Mini Sistem IOT (Home Appliances)

Kerkesat e projektit:

Detyra juaj eshte te ndertoni nje mini-sistem IoT. Fillimisht sistemi te ofron mundesine per ta bere set-up duke krijuar te gjitha pajisjet elektroshtepiake duke specifikuar:-Tipine pajisjes-Gjendjen (on/off)-vendin ku ndodhet(dhoma1, dhoma2)-atribute te tjera duke specifikuar limitet per vlerat min dhe max qe ato mund te marrin.Sistemi do te mare vlera per ndryshimin e vlerave te parametrave te pajisjeve dhe do i updatoje real-timene UI.Sistemi duhet te shfaqe alerte per vlerat kritike.Teknologjite e kerkuara: FrontEnd: Angular BackEnd: Spring Boot Database: MongoDB

Duke qene se detyra kerkon qe veprimet te kryhen ne real-time dhe ndermjet serverit dhe klientit, jane ndertuar dy pjese: back end ne SpringBoot dhe front end ne Angular dhe jane perdorur WebSocket.

# Back End

Struktura e projektit perbehet nga keto paketa:

* Model 🡪 Permban klasat model te Appliance, Attribute, Location dhe ApplianceDTO, ku secila permban atributet perkatese. Relacionet Appliance-Attribute jane one to many, pra nje appliance mund te kete disa attribute.
* Repository 🡪 Permban nderfaqen ApplianceRepository te cilen do e therrasim ne rastet e veprimeve CRUD ne databaze.
* Service 🡪 Permban klasen ApplianceService ne te cilen ndodhen te ghitha metodat per veprimet add, delete, update state, update value dhe read per appliances.
* Controller 🡪 Permban klasen ApplianceController ne te cilen jane bere annotations per te lidhur websocket me metodat e CRUD ne database. PErdoren Annotations @MessageMApping me pathin e duhur dhe @SendTo me destinacionin e percaktuar te websocket.
* Configuration 🡪 Permban klasen WebSocketConfig me te gjitha konfiguriemt per webSocket. Percaktohet endpoint, destinationPrefix dhe brokeri

# Front End

Per te ndertuar pjesen e front end eshte perdorur nje template nga Creative Tim, i cili eshte modifikuar sipas nevojave te projektit aktual. Projekti ne Angular eshte i perbere nga keto foldera:

* components - Ky folder eshte i ndare ne 3 subfolders (footer, sidebar dhe navbar) te cilet permbajne kodin per krijimin e 3 strukturave themelore te faqes.
* dashboard - Ky folder perfaqeson faqen kryesore, ne te cilen shfaqen te gjitha pajisjet, me te dhenat perkatese. Folderi permban keto file :

**dashboard.component.html** - Nderton faqen duke perdorur reactive forms, per te lejuar berjen e ndryshimeve nga perdoruesi ne real-time.

**dashboard.component.ts** - Permban funksionet qe bejne te mundur lidhjen me websocket dhe dergimin e te dhenave ne websocket, rritjen/uljen e vleres aktuale te nje atributi, shfaqjen e alarmit ne rast se kjo vlere eshte mbi leren maksimale, si dhe fshirjen e nje pajisjeje nga lista. Gjithashtu ne kete file behen disa stilizime te dashboard nepermjet typescript.

**dashboard.component.css** - Ben stilizimin e tabeles qe shfaq atributet e nje pajisjeje.

* icons - Ky folder permban disa ikona qe jane perdorur ne elemente te ndryshme te faqes.
* notifications - Ky folder nderton faqen ne te cilen perdoruesi mund te shtoje nje pajisje te re duke futur te dhenat perkatese ne fushat e dhena. Folderi permban:

**notifications.component.html** - Nderton faqen ne te cilen perdoruesi jep te dhenat per pajisjen e re, dhe nepermjet reactive forms i lejohet te shtoje te dhenat per me shume se 1 atribut nese eshte e nevojshme.

**notifications.component.ts** - Permban funksionet qe bejne te mundur shtimin e atributeve te reja, dhe dergimin e te dhenave ne websocket, ne menyre tetille qe pajisja e re te shtohet real-time ne faqen kryesore.

* table-list -Ky folder permban klasat qe bejne te mundur ndertimin dhe funksionimin e faqes ne te cilen shfaqet nje tabele me cdo pajisje te shtuar dhe te dhenat perkatese. Permban keto klasa:

**table-list.component.html** - Nderton faqen dhe tabelen qe shfaq ID, Type, Location, Attributes, State aktual dhe nje opsion per te nderruar manualisht statusin e pajisjes.

**table-list.component.ts** - Permban funksionet qe bejne te mundur shtimin real-time te pajisjeve te reja dhe ndryshimin e statusit te pajisjeve.

* websocket - Ky folder permban **WebSocketAPI.ts**, qe permban te gjithe funksionet qe bejne te mundur lidhjen me websocket, shkeputjen, shkembimin e te dhenave, si dhe afishimin e erroreve ne lidhje me funksionimin e Websocket.