

UNIVERZITA SV. CYRILA A METODA V TRNAVE

FAKULTA PRÍRODNÝCH VIED

Ústav počítačových technológií a informatiky

StorageTracker

Dokumentácia

Emanuel Cisár

Ročník: druhý

Predmet: Tímový projekt

Akademický rok: 2024/2025

Obsah

1	Úvod.....	6
1.1.1	Technické riešenie projektu.....	6
2	Časť pre neprihláseného užívateľa:	7
2.1	Úvodná obrazovka.....	7
	Vizuál: Ilustračná grafika skladu s policami a krabicami.....	7
2.2	Prihlasovacia obrazovka	7
2.3	Karta pre zabudnuté heslo	8
	Vizuál: Recyklovaný x-layout ako komponenta pre prihlasovací portál.....	8
3	Časť pre prihláseného užívateľa:	8
3.1	Bočný panel	9
	Funkcionalita: Kompaktný navbar na boku pre prehľad funkcionalít skladu. Ítemy v ňom sú komponentou a sú dynamicky upravované vzhľadom k pozícií užívateľa.	9
3.2	Nadpisový kontajner	10
	Funkcionalita: Tento layout je opäť komponentou s params, kde sa dosádza blok blade-html kódu pre dynamický header.....	10
3.3	Kontajner pre správu používateľa	10
	Funkcionalita: Užívateľ tu môže vidieť upozornenia a podrobnosti o svojom profile. Slúži ako primárny spôsob pre odhlásenie.	10
3.4	Prehľad dát užívateľa	10
3.5	Prehľad skladov	11
3.6	Prehľad položiek aktuálneho skladu.....	11
3.7	Prehľad záznamov o transakciách	12
3.8	Prehľad zamestnancov skladu.....	12
3.9	Prehľad dostupných produktov od poskytovateľa	13
3.10	Pridanie nového užívateľa	14
4	Štruktúra kódu.....	15
4.1	Controllers	15
4.1.1	AuthenticatedSessionController	15
	Zaoberá sa autentifikáciou používateľov a následným presmerovaním na príslušné dashboardy v závislosti od ich role.	15
	metódy:	15
	Funkcionalita:	15
4.1.2	RegisteredUserController	15
4.1.3	ProfileController	16
	zameriava sa na správu profilov používateľov, vrátane registrácie, úpravy a odstránenia.	16

Metódy:.....	16
Funkcionalita:	16
4.2 Middleware.....	16
4.2.1 CheckUserRole	16
4.3 Model.....	17
4.3.1 Branch.....	17
Atribúty:.....	17
Vzt'ahy:	17
Funkcionalita:	17
4.3.2 Customer	18
Atribúty	18
Vzt'ahy	18
Funkcionalita	18
4.3.3 Location	18
Atribúty	18
Vzt'ahy	18
Funkcionalita	18
4.3.4 Product.....	19
Atribúty	19
Vzt'ahy	19
Funkcionalita	19
4.3.5 Transaction.....	19
Atribúty	19
Vzt'ahy	19
Funkcionalita	20
4.3.6 User.....	20
Atribúty	20
Vzt'ahy	20
Funkcionalita	20
4.3.7 ProductListing: Zastaraný model.....	20
4.4 Gate 'role'	21
4.5 Factory	21
4.5.1 BranchFactory.....	21
4.5.2 CustomerFactory.....	22
4.5.3 LocationFactory	22
4.5.4 ProductFactory.....	22

4.5.5	Trieda UserFactory	22
4.6	Migrácie	22
4.6.1	Create Users	23
4.6.2	Create Products	23
4.6.3	Create Branches	23
4.6.4	Create Customers	24
4.6.5	Create Transactions	24
4.6.6	Create Location	24
	Tabuľka locations:	24
4.6.7	Create Product Listings	25
	Tabuľka branch_products:	25
4.7	DatabaseSeeder	25
4.8	Router	26
4.8.1	Autentifikácia	26
4.8.2	Profile	27
4.8.3	Web	27

1 Úvod

Ako študenti sme sa rozhodli popasovať s reálnym problémom, ktorý trápí mnohé logistické firmy – neefektívna správa skladov. Výsledkom našej práce je aplikácia StorageTracker, komplexný nástroj navrhnutý tak, aby zjednodušil a zefektívnil každodenné operácie v sklade.

Naším cieľom bolo vytvoriť intuitívne a užívateľsky prívetivé rozhranie, ktoré by umožnilo aj menej technicky zdatným zamestnancom ľahko ovládať aplikáciu. StorageTracker poskytuje ucelený prehľad o stave zásob, pohybe tovaru, objednávkach a ďalších dôležitých aspektoch skladového hospodárstva.

1.1.1 Technické riešenie projektu

Hlavný framework: Laravel

- Laravel je robustný PHP framework, ktorý nám poskytol pevný základ pre náš projekt. Jeho **MVC architektúra** nám umožnila jasne oddeliť logiku aplikácie od prezentácie. **Routing** zjednodušil správu URL adries a **Eloquent ORM** výrazne uľahčil prácu s databázou MySQL. Vďaka Laravelu sme mohli rýchlo vytvoriť základy aplikácie a sústrediť sa na jej špecifické funkcionality.

Frontend technológie:

- **Alpine.js:** Tento jednoduchý, no výkonný JavaScript framework nám umožnil pridať do našich HTML stránok interaktívne prvky bez nutnosti písať zložité JavaScript kódy. Využili sme ho napríklad na vytváranie **dynamických modálnych okien** a **interaktívnych formulárov**. Vďaka Alpine.js je naše rozhranie živšie a responzívnejšie.
- **Blade:** Ako templating engine Laravelu nám Blade umožnil vytvárať **zložité layouty** a **komponenty**, ktoré sa opakovali na viacerých stránkach. To výrazne zjednodušilo údržbu kódu a zlepšilo jeho čitateľnosť.
- **Tailwind CSS:** Tento utility-first CSS framework nám poskytol obrovskú flexibilitu pri tvorbe vzhľadu aplikácie. Vďaka **utility triedam** sme mohli rýchlo a presne upravovať vzhľad jednotlivých prvkov bez nutnosti písať vlastné CSS. To nám umožnilo vytvoriť **prispôsobený dizajn** a zároveň ušetriť čas.
- **Flowbite:** Táto knižnica UI komponentov nám poskytla hotové komponenty, ktoré sme mohli jednoducho integrovať do nášho projektu. Využili sme napríklad komponenty ako **tlačidlá**, **formuláre**, **karty** a **navbar**. Vďaka Flowbite sme získali moderný a profesionálny vzhľad aplikácie bez toho, aby sme museli všetko vytvárať od základu.

2 Časť pre neprihláseného užívateľa:

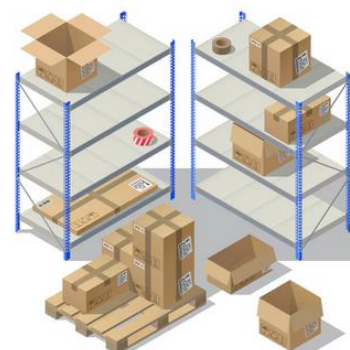
2.1 Úvodná obrazovka

Storage Tracker

Sweep into simple warehouse manager, which offers clean and fresh view for your company.

Log In →

About



Vizuál: Ilustračná grafika skladu s policami a krabicami.

Tlačidlá: "Log In" (Prihlásiť sa) a "About" (O nás).

Funkcionalita: Úvodná stránka slúži ako vstupný bod do aplikácie a poskytuje základné informácie o projekte.

2.2 Prihlasovacia obrazovka



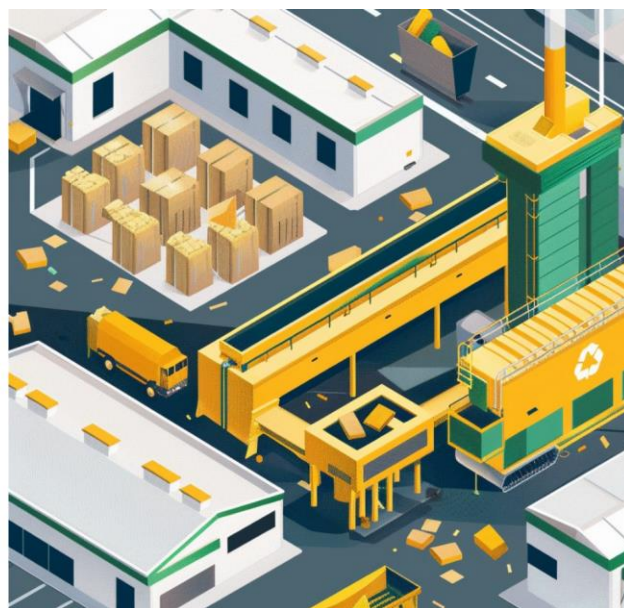
Welcome
Sign In

Email

Password

[Forgot password?](#)

Sign In

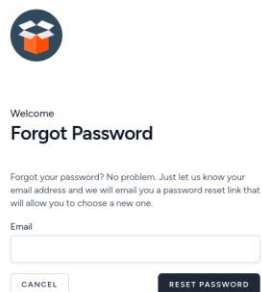


Vizuál: Prehľadné okno pre prihlásenie užívateľov

Tlačidlá: "Sign In" (Prihlásiť sa) a "Forgot password".

Funkcionalita: Prihlasovací portál, kde ako kľúč slúži email.

2.3 Karta pre zabudnuté heslo

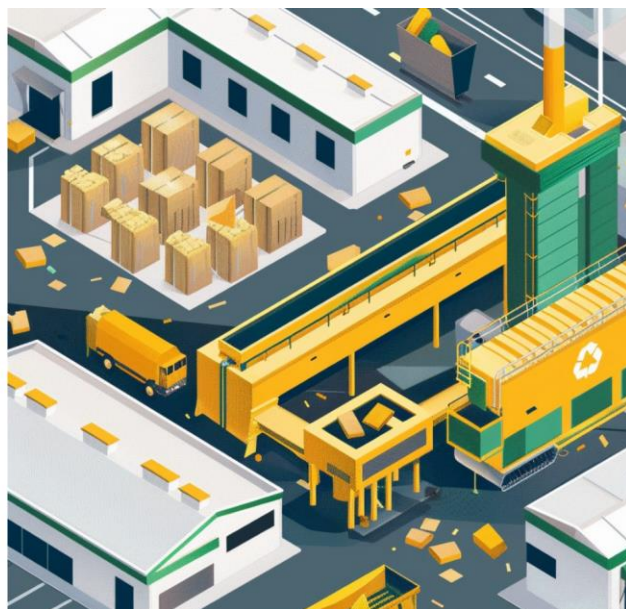


Logo: A circular icon with a stylized 'X' inside.

Welcome
Forgot Password

Forgot your password? No problem. Just let us know your email address and we will email you a password reset link that will allow you to choose a new one.

Email



Vizuál: Recyklovaný x-layout ako komponenta pre prihlasovací portál.

Tlačidlá: Cancel pre návrat na login stránku a Reset Password pre obnovenie hesla.

Funkcionalita: Obnovenie hesla vďaka emailu.

3 Časť pre prihláseného užívateľa:

V tejto sekcii je užívateľ prihlásený. Prihlásiť sa dá vždy iba raz. Aplikácia už prihláseného používateľa automaticky presmeruje z prihlasovacej obrazovky na hlavnú kartu užívateľa.

Aplikácia si tiež drží zoznam prihlásených užívateľov v session.

V našom projekte nerátame s viacerými pozíciami na jedného klienta. Tieto pozície sa delia na:

- Predavač 

Vytvára zásielky na odosielanie zo skladu a doručenie zákazníkom.

Môže predávať zásielky zo skladu aktuálnej pobočky ale nemá právo tieto dokupovať. Môže však o nové položky požiadať.

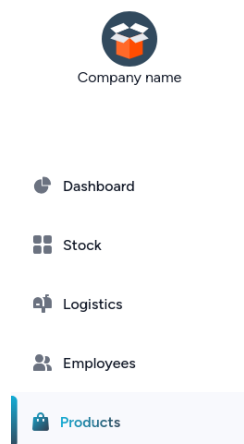
- Manažér 

Riadi zamestnancov a má prístup k nákupu nových položiek do skladu a presunom medzi nimi. Môže zamestnancov upravovať, nie však odstraňovať alebo pridávať.

- Administrátor 

Manažuje sklady a má právo pridávať nových zamestnancov, mazať ich a upravovať.
Pridáva do obehu nové položky ktoré sú dostupné k nákupu.

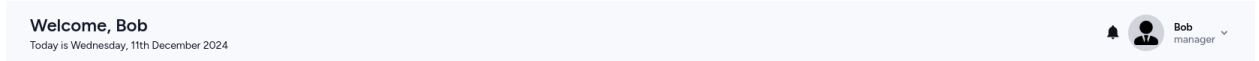
Každá pozícia má karty a funkcie, ktoré sú pre ňu špecifické.



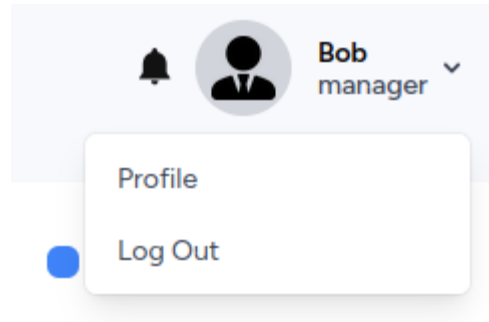
3.1 Bočný panel

Funkcionalita: Kompaktný navbar na boku pre prehľad funkcionalít skladu. Itemy v ňom sú komponentou a sú dynamicky upravované vzhľadom k pozícií užívateľa.

3.2 Nadpisový kontajner



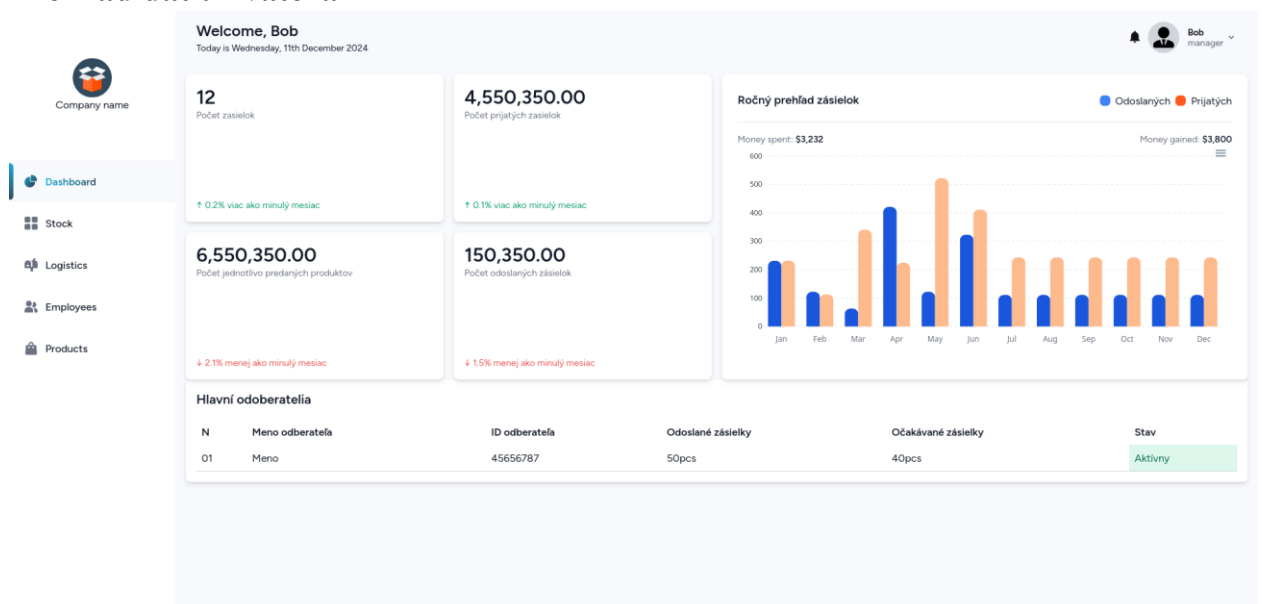
Funkcionalita: Tento layout je opäť komponentou s params, kde sa dosádza blok blade-html kódu pre dynamický header.



3.3 Kontajner pre správu používateľa

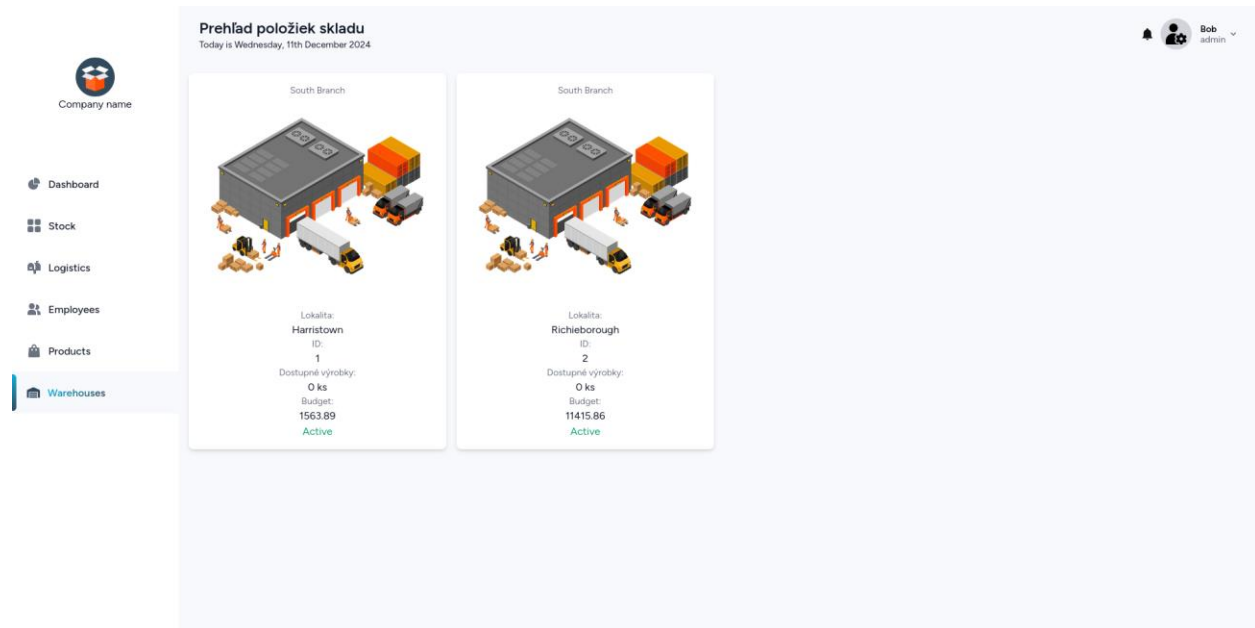
Funkcionalita: Užívateľ tu môže vidieť upozornenia a podrobnosti o svojom profile. Služí ako primárny spôsob pre odhlásenie.

3.4 Prehľad dát užívateľa



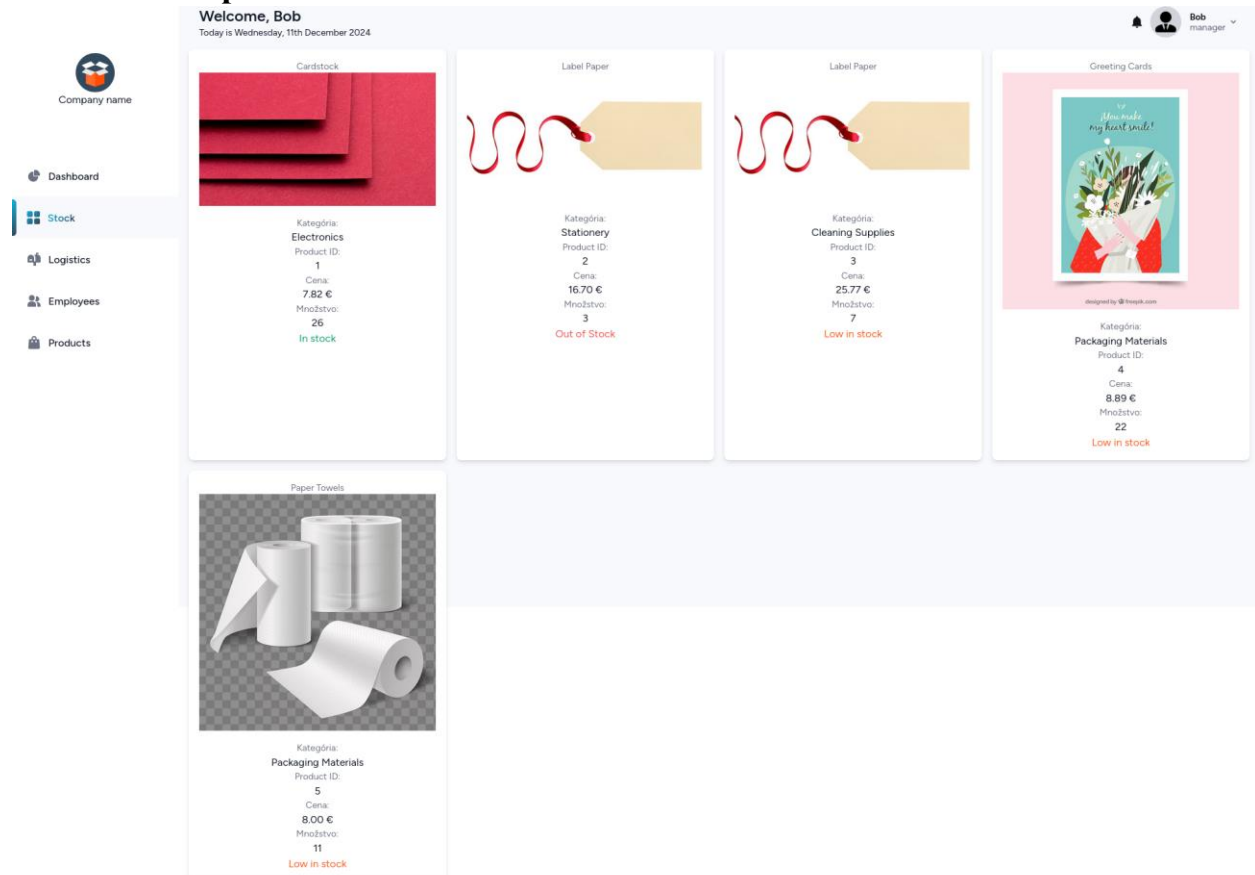
Na tejto komponente môže užívateľ vidieť prehľadné dáta o pohyboch v sklade ako aj tržby, straty, hlavných odoberateľov a celkovo štatistiky.

3.5 Prehľad skladov



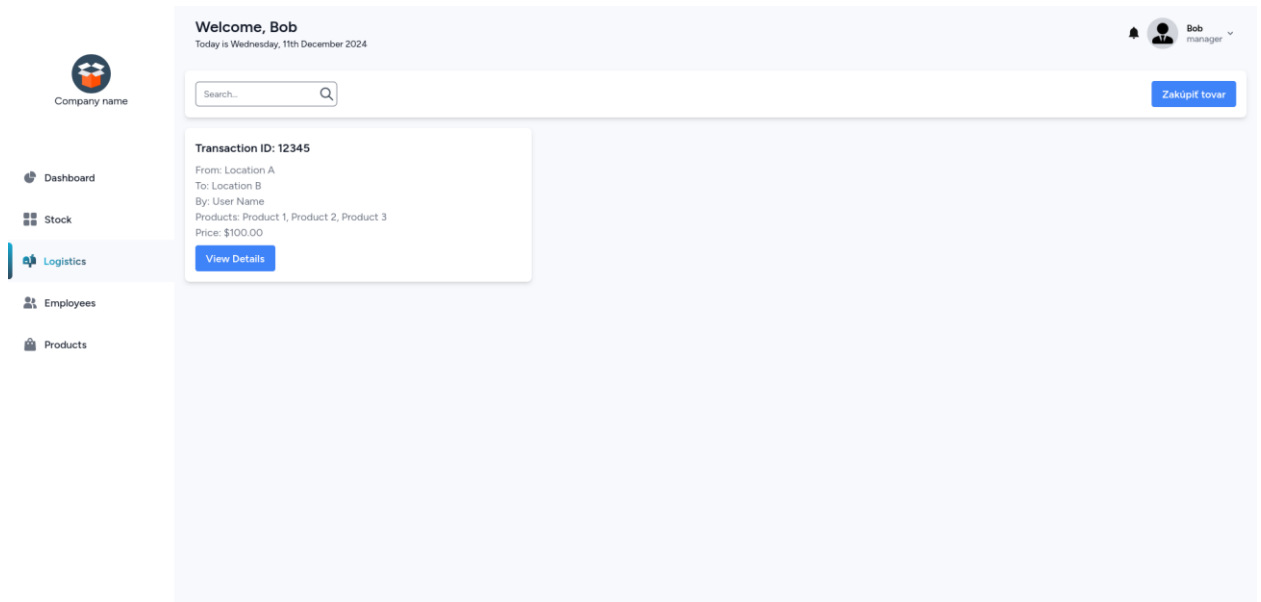
Tento Layout slúži pre admina na prepínanie sa medzi skladiskami.

3.6 Prehľad položiek aktuálneho skladu



Ako názov napovedá, na tejto karte sa nachádzajú položky, ktoré sú dostupné skladom. Sú načítavané z databázy a pre porovnanie stavov (zelená, oranžová, červená) je používaný ich počet 0-5, 5-25, 25 a viac.

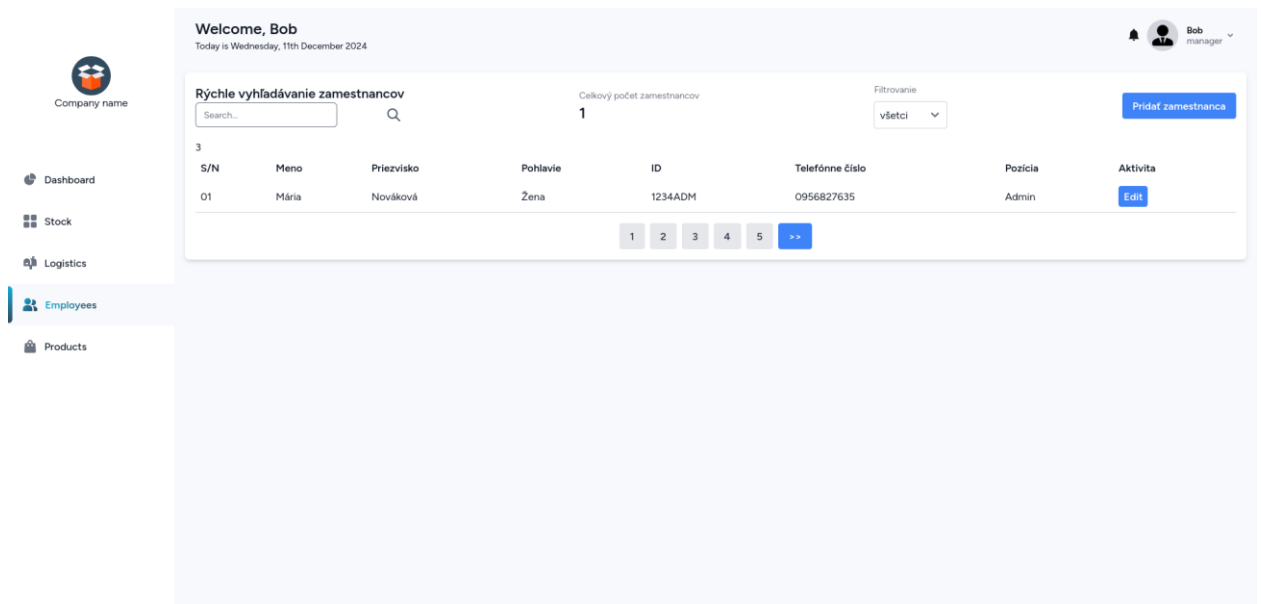
3.7 Prehľad záznamov o transakciách



Toto okno slúži k prehľadu zásielok. Mimo prehľadu transakcií v pobočke má odlišné významy podľa rolí

- Seller: Vytvára nové položky na export.
- Manager: Riadi presuny medzi pobočkami.

3.8 Prehľad zamestnancov skladu



Toto okno je dostupné iba pre manažéra a administrátora.

3.9 Prehľad dostupných produktov od poskytovateľa


Company name

Dashboard

Stock

Logistics

Employees

Products

Welcome, Bob
Today is Wednesday, 11th December 2024

Pridať nový produkt

ID	Názov	Popis	Cena	Kategória	Aktivita
1	 Cardstock	Voluptate ex hic provident non rem. Praesentium quo iure maxime perferendis qui. Dolorum voluptatibus dolorem et ullam sed. Eos quisquam explicabo incididunt et quisquam quae nam.	7.82	Electronics	Nákup
2	 Label Paper	Sed voluptatem officia magni saepe. Cupiditate facere eum at non. Aut aut dolor consequatur sapiente qui. Eius et tenetur quaeat minus laboriosam quisquam facere accusamus.	16.70	Stationery	Nákup
3	 Label Paper	Itaque qui corrupti ut mollitia. Sapiente aliquam sit ratione distinctio reprehenderit dolorem occaecati. Rerum dolore odit tenetur nostrum. Inventore tempore assumenda voluptas unde totam alias.	25.77	Cleaning Supplies	Nákup
4	 Greeting Cards	Atque fugit in explicabo velit qui. Odit laboriosam aut nemo aut. Dolor repellat optio dignissimos dolor cupiditate explicabo veniam.	8.89	Packaging Materials	Nákup
5	 Paper Towels	Non amet eos cum magni quisquam. Sed aut dolor vitae voluptatem quibusdam aut qui. Sit ea quia qui vitae.	8.00	Packaging Materials	Nákup


Company name

Dashboard

Stock

Logistics

Employees

Nákup produktov

Nákup produktov
Pridať produkt

[Back](#)

Nákup produktu

Prostredok

Príloha

Toto okno má opäť mimo prehľadu produktov v pobočke odlišné významy podľa rolí

- Seller: Vyžaduje nákup produktu od dodávkovateľa
- Manager: Nakupuje u dodávateľa a vidí požiadavky od predajcov, ktoré schvaľuje.

3.10 Pridanie nového užívateľa

The screenshot displays the 'Zamestnanci' (Employees) management interface. The left sidebar contains navigation links: Dashboard, Stock, Logistics, Zamestnanci (selected), and Nakup produktu. The main content area is titled 'Pridanie zamestnanca' (Add employee) and includes a 'Back' link. The form consists of a photo upload section on the left and a grid of input fields on the right. The input fields are: Name (First name and Surname), ID, Gender, Position (dropdown), Telephone number, Login, and Password. A 'Pridat' (Add) button is located at the bottom left of the form.

Pridanie zamestnanca

Upload photo

Pridat

Name:

ID:

Position:

Gender:

Telephone number:

Login:

Password:

Karta na pridanie zamestnanca je dostupná iba pre systémového admina. Karta pre úpravu pre admina, manažéra a užívateľa samotného.

4 Štruktúra kódu

4.1 Controllers

4.1.1 AuthenticatedSessionController

Zaoberá sa autentifikáciou používateľov a následným presmerovaním na príslušné dashboardy v závislosti od ich role.

metódy:

- **create():**
 - Skontroluje, či je používateľ prihlásený.
 - Ak áno, presmeruje ho na dashboard podľa role (admin, manager, seller).
 - Ak nie, zobrazí prihlasovací formulár.
- **store():**
 - Ošetrí prihlasovací proces.
 - Po úspešnom prihlásení presmeruje používateľa na príslušný dashboard na základe jeho role.
- **destroy():**
 - Odhlási používateľa a vymaže jeho reláciu.

Funkcionalita:

- **Autentifikácia:** Používa Laravelov zabudovaný systém autentifikácie na overenie prihlasovacích údajov.
- **Autorizácia:** Na základe role používateľa určuje, na ktorý dashboard bude presmerovaný.
- **Presmerovanie:** Po úspešnom prihlásení alebo odhlásení presmeruje používateľa na príslušnú stránku.

4.1.2 RegisteredUserController

Zaoberá sa **registráciou** nových používateľov, validáciou ich vstupných údajov, uložením do databázy a následným prihlásením a presmerovaním na príslušný dashboard na základe priradenej role.

Metódy:

- **create():**
 - Zobrazí formulár pre registráciu nového používateľa.
- **store():**
 - Validuje prijaté údaje (meno, email, telefónne číslo, heslo).
 - Vytvorí nového používateľa v databáze.
 - Priradí používateľovi počiatočné hodnoty (napr. plat, pohlavie).
 - Spustí udalosť Registered (napr. pre zaslanie emailu s potvrdením).
 - Prihlási nového používateľa a presmeruje ho na dashboard zodpovedajúci jeho role.

Funkcionalita:

- **Registrácia:** Umožňuje novým používateľom vytvoriť si účet.

- **Validácia:** Overuje, či sú zadané údaje správneho formátu a či už neexistuje používateľ s rovnakým emailom.
- **Uloženie údajov:** Uloží údaje nového používateľa do databázy.
- **Prihlásenie:** Automaticky prihlási novo zaregistrovaného používateľa.
- **Presmerovanie:** Presmeruje nového používateľa na dashboard zodpovedajúci jeho role (určenej pri vytváraní používateľa alebo neskôr).

4.1.3 ProfileController

zameriava sa na správu profilov používateľov, vrátane registrácie, úpravy a odstránenia.

Metódy:

- **store(Request \$request):**
 - **Registrácia nového používateľa:** Validuje prijaté údaje, vytvorí nového používateľa v databáze, priradí ho k vybranej pobočke s definovanou pozíciou a prihlási ho do systému.
 - **Priradenie k pobočke:** Používa model Branch a metódu attach na priradenie používateľa k konkrétnej pobočke s definovanou pozíciou.
- **edit(Request \$request):**
 - **Zobrazenie formulára pre úpravu profilu:** Načíta údaje prihláseného používateľa a zobrazí ich vo formulári na úpravu.
- **update(ProfileUpdateRequest \$request):**
 - **Aktualizácia profilu:** Aktualizuje údaje používateľa na základe údajov z formulára. Ak sa zmenil email, resetuje čas potvrdenia emailu.
- **destroy(Request \$request):**
 - **Odstránenie používateľského účtu:** Vyžaduje zadanie aktuálneho hesla na potvrdenie. Odhlási používateľa, odstráni jeho účet z databázy a vymaže reláciu.

Funkcionalita:

- **Registrácia:** Umožňuje novým používateľom vytvoriť si účet vrátane priradenia k pobočke a role.
- **Úprava profilu:** Umožňuje používateľom upravovať svoje osobné údaje.
- **Odstránenie účtu:** Umožňuje používateľom odstrániť svoj účet.
- **Validácia:** Overuje, či sú zadané údaje správneho formátu a či už neexistuje používateľ s rovnakým emailom.
- **Priradenie k pobočke:** Pri registrácii alebo úprave profilu umožňuje priradiť používateľa k konkrétnej pobočke a definovať jeho pozíciu.

4.2 Middleware

Middleware je funkcia, ktorá sa vykoná pred každou požiadavkou (request) a po každej odpovedi (response) v aplikácii. Slúži na filtrovanie požiadaviek a vykonávanie rôznych úloh, ako napríklad autentifikácia, autorizácia, logovanie a pod.

4.2.1 CheckUserRole

Slúži na overenie oprávnenia používateľa. Skontroluje, či má prihlásený používateľ požadovanú rolu (napr. admin, manager, seller) pre prístup k danej časti aplikácie.

Kontrola prihlásenia: Overí, či je používateľ prihlásený. Ak nie, presmeruje ho na prihlasovaciu stránku.

Získanie role: Zistí, akú rolu má používateľ v rámci svojej pobočky.

Porovnanie rolí: Porovná zistenú rolu používateľa s požadovanou rolou pre danú časť aplikácie (táto je odovzdaná ako parameter middleware).

Autorizácia: Ak sa role zhodujú, povolí prístup k požadovanej časti aplikácie. Ak nie, presmeruje používateľa na stránku s chybovým hlásením.

4.3 Model

Modely v Laravel predstavujú **abstrakciu databázových tabuliek** v rámci vašej aplikácie. Vďaka nim môžete pracovať s dátami v databáze oveľa intuitívnejšie a elegantnejšie, ako by ste to robili priamym písaním SQL dotazov.

4.3.1 Branch

Reprezentuje pobočku v aplikácii. Služí k definovaniu vzťahov medzi pobočkou a inými entitami, ako sú používatelia, produkty, transakcie, zákazníci a adresy.

Atribúty:

- **budget:** Celkový rozpočet pobočky.

Vzťahy:

- **users():** Mnoho-k-mnohým vzťah s modelom User. Umožňuje priradiť viacerých používateľov k jednej pobočke a naopak. Pivotná tabuľka user_branch ukladá informáciu o pozícii používateľa v pobočke.
- **products():** Mnoho-k-mnohým vzťah s modelom Product. Umožňuje priradiť viaceré produkty k jednej pobočke a naopak. Pivotná tabuľka branch_products ukladá informáciu o množstve produktu a časoch vytvorenia a aktualizácie.
- **transactions():** Mnoho-k-mnohým vzťah s modelom Transaction. Umožňuje priradiť viaceré transakcie k jednej pobočke a naopak. Pivotná tabuľka branch_transaction ukladá časové značky vytvorenia a aktualizácie.
- **customers():** Jeden-k-mnohým vzťah s modelom Customer. Jedna pobočka môže mať viacerých zákazníkov.
- **addresses():** Polymorfný vzťah s modelom Location. Umožňuje priradiť viaceré typy lokalít (napr. dodacia adresa, fakturačná adresa) k pobočke.

Funkcionalita:

- **Správa používateľov:** Umožňuje pridať, odstrániť a spravovať používateľov priradených k pobočke.
- **Správa produktov:** Umožňuje pridať, odstrániť a spravovať produkty dostupné v pobočke, vrátane informácií o množstve.
- **Správa transakcií:** Umožňuje sledovať transakcie spojené s pobočkou.
- **Správa zákazníkov:** Umožňuje spravovať zákazníkov priradených k pobočke.
- **Správa adries:** Umožňuje spravovať rôzne typy adries spojených s pobočkou.

4.3.2 Customer

Predstavuje zákazníka v aplikácii. Slouží na uchovávanie základných informácií o zákazníkovi, ako je jeho meno a obrázok.

Atribúty

- **image_url:** URL adresa obrázka zákazníka.
- **name:** Meno zákazníka.

Vzťahy

- **addresses():** Polymorfný vzťah s modelom Location. Tento vzťah umožňuje priradiť k zákazníkovi rôzne typy adries (napr. dodaciu, fakturačnú), pričom typ adresy je určený v pivotnej tabuľke.

Funkcionalita

- **Uchovávanie informácií o zákazníkoch:** Model slúži ako základ pre správu zákazníkov v aplikácii.
- **Priradenie adries:** Umožňuje priradiť k zákazníkovi jednu alebo viac adries rôzneho typu.

4.3.3 Location

Slúži na reprezentovanie rôznych typov adries v aplikácii. Vďaka polymorfným vzťahom môže byť jedna adresa priradená k rôznym typom modelov, ako napríklad k zákazníkom alebo pobočkám.

Atribúty

- **street_address:** Ulica a číslo domu.
- **city:** Mesto.
- **state:** Štát alebo kraj.
- **postal_code:** Poštové smerovacie číslo.
- **addressable_id:** ID modelu, ku ktorému adresa patrí.
- **addressable_type:** Typ modelu, ku ktorému adresa patrí (napr. "App\Models\Customer", "App\Models/Branch").

Vzťahy

- **addressable():** Polymorfný inverzný vzťah. Umožňuje získať model, ku ktorému adresa patrí. Napríklad, ak chceme zistiť, ku ktorému zákazníkovi patrí daná adresa, môžeme použiť `$location->addressable`.

Funkcionalita

- **Reprezentácia adries:** Slúži ako základ pre uchovávanie rôznych typov adries v aplikácii.
- **Polymorfné vzťahy:** Umožňuje priradenie jednej adresy k rôznym modelom.
- **Flexibilita:** Vďaka polymorfizmu môže byť model Location použitý v rôznych kontextoch.

4.3.4 Product

Reprezentuje produkt v aplikácii. Slouží na uchovávanie základných informácií o produkte, ako je jeho názov, cena a obrázok.

Atribúty

- **image_url:** URL adresa obrázka produktu.
- **name:** Názov produktu.
- **price:** Cena produktu.

Vzťahy

- **branches():** Mnoho-k-mnohým vzťah s modelom Branch. Tento vzťah umožňuje priradiť jeden produkt k viacerým pobočkám a naopak. Pivotalná tabuľka branch_products ukladá informáciu o množstve produktu v danej pobočke a časové značky vytvorenia a aktualizácie.
- **productListings():** TODO: Tento vzťah nie je ešte definovaný. Predpokladá sa, že bude slúžiť na spojenie produktu s nejakým zoznamom produktov (napr. zľavové akcie, odporúčané produkty).

Funkcionalita

- **Uchovávanie informácií o produktoch:** Slúži ako základ pre správu produktov v aplikácii.
- **Priradenie produktov k pobočkám:** Umožňuje spravovať dostupnosť produktov v jednotlivých pobočkách.
- **Správa zásob:** V kombinácii s pivotnou tabuľkou branch_products umožňuje sledovať množstvo produktu v jednotlivých pobočkách.

4.3.5 Transaction

Reprezentuje transakciu v aplikácii. Slouží na zaznamenanie pohybu tovaru alebo iných aktív medzi rôznymi miestami (napr. sklady, pobočky) a je spojený s konkrétnym používateľom, ktorý transakciu vykonal.

Atribúty

- **from_location:** Identifikátor miesta, odkiaľ sa tovar pohybuje.
- **to_location:** Identifikátor miesta, kam sa tovar pohybuje.
- **user_id:** Identifikátor používateľa, ktorý transakciu vykonal.
- **products:** Detaily o produktoch, ktoré sú predmetom transakcie (možno uložiť ako JSON alebo referencovať inú tabuľku).
- **price:** Celková cena transakcie.

Vzťahy

- **fromLocation():** Patrí k miestu pôvodu transakcie.
- **toLocation():** Patrí k miestu určenia transakcie.

- **user():** Patrí k používateľovi, ktorý transakciu vykonal.

Funkcionalita

- **Zaznamenávanie pohybu tovaru:** Slouží na sledovanie pohybu tovaru medzi rôznymi miestami.
- **Priradenie k používateľovi:** Umožňuje sledovať, ktorý používateľ vykonal danú transakciu.
- **Detaily o transakcii:** Uchováva informácie o produktoch, cene a iných relevantných údajoch.

4.3.6 User

Reprezentuje používateľa v aplikácii. Slouží na uchovávanie informácií o používateľoch, ako sú meno, email, heslo, telefónne číslo, pozícia a plat.

Atribúty

- **name:** Meno používateľa.
- **email:** Emailová adresa používateľa.
- **password:** Heslo používateľa (uložené v hašovanej forme).
- **phone:** Telefónne číslo používateľa.
- **position:** Pozícia používateľa v rámci organizácie (napr. administrátor, manažér, predajca).
- **salary:** Plat používateľa.

Vzťahy

- **location():** HasMany vzťah k modelu Branch. Tento vzťah je trochu neštandardný, pretože jeden používateľ môže mať viacero pobočiek (napr. administrátor).
- **branches():** BelongsToToMany vzťah k modelu Branch. Tento vzťah umožňuje priradiť viacerých používateľov k jednej pobočke a naopak. Pivotalná tabuľka user_branch ukladá informáciu o pozícii používateľa v danej pobočke.
- **getBranch():** Pomocná metóda, ktorá vráti prvú nájdenú pobočku spojenú s používateľom (predpokladá, že jeden používateľ má len jednu pobočku).
- **getRoleInBranch():** Pomocná metóda, ktorá vráti pozíciu používateľa v konkrétnej pobočke.

Funkcionalita

- **Reprezentácia používateľov:** Slúži ako základ pre správu používateľov v aplikácii.
- **Priradenie k pobočkám:** Umožňuje priradiť používateľov k jednej alebo viacerým pobočkám a sledovať ich pozície.
- **Autentifikácia:** Používa sa na overovanie prihlasovacích údajov používateľov.

4.3.7 ProductListing: Zastaraný model

Pôvodne zamýšľaný ako abstrakcia nad produktom, ktorá mala slúžiť na sledovanie množstva konkrétneho produktu v rôznych kontextoch. Napríklad sme mohli mať rôzne zoznamy produktov pre akcie, sklady alebo iné účely. Produkt by bol v tomto prípade len identifikátorom typu produktu v zozname.

Aký bol pôvodný zámer vzťahov?

- **products():** Mal byť vzťah jeden-k-jednému, čo naznačuje, že každý záznam v ProductListing by odkazoval na presne jeden produkt.
- **transactions():** Mal byť vzťah mnoho-k-mnohým, čo by umožnilo priradiť jeden zoznam produktov k viacerým transakciám a naopak.
- **branches():** Mal byť vzťah mnoho-k-mnohým, čo by umožnilo priradiť jeden zoznam produktov k viacerým pobočkám a naopak.

Prečo sa od neho upustilo?

- **Zjednodušenie:** Pivotné tabuľky ponúkajú jednoduchší a priamejší spôsob, ako spravovať takéto vzťahy.
- **Flexibilita:** Pivotné tabuľky umožňujú pridávať ďalšie stĺpce (napr. dátum platnosti, zľavu) bez nutnosti upravovať model.
- **Výkon:** V niektorých prípadoch môžu byť pivotné tabuľky výkonnejšie ako samostatné modely.

4.4 Gate 'role'

Gate 'role' slúži na overenie, či má prihlásený používateľ určitú rolu v rámci systému. Táto definícia sa využíva pri autorizácii prístupu k rôznym častiam aplikácie.

Ako to funguje:

1. **Konverzia rolí na pole:** Ak je parameter \$roles reťazec, prevedie sa na pole, aby sa dalo ľahšie porovnávať s aktuálnou rolou používateľa.
2. **Získanie prvej pobočky:** Získa sa prvá pobočka, ku ktorej je používateľ priradený. Ak používateľ nemá žiadnu pobočku, autorizácia zlyhá.
3. **Získanie aktuálnej role:** Zistí sa aktuálna rola používateľa v získanej pobočke.
4. **Porovnanie rolí:** Skontroluje sa, či sa aktuálna rola používateľa nachádza v zozname požadovaných rolí. Ak áno, autorizácia je úspešná.

V Blade šablónach môžeme použiť direktívu @can na overenie oprávnení.

V definícii trás môžete použiť middleware can.

4.5 Factory

4.5.1 BranchFactory

Slúži na **automatické generovanie falošných údajov o pobočkách** pre účely testovania alebo inicializácie databázy.

Funkcionalita:

- **Definícia štruktúry pobočky:** Táto trieda určuje, aké atribúty bude mať každá vygenerovaná pobočka. V tomto prípade sú to:
 - **name:** Názov pobočky (vyberá sa náhodne zo zoznamu preddefinovaných názvov)
 - **city:** Názov mesta, kde sa pobočka nachádza (generuje sa náhodne)
 - **budget:** Rozpočet pobočky (generuje sa náhodné číslo v určitom rozsahu)

4.5.2 CustomerFactory

Generuje falošné záznamy o **zákazníkoch a predajcoch**: Každý vygenerovaný záznam obsahuje:

- **Názov spoločnosti:** Vyberá sa náhodne zo zoznamov definovaných v konštantách.
- **URL adresy loga:** K vybranému názvu spoločnosti sa priradí zodpovedajúca URL adresa loga.

4.5.3 LocationFactory

Generuje falošné záznamy o **lokalitách**: Každý vygenerovaný záznam obsahuje:

- **ID zákazníka (voliteľné):** 50% šanca, že sa vytvorí nový záznam zákazníka a jeho ID sa priradí k lokalite.
- **ID pobočky (voliteľné):** 75% šanca, že sa vytvorí nový záznam pobočky a jeho ID sa priradí k lokalite.

4.5.4 ProductFactory

Táto trieda obsahuje dve konštanty:

- **PRODUCTS:** Obsahuje zoznam názvov produktov a ich zodpovedajúcich URL obrázkov.
- **CATEGORIES:** Obsahuje zoznam kategórií, z ktorých sa náhodne vyberá pre každý produkt.

Generuje falošné záznamy o **produktoch**: Každý vygenerovaný záznam obsahuje:

- **Náhodný výber produktu:** Pri každom vygenerovaní sa náhodne vyberie názov produktu zo zoznamu v triede ProductData.
- **Výber URL obrázku:** Na základe vybraného názvu produktu sa z triedy ProductData získa URL adresa jeho obrázku.
- **Náhodná kategória:** Pre každý produkt sa náhodne vyberie kategória z zoznamu v triede ProductData.
- **Generovanie popisu:** Využije sa knižnica Faker na vygenerovanie krátkeho popisu produktu.
- **Náhodná cena:** Náhodne sa vygeneruje cena produktu v rozmedzí od 5 do 30 s dvoma desatinnými miestami.
- **Vytváranie záznamov:** Na základe vygenerovaných údajov sa vytvorí nový záznam o produkte v databáze.

4.5.5 Trieda UserFactory

Generuje falošné záznamy o **užívateľoch**: Každý vygenerovaný záznam obsahuje:

- **name:** Náhodne vygenerované meno používateľa.
- **email:** Náhodne vygenerovaná unikátna e-mailová adresa.
- **email_verified_at:** Čas overenia e-mailovej adresy.
- **phone:** Náhodne vygenerované telefónne číslo.
- **salary:** Náhodne vygenerovaný plat v určitom rozsahu.
- **password:** Heslo, ktoré je predvolene nastavené na 'password'.
- **remember_token:** Náhodný token používaný pri pamätaní prihlásenia.

4.6 Migrácie

Migrácie v Laravel-e sú PHP súbory, ktoré definujú zmeny v štruktúre vašej databázy. Slúžia ako verzovaný systém, ktorý vám umožňuje sledovať vývoj databázy počas celého životného cyklu vašej

aplikácie. Každá migrácia reprezentuje konkrétny krok v tejto evolúcii, či už ide o vytvorenie novej tabuľky, úpravu existujúcej alebo jej odstránenie.

4.6.1 Create Users

Vytvorenie tabuľky users:

- **id:** Primárny kľúč pre identifikáciu používateľa.
- **name:** Meno používateľa.
- **email:** E-mailová adresa používateľa (unikátna).
- **email_verified_at:** Čas overenia e-mailovej adresy.
- **password:** Heslo používateľa (hašované).
- **phone:** Telefónne číslo používateľa.
- **salary:** Plat používateľa.
- **remember_token:** Token pre funkciu "Pamätať si prihlásenie".
- **timestamps:** Časové značky vytvorenia a aktualizácie záznamu.

Vytvorenie tabuľky password_reset_tokens:

- **email:** E-mailová adresa používateľa (primárny kľúč).
- **token:** Token na resetovanie hesla.
- **created_at:** Čas vytvorenia tokenu.

Vytvorenie tabuľky sessions:

- **id:** Unikátny identifikátor relácie.
- **user_id:** ID používateľa spojeného s reláciou.
- **ip_address:** IP adresa používateľa.
- **user_agent:** User-agent prehliadača používateľa.
- **payload:** Údaje o relácii.
- **last_activity:** Čas poslednej aktivity používateľa v relácii.

4.6.2 Create Products

Tabuľka "products" obsahuje nasledujúce stĺpce:

- **id:** Unikátny identifikátor každého produktu (autoinkrement).
- **image_url:** URL adresa obrázka produktu.
- **name:** Názov produktu.
- **category:** Kategória produktu.
- **description:** Popis produktu.
- **price:** Cena produktu (desatinné číslo s dvoma desatinnými miestami).
- **timestamps:** Časové značky vytvorenia a poslednej úpravy záznamu.

4.6.3 Create Branches

Táto migrácia definuje štruktúru dvoch databázových tabuliek: branches a user_branch.

Tabuľka branches:

- **id:** Unikátny identifikátor pobočky.
- **name:** Názov pobočky.
- **city:** Mesto, kde sa pobočka nachádza.
- **budget:** Rozpočet pobočky.
- **active:** Boolean hodnota indikujúca, či je pobočka aktívna.
- **timestamps:** Časové značky vytvorenia a poslednej úpravy záznamu.

Tabuľka `user_branch`:

- **id:** Unikátny identifikátor záznamu.
- **user_id:** Cudzí kľúč odkazujúci na používateľa.
- **branch_id:** Cudzí kľúč odkazujúci na pobočku.
- **position:** Pozícia používateľa v pobočke.
- **timestamps:** Časové značky vytvorenia a poslednej úpravy záznamu.

4.6.4 Create Customers

Tabuľka `customers`:

- **id:** Unikátny identifikátor každého zákazníka (autoinkrement).
- **image_url:** URL adresa obrázka zákazníka (ak je dostupný).
- **name:** Meno zákazníka.
- **timestamps:** Časové značky vytvorenia a poslednej úpravy záznamu.

4.6.5 Create Transactions

Tabuľka `transactions`:

- **id:** Unikátny identifikátor transakcie.
- **from_location_id:** Cudzí kľúč odkazujúci na počiatočnú lokalitu transakcie.
- **to_location_id:** Cudzí kľúč odkazujúci na cieľovú lokalitu transakcie.
- **by_user_id:** Cudzí kľúč odkazujúci na používateľa, ktorý transakciu vykonal.
- **price:** Cena transakcie.
- **timestamps:** Časové značky vytvorenia a poslednej úpravy záznamu.

Tabuľka `branch_transaction`:

- **id:** Unikátny identifikátor záznamu.
- **branch_id:** Cudzí kľúč odkazujúci na pobočku.
- **transaction_id:** Cudzí kľúč odkazujúci na transakciu.
- **timestamps:** Časové značky vytvorenia a poslednej úpravy záznamu.

4.6.6 Create Location

Tabuľka `locations`:

- **id:** Unikátny identifikátor lokality.
- **street_address:** Ulica a číslo domu lokality.
- **city:** Mesto lokality.
- **state:** Štát alebo krajina lokality.

- **postal_code:** Poštové smerovacie číslo lokality.
- **addressable_id:** Cudzí kľúč odkazujúci na model, ktorému adresa patrí (môže to byť pobočka alebo zákazník).
- **addressable_type:** Typ modelu, ktorému adresa patrí (napr. "App\Models\Branch" alebo "App\Models\Customer").

Tabuľka **branch_location**:

- **id:** Unikátny identifikátor záznamu.
- **branch_id:** Cudzí kľúč odkazujúci na pobočku.
- **location_id:** Cudzí kľúč odkazujúci na lokalitu.

Tabuľka **customer_location**:

- **id:** Unikátny identifikátor záznamu.
- **customer_id:** Cudzí kľúč odkazujúci na zákazníka.
- **location_id:** Cudzí kľúč odkazujúci na lokalitu.

4.6.7 Create Product Listings

Tabuľka **branch_products**:

- **id:** Unikátny identifikátor záznamu.
- **branch_id:** Cudzí kľúč odkazujúci na pobočku.
- **product_id:** Cudzí kľúč odkazujúci na produkt.
- **amount:** Množstvo produktu v pobočke.
- **timestamps:** Časové značky vytvorenia a poslednej úpravy záznamu.

Tabuľka **transaction_products**:

- **id:** Unikátny identifikátor záznamu.
- **transaction_id:** Cudzí kľúč odkazujúci na transakciu.
- **product_id:** Cudzí kľúč odkazujúci na produkt.
- **amount:** Množstvo produktu v transakcii.
- **timestamps:** Časové značky vytvorenia a poslednej úpravy záznamu.

4.7 DatabaseSeeder

DatabaseSeeder je nástroj v rámci Laravel frameworku, ktorý sa používa na inicializáciu databázy fiktívnymi dátami. V podstate "zasieva" databázu testovacími údajmi, aby sa s ňou dalo pracovať a testovať aplikáciu ešte pred tým, než sa do nej začnú vkladať reálne dáta.

Tento konkrétny seeder využíva tieto modely a generuje pre ne testovacie dáta:

- **User:** Vytvára fiktívnych používateľov s náhodnými údajmi.
- **Branch:** Vytvára fiktívne pobočky s náhodnými údajmi.
- **Product:** Vytvára fiktívne produkty s náhodnými údajmi.
- **Customer:** Vytvára fiktívnych zákazníkov s náhodnými údajmi.

4.8 Router

Router v Laravel sú jadrom každej web aplikácie. Definujú, ako aplikácia reaguje na rôzne HTTP požiadavky (GET, POST, PUT, DELETE, atď.).

4.8.1 Autentifikácia

Tento súbor rout definuje cesty (URL) pre rôzne funkcie súvisiace s autentifikáciou používateľov v Laravel aplikácii. Routy sú rozdelené do dvoch hlavných skupín: pre hostí (užívateľov, ktorí nie sú prihlásení) a pre prihlásených používateľov.

Metódy a Funkcionalita:

- **Prihlasovanie:**
 - GET login - Zobrazí formulár na prihlásenie (trieda AuthenticatedSessionController, metóda create).
 - POST login - Spracovanie prihlasovacieho formulára (trieda AuthenticatedSessionController, metóda store).
- **Registrácia:**
 - GET register - Zobrazí formulár na registráciu (trieda RegisteredUserController, metóda create).
 - POST register - Spracovanie registračného formulára (trieda RegisteredUserController, metóda store).
- **Obnovenie hesla:**
 - GET login/forgot-password - Zobrazí formulár na požiadanie o zresetovanie hesla (trieda PasswordResetLinkController, metóda create).
 - POST forgot-password - Odoslanie emailu s odkazom na obnovenie hesla (trieda PasswordResetLinkController, metóda store).
 - GET reset-password/{token} - Zobrazí formulár na zadanie nového hesla (trieda NewPasswordController, metóda create).
 - POST reset-password - Spracovanie formulára na zadanie nového hesla a jeho uloženie (trieda NewPasswordController, metóda store).
- **Verifikácia emailu:**
 - GET verify-email - Zobrazí výzvu na overenie emailovej adresy (trieda EmailVerificationPromptController).
 - GET verify-email/{id}/{hash} - Overenie emailovej adresy pomocou odkazu z emailu (trieda VerifyEmailController).
 - POST email/verification-notification - Odoslanie nového overovacieho emailu (trieda EmailVerificationNotificationController, metóda store).
- **Zmena hesla pre prihlásených používateľov:**
 - GET confirm-password - Zobrazí formulár na potvrdenie aktuálneho hesla pred zmenou (trieda ConfirmablePasswordController, metóda show).
 - POST confirm-password - Spracovanie formulára na potvrdenie hesla (trieda ConfirmablePasswordController, metóda store).
 - PUT password - Zmena hesla prihláseného používateľa (trieda PasswordController, metóda update).
- **Odhlásenie:**
 - POST logout - Odhlási používateľa z aplikácie (trieda AuthenticatedSessionController, metóda destroy).

Použitie Middleware:

- Všetky routy pre obnovenie hesla sú dostupné iba pre hostí (neprihlásených používateľov) pomocou middleware guest.
- Všetky routy súvisiace so správou hesla pre prihlásených používateľov (overenie, zmena) sú dostupné iba pre prihlásených používateľov pomocou middleware auth.

4.8.2 Profile

Tento súbor definuje rôzne routy (URL adresy) pre rôzne role používateľov v aplikácii: admin, manager a seller. Každá rola má prístup k špecifickým častiam aplikácie.

Middleware: Používajú sa middleware auth, verified a hasRole na kontrolu, či je používateľ prihlásený, overený a má príslušnú rolu.

Routy pre admina:

- /admin: Presmeruje na dashboard admina.
- /admin/dashboard: Zobrazí dashboard admina.
- /admin/stock, /admin/logistics, /admin/employees, /admin/products, /admin/warehouses: Zobrazia rôzne sekcie pre správu skladu, logistiky, zamestnancov, produktov a skladov.
- /admin/employee-add: Zobrazí formulár na pridanie nového zamestnanca.
- /admin/employee-add (POST): Spracováva formulár na pridanie nového zamestnanca.

Routy pre managera:

- /manager: Presmeruje na dashboard managera.
- /manager/dashboard, /manager/stock, /manager/logistics, /manager/employees, /manager/products: Zobrazia rôzne sekcie pre správu skladu, logistiky, zamestnancov a produktov.

Routy pre sellera:

- /seller: Presmeruje na dashboard predajcu.
- /seller/dashboard, /seller/stock, /seller/logistics, /seller/products: Zobrazia rôzne sekcie pre správu skladu, logistiky a produktov.

4.8.3 Web

Tento súbor je hlavným routerom a vyžaduje aj spúšťanie ostatných. Štruktúra:

- **Hlavná stránka:** / - zobrazuje úvodnú stránku
- **Chybová stránka:** /error - zobrazuje chybovú správu pre prihlásených používateľov
- **Profil:**
 - /profile - zobrazuje formulár na úpravu profilu
 - /profile (PATCH) - aktualizuje profilové informácie
 - /profile (DELETE) - maže používateľský účet
- **Autentifikácia a autorizácia:**
- Middleware auth a verified zabezpečujú prístup k chráneným trasám
- **Spracovanie požiadaviek:**
- Kontrolery AuthenticatedSessionController a ProfileController spracúvajú logiku pre autentifikáciu a správu profilov