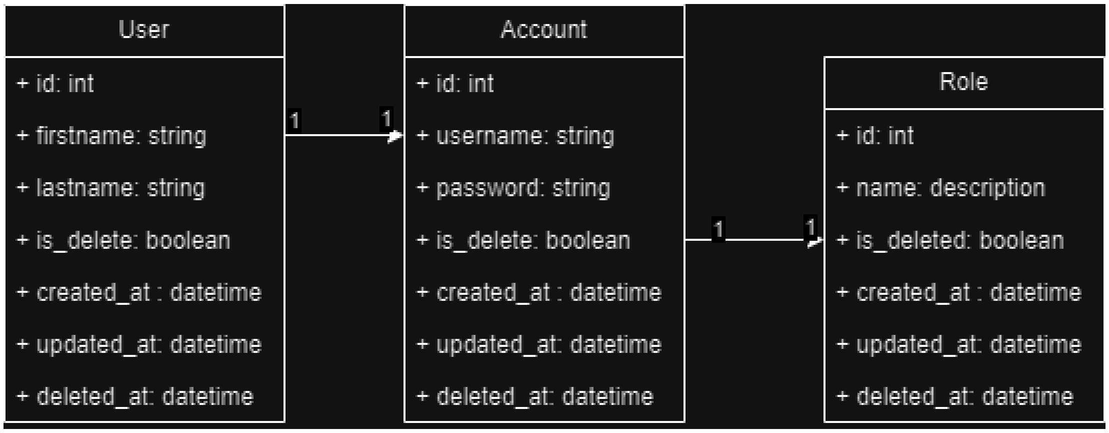
1. **Web Application:** Users Manager
2. **Description:** The main idea of this web application is managing users as an application Administrator, so the application was developed using clean architecture principles basically using DDD and TDD,
3. **User Stories:**
   - As a site administrator; I can create, read, update, delete roles, so I can add a role to new users.
   - As a site administrator; I can create, read, update, delete accounts associated to users, so I can change account information of users.
   - As a site administrator; I can create, read, update, delete users; so I can handle all operations on users
   - As a site administrator; I can login into web application; so only authorized administrators can manage users
4. **Business Rules**
   - Each user can have only one account to login into system
   - Each account can have only one role, multiple roles are not permitted
   - The main role needs to be called as Administrator
   - Each account will be identified by a unique email username
   - Each record will be deleted using logic elimination
   - Each record will have create, update and delete dates

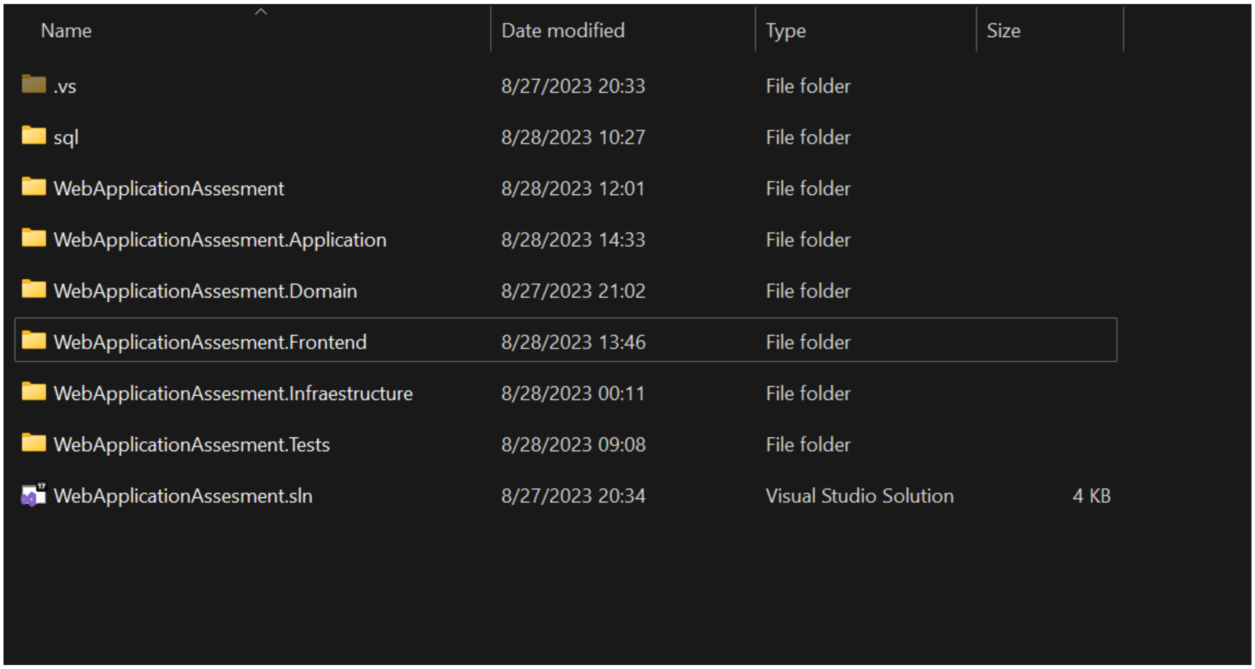
5. **Database Model**
   In a big picture the db model is represented using er diagram, where the main domain is User Management, this domain has 3 entities: User, Account and Role.
   One user has one account, so the relationship with account is 1 to 1.
   One account has one role, so the relationship with role is 1 to 1.
   Each entity has common columns like: created_at, updated_at, deleted_at, and is_deleted



6. **Technologies**
   To develop this application the following tech stack was used:
   - Programming Language: C#
   - Database: Postgresql
   - Frameworks: .NET 7, MVC, React

7. How to run application

- Restore database with sql file called db_backup.sql located in sql directory. To restore the db we can use the command: psql -U username -f "path/to/sql/file"
- In WebApi project called WebApplicationAssesment change your appsettings.json file to set user and password for application
- The application has a frontend client to consume the created api, the client was developet with vite and react, so to start the application run the commands: npm install and npm run dev, inside of the directory WebApplicationAssesment.Frontend



Name	Date modified	Type	Size
.vs	8/27/2023 20:33	File folder	
sql	8/28/2023 10:27	File folder	
WebApplicationAssesment	8/28/2023 12:01	File folder	
WebApplicationAssesment.Application	8/28/2023 14:33	File folder	
WebApplicationAssesment.Domain	8/27/2023 21:02	File folder	
WebApplicationAssesment.Frontend	8/28/2023 13:46	File folder	
WebApplicationAssesment.Infraestructure	8/28/2023 00:11	File folder	
WebApplicationAssesment.Tests	8/28/2023 09:08	File folder	
WebApplicationAssesment.sln	8/27/2023 20:34	Visual Studio Solution	4 KB

8. Public repository

The code implementation was uploaded to github:
https://github.com/pinzon1992/WebApplicationAssesment
If there is any problem in the compressed file you can check this repository