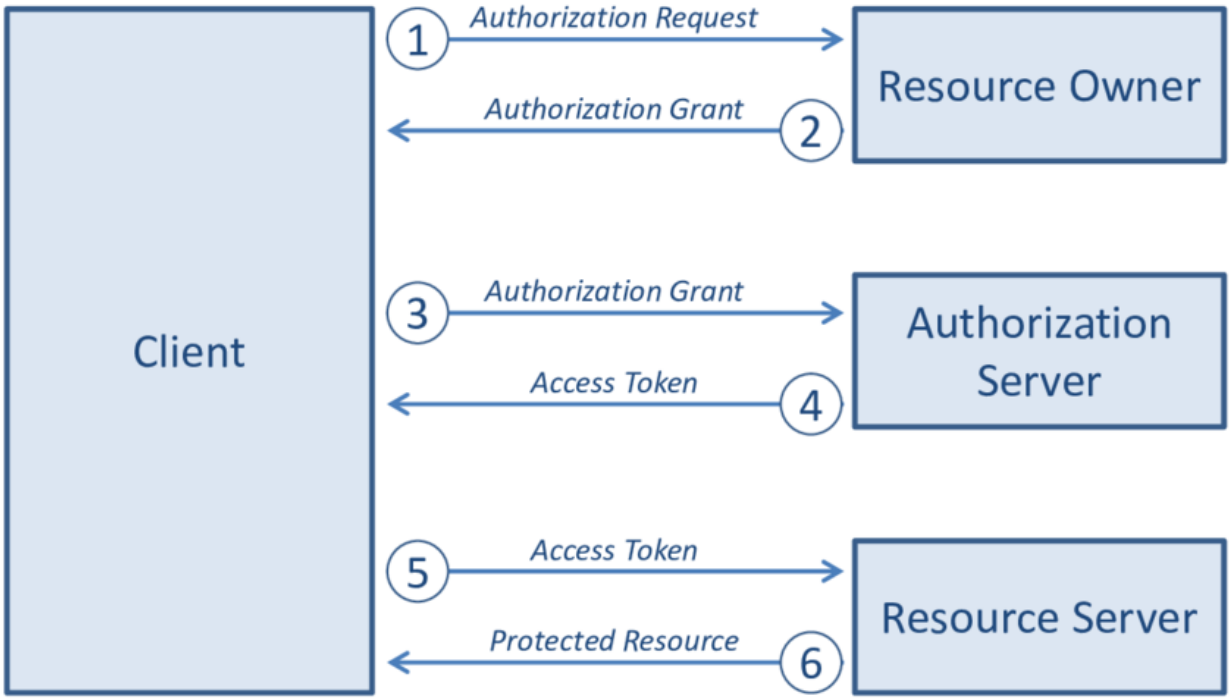
c. **Auth level 3 service** - This service will execute a CNN based model on input images obtained from the frontend. It will then run face detection as well as semantic analysis on these images. This will help in uniquely identifying the person.
Additional features - Semantic analysis will help to check if user is under some influence or pressure thus preventing forced logins.

After succesful authentictaion a JSON web token (JWT) will be stored in the users browser storage.

4. **Voting service -** Only authorized users with a valid JWT will be allowed to access this service. Voter requirements handled by this service -
   a. Application for new voter cards
   b. Updation requests of existing voter cards
   c. Casting votes

   Governing bodies/ commision requirements handled by this service -
   a. Approve new voter requests
   b. Submit form with final shortlisted candidates
   c. See statistics and counting after elections

5. **OAuth 2.0 -** OAuth 2.0 is the industry-standard protocol for authorization. It will facilitate secure communication between the various microservices present in the system. OAuth will be implemented with JWT's in the background to make authorization of users at different levels in the system possible. Below is a example of the workflow of this protocol.



6. **User cache -** Redis will be used to cache frequently accessed data to improve performance of the system. Redis stores this information in the form of key value pairs (Like a large JSON object). In case of a cache miss, User DB will be accessed.

7. **User database -** Sample schema will be as follows (may modify during implementation)

   {

   Aadhar_number: (string, required),
   Full_name: (string, required),
   Date_of_birth: (date, required),
   Password: (hash, required),

   }

8. **Voting database -** Schema yet to be decided

Security of the system

This system consists of three factor authentication (3FA), and propses new security concepts like encryption of OTP's and semantic analysis of users during authentication. All these will ensure complete security and the system will be free from DDoS attacks, phishing, shoulder surfing and also prevent forced logins. Also Oauth 2.0 protocol will make communication between services secure.

Scalability of the system

The system is based on microservice architecture and designed to scale vertically. Because each service is independent, failure of one will not affect peformnace of others. MongoDB being a cloud based storage will also scale up according to the traffic without we having to care about space and performance.

Future scope - System can be scaled horizontally by using multiple instances of the backend servers coupled with load balancing and NGINX. Databases can aslo be scaled by sharding and multiple instances.

Sample Product Workflow