Documenting the existing application from MDS

1. The list of initial project requirements

Completed

1. As a user, I want to be able to post pictures on my profile.
2. As a user, I want to be able to visit other people's profiles.
3. As a user, I must have a login form in order to enter my account.
4. As a user, I must have a signup form in order to create an account.
5. As a user, I want to be able to like posts and comments.
6. As a user, I want to be able to leave comments on posts.
7. As a user, I want to be able to follow other people and get followed by other people.
8. As a user, I want to see posts from people I follow in my main feed.
9. As a user, I want to have a profile picture, besides my posts.
10. As a user, I want to be able to expand replies to existing comments and reply to them.

Unachieved

1. As a business-owner, I want to be able to see how many people viewed my profile over time.  
    - This feature didn’t play an important role in the user experience.  
    - We will implement this feature.  
    - Better time management and planning.
2. As a user, I want to be able to make serverless video calls.  
    - We decided this would require too much work but it wouldn’t bring that much value to the app.  
    - We will not implement this feature.  
    - Better time management and planning.
3. As a user, I want to get content recommendations based on posts I liked previously.  
    - We didn’t find a feasible solution in the allocated time.  
    - We will not implement this feature.  
    - Better time management and planning.
4. As a user, I want to get notifications.  
    - We didn’t find a feasible solution in the allocated time.  
    - We will implement this feature.  
    - Better time management and planning.
5. Team description
6. Buzas Radu-Gabriel - Devops engineer; responsibilities: deployment and networking, CI/CD, server maintenance
7. Eftimie Petre-Laurentiu - Backend engineer; responsibilities: database management, REST api design, unit tests
8. Moarcas Cosmin - Full stack engineer; responsibilities: backend development, frontend development, bug fixing
9. Popa Stefan - Frontend engineer; responsibilities: frontend setup & development, UI design
10. Stoica Ioan - Frontend engineer; responsibilities: frontend development; he is no longer part of the team
11. Software Architecture Report

Explain what are the architectural decisions that you have made in the previous iteration, and comment on the effectiveness of those decisions. You are expected to use the diagrams that you previously created in order to explain these decisions.

* 1. What are the technologies that you have used, and why?  
       
     We used NodeJS so that the whole application would be written with the same programming language. We chose Express as the main backend framework for its popularity. For the session management we used Redis because it is stored in the RAM memory so it is faster than other options.  
       
     On the frontend side we used Angular because we were familiar with this framework.  
       
     During the deployment process we used Nginx and Github Actions for their ability to provide a fast and reliable solution.
  2. What are the architectural patterns you have implemented and were they appropriate, and why?
     1. We have used REST API pattern for its flexibility on data management
     2. Layered pattern for reduced dependency because the function of each layer is separate from the other layers and for easy testing
     3. Client server pattern which is common for web apps because users are the ones usually requesting the resources from the server
     4. Event driven architecture to ensure loose coupling of components

* 1. Were the coding principles established enforced successfully? (For example if you included linting in the application, did you respect it? Did you try to implement coding standards and practices?  
       
     We enforced some coding practices such as: naming conventions, removing unused imports, SOLID principles, and design patterns.
  2. What are the faults that were discovered during development that haven’t been addressed by the time of delivery?
* content is loading slowly in the main feed, leading to bad UX
* sometimes the user is redirected unnecessarily to login page
* errors in loading profile data when switching pages
* unauthenticated users can navigate to a post/profile by URL (we will either restrict this behavior or change the UI/UX for guests)
  1. Does any part of the project require refactoring?  
       
     There is no need for heavy refactoring, as we tried to focus on the overall quality of the app, so we tried to enforce best practices whenever possible. We kept our focus on maintainability, however there is always room for improvement.

Include the link to the MDS repository. [Backend repo](https://github.com/lowLevelGod/proiectMDS) [Frontend repo](https://github.com/popastefan10/proiectMDSAngular)