



.NET ARENDAJA PROOVITÖÖ 2021

KAS SUL ON OLNUD OLUKORDA, KUI SUL ON VAJA MÕNDA HDMI JUHET VÕI ÜLEMINEKUT, AGA SA EI OLE KINDEL, KAS SUL ON SEE OLEMAS?

OLED KINDEL, MIS ASJAD SUL KELDRIS, SAHVRIS, GARAAŽIS VÕI LADUDES ON?

Loome rakenduse, mis annab ülevaate antud olukorrast ehk saad luua enda jaoks lao- või inventuurisüsteemi.

LOOGE VEEBIRAKENDUS JÄRGMISTEST PÕHIMÕTETEST LÄHTUVALT

Tehnoloogia:

- Frontend: React* või Angular või muu
- Backend: .NET
- Andmebaas: MSSQL või PostgreSQL
- Rakenduse tüüp: MVC või Client-Server* (nt Rest API või GraphQL)

Kasutajad: Eraklient, Admin

*Eelistatud lähenemine, kuid mitte rangelt soovitud.

FUNKTSIONAALSED NÕUDED

1) Rakendusel peab olema esileht (nö reklaamleht) sisselogimata kasutajale.

2) Kasutaja peab saama esilehelt registreerida ja sisse logida.

3) Kasutajaid on kahte tüüpi – eraklient, rakenduse admin. Rakenduse admin on kasutaja, mis on juba eeldefineeritud ja tema ei pea saama registreerida. Admin peab saama vaid sisse logida ja admin kasutajal on eraldi vaated (Admin kasutajal on ainult statistika vaade, punkt 8).

4) Rakenduses peab saama luua piiramatult hoiustamiskohtade tasemeid asjade hoiustamiseks.

Näiteks Tuba-1/Karp-5/Kaust-2 või Osakond-3/Ladu-6/Rida-4/Kast-6/Karp-2. Iga kasutaja näeb vaid enda lisatud hoiustamiskohti.

5) Rakenduses peab võimaldama lisada igasse hoiustamiskohta tooteid/asju. Toodete/asjade salvestamisel peab olema võimalik lisada iga toote/asja kohta:

- pilti;
- nimetust;
- seerianumbrit (kui tootel on see olemas);
- ülesande käigus tuleb välja mõelda veel vähemalt 3 parameetrit.

6) Rakenduses peab saama otsida tooteid/asju kõikide parameetrite järgi, välja arvatud pilt. Otsing tagastab tabeli/list vaate. Iga kasutaja näeb vaid enda lisatud tooteid/asju.

7) Otsinguvaatest saab navigeerida toote/asja detailvaatesse. Detailvaatest saab avada punktis 5 loodud muutmise/loomise vormi.

8) Admin kasutaja peab saama näha iga kasutaja kohta statistikat. Näiteks kui palju tooteridu on mingil kasutajal. Ülesande käigus tuleb välja mõelda veel 3 võimalust, mida võiks statistika alt näha.

MITTEFUNKTSIONAALSED NÕUDED

- 1) Projekt peab vigadeta kompileeruma ja käima minema.
- 2) Koodi/Projekti failide struktuur on loogiline.
- 3) Andmemudeli ülesehitus jälgib parimad praktikaid (nt normaliseeritus).
- 4) Koodi lihtsus (mida lihtsam seda parem), Clean Code
- 5) Proovitöös üleantavad osad on git lingina või kokkupakitud zip failina:
 - readme.md – käimapaneku juhend;
 - kood;
 - andmebaas;
 - kui andmebaas on hostitud kuskil avalikult, siis piisab ka ühendusparameetritest (host:port,kasutaja,parool).

SOOVITUSED

Kindlasti proovi oma lahendust käima panna ka sõbra arvutil. Kõige hullem on see kui sinu arvutis kõik töötas ja meie arvutis ei lähe käima. See võib juhtuda näiteks siis kui mõni sõltuvus on vigasena lisatud.