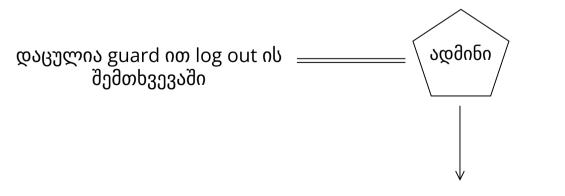
## E-commerce project Anzor Katsiashvili



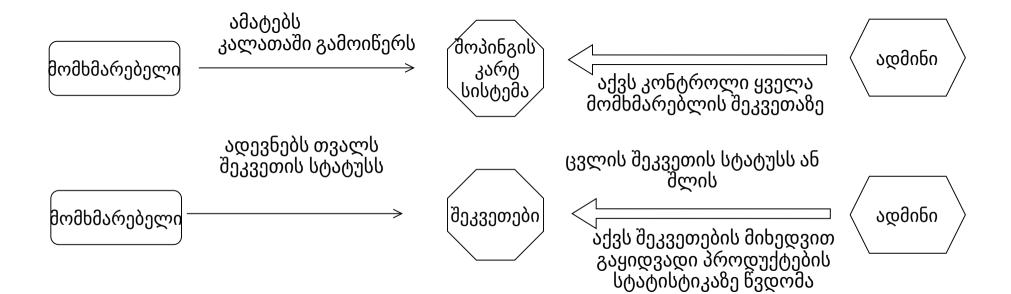
მომხმარებელთა როლები



მაღაზიის მენეჭერი

დაცულია guard ით log out ის შემთხვევაში მომხმარებელი

ხდება მომხმარებლის მისამართის შენახვა





ხედავს ფასდაკლებას და ფილტრავს პროდუქტებს დაკლებული ფასის მიხედვითაც

*(*ფასდაკლება)

პრომოკოდის დამატება

შეუძლია ფასდაკლებების დამატება წაშლა პროდუქტების

ჩამატება ამოშლა გააქტიურება

დეაქტივაცია ასევე თასდაკლების მონაცემების

ცვლილება

ადმინი

მომხმარებელი

შეუძლია პრომოკოდის გამოყენება (ამჟამად Lambda) და პროდუქტის სეილიდან შეკვეთა. (გადადის ჰოუმ როუტგი იქ იხსნება პროდუქტის ვიუ და ვიუს გათიშვის შემთხვევაში ბრუნდება სეილის

შეფასებებით და კომენტარებით

გვერდზე ფილტრავს პროდუქტს 3 മ8നത്. പ്രത്യാന്ത്യർം. თვალს ადევნებს მიტანის სერვისს

იწბილატით სიტხოთობში

ადმინი

პროდუქტი

აქვს წვდომა მსგავს კროდუქციაგი დეტალურად გახსნისას ამატებს პროდუქტს, შლის, ცვლის ჰოუმ ვიუზევე,

ფილტრავს პროდუქტს კატეგორიის და სუბკატეგორიის მიხედვით

კატეგორია

ამატებს, შლის და ცვლის კატეგორიას და სუბკატეგორიას

მომხმარებელი

მომხმარებელი

მომხმარებელი

ადმინი

ამატებს პროდუქტებს (โกซิโงลิโง)

> შეუძლია ფავორიტებიდან შეკვეთაც

ფავორიტებში დამატება





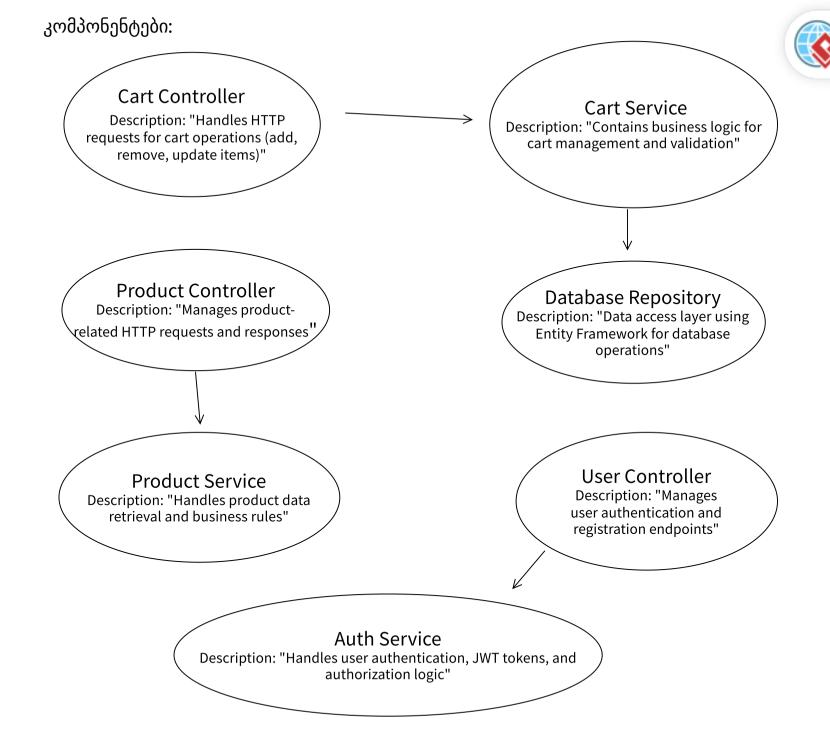
ამატებს შლის ცვლის მომხმარებელს

მომხმარებელი კი ცვლის პროფილის მონაცემებს

ფრონტი ტექნოლოგია Angular, TypeScript, HTML/CSS

ბექენდი ტექნოლოგია ASP.NET Core, C#

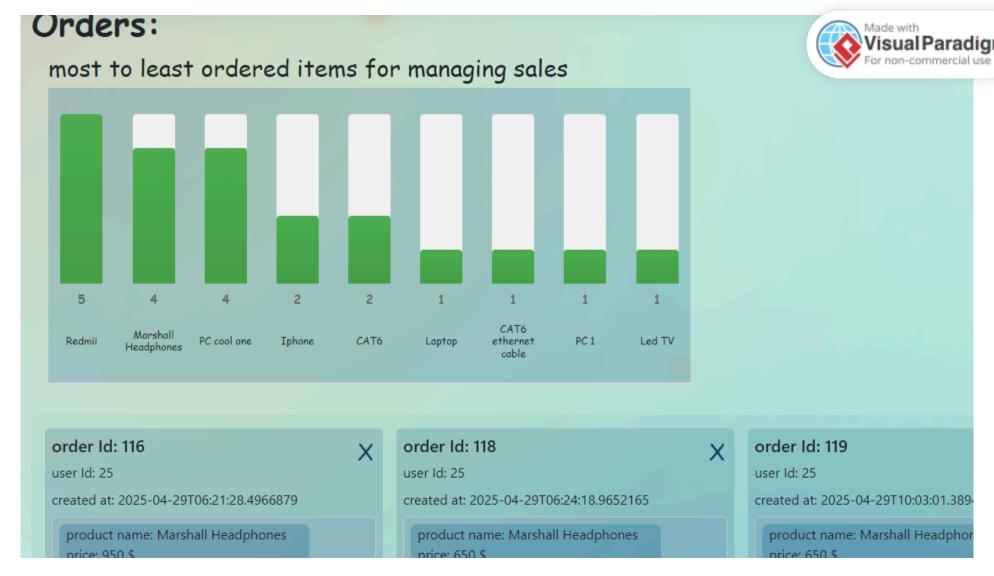
მონაცემთა ბაზა ტექნოლოგია SQL Server/MySQL



Made with

Visual Paradigm

All Services → Database Repository: "Access data layer"



ადმინის პანელის გაყიდვების გრაფიკისთვის მაქვს robust ფუნქციები, ვიყენებ async ngOnInit ს და safe loa უსაფრთხო ჩატვირთვისთვის, ქეშირებისგან თავის არიდებისთვის, error handling და კონსოლის კომენტარებით. cache busting - fresh data სთვის

ასევე ვიყენებ სთეით მენეჯმენტს სხვადასხვა კოპმპონენტებისთვის.

დატა ვალიდაცია - ვამოწმებ მონაცემები ვალიდურია თუ არა და ვიტენებ retry logic ს

ხდება პროდუქტის stock number ის დარეზერვება გადახდამდე

