Teknisk dokumentation

schema.service.ts

Denna klass hanterar soap anropen, läser svaret och skapar sedan schema post-objekt för varje schema post som kommer från servern. Denna klass bör endast anropas av getScheduleDataService

getSoapData(inputDate: Date): Promise<ScheduleEntry[]>

Denna metod skickar ett soap-envelope till **kronox.hig.se**s soapserver. Metoden delar också upp datumet till en string som passar i envelopet. Anropet till servern sker sedan med ett axios post-anrop.
Svaret som kommer sparas som ett XML-objekt och vilket gör det möjligt att sortera bort de extra delar som man får med.

readResurser(resurser: any)

Eftersom kursen och lokaler ligger ytterligare ett steg ner i hierarkin så används denna metod för att plocka ut dessa. Metoden skickar sedan tillbaka endast kursen och lokalen. Om det finns färre än 2 resurser så kommer endast en 0a att sparas i listan för att markera att något inte stämde.

class ScheduleEntry

Denna klass är de som kommer att representera en schema post. Innehåller endast en metod som beräknar hur många timmar som schema posten är.

get-schedule-data.service.ts

Denna klass används för att hämta alla schema poster. Eftersom servern inte tillåter att hämta fler än 300 schema poster så delas anropen till servern upp vilket hanteras av denna klass.

getSoapDataIterated()

Hämtar data från schemaService en dag i taget lika många gånger som den satta tidsperioden, det vill säga 7 eller 30.

fillArrayByTimePeriod(startDate: Date, numberOfDays: number)

Metoden tömmer först arrayen med schema poster och sparar startdatumet samt tidsintervallet och anropar sedan getSoapDataItterated() och väntar på att denna ska bli klar.

quanData/quan-data/quan-data-update.service.ts

Detta är en metod som används för att uppdatera det kvantitativa datat som visas längst ner innan footern. Servicen innehåller två stycken variabler som är observable som sedan booked, cost och unbooked prenumererar på. När en förändring sker uppdateras då även siffrorna som användaren ser på hemsidan.

Händelseförlopp

Inget data kommer att hämtas innan man har valt datum och valt tidsintervall.
När ett tidsintervall, 7 eller 30, är valt körs metoden `onSelect()` i `timeFilters.component.ts`. Denna metod gör ett anrop till `fillArrayByTimePeriod` i `getScheduleData.service.ts`.
Därefter mappas datat via grafernas typescript filer med hjälp av metoden `mapRooms` i klassen `roomMapService`.

Grafer

Graferna är skapade med hjälp utav Google charts för dokumentation kring dessa se länkar nedan:
Bubble graph: https://developers.google.com/chart/interactive/docs/gallery/bubblechart
Timelines: https://developers.google.com/chart/interactive/docs/gallery/timeline

Tips: ID måste ligga först och måste även vara i ett string-format. Var även noga med hur många variabler som anges och i vilken ordning.

Dependencies

Angular version 15.1.1
Axios version 1.2.3
Bootstrap version 5.2.3
Csv-parse version 5.3.3
Fast-xml-parser version 4.0.13
Ngx-print version 1.3.1

Förbättringsområden

- Hämtningarna från KronoX-servern kan effektiviseras genom att skapa en databas för att lagra alla schema-poster. Denna databas kan sedan uppdateras med jämna intervaller för att få det senaste ändringarna
- Grafernas storlek och presentation kan förbättras för en trevligare användarupplevelse användningskostnads-grafen använder just nu inte hela utrymmet.