# Hur man React ”ar”

Detta är en skriftlig dokumentation om hur man, enligt den information och bilagor jag framtagit, hur en skapar och sätter upp ett React projekt. Detta visar på hur man kommer igång på rätt sätt för att projektet ska fungera på bäst tänkbara sätt samt köras lokalt på ett smidigt sätt. För att göra detta behöver man fler delar och bibliotek än bara React. Dessa behöver man konfigurera och installera på olika sätt och till varje del finns det ofta information och specifikationer som kan vara bra att ha lite kunskap om, men även veta vad de olika delar står för och hur dessa påverkar projektet.

I denna rapport kommer jag förklara de olika delarnas och deras relevans i projektet. Alltså vad de olika delarna gör. Detta blir en beskrivning på vad som man måste använda för att kunna köra och jobba med ett ”vanligt” React projekt.

## ATOM IO – [1]

För det första så behövs det ett verktyg, eller mer bestämt ett program att skriva all kod i. I detta projektet kommer formaten HTML, CSS, JS och JSX användas. Därför är en passande kodeditor ATOM IO för detta projekt eftersom den innehåller allt vi behöver och är användarvänlig. I Atom kan man smidigt bygga på önskade packet för att underlätta sitt arbete. Man kan enkelt skapa sina egna tangentbords-genvägar, där av indentering av kod och autofyllfunktioner, som gör kodskrivningen effektivare.

Atom IO är en gratis open-source kodredigerare för operativsystemen Windows, Linux och macOS. Atom har stöd för plug-in program skriva i Node.js. Liksom de flesta andra konfigurerbara textredigerare tillåter Atom användare att installera paket och teman från tredje part för att anpassa funktionerna och utseendet på redigeraren. Paket kan installeras, hanteras och publiceras via Atoms pakethanterare apm. De flesta av de paketen har gratis programvarulicenser, är ”community-built” och underhållna. Atom har även en inbyggd Git-kontroll utvecklat utav GitHub. [2]

Atom släpptes från beta, som version 1.0, den 25 juni 2015. Dess utvecklare kallar det för en "hackable text editor for the 21st Century". [1]

Ett smidigt sätt att komma igång med ditt skrivande är att i konsolen gå in i sitt projekt och sedan utföra kommandot ”atom .” för att placeras på rätt plats i gränssnittet.

## NODE JS – [3]

Node JS är gratis och open-source. Node är en slags JavaScript miljö som möjliggör att kunna skriva JavaScript-kod utanför en webbläsare. [3] Node låter utvecklare använda JavaScript för att skriva kommandoradsverktyg och för serverns sida kunna köra scripts för att dynamiskt producera webbsidans innehåll innan sidan skickas till användarens webbläsare. Node.js representerar ett paradigm "JavaScript everywhere", som menar på att det enar webbapplikationsutveckling runt ett enda programmeringsspråk, i stället för olika språk för serverns sida och klientsidan. [4]

Att köra Node JS tillsammans med React i detta projekt är nödvändigt för att koden ska vara exekverbar. Detta används främst i React då man behöver sätta upp en miljö att ta med sin JavaScript till. Med andra ord så skapar Node en lokal server till React där man kan hantera data via JavaScript.

### NPM – [5]

Node hanterar även alla olika packet som man behöver till sitt projekt. Npm är en pakethanterare för JavaScript med stor lista över återanvändbara kodmoduler och paket. I konsolen hanterar man detta genom kommandot npm. Npm används för att anropa Nodes funktioner. I projektet används detta främst för att smidigt hantera alla moduler och de packet som används i projektet men också för att starta upp server, den miljön, som JavaScriptkoden kommer köras i.

* Skriv npm -v i terminalen för att kolla vilken version. Det ska vara en version som är högre än 3

### Package JSON – [6]

Detta är en fil som skapas då man börjar installera packet till ditt projekt. Själva filen innehåller mycket metadata om projektet. Mestadels används den för två saker; att hantera alla packet, beroenden och moduler som används av koden, men även olika skript som hjälper till att hantera byggnader och exekveringar med mera- [6]

## **REACT**

React är ett JS-bibliotek som huvudsakligen hjälper oss att dela upp vår applikation i flera komponenter och är huvudämnet i denna rapport. React är ett stort bibliotek som löser hur projektets filer kommer struktureras, hur projektets kod vanligtvis ska skrivas och som styr hur allt kommer visas för användaren. Mer information kan hittas i tillhörande rapporter, ”Mastering Web Apps” [x] och ”React From the Ground” [x].

### React – [7]

Om man är familjär med MVC (Model View Controller-arkitekturen av applikations sätt att fungera) , så används React i detta projekt som själva View modulen. [8] React är inte en fullständig MVC-ram eller någon annan form av ramverk. React är bara ett bibliotek för att rendera dina olika ”Views”.

### ReactDOM – [8]

Detta paket fungerar som ingångspunkt för att få tillgång till DOM och rendera till servern. Alltså kunna skriva ut den kod som man skrivit på servern till det användaren ser. Den är avsedd att vara kopplad till det vanliga React-paketet nämnt ovan, som skickas som React via Node.

### React Bootstrap – [10]

Bootstrap som är ett välkänt responsivt frontend-ramverk, som hjälper dig att designa webbplatser snabbare och lättare via olika komponenter. Ramverket innehåller HTML och CSS-baserade mallar för typografi, knappar, tabeller, navigering, etc. React stöder Bootstrap och förenklar användningen av flera komponenter. React Bootstrap är speciellt utvecklat för att bygga en responsiv frontend med Bootstrap i React.

## Redux – [11]

React men anger inte tydligt hur man ska hålla reda på data (State) och hur man hanterar alla händelser (Actions) korrekt. Redux är en förutsägbar data (State) behållare för JavaScript-appar. [11] Det hjälper dig att skriva program som beter sig aktivt, kan köras i olika miljöer (klienten & servern) och är lätta att testa.

I grunden tillåter Redux att vi kan bygga React applikationer som du men vill. Men deligerar all data åtgärder (State) och händelser (Actions) till Redux. Med Redux tillåter man alla komponenter att kommunicera och smidigt hitta det relevanta objektet.

## Firebase – [13]

Firebase är en realtidsdatabas. Firebase tillhandahåller en realtidsdatabas och backend som en tjänst. Tjänsten ger applikationsutvecklare ett API som gör att applikationsdata kan synkroniseras över klienter och lagras i Firebase's moln. [14]

Firebase Realtime-databasen är en molnhärdad databas. Data lagras som JSON och synkroniseras i realtid till varje ansluten klient. Firebase-databasen är en så kallad NO-SQL databas och är bara ett stort JSON-objekt där du kan lagra allt du vill ha inuti. [14]

I detta projektet används Firebase databas för registrering och autentisering, och hjälper till att kryptera lösenord och validera registrering på ett smidigt och tacksamt sätt.

## GitHub – [15]

GitHub Inc. är en webbaserad webbtjänst för versionskontroll med Git. Den används mestadels för datorkod. Den erbjuder alla funktioner för Git-versionen för distribuerad versionskontroll och källkodshantering (SCM), samt att lägga till egna funktioner. Det ger åtkomstkontroll och flera samarbetsfunktioner som buggspårning, funktionsförfrågningar, uppgiftshantering och wikis för varje projekt. [16]

Den här Självständiga Fördjupningen ligger på GitHub: <https://github.com/aregato/reactOnReact>

# Källförteckning

[1] – Atom IO - https://atom.io/

[2] - Atom Flight Manual - <https://flight-manual.atom.io/#the-native-web>

[3] - Node JS - <https://nodejs.org>

[4] – Node JS Foundation - <https://foundation.nodejs.org/>

[5] - A Beginner’s Guide to npm — the Node Package Manager, Michael Wanyoike & Peter Dierx, 2017 - <https://www.sitepoint.com/beginners-guide-node-package-manager/>

[6] - Specifics of npm's package.json handling, 2018 - https://docs.npmjs.com/files/package.json

[7] – React - <https://reactjs.org/>

[8] - React: Making faster, smoother UIs for data-driven Web apps, Paul Krill, 2014 -https://www.infoworld.com/article/2608181/javascript/react--making-faster--smoother-uis-for-data-driven-web-apps.html

[9] – React-DOM - <https://www.npmjs.com/package/react-dom>

[10] – React Bootstrap - <https://react-bootstrap.github.io/getting-started/introduction>

[11] – Redux - <https://redux.js.org/>

[12] - Understanding Redux + React in Easiest Way, Tatun Sharma, 2018 - <https://medium.com/@tkssharma/understanding-redux-react-in-easiest-way-part-1-81f3209fc0e5>

[13] – Firebase - <https://firebase.google.com>

[14] - Firebase Docs - <https://firebase.google.com/docs/database/>

[15] – GitHub - <https://github.com/>

[16] – GitHub Pours Energies into Enterprise, 2013 - https://techcrunch.com/2012/07/09/github-pours-energies-into-enterprise-raises-100-million-from-power-vc-andreesen-horowitz/?guccounter=1

# Tillhörande material

[t1] – Mastering Web Apps

[t2] – React From the Ground