***Implementation view***

Картина, която съдържа текст, диаграма, План, схематичен

Описанието е генерирано автоматично

1. Сървърна инфраструктура:

Backend сървъри, хостващи C# ASP.NET Web API с EntityFramework Core.

Сървъри за бази данни за съхраняване на потребителски данни, отчети за издаване и друга информация за приложението.

Файлови сървъри за съхраняване на изображения и други медии, качени от потребителите.

1. Работни станции и мобилни устройства на потребителите:

Работни станции и преносими компютри, използвани от потребителите на приложението за достъп до приложението CityVox.

Мобилни устройства (смартфони и таблети), използвани от потребителите за достъп до приложението CityVox.

Тези устройства използват уеб браузъри или специално създадено мобилно приложение, за да взаимодействат с фронтенда на CityVox.

1. Местна мрежа и интернет комуникация:

Комуникацията между работните станции на потребителите, мобилните устройства и бекенд сървърите се осъществява по интернет.

Инфраструктурата на локалната мрежа може да се използва в рамките на организацията за целите на вътрешната комуникация и управлението.

Комуникацията между frontend и backend се осъществява чрез HTTP и HTTPS протоколи, за да се осигури сигурно прехвърляне на данни.

1. Комуникация с бази данни:

C# ASP.NET Web API комуникира със сървърите на базата данни, като използва EntityFramework Core, който осигурява обектно-релационно съпоставяне (ORM) за взаимодействие с основната база данни.

API обработва заявки за удостоверяване на потребителя, доклади за издаване, данни от карти, публикации в социалните мрежи и други дейности, свързани с потребителя.

1. Прехвърляне и съхранение на данни:

Прехвърлянето на данни между frontend и backend се улеснява чрез API повиквания с помощта на axios.

Данните се съхраняват и извличат от базата данни с помощта на EntityFramework Core, която управлява взаимодействието с ентитетите на базата данни.

1. Мерки за сигурност:

Удостоверяването и оторизацията на потребителите се обработват с помощта на JWT токени, съхранявани в локално хранилище, и бисквитки, съдържащи само HTTPS RefreshTokens.

Протоколът HTTPS осигурява криптирана комуникация между фронтенда и бекенда, като предпазва чувствителните данни по време на предаването им.

В бекенда са приложени допълнителни мерки за сигурност, като например валидиране на данните, за да се предотврати неоторизиран достъп и нарушаване на сигурността на данните.

1. Балансиране на натоварването и мащабируемост:

Могат да бъдат приложени техники за балансиране на натоварването, за да се разпредели входящият трафик на приложенията между няколко бекенд сървъра, като се гарантира оптимална производителност и наличност.

Архитектурата на бекенда на приложението е проектирана да се мащабира хоризонтално, което позволява на системата да се справя ефективно с увеличения трафик на потребителите и изискванията за обработка на данни.