**SPA med SPA**

Detta var en kul uppgift med en del utmaningar, jag ägnade första dagen med att gå igenom gammalt material och brainstorma utseende på applikation samt vad som skulle finnas på sidan. Vilka funktioner behöver man, och framförallt hur ska man göra när man har två olika behandlingar med olika tider som kan bokas samma dag!?  
  
Jag började med att få till en tilltalande design (ja jag började i fel ände) men för mig var det viktigt att få en bild av hur en ”riktig” applikation skulle kunna se ut och kännas medans man jobbade och mitt fall inspirera. Blev imponerad av hur mycket man fick med sig när man importerade paketet React-calendar och det är det enda paketet som jag importerat.

Jag valde utifrån vilket datum som var valt i kalendern så skulle detta styra vilket datum kunden ville boka. Då får kunden en överblick över hela månaden och vilka veckodagar som var på specifika datum. Blev riktigt bra efter lite if satser beroende på ’states’ och en mapping som kollar vilka tider som var tillgängliga på vald datum och behandling.

Gick fort att få till en routing och bra boilerplate tack vare gammalt material från livekoding och man kunde lätt anpassa efter de nya behoven. Att jobba med React har varit kul efter lite invänjning då det känns som att inlärningskurvan är relativt stor, tar tid att vänja sig med useEffect() och ’states’. Men med Typescript så får man en hel del hjälp som man inte fått i vanilla javascript, vilket planat ut kurvan något och gjort det smidigt att jobba.

Eller som Matt Pocock sa i dokumentären Typescript Orgins "I could probably turn up to work drunk, and still code pretty good.". Du får feedback och intellisense vilket underlättar felsökningen och hjälper dig skriva korrekt från början.

Typescript har gjort att man jobbat fortare och med bättre, säkrare kod. React är lite annorlunda när du jobbar med ’states’ och setters till dessa, är applikationen i ett visst ’state’ så ska detta visas lite som villkorlig rendering. Man fick jobba komponentbaserat vilket underlättar när man ska få över funktioner och data till andra delar av applikationen, vilket man känner igen ifrån java.

Jag är väldigt nöjd över min lilla applikation, den känns smidig och tilltalande i design har en del feedback till användaren ifall bokningar inte går igenom. Sedan att det använder virtuell DOM gör att allt flyter på fint och upplevs som ett lokalt program på datorn utan fördröjning eller onödig rendering. Riktigt läckert!