Step by Step Guide - React & Redux Apps
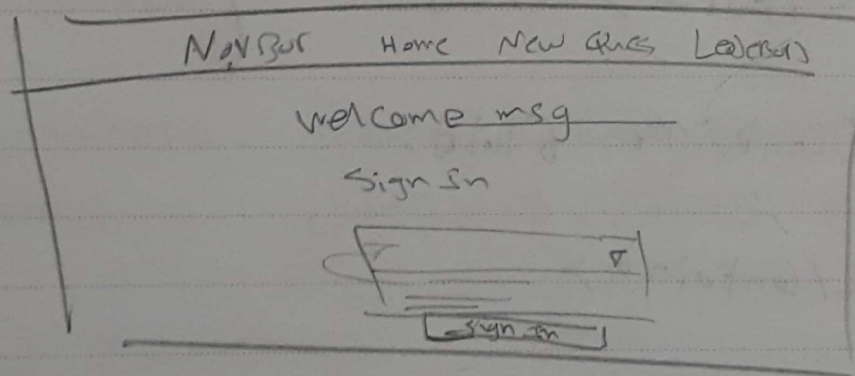
planning Stages
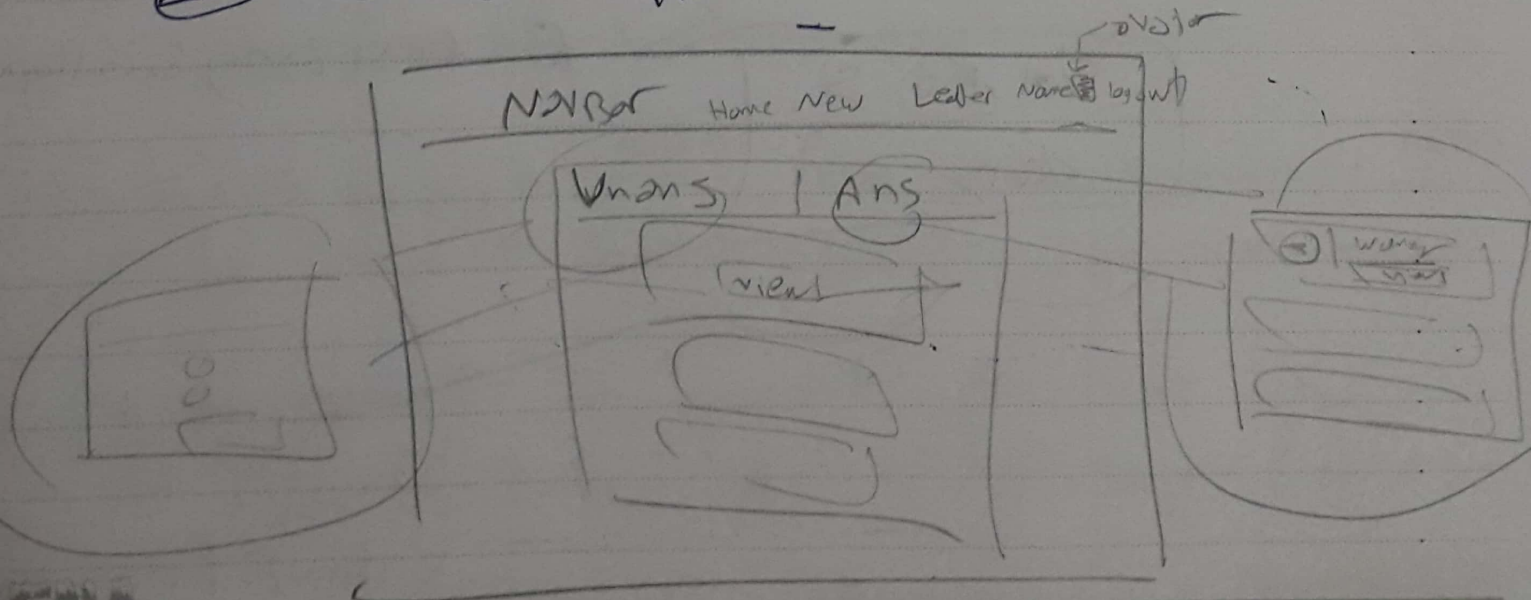
1. Identify what Each view should look like.
2. Break Each view into altierarcy of Comps.
3. Determine Events Happen in the App.
4. Determine what data Lives in the Store.)

(1) Identify what Each view should look Like
(Draw All of the view of the App)

(I) First view ı Login view انشاء ١ Login كلاس صفحة


```
NavBar    Home   New Ques   Leaderbon

         Welcome msg

         Sign In

         [                ▽ ]

            [ Sign In ]
```

(II) Second view ı Home View


```
NavBar   Home New   Leader  Name  log out)

      Unans, | Ans

         view      →
```

unanswerd ques

III) Poll : Third view (unAnswered)

```
┌──────────────────────────────────────┐
│  NAV Bar                             │
│   ┌────────────────────────────┐     │
│   │ Authorname                 │     │
│   │  ┌────┐  │ word            │     │
│   │  │ pic│  │ O ──            │     │
│   │  └────┘  │ O ──            │     │
│   │          │ submit          │     │
│   └────────────────────────────┘     │
└──────────────────────────────────────┘
```

IV) Fourth view:- Results (Answered)

```
┌──────────────────────────────┐        ┌──────────────┐
│  NAV Bar                     │        │ text (Op.1)  │
│   ┌──────────────────────┐   │        │ loading Bar  │
│   │ Author   Results     │   │───────▶│    50 %       │
│   │   ☰   ┌──────────┐   │   │        │ No of votes  │
│   │       └──────────┘   │   │        └──────────────┘
│   └──────────────────────┘   │
└──────────────────────────────┘
```

V) Fifth view:- answer ques/

```
┌──────────────────────┐              ↑
│ NAV                  │        ┌──────────┐
│  ┌────────┐          │        │          │
│  │  title │          │        └──────────┘
│  ┌────────┐          │
│  │  p..   │          │
│  └────────┘          │
└──────────────────────┘
```

VI) sixth view: Leader board.

```
┌──────┐    ┌───────────────────────────────┐
│ NAV  │    │  NAV                          │
│ ┌──┐ │    │   ┌────────────────────┐      │   │
│ └──┘ │    │   │ @│ name   score     │      │   │ Sum (Ans + un ans.)
│ ┌──┐ │    │   │  │ Ans ──           │      │   │
│ └──┘ │    │   │  │ unAns ──         │      │   ▼
│ ══   │    │   └────────────────────┘      │
└──────┘    └───────────────────────────────┘
      ┌┐           ┌────────┐
      └┘──         └────────┘
```

VII) seventh: New card

[3] Break Each View into a Hierarchy of Comps.

(I) Components of Login View :-

Login

Header

Form

[Submit]

App
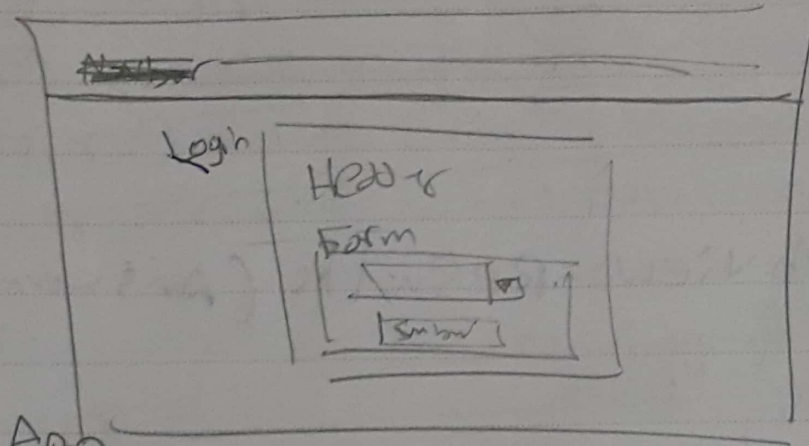
Reodt Comps. -
- Login: overall Comp Container
- Header: Some Title
- form:

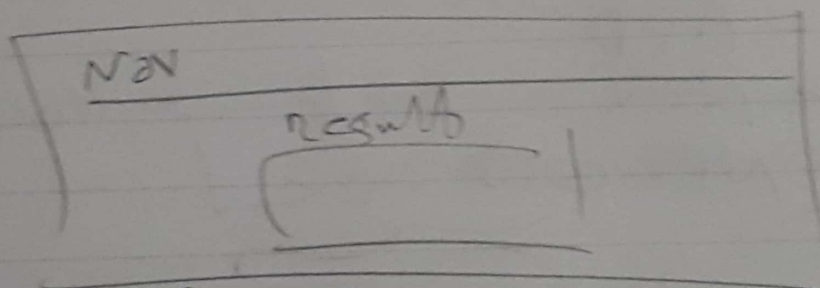(II) Componens of :- Home View
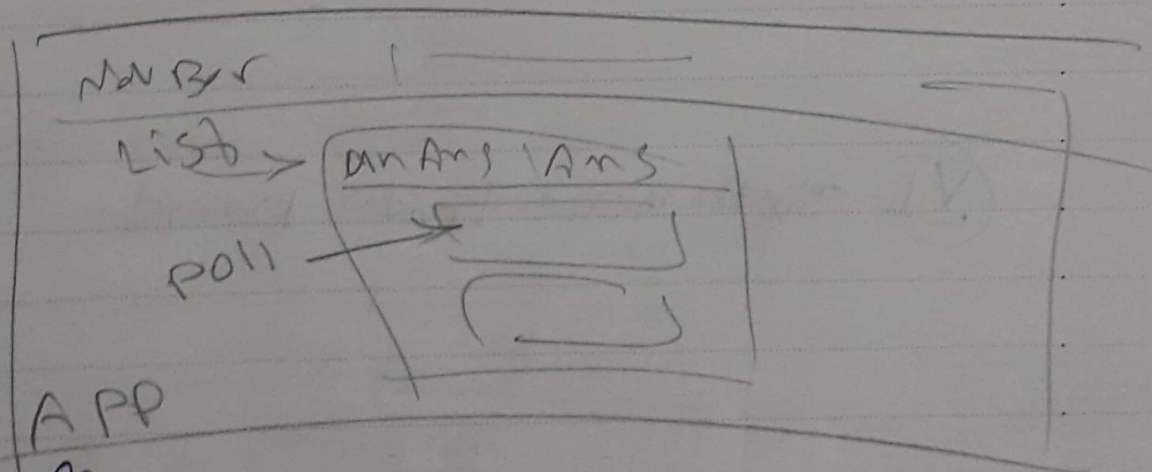
NavBar

List → UnAns \ Ans

Poll →

APP

Comp:-
- NavBar
- List of polls (Ans/unAns)
- Poll
- APP

(!!!) Comp of poll

```
┌─────────────────────────────┐
│  Nav                        │
│    ┌─────────────────┐      │
│    │ Poll            │      │
│    │ Author          │      │
│    │ ┌───┐ bodyLayout│      │
│    │ │avh│  o ────── │      │
│    │ └───┘  o ────── │      │
│    └─────────────────┘      │
│                             │
└─────────────────────────────┘
```

App

Comp of React:
  — Nav
  — poll (Card)
  — App

~~Comp~~ Results :

```
┌─────────────────────────────┐
│  Nav                        │
│        results              │
│       ┌──────────┐ │        │
│       └──────────┘ │        │
│                             │
└─────────────────────────────┘
```
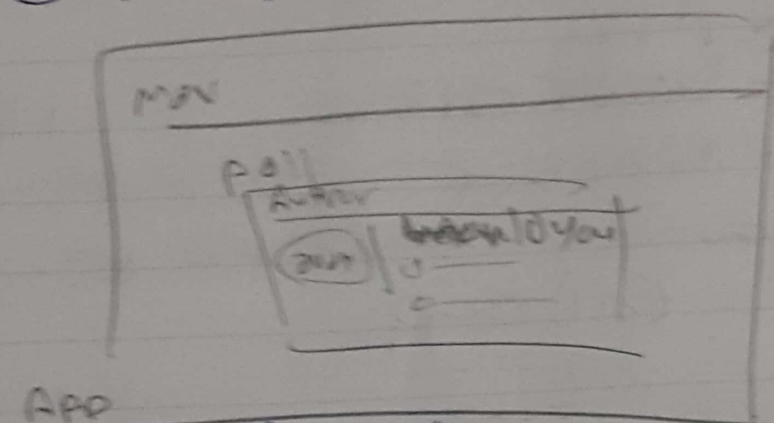
comp of Redct
  — App
  — NavBar
  — Results (card)

(✓) Comp of Ledderboard —

```
┌─────────────────────────────┐
│   Nav                       │
│    ┌────────┐      Ledercort │
│    └────────┘                │
│                             │
└─────────────────────────────┘
```
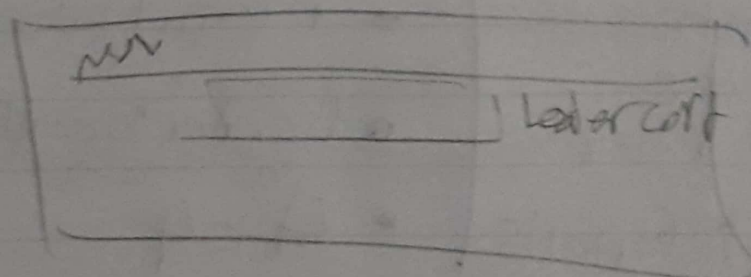
ReatComp:
  ∈ APP
  — Leader card
  → Navbar

Ⓥ NewCardCompsr

ReactCompr
  — APP
  — Nav
  — New card
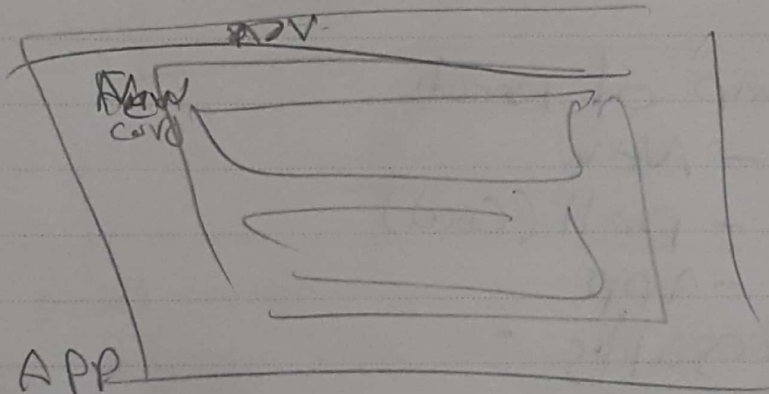
ⓋⅠⅠ Poll (404)

404

All components

- APP
- Login
- Home
- Nav Bar
- List of polls
  ↓ Ans   ↓ unAns

- poll
- Result
- Leaderboard
- 404

- New poll

SENA

[3] Determine what Events happen in the App

(I) APP :
- get users   & RECEIVE_USERS
- get Questions : RECEIVE_QUESTIONS

(II) Login:
- get Users % RECEIVE_USERS
- Set authuser & SET_AUTHED_USER

(III) Home - (List of Polls)
- unanswered Qs : get unanswered Questions
- Answered Qs: get Answered Questions

(IV) Poll view ⟶ get user of Users
               ⟶ get question of Questions

(V) poll question ⟶ get authedUser
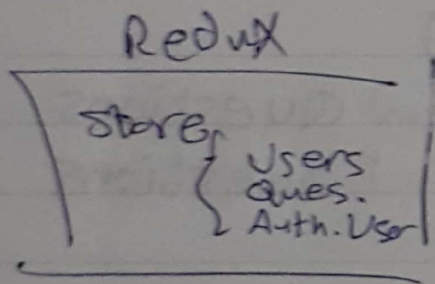   ⟶ get Questions
                     ⟶ Set option

(VI) Result :⟶ get question of Questions

(VII) New poll ⟶ get current Authed User
                     ⟶ Set text

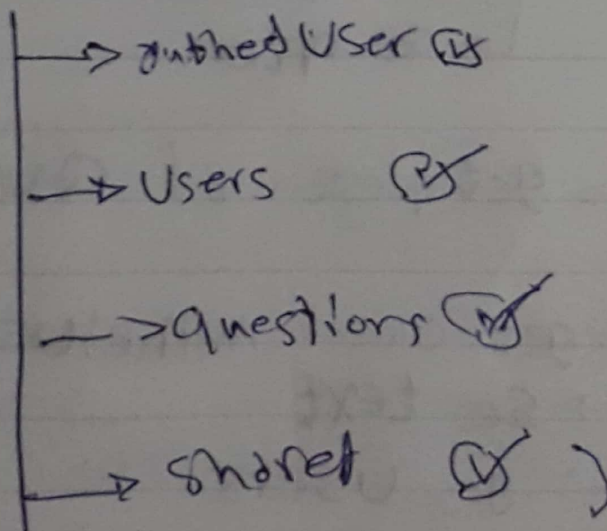(VIII) Leaderboard ⟶ get Users

[4] Determine wwt Data Lives in the store:-

I) Users & Actions , globally used → Redux store

II) Authed User → " " → Redux store

III) text of options → only used in New poll Gmp → UI state

IV) Options got? → only used in Result → UI Gmp.
(answered)

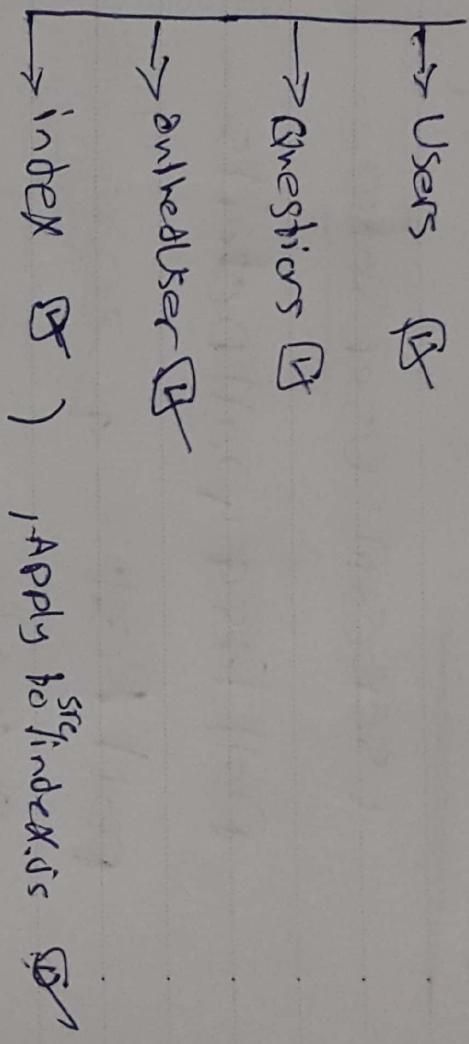Redux

```
     ┌─ Store ┌ Users
     │        { Ques.
     │        ┕ Auth. User
     └────────
```

⌒⌒⌒⌒⌒⌒⌒⌒⌒⌒⌒⌒

coding stages.
 (1) api.js :-   ☑

 ② Folder (actions : ☑
         ├─→ authed User ☑
         │
         ├─→ Users   ☑
         │
         ├─→ questions ☑
         │
         └─→ shared ☑)

③ folder (Reducer)s:
  ↳ Users ☑
  → Questions ☑
  → authedUser ☑
        ↳ index ☑)

                                    ,Apply to index.js ☑

④ folder (Middlewores:-
        ↳ logger ☑
        ↳ thunk ☑) , Apply to sec/index.js ☑

⑤ UI Compet.
      ↳ App.js ☑
      → Nav ☐
      → login