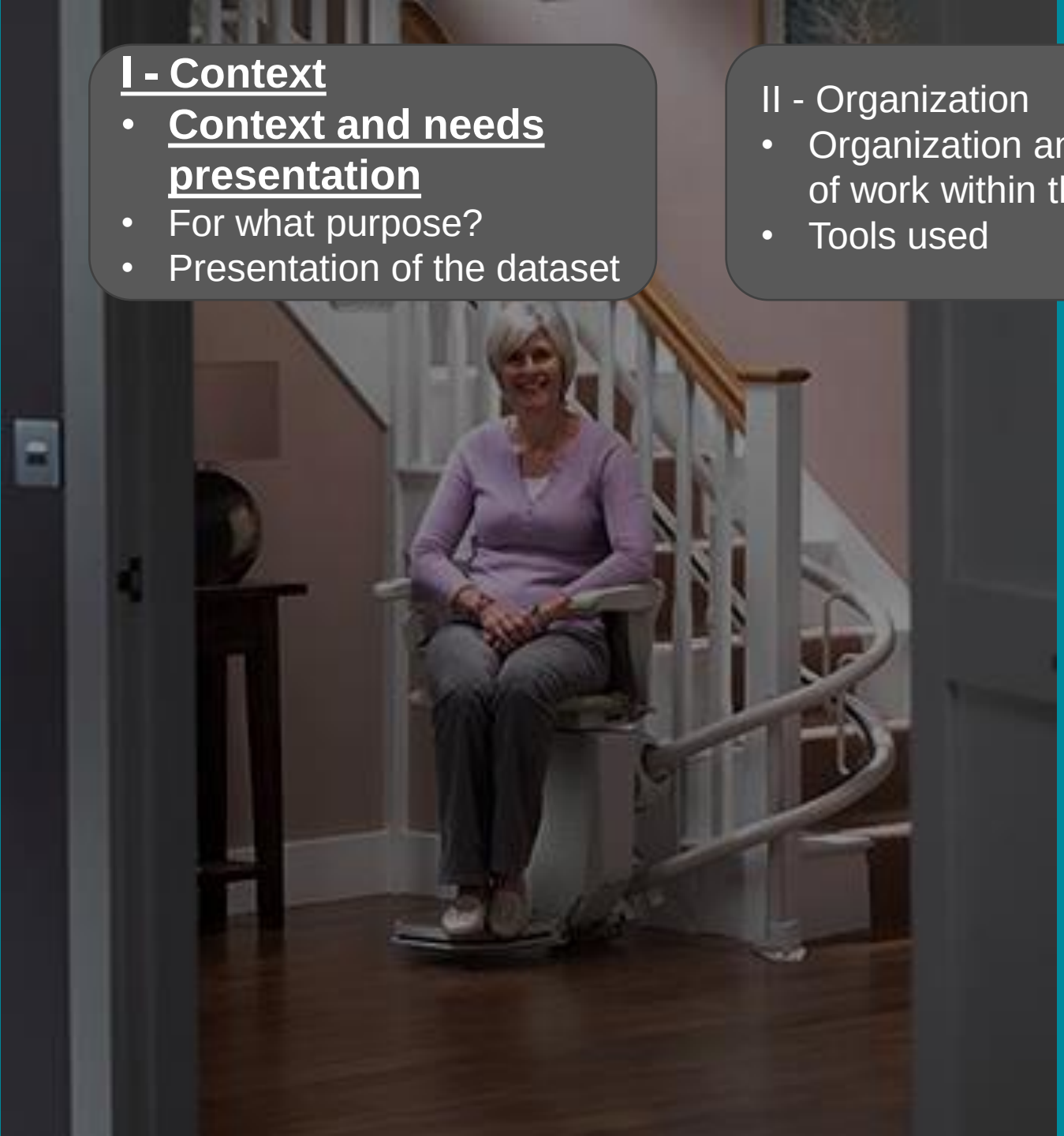
Data Analysis, Reporting, and Data Visualization

By Rose Chauvin, Clara Decros, Camille Carraco and Héléna Guertin

Mrs Belim, M. Bureau, M. Garnier and Mrs Dupuy

QUICK

- I - Context
- Context and needs presentation
- For what purpose?
- Presentation of the dataset
- II - Organization
- Organization and planning of work within the group
- Tools used
- III - Project
- Presentation interface
- Chart



I - Context

- **Context and needs presentation**
- For what purpose?
- Presentation of the dataset

II - Organization

- Organization and planning of work within the group
- Tools used

III - Project

- Presentation interface
- Chart

- **Public authorities**
- **Researchers and experts in domestic accident prevention**
- **Associations working on risk prevention**
- **Insurance companies**
- **The general public (parents, families, seniors)**

I - Context

- Context and needs presentation
- **For what purpose?**
- Presentation of the dataset

II - Organization

- Organization and planning of work within the group
- Tools used

III - Project

- Presentation interface
- Chart

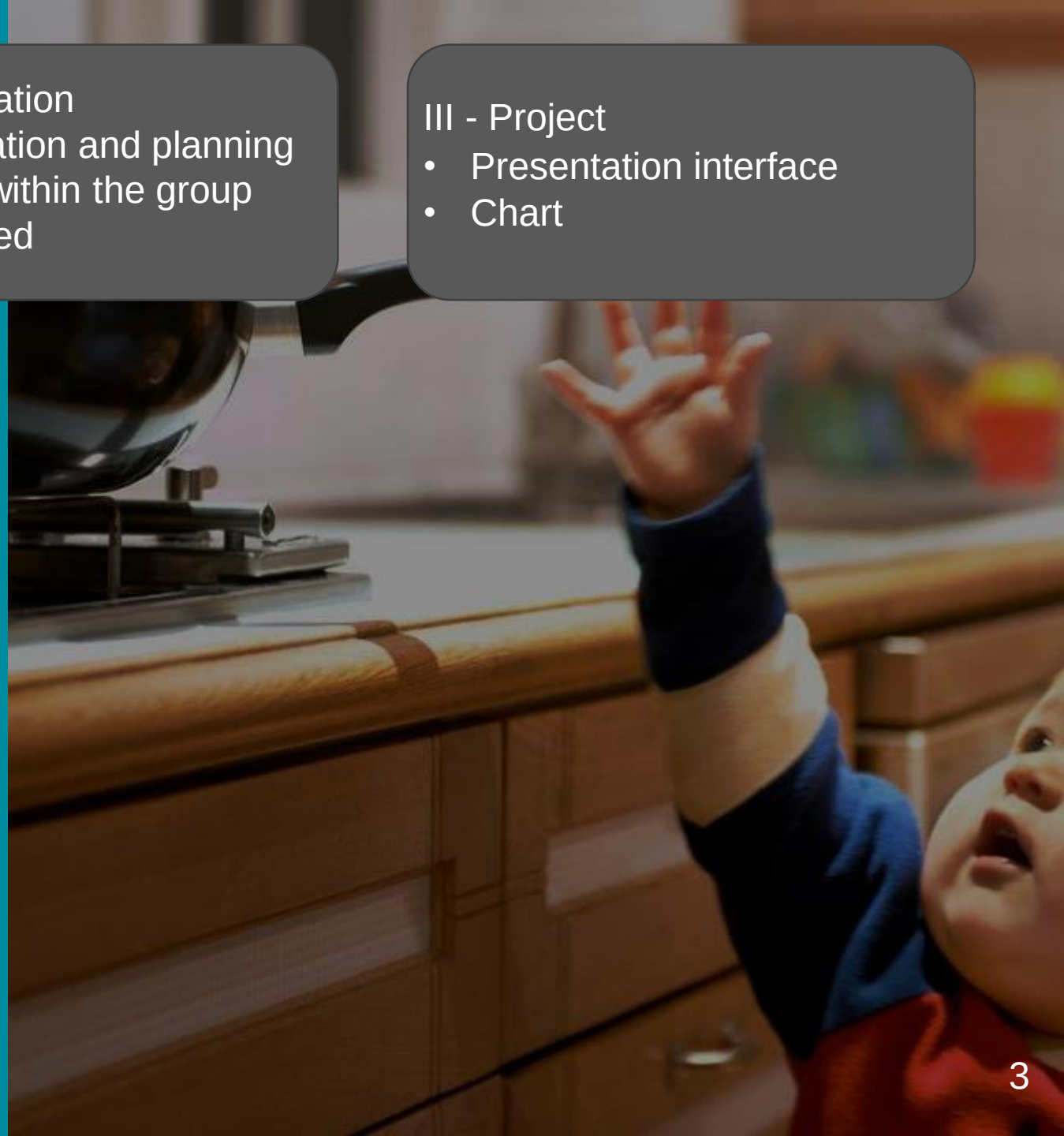
Risks of everyday life

Supports decisions

Inform and raise public awareness

Help with prevention

Create an automated tool



I - Context

- Context and needs presentation
- For what purpose?
- **Presentation of the dataset**

II - Organization

- Organization and planning of work within the group
- Tools used

III - Project

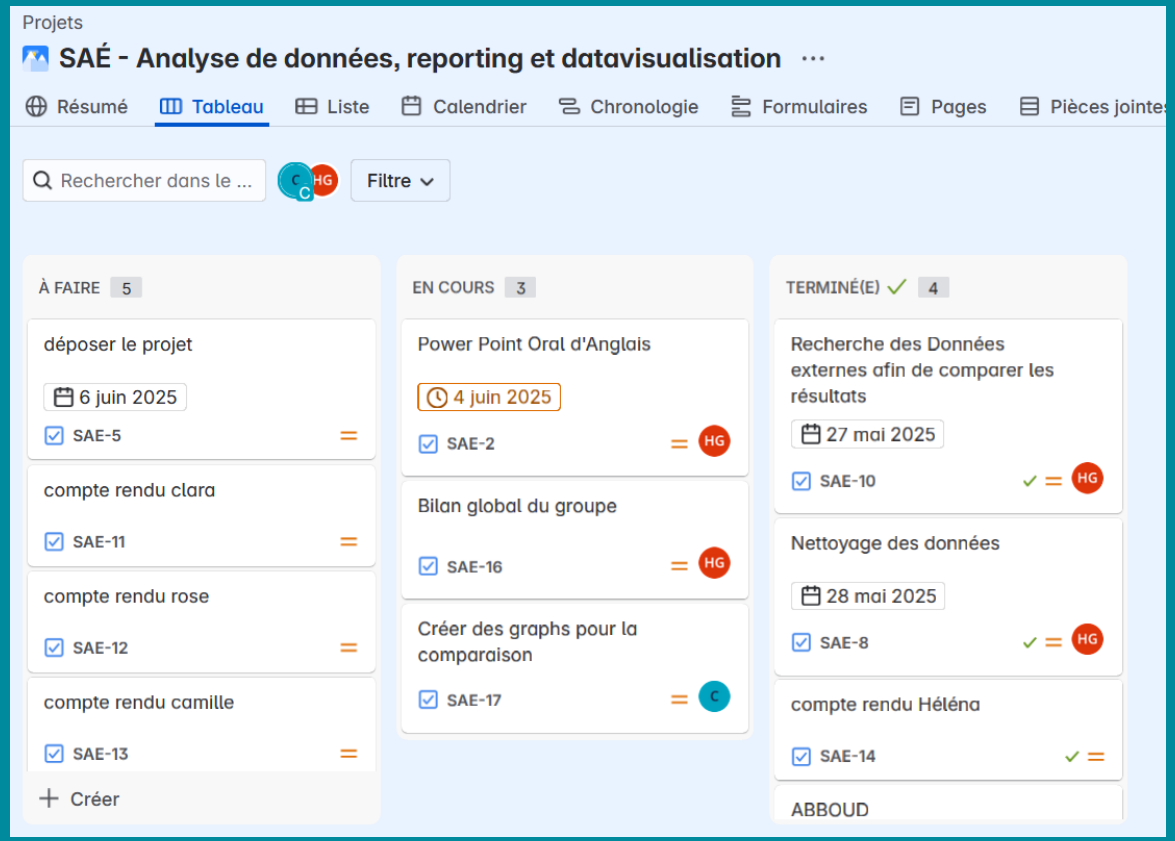
- Presentation interface
- Chart

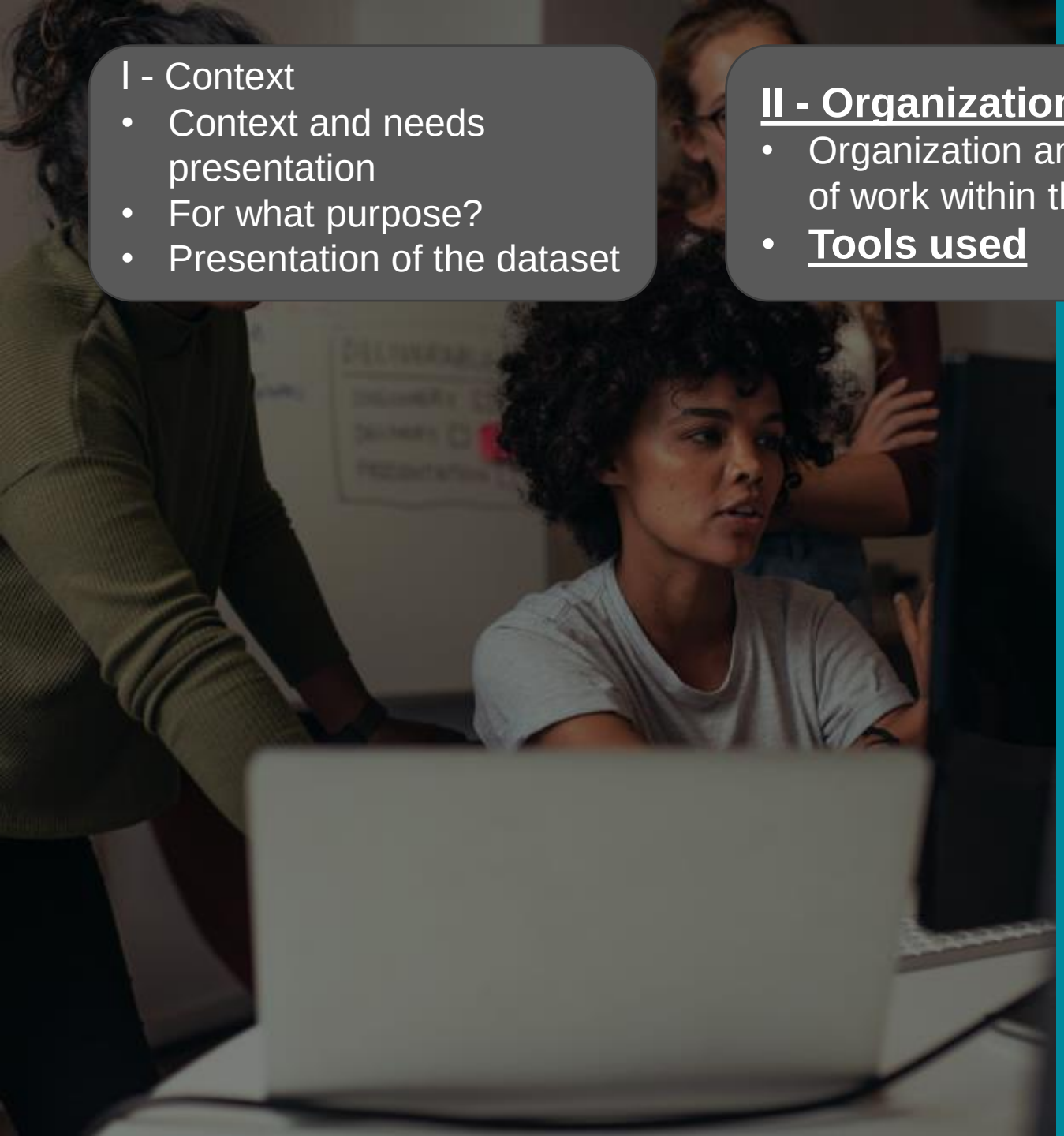
- Answer: volunteering
- Personal Information and Lifestyle
- Health and accidents
- Housing environment

I - Context
- Context and needs presentation
- For what purpose?
- Presentation of the dataset

II - Organization
- **Organization and planning of work within the group**
- Tools used

III - Project
- Presentation interface
- Chart





I - Context
- Context and needs presentation
- For what purpose?
- Presentation of the dataset

II - Organization
- Organization and planning of work within the group
- **Tools used**

III - Project
- Presentation interface
- Chart

I - Context
- Context and needs presentation
- For what purpose?
- Presentation of the dataset

II - Organization
- Organization and planning of work within the group
- Tools used

III - Project
- **Presentation interface**
- Chart

Library import

```
 8  import pandas as pd
 9  import json
10  from datetime import datetime
11  import re
12  import tkinter as tk
13  from tkinter import scrolledtext, filedialog, messagebox
14  import os
15  import subprocess
16  import time
17  import requests
18  import datetime as date
19  from pywinauto import Application
20  from datetime import date
```

Some def

```
141  def extraire_heure(date_str):
142      try:
143          date = pd.to_datetime(date_str, errors='coerce', dayfirst=True)
144          if pd.notna(date):
145              return date.strftime('%H:%M')
146      except Exception:
147          return ""
148      return ""
149
150  def get_tranche_age(age):
151      if pd.isna(age):
152          return ""  # laisser vide si âge inconnu
153      elif age < 18:
154          return "<18"
155      elif 18 <= age <= 35:
156          return "18-35"
157      elif 36 <= age <= 50:
158          return "36-50"
159      elif 51 <= age <= 70:
160          return "51-70"
161      elif age > 70:
162          return "70+"
163      else:
164          return ""
```

I - Context
- Context and needs presentation
- For what purpose?
- Presentation of the dataset

II - Organization
- Organization and planning of work within the group
- Tools used

III - Project
- **Presentation interface**
- Chart

Data cleaning

```
    # ------------- Suppresion des colonnes ------------- #

        df_BD.drop([col for col in df_BD.columns if col.lower().startswith("colonne")], axis=1, inplace=True)
        # Supprimer les colonnes commençant par "Colonne"
        df_Acc = df_Acc.drop([col for col in df_Acc.columns if col.startswith("colonne")], axis=1)
        df_BD = df_BD.drop([col for col in df_BD.columns if col.startswith("colonne")], axis=1)

    # ------------- Modification dans df_BD ------------- #

        if 'taille_cm' in df_BD.columns:
            df_BD['taille_cm'] = df_BD['taille_cm'].apply(convertir_taille)

        if 'annee_naissance' in df_BD.columns:
            df_BD['annee_naissance'] = df_BD['annee_naissance'].apply(convertir_annees_naissance)
            df_BD['annee_naissance'] = pd.to_numeric(df_BD['annee_naissance'], errors='coerce')
            # Calcul de l'âge
            annee_actuelle = date.today().year
            df_BD["age"] = annee_actuelle - df_BD["annee_naissance"]
            # Tranche d'age
            df_BD["Tranche_Age"] = df_BD["age"].apply(get_tranche_age)

        # Séparer 'DATE DE REMPLISSAGE' en 'DATE' et 'HEURE' pour df_BD (avec secondes)
        if 'date_remplissage' in df_BD.columns:
            df_BD['DATE'] = df_BD['date_remplissage'].apply(convertir_date)
            df_BD['HEURE'] = df_BD['date_remplissage'].apply(convertir_heure)
            # Supprimer la clonne
            # Supprimer la clonne
            df_BD.drop(columns=['date_remplissage'], inplace=True)
```

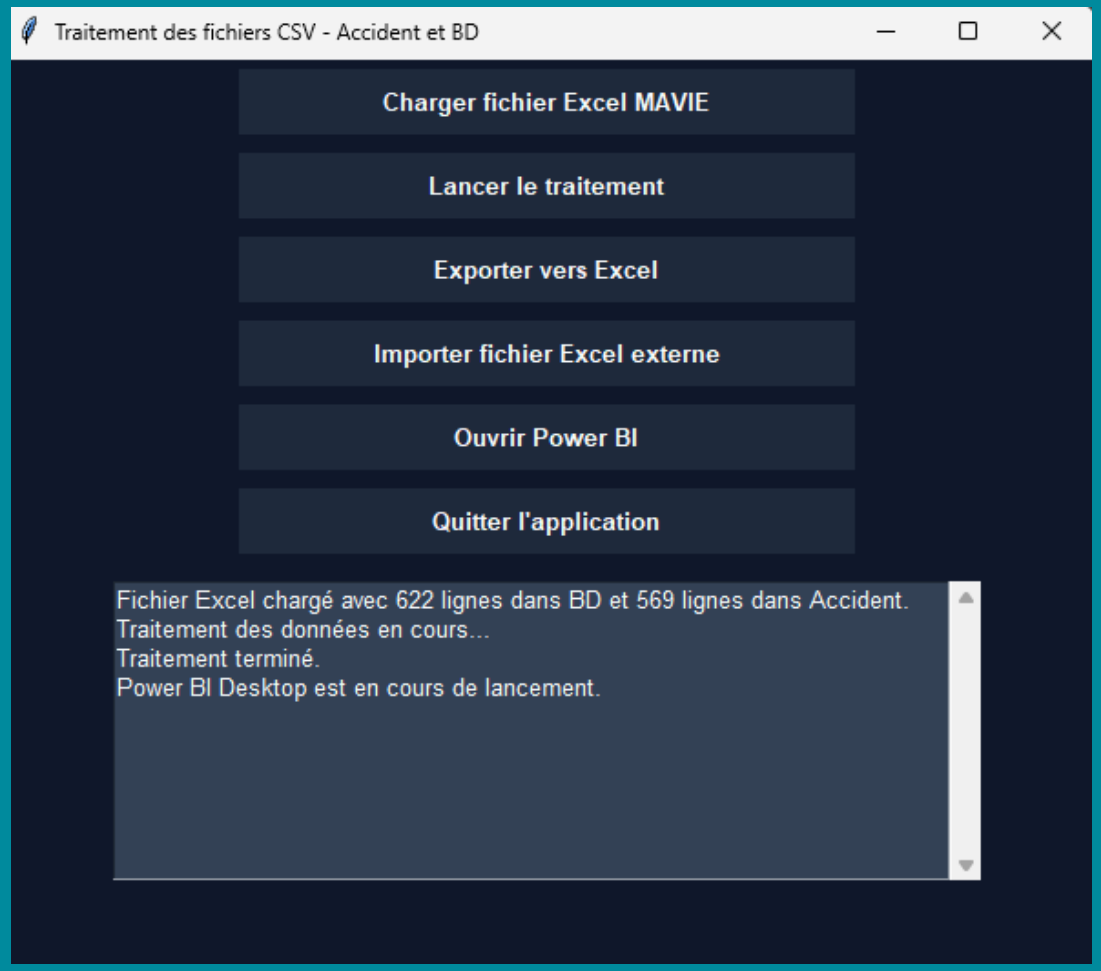
Interface code

```
    # --- Création de la fenêtre ---

    fenetre = tk.Tk()
    fenetre.title("Traitement des fichiers CSV - Accident et BD")
    fenetre.configure(bg=colors["bg"])

    # Taille fenêtre

    largeur, hauteur = 600, 500
    largeur_ecran = fenetre.winfo_screenwidth()
    hauteur_ecran = fenetre.winfo_screenheight()
    x = (largeur_ecran // 2) - (largeur // 2)
    y = (hauteur_ecran // 2) - (hauteur // 2)
    fenetre.geometry(f"{largeur}x{hauteur}+{x}+{y}")


    style_bouton = {
        "bg": colors["card"],
        "fg": colors["text"],
        "activebackground": colors["accent"],
        "activeforeground": colors["highlight"],
        "font": ("Arial", 10, "bold"),
        "relief": tk.FLAT,
        "width": 40,
        "padx": 5,
        "pady": 5
    }

    # --- Boutons ---

    btn_charger_excel = tk.Button(fenetre, text="Charger fichier Excel MAVIE", command=charger_fichier_excel, **style_bouton)
    btn_charger_excel.pack(pady=5)

    btn_traiter = tk.Button(fenetre, text="Lancer le traitement", command=lancer_traitement, **style_bouton)
    btn_traiter.pack(pady=5)
```

I - Context
- Context and needs presentation
- For what purpose?
- Presentation of the dataset

II - Organization
- Organization and planning of work within the group
- Tools used

III - Project
- **Presentation interface**
- Chart

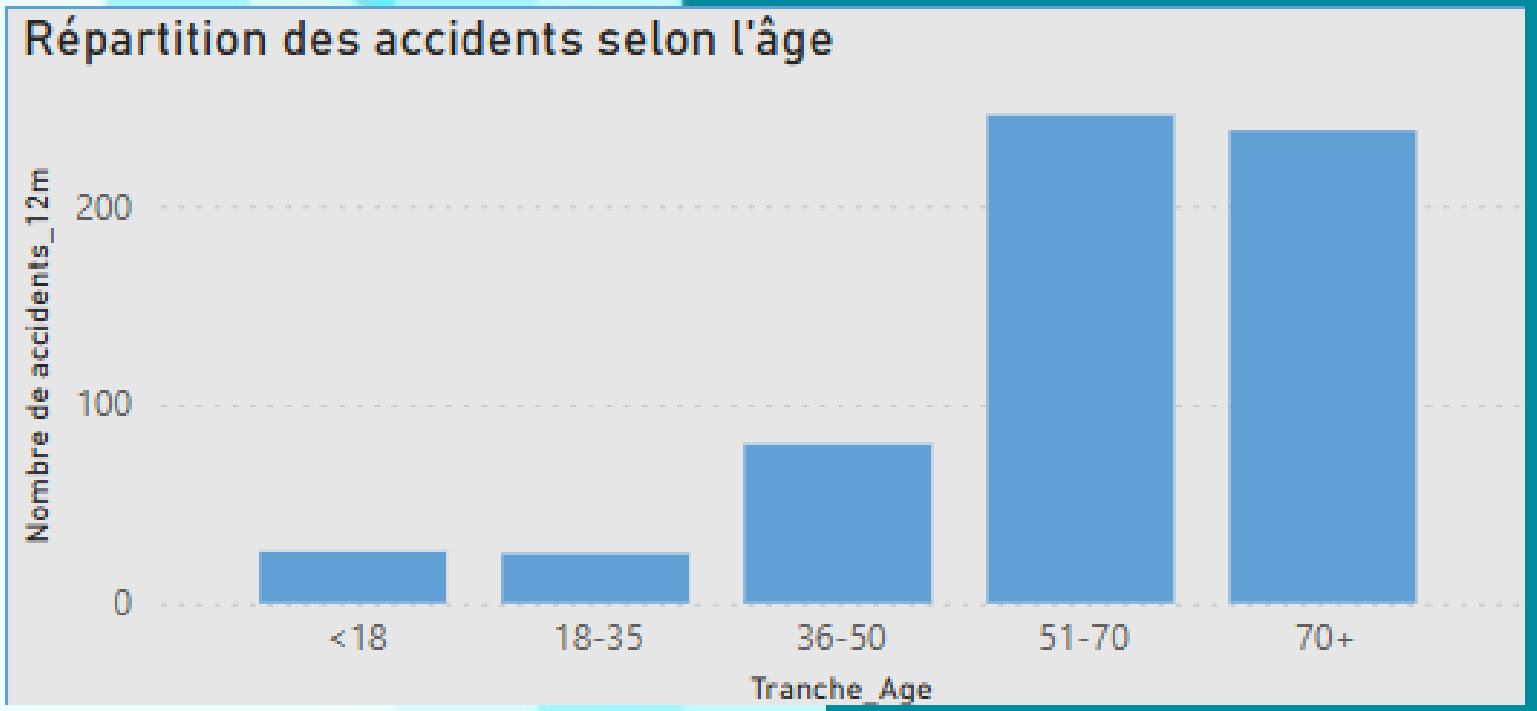


The interface

I - Context
- Context and needs presentation
- For what purpose?
- Presentation of the dataset

II - Organization
- Organization and planning of work within the group
- Tools used

III - Project
- Presentation interface
- **Chart**



To conclude