

Software Engineering Department

Braude College

Extended Project in Software Engineering

25-1-R-7

Capstone Project Phase B

תמונה שמכילה הנעלה, ריקוד, ספורט, החלקה על קרח

התיאור נוצר באופן אוטומטי

Supervisor:

Dr. Julia Sheidin and Dr. Avital Shulner Tal

Student:

Lital Leschinsky

GitHub:

<https://github.com/litalleschinsky/Gait_Detection>

Table of Contents

1. Abstract…………………………………………………………………………….4

. Method ………………………………………….…………………………..…..5-72

2.1 Study Background ………………………………………..…………………….5

2.1.1 Introduction to Gait and Stability……………………………………………5

2.1.2Gender Differences in Postural Control……………...………………….….5

2.1.3 Aging Effects on Balance and Gait……………………………………....….5

2.2 Study Materials …………………………………………………………….…6-7

2.2.1 Data Source: Experiment…………………………………………………....6

2.2.2 Data Review…………………………………………………………………...7

2.3 Introduction to the Problem ……….......................................................…..7

. Research Process……………………………………..……………………...8-113

3.1. Initial Data Understanding and Preprocessing…………...………………....8

3.2. Learning Visualization Techniques……………………………...………..…..8

3.3. Tool Evaluation and Selection………………………………………….....…..9

3.4. Visualization Design and the system GaitStab Development………….….9

3.5. The final System ……………………………………………….………..…...10

3.6. Code Implementation …………………………………..……………….…...11

4.Challenges……………………………………………………………….……11-12

5. Evaluation / Verification Plan …………………………….…………13-15

5.1 Testing Plan ……………………………………………………………….13-14

5.2 Evaluation by the User …………………………………………..……….14-15

6 Result and conclusions………………………………………………………15-16

7 User Documentation…………………………………………………….…...17-27

7.1 user guide…………………………………………………………………...17-23

7.2 Maintenance Manual………………………………………………………24-26

7.2.1 Overview………………………..……………………………………………24

7.2.2 Required Skills………………………………..………………………....….24

7.2.3 Data Format Requirements……………………………………………...…24

7.2.4 Development Environment……………………………………...………….25

7.2.5 Installation Instructions…………………………………….….……………25

7.2.6 Running the Application Locally……………………………………………25

7.2.7 Deployment Instructions……………………………………...…………….26

7.2.8 Code Architecture and Modifiability…………………………….………….26

7.3 Backend Folder Structure……………………………………….…………….27

8. REFERENCES ……………………………………………………..…………...28

9. APPENDIX …………………………………………………….…………….29-33

1.Abstract

In modern times, analyzing human movement, particularly gait patterns, serves as a key tool for understanding functional characteristics and quality of life indicators. This project focuses on developing an advanced visualization system for the clear and precise presentation of gait data, classified according to relevant demographic and clinical parameters such as age, gender, BriefBESTest.

The system displays changes in everyone's gait pattern over time and places these changes within the context of the group to which they belong. This approach enables comparison between personal trends and general group patterns, facilitating the identification of trends, outliers, and shared characteristics. The visualizations include complex graphs and charts that help users interpret individual changes alongside group effects.

The platform is designed to improve data accessibility and enhance the ability to analyze and understand gait data for researchers, professionals, and other stakeholders, without requiring deep technical expertise in biomechanics.

Keywords:

Human movement, visualization, BriefBESTest, gait data.

2. Method

2.1 Study Background

2.1.1 Introduction to Gait and Stability

Gait is a fundamental component of human movement, enabling individuals to perform daily activities while maintaining mobility and independence. It requires precise neuromuscular coordination to ensure balance and postural control throughout the walking cycle. A key aspect of safe and efficient walking is upright gait stability, which refers to the ability to minimize upper body oscillations during locomotion (Iosa et al., 2014)[[1]](#footnote-1). This ability evolves from early childhood and peaks in adulthood, but declines with age due to changes in muscular strength, sensory integration, and cognitive processing.

2.1.2 Gender Differences in Postural Control

Research indicates that gender plays a significant role in postural control. Women tend to demonstrate greater sway and gait variability, which is influenced by anatomical, hormonal, and biomechanical factors. Men generally benefit from higher muscle mass and lower center of gravity, which contribute to more stable gait patterns. These distinctions highlight the importance of incorporating gender-based considerations into gait analysis and interventions  
(Davarci & Anarim, 2023).[[2]](#footnote-2)

2.1.3 Aging Effects on Balance and Gait

Aging is associated with noticeable declines in balance and gait stability. These changes stem from reduced muscle strength, slower reaction times, and weakened sensory feedback mechanisms. Older adults often show increased gait variability and impaired postural control, raising the risk of falls. Studies suggest that targeted interventions—such as strength, balance, and cognitive training—can help mitigate these effects and support safer mobility (Peeters et al., 2019[[3]](#footnote-3); O’Hoski et al., 2015[[4]](#footnote-4)).

2.2 Study Materials

2.2.1 Data Source: Experiment

The dataset used in this project originates from an experiment conducted at Ariel University, under the supervision of DR.Lihi Deri. The experiment aimed to evaluate gait stability using smartphone-based accelerometer measurements. Participants were instructed to hold a smartphone while walking, allowing for continuous data capture over a 20-second interval (see Figure 1). The study collected data from 27 participants (14 men and 13 women), spanning a range of ages and physical characteristics, including gender, height, weight, and BMI.

Each participant’s physical activity level was assessed using the International Physical Activity Questionnaire (IPAQ), which quantifies the intensity and frequency of physical activities performed during a typical week (Peeters et al., 2018)[[5]](#footnote-5). Additionally, participants completed the BriefBESTest—a clinical tool for evaluating balance and stability, with scores ranging from 0 (complete instability) to 24 (high stability, comparable to young adults) (O’Hoski et al., 2014).[[6]](#footnote-6)

This experiment served as a pilot study for examining how gait data can be used to predict demographic characteristics, particularly age groups. Leveraging smartphones for gait analysis introduces a low-cost, accessible method for monitoring mobility, with promising applications in fall prevention and early detection of motor impairments.

Figure 1: Illustration of the smartphone holding position during the data collection process.

תמונה שמכילה טקסט, צילום מסך, עיצוב גרפי, תכונות מולטימדיה

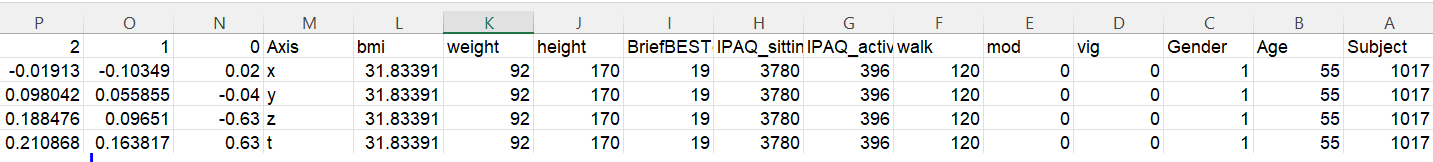
התיאור נוצר באופן אוטומטי

2.2.2 Data Review  
The gathered dataset includes 108 rows representing 27 participants, each contributing four rows corresponding to acceleration readings across three axes (X, Y, Z) and a combined total (T). These readings span 2,000 columns per row, capturing high-frequency (100Hz) motion data over a 20-second period. The dataset also includes demographic attributes (age, gender, BMI, height, weight), physical activity levels (IPAQ scores), and balance/stability assessments (BriefBESTest scores).

To enrich the dataset further, summary statistics were calculated for each axis—such as minimum, maximum, and standard deviation—which are known to correlate with stability indicators in gait analysis literature. These features provide a foundation for identifying mobility patterns, risk factors, and demographic distinctions within the data.

Figure 2:

displays a portion of the raw dataset for one participant.



2.3 Introduction to the Problem

The data collected is currently stored in an extensive Excel file, consisting of many features and measurements (see Figure 2). While rich in content, the data in its raw form is difficult to interpret and analyze effectively. The primary challenge lies in the lack of intuitive presentation, which hinders pattern recognition and makes it harder to assess relationships between gait parameters and demographic or clinical variables.

The main objective of this project is to develop a clear and interactive visualization system that transforms complex data into accessible, insightful visuals. This system enables users to track changes in gait stability over time, compare individual performance to group trends, and identify significant patterns or anomalies. A key aspect of the challenge involves choosing the right type of graph or chart for each data element that is both aesthetically appealing and analytically meaningful.

The visualization approach must help users, whether researchers, clinicians, or other stakeholders—,gain practical insights without needing deep expertise in biomechanics or data analysis.

3. Research Process

As already mentioned, the data for this research project was obtained from an experiment conducted at Ariel University, under the supervision of Dr. Lihi Deri. Throughout the project, I held regular meetings with my academic supervisors, who provided continuous feedback and critical insights that significantly shaped the development process, both technically and conceptually.

3.1 Initial Data Understanding and Preprocessing

The project began with studying the domain by providing a relevant literature review (phase A of the capstone project). It continued with a thorough examination of the dataset, which contained accelerometer readings collected during walking sessions. A key milestone included a direct consultation with Dr. Deri, which helped clarify the structure and context of the experimental data. For example, as a result, we understood that the data contains the following: each participant was represented by four consecutive rows of data, corresponding to motion readings along three spatial axes (X, Y, Z) and a total acceleration value (T). One of the early challenges was to restructure the data so that every group of four rows would be aggregated into a single unit representing one individual. This transformation was essential for preparing the data for analysis and enabled clear comparisons across participants. In addition to motion data, the dataset included key demographic and clinical variables, such as age, gender, height, weight, BMI, physical activity level (based on the International Physical Activity Questionnaire – IPAQ), and balance scores (based on the BriefBESTest).

3.2 Learning Visualization Techniques

Following data structuring, the next step involved exploring and selecting appropriate visualization techniques that could effectively represent the multidimensional and temporal nature of gait data. Static graphs were quickly found insufficient, prompting an investigation of more advanced visual formats. I explored strategies for visualizing:

* Time-evolving motion signals
* Demographic and clinical characteristics
* Inter-variable relationships and trends over time

Special focus was placed on time-based visualizations, as gait is inherently dynamic. I experimented with chart types such as time-series plots, heatmaps, and parallel coordinate plots to capture fluctuations, stability patterns, and group trends.

3.3 Tool Evaluation and Selection

The initial development was carried out in Google Colab, where Python scripts were used for data preprocessing and testing various visualization methods with libraries such as pandas, matplotlib, seaborn, and plotly. I later evaluated external tools such as Tableau, which offers user-friendly dashboards but lacks flexibility for large-scale, time-series processing.

Though I briefly explored the integration of Python into Tableau through TabPy, and considered using D3.js for full customization, the complexity and development overhead of the latter made it less practical. Ultimately, I chose to implement the solution using the Spyder IDE and the Streamlit framework, which enabled interactive, web-based dashboards with real-time updates and file uploads.

3.4 Visualization Design and the system GaitStab Development

The visualization phase began with the development of basic static graphs using Python libraries such as Matplotlib and Seaborn. These early visualizations were useful for initial exploration, helping to understand the structure and patterns within the dataset. At this stage, I experimented with multiple visualization types, including line graphs for time series data and bar charts for demographic summaries. However, these static formats were limited in functionality. Each modification—such as changing filters or comparing specific subgroups—required manual code changes and re-execution, which made the analysis process inefficient and inaccessible for non-technical users.

To address these limitations, I transitioned to using tools that support visualizations with user control. This allowed the system to adjust visual output based on user input, without requiring changes to the underlying code. Various chart types were evaluated based on their clarity, effectiveness, and alignment with the project objectives. The selected visualizations include:

* Time-series plots to illustrate changes in gait stability over time.
* Pie charts to present demographic distributions such as gender and age categories.
* Parallel coordinates plots for comparing multiple numerical variables across participants.
* Grouped bar charts for summarizing balance test scores by subgroup.

The design of these visualizations was refined through feedback received in several meetings with academic supervisors. The focus remained on building visual tools that would be both scientifically valid and usable by individuals without technical expertise.

3.5 The final System

Following the visualization design process, the system was implemented using the Streamlit framework to provide a comprehensive, user-friendly solution for gait data analysis. While earlier development phases relied on static graphs generated through Python scripts, the final system integrates all functionality into a unified software interface that enables flexible exploration and interpretation of the data.

The system allows users to upload gait measurement files in Excel or CSV format, after which the data is automatically validated, cleaned, and structured. Participants are categorized by demographic and clinical attributes, including age group, gender, and BriefBESTest score. Users can apply filters based on these categories and further customize the visual output by selecting the axis of motion (X, Y, Z, or T), statistical method (mean or median), and time intervals to display.

Three main modes of analysis are available within the system:

* Individual Walking Pattern: Presents a participant’s motion over time along a selected axis, with the option to highlight specific subjects or compare across axes.
* Group Walking Pattern: Displays aggregated walking patterns for selected subgroups, enabling comparisons across age, gender, and balance categories.
* Distribution View: Provides an overview of participant characteristics through demographic summaries and multi-variable comparisons.

Visualizations are updated automatically based on user selections, allowing for immediate feedback without the need to re-run code or reload the interface. Users may also export graphs as image files and interact with them using tools such as zoom, pan, and reset.

The modular architecture of the system makes it highly adaptable for future use with different datasets or extended analysis scopes. By supporting both individual and group-level exploration, the system allows for early identification of gait instability or balance impairments, assists in clinical assessments, and promotes informed, data-driven decision-making. The design emphasizes clarity, usability, and accessibility, making the tool suitable for both professional and research contexts—even for users without specialized technical backgrounds.

3.6 Code Implementation

The system was implemented in Python with an emphasis on clarity, efficiency, and modularity. The codebase is structured around distinct logical sections that correspond to the main analysis modes: distribution overview, individual walking pattern, and group walking pattern. Each section operates independently while sharing a common preprocessing pipeline and unified data structure.

To reduce code duplication, shared operations—such as statistical calculations and time filtering—were encapsulated in helper functions (e.g., compute\_stat). These functions are reused across views with different parameters, maintaining consistency while simplifying maintenance.

A significant aspect of the implementation was the use of groupby operations to aggregate and process participant data efficiently. For example, individual-level graphs are generated by grouping the data by subject, then computing the mean or median per timepoint. This logic avoids repetitive looping and supports clean separation between data and presentation layers.

The system relies on Streamlit’s session\_state to retain processed data between views, ensuring fast navigation without reloading. Conditional logic is used throughout to adjust visual output based on user input without repeating code for each scenario. Filters are applied in a layered fashion—first by demographic attributes (gender, age), then by axis, time range, and statistical method—resulting in a powerful yet compact control flow.

The final architecture is optimized for extensibility and performance. It can be easily adapted to support additional analysis modes, new filters, or extended datasets, without major structural changes.

4.Challenges

Streamlit Limitations – While Streamlit enabled rapid development and easy web deployment, it also introduced some constraints. One of the key limitations was the lack of native support for click-based event handling on graphs. This restricted the ability to let users click directly on chart elements (e.g., a line in a graph) to isolate or highlight a specific participant.

To address this, I implemented an alternative solution using a sidebar selection component. Instead of interacting directly with the graph, users can select a specific participant using a dropdown menu labeled "Select Subject to Highlight." Upon selection, the system refreshes the graph and clearly emphasizes the chosen participant’s walking pattern while dimming the others. This approach provided the needed functionality in a stable and user-friendly manner, without relying on custom JavaScript or external extensions.

This design decision was made after testing click-based libraries such as streamlit-plotly-events, which proved to be unstable and difficult to integrate across multiple views. The final solution balances interactivity with reliability and ensures consistent user experience across platforms.

Making the System Accessible Online – As part of the project requirements, it was essential that the final system would be accessible through a standard web browser, without requiring users to install software or run code locally. Ensuring online access was critical for usability, particularly given the intended audience of clinicians and researchers who may lack programming expertise.

To address this, I researched multiple deployment options for hosting Python-based web applications. I initially explored platforms such as Heroku and PythonAnywhere, which support a wide range of application types but require more manual configuration and setup. These platforms were less suited for Streamlit-based systems, particularly when it came to file uploading and layout rendering.

Eventually, I selected Streamlit Cloud, which provides direct integration with GitHub and is optimized for hosting Streamlit applications. This platform allowed me to deploy the system directly from my GitHub repository and generate a public URL for convenient access. It also simplified the update process, as changes pushed to the repository were automatically reflected in the live application.

This solution met all accessibility requirements and ensured that the system could be easily shared with stakeholders and tested across environments without technical barriers.

Working Independently -This project was completed entirely on my own, without a partner or team. I was responsible for all aspects of the work from planning and time management to technical decisions and troubleshooting while maintaining motivation and consistency throughout the process.

5. Evaluation / Verification Plan

5.1 Testing Plan

To ensure the reliability, functionality, and robustness of the developed system, a structured testing plan was implemented. The testing process focused on verifying that all core components operate as intended under a variety of conditions and user interactions. Each function was tested systematically, with expected outcomes predefined and compared against actual results. Tests included both functional and usability aspects of the system, ranging from file upload validation and dynamic graph updates to the responsiveness of the user interface and data integrity across different views. The table below summarizes the key tests performed, along with the expected behavior and actual outcomes recorded during the validation process.

|  |  |  |  |
| --- | --- | --- | --- |
| # | Test Name | Description | Expected Result |
| 1 | File Upload Confirmation | Upload a compatible file | Confirmation message appears; file size is displayed |
| 2 | Invalid File Upload | Upload an incompatible file | An error message appears indicating the file is invalid |
| 3 | Remove Uploaded File | Click the ‘X’ to remove the uploaded file | File is removed; interface resets and allows new file upload |
| 4 | Sidebar Navigation | Navigate between system sections using the sidebar | The displayed graphs update correctly and reflect selected views/filters |
| 5 | Pie Chart Display | View demographic data using pie charts | Charts are clearly displayed and understandable |
| 6 | Parallel Coordinates Graph – Visibility | View the graph with multiple variables across axes | The graph is readable and shows data distribution clearly |
| 7 | Parallel Coordinates Graph – Download and Zoom Features | Use built-in tools to zoom, enlarge, or download the graph | Functionality works as intended |
| 8 | Parallel Coordinates Graph – Filtering | Enable selection of specific participants’ lines for clearer comparison within the graph. | Graph updates accordingly. |
| 9 | Walking Pattern Graph – Clarity | View an individual’s walking acceleration graph | The graph is readable and clearly illustrates the walking pattern |
| 10 | Walking Pattern Graph – Filter Response | Change demographic/clinical filters | Graph updates accordingly |
| 11 | Highlight Specific Subject – Graph Selection | Select a specific subject ID | Only the selected subject’s data is highlighted and displayed correctly |
| 12 | View All Axes for a Single Participant | Display all motion axes (X, Y, Z, T) for a selected participant | Each line is clearly distinguishable and labeled |
| 13 | Group Walking Analysis Graph – Visibility | View group-level aggregated walking patterns | The graph is clearly rendered and easy to interpret |
| 14 | Group Walking Analysis Graph – Filter Responsiveness | Apply filters (e.g., gender, age, test score) | Graph updates dynamically and accurately |
| 15 | Graph Load Time | Load any graph | Graph loads within 3–5 seconds under a stable internet connection |
| 16 | System Responsiveness (General) | Use the system on different screen sizes (e.g., tablet, desktop) | Layout and visuals adapt smoothly without loss of functionality |

5.2 Evaluation by the User

To evaluate the usability and clarity of the system, structured feedback was collected from three users with relevant experience in data analysis and visualization:

* Dr. Lihi Deri, the supervisor of the original experiment and a domain expert in clinical gait research
* A graduate student in data science, with experience working on visual dashboards and statistical tools
* A software engineering peer, familiar with user experience and interface design

Each evaluator interacted independently with the three main views of the system:

* Distribution View
* Individual Walking Pattern
* Group Walking Pattern

They were asked to complete two questionnaires:

1. A custom feedback form evaluating each view on clarity, usability, complexity, and effectiveness (see Appendix A, Tables A1–A3)
2. The System Usability Scale (SUS) – a standardized 10-item questionnaire designed to assess overall system usability (see Appendix B)

Summary of Feedback

Across all views, the evaluators reported high levels of satisfaction and ease of use. The interface was described as intuitive, even for users unfamiliar with gait data. Filters were deemed helpful and responsive, and the ability to switch between views made it easy to explore the data from different angles.

Dr. Lihi Deri specifically noted that the final version successfully incorporated the feedback discussed in earlier stages of development. She found the visual outputs—particularly the parallel coordinates plot and the group comparison charts—clear and informative.

The additional evaluators highlighted the effectiveness of the sidebar filtering mechanism and appreciated the graph customization options (e.g., trend line type, time range). One user suggested that brief textual explanations (tooltips) might help first-time users interpret more complex graphs.

SUS Results

All three evaluators completed the System Usability Scale (SUS) questionnaire. The average score was 92.5 out of 100, indicating excellent usability. The evaluators reported high confidence using the system, and all agreed they would be able to learn and use it without assistance.

A detailed breakdown of the questionnaire responses and SUS scores is provided in Appendix B, Tables B1 and B2.

6.Result and conclusions

The primary objective of this project was to develop an intuitive and flexible system for visualizing gait data, with the ability to analyze and compare both individual and group walking patterns. The system was also required to support meaningful filtering by demographic and clinical factors (such as age, gender, and balance scores), and to remain accessible to non-technical users such as clinicians and researchers.

These goals were fully achieved. The system allows users to upload raw datasets, which are automatically validated, restructured, and processed. It supports multiple views for visual exploration, including demographic summaries, participant-specific motion trends, and group-level comparisons. Users can easily adjust parameters such as time intervals, axes, statistical methods, and filters—all without writing code.

From a technical standpoint, the system meets the requirements for stability, modularity, and clarity. The codebase is cleanly structured and supports future extensions with minimal effort. The filtering and visualization mechanisms are efficient and avoid unnecessary repetition by leveraging reusable functions and structured control logic.

User evaluations confirmed the effectiveness and usability of the system, with high scores on both custom feedback forms and the SUS questionnaire. The interface was found to be clear, informative, and easy to navigate. These findings support the conclusion that the system fulfills its intended purpose as a practical and accessible analysis tool.

Goal Alignment

All major project milestones were successfully met:

* A functioning system was developed and deployed online
* All core features specified in the initial plan were implemented
* Usability and clarity were verified through structured evaluation

The system is ready for practical use and can also serve as a foundation for further research or clinical applications.

7.User Documentation

7.1 User Guide

Upon launching the system, the user is presented with a structured and user-friendly interface. The screen is divided into two main areas:

* A persistent sidebar located on the left side of the screen
* A main content area, which updates according to the user's selections

The sidebar menu serves as the central navigation and interaction hub. It is always visible and allows users to:

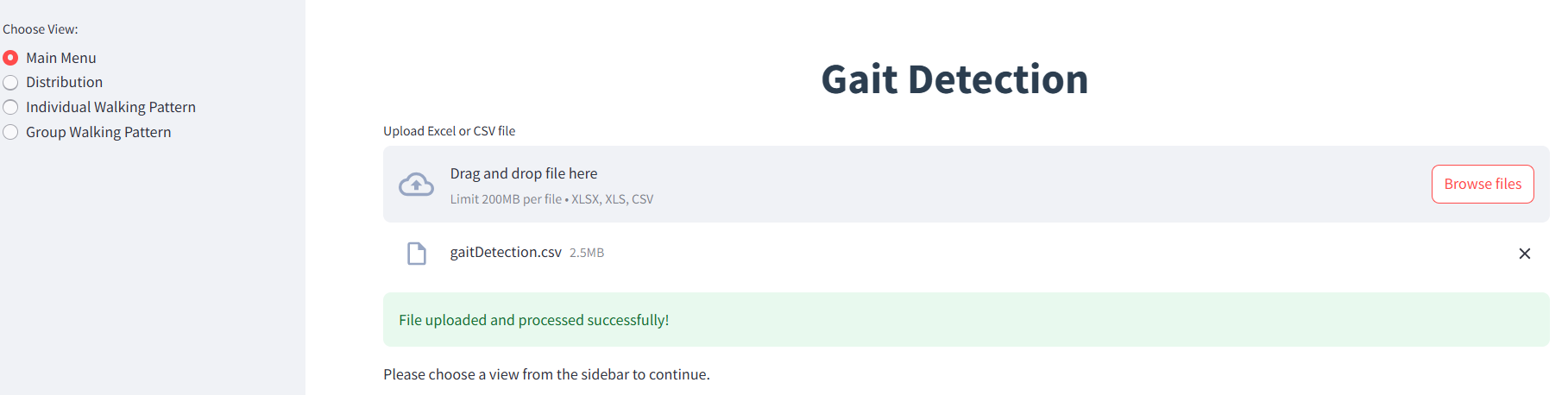
* Switch between the three main system views:
  + Distribution View
  + Individual Walking Pattern
  + Group Walking Pattern

This layout supports seamless exploration and ensures that the user remains oriented throughout the analysis process.

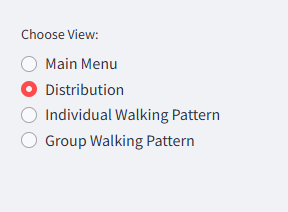
Before accessing any visualizations, the user must upload a dataset. This is done from the Main Menu (the default view selected in the sidebar).



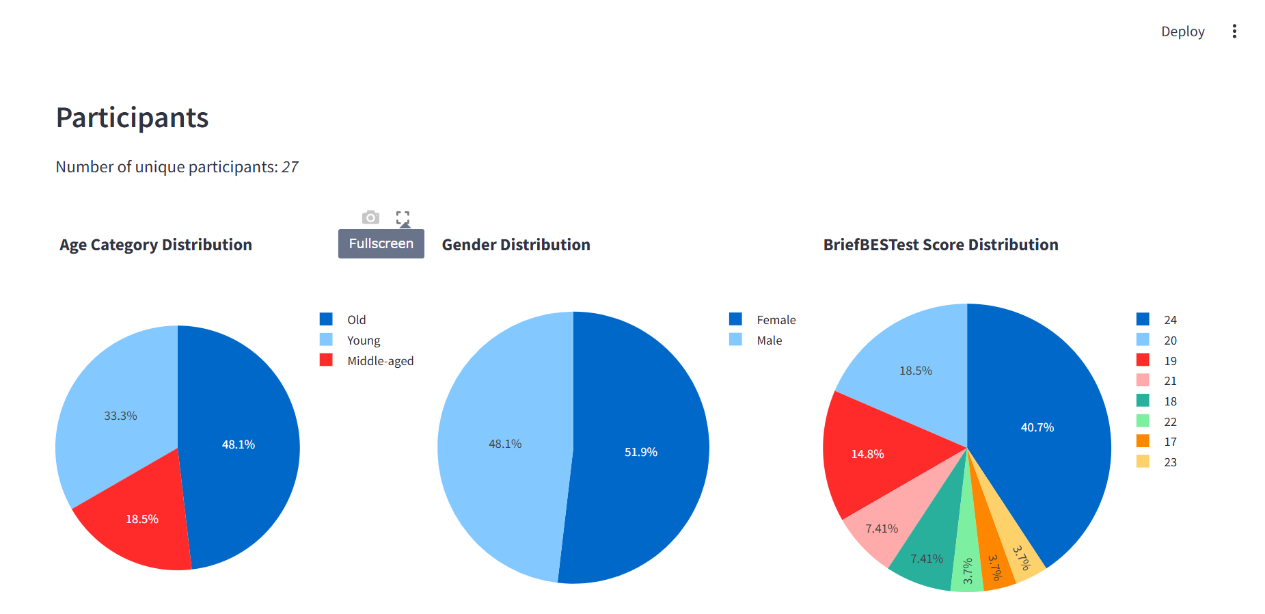
Upon entering the system, the user is prompted to upload a dataset using the **“Browse Files”** button in the Main Menu view. Supported file types include CSV and Excel formats.



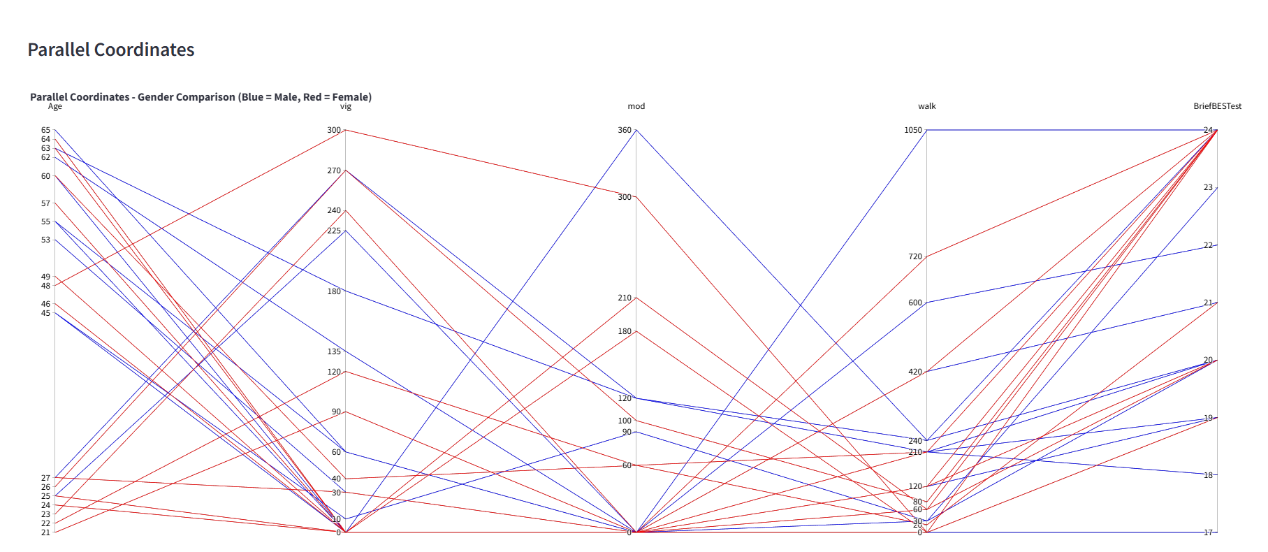
A confirmation message will appear, indicating that the file has been successfully uploaded. The size of the uploaded file will also be displayed. You can navigate between different sections of the system at any time using the sidebar menu located on the left side of the screen. At any stage, you may remove the uploaded file by clicking the 'X' button and upload a new one as needed.



The "Distribution" option is selected; the system will navigate to the following sequence of screens:

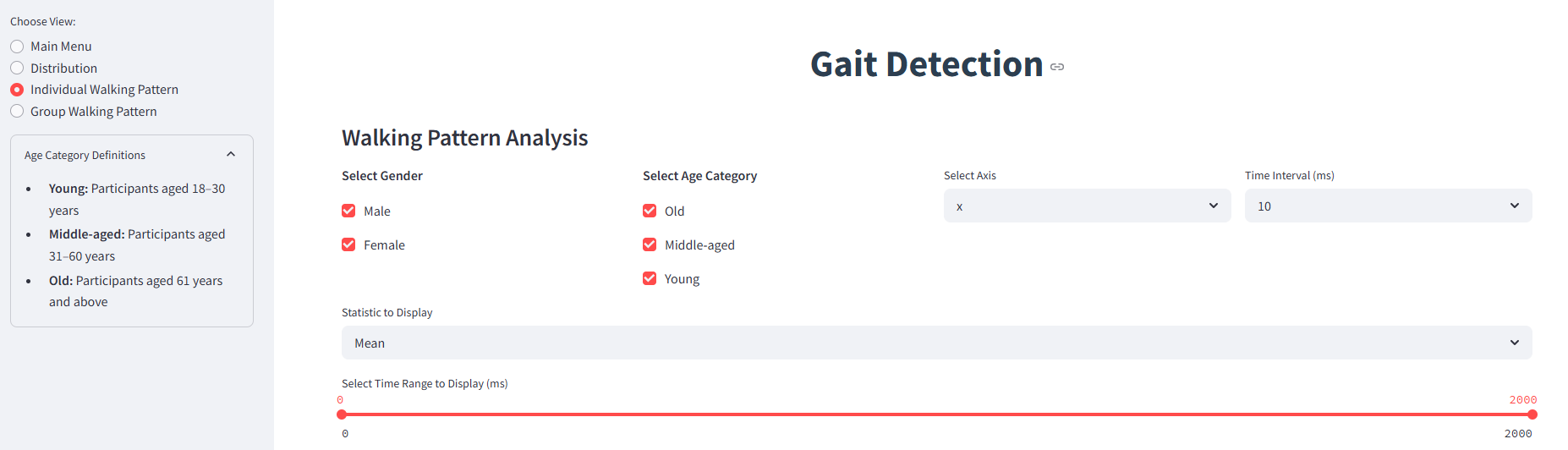
 Initially, the number of participants included in the dataset is displayed. Following this, three pie charts are presented, each enabling comparative analysis of different gait-related characteristics, followed by a relevant legend.

Each chart includes the following interactive features: Clicking the Fullscreen icon allows users to enlarge the chart for clearer viewing. Clicking the camera icon enables users to download the chart as a PNG image. The category list next to each chart is interactive: clicking once on a colored square will temporarily hide that category from the chart. Clicking again on the same square will restore it.

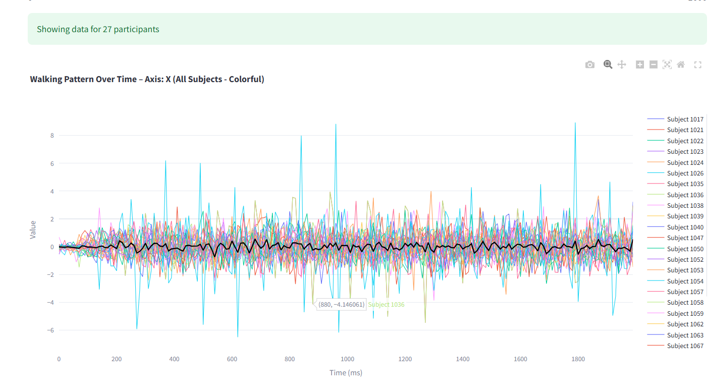
 This chart presents multiple numerical attributes (such as age, physical activity levels, and BriefBESTest) as vertical axes. Each participant is represented by a line that connects their values across all axes. The chart uses color coding—blue for male participants, red for female—to support visual differentiation by gender.

Users can interact with the plot by clicking and dragging on an axis to highlight a specific range of values. This action filters the chart in real time and emphasizes participants within the selected range. Multiple axes can be selected simultaneously, allowing users to focus on specific subsets of the data.  
Double-clicking on any axis resets the chart to its original state, re-displaying all lines and removing any active selections.

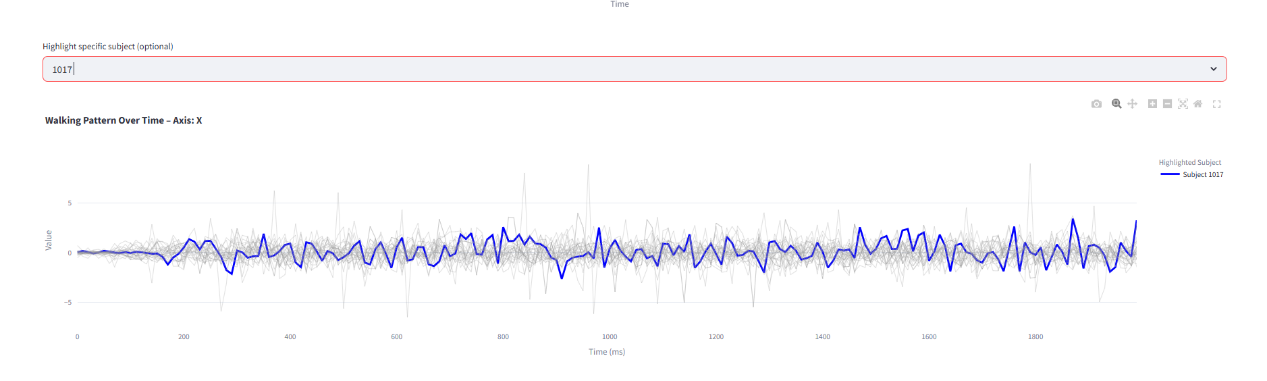
The "Individual Walking Pattern" option is selected; the system will navigate to the following sequence of screens:



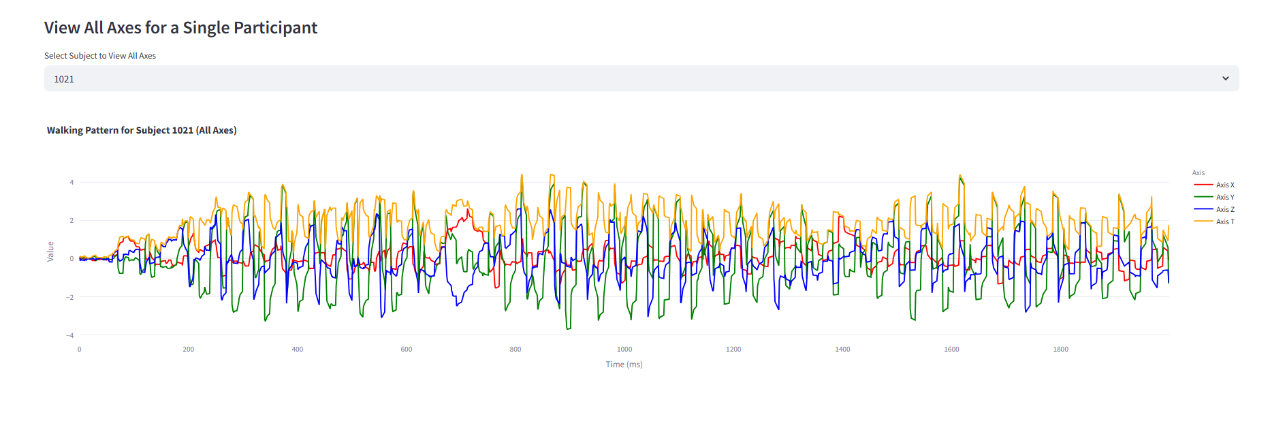
At the beginning, users filter the data based on gender, age range, and the walking progress time interval. These filters help focus the analysis on specific subgroups and time frames. Next, users can choose to display a trend line on the graph, selecting either the mean or the median as the basis for this line. Additionally, users can define the specific time range to display using a slider that spans from 0 to 2000 data points, allowing them to zoom into segments of the walking progress. On the sidebar, there is a clarification regarding the age categories.



The graph shows the walking progress of all participants, each represented by a different color for clarity. On the side, a legend identifies each participant by their assigned color.A black line represents the selected trend line (mean or median), summarizing the overall walking progress within the chosen time frame.



On this graph, users can select a specific participant by their unique ID from the list. Once selected, the corresponding participant is highlighted within the graph, allowing for easy comparison of their progress over time relative to the other participants. This feature enables a focused analysis of how the individual’s data changes in relation to the overall trends.



By selecting a participant's ID from the list, their walking progress is displayed across four color-coded axes, showing changes in their gait over time.

The "Group Walking Pattern" option is selected; the system will navigate to the following sequence of screens:

תמונה שמכילה טקסט, גופן, תוכנה, מספר

תוכן שנוצר על-ידי בינה מלאכותית עשוי להיות שגוי.

As a preliminary step, users filter the data based on age group, gender, and BriefBESTest classification (normal or abnormal). They also select the desired time interval using the “Select Time Interval” slider, which spans from 10 to 200 data points. Additionally, users choose whether the trend line will reflect the mean or the median. Once all selections are made, clicking the “Update Graph” button initiates the generation of the visualization. The sidebar provides a clear explanation of the age categories and the meaning of BriefBESTest values.

תמונה שמכילה צילום מסך, צבעוני, עיצוב

תוכן שנוצר על-ידי בינה מלאכותית עשוי להיות שגוי.

תמונה שמכילה טקסט, צילום מסך, גופן, אלגברה

תוכן שנוצר על-ידי בינה מלאכותית עשוי להיות שגוי.

Based on the filtering selections, the resulting graph displays the walking patterns for each group formed by the Cartesian product of the selected gender, age category, and BriefBESTest classification. Each line on the graph represents a unique group defined by a specific combination of these parameters. The lines are color-coded to distinguish between the different groups, and a legend on the right-hand side clearly indicates which line corresponds to which group. This visualization allows for an effective comparison of walking behavior across multiple demographic and clinical categories over time. In addition, below the graph, the system provides a "Groups with No Data" section. This section lists all possible combinations of gender, age category, and BriefBESTest classification that were selected in the filters but have no corresponding data in the dataset. This feature ensures transparency by highlighting which group intersections are missing, helping users better understand data availability and interpret the results accordingly.



Interactive Features (Common to All Views)

Each graph includes a built-in interactive toolbar with the following tools:

* Download plot as a PNG – Saves the current graph as an image
* Box Zoom – Zooms into a selected rectangular area
* Pan – Allows shifting the view horizontally or vertically
* Zoom In / Zoom Out – Adjusts the magnification level
* Autoscale – Fits the full data back into view
* Reset Axes – Returns to the original axis configuration
* Toggle Spike Lines – Shows or hides guide lines when hovering over data points

Additionally, graphs that contain category legends (such as pie charts or line plots with multiple series) allow users to click on a category label to temporarily hide or show specific elements. This functionality enables focused comparison by removing visual clutter.

7.2 Maintenance Manual

7.2.1 Overview

This maintenance guide is intended for any future developer or researcher who wishes to continue, expand, or modify the gait visualization system. It outlines the technical structure of the project, the required software and libraries, the correct data format, and step-by-step instructions for running and deploying the application. This manual ensures that the system remains understandable, maintainable, and accessible.

7.2.2 Required Skills

To effectively understand, modify, or extend the system, the following skills are recommended:

* Proficiency in Python, particularly in data analysis and visualization.
* Familiarity with libraries such as pandas, plotly, and streamlit.
* Basic knowledge of web development concepts and user interface design.
* Ability to manage virtual environments and work with IDEs like Spyder or VS Code.
* Understanding of GitHub and basic deployment processes.

7.2.3 Data Format Requirements

The system relies on a structured Excel (xlsx) file. It is essential that any dataset used maintains the exact format as used in the original project:

* Each set of 4 rows corresponds to a single participant.
* The columns must be identical in name, order, and data type to those in the original file.
* The code processes the file based on this specific structure, and incorrect formatting will prevent successful parsing or lead to invalid outputs.

Before using a new dataset, verify that the structure matches the original format precisely.

7.2.4 Development Environment

The code was developed and tested using the following setup:

* Programming Language: Python 3.10
* Development Environment: Spyder (via Anaconda distribution)
* Platform: Windows 10
* Web Framework: Streamlit
* Additional Tools: Command Prompt for launching the app

The entire codebase is contained in a Python file named dashboard.py, located at the path: C:/Users/user/.spyder-py3/dashboard.py

The code should be stored and executed from this or a similar fixed path unless explicitly modified.

7.2.5 Installation Instructions

To prepare the environment for running the code, follow these steps:

1. Install Anaconda (recommended for package and environment management):  
   <https://www.anaconda.com/>
2. Install Spyder, if not included with Anaconda: This IDE was used for development and provides an intuitive Python coding environment.
3. Install Required Python Libraries: Open the Anaconda Prompt or terminal and run the following command: pip install pandas plotly openpyxl streamlit
4. Ensure that the working directory in Spyder is set correctly to the folder containing dashboard.py.

7.2.6 Running the Application Locally

To run the system locally as a web app, use the Command Prompt:

1. Navigate to the working environment directory: cd C:/Users/user/.spyder-py3/
2. Execute the following command: streamlit run dashboard.py

This will launch the application in a web browser on your local machine (typically at http://localhost:8501).

7.2.7 Deployment Instructions

To make the system accessible online, follow these steps:

1. Upload the entire project (including dashboard.py and a sample dataset) to a GitHub repository.
2. Visit the Streamlit sharing platform:  
   <https://share.streamlit.io/new>
3. Connect your GitHub account and select the repository that contains the project.
4. Configure the deployment settings (e.g., select the dashboard.py file as the entry point).
5. Streamlit will generate a public link for your app, allowing anyone to access it via a web browser — without the need to install any software locally.

This deployment method is user-friendly and efficient, especially for non-technical stakeholders who need to interact with the dashboard.

7.2.8 Code Architecture and Modifiability

The code is modular and divided into logical sections to enable easy extension or adaptation:

* Data Loading: Reads and processes the Excel file.
* Preprocessing: Reshapes the data to ensure that every 4 rows are grouped per individual.
* Visualization Functions: Generates pie charts, parallel coordinates plots, and time-series graphs using Plotly.
* Streamlit App Logic: Defines the layout, sidebar filters, and interactivity of the system.

To change the systems’s behavior or appearance:

* Modify the logic inside dashboard.py, particularly within the Streamlit St. Sidebar and plotting function blocks.
* If new columns are added to the dataset, ensure preprocessing functions are updated accordingly.

7.3 Backend Folder Structure

The backend of the Gait Analysis system is structured in a modular and maintainable way, ensuring clarity, reusability, and extensibility. Although the application is developed using Python and Streamlit (a frontend-backend hybrid framework), the backend logic is separated into logical components responsible for data ingestion, preprocessing, visualization, and UI rendering.

1. dashboard.py

This is the main script that serves as the application’s entry point. It handles the overall layout, logic flow, and user interactions.

Key responsibilities:  
Loads and processes gait data from an Excel file.  
Handles user input from the sidebar.  
Generates dynamic visualizations using matplotlib and seaborn.  
Provides both group-level and individual-level views of the data.

2. data\_processing.py

A helper module responsible for cleaning, structuring, and preparing the data for analysis.

Typical functions:  
Reading raw Excel data.  
Grouping every 4 rows as a single individual's measurements.  
Filtering data based on clinical or demographic attributes.  
Returning data subsets for downstream visualization.

3. visualizations.py

Contains reusable plotting functions that are imported and used in dashboard.py.  
Key capabilities:  
Line plots and bar charts for temporal gait metrics.  
Dynamic color schemes based on grouping.  
Separated visualizations for group vs. individual data.  
Ensure consistency and customization in all plots.

4. data/ Folder

This folder stores the Excel file used by the system. The input file must follow a strict format.

File requirements:  
Each subject's data must be organized in four consecutive rows.  
Column names must remain identical to the original template.  
The file must be saved in .xlsx format and placed in this directory for the application to function.

8. REFERENCES

1. Iosa, M., Fusco, A., Morone, G., & Paolucci, S. (2014). *Development and decline of upright gait stability*. Frontiers in Aging Neuroscience. <https://doi.org/10.3389/fnagi.2014.00014>
2. Davarci, E., & Anarim, E. (2023). *Gender Detection Based on Gait Data: A Deep Learning Approach With Synthetic Data Generation and Continuous Wavelet Transform*. IEEE Access, 11, 108833–108851. <https://doi.org/10.1109/ACCESS.2023.3280814>
3. Peeters, G., van Schoor, N. M., Cooper, R., Tooth, L., & Kenny, R. A. (2019). *A comprehensive assessment of risk factors for falls in middle-aged adults: Coordinated analyses of cohort studies in four countries*. Osteoporosis International, 30(10), 2099–2117. <https://doi.org/10.1007/s00198-019-05075-8>
4. Peeters, G., Cooper, R., Tooth, L., van Schoor, N. M., & Kenny, R. A. (2018). *Should prevention of falls start earlier? Coordinated analyses of harmonised data on falls in middle-aged adults across four population-based cohort studies*. PLOS ONE, 13(8), e0201989. <https://doi.org/10.1371/journal.pone.0201989>
5. O’Hoski, S., Sibley, K. M., Brooks, D., & Beauchamp, M. K. (2015). *Construct validity of the BESTest, mini-BESTest and briefBESTest in adults aged 50 years and older*. Gait & Posture, 42(3), 301–305. <https://doi.org/10.1016/j.gaitpost.2015.06.003>
6. O’Hoski, S., Winship, B., Herridge, L., Agha, T., Brooks, D., Beauchamp, M. K., & Sibley, K. M. (2014). *Increasing the Clinical Utility of the BESTest, Mini-BESTest, and Brief-BESTest: Normative Values in Canadian Adults Who Are Healthy and Aged 50 Years or Older*. Physical Therapy, 94(3), 334–342. <https://doi.org/10.2522/ptj.20130199>

**Additional References Used in Phase A**

* Mehmood, Z., & Javed, M. (2021). *Tableau for Data Visualization: A Comparative Analysis*. International Journal of Information Engineering and Electronic Business, 11(1), 30–40. <https://www.mecs-press.org/ijieeb/ijieeb-v11-n1/IJIEEB-V11-N1-3.pdf>
* Gehlenborg, N., O’Donoghue, S. I., & Pfister, H. (2021). *A Survey of Visualization Tools for Data Science and Machine Learning*. arXiv:2108.02299. <https://arxiv.org/pdf/2108.02299>

9. APPENDIX

Appendix A – Visualization Evaluation Forms

Table A1 – Distribution View Evaluation

|  |  |  |  |
| --- | --- | --- | --- |
| Statement | User 1 | User 2 | User 3 |
| I thought the visualization was enjoyable | 7 | 6 | 6 |
| I found the visualization unnecessarily complex | 2 | 3 | 2 |
| I thought the visualization was easy to use | 7 | 6 | 6 |
| The use of visualization requires considerable mental effort | 2 | 3 | 2 |
| The use of visualization was clear and understandable | 7 | 7 | 6 |
| The use of visualization was often frustrating | 1 | 2 | 1 |
| I thought the visualization was effective | 7 | 6 | 7 |
| I imagine that most people would easily learn how to use this visualization | 7 | 6 | 6 |

Table A2 – Individual Walking Pattern Evaluation

|  |  |  |  |
| --- | --- | --- | --- |
| Statement | User 1 | User 2 | User 3 |
| I thought the visualization was enjoyable | 6 | 7 | 6 |
| I found the visualization unnecessarily complex | 3 | 2 | 2 |
| I thought the visualization was easy to use | 6 | 7 | 6 |
| The use of visualization requires considerable mental effort | 3 | 2 | 3 |
| The use of visualization was clear and understandable | 6 | 7 | 6 |
| The use of visualization was often frustrating | 2 | 1 | 2 |
| I thought the visualization was effective | 6 | 7 | 6 |
| I imagine that most people would easily learn how to use this visualization | 6 | 7 | 6 |

Table A3 – Group Walking Pattern Evaluation

|  |  |  |  |
| --- | --- | --- | --- |
| Statement | User 1 | User 2 | User 3 |
| I thought the visualization was enjoyable | 6 | 6 | 7 |
| I found the visualization unnecessarily complex | 2 | 3 | 2 |
| I thought the visualization was easy to use | 6 | 6 | 6 |
| The use of visualization requires considerable mental effort | 3 | 3 | 2 |
| The use of visualization was clear and understandable | 6 | 7 | 6 |
| The use of visualization was often frustrating | 2 | 2 | 1 |
| I thought the visualization was effective | 6 | 7 | 7 |
| I imagine that most people would easily learn how to use this visualization | 6 | 7 | 6 |

Appendix B – System Usability Scale (SUS) Results

Table B1 – SUS Item Scores per User

|  |  |  |  |
| --- | --- | --- | --- |
| SUS Question | User 1 | User 2 | User 3 |
| I think that I would like to use this system frequently | 7 | 7 | 7 |
| I found the system unnecessarily complex | 1 | 2 | 1 |
| I thought the system was easy to use | 7 | 7 | 6 |
| I think that I would need support to use this system | 1 | 2 | 1 |
| I found the various functions in this system were well integrated | 7 | 6 | 7 |
| I thought there was too much inconsistency in this system | 1 | 2 | 1 |
| I would imagine that most people would learn to use this system very quickly | 7 | 7 | 7 |
| I found the system very cumbersome to use | 1 | 1 | 2 |
| I felt very confident using the system | 7 | 7 | 6 |
| I needed to learn a lot of things before I could get going with this system | 2 | 2 | 1 |

Table B2 – SUS Scores by User

|  |  |
| --- | --- |
| User | Raw Score (out of 100) |
| User 1 | 95.0 |
| User 2 | 90.0 |
| User 3 | 92.5 |
| Average | 92.5 |

1. Iosa, M., Fusco, A., Morone, G., & Paolucci, S. (2014). *Development and decline of upright gait stability*. Frontiers in Aging Neuroscience. <https://doi.org/10.3389/fnagi.2014.00014> [↑](#footnote-ref-1)
2. Davarci, E., & Anarim, E. (2023). *Gender Detection Based on Gait Data: A Deep Learning Approach With Synthetic Data Generation and Continuous Wavelet Transform*. IEEE Access, 11, 108833–108851. <https://doi.org/10.1109/ACCESS.2023.3280814> [↑](#footnote-ref-2)
3. Peeters, G., van Schoor, N. M., Cooper, R., Tooth, L., & Kenny, R. A. (2019). *A comprehensive assessment of risk factors for falls in middle-aged adults: Coordinated analyses of cohort studies in four countries*. Osteoporosis International, 30(10), 2099–2117. <https://doi.org/10.1007/s00198-019->

   05075-8 [↑](#footnote-ref-3)
4. O’Hoski, S., Sibley, K. M., Brooks, D., & Beauchamp, M. K. (2015). *Construct validity of the BESTest, mini-BESTest and briefBESTest in adults aged 50 years and older*. Gait & Posture, 42(3), 301–305. <https://doi.org/10.1016/j.gaitpost.2015.06.003> [↑](#footnote-ref-4)
5. Peeters, G., Cooper, R., Tooth, L., van Schoor, N. M., & Kenny, R. A. (2018). *Should prevention of falls start earlier? Coordinated analyses of harmonised data on falls in middle-aged adults across four population-based cohort studies*. PLOS ONE, 13(8), e0201989. <https://doi.org/10.1371/journal.pone.0201989> [↑](#footnote-ref-5)
6. O’Hoski, S., Winship, B., Herridge, L., Agha, T., Brooks, D., Beauchamp, M. K., & Sibley, K. M. (2014). *Increasing the Clinical Utility of the BESTest, Mini-BESTest, and Brief-BESTest: Normative Values in Canadian Adults Who Are Healthy and Aged 50 Years or Older*. Physical Therapy, 94(3), 334–342. <https://doi.org/10.2522/ptj.20130199> [↑](#footnote-ref-6)