This application is going to be used to create bookmarks that will have it’s own structure. This will allow users to have bookmark collections. With its own template. Such as bookmark collection of coding problems with its own structure. Bookmark collection of movies with its own structure.

Each structure/template will have mandatory and optional fields.

You can share and use templates from other users. Each template can be public and private

Batch operation to modify, delete, update templates should be available

How things will work

You login / use locally

Your local device is going to hold a client key if using online

Use JWT for now. Implement your own security check in future

DB Model

1. Each user is going to be in db record.

Users will have its general info such as name, dp etc etc

1. Template is going to have its own table

Template will have unique ID along with other info such as who created it, the structure, etc

It can be public and private. If its public then it is going to be visible to all other users

1. Bookmark

Bookmark will have id to link it to the user whom it belongs to as well as the template it is using along with other details

1. Images are going to be stored in template ID folders IF a template has support for image, it can have sub folders if it has many fields that needs image

Main Services:

1. Public gateway
   1. Used by client to interact with the backend.
   2. Will have to send client ID to verify who the user is, it will be invalid after certain amount of time
   3. Public gateway will also have security check to validate the requester using the client ID and then forward the request to other services
2. User service:
3. Bookmark service:

Communication will be figured along the way

1. Login/Signup : gateway.login(email,password) -> userService.login();
2. Create/Get Template : gateway.createTemplate(userinfo,template info)->bookmarkservice.createTemplate(userinfo,templateinfo)
3. Create/Get bookmark : gateway.createBookmark(userinfo,templateinfo,bookmarkinfo)->bookmarkservice.createBookmark