# React-on-React

Rapport skriven av

Hannes Linnér

## Introduktion och förklaring av problemområde

Denna rapport grundar sig kring utvecklingen moderna webbapplikationer. I dagens läge är internet något som hela tiden utvecklas och fortsätter att växa och förbättras. Plattformar och webbapplikationer blir snabbare och snabbare samt innehåller allt mer funktionalitet. I takt med att detta är möjligt så skapas också nya tekniker för att utveckla dessa plattformar eller applikationer som finns ute på internet. För att arbetet med att utveckla moderna webbapplikationer ska vara möjligt och bli smidigare skapas olika verktyg som utvecklare använder sig utav. Förutom alla olika språk som man kan programmera i, kan man också använda sig av verktyg inom språken. Dessa verktyg kallas bibliotek eller ramverk och innehåller för det mesta redan skriva funktioner. Dessa biliotek och ramverk kan användas med olika fördelar till olika saker. I detta självständiga fördjupningsarbete kommer jag att titta närmare på biblioteket React och dess betydelse för utvecklandet av en modern webbapplikation. React är utvecklat för språket JavaScript vilket idag har blivit mycket användbart för utveckling av webbsidor och kan användas för att både skapa en backend och en frontend av applikationen. Alltså kan man med JavaScript bygga en fullskalig webbapplikation. [1]

React är skapat och underhålls utav Facebook och deras utvecklare [2]. Reacts är ett bibliotek och har blivit mycket populärt i webbutveckling. Med stor sannolikhet beror på att det funkar så bra hos Facebook. Men även stora företag som Netflix, Airbnb och Walmart stödjs utav React. [3] På arbetsmarknaden kan man se hur olika verktyg driver kända plattformar som Facebook och Netflix och blir snabbt populära hos företagen och arbetsmarknaden eftersom det visar att verktyget funkar bra.

React är ett JavaScript bibliotek som huvudsakligen hjälper oss att dela upp vår applikation i flera komponenter. [1] React styr det som syns på en webbsida och hanterar data som ska förändras utan att sidan ska behöva uppdateras. React gör det möjligt att dynamisk interagera med användaren på ett modernt sätt. [2] Eftersom React skrivs i JavaScript kan sättet att skriva själva koden kännas igen. I denna fördjupning kommer därför innehålla utvecklandet av ett antal mindre applikationer som sedan leder till en webbapplikation. Webbapplikationen kommer vara skriven i React och analyseras för att kunna bedöma fördelar.

Denna självständiga fördjupning handlar om att lära sig arbeta med verktyget React som spets i frontend-utvecklingen. Projektets mål är att ta fram relevant information om biblioteket och dess påverkan i en modern webbapplikation. Projektet kommer vara en webbapplikation och kommer vara grund för att ta reda på varför React har blivit så använt bland företagen på arbetsmarknaden och varför många använder React för sina webbsidor.

## Definition av problem

Hur utvecklas en modern webbapplikation? En modern webbapplikation kan i dagens läge göra mycket för dess användare. Den jobbar ofta dynamisk med hur användare interagerar med systemet och ändrar på informationen som visas. De moderna applikationerna byggs i alla dess storlekar och innehåller allt mer funktionalitet. Kollar man på arbetsmarknaden kan man urskilja hur vissa tekniker för att utveckla webbapplikationer är mer populära och använda än andra. I dagens läge finns det flera sätt att göra detta på och därför med olika fördelar. React är en av dessa tekniker eller egentligen kallat ett bibliotek [1]. React används tillsammans med JavaScript och har blivit väldigt populärt bland företagen som bygger webbapplikationer. Denna fördjupning kommer undersöka varför det har blivit så populär och vilken betydelse det har för en modern webbapplikation.

Hur gör man sin webbutveckling smidigare med ett högre resultat? Vanlig HTML som används för att skapa webbsidor börjar kännas föråldrad. Det är sällan någon som bara använder sig av endast HTML, CSS och JavaScript utan man har ofta fler verktyg för att förbättra systemet. Vanlig HTML är inte återanvändningar på samma sätt som t.ex. klasser i objekt orienterad programmering utan måste skrivas på varje HTML-sida. Resulterar ofta i att man får skriva flera html-filer som vanligtvis har en liknande struktur. Detta gör att utveckling och ändringar tar ofta lång tid i stora projekt. React är ett hopp om att möjliggöra detta. Med React skapar man förutsättningar att skapa komponenter eller templates som man sedan kan återanvända i sin kod på ett sätt som påminner om att skriva objektorienterat. Vilket är ett relativt nytt sätt att jobba med webbutveckling.

Ett problem att tillägga kan vara att React har en hög inlärningskurva vilket betyder att det ofta tar ett tag att lära sig förstå hur man ska utveckla med det. Tills man har kommit på hur det funkar, då kommer det gå effektivare än tidigare.

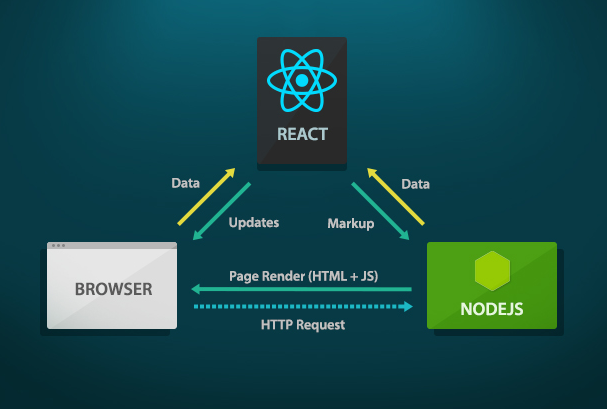
## Översikt över relaterat arbete

Stora delar av inhämtning av fakta och information kommer ske genom både videolektioner och uppgifter på utbildningsplattformen Udemy [4]. I två olika kurser kommer React gås igenom på olika sätt för att täcka den kunskap som behövs för att börja sätta sig in i React. Den ena kursen går igenom React från grunden. Vad det innehåller, hur man arbetar med det och vad man kan göra. Den andra kursen visar mer på hur man utvecklar med det. Hur man bygger upp sitt projekt och hur man ska strukturera för att det ska fungera på bästa sätt. För att få en bra bild av hur ett bibliotek fungerar är det bra att själv testa sig fram.

Artiklar har undersökts för att hitta relaterade arbeten och få en översikt över ämnet som valts. Här har tre relevanta artiklar valts ut för att skapa se liknande arbeten. I artikeln ”Modern Web-Development using ReactJS” [1], finns fakta om Reacts delar och vad det har för påverkan i en moden webbapplikation. Artikeln ”Facebook's Functional Turn on Writing JavaScript” [5], berättar hur Facebook som utvecklat React har tänkt med biblioteket. I den sista artikeln ” Using React JS for Web Applications” [6] erbjuder steg för steg information om hur man använder React JS.

## Beskrivning av möjliga lösningar

Det finns i nuläget många olika sätt att jobba och massor med verktyg att använda föra att utveckla webbapplikationer. Förutom de olika programmeringsspråken som finns kan man även använda ramverk och bibliotek för att underlätta mycket i sin utveckling och använda sig av redan skapad kod. Som till exempel React som är ett bibliotek med funktioner som ska framställa allt som har med gränssnittet, av webbapplikationen, att göra.

React är som tidigare nämnt ett verktyg för att framställa allt som har med gränssnittet till webbapplikationen att göra. Ett liknande verktyg till React är Angular JS. Angular är som React JavaScript verktyg och har liknande syfte. Både dessa räknas som själva vyn i en MVC arkitektur när man jobbar med webbapplikationer. Dessa verktyg har fokus på att öka användarupplevelsen ytterligare med hjälp av prestandaförbättringar i form av snabbare renderings- och responstider. React, har introducerat ett helt nytt arkitekturmönster för att både hantera applikationens tillstånd och dataflöde. React erbjuder även stöd för native applikationsutveckling och möjliggör att på ett enkelt sätt rendera från server-sidan. Något som är svårt att åstadkomma med en applikation utvecklad med Angular [7]. I artikeln ” React vs Angular : Slaget om användarupplevelsen” [7] utforskar man fördelar och nackdelar närmare med de båda verktygen.

Reacts arkitektur i en fullskalig applikation

## Diskussion om befintliga lösningar med fördelar och nackdelar

Att använda dessa verktyg kan vara svårt till en början. Med nya sätt att skriva kod och regler för att funktionerna ska fungera rätt tar det tid innan man förstår hur man ska göra. Men att använda redan skrivna funktioner via biblioteket är något som man i slutändan kommer att spara tid på. Nackdelen är att det tar lite tid att komma igång med. React har även en hög inlärningskurva.

I nu läget, utan att använda dessa verktyg behöver webbsidan uppdateras för att information på sidan ska förändras. Detta är något som inte längre känns särskilt modernt och är inte heller alltid särskilt användarvänligt. När man lär sig dessa verktyg kommer man kunna få sidan att bete sig mer dynamisk och lättare kunna interagera med användaren. Med dessa bibliotek kan man också lättare skriva återanvändningsbar kod.

## Beskrivning av den slutgiltiga lösningen som valts

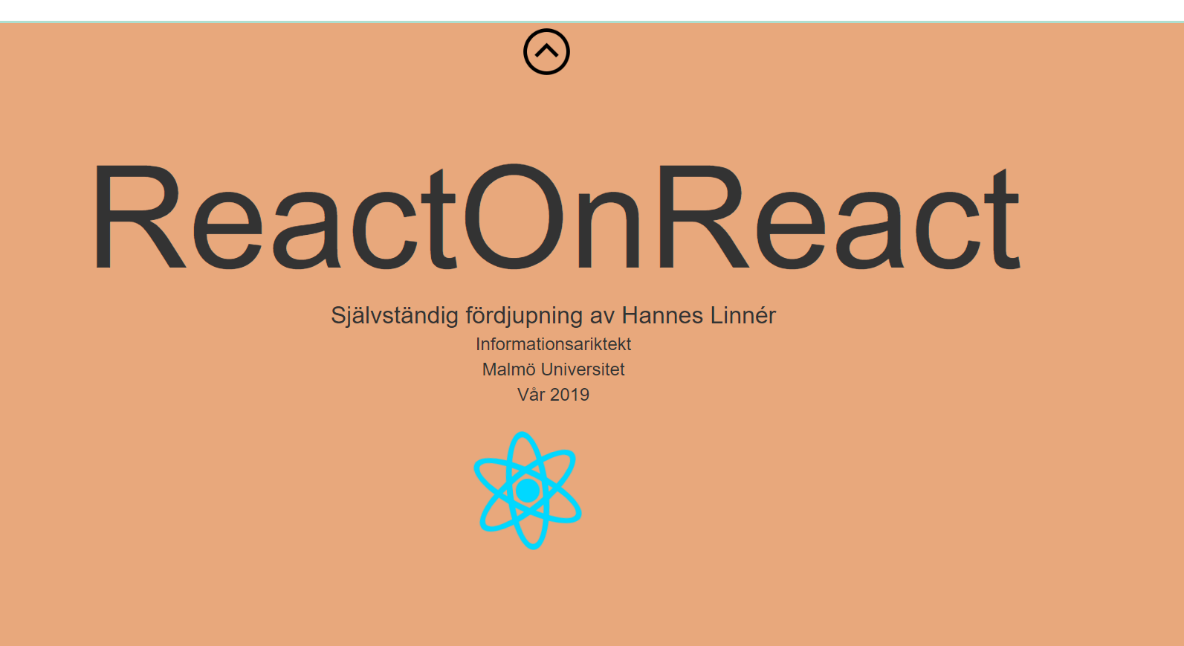
Jag valde att bygga en webbapplikation som mitt slutprojekt i kursen Självständigt fördjupning. Webbapplikationen sammanfattar det jag gått igenom under mitt projekt och innehåller de delar som jag tycker är viktiga. Applikationen innehåller en registrering samt en logga in funktion. Registrering krävs för att kunna logga in och man måste vara inloggad för att kunna se resten av systemet. Vid inloggad stadie kan man med hjälp av ett gränssnitt lägga till uppgifter till listor. Dessa kan också slutföras och tas bort. Uppgifternas användare är synlig. Utan att vara inloggad kan man se projektets presentation och även alla delarna utav detta projekt

### Appens arkitektur

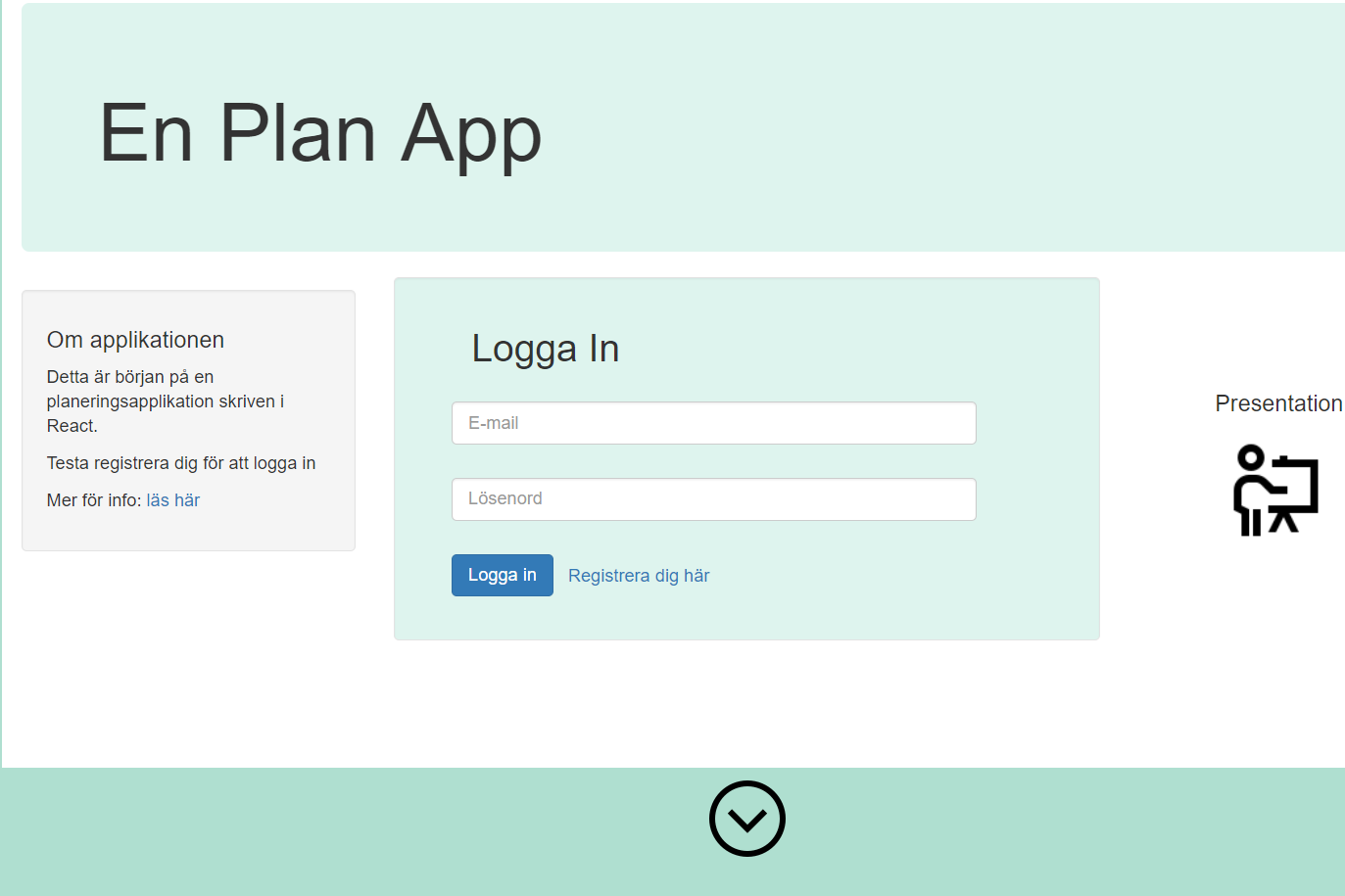
För att kunna köra React så krävs det tillgång till Node eller liknande för att kunna skapa en server. Node är en slags JavaScript miljö som möjliggör att kunna skriva JavaScript-kod utanför en webbläsare. React använder sig i detta projekt biblioteket Redux som hjälper dig att skriva program som beter sig aktivt, kan köras i olika miljöer (klienten & servern) och är lätta att testa. Projektet använder sig även av en databas som är bland annat utvecklad för att stöda React. För webbapplikationens design har CSS verktyget Bootstrap används. För React finns det ett special utvecklat Bootstrap där man enklare i sin Bootstrap kod kunna använda sig utav dess styling.

* Visa på hur projektet har skapats efter förutsättningar kring Reacts bibliotek.
  + Skriva om de olika ”tillägg” 🡪De olika paketen, förutom React + ReactDOM, som används för att kunna enkelt skapa rätt struktur för React.

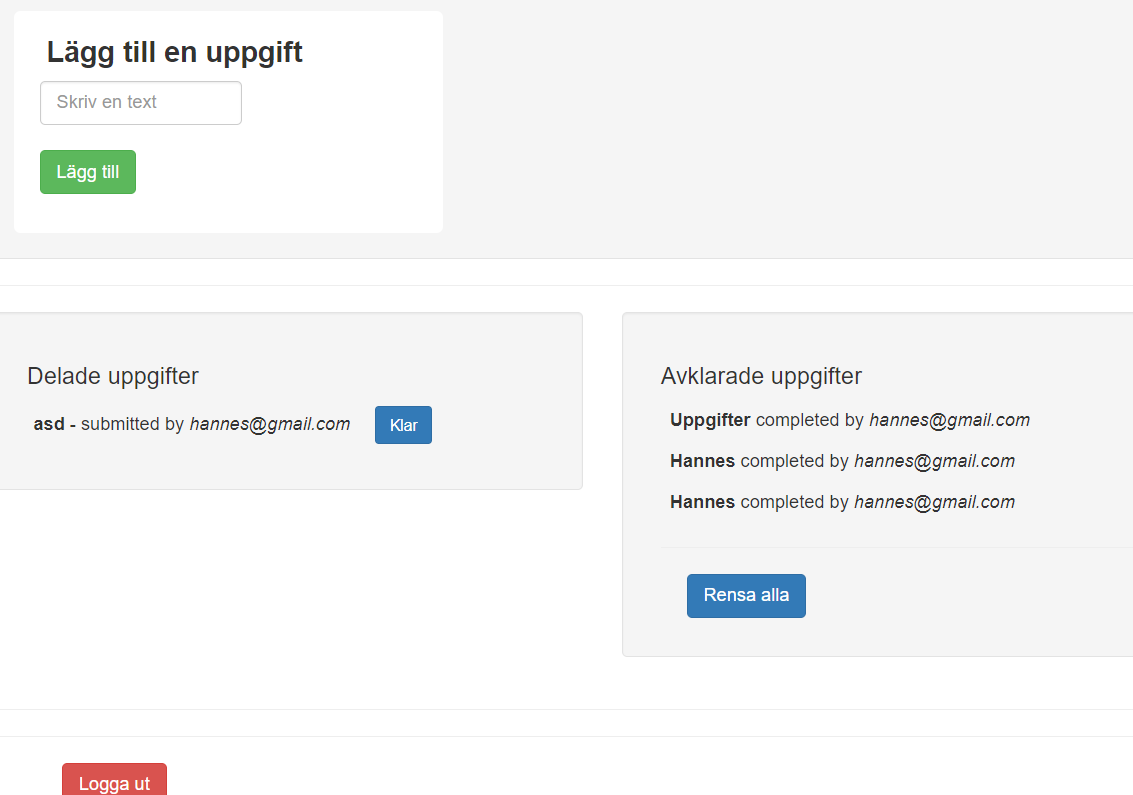
## Illustraion utav resultat

 Bilderna sammanfattar slutprojektets tre olika delar. Webbapplikationen som valdes att göras som slutprojektet innehåller en presentation över det självständiga fördjupningsarbetet. Där de olika delarna överskådligt gås igenom. Webbapplikationen innehåller en registrering där man kan registrera sig på sidan för att kunna logga in på systemet. Den tredje delen och bilden är när man har loggat in på systemet. Då kan man lägga till uppgifter som alla användare kan se. Man kan också flytta runt dessa uppgifter mellan olika listor samt rensa dem. De två sista bilderna visar på strukturen av filerna i min självständiga fördjupning. Bilden till vänster visar webbapplikationens innehåll och struktur. Den sista bilden visar projekt mappen med allt som projektet har innehållit. Denna finns att observera på GitHub.

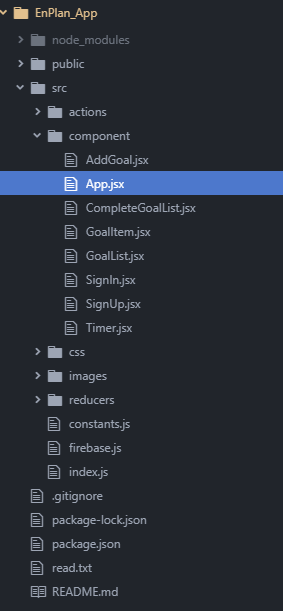
Presentation av projekt i webbapplikationen



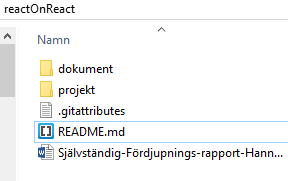
Inloggningssidan av webbapplikationen 1



Planeringsdelen av webbapplikationen



Webbapplikationens mappstruktur



Självständiga fördjupningens innehåll

## Egna reflektioner

Jag tror att React har blivit så populärt eftersom hela sättet att jobba känns modernt. React hjälper en att effektivisera sitt arbete och ger en ett professionellt resultat. I dagen läge är det viktigt att ha en webbsida som flyter på bra och reparerar på interaktion från användaren. Detta löser React på ett inte allt för komplicerat sätt.

React kommer göra många saker lättare för mig i framtiden. De stora projekten kommer bli mer strukturerade och det kommer därför vara lättare att hitta saker i dem. Detta leder också till att det kommer bli lättare att testa olika delar av projektet samt kunna ändra på ställen utan att behöva göra det överallt. Koden man skriver i React gör att koden känns mindre redundant.

## Projektplannering

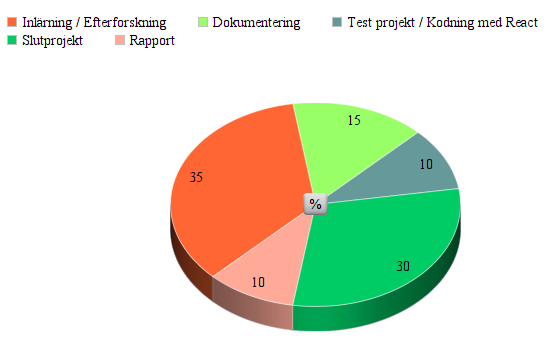
På hög abstraktionsnivå kan man säga att detta projekt innehöll fem olika delar. Till en början behövde jag göra efterforskning för att veta hur jag skulle gå tillväga och för att förstå React.

Till min efterforskning skrev jag olika typer utav Dokumentering för att samla det jag lärt mig och upptäckt. Under tider gjorde jag även olika Testprojekt för att se hur man använder React rent praktiskt och på vilka olika sätt man kan göra detta.

Efter det kände jag mig redo för att börja bygga (Slutprojekt) en planerad webbapplikation för att visa det jag lärt mig med React.

Till sist skrevs en Rapport över arbetet.

Projektet ligger uppe på GIT om man vi se det fullständiga projektet med alla delar inkluderade.

<https://github.com/aregato/reactOnReact>

Tidsestimering i procent

## Exempel på hur man behöver mer kunskap för att utveckla dem

Webbapplikationen som byggdes som slutprojekt av denna självständiga fördjupning är bara en början av på något som tillslut skulle kunna bli en stor mång funktionell webbapplikation.

* ***????***
* Beskriv hur man kan vidare utveckla den appliktion som jag håller på med. Bygga på delar och skapa nya komponenter.

## En relevant och välfungerande reflektion om egeninlärningsprocessen

* Något om hur det finns stora mängder av information att hitta kring React eftersom det har blivit så poplärt.
  + Men också hur olika är bättre än andra.
  + Hur React är relativt nytt och hela tiden utvecklas för at bli bättre
    - Därför ändras biblioteket och nya funktioner samt nya sätt att skriva på uppkommer.
      * Detta bidrar till att information kring React kan vara föråldrad.
      * Även olika packet som används i olika tutorials kan vara antigen uppdaterade eller föråldrade, och inte går att använa på samma sätt som man gjorde från början.
* What did you learn in this course?

## En relevant och välfungerade reflektion om erfarenheterna under denna kurs

* Ett bra sätt att förbereda sig inför att skriva C-uppsats.
  + Ansvar, struktur & logistik.
* Aktuellt att kunna lära sig på egenhand att förstå nya utvecklingsmetoder.
  + Detta är ett yrke där saker utvecklas hela tiden. Nya metoder och smartare sätt att göra saker är alltid aktuellt.
* How did it work out for you to read a course in this way?
* What could be
* improve

## Källförteckning

[1] - Modern Web-Development using ReactJS. - Aggarwal, Sanchit - International Journal of Recent Research Aspects. Mar2018, Vol. 5 Issue 1, p133-137. 5p.

[2] – React JS - <https://reactjs.org/>

[3] - 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

[4] – Udemy - https://www.udemy.com/

[5] - React: Facebook's Functional Turn on Writing JavaScript - Hunt, Pete - O'Shannessy, Paul - Smith, Dave - Coatta, Terry - COMMUNICATIONS OF THE ACM; DEC 2016, 59 12, p56-p62, 7p. - ASSOC COMPUTING MACHINERY

[6] - Using React JS for Web Applications. - Bhatia, Ashish Singh - Open Source for You; May2016, Vol. 4 Issue 8, p56-58, 3p – 2016

[7] - React vs Angular : Slaget om användarupplevelsen, Alkrot, Magnus, - Mittuniversitetet, Avdelningen för informations- och kommunikationssystem, 2016.

## Tillhörande

[x] – Mastering Web Apps

[x] – StartUp

[x] – SetUp

[x] - Abstract