

# Raport de Proiect: Mailharvest

Sviridov Andrei - `andrei.sviridov00@e-uvv.ro`

Scofert, Andrei - `andrei.scoferta00@e-uvv.ro`

Schiopu Raul Andrei - `raul.schiopu02@e-uvv.ro`

**Team leader:**

S, Chiopu Raul Andrei - `raul.schiopu02@e-uvv.ro`

**Coordonator:**

Nokia - Gabriel Pistolescu - `gabriel.pistolescu@nokia.com`

June 4, 2024

## MAILHARVEST

**Nokia**

Bulevardul Republicii 21

## Contents

<b>1</b>	<b>Arhitectura Programului</b>	<b>3</b>
1.1	Introducere . . . . .	3
1.2	Frontend (React) . . . . .	3
1.3	Backend (Python) . . . . .	3
1.4	Mediul de Dezvoltare . . . . .	3
1.5	Modele Arhitecturale . . . . .	3
1.6	Principii Generale de Funcționare . . . . .	3
1.7	Detalii despre Sistemul de Operare . . . . .	4
1.8	UseCase . . . . .	4
<b>2</b>	<b>Descrierea Componentelor (Modulelor/Funcționalităților)</b>	<b>4</b>
2.0.1	Modulul de Extracție și Filtrare a Emailurilor din Gmail . . . . .	4
2.0.2	Modulul de Vizualizare și Salvare a Statisticilor . . . . .	5
2.0.3	Modulul de Trimitere pe Gmail a Statisticilor . . . . .	5
2.0.4	Modulul de Salvare a Filtrelor . . . . .	6

<b>3</b>	<b>Descrierea comunicării între module</b>	<b>6</b>
3.0.1	Canalul de Comunicare între Modulul de Extracție și Filtrare a Emailurilor și Modulul de Vizualizare . . . . .	6
3.0.2	Canalul de Comunicare între Modulul de Salvare a Filtrelor și Modulul de Extracție și Filtrare a Emailurilor . . . . .	7
<b>4</b>	<b>Structuri de baze de date și fișiere</b>	<b>7</b>
4.1	Baza de date "mailharvest" (mediul de dezvoltare: XAMPP cu phpMyAdmin și MySQL) . . . . .	7
4.1.1	Tabelul "users" . . . . .	7
4.1.2	Tabelul "filters" . . . . .	7
4.2	Structura Fișierelor . . . . .	8
4.3	Schemă Generală a Bazei de Date . . . . .	8

# 1 Arhitectura Programului

## 1.1 Introducere

Această secțiune descrie arhitectura programului Mailharvest, care include componentele sale principale și modul în care acestea interacționează între ele.

## 1.2 Frontend (React)

Interfața utilizatorului este dezvoltată folosind React și alte biblioteci externe pentru a crea o experiență interactivă și receptivă. Se utilizează tehnici de routing pentru a gestiona navigarea între diferitele pagini și componente ale aplicației. Pentru interacțiunea cu utilizatorul și prezentarea datelor, se utilizează componente React precum form-uri, tabele și grafice.

## 1.3 Backend (Python)

Backend-ul este dezvoltat în Python pentru a gestiona logica de afaceri, autentificare și extragerea datelor. Am utilizat Flask ca framework pentru backend, oferind rute de API pentru comunicarea cu frontend-ul și pentru gestionarea cererilor de la utilizatori. Pentru extragerea emailurilor și filtrarea acestora, se folosesc biblioteci Python externe. Arhitectura generală a sistemului este orientată pe modelul client-server, în care frontend-ul (clientul) comunică cu backend-ul (serverul) prin intermediul cererilor HTTP. Serverul procesează aceste cereri și returnează răspunsuri corespunzătoare. Datele sunt stocate și gestionate într-o bază de date, MySQL.

## 1.4 Mediul de Dezvoltare

Pentru dezvoltarea și testarea aplicației, se utilizează un mediu de dezvoltare integrat (IDE) precum Visual Studio Code și PyCharm. Git este folosit pentru controlul versiunilor codului sursă, iar GitHub pentru gestionarea și colaborarea în echipă.

## 1.5 Modele Arhitecturale

Arhitectura programului/sistemului MailHarvest urmează modelul client-server, în care frontend-ul (clientul) comunică cu backend-ul (serverul) prin intermediul cererilor HTTP.

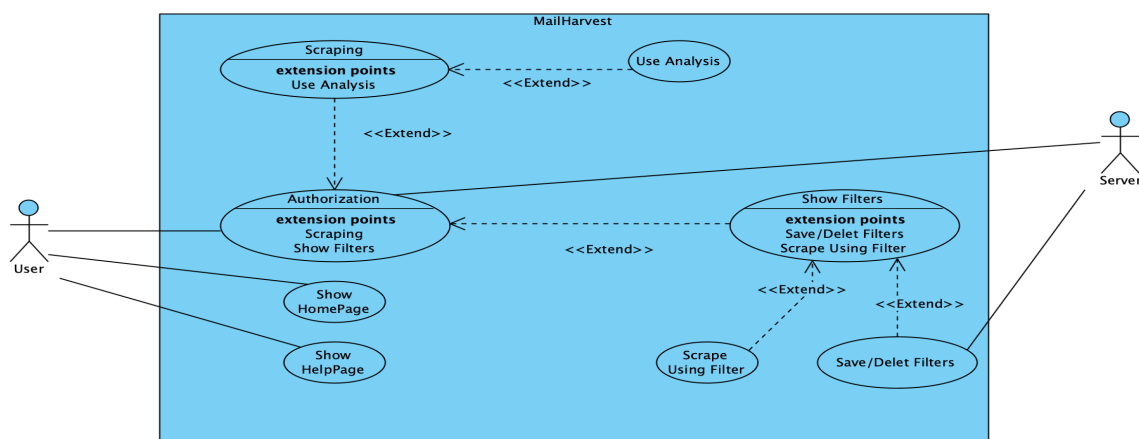
## 1.6 Principii Generale de Funcționare

Utilizatorii se înregistrează și se autentifică în aplicație cu un nume de utilizator și o parolă de aplicație (AppPassword) pentru a asigura securitatea accesului la date. Aplicația permite utilizatorilor să extragă emailuri din Gmail și să le filtreze conform criteriilor specificate. După filtrare, utilizatorii pot descărca emailurile într-un format CSV pentru analiza ulterioară. Utilizatorii pot salva și reutiliza filtrele utilizate pentru a eficientiza procesul de extragere și analiză a emailurilor.

## 1.7 Detalii despre Sistemul de Operare

Aplicația este dezvoltată pentru a fi independentă de sistemul de operare și poate fi rulată pe diverse platforme, inclusiv Windows, macOS și Linux. Backend-ul poate fi implementat și rulat pe un server care rulează orice sistem de operare compatibil cu Python, cum ar fi Xampp.

## 1.8 UseCase



## 2 Descrierea Componentelor (Modulelor/Funcționalităților)

### 2.0.1 Modulul de Extracție și Filtrare a Emailurilor din Gmail

**Descriere:** Acest modul se ocupă de extragerea emailurilor din contul de Gmail al utilizatorului și aplicarea de filtre pentru a selecta doar emailurile relevante.

#### Intrări:

- Credențialele de autentificare pentru contul de Gmail al utilizatorului
- Criteriile de filtrare pentru selecția emailurilor (ex: expeditor, subiect, dată etc.)

#### Prelucrări:

1. Autentificarea în contul de Gmail folosind credențialele furnizate
2. Extracția emailurilor din inbox-ul utilizatorului
3. Aplicarea criteriilor de filtrare pentru a selecta emailurile relevante

#### Ieșiri:

- Lista emailurilor extrase și filtrate conform criteriilor specificate

#### Interacțiuni cu alte Module:

- Interacționează cu Modulul de Salvare a Filtrelor de Căutare pentru a salva setările de filtrare reutilizabile

### 2.0.2 Modulul de Vizualizare și Salvare a Statisticilor

**Descriere:** Acest modul permite utilizatorului să vizualizeze statistici și analize pe baza mesajelor extrase și filtrate anterior și să le salveze pentru referințe ulterioare.

**Intrări:**

- Lista emailurilor extrase și filtrate anterior

**Prelucrări:**

1. Generarea diferitelor statistici și analize (ex: numărul total de emailuri, distribuția pe expeditor, analiza conținutului etc.)
2. Vizualizarea statisticilor sub formă de grafice sau tabele pentru o înțelegere mai bună
3. Salvarea statisticilor generate într-un format ușor de accesat și de distribuit (ex: PDF)

**Ieșiri:**

- Statistici vizuale sub formă de grafice sau tabele
- Fișier PDF conținând raportul statistic complet

**Interacțiuni cu alte Module:**

- Interacționează cu Modulul de Extracție și Filtrare a Emailurilor pentru a obține lista de emailuri necesară pentru generarea statisticilor

### 2.0.3 Modulul de Trimitere pe Gmail a Statisticilor

**Descriere:** Acest modul permite utilizatorului să trimită statisticile și analizele generate pe emailul său de Gmail pentru a le accesa și distribui mai ușor.

**Intrări:**

- Statisticile și analizele generate anterior
- Credențialele de autentificare pentru contul de Gmail al utilizatorului

**Prelucrări:**

1. Autentificarea în contul de Gmail folosind credențialele furnizate
2. Crearea unui mesaj de email care conține statisticile și analizele generate anterior
3. Trimiterea mesajului de email către adresa specificată de utilizator

**Ieșiri:**

- Confirmarea trimiterii cu succes a mesajului de email

**Interacțiuni cu alte Module:**

- Interacționează cu Modulul de Vizualizare și Salvare a Statisticilor pentru a obține statisticile și analizele necesare pentru trimitere

## 2.0.4 Modulul de Salvare a Filtrelor

**Descriere:** Acest modul permite utilizatorului să salveze filtrele de căutare utilizate în procesul de extragere și filtrare a emailurilor pentru a le reutiliza mai ușor în sesiuni de lucru ulterioare.

**Intrări:**

- Configurația filtrelor de căutare specificată de utilizator

**Prelucrări:**

1. Salvarea configurației filtrelor într-o formă persistentă (ex: bază de date, fișier de configurare etc.)
2. Asocierea fiecărui set de filtre salvate cu un nume sau o etichetă pentru o identificare mai ușoară

**Ieșiri:**

- Confirmarea salvării cu succes a filtrelor

**Interacțiuni cu alte Module:**

- Interacționează cu Modulul de Extracție și Filtrare a Emailurilor pentru a obține configurația filtrelor de căutare

## 3 Descrierea comunicării între module

### 3.0.1 Canalul de Comunicare între Modulul de Extracție și Filtrare a Emailurilor și Modulul de Vizualizare

**Modul Expeditor:** Modulul de Extracție și Filtrare a Emailurilor

**Modul Receptor:** Modulul de Vizualizare

**Mod de Comunicare:** Interfață API

**Parametri:**

- Lista de emailuri filtrate și pregătite pentru vizualizare

**Semnificație:** Acest canal de comunicare permite transferul eficient al rezultatelor filtrării emailurilor către modulul responsabil cu vizualizarea lor.

**Restricții:**

- Volumul de date transmis trebuie să fie gestionabil pentru a evita întârzierile în afișarea rezultatelor
- Datele transmise trebuie să fie protejate împotriva accesului neautorizat pentru a asigura confidențialitatea informațiilor utilizatorilor

### 3.0.2 Canalul de Comunicare între Modulul de Salvare a Filtrelor și Modulul de Extracție și Filtrare a Emailurilor

**Modul Expeditor:** Modulul de Salvare a Filtrelor

**Modul Receptor:** Modulul de Extracție și Filtrare a Emailurilor

**Mod de Comunicare:** Interfață API

**Parametri:**

- Configurația filtrelor de căutare salvate

**Semnificație:** Acest canal de comunicare permite transferul configurației filtrelor salvate către modulul responsabil cu extragerea și filtrarea emailurilor. Configurația filtrelor este transmisă pentru a fi utilizată în procesul de căutare a emailurilor conform preferințelor utilizatorului.

**Restricții:**

- Configurația filtrelor transmisă trebuie să fie validă pentru a asigura că rezultatele extragerii sunt conforme cu așteptările utilizatorului
- Accesul la configurația filtrelor trebuie să fie restricționat la utilizatorii autorizați pentru a preveni modificările neautorizate

## 4 Structuri de baze de date și fișiere

### 4.1 Baza de date "mailharvest" (mediul de dezvoltare: XAMPP cu phpMyAdmin și MySQL)

#### 4.1.1 Tabelul "users"

- **Numele Tabelului:** users
- **Atribute:**
  - id (*int(11)*): Identificator unic al utilizatorului
  - gmail (*varchar(255)*): Adresa de email a utilizatorului
  - password (*varchar(255)*): Parola utilizatorului, stocată în formă criptată
- **Restricții:**
  - id: Cheie primară, auto-incrementată
  - gmail: Unică, pentru a evita duplicarea adreselor de email
  - password: Criptată pentru a asigura securitatea datelor utilizatorilor

#### 4.1.2 Tabelul "filters"

- **Numele Tabelului:** filters
- **Atribute:**
  - id (*int(11)*): Identificator unic al filtrului
  - user\_id (*int(11)*): Cheia externă către utilizatorul asociat filtrului

- sender (*varchar(255)*): Expeditorul emailului
- words\_in\_body (*text*): Cuvintele cheie din corpul emailului
- from\_date (*date*): Data de la care sunt extrase emailurile
- to\_date (*date*): Data până la care sunt extrase emailurile

- **Restricții:**

- id: Cheie primară, auto-incrementată
- user\_id: Cheie externă care se referă la id-ul utilizatorului din tabela "users"
- words\_in\_body, sender: Permite valori nule pentru a permite filtrarea flexibilă

## 4.2 Structura Fișierelor

- **Numele Fișierelor:**

- Backend: Fișier principal al aplicației Flask
- main.py: File principal al aplicației Flask
- config.py: File de configurare pentru conexiunea la baza de date și alte setări

## 4.3 Schemă Generală a Bazei de Date

