Precision Ketogenic Therapy (PKT) Intake Calculation App

Tyler C. Mueller

CIS4914, Senior Project
Department of CISE
University of Florida

Advisor: Dr. Peggy L. Borum, *email:* prb@ufl.edu
Department of Neurology
University of Florida, Gainesville, FL 32611

Date of Report: 29 Nov 2021

Abstract

The Precision Ketogenic Therapy (PKT) project, as outlined by the UF Division of Pediatric Neurology, is a program designed to counteract seizures in young seizure patients through the use of diet. By controlling the meal intake of a patient via a strictly-monitored and precisely-manipulated ketogenic diet, the PKT program aims to improve the quality of life of seizure patients. Over the long life of the program, there have been tangible cases of success, with patients reporting less severe and less frequent seizures than they experienced prior to being placed under diets established by PKT. The current issue with the program are outdated methods that have been hampering the program's efficacy. For most of the PKT project's life span, Microsoft Excel sheets have been used, which have begun to become slow in terms of querying individual tables and columns given the massive amount of data stored and utilized by the program. Additionally, clinical staff associated with PKT have primarily been performing calculations of a patient's intake by hand, another area in need of automation.

The primary goal of this project is the production of a Web application for the purpose of assisting the PKT team with a transition from Excel sheets and hand-made calculations to a more automated system. This Web application is a multi-component project being developed by several CISE students in parallel, and the component being focused on for this project is the diet record and intake calculation systems. Diet record refers to the meals a patient has been eating in relation to PKT recipes and the completeness of their consumption of the meal, while intake calculation refers to the daily and average intake of a patient over a period of time based on their diet record. This application is built using Python and the Flask framework, with PostgreSQL as the database software. Development and maintenance of the application is expected to be continued after the project's completion by other students, but the objective is to provide something functional and easily usable by PKT clinical staff.

1. Introduction

The segment of the PKT application that encompasses the diet record and intake calculator is a critical component of the greater app as a whole, by building on the frameworks provided by previous PKT contributors, designing and implementing new HTML interfaces for interacting with the program, and making connections and schema between the diet record and intake calculator segments of the PKT database and the database schema of other contributors. This application is designed to operate within UF's ResVault data protection service, due to the

HIPAA-protected nature of the clinical data that will be deployed to the application, and as such is tested on local and virtualized environments using WSL and Ubuntu. Though this project is considered a solo project, it is connected to and references components developed by other CISE students currently working on the PKT application.

The benefits of this project are immediate in that creating a more efficient system for PKT calculations is providing a useful service that helps to assist young seizure patients in managing their seizures. In addition, working on this project provides perspective on how medicine and computer science interact, which is a subject of personal interest. Additionally, there is the implicit benefit of furthering personal understanding of software development and how to build effective apps using the Python language.

Currently, the project is projected to have a functional demonstration of use on the ResVault service around December 2, though some extra tweaks may be made during the interim between ResVault's scheduled maintenance and the final deliverable. Current problems being worked towards are building an interface consistent with other users and testing full integration with the database schemas of other users, but these are predicted to be resolved by the time of the demonstration on December 2.

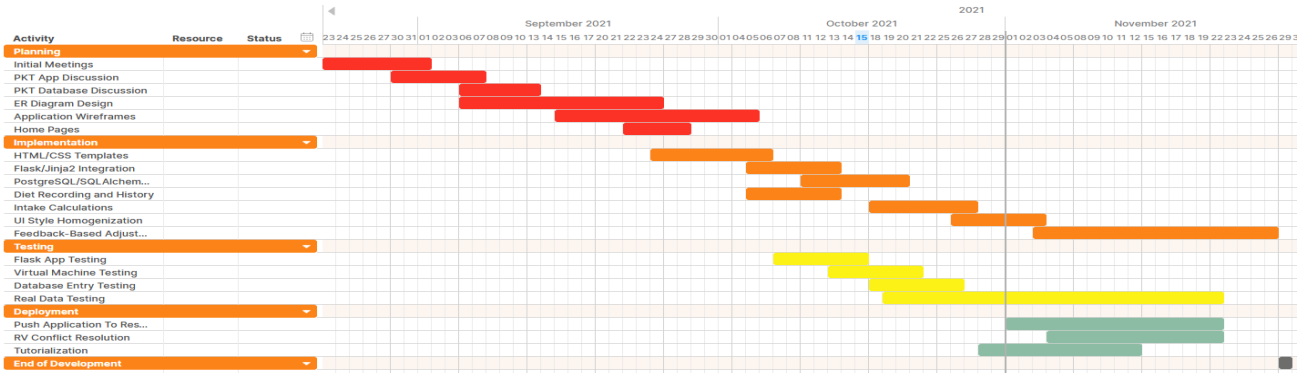

Figure 1. Gantt chart depicting the projected schedule of development for the diet record and intake calculation segments of the PKT application, ranging from August to November.

1.1. Problem Domain

The main problem of CISE students working on the PKT application is software development and applying existing software solutions in a meaningful way towards the development of a user interface for clinical staff. By undertaking this project, students are acting to design a Web application in accordance with software design principles. The project's target audience are clinical staff members who, while knowledgeable in the field of pediatric

neurology, may not be as savvy with database systems and thus would have use for a user interface and application that allows them to interact with and perform calculations using stored data. The benefit of having a dedicated Web application will also allow further CISE students with interest in working with PKT a library of past work to be able to review, maintain, modify, and build upon as they expand the application further and more data is introduced to the system.

1.2. Literature Review

For the greater part of a century, ketogenic diets have been used to manage seizure-causing disorders such as epilepsy. While the exact mechanism of ketogenic diets' effect on the brain that prevents seizures is poorly understood, such programs have demonstrated success in minimizing the frequency and intensity of seizures, such as a trial run by University College London from 2001 to 2006.[1] The results of this study found that young patients undergoing a ketogenic diet had a median seizure percentage of slightly above half of the control group's seizure percentage, demonstrating a tangible success in reducing seizure risk. Typical ketogenic programs for seizure management tend to be discontinued from treatment after two to three years, however, the Precision Ketogenic Therapy program at UF aims for long-term support of patients. PKT is a system of personalized treatment plans tailored to a patient's needs, with customized recipes prepared for each patient. The heart of the program is the Foodomics Database, a collection of foods used to assemble PKT meals with adequate amounts of nutritional information supplied for each ingredient in the database [2, 3].

The main issue currently encountered by the PKT staff is one of utility. For storing diet information and the Foodomics Database, Microsoft Excel has been utilized [3]. While Microsoft Excel is a useful program, it begins to become untenable in terms of speed when large amounts of rows, such as the amount necessary for Foodomics, are being queried through in Excel [4]. The slow speed of Excel queries have become a point of consternation for PKT staff members. The use of relational databases, such as PostgreSQL, would achieve a similar effect to the extant Excel spreadsheets while also being more optimized for both handling large amounts of data and filtering out any entries that are not necessary for a query, therefore avoiding the slowdown has has affected operations. Additionally, certain calculations, such as for recipes, are done through Excel [3] and could be expedited through a programmatic solution designed specifically to handle these calculations using both Postgres queries and Python. Python offers advantages over

Excel when it comes to the kind of robust calculations necessary for PKT, in addition to being able to handle large amounts of data with more ease than Excel [5].

2. Technical Approach

The main solution to the need for a system of diet record and intake calculation comes in the form of a multi-page Web application integrated into a central application. This Web application is designed to allow clinical staff to input diet information retrieved from patients, and use this information to determine the daily, total, and average intake of proteins, fats, carbohydrates, and calories over an arbitrary period of time, and compare their average values to the patient's diet prescription for each of the main four items being tracked. Additionally, we must account for the quality of reports based on how much estimation and inference is performed by a clinical staff.

The CISE student body working on the PKT application have utilized a Python-based solution to connect a set of Web templates to databases for this application's main functionality. The primary database software used in the production and deployment of the application is PostgreSQL, an improved variant of the SQL database language, which is integrated with the main Python application via the SQLAlchemy package. The diet record and intake calculator systems do not deviate from the standard established in this regard. For displaying content, HTML templates with fields populated using Flask's built-in Jinja2 package are utilized. Each individual component of the project has its own routes and Python file representing it within the PKT application, which are all managed by a central application file that connects the PostgreSQL database to our Flask session for all component files of the application.

When accessed, a clinical staff user will be brought to a home screen, prompting them to enter in a medical record number specific to their patient. This number is used to access the diet records and intake calculator reports of a patient, and is preserved for this session to help simplify the process of adding data to a patient and reviewing their personal history. From the home screen, a staff member can access pages for adding a new diet record entry, generating a new intake report, or viewing the history of each. The home screen will in the future offer an option for a data dictionary, to help new users understand the terminology and functionality of the program.

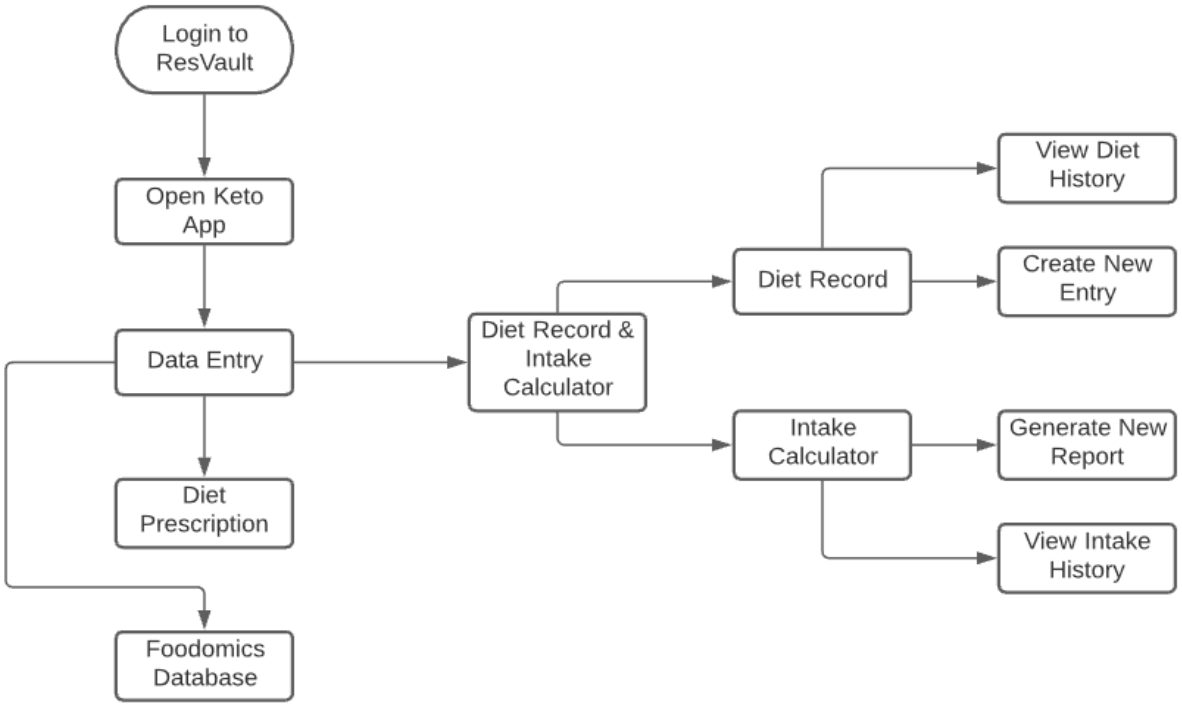

Figure 2. Flowchart depicting the flow of how a clinical staff member will access the diet record and intake calculation sections from the app in ResVault.

The diet record page will prompt a clinical staff member to input the date on which this patient has eaten something, a meal type plus additional information in the case of bolus/continuous feeding, the quality of reporting (where a value of 1 is perfect or near-perfect reporting, and a 3 is considerably incomplete reporting), the recipe ID of the meal the patient has eaten, and the percentage amount of each ingredient eaten. An immediate issue that comes up while developing was the need for dynamically modifying the HTML content of the application. A patient's meals can have any number of ingredients, and static HTML templates alone would either not be able to account for every ingredient in a meal, or clutter the space with an inelegant amount of input boxes. The solution to this issue was the use of AJAX routing to perform a call to a Flask route that would generate an appropriate amount of input boxes once a recipe ID was provided, allowing for accounting for all possibilities without requiring a page refresh. Similar AJAX routing is done to retrieve and display database entries when viewing previous diet records and intake calculator reports. AJAX routing is also used for submitting diet record and intake calculator entries, as each type has a specialized POST route that takes in the submitted

information and produces database entries depending on what is needed for the table being accessed, and commits these changes to the database.

The intake calculator section uses a provided start and end date, as well as a chosen generation style. Once an intake report is chosen for generation, the SQLAlchemy package queries the diet record table in the PKT database, retrieving all entries for our selected patient within our selected inclusive time range. Depending on the generation style, we perform two different calculations: an average report, a single report per day, or both styles simultaneously. The daily report uses Python's date structures to further pare down the retrieved information from the PostgreSQL database, while the average report is a lump sum of the total intake for proteins, fats, carbohydrates, and calories, all values averaged by the amount of days in the intake report. The daily intake from the diet prescription is also queried for and utilized, with average intake values being compared to the recommended intake to retrieve a percentage ratio.

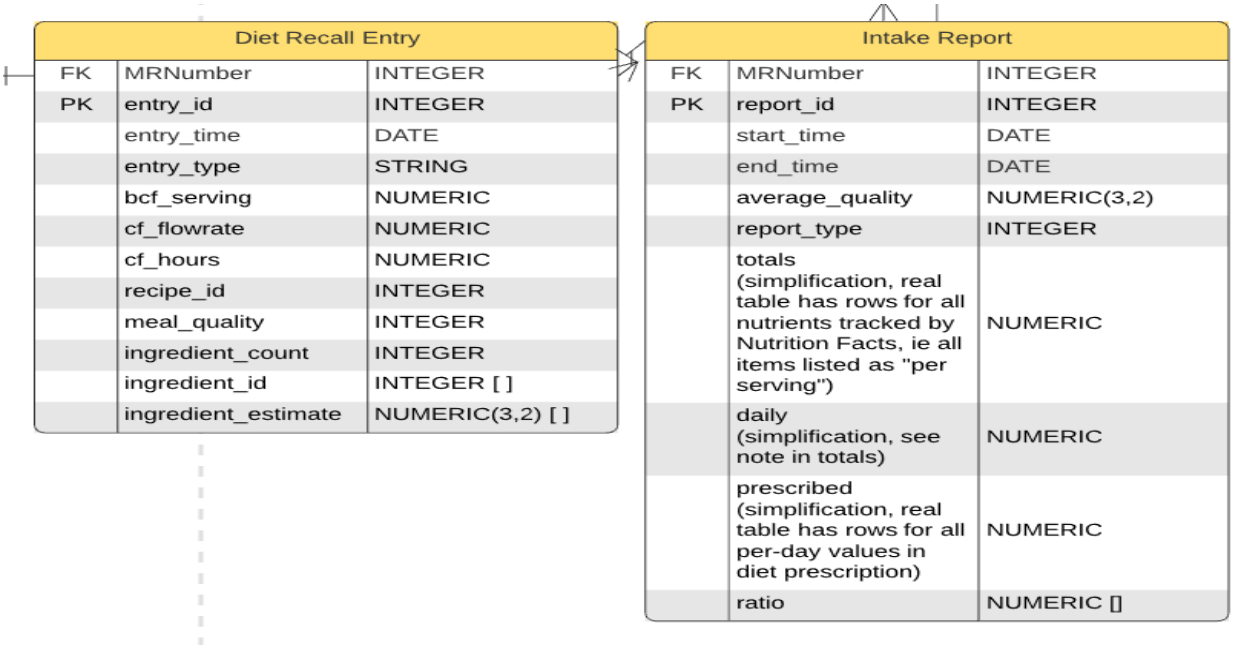

Figure 3. Old database schema used in the diet record and intake calculator. PostgreSQL arrays are utilized for ingredient and nutrient information.

The additions to the database for diet record and intake calculation are two tables, one for diet records (Figure 3, left) and intake calculator reports (Figure 3, right). Both use the medical record number as their foreign key, this key belonging to a table representing patient information. The primary key is an entry ID, which is generated by PostgreSQL upon a new entry being deployed to the database and unique to the program. Diet records store entry times and types,

with bolus servings and continuous flow rate/hours being stored and used if the type indicates boluses or flow. The recipe ID, ingredient types, counts, and estimates are stored within a diet record for use by the intake calculator, the types and estimates being arrays. Meal quality is stored as an integer. The intake report table stores the start and end dates of a calculation's range, the type of report (daily or average), the average quality stored as a float, and the total, daily average, diet prescription amount, and daily average-to-diet prescription amount ratio for each tracked nutrient within an array.

Testing primarily occurred within local and virtualized environments. The Docker software package was used for both starting up the PKT application and monitoring the output program's logging and errors. For the local environment, the Windows Subsystem for Linux (WSL) with Ubuntu was utilized for testing. The virtualized environment utilized Ubuntu 20.04 for testing, as the ResVault service is primarily accessed through virtual machines. Before the point where other PKT application developers were sharing their full database models, abstractions similar to their prototype models from the shared ER diagram were utilized in testing, and the program was accordingly refactored to adopt their database schemas once they became available.

One of the primary challenges in developing the PKT program, aside from interpersonal communication, is the visualization and implementation of how its various components interact with one another. SQLAlchemy provides a class structure for translating and interpreting PostgreSQL models to and from the primary database, which assists the implementation by minimizing the amount of effort required to make meaningful queries and updates to the database.

3. Results

Results for this project are currently qualitative instead of quantitative, centering around the PKT clinical staff's comments on the designed interface and its functionality. As the interface is still a work in progress, quantitative measures of the system's performance are to be measured after the live demonstration. In testing, the program has functioned primarily as expected, providing daily and average intake reports and storing diet records and intake reports within the database without issue. The intake calculator is able to accurately operate within selected date ranges, and provides an accurate calculation for each diet record. Tests have been run on both local and virtual environments to ensure that the program functions properly within multiple

environments, the latter environment being necessary for proper operation in the ResVault service.

Currently, there has been a positive response towards prototype interfaces from development staff as progress on the project continued. Though clinical staff have not had much chance at a hands-on experiment with the interface, demonstrations of the program operating have gone over well, and points of criticism and suggestions are being accounted for in the development of the project. One early suggestion was the inclusion of a quality grading system for diet records (where the quality of reporting clinical staff receive from patients can be measured) that could be averaged in intake calculations, which has been implemented. A main point of criticism currently being addressed during the end of development is the need for a more user-friendly interface, as the current interface is built off a wireframe.

Areas that are currently in need of improvement are the design of the interface to bring it in line with the standards established by other parallel developers, the handling of bolus and continuous feeding and how such statistics modify ingredient consumption, and the implementation of requested features such as deletion and modification of diet records from the history section. These are projected to be completed by the time of the ResVault demonstration, with an intended final completion date of December 10 if this is not feasible. Progress on the application will extend past the fall semester, and be continued by other CISE students and computer scientists who will be trained in the operation and code structure of the application.

Standards and Constraints

Standards: All code for this project has been written in Python 3.8, employing the Python Enhancement Proposal (PEP) 8 formatting standard outlined by the Python Software Foundation.
Constraints: The PKT application is intended to run under both Docker images and the ResVault service, necessitating the use of virtual machines for access to the application in the production build.

4. Acknowledgments

Acknowledgments for the project go to Dr. Peggy Borum, the head of the Precision Ketogenic Therapy program and project advisor for the diet record and intake calculator systems, for her advice and criticisms of the demonstrations given. Additional acknowledgments go to CISE colleagues working on PKT projects: Paige Applegate, Devin Blem, Dylan Booth, Fareed Khamitov, Sean O'Reilly, Berlin Sankar, Adam Weizman, and Scott Wilkinson. Further

acknowledgments also go to members of the PKT clinical staff for their assistance in understanding the program and feedback on the interface: Harleen Kahlon and Kaitlin Taylor.

5. References

[1] Neal, E. G., Chaffe, H., Schwartz, R. H., Lawson, M. S., Edwards, N., Fitzsimmons, G., Whitney, A., & Cross, J. H. (2008). The ketogenic diet for the treatment of childhood epilepsy: A randomised controlled trial. *The Lancet Neurology*, *7*(6), 500–506. https://doi.org/10.1016/s1474-4422(08)70092-9

[2] Alexiadas, C., Faloona, M., Farfante, S., Ho, J., Joseph, P., Kahlon, H., Markatia, A., Maurer, H., Mudra, M., Phung, T., Rendina, A., Swift, M., Villancio-Wolter, K., Volonterio, R., & Sharma, V. (2021, March 14). *Precision Ketogenic Therapy for Every Stage of the Life Cycle*. CURE: Precision Ketogenic Therapy. Retrieved November 29, 2021, from https://cur.aa.ufl.edu/public_presenter/cure-precision-ketogenic-therapy/.

[3] Allen, H. J. (2016). *Precision ketogenic therapy for seizures: Metabolic basis for change in nutrients changing phenotype* (dissertation). ProQuest Dissertations Publishing, Ann Arbor, MI.

[4] Tyszkiewicz, J. (2010). Spreadsheet as a relational database engine. *Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data*. https://doi.org/10.1145/1807167.1807191

[5] Grupman, C. (2019, December 3). *Excel vs python: How to do common data analysis tasks*. Dataquest. Retrieved November 29, 2021, from https://www.dataquest.io/blog/excel-vs-python/.

6. Biography

Tyler Christian Mueller (born January 30, 1998) is a senior Computer & Information Science & Engineering student at the University of Florida, and a former student of Santa Fe College. Currently, he plans on graduating in December 2021 and pursuing a job at Epic Systems, a medical software firm located in Verona, Wisconsin, as a software engineer with a focus on refining extant systems. He has an academic understanding of the C/C++, Java, and Python programming languages and development techniques such as the construction of programming language lexers and parsers. Current goals include furthering an understanding of how medicine and computer science are connected, as well as deepening a personal understanding of software development techniques and methodologies. He has an interest in the development of video games for personal enjoyment.