**Dokumentacja techniczna projektu**

1. **Opis wymagań**

Gość rejestruje się w portalu.  
Użytkownik loguje się w portalu.  
Użytkownik ma możliwość edycji swoich danych.  
Użytkownik może zmieniać ilość pieniędzy w portfelu  
Użytkownik wymienia waluty.  
Użytkownik wylogowuje się z portalu.  
Administrator loguje się w portalu.  
Administrator zarządza dostępnymi walutami.  
Administrator wylogowuje się z portalu.

1. **Słownik pojęć**

Gość -> Osoba niezarejestrowana w portalu  
Użytkownik -> Osoba zarejestrowana w portalu  
Dane -> Dane osobowe  
Portfel -> Indywidualne konto z walutami  
Waluta -> Środek pieniężny wykorzystywany do transakcji

1. **Aktorzy systemu**
2. **Przypadki użycia**

Użytkownik:  
- logowanie się do systemu  
- zarządzanie kontem  
- dokonywanie transakcji

Gość:  
- możliwość rejestracji  
- przeglądanie oferty

Administrator:  
- logowanie się do systemu  
- zarządzanie kontami  
- zarządzanie walutami  
- monitorowanie płynności działania systemu

1. **Model dziedziny biznesowej**

OSOBA

- imię: string  
- nazwisko: string  
- mail: string  
- adres: string  
- portfel: double

GOŚĆ

UŻYTKOWNIK

IP: string

*Rejestruje się.  
Przegląda kursy.*

IP: string  
ID: string  
Hasło: string

*Loguje się.  
Dokonuje transakcji*

PORTFEL

TRANSAKCJA

- rodzajWaluty: string  
- iloscWaluty: double

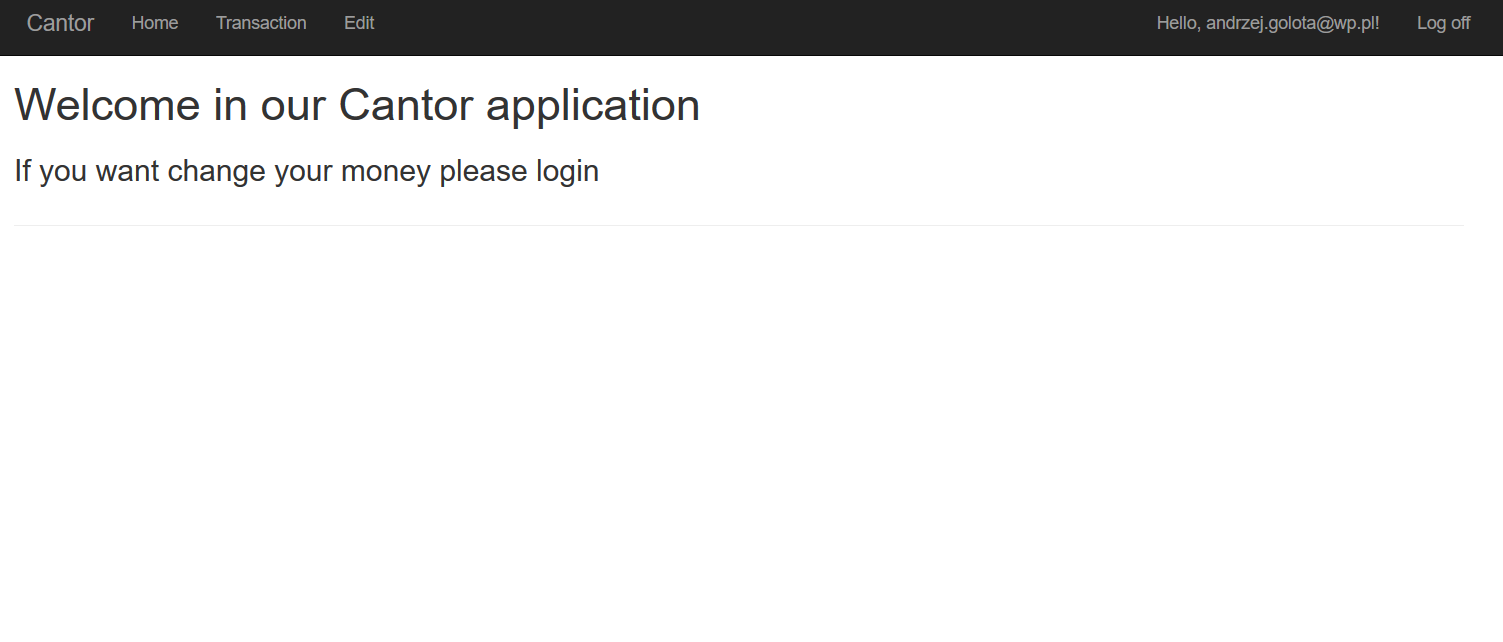
- data: date/time  
- ID: string  
- iloscWaluty: double  
- kurs: double

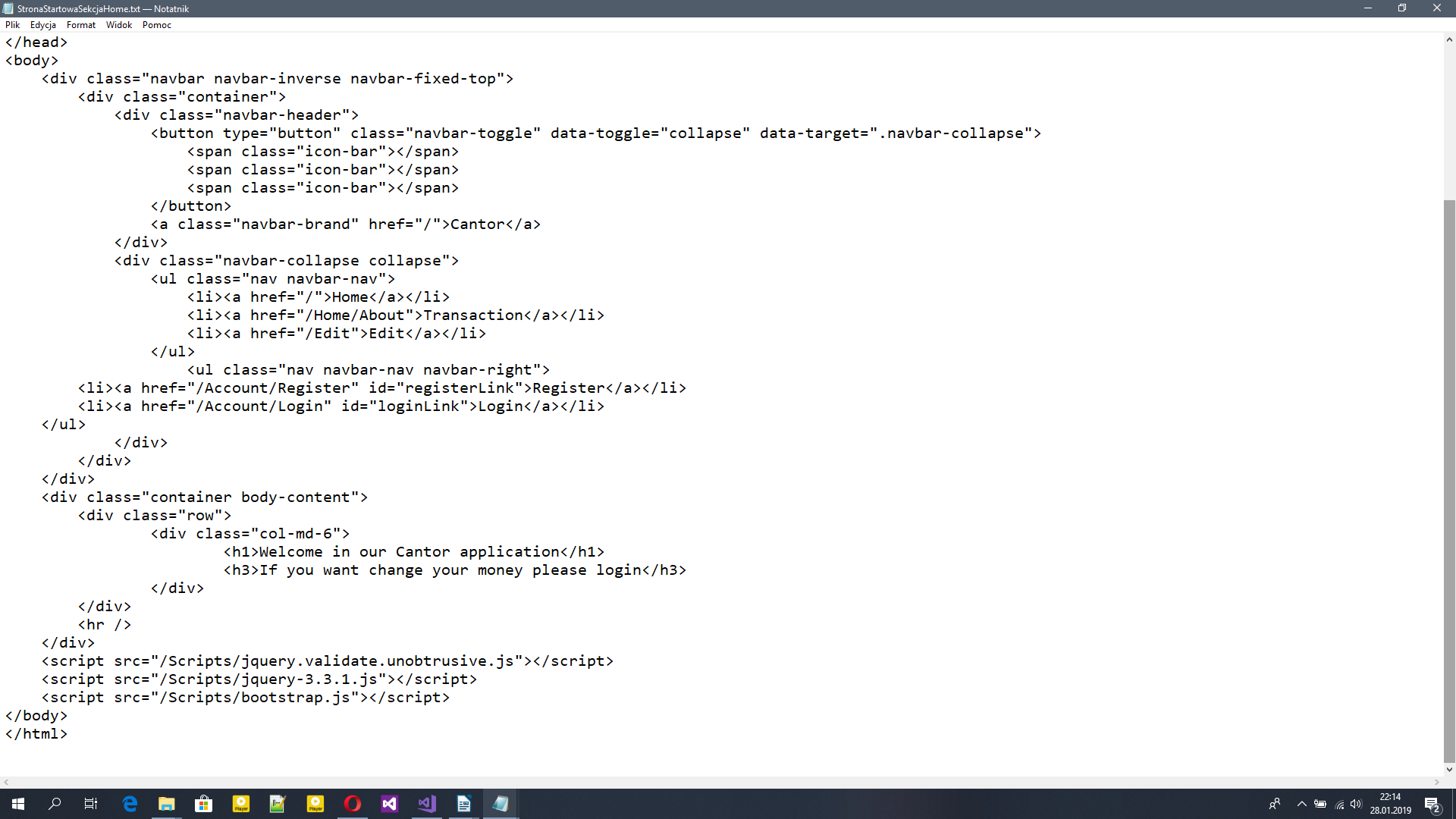
1. **Struktura i powiązania stron WWW**

Całość aplikacji znajduje się na serwerze ASP.NET Core. Za stronę wizualną odpowiada Framework Bootstrap. Wykorzystywany jest w niej model Model-View-Controller (MVC), a kod zapisany głównie w technologii HTML 5 oraz CSS 3. Bieżący kurs walut jest aktualizowany za pomocą JSON’a. Podczas dokonywania transakcji wykorzystywana jest technologia AJAX tak by transakcje się dokonywały bez przeładowywania całej strony. W aplikacji używany jest Dependency Injection za pomocą Autofac'a. Jego celem jest to by absolutnie każdy element aplikacji był całkowicie niezależny od siebie. Do manipulacji danymi w bazie danych CRUD(Create, Read,Update, Delete) wykorzystujemy ORM a konkretnie Entity Framework. Proces autoryzacji użytkowników (logowania i rejestracji) korzystamy z biblioteki ASP.NET Identity.

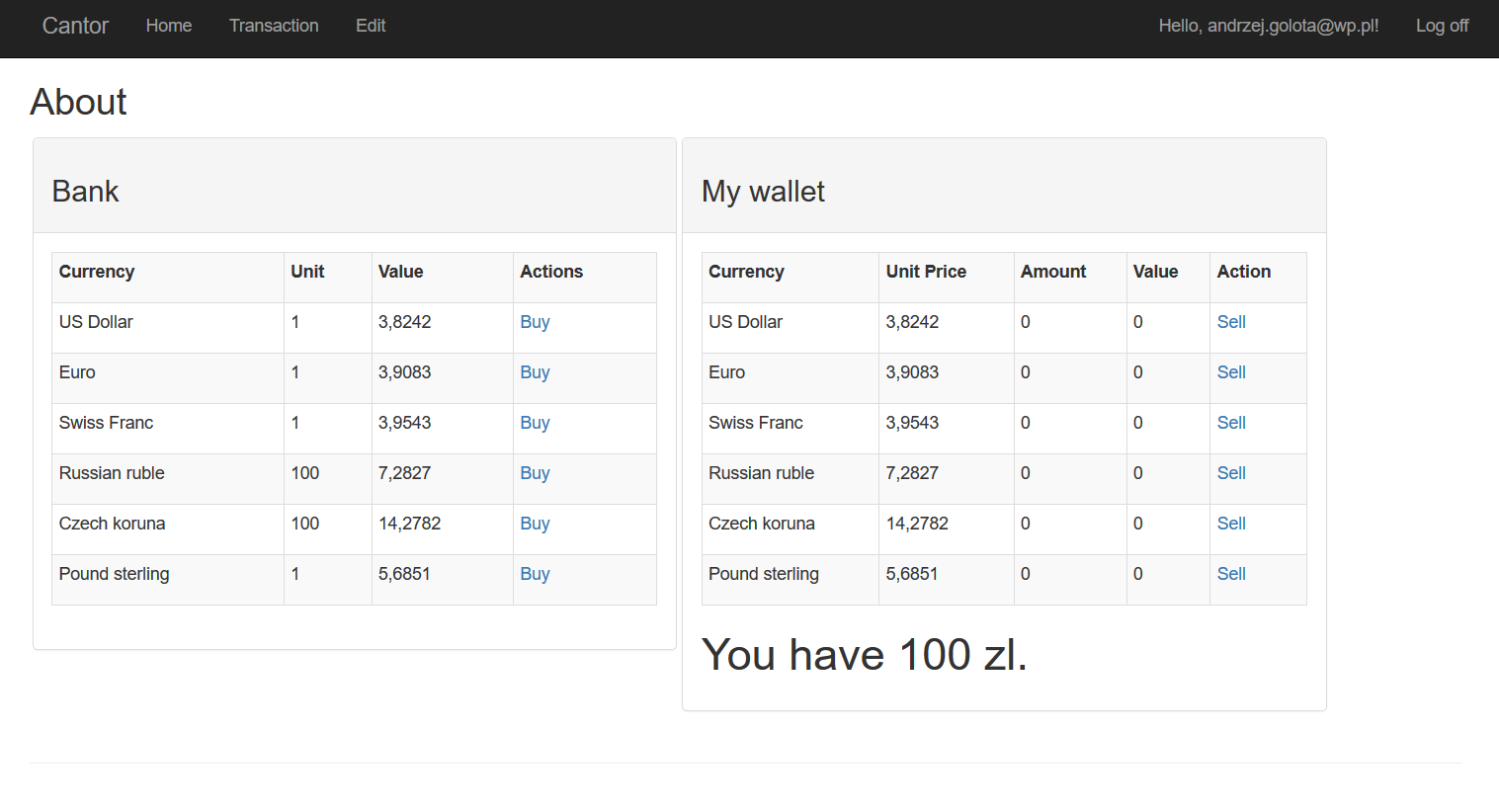
1. **Projekt graficzny i semantyczny stron portalu**

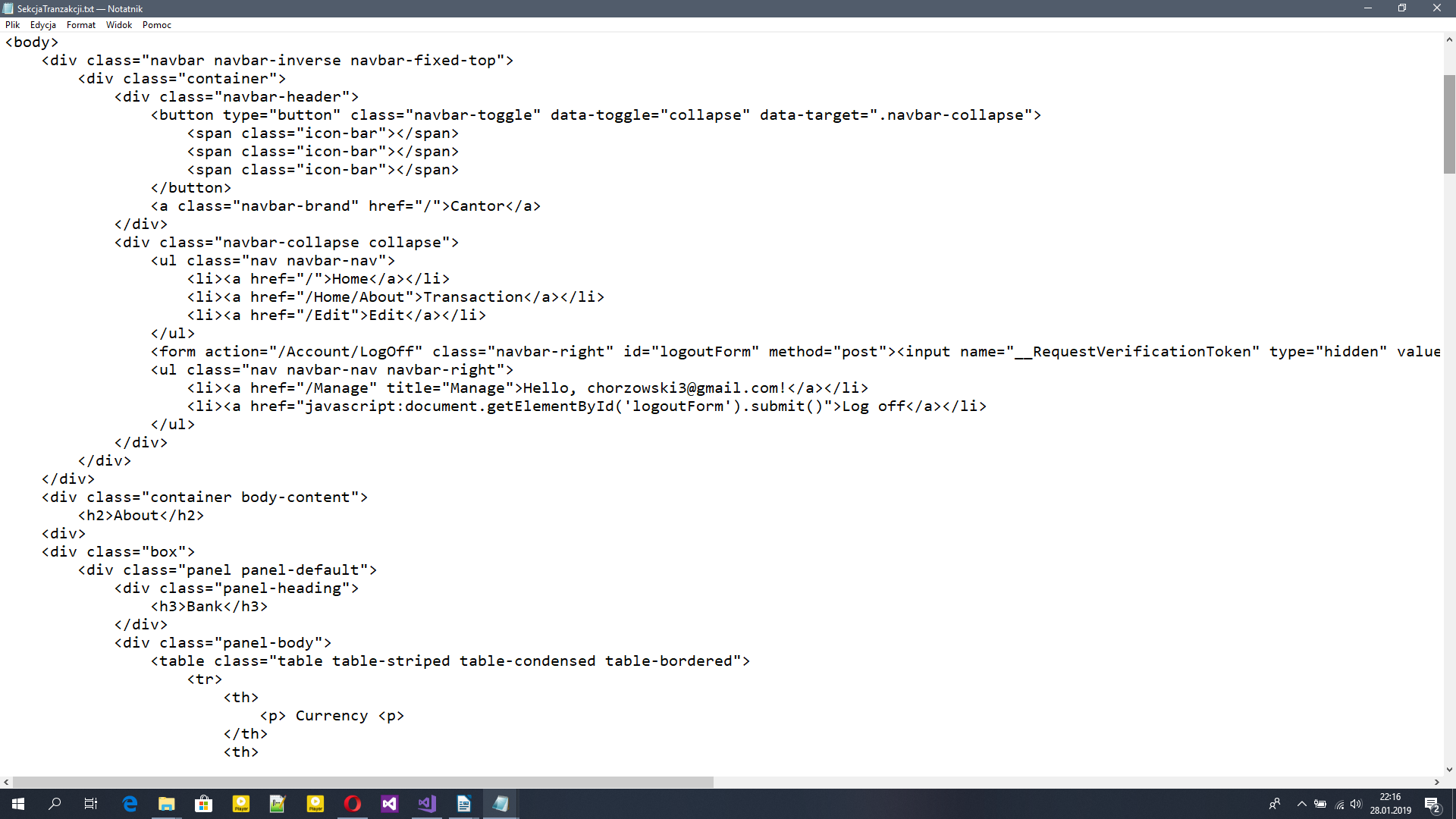
**Strona główna**

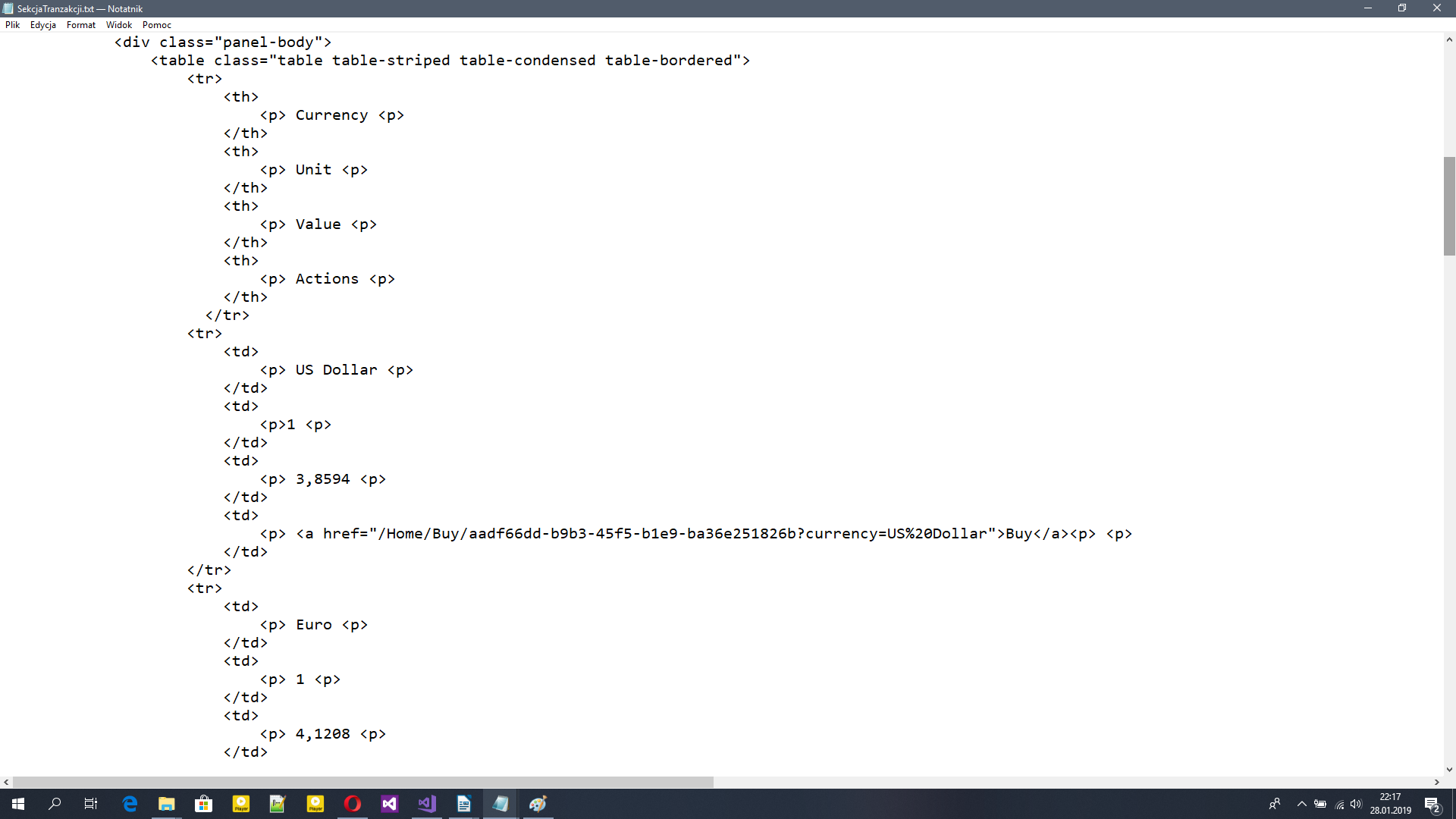


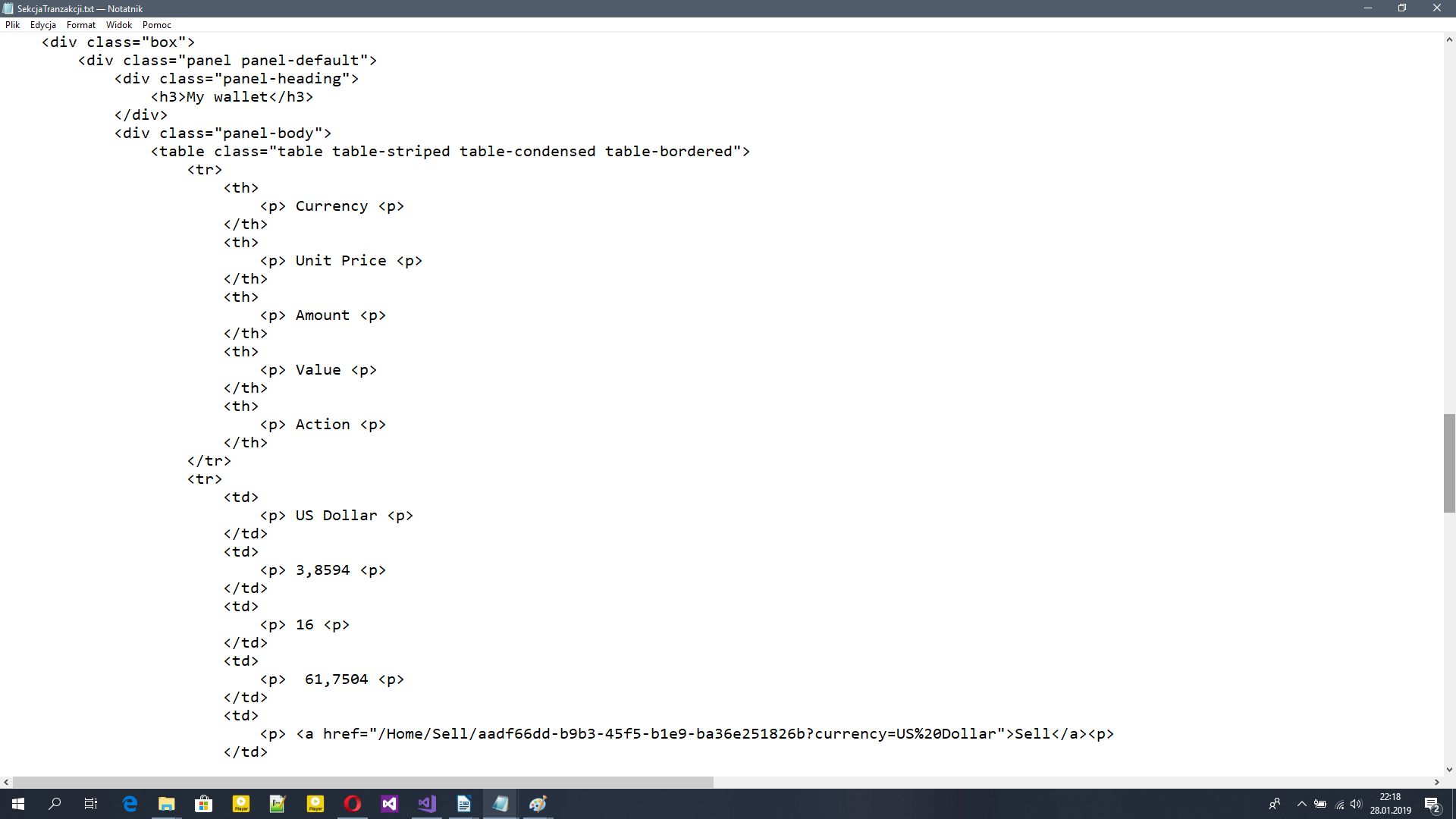
****

**Transacion**

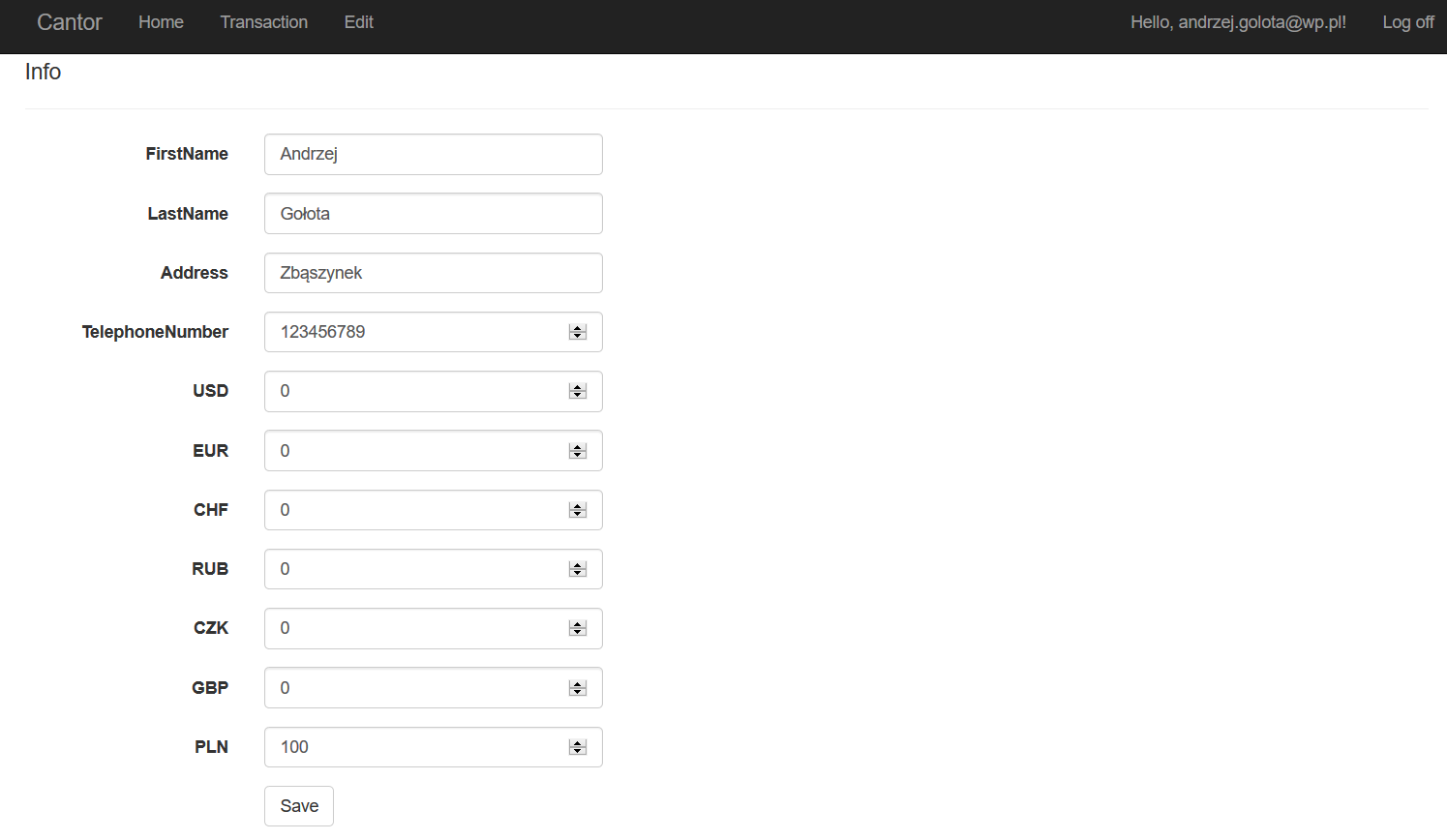


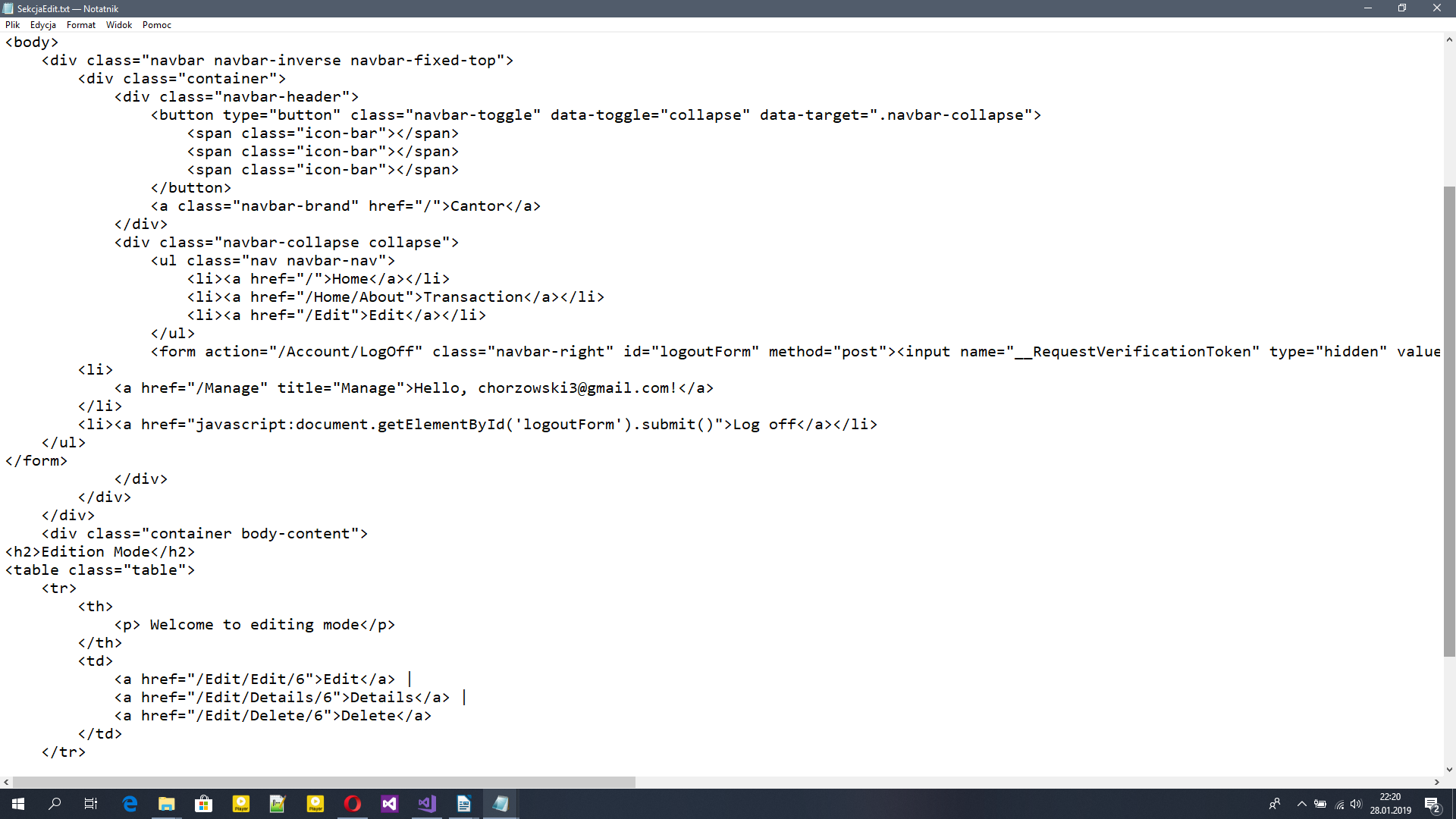
****

****

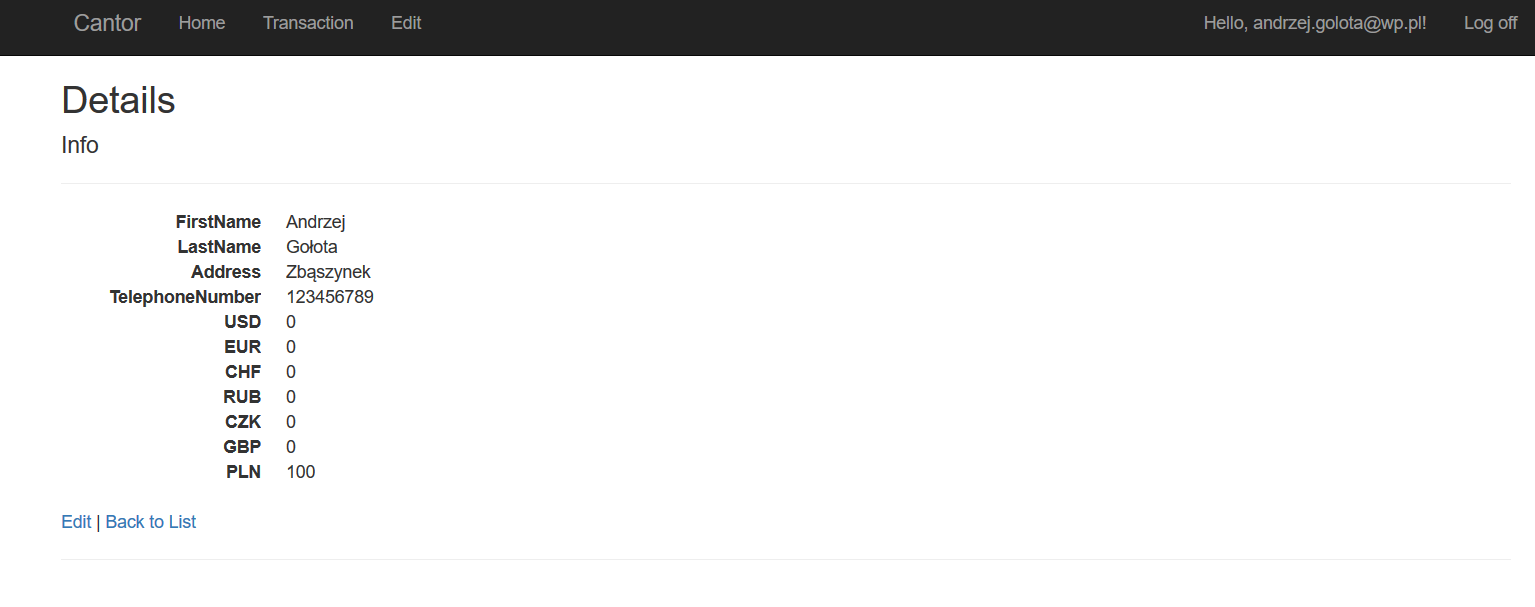
****

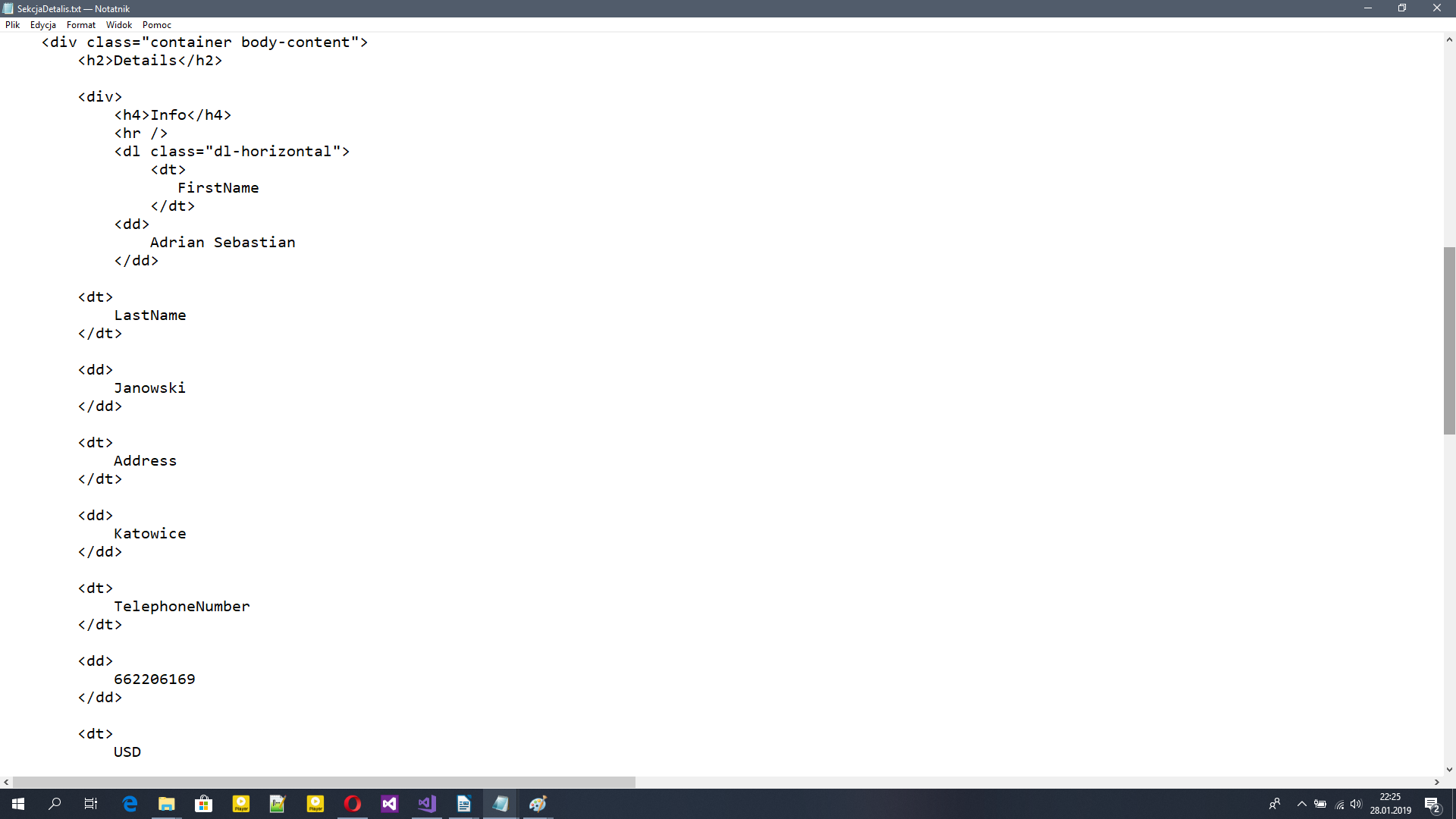
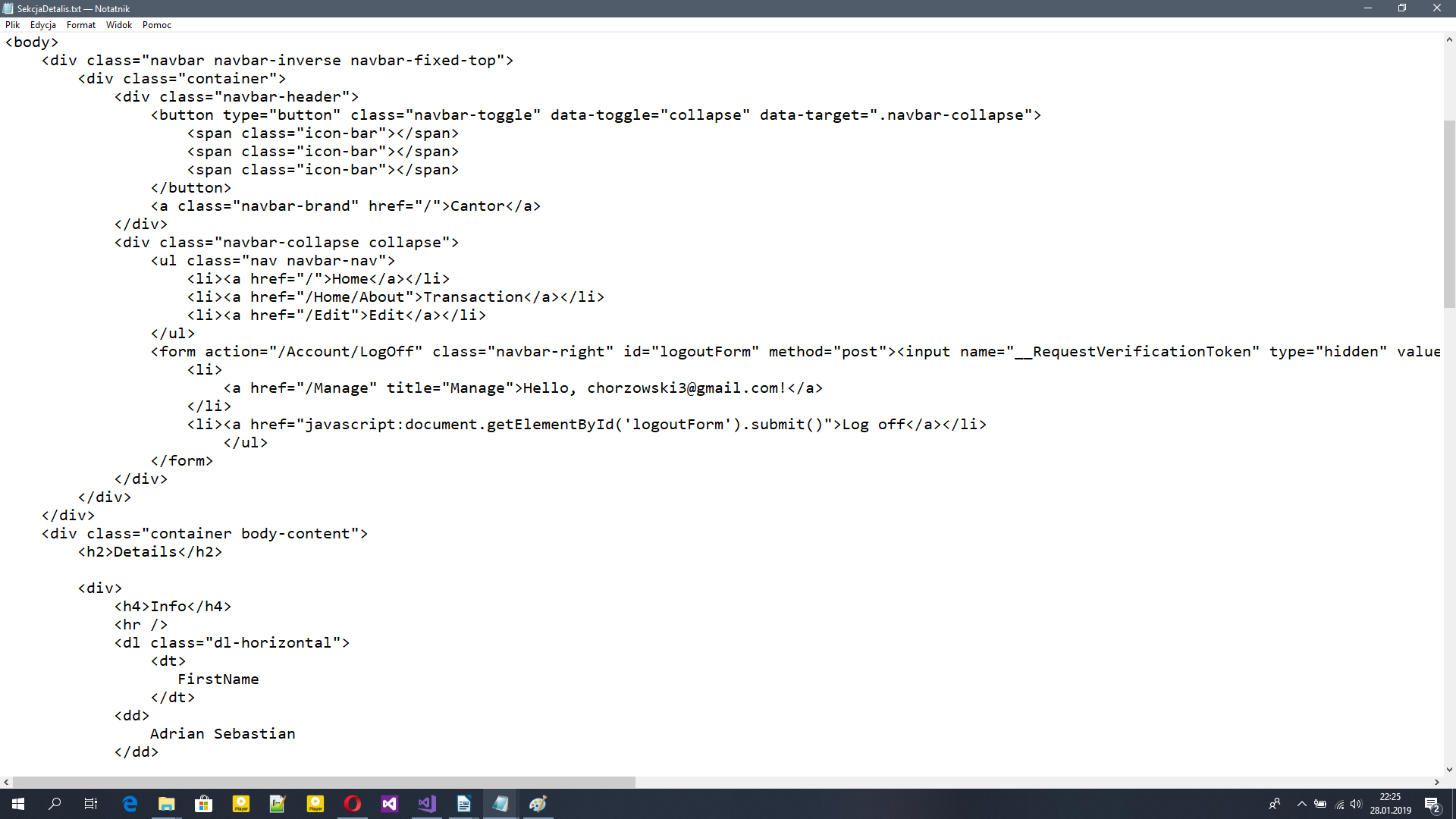
**Edit**



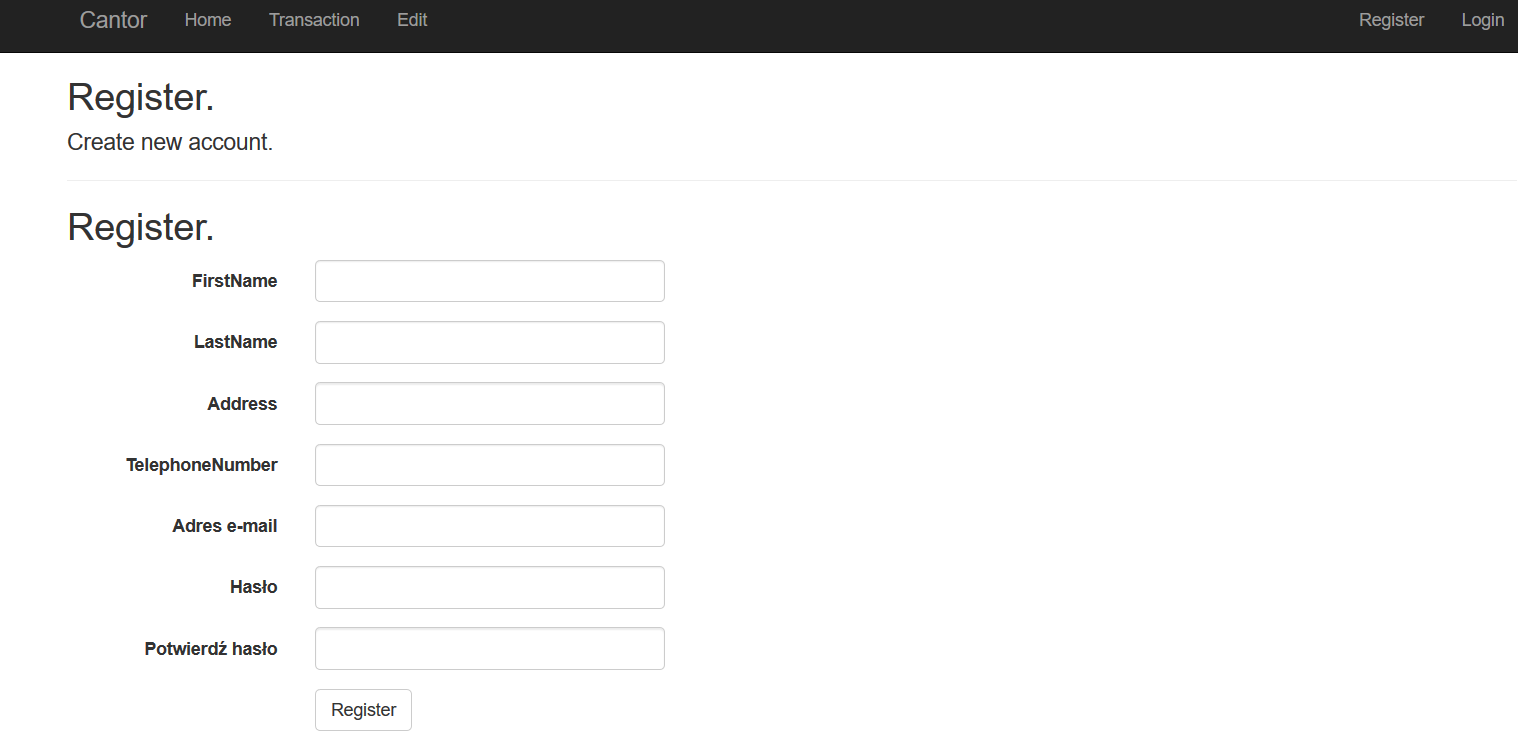
****

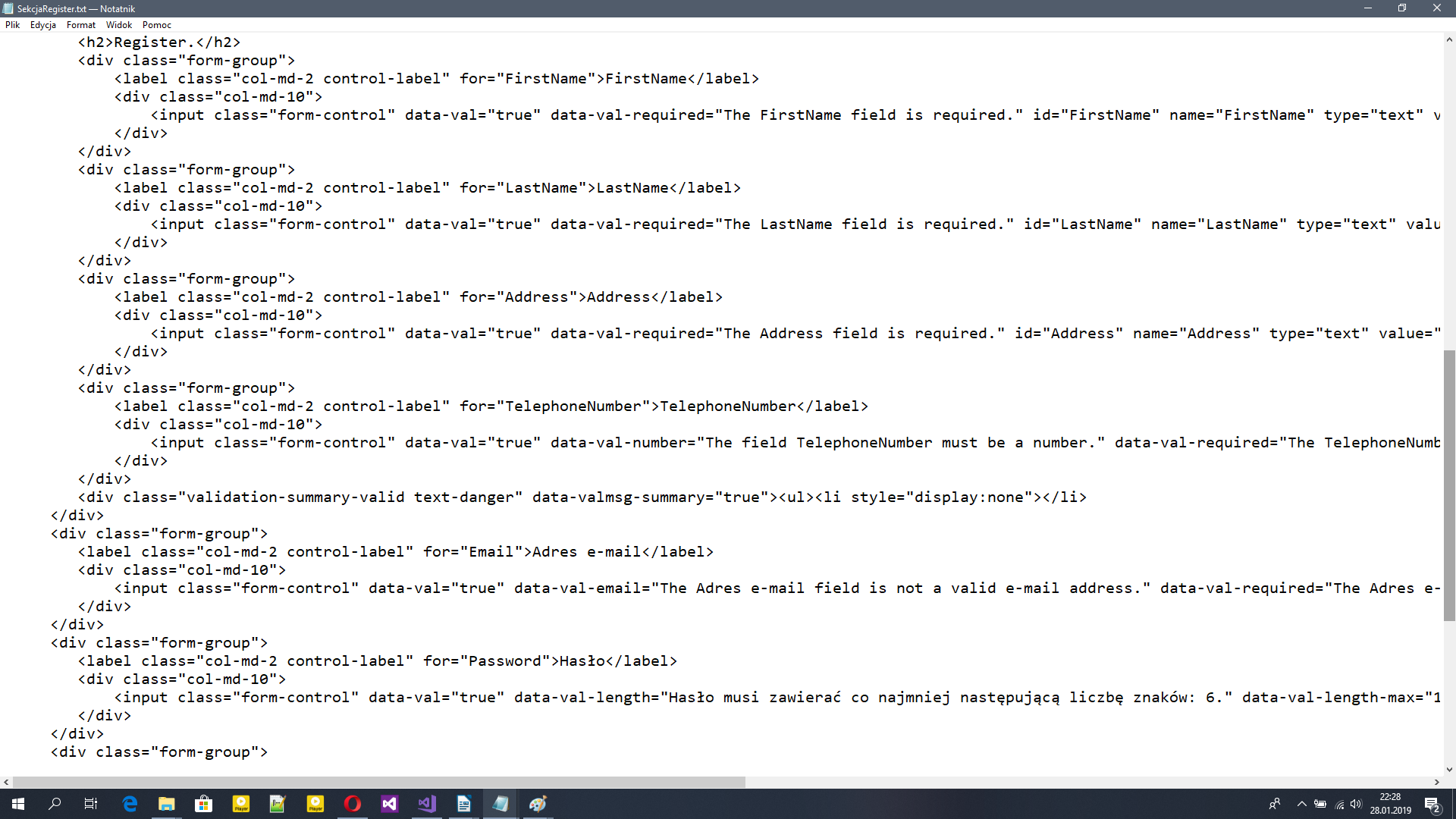
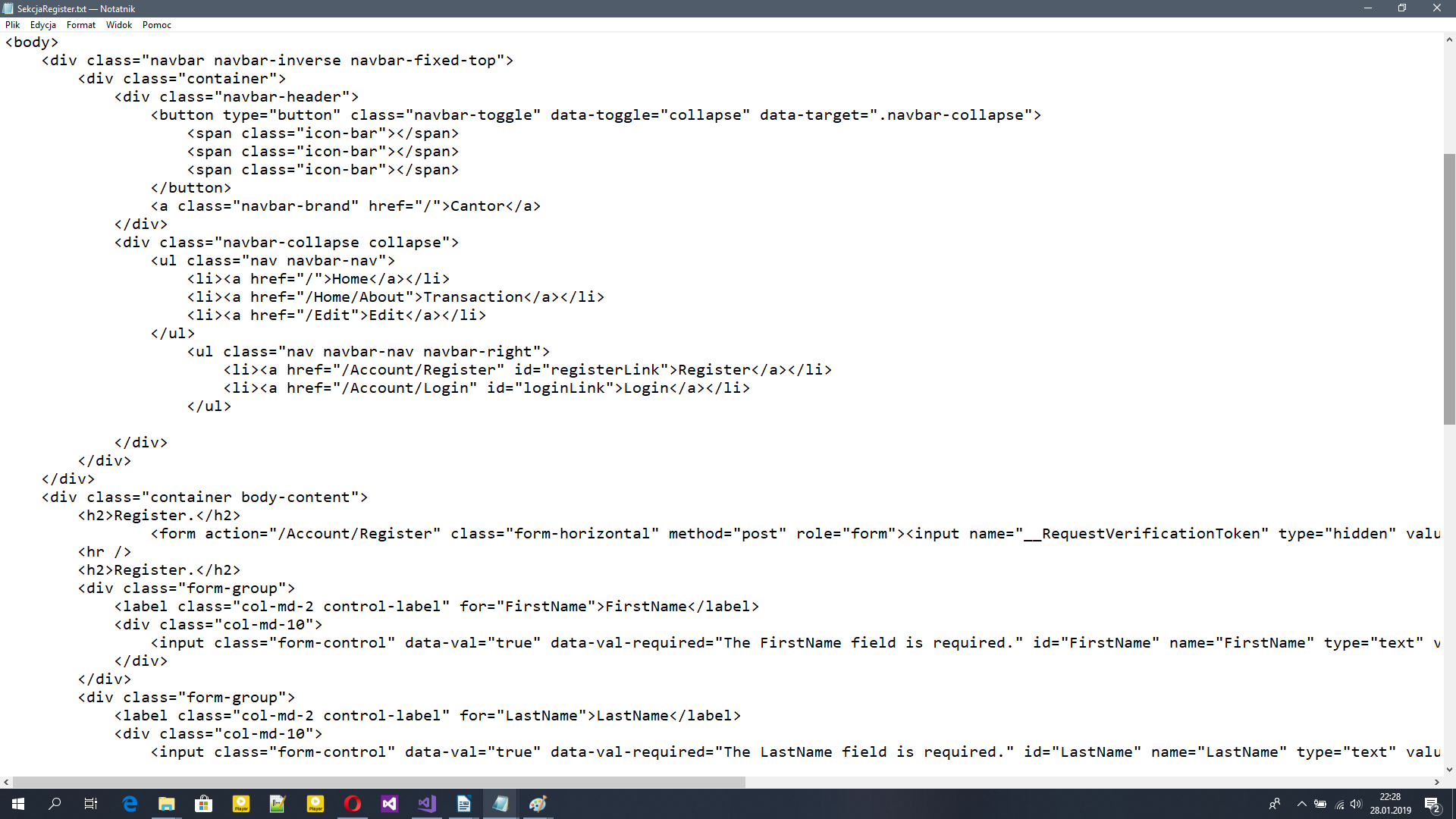
**Details**



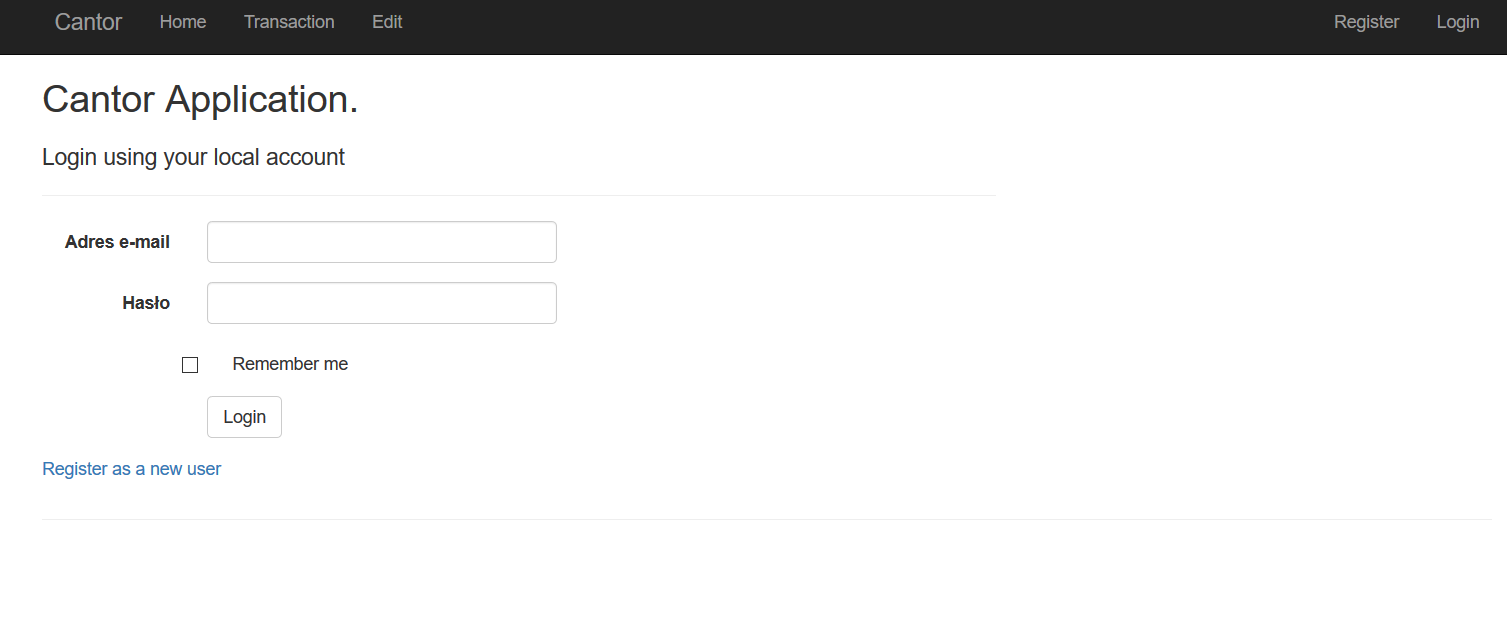
****

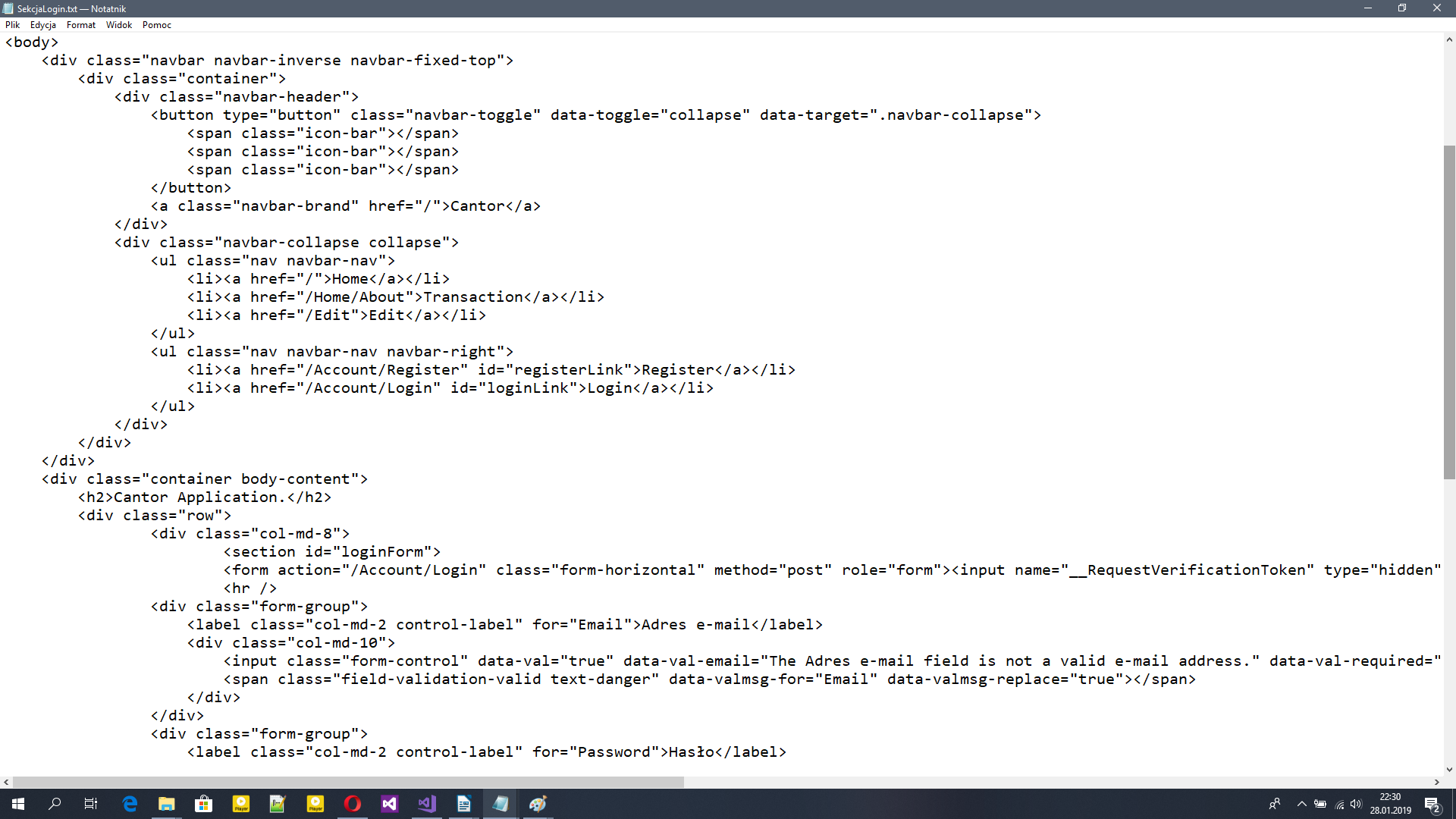
**Rejestracja**



****

**Logowanie**



W pełni sprawna aplikacja znajduje się pod adresem :

http://msaj.hostingasp.pl