

Документација за платформа за трансфер на фајлови – WeShare

Изработил:

Бојан Симичиев 191215

Игор Даниловски 191273

Содржина

Вовед	3
Цели	3
Начин на функционирање	3
Не-најавени корисници	3
Најавени корисници	4
Функционални и Нефункционални барања	4
Функционални барања	4
Нефункционални барања	5
Техничка спецификација	5
Архитектура	5
1. Кориснички интерфејс	5
2. Апликациски слој (Бизнис логика)	5
3. База на податоци	7
4. Место за чување фајлови (File Storage)	7
5. Автентикација и Авторизација	8

Вовед

Овој документ претставува документација за платформата за трансфер на фајлови WeShare и во него ќе бидат опишани целите, функционалностите, карактеристиките на нашата платформа. Понатаму ќе биде обезбеден и технички опис на начинот на функционирање на одредени аспекти и функционалности на платформата.

Со ова, ќе биде претставена една целосна слика за работата и предностите на нашата платформа за трансфер на фајлови.

Цели

Целите на овој документ се:

1. Да даде целосно разбирање за можностите и функционалностите на WeShare.
2. Да даде слика за начинот на функционирање на платформата.
3. Да даде слика за различните функционалности помеѓу корисниците со и без профил, како и бенефициите кои корисниците со профил ги имаат врз оние без профил.
4. Да даде опис и техничка спецификација за тоа како функционираат одредени аспекти од платформата.

Начин на функционирање

WeShare е веб-базирана платформа која им овозможува на корисниците брз, безбеден и едноставен пренос на фајлови помеѓу еден или повеќе корисници. Оваа платформа работи на начин така што кога корисниците влегуваат на веб платформата им се прикажува екран на кој што ќе имаат можност да прикачат фајлови од различни екстензии но кои веќе се наоѓаат на нивниот компјутер. По успешното прикачување на фајловите, корисниците добиваат краток URL линк кој може да биде споделен со било кој, кој потоа, со пристап до линкот, ќе има можност да ги прегледа и преземе прикачените фајлови.

Покрај основните функционалности кои се достапни за сите корисници, WeShare подржува и опција со најавување на корисници. Во следната секција ќе дадеме мал приказ на функционалностите на 2-та типа на корисници и бенефициите кои корисниците би ги имале доколку се креираат профил и се најават на платформата.

Не-најавени корисници

- Прикачување на фајлови (еден или повеќе) **со вкупно ограничување** на големината на фајловите од **25MB**.
- Креирање на линк со прикачените фајлови кој би имал **рок на валидност од 7 дена**.
- Пристап до претходно испратен линк и преглед и преземање на фајловите кои се наоѓаат на него.

Најавени корисници

- Прикачување на фајлови (еден или повеќе) со **без органичување** на големината на фајловите.
- Креирање на линк со прикачените фајлови кој со **рок на валидност** по **желба на корисникот**.
- Креирање на линк со прикачени фајлови **без рок на валидност** (валиден е **засекогаш**)
- Пристап до претходно испратен линк и преглед и преземање на фајловите кои се наоѓаат на него.
- Пристап до Dashboard на кој ќе бидат прикажани сите линкови кои тие ги креирале со детали за истите и можност за бришење односно инвалидирање на истите.

Една од значајните предности на WeShare е дека сите прикачени фајлови се сместуваат на серверите на Amazon Web Services (AWS) S3. Ова осигурува висок степен на безбедност и достапност на прикачените фајлови.

Функционални и Нефункционални барања

Функционални барања

- Регистрација и најава на корисници
 - Корисниците да имаат можност да креираат профил со валидна е-маил адреса и лозинка.
- Прикачување на фајлови
 - Корисниците да имаат можност да прикачат фајлови со различна екстензија.
 - Корисниците да имаат можност да прикачат повеќе фајлови одеднаш.
- Креирање на линк со датум на валидност (**за најавени корисници**)
 - Најавените корисници мора да имаат подесување со кое што ќе може да менаџираат со датумот на валидност на еден креиран линк.
- Споделување фајлови
 - Корисниците мора да имаат можност да креираат линк со нивните прикачени фајлови и да го споделат истиот.
- Пристап до веќе прикачени фајлови
 - Корисниците мора да имаат пристап до фајловите кои се прикачени во рамки на еден линк (доколку истиот е валиден).
- Преземање на прикачени фајлови
 - Покрај пристапот корисниците мора да бидат во можност да ги преземат фајловите кои ги нуди линкот до кој што пристапиле.

- Admin Dashboard (за најавени корисници)
 - Најавените корисници мора да имаат пристап до линковите коишто досега ги креирале како и до информации за истите.
 - Покрај пристап тие мора да имаат и можности за менаџирање со истите.

Нефункционални барања

- Безбедност
 - Забранет пристап до линкови кои се “disabled” или “invalid”.
- Перформанси
 - Платформата треба да подржува брзо прикачување и преземање на фајловите.
- Компатибилност со разните веб прелистувачи
 - Платформата мора да биде достапна од сите веб прелистувачи.
- Сигурност
 - Интегритет на податоците и спречување на загуба на информациите.

Техничка спецификација

Архитектура

Целосната архитектура на платформата се состои од следните компоненти:

1. Кориснички интерфејс

Корисничкиот интерфесј е местото каде што корисниците имаат интеракција со платформата. Оваа компонента им ги овозможува на корисниците сите функционалности кои ги подржува платформата односно им овозможува брзо и безбедно прикачување, преглед и преземање на фајлови меѓу корисници.

2. Апликациски слој (Бизнис логика)

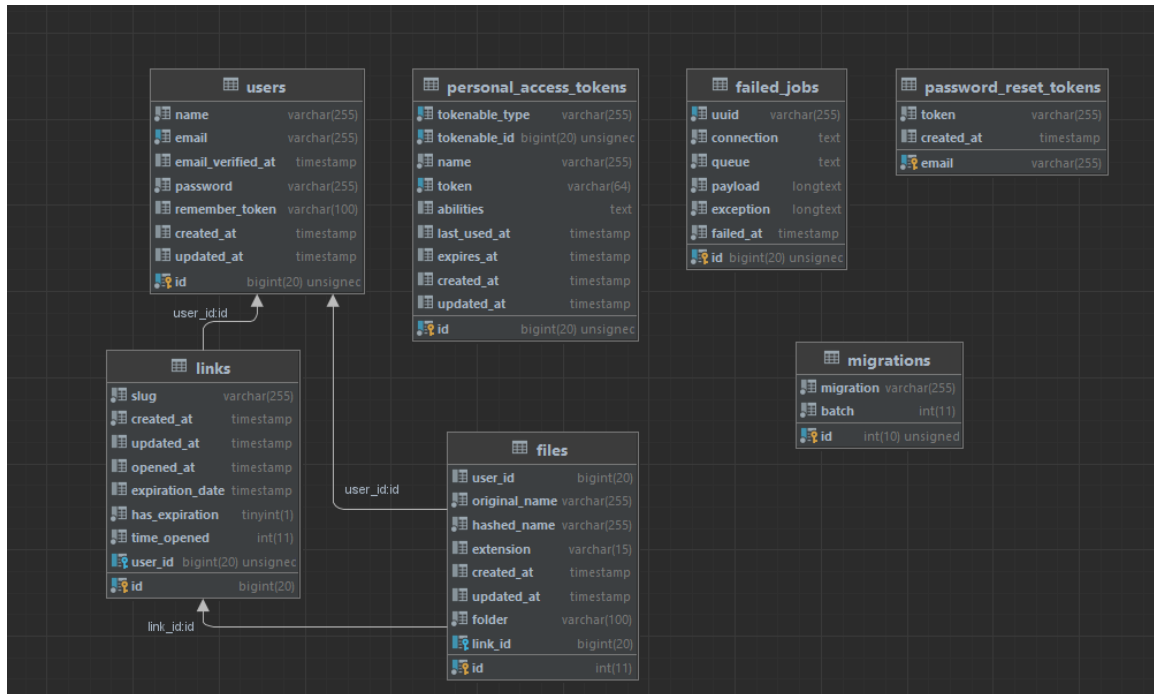
Овој слој ја содржи целата бизнис логика на платформата и се грижи за автентикацијата на корисниците, прикачувањето на фајловите, нивно прегледување и преземање и слично. За прикачувањето и преземањето на фајловите овој слој е интегриран со серверите за складирање на Amazon Web Services односно со Amazon Simple Storage Service (AWS S3) кои овозможуваат складирање и преземање на податоци како што се документи, слики, видеа и фајлови со останати екстензии.

Што се однесува до позадината (backend) на платформата, таа се состои од:

- Контролери (Controllers)
 - Каде што се поврзани крајните точки (endpoints) со погледите кои соодветствуваат во нив. За процесирање на информациите кои доаѓаат и излегуваат односно се препраќаат од страна на контролерите се користат сервисите кои манипулираат со наведените информации.
 - Такви контролери на платформата се:
 - ProfileController
 - LinkController
 - FileUploadController
 - AdminDashboardController
- Сервиси (Services)
 - Дел кој што е задолжен за бизнис логиката односно менаџирањето со информациите кои се проследени од страна на контролерите и користење на репозиториумите за манипулирање со базата на податоци.
 - Такви на платформата се:
 - LinkService
 - FileUploadService
- Репозиториуми (Repositories)
 - Репозиториумите слушат за комуникација помеѓу апликацијата и базата со податоци и во нив се наоѓаат строго CRUD операции
 - Такви се:
 - LinkModelRepository
 - FileModelRepository
- Погледи (Views)
 - Погледите го сочинуваат корисничкиот интерфејс на платформата
- Модели
 - Моделите претставуваат приказ на објектите односно ентитетите кои се запишуваат во база но може и да претставуваат приказ на објекти кои треба да му се прикажат на корисникот или пак да се проследат низ некој погорен слој од архитектурата (DTOs)
 - Пример за модели на платформата:
 - User
 - LinkModel
 - FileUploadDTO
 - FileName
 - FileModel

3. База на податоци

Базата на податоци служи за чување на информациите кои пристигнуваат на платформата односно во неа се чуваат линковите, корисниците, врските кон фајловите и сл. Во рамки на овој проект се користи MySQL. Во прилог се наоѓа релациониот дијаграм од структурата на базата на податоци:



4. Место за чување фајлови (File Storage)

Местото за чување односно складирање на фајловите е Amazon Simple Storage Service (AWS S3) која всушност претставува високо скалабилна услуга во облак за складирање. Работи на принцип на кофички (buckets) кои служат како контејнери за складирање на податочните објекти во AWS облакот.

Клучни карактеристики на AWS S3 се:

- Скалабилност и Издржливост
 - Високо скалабилни и издржливи кофички
- Разновидност на типови на податоци
 - Широк спектар на податоци кои можат да се складираат (текст, слики, видеа, back-ups и сл.)
- Контрола на пристап
 - Механизми за контрола на пристап до складираните податоци
- Енкрипција на податоци
 - Серверска и клиентска енкрипција

5. Автентикација и Авторизација

Конкретно се користи JWT механизам за автентикација на корисниците