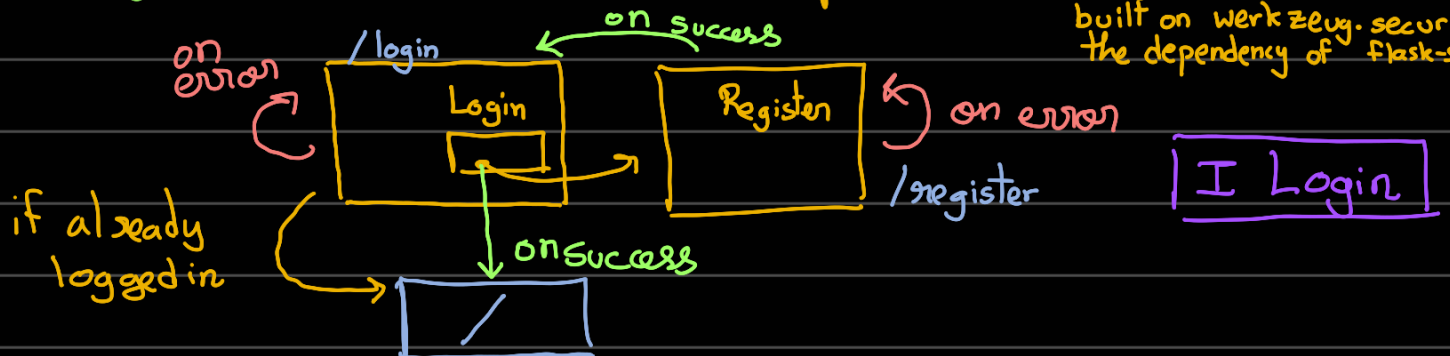


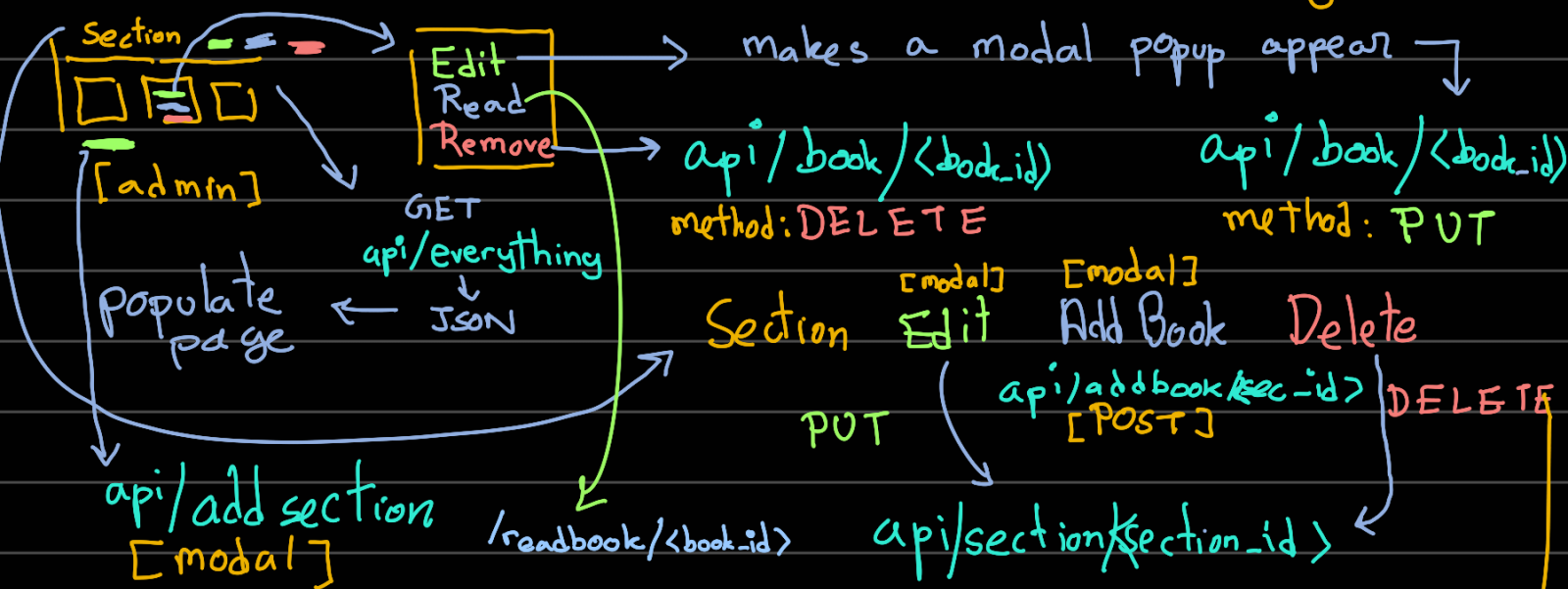
Bootstrap + Vue, Bootstrap-Table, Vue Router, ChartJS

- Login Page: POST fetch Validation & Vue param update. No refreshes.
- Register Page: sessionStorage for form, client as well as server side validation for all fields, POST submit via fetch, no refresh.
- Error updates are shown with a fade in & out for clarity.

Login Backend: Salted & Hashed password storage. [custom code built on werkzeug security, the dependency of flask-security]



Home Page: Simple custom RBAC decorators used. Cookie based authentication. Vue Router used for making both SPA.



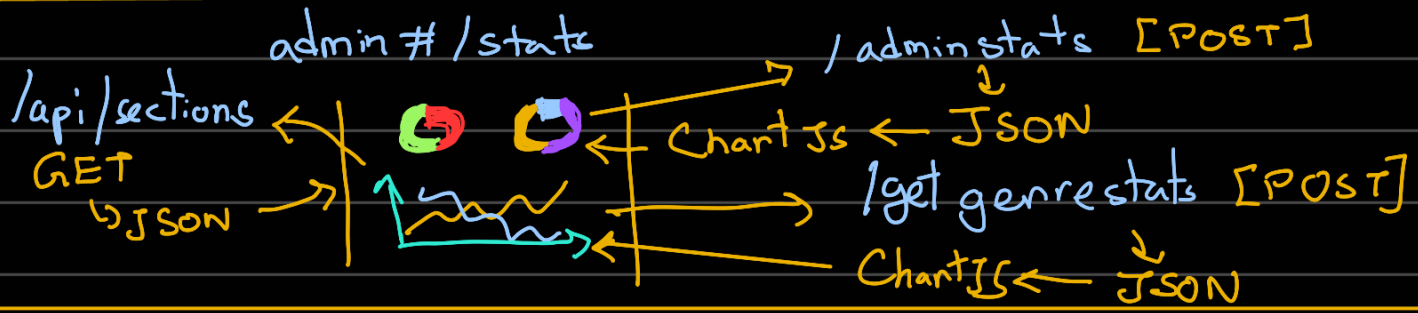
/admin #/requests

Requests waiting approval

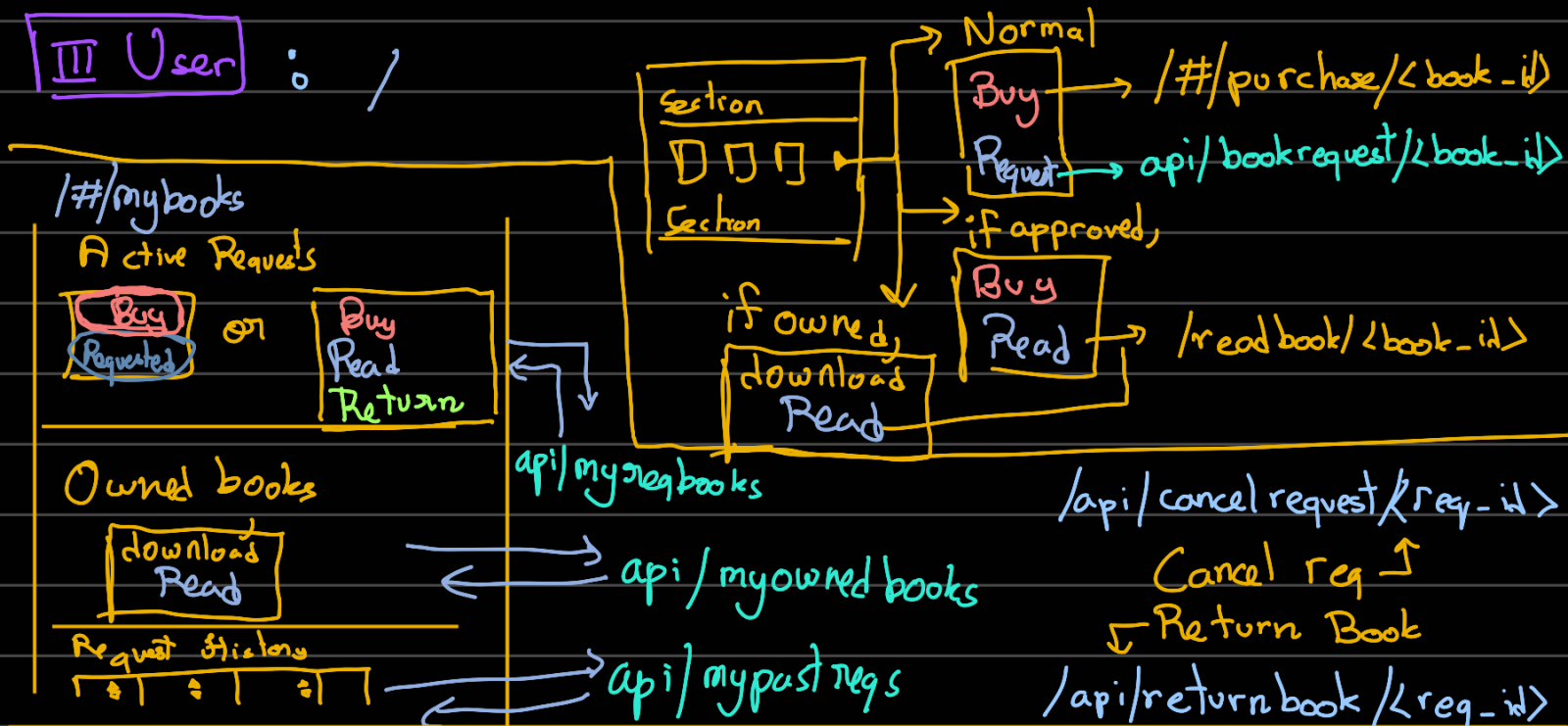


JS sorting & searching via bootstrap-table
makes JSON out of selected rows & sends POST to api/approve or api/reject
GET
/api/pendingreqs populate table

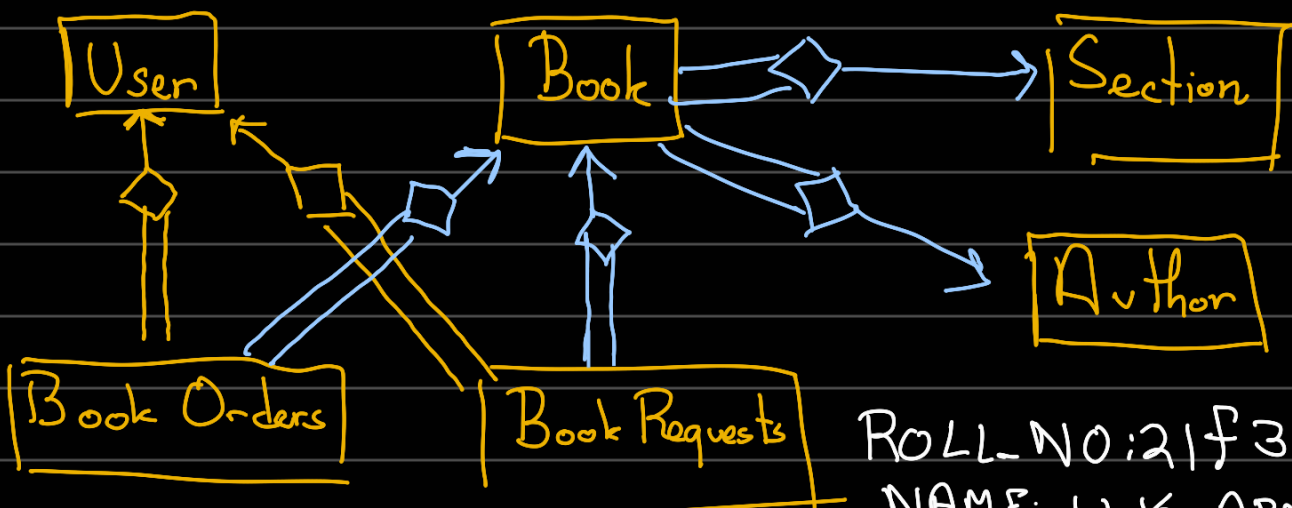
GET
Active Requests uses /api/active reqs → populate
↳ /api/revoke → revoke a JSON array of reqs



III User :



- * Almost all API endpoints & the request pattern is shown above.
- * My approach to the PS was simple: Make reactive pages Fully templated in VueJS & fetch any data with API calls.
- * Redis caching, Celery scheduled crontasks were both implemented. Users get daily reminder emails & admins get monthly reports.



ROLL NO: 21F3002715
NAME: U.K. ARVINDAN