**DUMP Drive**

Za ovotjedni domaći je potrebno izraditi DUMP Drive konzolnu aplikaciju gdje ćete moći spremati svoje datoteke i djeliti ih s drugim članovima. Za aplikaciju potrebno je izraditi sljedeće feature:

-autentifikacija

-privatne mape i datoteke

-dijeljene mape i datoteke

FUNKCIONALNI ZAHTJEVI

Flow korištenja aplikacije bi trebao biti sljedeći:

1. Pokretanjem se otvori početni menu sa opcijama logina postojećeg korisnika ili registracije novog

* Odabirom logina se od korisnika traži prvo mail, a zatim lozinka. Ukoliko je netočna kombinacija maila i lozinke unesena, korisnika se vraća na login proces ali uz timeout login procesa za 30 sekundi kako bi bili sigurni da nije bot.
* Ukoliko odabere registraciju, od korisnika se očekuje unos maila, nove i potvrdne lozinke koje moraju biti identične te kao treći korak generira se i ispiše random string (sastavljena od barem jednog slova i jedne brojke) koja služi kao captcha da se ne registrira bot. Korisnik mora ponoviti unos ispisane generirane riječi.
  + Potrebno je validirat unos adrese tako da mora biti osiguran sljedeći format:
    - [string min 1 chara]@[string min 2 chara].[string min 3 chara]
  + Potrebno je validirati da ne postoji prethodno korisnik sa odabranom adresom (svaka adresa korisnika je unikatna)

2.Ulaskom u aplikaciju nudi se izbornik korisniku koji prikazuje sljedece opcije:

* Moj disk
* Dijeljeno sa mnom
* Postavke profila
* Odjava iz profila

**Moj disk**

Ulaskom u moj disk biti će ispisani svi folderi i fileovi koji se nalaze u tvom disku sortirani na način da se prvo ispišu mape pa se datoteke ispišu po datumu kad su zadnji put uređene.

Format bi trebao biti ovakav:

* + mape sortirane po nazivu
  + datoteke sortirane po datumu lastChanged

Za sve akcije manipuliranja mapama, datotekama itd ćete koristiti komande.

**Komande:**

Komande koje će se trebati implementirati su:

* + help - za ispis svih komandi
  + stvori mapu ‘ime mape’ – za stvaranje mape na trenutnoj lokaciji
  + stvori datoteku ‘ime datoteke’ – za stvaranje datoteke na trenutnoj lokaciji
  + uđi u mapu ‘ime mape’ – za ulazak u mapu
  + uredi datoteku ‘ime datoteke’
  + izbriši mapu/datoteku ‘ime mape/datoteke’
  + promjeni naziv mape/datoteke ‘ime mape/datoteke’ u ‘novo ime mape/datoteke’
  + povratak

Za stvaranje ,brisanje te preimenovanje mapa/datoteka treba biti dialog potvrde.

**Uređivanje datoteka**

Kad se uđe u datoteku piše se tekst koji se želi pohranit na iduci nacin:

* + upisuje se tekst koji treba predstavljat jedan red
  + za novi red treba pritisnuti enter te se prelazi u novi
  + za brisanje te prelazak na prošli red treba se na prazan unos pritisnuti backspace

Za uredivanje datoteke takoder moraju postojati iduce komande:

* + help – ispisuju se sve komande za uređivanje datoteke
  + spremanje i izlaz
  + izlaz bez spremanja

Kad se odabere opcija spremanje i izlaz uz promjenu teksta pohranjuje se podatak da se datoteka spremila u tom trenu.

Za razliku od komandi navigacije kroz mape, komande za uređivanje datoteka moraju počinjati sa posebnim znakom (:) da program može prepoznati da se ne unosi novi red

**Dijeljene mape i datoteke**

Za dijeljenje mape ili datoteke postoji zaseba komanda u navigaciji kroz mape:

* + podjeli mapu/datoteku s ‘email’- ova komanda shera mapu/datoteku s korisnikom koji posjeduje uneseni mail
  + prestani dijeliti mapu/datoteku s ‘email’

Kad se podjeli mapa/datoteka s korisnikom taj korisnik moze vidjeti/uređivati dijeljene stavke u početnom izborniku ‘dijeljene mape i datoteke’

**Uređivanje zajedničkih datoteka**

Uređivanje dijeljenih datoteka je ista kao i kod privatnih samo ima iduče stavke:

* + korisnik koji nije vlasnik dadoteke može izbrisati datoteku iz vlastite shared mape ali se ta datoteka ne briše na lokaciji vlasnika
  + oba korisnika mogu uređivati datoteku i ta se promjena pohranjuje na obe lokacije
  + oba korisnika mogu dodavati komentare u dokumentu

**Rad s komentarima:**

Korisnik manipulira s komentarima na idući način:

Kad se uđe u dokument mora se upisati komada:

* otvori komentare- gdje se ispisuju svi komentari u format id-email-datum te ispod pise sadrzaj komentara

U prikazu komentara upisuju se komande:

* dodaj komentar
* uredi komentar
* izbriši komentar

**Postavke profila**

Ulaskom u postavke profila korisniku se nude opcije da promjeni šifru ili email. Za promjene se trebaju izvršiti iste validacije kao za registraciju.

**Odjava iz profila**

Deautenticira korisnika i vraća ga na početni autentikacijski menu.

**Tehnički zahtjevi**

Potrebno je razložiti implementaciju aplikacije na 3-tier arhitekturu, pri tom koristeći upute i savjete dane na prethodna dva predavanja te u demo projektu EF-a.

Nužno je da svi podatci budu pohranjeni u vidu baze podataka.

Potrebno je setupirati dotnet projekt te ubaciti pakete pokazane na prethodnom predavanju u odgovarajuće projekte te isto tako prilikom implementacije koristiti migracije kako bi se model baze podataka mogao replicirati na environmentu osobe koja pokreće projekt lokalno (reviewer domaćeg).

Naglasak je na objektno orijentiranom dizajnu koda pa tako treba dobro promisliti o implementaciji hijerarhije određenih entiteta i parent child relacija.

Potrebno je napraviti detaljan seed podataka, koji će pritom biti automatski populiran u bazu podataka prilikom buildanja projekta (pokazano na predavanju).

Svi upiti i filtrirani ispisi se moraju implementirati uz pomoć method sintakse **LINQ-a.**

**BONUS ZADATAK**

Za navigaciju kroz fileove implementirajte da se na strelice odabiru mape odabiru opcije te se na enter ulazi u mapu/ureduje datoteka.

Za ulazak u navigacijski mode treba upisati komandu: navigacija

Za izlazak iz navigacijskog moda treba pritisnuti escape.

**NAPOMENA**

Za implementaciju mape i datoteke koristite objekte koji bi ih trebali “predstavljati” nemojte koristit prave mape i datoteke.

Korisni linkovi

[EF docs](https://learn.microsoft.com/en-us/ef/core/)

[EF overview](https://www.learnentityframeworkcore.com/)

[Modeling](https://learn.microsoft.com/en-us/ef/core/modeling/)

[Model configuration](https://www.entityframeworktutorial.net/efcore/fluent-api-in-entity-framework-core.aspx)

[Queries](https://learn.microsoft.com/en-us/ef/core/querying/)

[Data management](https://learn.microsoft.com/en-us/ef/core/saving/basic)

[OO inheritance](https://learn.microsoft.com/en-us/ef/core/modeling/inheritance)

[Migrations](https://learn.microsoft.com/en-us/ef/core/managing-schemas/migrations/?tabs=dotnet-core-cli)

[Postgres connection string examples](https://www.connectionstrings.com/postgresql/)

Rok: **30. prosinca 2024 EOD.,**

Nema produživanja roka tako da krenite na vrime!

Repozitorij treba imenovati na sljedeći način: **Internship-7-Drive**

Demo projekt: <https://github.com/dujesa/todo-app>