תמונה שמכילה גופן, טקסט, לוגו, גרפיקה

התיאור נוצר באופן אוטומטי

**Capstone Project Phase B**

**25-1-R-5**

**ParkSmart**

Research of Pattern Identifying and Presentation Based on Personal Data for Support and Quality of Life Improvement for Parkinson's Patients.

**Supervisors:**

**Dr. Julia Sheidin**

**Dr. Avital Shulner Tal**

**Noam Vallach –** [**Noam.vallach@e.braude.ac.il**](mailto:Noam.vallach@e.braude.ac.il)

**Yuval Shekel –** [**Yuval.shekel@e.braude.ac.il**](mailto:Yuval.shekel@e.braude.ac.il)

[**GitHub Link**](https://github.com/YuvalShekel1/ParkSmart)

Table of Contest

[**1. System Implementation** **4**](#_Toc203865291)

[1.1 Background 4](#_Toc203865292)

[1.2 Data Preprocessing 6](#_Toc203865293)

[1.2.1 Initial Data Structure 6](#_Toc203865294)

[1.2.2 Data Processing 6](#_Toc203865295)

[1.2.3 Data Processing Challenges 7](#_Toc203865296)

[1.3 Description of Tools and Algorithms 8](#_Toc203865297)

[1.3.1 Linear Regression as the Core Algorithm 8](#_Toc203865298)

[1.3.2 Main Tools We Used 9](#_Toc203865299)

[1.3.3 Complete System Architecture 9](#_Toc203865300)

[1.4 Challenges We Encountered 10](#_Toc203865301)

[1.4.1 Runtime Issue - Transition from Streamlit to Gradio 10](#_Toc203865302)

[1.4.2 Data Processing and Enrichment Challenge 10](#_Toc203865303)

[1.4.3 Data Quality Challenge 11](#_Toc203865304)

[**2. Results and Pattern Analysis** **11**](#_Toc203865305)

[2.1 Types of Insights and Patterns 12](#_Toc203865306)

[2.1.1 Basic Insights by Categories 12](#_Toc203865307)

[2.1.2 Advanced Insights and Complex Combinations 12](#_Toc203865308)

[2.1.3 Personalized Insights 12](#_Toc203865309)

[2.1.4 Pattern Analysis Example 13](#_Toc203865310)

[**3. Evaluation and Feedback** **14**](#_Toc203865311)

[3.1 Technical System Evaluation 14](#_Toc203865312)

[3.1.1 Runtime Performance 14](#_Toc203865313)

[3.1.2 Accuracy and Reliability of Identified Patterns 14](#_Toc203865314)

[3.1.3 Accuracy Level of Linear Regression Algorithm 15](#_Toc203865315)

[3.2 System Usability Evaluation - SUS Questionnaire 16](#_Toc203865316)

[3.2.1 Usability Questionnaire Description 16](#_Toc203865317)

[3.2.2 Testing Methodology 17](#_Toc203865318)

[3.2.3 Usability Questionnaire Results 18](#_Toc203865319)

[**4. User Guide** **19**](#_Toc203865320)

[**5. Maintenance Guide** **26**](#_Toc203865321)

[5.1 System Requirements and Setup 26](#_Toc203865322)

[5.1.1 Basic Requirements 26](#_Toc203865323)

[5.1.2 Work Environment Setup 26](#_Toc203865324)

[5.2 Deployment and Operation 26](#_Toc203865325)

[5.2.1 Local Operation 26](#_Toc203865326)

[5.2.2 Cloud Deployment 26](#_Toc203865327)

[5.3 Data Management and Caching 27](#_Toc203865328)

[5.3.1 Data File Structure 27](#_Toc203865329)

[5.3.2 Caching System 27](#_Toc203865330)

[5.4 System Modules 27](#_Toc203865331)

[5.4.1 Data Processing Module 27](#_Toc203865332)

[5.4.2 Pattern Analysis Module 27](#_Toc203865333)

[5.4.3 User Interface Module 28](#_Toc203865334)

[5.5 Common Troubleshooting 28](#_Toc203865335)

[5.5.1 Performance Issues 28](#_Toc203865336)

[5.5.2 Translation and Nutrition Data Issues 28](#_Toc203865337)

[5.5.3 Deployment Issues 28](#_Toc203865338)

[5.6 Backup and Recovery 28](#_Toc203865339)

[5.6.1 Backup Procedures 28](#_Toc203865340)

[5.6.2 Failure Recovery 29](#_Toc203865341)

[5.7 Future Planning 29](#_Toc203865342)

[5.7.1 Development Areas 29](#_Toc203865343)

[5.7.2 Update Procedures 29](#_Toc203865344)

[**6. Conclusions and Future Work** **29**](#_Toc203865345)

[6.1 Evaluation of Project Goal Achievement 29](#_Toc203865346)

[6.2 Lessons Learned from the Project 30](#_Toc203865347)

[6.3 Recommendations for Future Development 30](#_Toc203865348)

[6.4 Potential Impact and Future Applications 31](#_Toc203865349)

[6.5 Conclusions 31](#_Toc203865350)

[**7. Appendix** **32**](#_Toc203865351)

[7.1 System Usability Scale (SUS) – Parkinson's Pattern Identifying 32](#_Toc203865352)

[**References** **36**](#_Toc203865353)

# 1. System Implementation

## 1.1 **Background**

During Part A of the project, we based our research on an identified need among Parkinson's patients to obtain personal insights from their daily data. We concluded that identifying patterns in the daily behaviors of Parkinson's patients and the ability to present them clearly and accessibly can significantly impact their quality of life.

Parkinson's disease is a progressive neurodegenerative disorder that primarily affects the dopaminergic system in the brain, causing gradual deterioration in motor and cognitive abilities [1]. The disease is characterized by both motor symptoms such as tremor, bradykinesia, rigidity, and postural instability, as well as non-motor symptoms including depression, sleep disorders, and cognitive impairment [2]. These symptoms create substantial challenges in daily functioning, with patients often experiencing difficulties in performing routine activities and maintaining their quality of life [3].

Managing Parkinson's disease requires a personalized approach, as symptoms and treatment responses vary considerably among patients [4]. Digital health technologies allow patients to collect and monitor data on their daily activities, nutrition, medication use, and symptoms. However, the vast amount of data generated often remains underutilized, as accessible tools for identifying meaningful patterns and deriving actionable insights are lacking [5]. This gap highlights the critical need for user-friendly systems that can transform raw personal health data into comprehensible insights that empower patients to make informed decisions about their daily management strategies.

Our project is a continuation of a previous project that focused on collecting data from a Parkinson's patient. It is important to note that we did not connect directly to the system developed in the previous project, but rather received from them a JSON file containing all the data collected from the patient over time. This file served as the basis for our work and with it we developed the entire system.

It should be noted that the system we developed is not limited only to Parkinson's patients. The system was designed generally to receive daily data from patients and extract insights from them, and therefore it can serve a wide range of chronic diseases and medical conditions. The principles of identifying patterns in nutrition, activity, medication, and symptom data are relevant to many patients who need personal monitoring and understanding of the effects of their daily activities on their health condition.

The goal of the system is to allow the specific user for whom the system is intended to upload a JSON file containing their personal data and receive advanced analysis that displays identified patterns in different categories. The system specializes in analyzing activity, medications, symptoms, and nutrition, with patterns displayed using linear regression based on their impact on the user's mood, Parkinson's state or Physical state.

The development process was based on a user-centered design approach. The development included selecting technological tools suitable for a target population with cognitive and motor challenges, while ensuring the creation of a simple and accessible interface. The system is implemented using the Gradio platform for building interactive user interfaces, with a linear regression algorithm chosen for its transparency and clear explanatory capability.

The technical architecture includes components for processing multilingual data, integration with external databases for enriching nutritional information, and cloud deployment that enables global accessibility to the system. The system design was based on principles of accessibility and user-friendliness, in accordance with recommendations for developing digital tools for chronic patients [6]. The results are presented using an intuitive color system that helps users quickly understand which factors have a positive (green), negative (red), or no significant (black) effect on their health condition.

## 1.2 Data Preprocessing

### **1.2.1 Initial Data Structure**

The data collected from Michael, a Parkinson's disease patient who serves as the specific user of our system, is organized in a JSON file containing five main categories:

* The nutrition category includes information about consumed food, date and time of consumption, and after processing also detailed nutritional values.
* The activities category contains records of physical or daily activities, including exact date and time, duration, and intensity.
* The medications category records precise dosages and administration times.
* The symptoms category describes reported symptoms along with severity ratings, each accompanied by a specific date and time of occurrence.
* The feelings category records daily mood, Parkinson's state, and physical’s state condition with precise timestamps and notes.

### **1.2.2 Data Processing**

The raw data underwent a comprehensive processing pipeline that prepared it for analysis. The system identified Hebrew texts and translated them to English using a two-stage translation mechanism: first, a manually constructed local dictionary containing verified translations for common expressions, and second, the translatepy library connected to an external API for new values not found in the dictionary. All translations were saved in a local cache to ensure consistency and improve performance. For nutritional information, the system added detailed nutritional values (proteins, carbohydrates, fats and dietary fibers) for every reported food item using a local data repository containing 60+ basic food items with complete nutritional profiles. When a food was not found in the local data repository, the system searched for it in an external data repository through the USDA FoodData Central API with intelligent search algorithms, combined with a cache mechanism to prevent duplicate API calls and improve performance. The system also calculated nutritional values for complex meals and partial portions using specialized algorithms that parse food descriptions, identify terms that indicate portion size, such as "half," "quarter," or specific gram measurements, and perform weighted nutritional calculations based on ingredient ratios. Additionally, the system standardized date formats using pandas datetime conversion functions and performed comprehensive quality checks including merging similar medication fields, automatic creation of missing category fields, and data validation to ensure temporal consistency across all records for reliable analysis.

### **1.2.3 Data Processing Challenges**

The initial processing of the data created several complex technical challenges that required creative and customized solutions.

The first challenge focused on handling the multilingual complexity of the data. The original data included a mixture of Hebrew and English, where the Hebrew part included not only individual words but also complex expressions and descriptions of feelings and activities. This situation created problems in data analysis. To solve this, we developed a dedicated translation mechanism consisting of two integrated layers .In the first layer, we manually created a fixed and comprehensive dictionary containing accurate and verified translations for common and recurring values. This dictionary ensures fast, consistent, and reliable translation for frequently used expressions .The second layer handles dynamic translation of new or uncommon values not found in the fixed dictionary, using an external API accessed via the translatepy library. These translations are then automatically cached to avoid repeated calls and improve performance over time .This dual-layer approach ensures both accuracy and scalability, enabling smooth processing of multilingual.

The second challenge focused on the lack of detailed nutritional information. The original data included only general food names like "half pita with peanut butter" or "vegetable soup with ptitim," without any information about nutritional values. This situation prevented structured and meaningful nutritional analysis, which is an essential part of understanding the relationship between nutrition and user's mood, Parkinson's state and physical state. The implemented solution included developing a three-tier approach for handling nutritional information. In the first layer, we manually created a local dictionary containing nutritional data for common food items, with protein, carbohydrate, fat, and dietary fiber values sourced from standard nutritional references. The second layer includes an algorithm for calculating nutritional values of complex meals, identifying portion sizes like 'half pita' or '50 grams chicken' and calculating the precise nutritional values accordingly. The third layer serves as backup through integration with the existing USDA FoodData Central API for foods not found in our local dictionary.

The third challenge included normalization and adaptation of different formats. The system performs standardization of date formats, merging of similar fields such as different medication fields, and ensuring consistency throughout the data structure. This process is vital for performing reliable and accurate analysis of the various patterns.

## 1.3 Description of Tools and Algorithms

### **1.3.1 Linear Regression as the Core Algorithm**

The choice of the central algorithm for the system was based on several important technical and practical considerations. Linear regression was chosen as the basic algorithm for pattern identification due to its transparency and simplicity in explaining results to the user. Linear regression analysis is used to predict the value of a variable based on the value of another variable[7]. Unlike more complex machine learning algorithms that operate as a "black box," linear regression allows direct understanding of each variable's impact on the final result. The algorithm demonstrates the ability to show precise quantitative impact through regression coefficients, so the user can understand exactly how much each activity, medication, food or symptom affects their condition.

### **1.3.2 Main Tools We Used**

Git- Project development management was based on extensive use of Git for code version control and efficient collaboration between team members. Git is a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency [8]. The electronic repository was organized in a structured manner containing the main system file and detailed documentation of all changes and processes performed. This approach enabled precise tracking of development progress and convenient access to previous versions when needed.

Gradio **-** The choice of the Gradio platform for building the user interface was based on several significant advantages. Gradio is an open-source Python library for creating simple and interactive interfaces for machine learning models. It allows you to quickly build web-based user interfaces that enable users to interact with your machine learning models, provide input and receive model predictions or responses[9], which made the development process efficient and fast. One of the significant advantages of Gradio is its simple and efficient support for file handling. The system allows the user to easily upload their JSON file and use the processed results without requiring advanced technical knowledge.

Render **-** Cloud Deployment and Execution The Render platform was chosen as a solution for deploying the system in the cloud and making it available to the user at any time and from anywhere. One of the main advantages of Render is its automatic synchronization system with the Git repository. Every time an update is made to the code and sent to Git, the system automatically updates to the new version, but requires manual activation of the server for each update.

### **1.3.3 Complete System Architecture**

The system operates according to a structured and efficient architecture that begins with uploading a JSON file by the user through the Gradio interface. The data goes through a comprehensive processing stage that includes automatic translation of Hebrew texts using a local repository for recurring expressions and referencing an external library for content not found in the repository, enrichment with detailed nutritional information through a local nutritional database and USDA FoodData Central API for missing items, and standardization of date formats. After the initial processing stage, the processed data is fed into the pattern analysis module based on advanced linear regression. This module performs multiple and comprehensive analyses on all different categories, including activities, medications, symptoms, and nutrition. The results are processed into a user-friendly format and displayed using an intuitive color system that distinguishes between positive effects (green), negative effects (red), and neutral effects with no significant impact (black).

## 1.4 Challenges We Encountered

### **1.4.1 Runtime Issue - Transition from Streamlit to Gradio**

The first and significant technical challenge we faced centered on system performance. During initial development, we chose Streamlit as the platform for building the user interface, but during testing we discovered that the runtime for data processing was too long. Processing a typical data file took several minutes, creating a poor user experience that was impractical for daily use.

The solution included a comprehensive transition to the Gradio platform which showed significantly improved performance. Gradio provides better capability for handling large and complex files. The transition required partial code rewriting but resulted in significant improvement in response times.

### **1.4.2 Data Processing and Enrichment Challenge**

Processing the raw data created two main challenges that required advanced technical solutions. The first challenge focused on handling multilingual data, where the original file contained a mixture of Hebrew and English with complex expressions and emotional descriptions in Hebrew that complicated the analysis process. The second challenge concerned the lack of detailed nutritional information, as the data included only general food names without nutritional value specifications, which limited nutritional analysis capabilities.

To solve the linguistic challenge, the system implements a two-stage approach: a manually created local translation repository for common expressions ensuring fast and consistent translation, and connection to an external translation service with caching for new texts not found in the fixed repository. To address the lack of nutritional information, a multi-layered system was developed including a manually created nutritional dictionary, algorithms for calculating complex meals and partial portions, and connection to the USDA FoodData Central API as a backup solution. This process ensures that the processed data is linguistically uniform and nutritionally enriched.

### **1.4.3 Data Quality Challenge**

The final challenge focused on existing data quality challenges. Relatively small datasets, inconsistent or missing data, and high variability in documentation methods can lead to unreliable or misleading results. This situation poses a significant risk to system reliability and the insights it provides.

The solution included multiple layers of quality checks and data validation. We added minimum requirements for the amount of data needed to perform each type of analysis, so the system would not display results based on insufficient information.

# 2. Results and Pattern Analysis

The system we created successfully identifies and displays complex and meaningful patterns from the personal data of Parkinson's patients. The initial data processing stage requires approximately 90 seconds for complete dataset processing, including Hebrew text translation and nutritional data enrichment. The results are presented in four main categories corresponding to different life domains that affect the patient's condition: activities, medications, symptoms, and nutrition. Each category is analyzed separately using advanced linear regression, providing precise quantitative insights about each factor's impact on mood, Parkinson's state, or physical state. The system requires a minimum of 3 observations per pattern to ensure statistical reliability, which directly affects the number of insights that can be derived - typically generating 8-12 significant patterns per category when sufficient data is available.

The system displays results using colors that clearly distinguish between positive, negative and neutral effects. This design allows the user to identify at a glance which factors help them and which harm their condition. Positive patterns are displayed in green, negative patterns in red, and neutral patterns in black, representing patterns without significant impact for information completeness and providing a comprehensive picture. This color system is especially crucial for Parkinson's patients, who face cognitive challenges and may have difficulty processing complex information. Clear visual presentation facilitates pattern identification and quick decision-making in daily life.

## 2.1 Types of Insights and Patterns

### **2.1.1 Basic Insights by Categories**

The system provides four main types of analysis suited to the different life domains of Parkinson's patients. Activity analysis reveals which activities improve the condition and which harm it, considering intensity and duration of activity. Medication analysis identifies effects of different dosages. Symptom analysis provides insights about the relationship between specific symptoms and general condition. Nutrition analysis reveals which foods and nutritional components support status and which might be harmful.

### **2.1.2 Advanced Insights and Complex Combinations**

Beyond basic analysis, the system provides advanced insights that combine multiple variables together. For example, the system provides insights about the combination of activity type, activity duration, and intensity. The system can identify, for instance, that table tennis at high intensity for 30 minutes improves mood, while the same activity for an hour and a half might worsen the Parkinson's condition.

### **2.1.3 Personalized Insights**

All insights are tailored to the specific user's profile and behavioral patterns. The system is based on the user's personal data and generates insights that relate only to them. This approach ensures that insights will be relevant and practical for the specific user's daily life.

### **תמונה שמכילה טקסט, גופן, צילום מסך תוכן שנוצר על-ידי בינה מלאכותית עשוי להיות שגוי.2.1.4 Pattern Analysis Example**

Figure 1. Activity Pattern Analysis Interface - Basic and Detailed Results

Figure 1 presents an analysis of the impact of activities on the user's Parkinson's state in the system. The system identified various patterns using the linear regression algorithm, with results presented at two analysis levels: basic and detailed. The basic analysis shows general effects of activity types and intensities, while the detailed analysis reveals complex patterns that combine duration, intensity, and activity type together.

In the basic part of the analysis shown on the left side of Figure 1, the system identified that table tennis positively affects the Parkinson's state with an average decrease of 12.8%, as shown in green. Similarly, moderate intensity activity showed a positive effect with a 2.2% decrease in Parkinson's state. In contrast, assembling garden chairs showed a negative effect with a 12.8% increase in Parkinson's state, and low-intensity activity caused a 2.2% deterioration. Activity duration as an independent factor showed no significant impact, as displayed in black.

The detailed analysis reveals more interesting patterns, as demonstrated in the right column of Figure 1. Table tennis for more than 60 minutes at moderate intensity shows the highest positive effect with a 13.8% decrease in Parkinson's state. In contrast, table tennis for less than 30 minutes shows a negative effect with a 9.4% increase, even when performed at moderate intensity. This insight demonstrates the importance of proper combination between activity duration and intensity, not just choosing the right activity.

The system implemented advanced quality control mechanisms to ensure pattern reliability. Each pattern is displayed only if there are at least 3 observations supporting it, and the system displays only regression coefficients that exceed a significance threshold of 0.2 for activities. The color system adapts itself to the type of field being examined - for Parkinson's state, a decrease in the measure is displayed as positive (green) and an increase as negative (red).

The system identifies personal and focused patterns. Instead of providing a general insight about table tennis, the user can see that performing the activity for over 60 minutes at moderate intensity is associated with a positive effect. These data-driven patterns can support personalized daily planning and help users make informed decisions regarding activity choices.

# 3. Evaluation and Feedback

## 3.1 Technical System Evaluation

### **3.1.1 Runtime Performance**

The system showed significant improvement in runtime performance after transitioning from the Streamlit platform to Gradio. In tests we conducted, data file processing time decreased from approximately 10 minutes to 90 seconds, representing a reduction of 8.5 minutes and an 85% improvement in processing speed. This improvement enables a good user experience that is practical for daily use.

The relatively long time stems from the technical complexity of processing procedures, including automatic translation of Hebrew texts, enriching nutrition data from external APIs, and normalizing date formats. Each step is performed with extra caution to ensure data accuracy and reliability.

### **3.1.2 Accuracy and Reliability of Identified Patterns**

The system implements several advanced quality control mechanisms to ensure the reliability of the identified patterns. First, the system requires at least 3 occurrences of each item (such as a specific medication, activity, or symptom) in order to include it in the analysis. Items that appear fewer than 3 times in the dataset are automatically filtered out, as their patterns are not considered statistically reliable.

In addition, the system applies a statistical significance threshold. Only patterns with a regression coefficient of at least ±0.1 are considered significant and shown to the user. If a pattern falls below this threshold, it is either presented in black—if it still provides some informational value—or excluded entirely when there is insufficient data to determine a meaningful trend.

To improve accuracy, the system also performs smart matching between events and outcome measures (such as mood). For medications, mood reports are evaluated within 4 hours after administration, which reflects the typical timeframe during which medication effects are expected. For meals, mood changes are assessed within 3 hours after eating. In the case of physical activities, the system analyzes mood trends from the end of the activity until the end of the same day.

Important clarification: The system takes into account the scale direction of each target field. For “My Mood” and symptom ratings, higher values mean improvement, while for “Parkinson’s State” and “Physical State”, lower values mean improvement.

Together, these mechanisms help ensure that the patterns presented to the user are not only statistically meaningful, but also personalized and clinically relevant.

### **3.1.3 Accuracy Level of Linear Regression Algorithm**

The choice of linear regression as the core algorithm was deliberate and based on several key considerations. First, transparency: unlike black-box models, linear regression offers full interpretability. Each regression coefficient directly quantifies the impact of a variable on the outcome, enabling statements such as: "Walking increases my mood by 1.8% on average."

Second, the model's explanatory capability allows the system to present clear, numerical insights to users. For example, a medication may be shown to reduce physical symptoms by 0.25 or that protein intake increases mood by 0.4 on average, helping patients better understand what works for them.

Third, linear regression is well-suited for personal datasets with relatively small sample sizes. In our case, the system requires a minimum of 3 data points per variable to include it in the model. Most insights are generated from 5 to 40 observations, depending on data availability. On average, each category (activity, medication, symptom, nutrition) yields 8–12 valid patterns when sufficient data is present.

To ensure result reliability, the system applies a minimum significance threshold of ±0.1 to the regression coefficients. Patterns below this threshold are either displayed in black for reference. In internal validation tests, the average R² value of the regressions ranged from 0.35 to 0.72, depending on the category and amount of input data, indicating a reasonable fit for personal-level health data.

Together, the statistical criteria, filtering mechanisms, and the linear regression model ensure a balance between simplicity, clarity, and analytical robustness, making the system practical and trustworthy for daily decision-making.

## 3.2 System Usability Evaluation - SUS Questionnaire

For evaluating the system, we created SUS questionnaire, and 24 participants preformed a test in the system and evaluated it. The participant filled out a few informative questions, then they were presented with the test scenario. After understanding and performing the test scenario, they were requested to answer some post-test questions.

### **3.2.1 Usability Questionnaire Description**

In order to evaluate the usability and convenience level of the system we created, we implemented the SUS (System Usability Scale) questionnaire - a standard and recognized industry tool for measuring the usability of technological systems. The System Usability Scale (SUS) is a questionnaire that is used to evaluate the usability of products and services. These survey questions are used as a quantitative method to evaluate and get valuable insights into the usability of a wide variety of new systems, whether software or hardware[10]. This questionnaire is considered one of the most reliable and efficient tools for evaluating user experience.

The system usability scale consists of only 10 questions, which are answered using a Likert scale[10]. These questions examine different aspects of system usability: ease of use, system complexity, integration of various functions, interface consistency, ease of learning, and the user's confidence level during use. The final result is calculated as a normalized score between 0-100.

For evaluating our system, the SUS questionnaire provides an assessment tool specifically tailored to the unique challenges of a pattern analysis system for Parkinson's patients. The questionnaire allows us to evaluate whether the interface is simple and intuitive enough for users who may face cognitive and motor challenges resulting from the disease. Additionally, the questionnaire helps examine critical aspects such as ease of understanding the color-coded analysis results (green-red-black), efficiency of the JSON file upload process, and users' confidence level in interpreting the insights generated by the system. In addition to the ten standard SUS questions, 7 additional questions were added to evaluate users’ understanding of the patterns identified by the system – including one open-ended question. These additions enabled more personalized measurement tailored to the unique goals of our system.

The complete SUS questionnaire used in this study, including the additional questions specific to pattern understanding, can be found in the appendix.

### **3.2.2 Testing Methodology**

The questionnaire was specifically designed for our system and intended to test users' ability to perform the system's main task: uploading a JSON file, selecting the relevant mood field ("Parkinson's State"), and analyzing the impact of nutrition on the condition. This task was chosen because it represents the system's functionality and allows comprehensive evaluation of the user experience from the beginning of the process to receiving insights.

In addition to the standard SUS questionnaire, we added an additional section that specifically examines understanding of patterns that the system identifies. This part includes 7 additional questions focusing on result clarity, pattern identification effectiveness, and the difficulty level in understanding the presented insights. The questionnaire also includes an open-ended question that allows users to describe in their own words the insights they understood from the identified patterns, providing important qualitative information about the system's effectiveness in conveying messages.

In order to obtain a diverse and representative evaluation of the system's usability, we distributed the questionnaire among diverse populations. The diversity included different age groups (from young to elderly), different levels of technological literacy (from basic to advanced users), and diverse areas of interest. A total of 24 participants completed the questionnaire, with 8 participants aged 20-35, 10 participants aged 36-55, and 6 participants aged 56-70. Regarding technological literacy levels, 7 participants self-rated as having basic technological skills, 12 participants rated themselves as intermediate users, and 5 participants identified as advanced users.

This diverse sample supports the goal of obtaining broad, representative feedback across multiple demographics. The rationale behind this choice is that a system for analyzing personal health data should be accessible to a broad audience and not rely on a high level of technical knowledge. It is important that the system be usable both for young people and for older people, both for users with technological background and for users who are less familiar with technology. This diversity allows comprehensive evaluation of the system's accessibility and identifies usability issues that might interfere with different user groups.

### **3.2.3 Usability Questionnaire Results**

The usability questionnaire results showed excellent system performance in all examined parameters. The system received high ratings in all usability categories, indicating that it provides a quality and accessible user experience. The average SUS score achieved was 82.3 out of 100, significantly above the industry average of 68 points and exceeding the 80-point threshold that defines excellent usability. All 24 participants successfully completed the task with a 100% success rate. Completion times were reasonable, averaging 4.2 minutes per participant, with 87% of participants completing the task in under 5 minutes.

Particularly positive results were observed in the high success rate across all age groups and technological literacy levels. The success rate was 100% across all groups, including participants who rated themselves as having a basic level of technological literacy. This indicates that the goal of creating a system accessible to a broad audience was successfully achieved. Participants noted particularly high performance in ease of use, interface clarity, and integration of various functions.

In the additional part of the questionnaire, which examined understanding of behavioral patterns, the results were particularly positive. 92% of participants noted that the identified patterns are clear and understandable, and 96% agreed that presenting results through the color system significantly facilitates understanding of positive and negative effects. The insights that participants described showed 89% correct understanding of the patterns identified in the system.

One area for future improvement was identified when 33% of participants noted that data processing time was perceived as too long, although it did not prevent them from completing the task.

The usability questionnaire results confirm the success in developing an accessible and efficient system for identifying patterns in personal health data. The SUS score of 82.3 is considered excellent according to industry standards and places the system in the top 10% of evaluated systems. The high usability level among diverse populations indicates that the system achieves the goal of being a practical tool for chronic patients regardless of their technical knowledge level.

Summary table of questionnaire averages:

|  |  |
| --- | --- |
|  | Average Score |
| Age | 43.83 |
| Time to Complete (min) | 4.2 |
| SUS\_Q1 | 4.17 |
| SUS\_Q2 | 1.67 |
| SUS\_Q3 | 4.5 |
| SUS\_Q4 | 1.79 |
| SUS\_Q5 | 4.12 |
| SUS\_Q6 | 1.88 |
| SUS\_Q7 | 4.12 |
| SUS\_Q8 | 1.62 |
| SUS\_Q9 | 4.33 |
| SUS\_Q10 | 1.75 |
| Pattern\_Q1 | 4.25 |
| Pattern\_Q2 | 4.92 |
| Pattern\_Q3 | 2.29 |
| Pattern\_Q4 | 2.33 |
| Pattern\_Q5 | 4 |
| Pattern\_Q6 | 3.83 |
| Pattern\_Q7 | 4.96 |

**Full results and raw data can be found in the links:**

[Full Questionnaire Data (Excel)](https://github.com/YuvalShekel1/ParkSmart/blob/main/sus_questionnaire_responses_parkinson.xlsx) – includes all user responses in structured format

[SUS and Custom Questionnaire Forms](https://github.com/YuvalShekel1/ParkSmart/blob/main/system_usability_scale%5B1%5D.docx) – forms filled by users

**Examples of open-ended responses from the user questionnaires:**

-Did you have any questions during use?

" תוך כמה זמן לוקח לקובץ לעלות?"

-What insights did you understand from the identified behavior patterns?

" שמתי לב שמזונות עשירים בחלבון עוזרים לשפר את מצב הפרקינסון."

"הבנתי שמאכלים מתוקים כמו ריבת פירות ופודינג סויה משפיעים לרעה על מצב פרקינסון.""

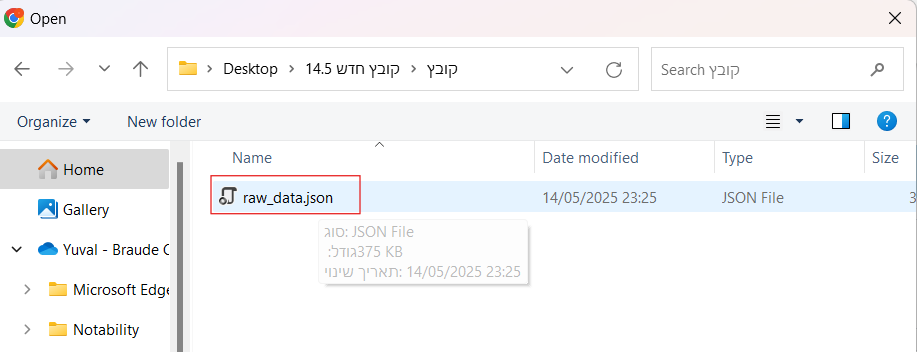
# 4. User Guide

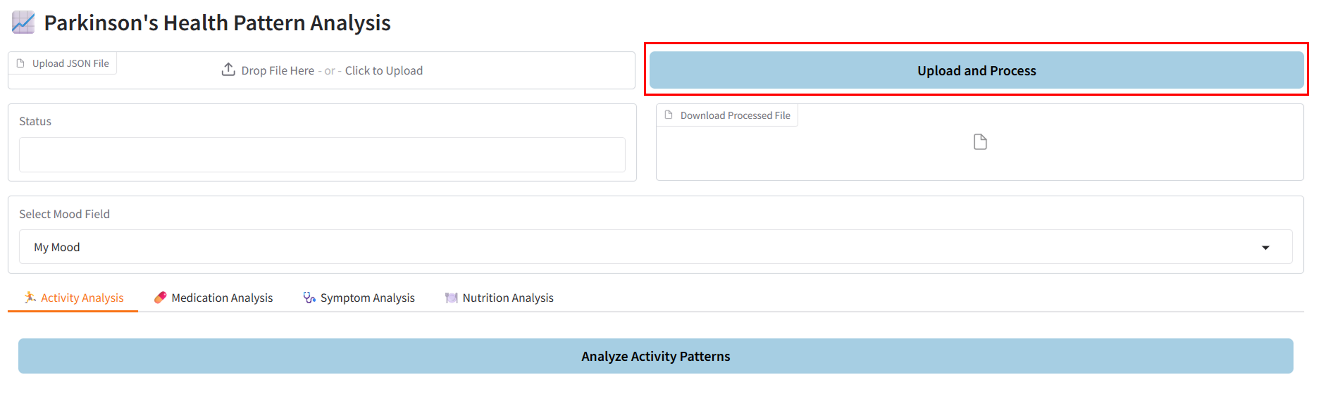
**Introduction**  
The Parkinson's Pattern Analysis System helps individuals with Parkinson's disease and their caregivers identify patterns between lifestyle factors and changes in well-being and health indicators.  
The system allows you to upload a personal data file (JSON format), which includes daily reports on: Feelings, Symptoms, Activities performed, Medications taken and Nutrition  
By analyzing the data, the system provides insights into how activities, medications, nutrition, and symptoms affect personal well-being over time.  
These insights are presented clearly and simply, helping users understand which habits and factors have a positive, negative or no impact on their condition.

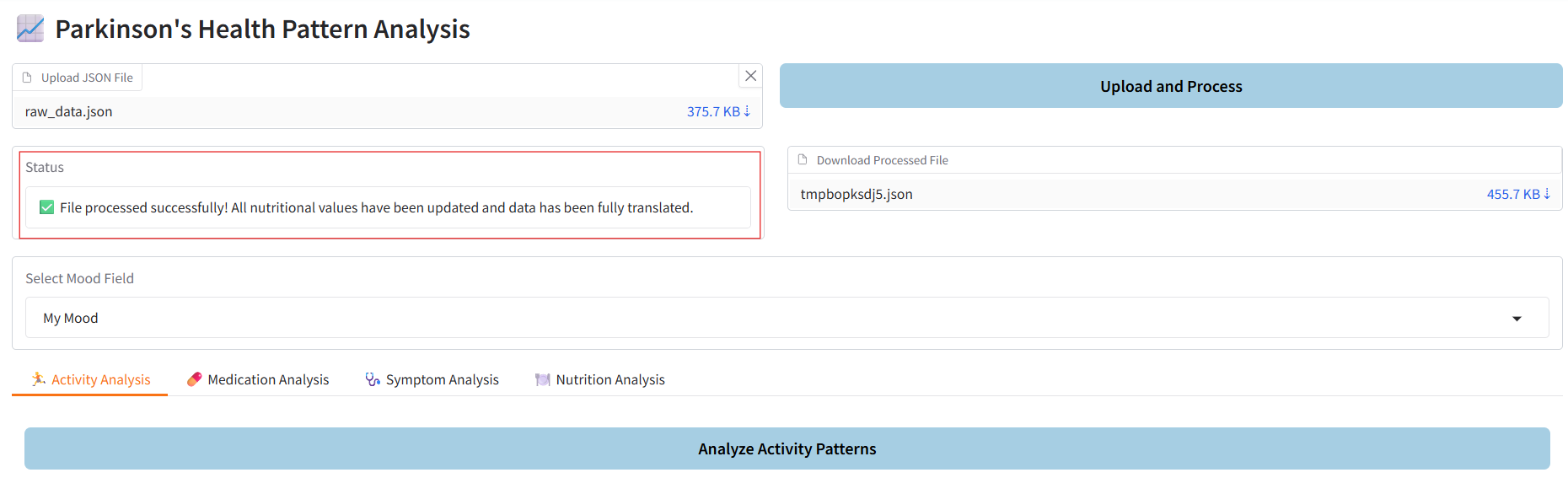
**Uploading a Data File**  
1. Click Upload JSON File.

תמונה שמכילה טקסט, צילום מסך, קו, גופן

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.  
2. Select your data file from your computer.

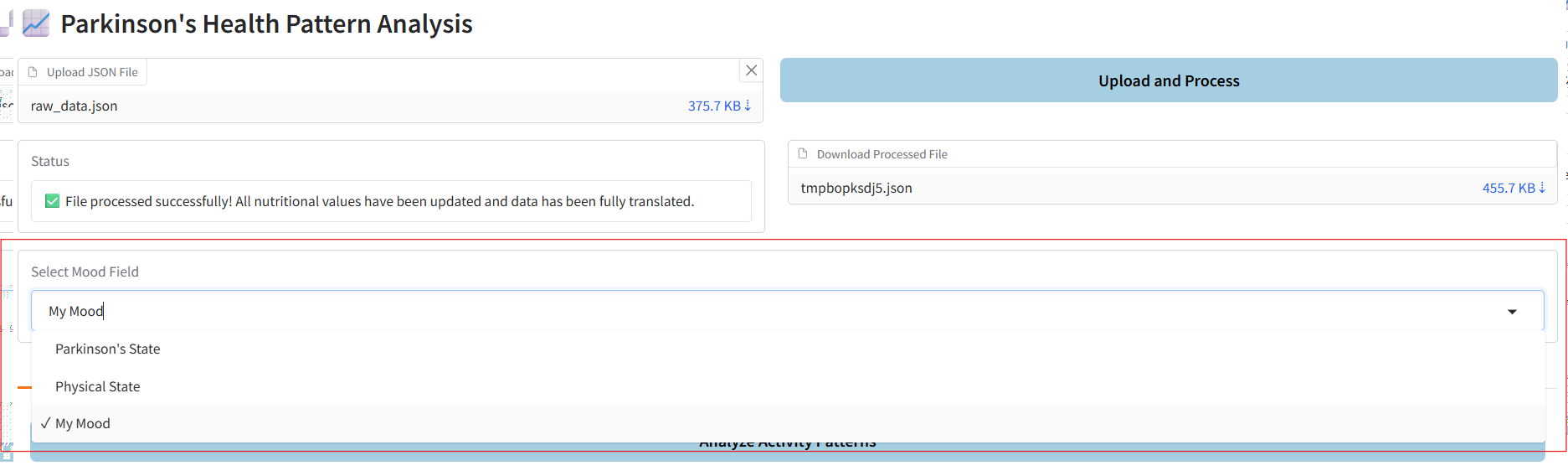
  
3. Click Upload and Process to begin processing the file.

  
4. After processing is complete, a status message will indicate success.

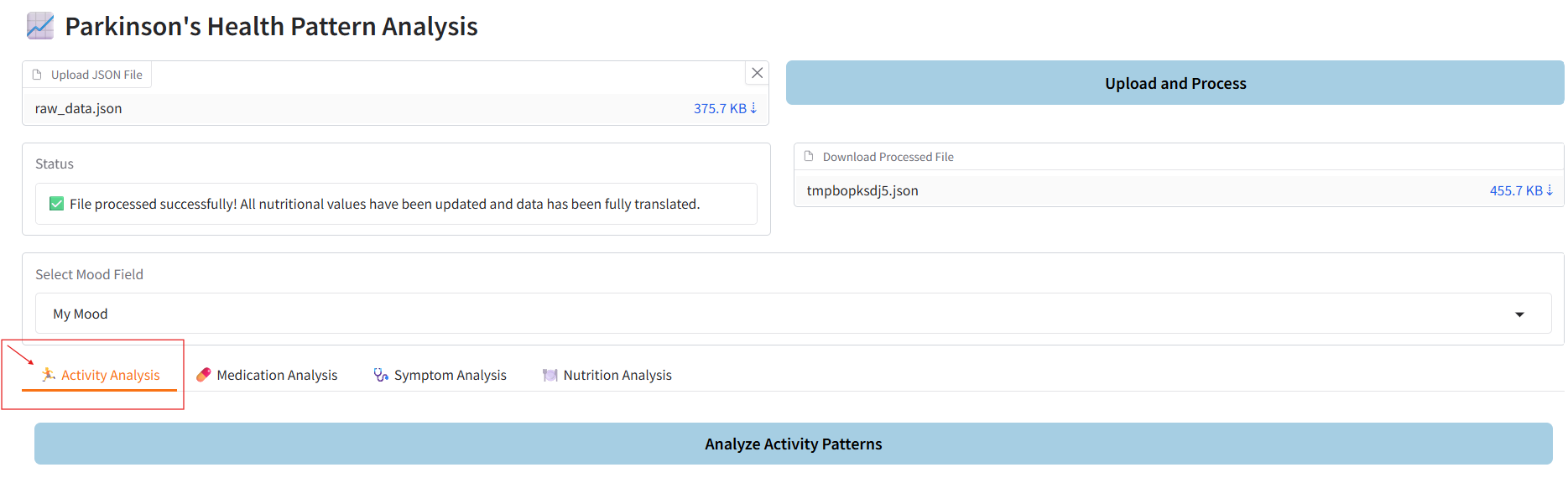
  
5. You can download the processed file by clicking Download Processed File.  
תמונה שמכילה טקסט, גופן, קו, מספר

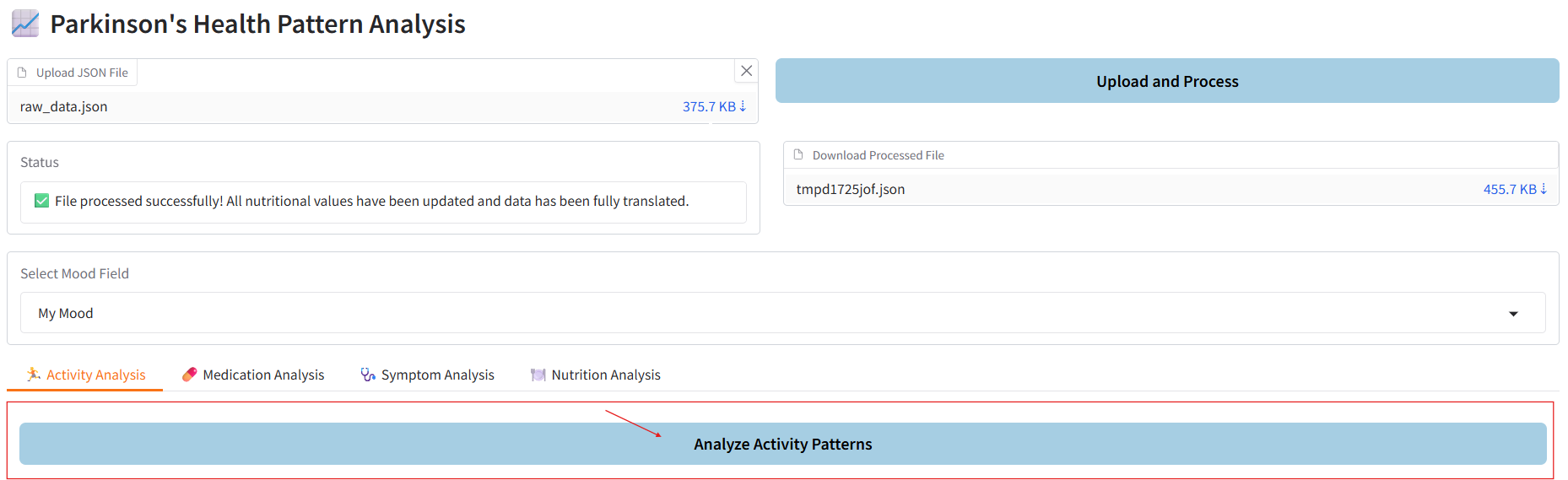
תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.

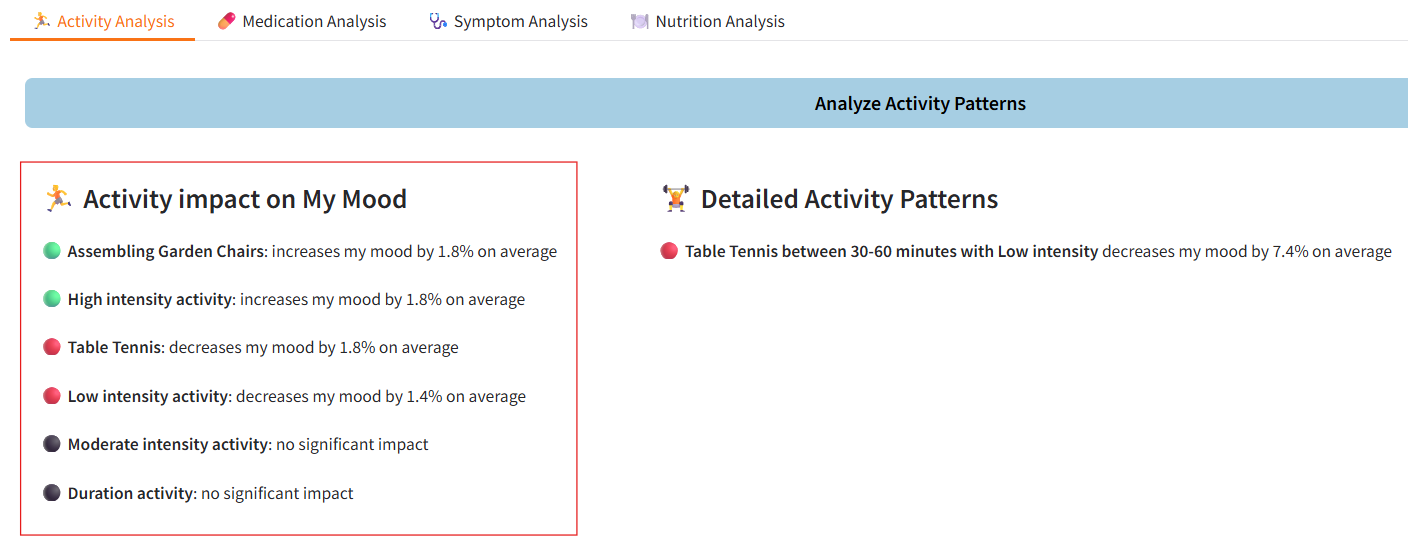
**Selecting a Mood Field**  
 Choose the mood field you wish to analyze from the Select Mood Field dropdown.  
 Options:  
 - Parkinson's State  
 - Physical State  
 - My Mood



**Activity Analysis**  
1. Go to the Activity Analysis tab.

  
2. Click Analyze Activity Patterns.

  
3. Insights will be displayed showing how different activities affect the selected mood field.

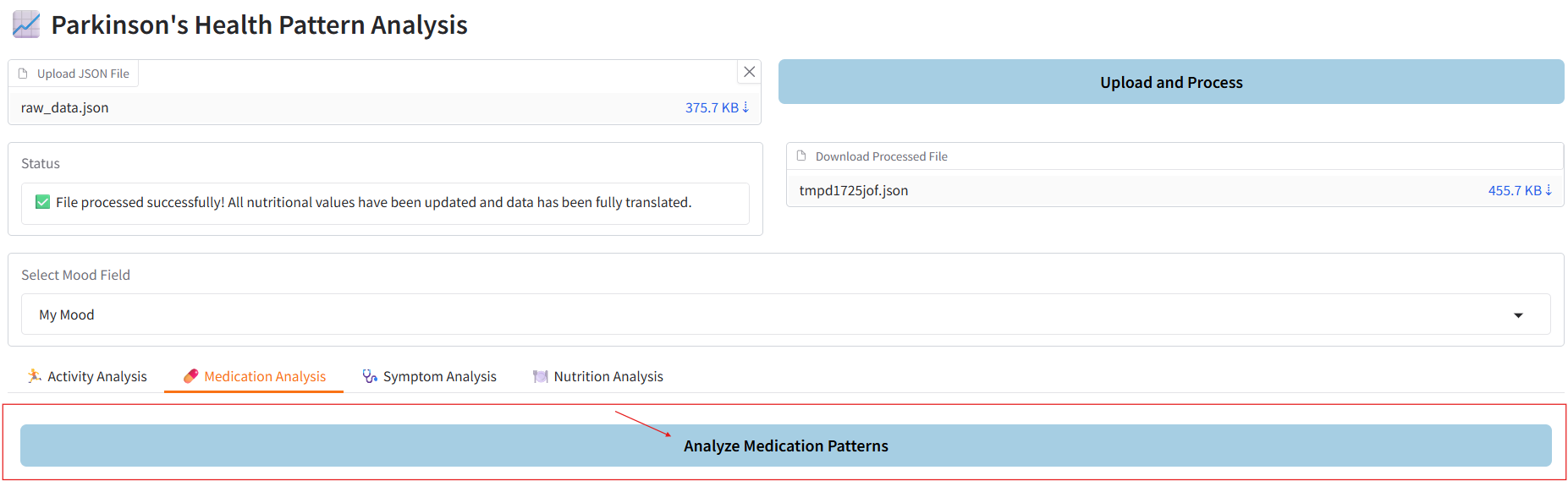
  
4. The analysis includes:  
 - Overall impact of activity types.  
 - Impact based on activity duration.  
 - Impact based on activity intensity.  
 - Combined effects of duration and intensity.  
תמונה שמכילה טקסט, צילום מסך, גופן, דף אינטרנט

תוכן שנוצר על-ידי בינה מלאכותית עשוי להיות שגוי.

**Medication Analysis**  
1. Go to the Medication Analysis tab.

תמונה שמכילה טקסט, צילום מסך, גופן, קו

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.  
2. Click Analyze Medication Patterns.

  
3. Insights will be displayed showing how different medications affect the selected mood field.

תמונה שמכילה טקסט, צילום מסך, תוכנה, דף אינטרנט

תוכן שנוצר על-ידי בינה מלאכותית עשוי להיות שגוי.  
4. The analysis includes:  
 - Impact of individual medications.  
 - Impact based on time window after taking the medication.  
 - Impact of medication sequences.

תמונה שמכילה טקסט, צילום מסך, גופן, תוכנה

תוכן שנוצר על-ידי בינה מלאכותית עשוי להיות שגוי.

**Symptom Analysis**  
1. Go to the Symptom Analysis tab.

תמונה שמכילה טקסט, גופן, קו, מספר

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.  
2. Click Analyze Symptom Patterns.

תמונה שמכילה טקסט, גופן, מספר, קו

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.  
3. Insights will be displayed showing how various symptoms affect the selected mood field.

תמונה שמכילה טקסט, צילום מסך, דף אינטרנט, גופן

תוכן שנוצר על-ידי בינה מלאכותית עשוי להיות שגוי.

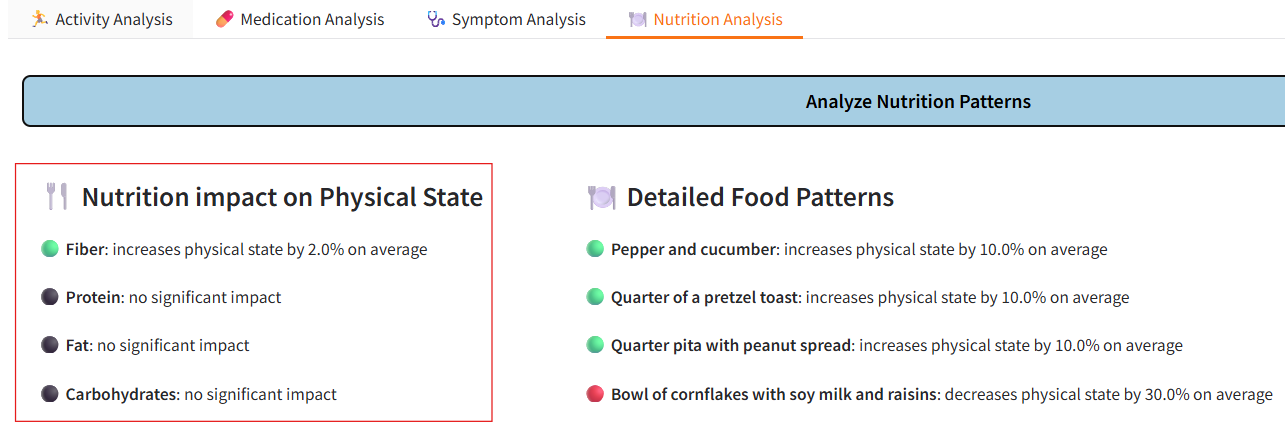
**Nutrition Analysis**  
1. Go to the Nutrition Analysis tab.

תמונה שמכילה טקסט, גופן, קו, מספר

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.  
2. Click Analyze Nutrition Patterns.

תמונה שמכילה טקסט, גופן, קו, מספר

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.  
3. Insights will be displayed showing how nutritional components (protein, fat, carbohydrates, fiber) affect the selected mood field.

  
4. Insights will also show the impact of specific foods.  
תמונה שמכילה טקסט, צילום מסך, גופן, תוכנה

תוכן שנוצר על-ידי בינה מלאכותית עשוי להיות שגוי.

# 5. Maintenance Guide

This guide provides maintenance instructions for the ParkSmart system for analyzing patterns in Parkinson's patients, intended for developers and technical maintenance personnel.

## 5.1 System Requirements and Setup

### **5.1.1 Basic Requirements**

The system requires Python 3.8 or higher, with required libraries including Gradio, Pandas, NumPy, Scikit-learn, Translatepy, and Requests. Stable internet connection is required for access to the USDA FoodData Central API. The system also requires pip for package management. All dependencies can be installed using a requirements.txt file.

### **5.1.2 Work Environment Setup**

The system is deployed and maintained via the Render cloud platform. The project is synchronized with GitHub, and all required dependencies are automatically installed using the requirements.txt file. The main application file is app.py, which contains the system's logic, interface, and routing, and is automatically executed by the Render platform. An environment variable named USDA\_API\_KEY must be configured to enable access to nutritional data from the USDA FoodData Central API.

## 5.2 Deployment and Operation

### **5.2.1 Local Operation**

The system can be operated locally by running the app.py file. By default, it is accessible through a web browser at http://localhost:5000, unless a different port is specified using the PORT environment variable. This mode is intended for development and testing purposes.

### **5.2.2 Cloud Deployment**

The system is adapted for deployment on the Render platform with automatic synchronization to GitHub. Every code update triggers automatic system update, but requires manual server activation after the update.

## 5.3 Data Management and Caching

### **5.3.1 Data File Structure**

The system accepts a JSON file containing five main categories: nutrition, activities, medications, symptoms, and feelings. Each category is organized with timestamps and relevant data.

### **5.3.2 Caching System**

The system uses temporary in-memory caching during runtime. Hebrew expressions that are translated dynamically using the translatepy library are stored in a runtime cache to avoid translating the same terms multiple times. Similarly, nutritional data retrieved from the USDA API is cached temporarily to prevent redundant external requests. These caches are cleared upon system restart, as they are not saved to permanent storage.

## 5.4 System Modules

### **5.4.1 Data Processing Module**

This module handles the preprocessing of uploaded JSON files.

It translates selected Hebrew fields to English using a layered approach: first checking a predefined local dictionary, then a runtime cache that updates during execution, and finally using an automatic translation service powered by the translatepy library if needed.

Similarly, nutritional data is enriched using a local nutrition dictionary, a temporary cache, and queries to the USDA API when necessary.

The module also standardizes timestamps and performs basic validation to ensure consistent data structure and types.

The module also supports downloading the translated and enriched JSON file after processing.

### **5.4.2 Pattern Analysis Module**

This module performs statistical analysis using linear regression algorithm. It includes tunable parameters such as minimum number of observations per pattern and statistical significance threshold.

### **5.4.3 User Interface Module**

This module is based on Gradio and provides an accessible interface for users. It includes accessibility adaptations and an intuitive color system.

## 5.5 Common Troubleshooting

### **5.5.1 Performance Issues**

Long processing time may result from external delays (such as the translation service or the USDA FoodData API) or from uploading large and complex JSON files. Improving internet connectivity or simplifying the uploaded file may help reduce processing time.

### **5.5.2 Translation and Nutrition Data Issues**

Missing translations are handled using a layered approach. The system first checks a predefined translation dictionary, then a runtime cache of previously translated terms, and finally uses an automatic translation service if needed. To correct translation issues, the predefined dictionary can be manually updated in the code.

Nutritional data issues may arise if the USDA API key is missing or invalid, or if a food item is not found in the USDA database. In such cases, the food will be skipped and no nutritional values will be added. To resolve this, verify the API key in the environment settings and, if needed, update the local nutrition dictionary in the code with the relevant food data.

### **5.5.3 Deployment Issues**

Deployment issues on Render may occur due to missing environment variables (such as the USDA API key), incorrect configuration, or missing dependencies in the requirements.txt file.

To resolve such issues, verify that all required libraries are listed, ensure that the environment variable is properly set, and restart the service if needed.

If the system still fails to deploy, review the Render build logs for error messages.

## 5.6 Backup and Recovery

### **5.6.1 Backup Procedures**

The system does not persistently store cache or configuration files. To ensure operational continuity, it is recommended to back up the codebase and the predefined translation dictionary via GitHub. This provides a reliable recovery option in case of data loss or deployment failure.

### **5.6.2 Failure Recovery**

In case of failure, the system can be restored from the latest version in Git.

## 5.7 Future Planning

### **5.7.1 Development Areas**

Future planning includes adding more advanced machine learning algorithms, improving the interface with visual graphs, developing an alert and recommendation system, and supporting additional chronic diseases.

### **5.7.2 Update Procedures**

Every update requires preliminary testing and system validation after the update to ensure proper operation.

# 6. Conclusions and Future Work

## 6.1 Evaluation of Project Goal Achievement

The project successfully achieved most of the goals defined in the planning stage. The central goal - creating a system for identifying and displaying patterns from Parkinson's patient data - was fully realized. The system successfully processes complex and multilingual data, identifies significant patterns in four main categories (nutrition, activity, medications, and symptoms), and displays them in a user-friendly interface.

The system demonstrated high reliability in trend predictions and ability to identify significant patterns in categories with sufficient data. Quality control mechanisms, including requiring a minimum of 3 observations per pattern and statistical significance thresholds, ensured that displayed patterns are reliable and have practical value.

The system successfully adapted itself to the specific needs of the target user. The emphasis on nutritional analysis, his explicit request, was fully implemented with development of a detailed module for analyzing the impact of nutritional components and specific foods.

## 6.2 Lessons Learned from the Project

The ParkSmart project led to several key insights.

A major technical lesson was the importance of choosing the right development platform: transitioning from Streamlit to Gradio significantly improved system responsiveness and user experience. This reinforces the value of evaluating platforms early to ensure optimal performance and usability.

In terms of data processing, we encountered the challenges of handling multilingual content and learned the importance of building layered mechanisms for translation and data enrichment. This highlights the need for adaptable data handling approaches in systems designed for real-world input.

The choice of a linear regression algorithm was found to be appropriate for the system's needs, as its transparency and ease of interpretation supported user trust and comprehension, even if more complex models could offer marginally better accuracy. This suggests that model explainability is often more valuable than maximum precision in user-facing applications.

From a usability perspective, we found that combining a clear color scheme (green, red, black) with a minimum threshold of three data points per pattern helps users interpret insights with confidence while maintaining statistical credibility. This approach proved especially effective for users with limited technical background, and can serve as a guideline for designing intuitive and trustworthy health-related data systems.

The high SUS score of 82.3 demonstrated that focusing on clarity, accessibility, and visual simplicity leads to broad acceptance among diverse users. This confirms that accessibility should be a design priority in health-related digital tools.

Finally, we learned that by designing the system to be modular and scalable, it can be adapted to support other chronic conditions beyond Parkinson’s disease — a direction that holds promise for future expansion. This proves the long-term value of investing in flexible and generalizable system architecture.

## 6.3 Recommendations for Future Development

The next logical step is developing a recommendation system based on identified patterns. Such a system could recommend dietary changes based on identified positive and negative patterns, suggest optimal timing for medication administration, advise on types of activity and recommended durations, and identify developing symptoms and recommend consulting a physician.

Additional system development could include expanding it to additional diseases. As noted, the system can also adapt to identifying patterns in other chronic diseases such as diabetes, cardiovascular diseases, and other chronic diseases requiring continuous personal monitoring.

Additionally, the user experience in the system can be developed in several key directions: adding graphs and visualizations that illustrate patterns to the user in a visual and clear way, creating an alert notification system for unusual or interesting patterns, and developing the ability to export detailed reports that can be presented to medical teams.

## 6.4 Potential Impact and Future Applications

The system represents a significant step toward personalized medicine care. The ability to identify unique patterns for each patient and generate tailored insights can change the way chronic diseases are treated. Instead of general treatment given to everyone, patients will be able to receive data-based recommendations specific to their personal condition.

The system can integrate with existing health systems and become a standard tool for monitoring chronic patients. This can lead to significant improvement in treatment quality and reduction of healthcare costs through better prevention of deteriorations and hospitalizations.

## 6.5 Conclusions

This project succeeded in creating a strong foundation for a pattern analysis system in personal health data. The combination of advanced technologies with focus on specific user needs created a tool with great potential for improving the quality of life of chronic patients. The project's success demonstrates the possibility of developing personalized technological solutions that can serve as a foundation for the next generation of digital health tools.

The path for continued system development is clear and challenging, with potential for significant impact on the field of personalized medicine. The achievements accomplished in this project constitute an important step toward a future where every patient will be able to receive data-based insights about their health condition and improve their quality of life through informed decisions.

# 7. Appendix

## 7.1 System Usability Scale (SUS) – Parkinson's Pattern Identifying

**Participant’s Information:**

Name (optional): \_\_\_\_\_\_\_\_\_\_\_\_

Age: \_\_\_\_\_\_\_\_\_\_\_\_

Gender:

◯ Male

◯ Female

◯ Other

Technology proficiency (please select one):

◯ Basic (uses the internet occasionally)

◯ Intermediate (regular user of online tools)

◯ Advanced (familiar with programming or data analysis)

Have you used a data analysis tool before?

◯ Yes

◯ No

**Test Scenario**

You receive a JSON file containing daily data recorded by a person with Parkinson's disease.  
The data includes: Feelings, Symptoms, Activities, Medications, and Nutrition.  
Your task is to upload the file to the system, select "Parkinson's State" as the mood field, and analyze the impact of Nutrition on the Parkinson's state.  
Finally, review the insights generated by the system.

**Post-Task Feedback**

Time to complete the task: \_\_\_\_\_\_\_\_\_\_\_\_ minutes

Were you able to complete the task?

◯ Yes

◯ No

If not, what was difficult? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Did you have any questions during use?

◯ Yes

◯ No

If yes, please specify: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

What insights did you understand from the identified behavior patterns?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

**System Usability Scale (SUS)**

**Section 1: System Evaluation**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | |  | Strongly Agree |  |  |  | Strongly Disagree | |  | 1 | 2 | 3 | 4 | 5 | | I think that I would like to use the system frequently. |  |  |  |  |  | | I find the system unnecessarily complex. |  |  |  |  |  | | I think the system is easy to use. |  |  |  |  |  | | I need the support of a technical person to use the system. |  |  |  |  |  | | I find that the various functions in the system are well integrated. |  |  |  |  |  | | I think there is too much inconsistency in the system. |  |  |  |  |  | | I think that most people would easily learn how to use the system. |  |  |  |  |  | | I find the system cumbersome to use. |  |  |  |  |  | | I feel very confident using the system. |  |  |  |  |  | | I need to learn a lot of things before I can get going with the system. |  |  |  |  |  | |

**Section 2: Understanding of Identified Behavior Patterns**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Strongly Agree |  |  |  | Strongly Disagree |
|  | 1 | 2 | 3 | 4 | 5 |
| I think that the identified behavior patterns are helpful and informative. |  |  |  |  |  |
| The identified behavior patterns are clear and understandable. |  |  |  |  |  |
| I often feel frustrated when trying to understand the behavior patterns. |  |  |  |  |  |
| Understanding the identified behavior patterns requires considerable mental effort |  |  |  |  |  |
| I think that the behavior pattern identification is effective in helping me understand the data |  |  |  |  |  |
| I imagine that most people would easily understand the identified behavior patterns. |  |  |  |  |  |
| |  | | --- | | The color system (green = positive, red = negative, black = neutral) helped me understand the effects of different behaviors. | |  |  |  |  |  |

# References

**[1]** Postuma, R. B., Berg, D., Stern, M., Poewe, W., Olanow, C. W., Oertel, W., ... & Deuschl, G. (2015). MDS clinical diagnostic criteria for Parkinson's disease. Movement Disorders, 30(12), 1591-1601. <https://doi.org/10.1002/mds.26424>

**[2]** Chaudhuri, K. R., Healy, D. G., & Schapira, A. H. (2006). Non-motor symptoms of Parkinson's disease: diagnosis and management. The Lancet Neurology, 5(3), 235-245. <https://doi.org/10.1016/S1474-4422(06)70373-8>

**[3]** Martinez-Martin, P., Rodriguez-Blazquez, C., Kurtis, M. M., & Chaudhuri, K. R. (2011). The impact of non-motor symptoms on health-related quality of life of patients with Parkinson's disease. Movement Disorders, 26(3), 399-406. <https://doi.org/10.1002/mds.23462>

**[4]** Titova, N., & Chaudhuri, K. R. (2017). Personalized medicine in Parkinson's disease: Time to be precise. Movement Disorders, 32(8), 1147-1154. <https://doi.org/10.1002/mds.27027>

**[5]** Espay, A. J., Bonato, P., Nahab, F. B., Maetzler, W., Dean, J. M., Klucken, J., ... & Movement Disorder Society Task Force on Technology. (2016). Technology in Parkinson's disease: Challenges and opportunities. Movement Disorders, 31(9), 1272-1282. <https://doi.org/10.1002/mds.26642>

**[6]** Steinhubl, S. R., Muse, E. D., & Topol, E. J. (2015). The emerging field of mobile health. Science Translational Medicine, 7(283), 283rv3. <https://doi.org/10.1126/scitranslmed.aaa3487>

**[7]** IBM. (2021, August 18). Linear regression. IBM Think Topics. <https://www.ibm.com/think/topics/linear-regression>

**[8]** Git. (n.d.). Git. <https://git-scm.com/>

**[9]** Srinandh. (n.d.). Gradio: A Python-based framework for machine learning enthusiasts. Medium. <https://medium.com/@srinandh28/gradio-a-python-based-framework-for-machine-learning-enthusiasts-3c617f06bae6>

**[10]** QuestionPro. (n.d.). System usability scale. QuestionPro Blog. <https://www.questionpro.com/blog/system-usability-scale/>