**Technická dokumentácia:**

Práca je robená vo frameworku Angular. Využíva knižnice [Angular Material](https://material.angular.io/) a [Tailwindcss](https://tailwindcss.com/) pre vzhľad a úpravu stránky. Využíva aj knižnicu [ngx-csv](https://www.npmjs.com/package/ngx-csv) pre generovanie CSV dokumentov.

Pre chod stránky je potrebne spustiť príkaz „ng serve“ vo koreňovom adresári projektu.   
Dependencie:   
- ng add @angular/material  
- npm install --save ngx-csv  
- https://tailwindcss.com/docs/guides/angular

Momentálne aplikácia využíva set dummy dát vytvorených vo main-page.component.ts. Tieto dáta by inak prichádzali zo API volaní. Dáta sú rozdelené na tri intefaces. Základom je interface „Person“. Obsahuje položky pre: meno, priezvisko, email, telefónne číslo, rodné číslo a vek. Z tejto interface následne dedia interface Customer a Personel. Každá z nich má pridané identifikačné číslo, a zamestnanec má aj položku pre fotku. Posledné interface je Contract. Ten obsahuje položky: registračné číslo, inštitúcia, klient, administrátori, dátum vytvorenia, dátum validácie a dátum uzatvorenia. Následne sú tieto dáta zdieľané do ostatných komponent. Medzi základné komponenty patria contact, datail a header.

Zobrazovanie údajov:

Zobrazovanie všetkých údajov sa deje na komponente contract-box. Všetky potrebné údaje sú dedené pomocou @Input. Tieto údaje sú následne zobrazené pomocou \*ngFor. Knižnica tailwind pomáha so zobrazením do mriežky a responzívne sa mení podľa rozlíšenia stránky. Každý element: contract, customer aj employee je rozkliknuteľný pre zobrazenie detailov o entite. V prípade detailov entity contract je ešte možnosť otvorenia bottom sheet detailov jak o entite zákazník tak o správcoch zmluvy.

Vytváranie entít:

Vytváranie sa vykonáva vo komponente header-component. Po rozkliknutí elementu „Actions“ sa zobrazia tri podobné možnosti. Na zaznamenávanie vstupu je použitá knižnica FormBuilder a FormControl. Dáta sú zabalené do FormGroup a ich štruktúra napodobňuje štruktúry pre Contract, Customer a Personel. Tlačidlo pre uloženie sa objaví po vyplnení všetkých fieldov. Na uloženie jednotlivých skupín sú použite tri funkcie. onSaveContract(), onSaveCustomer() a onSaveEmployee(). Všetky fungujú na tom istom princípe. Vytvorí sa instancia potrebných interfaces ktorá sa následne naplní dátami z priliehajúcej FormGroup. Tieto sú následne pomocou @Output emitované a zachytávané na main-page komponente. Tam sú jednoducho pushnuté do zoznamu entít. Keby je program prevedený cez API dáta by boli pridané do celku a následne uložené pomocou POST, alebo PUT.

Vymazávanie entít:

Vymazávanie sa vykonáva na komponente contract-box. Každý kontrakt, zákazník aj zamestnanec majú toto tatíčko vo svojich detailoch. Po kliknutí sa zobrazí dialógové okno. Funkcionalita je vykonaná za pomoci knižnice MatDialog. Pre každý dialog je vytvorená vlastná komponenta. Po jej zavretí odoberáme jej odpoveď. Na základe odpovedi sa rozhodne či je odpoveď emitovaná alebo nie. Emituje sa znova do komponenty main-page. Tam je na základe identifikujúceho čísla entita nájdená a „splicnuta“ zo zoznamu.

Editovanie entít:

Editovanie je znova implementované na komponente contract-box. Taktiež má svoje tlačidlo v detailoch entity. Po stlačení sa otvorí dialógové okno podobne ako u delete. Zaznamenávanie údajov je implementované tak isto ako pri funkciách create. Výsledok je emitovaný do komponenty main-page, kde je následne podľa registration\_number vymenení za originálny.

Vyhľadávanie entít:

Vyhľadávanie kombinuje techniky použité v predošlých funkciách. String ktorý emituje porovnáva s menom a priezviskom buď zákazníkov alebo zamestnancov. Zhody sú pushnuté do nového arraju a jeho hodnoty potom uložené do pôvodnej premennej.

Sťahovanie CSV:

Posledným tlačidlom u detailu entít je tlačidlo pre stiahnutie CSV. Do premennej data sú uložené všetky príslušné údaje danej entity. Do premennej options sú zas nastavené parametre CSV. Následne sú tieto údaje predané do „ngxCsv“ funkcie.