Software Engineer

* This is an archive of knowledges for a Software Engineer or a Software Developer.

# Test-Environment:

* Replit

# Key Terms

1. **ORM (Object–relational mapping):** Object-Relational Mapping (ORM) is a technique that lets you query and manipulate data from a database using an object-oriented paradigm.
2. **Schema (DB):**

# Stacks:

# Front-End:

## React

### Documentation

React-router-dom: <https://reactrouter.com/docs/en/v6>

### React-router

#### v.6

##### useNavigation

useNavigation is to used to navigate to a different route.

### Hooks

#### useState

The useState is a Hook (function) that allows you to **have state variables** in functional components.

#### useEffect

The Effect Hook lets you perform side effects in function components.

* Data fetching, setting up a subscription, and manually changing the DOM in React components are all examples of side effects.
* It runs both after the **first render** and after every **update**.

Example 1:

useEffect(() => {

authService.onAuthStateChanged((user) => {

if (user) {

setIsLoggedIn(true);

} else {

setIsLoggedIn(false);

}

setInit(true);

});

}, []);

#### useForm

With useForm, you can register all inputs using the register function.

**Other functions:**

* Data input validation: using HandleSubmit function.
  + HandleSubmit immediately returns another function onSubmit.
  + RegisterOption will show options that you can choose to validate input data.
  + You can write a message when you don’t fulfill the requirements in an argument of required key of Register function.

Notable Library

* + Recoil: A state management library for React
    - Flexible shared state
    - Derived data and queries
    - App-wide state observation

## JS

#### Ternary Operators

The conditional (ternary) operator is the only JavaScript operator that takes three operands.

* Reduces amount of code to create if…else statement

Ex: {isLoggedIn ? <button>Logout</button> : <button>Login</button>}

## NextJS

NodeJS and TypeScript ORM, a tranlator between the typescript code and the DB.

### Documentation

Official: <https://nextjs.org/docs>

## Flask

# Back-End:

## Django

Django is an open source back-end server-side web framework, written in Python.

## Prisma

### Documentation

<https://www.prisma.io/docs/getting-started>

### Prisma Studio: Database admin console

A default admin console.

Functions:

* View Schema.
* View Model
* Add record.

### Questions:

#### How does it work?

Given schema, Typescript generates a client written in JavaScript that communicates to DB to update/modify data.

## PlanetScale

PlanetScale is the MySQL-compatible serverless database platform.

* It is a database platform. They provide DB.
* It provides a server that we don’t have to maintain.
* MySQL-compatible: somewhat compatible.

### Facts

* They are using Vitess (Google-made scaling app created for Youtube.com).
* Vitess is an open-source technology. Vitess is a database clustering system for horizontal scaling of MySQL.
  + Horizontal Scale
  + High Availability
  + MySQL Compatibility
  + Online Schema Migration
  + Materialized Views
  + Kubernetes Native

### Useful Concepts

#### Connect to PlanetScale

#### Push to PlanetScale

#### Prisma Client

#### Prisma Studio

## Firebase

### Documentations

1. Firebase Official: <https://firebase.google.com/docs>
2. JavaScript Firebase SDK: <https://firebase.google.com/docs/reference/js>
3. Authentication: <https://firebase.google.com/docs/auth>

### Introduction

* Initially DB; acquired by Google; extended.
* Firebase is an umbrella of a back end as a service (BaaS)
* **Functions**
  + Cloud Firestore
  + Firebase ML
  + Cloud Functions – like AWS lambda
  + Cloud Storage – like AWS S3; file storage.
  + Hosting
  + Authentication – allows you to have the authentication ready in less than 10 minutes.
  + Realtime Database
  + **Analytics:**
    - Crashlytics
    - Performance Monitoring
    - Teset Lab: test websites in different devices
    - App Distribution
  + Grow your business
    - Data analysis related.
* Competitors: **AWS Amplify** – mostly same functions + GrpahQL API, and Rest API.

*When would you use Firebase?*

* You would use Firebase to get started with your idea because you can build what you intended very quickly.
* You can test your idea in a big scale because it has many components.
* If you are competing, you can finish faster than your competition, put it in the world.
* If people start liking you app, you can find your own platform to build a real app.

*How much is Firebase?*

* Firebase is not free 100% if you are uploading and updating etc.
* Free features:
  + A/B Testing
  + Analytics
  + App Distribution
  + App Indexing
  + Authentication
    - Verify with phone number 10k users for free.
    - Other verification is free.
  + Cloud Firestore 1GiB – They start charging $.18/GiB
  + Document write: 100,000 -> 18 cents.
  + Cloud functions
  + Hosting – 10GB.
  + Storage – 5GB free - 0.026/GB

### Project

#### nc-twitter-clone

*How do you start Firebase?*

1. Start a project
2. Once a project is created, create an app
   1. ADD Firebase SDK

*What is Firebase SDK?*

* There are configuration and HTML addition instructions.
* You can either follow the instructions shown at the page or look at JavaScript Firebase SDK documentation.

*How many ways to implement Firebase?*

# Programming language:

1. *CDN – put in the link.*
2. *Use firebase with npm.*

## Python

*Why learn Python?*

* Python has the biggest community of developers that can offer help.
* Python syntax looks beautiful. It looks like English (Human language).
* Python is a high-level programming language.
* You can use python to build
  + Web Application – Back-end and Front-end
  + Artificial intelligence
  + Data related apps
    - Data mining
    - Data visualization
    - Data scrapping
  + Hacking
  + Pen testing

### Documentations

1. Tkinter – Build desktop app UI - <https://docs.python.org/3/library/tkinter.html>