A Context-specific and Customizable Nutritional Information Delivery Application for Type II diabetic patients

By

Zainabu Muti Chizi

099204

A Thesis submitted in Partial Fulfilment of the Requirements for the Degree of Master of Science in Information Technology at Strathmore University

School of Computing and Engineering Sciences
Strathmore University
Nairobi, Kenya

April, 2024

Declaration and Approval

Declaration

I, Zainabu Muti Chizi, declare that this work has not been previously submitted and approved for the award of a degree by this or any other University. To the best of my knowledge and belief, the research proposal contains no material previously published or written by another person except where due reference is made in the thesis itself.

Name: Zainabu Muti Chizi

Signature: ______________ Date: 7th April 2024

Approval

The research proposal of Zainabu Muti Chizi was reviewed and approved (for examination) by the following:

7th April 2024

Dr. Vincent Omwenga

Senior Lecturer, School of Computing and Engineering Sciences

Strathmore University

Dr. Bernard Shibwabo

Director of Graduate Studies

Strathmore University

Abstract

Diabetic Type II patients require specific nutritional information. This information is a necessary part of their self-management process. This process includes lifestyle factors such as dietary modification. Type II diabetic patients take up a large population of the total number of the diabetes patients in the country. The ratio of these patients to the health workers available would leave them overwhelmed. The information given is often scarce or poor. It is also generalised, assumed to fit all patients. This study aimed at developing a nutritional information delivery application that is context-specific and customizable. The application collects the specific contextual needs that impact the patients, processes it efficiently and delivers the nutritional information needed to its users, the nutritionist. It applies a rule-based representation technique to deliver the nutritional information. The study utilized the Kenya Food Composition Table that provides locally available food with caloric nutrients, and reports containing nutrition protocol to come up with the rules. These rules in combination with context factors outputs much more relevant nutritional information easily accessible to its users.

Keywords: *context, customised, nutrition information, rule-based representation model, type II diabetes*

Table of Contents

List of Figures

List of Tables

List of Equations

Abbreviations/Acronyms

BMR – Basal Metabolic Rate

FAO – Food and Agricultural Organization

GHG – Greenhouse Gases

GUI – Graphical User Interface

LCA – Life Cycle Assessment

LLM – Large language Model

LSD - Lean Software Development

NACOSTI - National Commission for Science, Technology, and Innovation

SU-IERC - Strathmore University - Institutional Ethics Review Committee

TDEE - Total Daily Energy Expenditure

UNESCO - United Nations Educational, Scientific and Cultural Organization

UNICEF - United Nations International Children's Emergency Fund

WHO – World Health Organization

Definition of Terms

Non-communicable diseases – These are chronic conditions that do not result from an acute or infectious process and are non-transmissible (UNICEF, 2021).

Peri-urban areas – These are zones between urban and rural land uses, and are located in between the outer limits of the urban centres and the rural environment (UNESCO, 2014)

Chapter 1: Introduction

1.1 Background

Diabetes Mellitus is a non-communicable disease that occurs when the pancreas does not produce enough insulin, or when the body cannot effectively use the insulin it produces. This is one of the most common diseases globally, with an estimated 536.6 million people (10.5%), at the age 20-79 years being prevalent to it (International Diabetes Federation, 2021). This figure has been predicted to rise to 12.2% by 2045. This disease can present in various forms, mainly Type I diabetes, Type II diabetes and Gestation diabetes mellitus. Type II diabetes occurs from failure of the body to produce insulin, failure of the body to effectively use insulin or both. More than 95% of people have type II diabetes (WHO, 2023).

For successful management of diabetes, a combination of lifestyle changes and pharmacological drugs is prescribed (Khardori, 2023). Life changes required include dietary modifications, stress management, physical activity and improved sleeping patterns (Borse et al., 2020). Dietary adjustments made require specific caloric nutrients, with a prescribed or limited amount of particular food types (Marín-Peñalver et al., 2016). Carbohydrates are a huge contributor to glucose in the body, and for diabetic patients would make their blood sugar too high. A low carb diet is recommended to them. Other recommendations include eating foods high in fibre, to avoid unsaturated fats, including fluids and about 5 portions of vegetables and fruits (NHS Choices, 2023). To maintain this, the patient needs to have knowledge on calorie intake counting. This information might not be readily available to all. According to Kiberenge et al. (2010), knowledge of diabetes is poor in the country and nationwide education is necessary.

Type II diabetes has affected people of all ages. In Kenya, the prevalence of diabetes is estimated to be at 4%, in adults ages 20-79 years, with about 60% of people not knowing they have diabetes (International Diabetes Federation, 2021). This population is spread across the country, with most of them hailing from urban and peri-urban areas (Ministry of Public Health and Sanitation, 2010a). The disease will affect them differently, as they experience different physiological, environmental, social and cultural factors (Ministry of Public Health and Sanitation, 2010b). Human needs are also constantly variant, depending on preferences and disposition. These variant factors form the human context or environment. A context specific application efficiently explores these factors to present the user with highly specific information for use (Omwenga, 2016). A customizable application is one that will be able to handle these context factors that are specific to these individuals. Due to the rapidly changing characteristic of context factors, it would be important to allow end users to personalise their applications to better suit their needs at the time (Ghiani et al., 2017). This would mean the patients get personalised results for specific queries input.

1.2 Problem statement

Specific dietary requirements are prescribed to diabetic patients. This meal requires a specific caloric count to avoid the danger of a blood sugar spike, which might result in complications and a stronger dosage of medication (Wu et al., 2014). A dietician or clinician is responsible for this assessment and to provide nutritional advice to the patient. Kenya faces a problem of a shortage in human resources. This is a shortage of properly trained medical staff, with those fully equipped to deal with diabetes patients very few (Ministry of Health, 2021). This would mean that the ratio of patient to medical staff is highly unbalanced, at about 1:20000 (Karega, 2022). This would lead to some of them not getting the care they need nor information that is of high quality (Katambo, 2021). Health information systems in Kenya are also facing issues of incompleteness and accuracy, which is an important factor when considering if it can be used by a patient (Ministry of Health, 2021). Additionally, the national clinical guidelines report (Ministry of Public Health and Sanitation, 2010b) recommends meal plans be made with locally available foods to reduce the financial strain on the patient.

Dietary recommendation applications have been made to fit anyone pursuing a healthy life. Those specific to diabetes have a log book-like structure that does not contain the necessary and specific information needed by diabetic patients (Gervais, 2023). With context factors of the patients to consider, a static application that does not adapt to various changes presented (Ghiani et al., 2017) would not offer real-time food options to the users. A successful mobile application should have the ability to take in and work with the changing human needs and environment therein (Omwenga, 2016). Therefore, a context-aware application would be developed to scale, adapt and respond to rapidly changing user needs (Unai Gangoiti et al., 2022).

1.3 Research objectives

1.3.1 General objective

This study aims to develop a context-specific and customizable nutritional information delivery application for Type II diabetic patients that provides suitable information based on a user's needs

1.3.2 Specific objectives

1. To investigate the current methods for provision of nutrition information to Type II diabetic patients in Kenya.
2. To analyse techniques used to design context-specific and customizable applications.
3. To develop a context-specific and customizable nutritional information delivery application for Type II diabetic patients
4. To validate the performance of the application

1.4 Research questions

1. What methods are being used to provide nutritional information to diabetic patients?
2. What techniques are used to develop context-specific and customizable applications?
3. How will the nutritional information delivery application be developed?
4. What methods will be used to validate the performance of the application?

1.5 Justification

There needs to be an overall improvement in the education of diabetes, creation of awareness and screening programs (Shannon et al., 2019). The study sought to provide nutritional information to the country. This information is essential not just to the diabetic patients, but also their caregivers. It would make self-management of diabetes much easier, with information having been consolidated and readily available in a mobile application. It would also serve as a source of information for dieticians and other medical professionals that need to give dietary recommendations of easily available local foods in their locale. These professionals need to be equipped with the right information to avoid misinformation or malpractice (Nutritionists and Dieticians Act, 2012)

1.6 Scope and limitations

The study focused on delivering nutritional information to Type II diabetic patients. This is due to the prevalence of the disease being higher than other types of diabetes. The study used relevant publicly available datasets to develop the application. The study was limited to the Kenyan locale, with its results tailored to provide results within the Kenyan context both geographically and in language.

Chapter 2: Literature review

2.1 Introduction

This chapter discusses the related works that support the proposed study and meet the first and second objective listed in section 1.3.2. It discusses the management of diabetes and the current methods used to assess nutritional information, context factors and techniques used to develop context-specific and customizable applications. The last section contains the proposed conceptual framework.

2.2 Diabetes Management

Type II diabetes in Kenya is responsible for 85% of the national's health burden (Jones, 2013). According to the World Health Organization (WHO) factsheet (2023), risk factors for this disease include being overweight, a low physical activity level and genetics. Pre-diabetics is also a risk factor of type II diabetes. This is a blood sugar level above normal but below the level indicative of diabetes (Bansal, 2015). These people are at risk level, and are likely to convert into type II diabetic patients (Mohamed et al., 2018). Type II diabetes can be prevented and managed successfully. The primary intervention for prevention of diabetes is recommendation of a lifestyle change that would tackle the risk causing factors of diabetes. These steps are self-managing, with a heavy reliance on the information available to the patient. The secondary prevention is provision of treatment to patients showing early signs of diabetes. These signs can be managed to prevent hospitalisation. Health workers play the major roles of disseminating relevant information to patients through promotion and public health education (Ministry of Public Health and Sanitation, 2010b).

The treatment and management of diabetes can be done using pharmacological methods and non-pharmacological methods.

2.2.1 Pharmacological methods.

These methods involve the use of drugs to treat and manage diabetes (Ministry of Public Health and Sanitation, 2010b). These can be oral agents or injectable agents. Oral agents are drugs that are swallowed and used to lower the glucose levels in your body (Farinde, 2021). This would assist the body use the insulin effectively. This is a treatment plan that involves use of medicinal drugs.

While helpful, oral agents have various side effects which include cardiovascular effects, nausea, abdominal discomfort or pain (Mohamed et al., 2018). Injectable agents involve the use of injections. The most common injectable agent is insulin. These agents have some side effects, including abdominal pains and in some agents a loss in weight. The sight of injection might also be irritated due to reactions of antibodies to RA-GLP1, which are some of the drugs introduced in the body (Mohamed et al., 2018). It is because of these side effects that patients are encouraged on behavioural modification instead (Watts, 2022).

2.2.2 Non-pharmacological methods

These are methods that encourage behavioural and lifestyle changes (Ministry of Public Health and Sanitation, 2010b). These lifestyle changes include diet modification, increase in exercise, regulation of sleep patterns and avoiding stress factors or stressful situations (Marín-Peñalver et al., 2016). This method requires an active involvement of the user because it requires self management. This is done by goal setting and strict follow-ups to goals made (Gray & Threlkeld, 2019). The method requires the patient to be knowledgeable in diabetes, its prevention and its care. They should understand calorie count for diet modification, pick the right physical activities to take part in and maintain a recommended amount of sleep, which is about 6-7 hours (Borse et al., 2020). This can be overwhelming for some users, especially the access to this information.

According to the national clinical guidelines for management of diabetes mellitus report (Ministry of Public Health and Sanitation, 2010b), a lifestyle change is highly recommended as one of non-pharmacological management methods. It is the first step for management. This step uses the combined therapy of lifestyle change and a medical drug. Monitoring is then done, to ensure that the glycaemia goal is met (figure 2.1). Other steps are then introduced in cases where the first step is not sufficient.

Patients are able to access the necessary nutritional information from health workers (Kiberenge et al., 2010). Follow-up sessions done to monitor blood glucose levels are also used to advise the patient on information needed based on the result of the progress of the disease or questions asked.

Other sources of information are in training sessions held by various organisations. These sources are minimal, scattered and their sessions uncoordinated, and could therefore be increased to become available to more people, especially those with diabetes (Kiberenge et al., 2010).

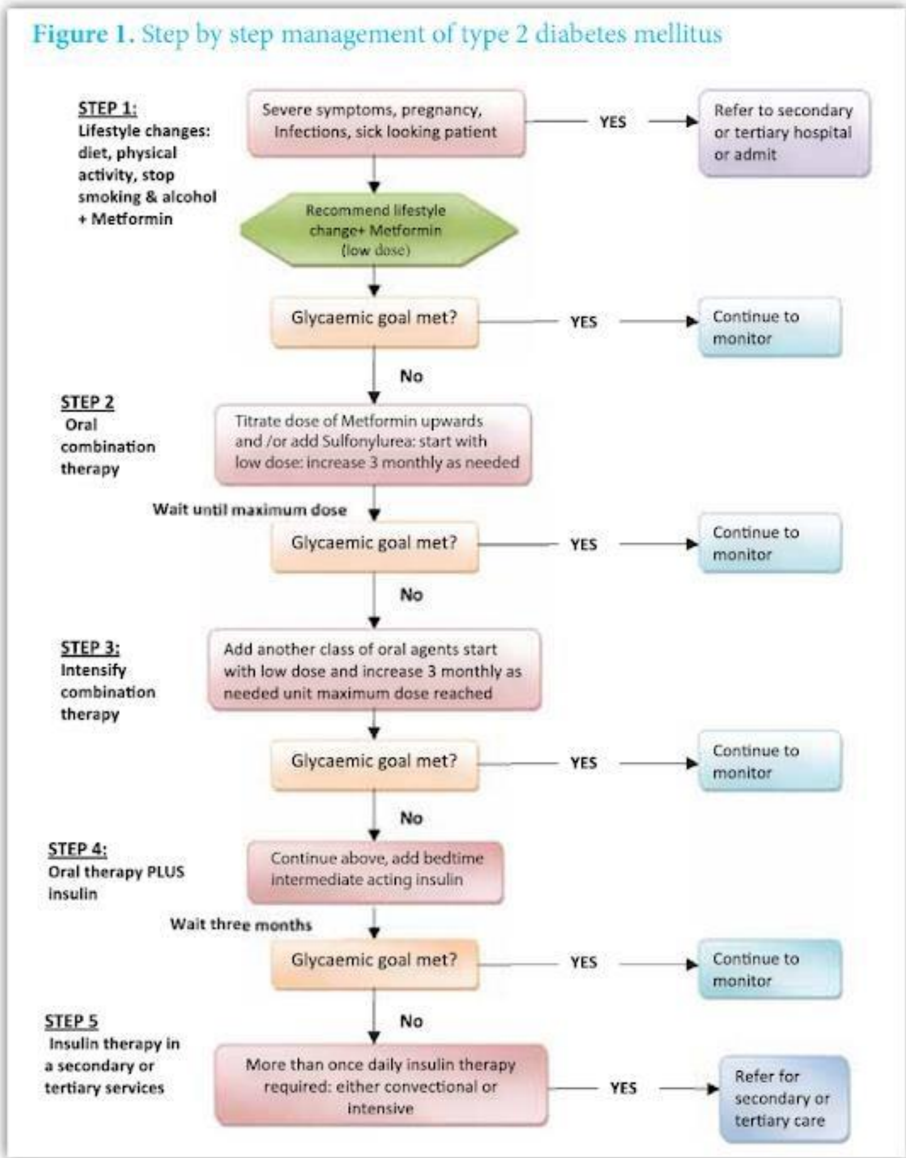


Figure 2.1: Step by Step management of type 2 diabetes mellitus (Ministry of Public Health and Sanitation, 2010b)

2.3 Context factors

Context, in context-aware computing, has been defined as any information that can be used to characterise the situation of an entity, where an entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and the application themselves (Salber et al., 2004). These factors are further distinguished into four categories. These are identity, location, time and status (Dey, 2001). Identity refers to the ability to assign a unique identifier to a particular entity (Salber et al., 2004). Location refers to more than just the place or geo-location of the entity. It encompasses proximity or orientation, or any other information that can be used to deduce geo-location information (Musumba & Nyongesa, 2013). Status is a factor describing what can be sensed from an entity. Salber et al. (2004) gives various examples of characteristics that can be considered to be the status of the entity. In people, this would be the physiological status, mood, or activity being undertaken. In software or computing components, this can be the status of the CPU, or the existence of resources, such as files in a file system. Time is perceived as a factor that helps characterise a situation. It helps grasp historical information that is important to the application (Musumba & Nyongesa, 2013).

Type II diabetic patients come from across all ages and cultures. Lifestyle change to include physical activity and dietary change has been known to positively impact and has been used as a preventative measure, hence leading to the reduction of the mortality rate (Puska, 2002). Nutrition delivery applications have been recommended to help patients follow dietary recommendations and improve and monitor their own behaviours (Lee-Fang Teong et al., 2022). With mobile phones being a constant part of our lives, nutrition delivery applications are recommended to be mobile based. This helps for ease of access and constant use in their daily routine. It also helps in behavioural change interventions that are needed for diabetic patients (Fedele et al., 2017). Different patients would have different preferences and expectations. The national clinical guidelines for management of diabetes mellitus report (Ministry of Public Health and Sanitation, 2010b) has recommended prescription of financially feasible diets. This would mean diets from local foods that have the proper nutrient requirement. Relevant information on these food types is at the minimum. Knowledge of diabetes is poor in various communities in the countries. This impacts the care and self-management of the disease (Kiberenge et al., 2010). Perception of the disease is also ruled as poor, leading to the need of culturally sensitive means of intervention and education (Phrashiah Githinji et al., 2022). Various factors such as physiological conditions, economic considerations, location of the user and cultural perceptions and beliefs are some of the context factors that can be considered. With these factors in mind, proper interventions can be created to combat the disease.

2.4 Context-centred applications

Context is important and useful in the creation of applications. Context-aware applications take in relevant information from the users, process it to produce relevant information or services to the users (Dey, 2001). Context factors are different in different applications and situations. The applications that use context information are mobile ubiquitous services that offer the benefits of adaptation, personalization and proactivity. These applications can change depending on the context of the user, adjust to user's needs and context or offer them features to adjust the settings and usage as needed and act flawlessly without the user's intervention (Makkonen et al., 2009).

The basic functions of context aware applications are summarised as to present information, perform automatic execution of a service and storage and easy retrieval of contextual information for later use. Presentation of information would occur either by showing it to the user, such as location data or presenting certain outputs to the users to advise on selection of actions, such as presenting nearby printers that the user can choose from (Salber et al., 2004). Performing automatic execution is the process of the application responding to certain triggers set and performing a task or automatically reconfiguring due to change in context (Musumba & Nyongesa, 2013). An example would be recalculation of a new route on a map when a user of a context-aware car navigation system misses a turn indicated in the initial route. Retrieval of information involves the process of tagging information with relevant contextual information to allow for easy use and access (Musumba & Nyongesa, 2013). A meeting notes would tag notes based on time and location where the notes were taken. This function would then help in customization of the application.

Inputs of context aware applications depend highly on the task to be performed. In nutritional information delivery applications, these factors can be used to tailor the right dietary recipe and food items to its users (König et al., 2021). Context factors are often ever changing, dynamic and diverse, reflecting the current state of the user, the environment and the operating system. The applications processing this information need to rapidly adapt and not be static in terms of service delivery (Cuddy et al., 2005). Context-centred applications are therefore recommended to be able to handle the different requirements of the users (Salber et al., 2004).

2.4.1 Attributes of context-aware applications

Mobile applications that are context-centred have a common net effect required for the proper working of the application to be obtained (Omwenga, 2016). These applications function in a highly dynamic environment that takes in context factors and identifies specific challenges, opportunities and constraints (Makkonen et al., 2009). They leverage these attributes to handle the ever-changing nature of context information, and provide value to its users. These attributes are discussed below.

2.4.1.1 Proximate selection

This is a technique that rationalises that objects are in close proximity to each other, and would be easier for an entity to choose (Omwenga, 2016). The entities in the system could be the user, the device or the computer components. The location data is taken into account, to give proximity to nearby resources using various location sensors. There are factors that should be considered for user interfaces developed with location proximity in mind. These are bandwidth and user interface display. These factors would affect the functioning of the application. Bandwidth could cause network traffic on the applications server and imagery to present location data, whether in maps or texts, must take in account capabilities of the screen and communication bandwidth (Schilit et al., 2002). This attribute is therefore a guiding principle for frameworks and architecture that use locale data. An example is using a coarse grain view or approximate location to map out locations. This tackles the problem of traffic.

2.4.1.2 Automatic contextual reconfiguration

According to Schilit et al. (2002), this is a process of removing, adding or altering connections between components. These changes occur based on changes in context. The hosts in a particular vicinity will change based on the user locations. If system configuration relies on the location, then the system will adapt to the user’s location. This attribute contributes to the adaptability function of context-centred applications. Depending on other factors, it could also inform personalization of applications. Customization of context factors by the end user would help in managing context factors, as they will be aware of what an application considers relevant context in different situations (Ghiani et al., 2017). This would help in avoiding confusion to the user (Schilit et al., 2002).

2.4.1.3 Contextual information and commands

Contextual commands occur when queries on contextual information are issued. With changes in contextual data, results of the queries may vary. Likewise, a combination of multiple queries would lead to a different result (Schilit et al., 2002). This attribute is therefore important to understand the human situation and data collected from that situation to be able to create the correct queries for the system. The commands made should factor in all the various changes in data. If third parties are involved in application, security of data should be put into great consideration, especially as technology advances expose application to new threats (Buthpitiya et al., 2012).

2.4.1.4 Context-triggered actions

These are defined by simple IF-THEN rules. These rules contribute to the reasoning mechanism of this application, and is a fundamental requirement of the system (Vahdat-Nejad, 2019). These rules are created to trigger or act when certain conditions are met (Omwenga, 2016). With data in use changing rapidly, and the state of the device or software also changing, rules should be made to not trigger unnecessarily. They also must not respond too slowly as the system might seem sluggish and slow. The applications can therefore have rules based commands which require specific action to happen first and are not automatically triggered as context-trigger actions are (Schilit et al., 2002). This creates an organised and smoother application, as triggers can be reminders, for example.

2.4.2 Features of context awareness

Dey (2001) categorised the different and important features of context-aware applications. This was done to provide a generalised outlook to fit all existing applications that are context-specific. Zhang et al. (2009) proposed context management that would include context acquisition, abstraction, interpretation, aggregation and inference. These issues cater for acquiring, processing, reasoning mechanism and presentation of data. They also satisfy the basic functions of context specific applications discussed in section 2.4. Below are the features of these applications.

2.4.2.1 Context acquisition.

This is the process of acquiring or collecting inference information. This can be done from sensors, user profiling or software agencies (Musumba & Nyongesa, 2013). It is done with minimal user interaction. This information can include the user's location or the time of day from the device.

User profiling can contain information that feeds into the preferences of the user. This can come from the actions done in an application, such as clicking on icons (Zhang et al., 2009).

2.4.2.2 Context abstraction.

This involves making information relevant to the application. The raw data collected has to be made available in a way that would be most useful to the application and its uses. Location data collected by a GPS sensor, for example, will be processed to get the specific details to be used in a location-aware application (Zhang et al., 2009). Context widgets help provide information to the operating environment by presenting it in a generic manner, regardless of how it was sensed (Dey, 2001). This acts as a mediator between the application and this environment. They also insulate the applications from presentation concerns (Salber et al., 2004).

2.4.2.3 Context interpretation

This is the process of transforming context information into meaningful information to be used by an application. This involves analysis of the context factors and understanding how they are used (Zhang et al., 2009). This forms the core of context-aware application designing principles. This representation of context factors can be done by rules that are written to allow easy integration of context. This would then offer the application deductive and deductive reasoning ability, that allow it to produce the desirable output (Zhang et al., 2009). There are various modelling methods used to represent context, as shown in the figure 2.2 below, highlighting their advantages and disadvantages.

Method	Description	Pros	Cons
Key-value model	Using a list of attributes in a key-value for modeling contextual information [e.g., Schilit et al. 1993]	Very simple and easy to manage	May have a lot of blank slots (missing values); lack capabilities for sophisticated structuring for enabling efficient context retrieval algorithms
Markup scheme model	Using a hierarchical data structure that consists of markup tags with attributes and content to model context [e.g., Brown et al. 1997]	Able to cover the higher dynamics of contextual information	May be difficult and non-intuitive to capture complex contextual relationships and constraints
Object-oriented model	The detail of context processing is encapsulated at an object level. Access to contextual information is provided through specified interfaces only [e.g., Cheverst et al. 1998]	Employing encapsulation and reusability to cover parts of the problems arising from the dynamics of context	Not responsive to changes in context space; require low-level implementation agreement between applications to ensure interoperability
Logic-based model	Defining context as facts, expressions, and rules by using logic representation (e.g., predicate logic proposed by Ranganathan and Campbell [2003])	High degree of formality	They use crisp rules, which may not be suitable for dealing with context featured by uncertainty and ambiguity
Ontology-based model	Specifying different types of context and their relationships in a uniform way [de Almeida et al. 2006; Korpipää et al. 2003; Weißenberg et al. 2004].	Enabling contextual knowledge sharing and reuse in a ubiquitous computing system	Can only be used when context ontology can enumerate all possible context events that can trigger actions, which is often difficult; in most cases, ontologies are not publicly available

Figure 2.2: Context representations methods (Zhang et al., 2009)

The modelling method used is the rule-based representation method. Rule based systems represent information in a set of rules, in the form of IF <condition> THEN <action>. This model has also been used as a support for reasoning for the application (Nalepa & Bobek, 2014). The system consists of elements that help in its working. These are the knowledge base, the inference engine and the fact base (Vahdat-Nejad, 2019). The rules created for the application are stored in the knowledge base. It also stores facts and data of the system that will be used in the rules. This acts as the long-term memory of the model (Hayes-Roth, 1985). The checking of rules to ensure all conditions are met is done in the fact base while the inference engine implements and interprets the rules (Nalepa & Bobek, 2014). An example of a rule-based system model is shown in figure 2.3 below.

Rule based models are advantageous in that they form a powerful reasoning mechanism for the application, as compared to other modelling techniques. They are also versatile to allow insertion of data, and support expansion of the knowledge base (Nalepa & Bobek, 2014). New rules can be inserted to the knowledge base with the newer facts and inputs to make it dynamic. They have the ability to combine the results of various rules, and therefore resolving a wide range of complex problems (Hayes-Roth, 1985)

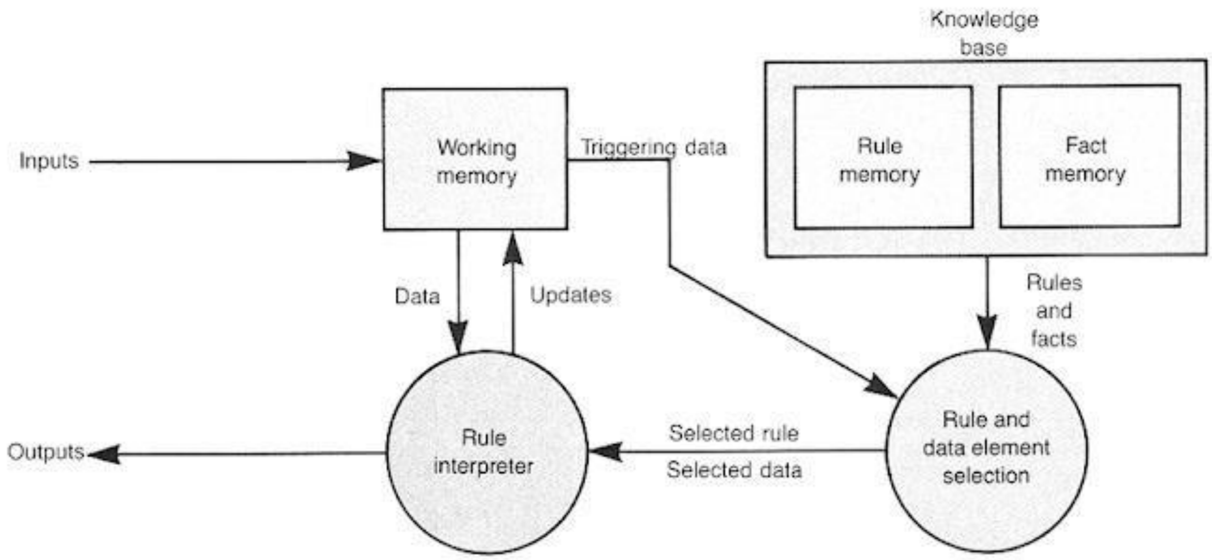


Figure 2.3: Basic features of a Rule Based System (Hayes-Roth, 1985)

2.4.2.4 Context aggregation

This is the process of integrating information that is needed by the application. This information can be from the various sources of data that the application had. This information is then filtered to be able to provide the necessary output needed by the user (Zhang et al., 2009). Aggregators gather logically related information and make it available for use, which simplifies the process of access of information (Salber et al., 2004). They act as a proxy to context for applications (Dey, 2001).

2.4.2.5 Context inference

This is the ability of the application to sense user context in real time and for it to adapt to changes (Zhang et al., 2009). Contextual filtering helps in determining the information to be output to the user. This can be done using historical data and based on the current context factors presented. Rule matching also influences this step, as it determines what service can be delivered (Sridevi et al., 2012). This means that different information could be shown to the user. For example, the time of day would impact the results for meal plans available to be shown, giving a different result in the morning, that is breakfast, or lunch for afternoons (Zhang et al., 2009).

2.5 Techniques used in developing context-specific and customizable applications

This section discusses context-specific and customizable applications developed, focusing on the techniques used in development and design of these tools.

2.5.1 Learnings from Developing a Context-Specific LCA Tool for Buildings—The Case of LCAbyg 4

Kanafani et al. (2021) developed a tool that tackles the development problem in the construction industry. The Paris agreement states that various countries will limit global warming by 2030. The building sector is responsible for about 70% of greenhouse gases (GHG). The researcher developed a tool, based on the Life Cycle Assessment (LCA) Tool used to measure these GHG emissions. This tool was to help perform assessment of a building and corrective action be taken by the Denmark government. The tool was customised to meet building regulation laws in Denmark, information flow, the design methods and construction method. The tool's input included some default data. However, depending on the stage of the building, inputs may vary and would require user input. This data would also vary and change as design diagrams change, building properties also vary and time taken for the changes to occur is also different for different buildings. The application used an integrated model structure that hid context information from the environmental data. The data sources were interlinked or connected in a layered approach.

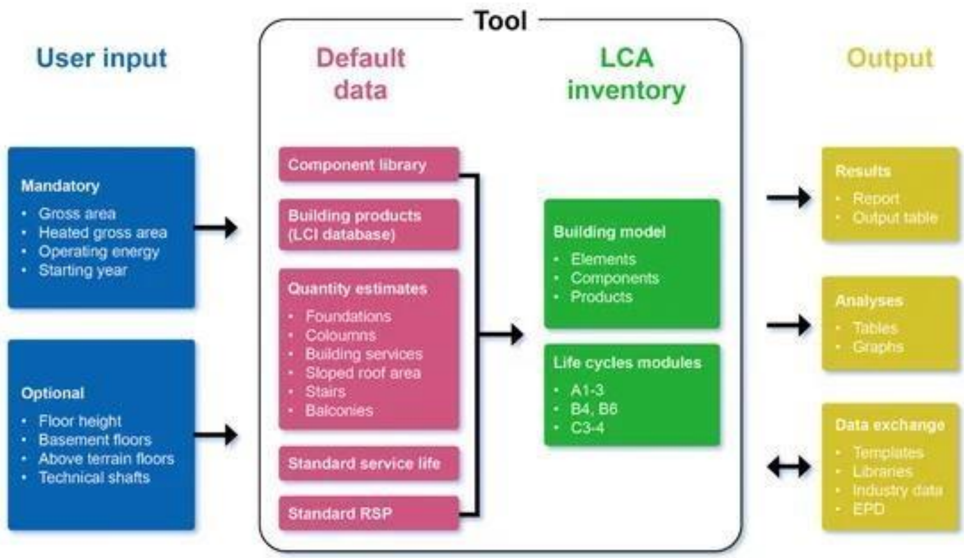


Figure 2.4: A context-specific LCA tool for GHG measurement (Kanafani et al., 2021)

The system was rated as useful by 69% of the participants in the study. A survey administered showed that some of the participants had used the application in other ways, which was also useful to them and their uses. This included building inspection or error correction. The researcher suggested an improvement in the model that does better data exchange than the one developed. A suggestion to expand the application to fit other user needs better was also added.

2.5.2 Fill in the Blank: Context-aware Automated Text Input Generation for Mobile GUI Testing

Liu et al. (2023) developed an application that can be used in the Graphical User Interface (GUI) Testing tool. This tool is created to provide better testing capabilities as compared to other GUI testing tools. Its performance was at 87%, when tested in 106 apps from Google play, which was higher than most tools. The application was modelled in the "pre-train, prompt and predict" paradigm of the large language model (LLM). This model could understand and receive local and global contexts available. The researchers extracted information, and grouped them into local and global context information, and input widget information. This information was put through a prompt generator that preprocessed the data, created language patterns and rules. The rules used the patterns created, and could be combined depending on the context and application it was tested on.

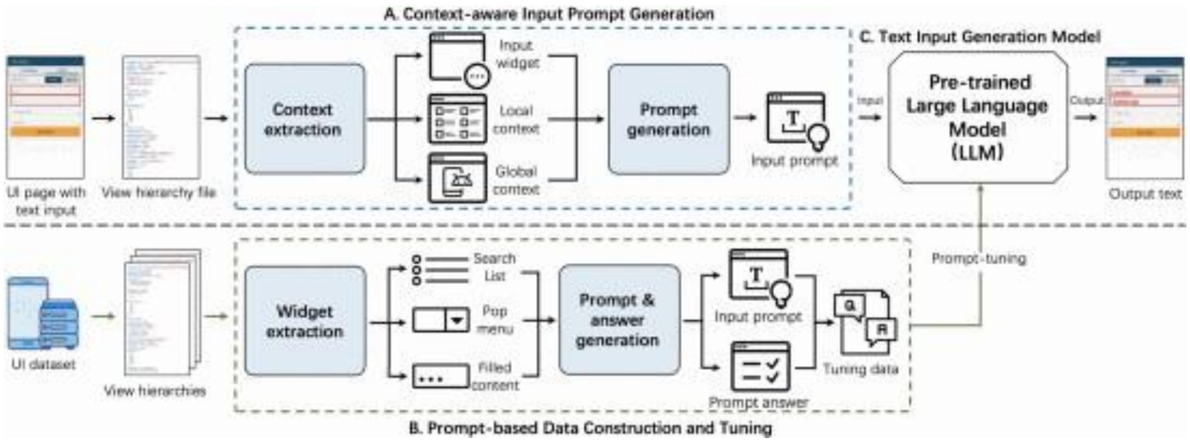


Figure 2.5: Q-typist, an LLM context aware application (Liu et al., 2023)

The application was able to generate input text automatically with context extraction done to find out what information was needed in the next page or textbox. The authors however suggest a need for improvement in the context extraction stage, as it lacked enough information to generate the correct inputs.

2.5.3 App2Vec: Context-Aware Application Usage Prediction

Wang et al. (2021) created an application that could sense and predict when and where certain apps are used. App2Vec uses contextual factors to predict this information. This application captures the spatio-temporal context information to accurately predict this information. The application uses an algorithm that is a combination of multimodal embedding and Bayesian mixture model to perform this prediction. Multimodal embedding helps in storing all context factors gathered in the same size. The Dirichlet process in the model was used to enhance the flexibility of the model. The Bayesian model was used to represent context factors as vectors in the processing of data. The application was able to map the intention of the user and the app being used.

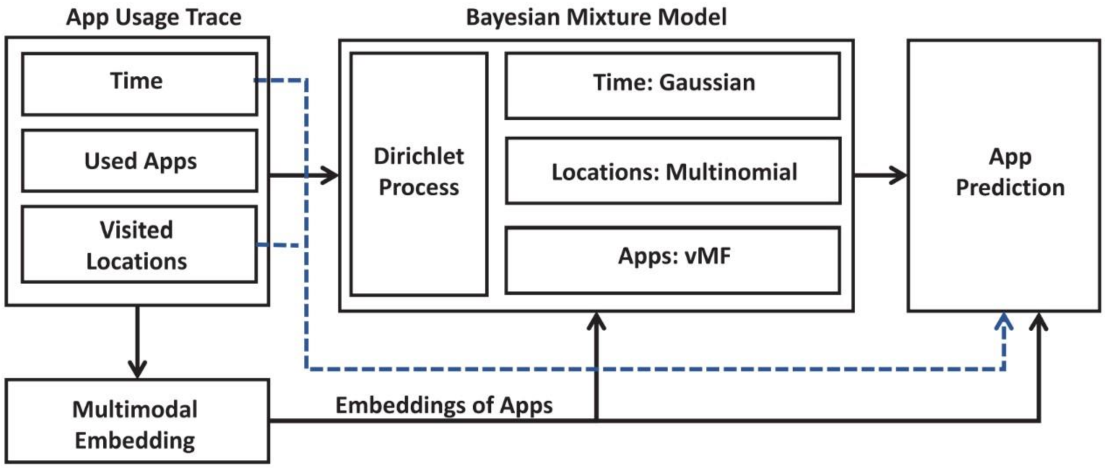


Figure 2.6: App2Vec, a usage prediction application (Wang et al., 2021)

The researchers recommend the use of context-aware usage models to overcome and deal with the randomness of human behaviour. These techniques have been found to reveal hidden semantic data that could be useful to other applications.

2.6 Conceptual framework

In this study, the rule based representation model was adopted to contextual model the data. The Rule Based Model receives data from the user, who inputs blood glucose level and other factors such as weight and height. Another data source is the contextual information, which is collected from sensors embedded in the user's devices. Food items and food preparation data is used to answer conditions in the rules set, which are based on contextual information and user inputs. The data is further filtered and interpreted in the contextual filtering stage. Relevant food items, with nutritional values are then translated into local dialects and output to the user. This data will be accessed via the user's devices.

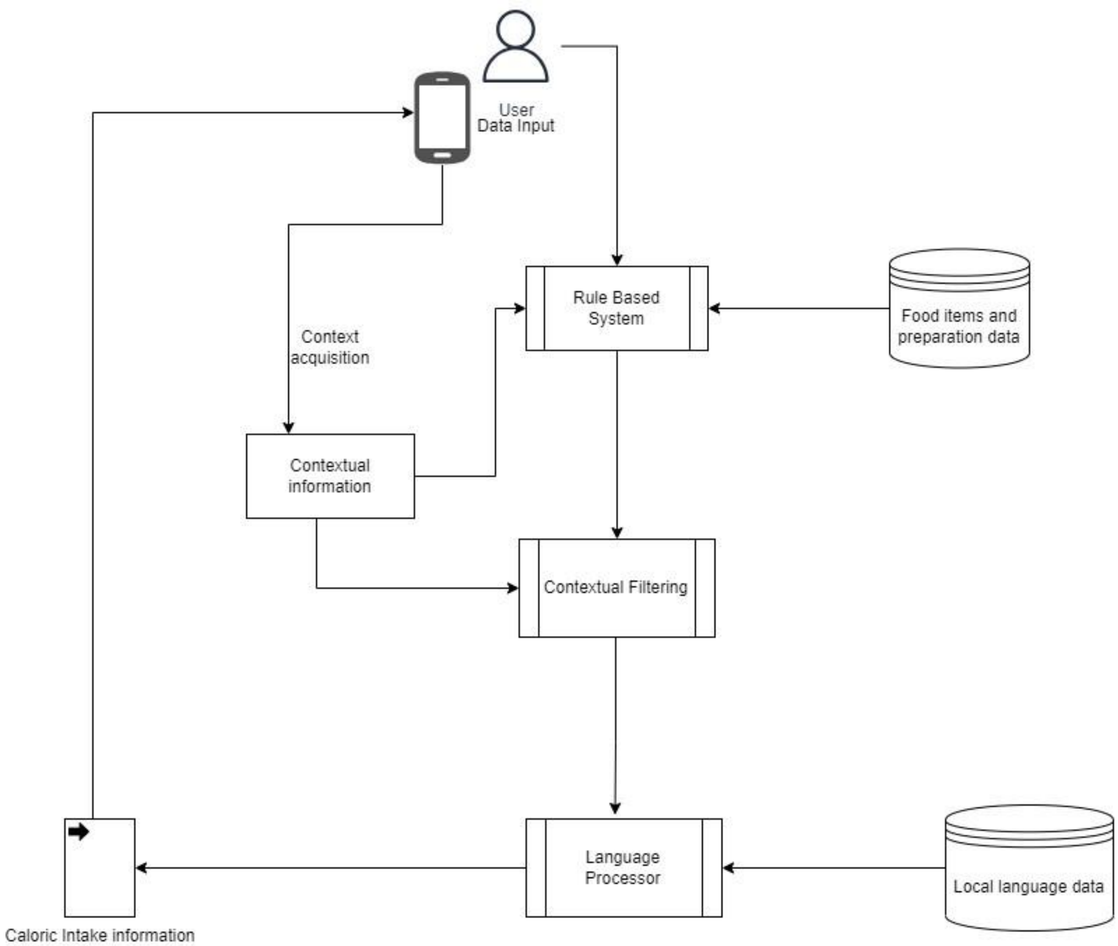


Figure 2.7: Conceptual Framework

Chapter 3: Methodology

3.1 Introduction

This chapter details the approach to achieve the third and fourth objectives, as listed in section 1.3.2. It discusses the process of conducting the study, the methodology selected, the target population and sample to be used and data collections techniques to be applied. Ethical considerations and concerns with regard to research quality are addressed.

3.2 Research design

The design used is the quasi-experimental research design. This design is commonly used to measure the cause-effect of a study or experiment, and is commonly used in psychological, medical or informatics study (Thyer, 2012). This is because the application developed uses its effect on the population to determine its effectiveness in delivery of information. This design is recommended for studies where the researcher has no control over the treatment, but has introduced a means that could improve the situation or disease. Such is the case with this study (Harris et al., 2006).

3.3 System development

The method used in this study is the Lean Software Development methodology. It is a part of the agile system methods (Kumar et al., 2019). Agile methods are based on iterative and incremental development methods. The major characteristics of agile methods is its adaptive and evolutionary development, and its flexibility and fast response to change (Hayat et al., 2019). It also promotes communication between the developer and the target population of the system. The principles of agile methodology encourage deadline emphasis and progress, which is measured by the working of the system in development. It also encourages simplicity at its core, both in communication to its potential users and in the changes made to the system (Kumar et al., 2019).

The lean software development (LSD) methodology is an iterative process that focuses on value creation. This can be to the product to be developed and the service it offers. Once value has been added, the non-value activities are eliminated (Shahzeydi & Gandomani, 2016). It also encourages learning fast, adopting and delivering a valuable product at the end of the process (Kumar et al., 2019). The figure below, figure 3.1, shows the steps for LSD.

Figure 3.1: Lean Methodology (Shahzeydi & Gandomani, 2016)

3.3.1 Identify value

This step emphasises on the identification of all value activities. These are the steps that help in creation of the plan (Shahzeydi & Gandomani, 2016). In this study, these activities were identifying the resources needed for the successful implementation of the application such as accessing the right dataset to the project, deciding on the programming language to be used, the people that would need to be involved in the study and resources needed for the validation of the application's performance.

3.3.2 Map the value stream

This step is for creation of the plan. Once all value items have been identified, the project owner maps it out (Shahzeydi & Gandomani, 2016). The planning helps in the speed the project will move through. At the same time, one can identify all resources missing or to be acquired easily. In this study, this was utilized in the creation of timelines, accessing any resources needed to complete the development of the application, which would included learning and researching further on programming languages to be used and techniques needed.

3.3.3 Create flow

A flow stream encourages elimination of any wastage in the process (Shahzeydi & Gandomani, 2016). This was implemented in the study by perfecting the plan, and completing other tasks identified instead of being at a stand-still.

3.3.4 Establish Pull

This step encourages creation of demand for the product or service to be produced. This satisfies the condition that there is no wastage by producing more than is needed (Shahzeydi & Gandomani, 2016). In the study, gathering of requirements satisfied this step. The researcher reviewed relevant literature that helped in the development of the tool. This ensures that the product developed is relevant and serves a need in the society.

3.3.5 Seek perfection

Perfection is a process acquired by continuous improvement. This step encourages the iterative process, where the other steps are revisited to get a better plan overall (Poppendieck & Cusumano, 2012). This study achieved this by focusing on the objectives listed and regularly seeking feedback from the potential users.

3.4 Target source of study information

This study aimed to develop a context-specific and customizable nutritional information delivery application for Type II diabetic patients that provides suitable information based on a user's needs. To achieve this, an investigation on the methods currently used in the country for provisioning of nutrition information to Type II diabetic patients in Kenya were done. This helped the researcher understand in a contextual manner how the information is provisioned, what kind of information is provisioned, the actors involved and any existing protocols that guide in the dissemination of the information. The target source of this information included existing literature on how nutritional information and related information is disseminated in Kenya and policy documents/reports. The data collected exists as published reports by the Ministry of Public Health and the Food and Agricultural Organization (FAO & Government of Kenya, 2018; FAO et al., 2018) available online.

3.5 Data collection

Literature review techniques including thematic analysis were applied to gather the secondary data from existing documents detailing the information dissemination methods. The data on the guidelines and methods for information dissemination were accessed at https://ncdak.org/wp-content/uploads/2021/07/Kenya-National-Clinical-Guidelines-for-the-Management-of-Diabetes-2nd-Editiion-2018-1.pdf. Other data elements used in this study include food recipes for various communities in Kenya. This data was obtained from the existing dataset on the Kenya Food Composition Tables dataset, that contains the various food items consumed and their caloric values (FAO & Government of Kenya, 2018). This dataset is accessed at https://nutritionhealth.or.ke/programmes/healthy-diets-physical/food-composition-tables/.

3.6 Data analysis

The reports sourced online were converted into csv files to allow ease of usage and manipulation. The dataset was subject to pre-processing analysis to check on completeness of the data. The data was then cleaned to leave the needed and relevant data. Correctness of the data was also conducted to ensure that the values are numerical, or string where needed. To ensure that the information is valid and correct, missing data and any outliers have been handled correctly. Exploratory analysis involving descriptive statistical analysis like measures of central tendencies was conducted. The results of the descriptive analysis have been presented using tabular and graphical representation techniques. The cleaned data was further subject to inferential analysis to establish relationships and patterns between different variables in the datasets. This facilitated in the development of rule based model to be used in the development of the application. The rule based model was also evaluated and information interpreted into meaningful information to the users.

3.7 Research quality

The research quality used the aspects of reliability and validity to ensure that the research was carried out correctly. Reliability involves the consistency and reproducibility of the results of a study (Thanasegaran, 2009). The internal consistency of the research was assessed to ensure that the results obtained attain consistent results. The use of Cronbach's coefficient alpha was used to assess the inter-item reliability and attainment of consistent results (Mohsen Tavakol & Dennick, 2011). Validity of the research measures whether the results produced are accurate, and if they address what they are intended to measure (Golafshani, 2015). This study applied content validity measure to establish whether the recommendation made are meaningful, accurate and relevant to the user of the application (Thanasegaran, 2009).

3.8 Ethical considerations and dissemination

This study made use of publicly available datasets. The researcher ensured to communicate with the owners of these datasets to notify them on the use of the datasets. The study was conducted subject to an ethical approval from the Strathmore University - Institutional Ethics Review Committee (SU-IERC) and the National Commission for Science, Technology, and Innovation (NACOSTI). This was to ensure alignment of values to both the school and the country. All works used to guide this research have been acknowledged through citation and references.

Chapter 4: System Analysis and Design

4.1 Introduction

This chapter details the system requirement analysis, which include the requirements for the nutrition delivery tool. It also includes the UML diagrams used to show the system design and architecture, actors involved and their interactions.

4.2 System Analysis

This study aimed to develop a context-specific and customizable nutritional information delivery application for Type II diabetic patients that provides suitable information based on a user's needs. This analysis outlines the functional and non-functional requirements, and the gathering of the implemented requirements.

4.2.1 Requirements gathering

Data used in this study were from existing literature and data sources. Data collection involved accessing these reports online. The application required guidelines and nutrition information, methods of provision and dissemination and protocols guiding this. The report (available at https://ncdak.org/wp-content/uploads/2021/07/Kenya-National-Clinical-Guidelines-for-the-Management-of-Diabetes-2nd-Editiion-2018-1.pdf.) was used to inform the nutrition requirements of diabetes type II patients. The methods used include clinical visits by patients at different intervals and education programs (The National Diabetes Prevention and Control Program, 2018). The dietician and patient agree on a set of goals to meet, varying from weight goals to diet regimen discipline. The nutritionist takes the instructor role in this interactions. Nutrition information is then provided. In this report, the guidelines are not in any specified units, rather using vague portions such as 'a variety of', 'increase intake of' or 'limit the intake of'. Therefore, a tool that uses quantities and provides food items of said quantities is needed.

Data on food items and recipe was also used. The existing dataset (available at https://nutritionhealth.or.ke/programmes/healthy-diets-physical/food-composition-tables/) was used to make food items recommendations. This dataset contains local food items, raw or cooked, and its accompanied caloric values. These values include the nutrients available in the food and its total energy provided. The data was extracted from the reports using tabular and cleaned to form the main dataset in a csv that was easy to manipulate and work with.

4.2.2 Functional Requirements

These are requirements that describe what is expected of the application when in use. They cover the functionalities and operations of the operations, and the responses expected from user inputs. The following are the functional requirements:

i. A user should be able to input their demographic details
ii. Calculate energy needed and distribution per meal and in macronutrients
iii. Intuitively acquire context factors such as time from the user's device
iv. Provide nutritional information based on context factors and caloric distribution.
v. Manage and resolve maintenance issues by the administrators of the application
vi. Save and update of details by all users of the system

4.2.3 Non-functional Requirements

These are constraints on services offered by the system. They deal with design, quality features and external factors that have an influence on the performance of the application. These non-functional requirements are:

i. The system should be easy to use for the different potential users of the application.
ii. The system should provide outputs that are accurate with minimal errors experienced.
iii. The system should be available to all users and is easily accessible for use.
iv. The system should give consistent results to its users, thereby being reliable.
v. The system should be able to handle multiple users at the same time
vi. The system should allow easy correction of its errors by administrators when they occur

4.3 System Design

This is the process of developing abstract models to represent different perspectives and interactions in the application (Wainaina, 2022). This section details the different diagrams involved in the different stages of the design process of the system.

4.3.1 System Architecture

The architecture shown in figure used a layered approach. The bottom layer, layer D represents a data store of the datasets used in the study. It also includes the data stored after the model has taken in the user inputs. The user interacts with the system at Layer A, which is the first layer. The second layer represents the functions and forms, that process the data taken from the user and the nutritional information.

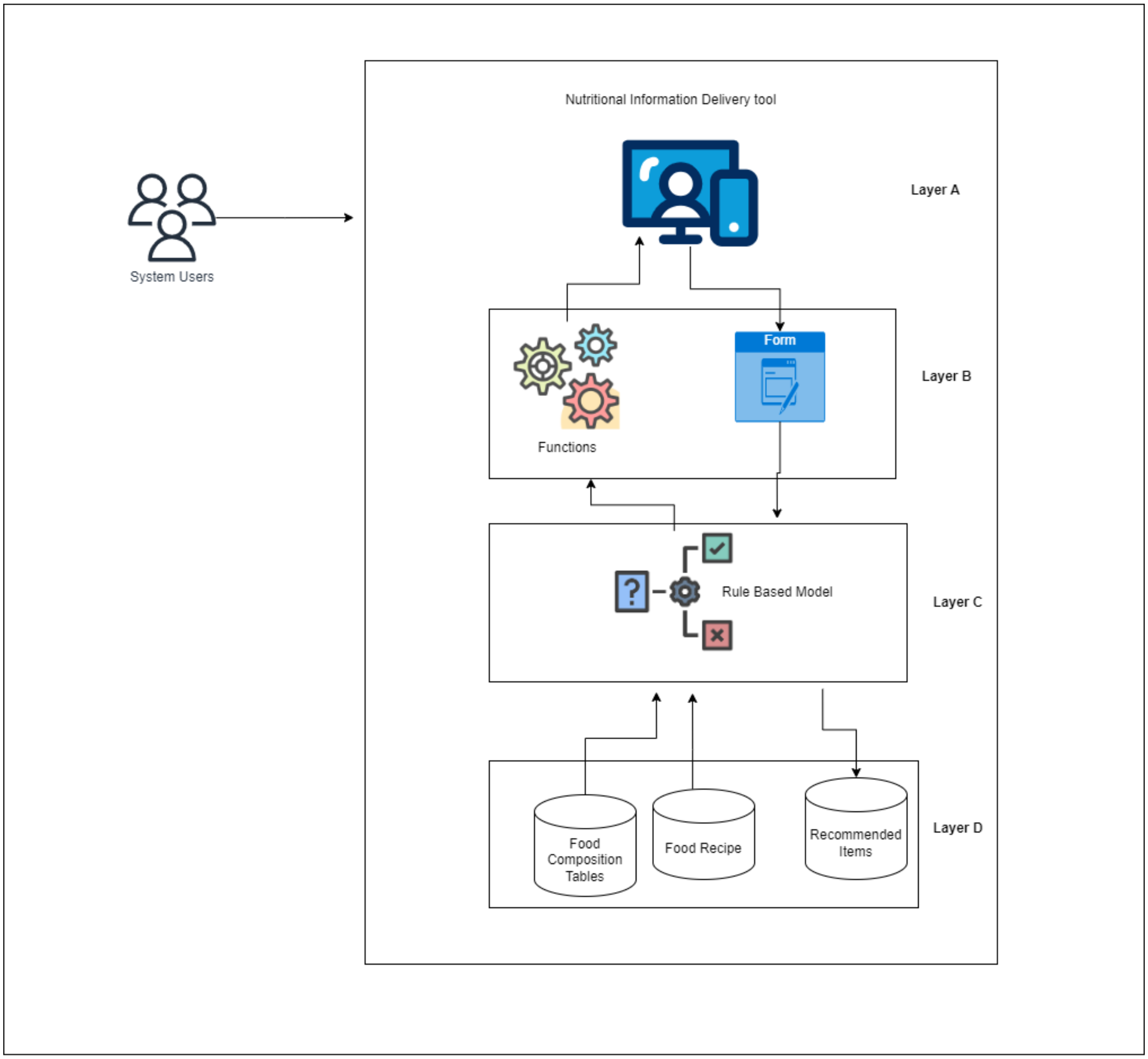


Figure 4.1:System Architecture

4.3.2 Use Case Diagram

This diagram in figure 4.2 shows the interaction between the user and the application tool. This entails the use case and its relation to the actors of the system, which are the users and the model.

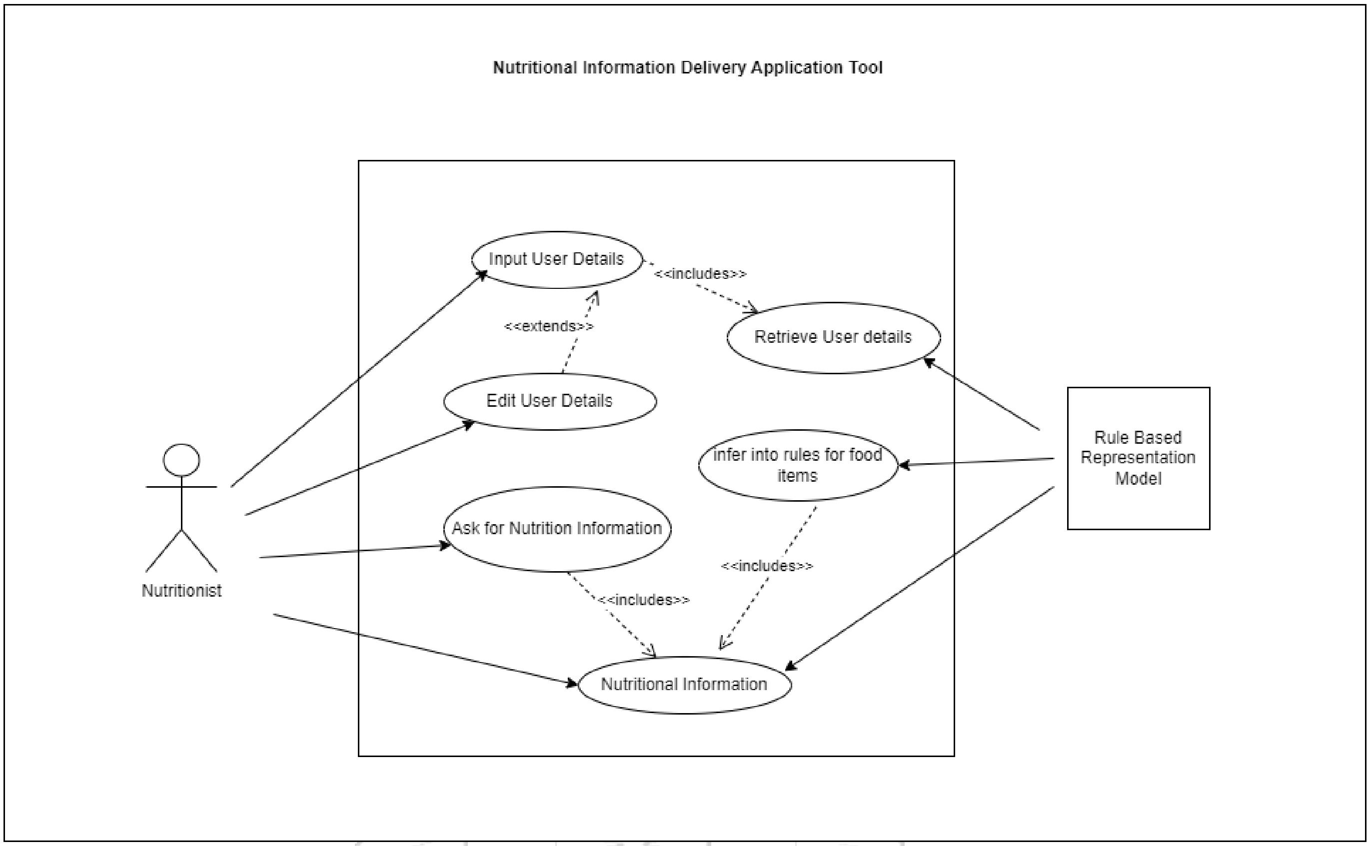


Figure 4.2: Use case diagram

4.3.3 Sequence Diagram

The diagram in figure 4.3 shows how processes interact with each other. It includes the user of the system and various services that are available in the application. The model processes the information and returns the recommended items. The services help in handling data and any errors occurring when in use.

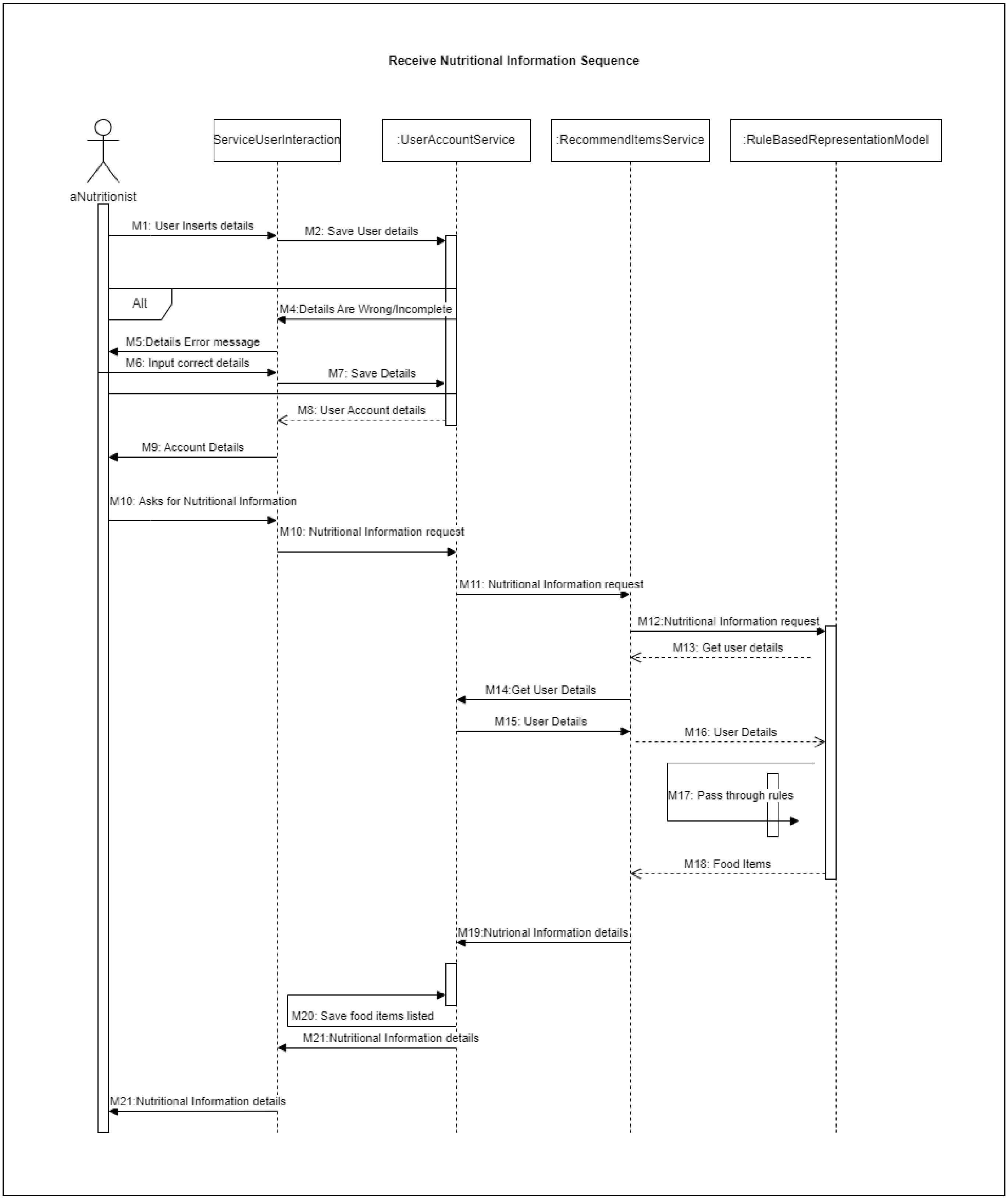


Figure 4.3:Sequence diagram

4.3.4 Class Diagram

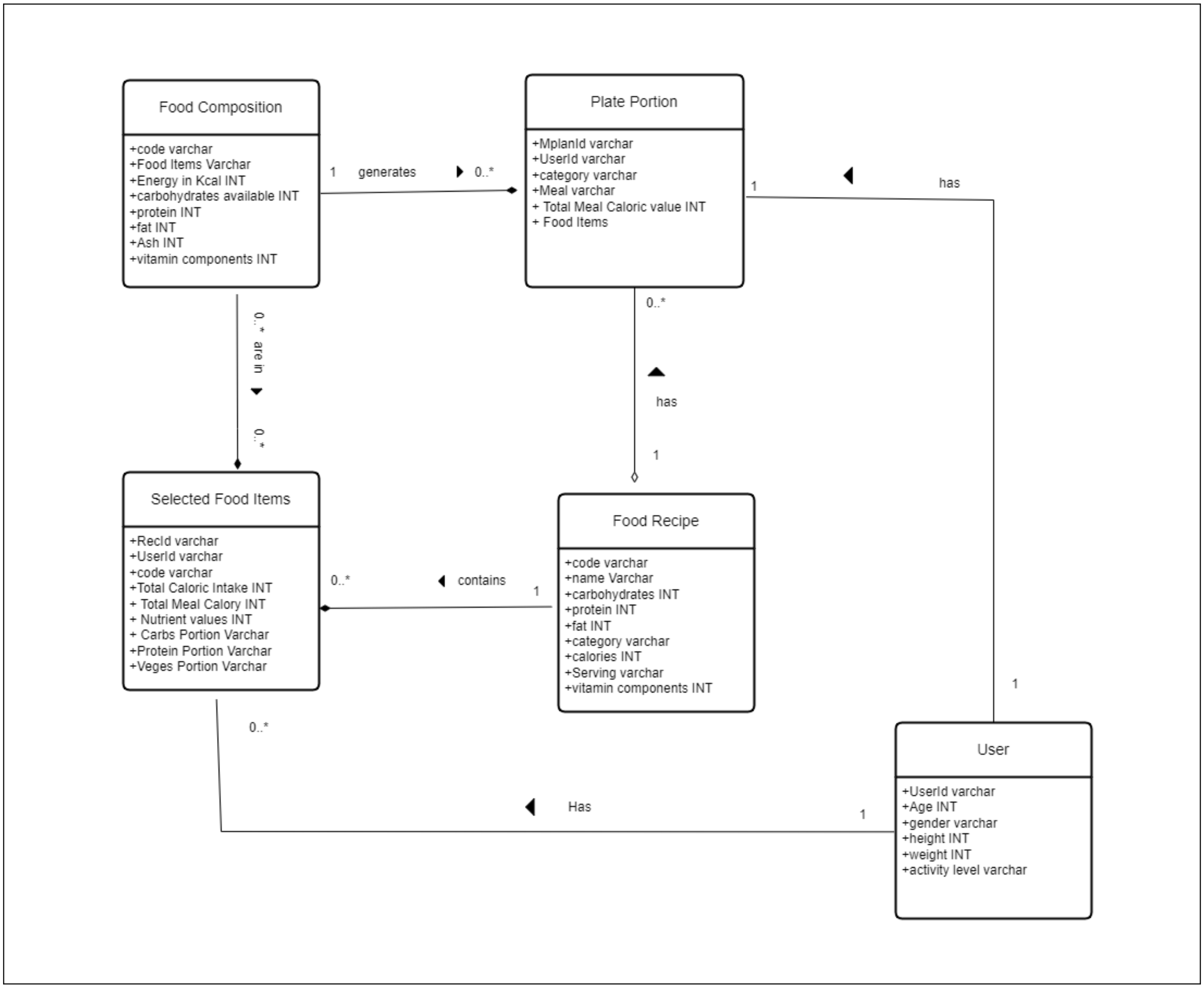


Figure 4.4:Class Diagram

4.3.5 Database Schema

This is an illustration of the database used to store of the data and the relationships to each other.

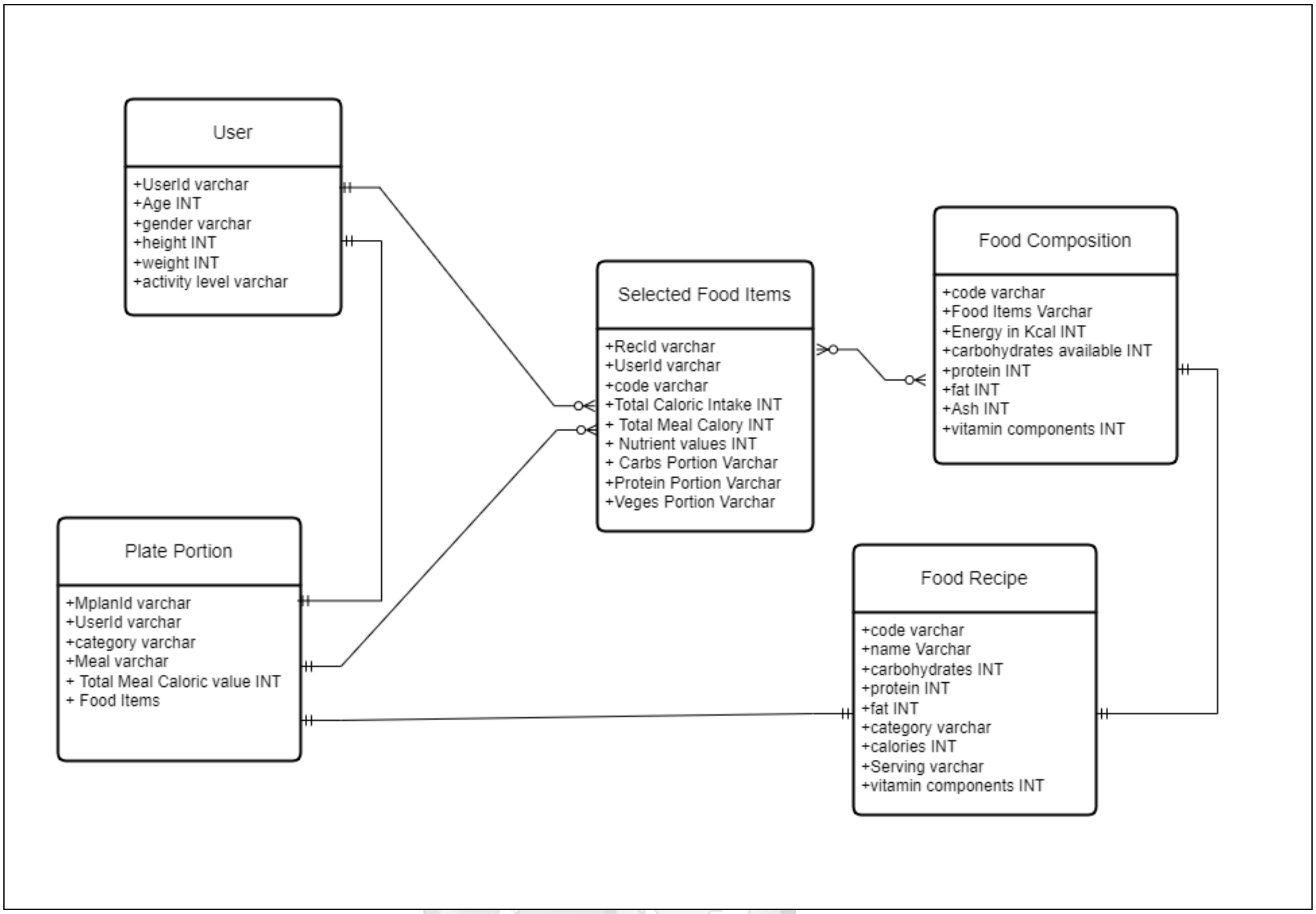


Figure 4.5:Database Schema

4.3.6 Wireframes

These are visual representations of what the application screens will look like. Figure 4.6 represents the sign in and sign in screens. These will take in the user details. Figure 4.7 represents the resulting screen providing the nutritional information using the plate model theory from nutrition therapy (USDA Center for Nutrition Policy and Promotion (CNPP), 2020).

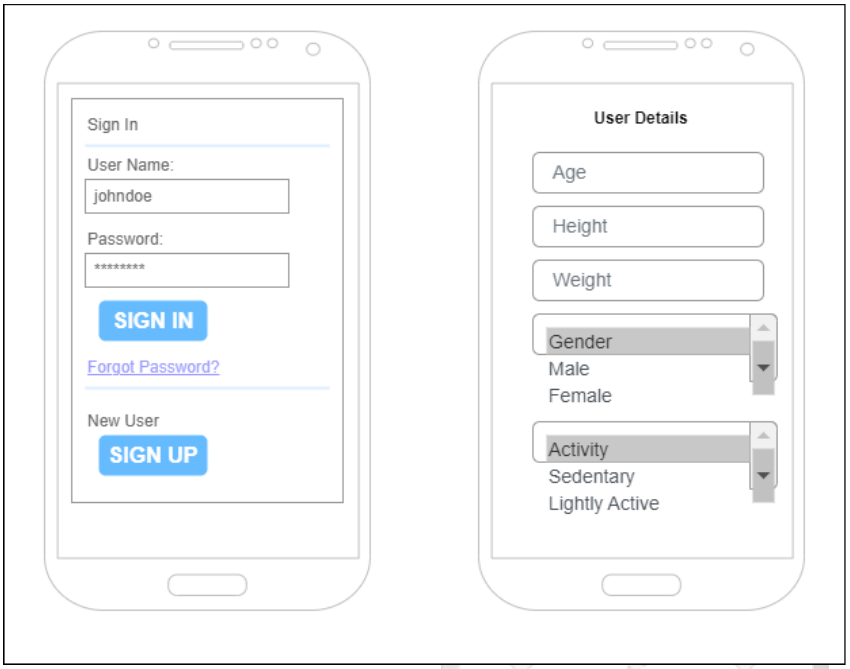


Figure 4.6: Sign in and sign up screens

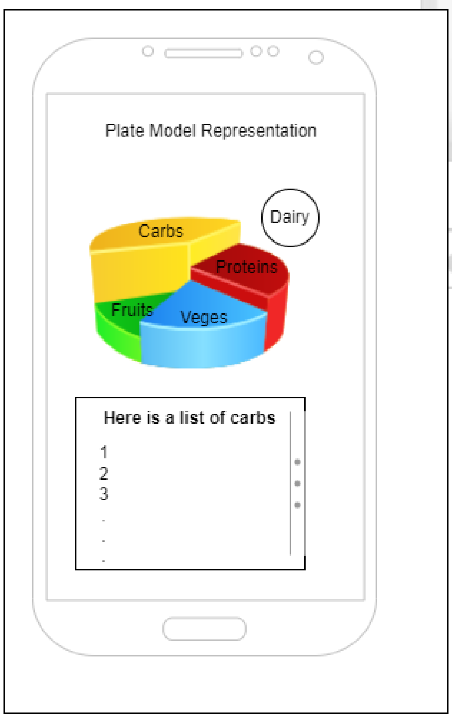


Figure 4.7: Result representation screen

Chapter 5: System Implementation and Testing

5.1 Introduction

This chapter details the implementation and testing of the tool. The development environment and requirements are listed. The implementation process is discussed, from the analysis of the data to the implementation of the rules created for the rule-based representation model. Tests done to validate its functionality are also discussed.

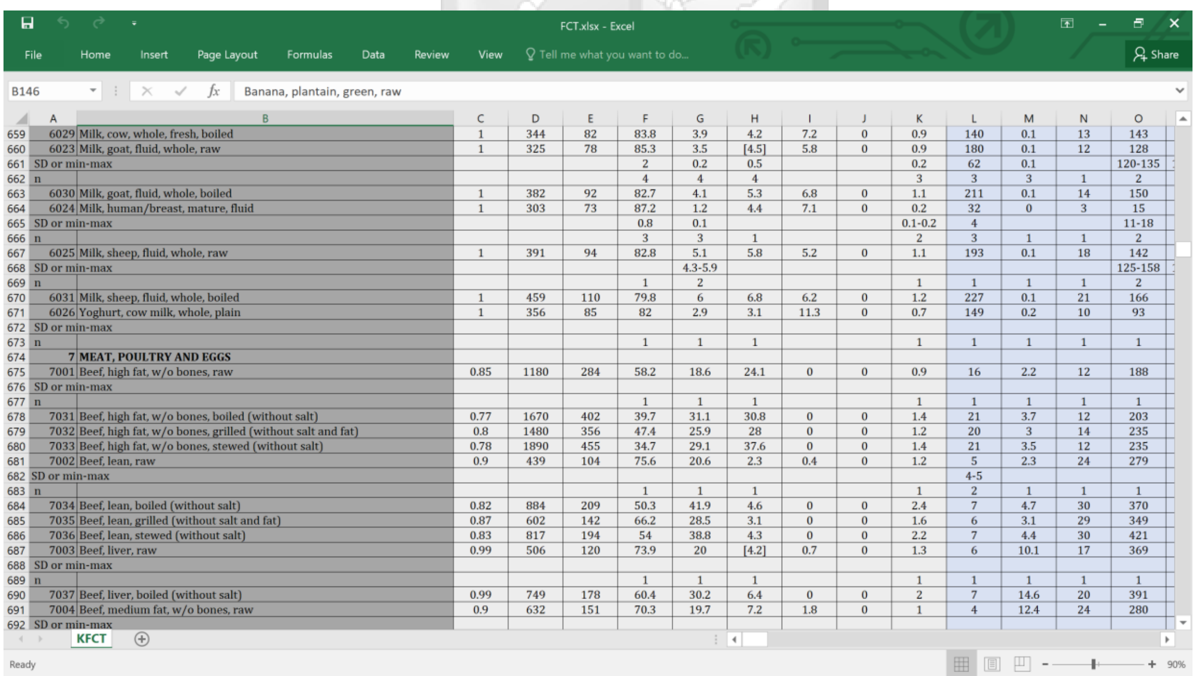
5.2 System Development and Implementation Environment

5.2.1 System Development

The section details the development process of the system. It discusses the process of data analysis and the development of the model including creation of the rules.

5.2.2 Data analysis

The data was extracted from the reports selected using Tabula. This application enables the user to extract tables from pdf files. This information was saved in an excel sheet, containing 1242 rows of data. Figure 5.1 shows a snapshot of the data.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
659	6029	Milk, cow, whole, fresh, boiled	1	344	82	83.8	3.9	4.2	7.2	0	0.9	140	0.1	13	143
660	6023	Milk, goat, fluid, whole, raw	1	325	78	85.3	3.5	[4.5]	5.8	0	0.9	180	0.1	12	128
661	SD or min-max					2	0.2	0.5			0.2	62	0.1		120-135
662	n					4	4	4			3	3	3	1	2
663	6030	Milk, goat, fluid, whole, boiled	1	382	92	82.7	4.1	5.3	6.8	0	1.1	211	0.1	14	150
664	6024	Milk, human/breast, mature, fluid	1	303	73	87.2	1.2	4.4	7.1	0	0.2	32	0	3	15
665	SD or min-max					0.8	0.1				0.1-0.2	4			11-18
666	n					3	3	1			2	3	1	1	2
667	6025	Milk, sheep, fluid, whole, raw	1	391	94	82.8	5.1	5.8	5.2	0	1.1	193	0.1	18	142
668	SD or min-max						4.3-5.9								125-158
669	n					1	2				1	1	1	1	2
670	6031	Milk, sheep, fluid, whole, boiled	1	459	110	79.8	6	6.8	6.2	0	1.2	227	0.1	21	166
671	6026	Yoghurt, cow milk, whole, plain	1	356	85	82	2.9	3.1	11.3	0	0.7	149	0.2	10	93
672	SD or min-max														
673	n					1	1	1			1	1	1	1	1
674	7	**MEAT, POULTRY AND EGGS**													
675	7001	Beef, high fat, w/o bones, raw	0.85	1180	284	58.2	18.6	24.1	0	0	0.9	16	2.2	12	188
676	SD or min-max														
677	n					1	1	1			1	1	1	1	1
678	7031	Beef, high fat, w/o bones, boiled (without salt)	0.77	1670	402	39.7	31.1	30.8	0	0	1.4	21	3.7	12	203
679	7032	Beef, high fat, w/o bones, grilled (without salt and fat)	0.8	1480	356	47.4	25.9	28	0	0	1.2	20	3	14	235
680	7033	Beef, high fat, w/o bones, stewed (without salt)	0.78	1890	455	34.7	29.1	37.6	0	0	1.4	21	3.5	12	235
681	7002	Beef, lean, raw	0.9	439	104	75.6	20.6	2.3	0.4	0	1.2	5	2.3	24	279
682	SD or min-max											4-5			
683	n					1	1	1			1	2	1	1	1
684	7034	Beef, lean, boiled (without salt)	0.82	884	209	50.3	41.9	4.6	0	0	2.4	7	4.7	30	370
685	7035	Beef, lean, grilled (without salt and fat)	0.87	602	142	66.2	28.5	3.1	0	0	1.6	6	3.1	29	349
686	7036	Beef, lean, stewed (without salt)	0.83	817	194	54	38.8	4.3	0	0	2.2	7	4.4	30	421
687	7003	Beef, liver, raw	0.99	506	120	73.9	20	[4.2]	0.7	0	1.3	6	10.1	17	369
688	SD or min-max														
689	n					1	1	1			1	1	1	1	1
690	7037	Beef, liver, boiled (without salt)	0.99	749	178	60.4	30.2	6.4	0	0	2	7	14.6	20	391
691	7004	Beef, medium fat, w/o bones, raw	0.9	632	151	70.3	19.7	7.2	1.8	0	1	4	12.4	24	280
692	SD or min-max														

Figure 5.1: Snapshot of the Kenya Food Composition Table

The excel sheet was then subjected to preprocessing checks. The researcher chose to use R programming language to perform the analysis needed on the files. The libraries readxl, tidyverse, dplyr and fs were used to perform the preprocessing. The data was cleaned to leave only the needed data items. This entailed the removal of columns with nutrients that was not relevant to the study. The columns were renamed appropriately and correctness of values ensured by changing nutrient values into numerical values. Figure 5.2 is the code used to achieve the preprocessing.

```
> library(readxl)
> FCT <- read_excel("C:/Users/USER/PycharmProjects/RBSrules/content/FCT.xlsx")
> View(FCT)
> library(tidyverse)
> library(fs)
> food_dir <- '~/School/Thesis code/Plan D'
> food_items <- as.data.frame(read_excel(paste(food_dir, "FCT.xlsx", sep = "/")))
> head(food_items)
> View(food_items)
> wfood_items <- food_items[!apply(food_items, 1, function(row) any(row == "SD or min-max")), ]
> wfood_items <- food_items[c(1,2,5:12,14,16,17,20,29,30)]
> View(wfood_items)
> colnames(wfood_items)<-c('code', 'Food Item', 'Energy in Kcal', 'Water', 'Protein','fat', 'carbohydrate available', 'Fibre', 'Ash', 'Calcium', 'Magnesium','Potassium', 'Sodium', 'Vitamin A-RAE','Vitamin B12', 'Vitamin C')
> w_food_items <- wfood_items[-c(3,2,1),]
> View(w_food_items)
> sum(is.na(w_food_items$`Food Item`) | (w_food_items$`Food Item`==""))
> wfood_items <- w_food_items[!(is.na(w_food_items$`Food Item`) | w_food_items$`Food Item` == ""), ]
> sum(is.na(w_food_items$`Food Item`) | (w_food_items$`Food Item`==""))
> View(wfood_items)
> sum(duplicated(wfood_items$`Food Item`))
> wfood_items <- wfood_items[!duplicated(wfood_items$`Food Item`), ]
> rm(w_food_items)

> library(dplyr)
> wfood_items <- wfood_items %>% mutate_at(c('Energy in Kcal', 'Water', 'Protein','fat', 'carbohydrate available', 'Fibre', 'Ash', 'Calcium', 'Magnesium','Potassium', 'Sodium', 'Vitamin A-RAE','Vitamin B12', 'Vitamin C'), as.numeric)
```

Figure 5.2: Data cleaning code snippet

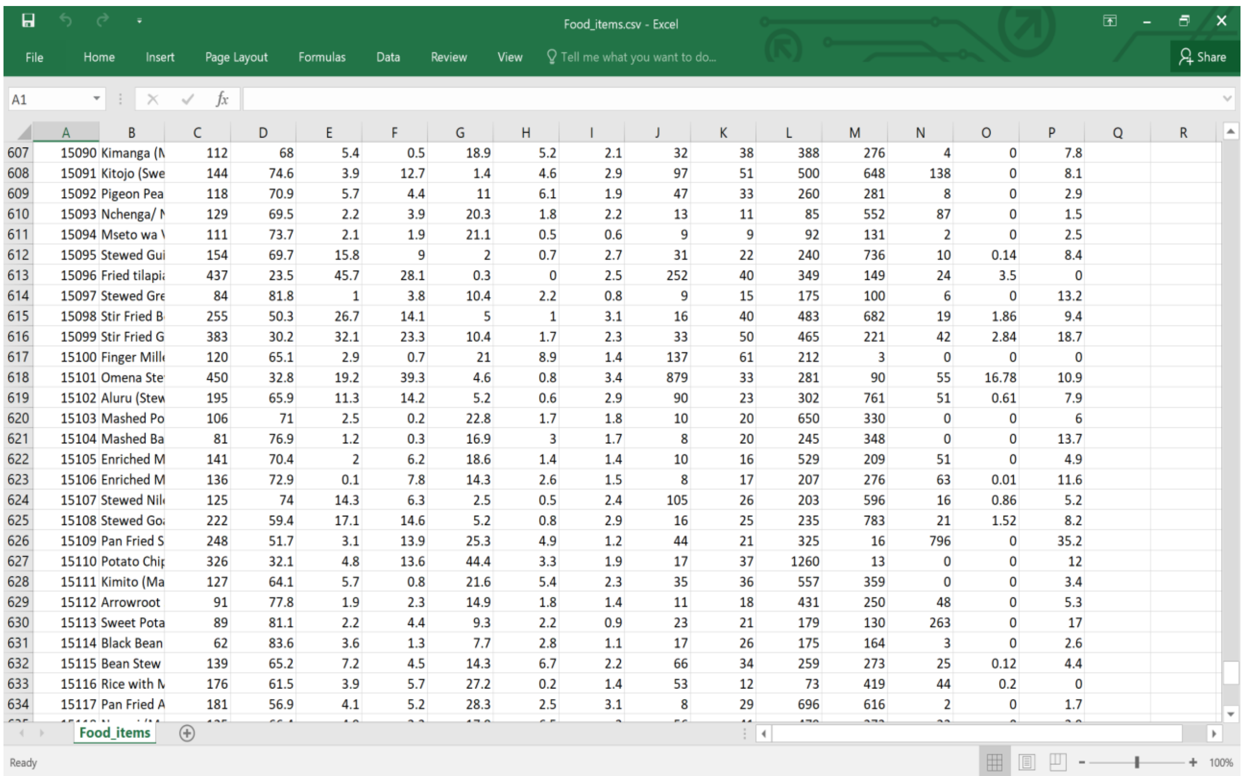
Missing and null values were handled appropriately, as shown in figure 5.3 and code snippets in Appendix B, figure B.1. and the file converted into a csv file for easier manipulation.

```
> food_items_ <- wfood_items[!rownames(wfood_items) %in% c("NA","NA.1", "NA.2", "NA.3", "NA.4", "NA.5", "NA.6", "NA.7", "NA.8", "NA.9", "NA.10", "NA.11", "NA.12", "NA.13", "NA.14"), ]
> sum(is.na(food_items_))
> food_items_[is.na(food_items_)] <- 0

> write.csv(food_items_, paste(food_dir, "Food_items.csv", sep = "/"), row.names = FALSE)
```

Figure 5.3: Handling missing values

Figure 5.4 shows a snapshot of the csv with 16 columns and 657 rows of data.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
607	15090	Kimanga (M	112	68	5.4	0.5	18.9	5.2	2.1	32	38	388	276	4	0	7.8
608	15091	Kitojo (Swe	144	74.6	3.9	12.7	1.4	4.6	2.9	97	51	500	648	138	0	8.1
609	15092	Pigeon Pea	118	70.9	5.7	4.4	11	6.1	1.9	47	33	260	281	8	0	2.9
610	15093	Nchenga/ N	129	69.5	2.2	3.9	20.3	1.8	2.2	13	11	85	552	87	0	1.5
611	15094	Mseto wa V	111	73.7	2.1	1.9	21.1	0.5	0.6	9	9	92	131	2	0	2.5
612	15095	Stewed Gui	154	69.7	15.8	9	2	0.7	2.7	31	22	240	736	10	0.14	8.4
613	15096	Fried tilapi	437	23.5	45.7	28.1	0.3	0	2.5	252	40	349	149	24	3.5	0
614	15097	Stewed Gre	84	81.8	1	3.8	10.4	2.2	0.8	9	15	175	100	6	0	13.2
615	15098	Stir Fried B	255	50.3	26.7	14.1	5	1	3.1	16	40	483	682	19	1.86	9.4
616	15099	Stir Fried G	383	30.2	32.1	23.3	10.4	1.7	2.3	33	50	465	221	42	2.84	18.7
617	15100	Finger Mill	120	65.1	2.9	0.7	21	8.9	1.4	137	61	212	3	0	0	0
618	15101	Omena Ste	450	32.8	19.2	39.3	4.6	0.8	3.4	879	33	281	90	55	16.78	10.9
619	15102	Aluru (Stew	195	65.9	11.3	14.2	5.2	0.6	2.9	90	23	302	761	51	0.61	7.9
620	15103	Mashed Po	106	71	2.5	0.2	22.8	1.7	1.8	10	20	650	330	0	0	6
621	15104	Mashed Ba	81	76.9	1.2	0.3	16.9	3	1.7	8	20	245	348	0	0	13.7
622	15105	Enriched M	141	70.4	2	6.2	18.6	1.4	1.4	10	16	529	209	51	0	4.9
623	15106	Enriched M	136	72.9	0.1	7.8	14.3	2.6	1.5	8	17	207	276	63	0.01	11.6
624	15107	Stewed Nil	125	74	14.3	6.3	2.5	0.5	2.4	105	26	203	596	16	0.86	5.2
625	15108	Stewed Go	222	59.4	17.1	14.6	5.2	0.8	2.9	16	25	235	783	21	1.52	8.2
626	15109	Pan Fried S	248	51.7	3.1	13.9	25.3	4.9	1.2	44	21	325	16	796	0	35.2
627	15110	Potato Chip	326	32.1	4.8	13.6	44.4	3.3	1.9	17	37	1260	13	0	0	12
628	15111	Kimito (Ma	127	64.1	5.7	0.8	21.6	5.4	2.3	35	36	557	359	0	0	3.4
629	15112	Arrowroot	91	77.8	1.9	2.3	14.9	1.8	1.4	11	18	431	250	48	0	5.3
630	15113	Sweet Pota	89	81.1	2.2	4.4	9.3	2.2	0.9	23	21	179	130	263	0	17
631	15114	Black Bean	62	83.6	3.6	1.3	7.7	2.8	1.1	17	26	175	164	3	0	2.6
632	15115	Bean Stew	139	65.2	7.2	4.5	14.3	6.7	2.2	66	34	259	273	25	0.12	4.4
633	15116	Rice with M	176	61.5	3.9	5.7	27.2	0.2	1.4	53	12	73	419	44	0.2	0
634	15117	Pan Fried A	181	56.9	4.1	5.2	28.3	2.5	3.1	8	29	696	616	2	0	1.7

Figure 5.4: Snippet of preprocessed csv

Exploratory analysis to establish relationships was done. This included the descriptive and inferential analysis. The code used to achieve this is seen in figure 5.5.

```
# descriptive analysis
> summary(wfood_items)

# box plot
> boxplot(wfood_items[,3:16])
```

```
# box plot for macro nutrients
> boxplot(wfood_items[,5:7])
#box plot for micros
> boxplot(wfood_items[,10:13])

# correlation-- inferential
> corr_matrix <- cor(wfood_items[,3:16])
library(corrplot)
> corrplot(corr_matrix, method = "color")
> corrplot(corr_matrix, type="lower", method = "square")
> hist(food_items_$`Energy in Kcal`)
```

Figure 5.5: EDA Code snippet

Box plots and histograms were plotted to study these patterns, as seen in the Appendix B, figure B.2. A correlation matrix was used and the output is as seen in figure 5.6 below.

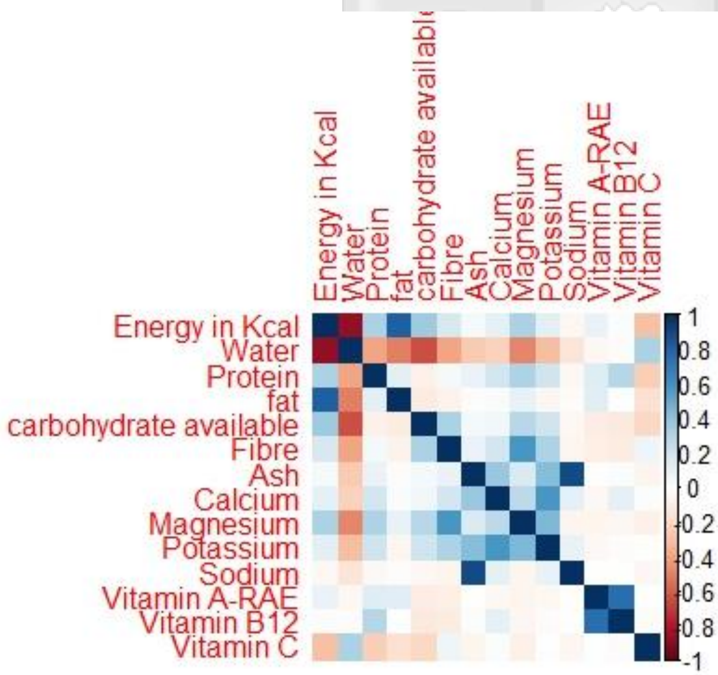


Figure 5.6: Correlation Matrix

5.2.3 Rule-based representation model development

Rules were made from the data points in the final csv file in the snapshot in figure 5.4 above. The study used the guidelines and protocols from the report on nutritional guidance in Kenya. The report put an emphasis on the macronutrients portions. This is due to the influence of these nutrients on the blood glucose level of diabetic patients (Gray & Threlkeld, 2019). The rules that were created to guide the application were focused on the macronutrients. It followed the guidelines on portion control and percentages of macronutrients in each meal. These quantities mentioned relied on the total daily energy intake a person needs. The Basal Metabolic Rate (BMR) is used to calculate the Total Daily Energy Expenditure (TDEE) (Thais Steemburgo et al., 2019). BMR is defined as the energy needed for the body to perform the most basic daily normal activities (Doros et al., 2015). The TDEE is an extension of the BMR, as it multiplies it by a constant depending on the physical activity level of the user (Melanson, 2017). While the BMR is a more accurate way of getting the total energy a patient can intake, constraints such as time, qualified and trained personnel and equipment cost make predictive equations an alternate method of acquiring this information (Thais Steemburgo et al., 2019).

The prediction equations use the demographic features of the user to get the caloric value. These features include the age, gender, height, weight and their activity levels which could range from as active as a professional athlete to a sedentary and low activity lifestyle. The equation (Equation 5.1) utilized is the revised Harris Benedict equation. A study by Thais Steemburgo et al. (2019) revealed that this would be the closest to the clinical measurement that would have been done.

$$BMR_{men} = 88.362 + 13.397 * W + 4.799 * H - 5.677 * A$$

$$BMR_{women} = 447.593 + 9.247 * W + 3.098 * H - 4.33 * A$$

Equation 5.1: The Revised Harris-Benedict Equation

Therefore, the calculated TDEE value was used in the estimation of the total caloric intake. According to Ostendorf et al. (2019), the total energy intake can be interpreted and be represented by the TDEE. The value from this equation was the ‘foundation’ of all the rules. The pseudocode in figure 5.7 shows how the equation 5.1 above was used.

```
FUNCTION calculate_tdee(gender, age, height, weight, activity_levels):
   IF gender.lower() EQUALS 'male' THEN
      // Harris-Benedict equation for men
      bmr = 88.362 + (13.397 * weight) + (4.799 * height) - (5.677 * age)
   ELSE IF gender.lower() EQUALS 'female' THEN
      // Harris-Benedict equation for women
      bmr = 447.593 + (9.247 * weight) + (3.098 * height) - (4.330 * age)
   ELSE
      RAISE ValueError("Gender should be 'male' or 'female'.")

   // Calculate TDEE based on activity level
   tdee = bmr * activity_levels
   RETURN tdee
END FUNCTION
```

Figure 5.7: Total Energy Intake pseudocode

The result from the TDEE was also used to calculate the distribution of calories of each meal taken. An assumption was made, where an individual takes three meals a day and an optional snack. The distribution was done with the percentages, as guided by a study by Institute of Medicine (US) Committee to Review Child and Adult Care Food Program Meal Requirements et al. (2011). It was stated that on average, 21% of total intake is consumed at breakfast, 31% at lunch, 35% at dinner, and the rest (13%) left for snacks. These quantities outlined were used to get the meal caloric values, as shown in the pseudocode in figure 5.8. The meal distribution percentages were put in a dictionary (Appendix B, figure B.8) for use by this pseudocode.

```
FUNCTION calculate_meal_calories(tdee, meal_distribution, macronutrients_distribution):
      meal_calories = {}

  FOR EACH meal, percentage IN meal_distribution.items() DO
      // Calculate calories for the meal based on TDEE and percentages
      meal_calories[meal] = {'calories': tdee * percentage}

  FOR EACH nutrient, (min_percentage, max_percentage) IN macronutrients_distribution.items() DO
         meal_calories[meal][nutrient] = (min_percentage, max_percentage)
      END FOR
   END FOR

RETURN meal_calories
END FUNCTION
```

Figure 5.8: Meal caloric value pseudocode

According to The National Diabetes Prevention and Control Program (2018), portions of carbohydrates should be about 45-60% of total energy intake. This varies from the protein intake recommended at 15-20% while fats consumed should be at less than 35% of the food taken in the day. This was used as reference to create a rule on the portions of food to be served daily. The result from the TDEE was distributed to each meal using the percentages given. Figure 5.9 shows the pseudocode of the same.

```
FUNCTION calculate_macronutrient_components(meal_calories, macronutrients_distribution):
    meal_macronutrients = {}

  // Iterate over each meal and its corresponding calories in the meal_calories dictionary
  FOR EACH meal, calories IN meal_calories.items() DO
        macronutrients = {}

    // Iterate over each macronutrient and its distribution range in the macronutrients_distribution
dictionary
    FOR EACH nutrient, (min_percentage, max_percentage) IN macronutrients_distribution.items() DO
      // Calculate minimum and maximum calories for the macronutrient based on its percentage
distribution
      min_calories = calories * min_percentage
      max_calories = calories * max_percentage

      // Store the calculated minimum and maximum calories for the macronutrient in the macronutrients
dictionary
      macronutrients[nutrient] = (min_calories, max_calories)
    END FOR

    // Store the macronutrients dictionary for the meal in the meal_macronutrients dictionary
    meal_macronutrients[meal] = macronutrients
  END FOR

  // Return the meal_macronutrients dictionary containing macronutrient components for each meal
  RETURN meal_macronutrients
END FUNCTION
```

Figure 5.9: Distribution of macronutrients

These values are then used to create a rule that picks food items within the maximum and minimum grams, after the calories into grams. The pseudocode in figure 5.10 below shows the development code of the same.

```
FUNCTION calculate_macronutrient_components_in_grams(meal_calories):

   meal_macronutrients_grams = {}
   FOR EACH meal, nutrients IN meal_calories.items() DO
      // Initialize an empty dictionary to store macronutrient components in grams for the meal
      macronutrients_grams = {}

           FOR EACH nutrient, percentages IN nutrients.items() DO
         IF nutrient EQUALS 'calories' THEN
            CONTINUE  // Skip 'calories' key
         END IF

         min_percentage, max_percentage = percentages

         min_calories = nutrients['calories'] * min_percentage
         max_calories = nutrients['calories'] * max_percentage

         // Convert minimum and maximum calories to grams using calories_to_grams function
         min_grams = calories_to_grams(min_calories, nutrient)
         max_grams = calories_to_grams(max_calories, nutrient)

         macronutrients_grams[nutrient] = (round(min_grams, 1), round(max_grams, 1))
      END FOR

      meal_macronutrients_grams[meal] = macronutrients_grams
   END FOR

   RETURN meal_macronutrients_grams
END FUNCTION

FUNCTION calories_to_grams(calories, nutrient):
   IF nutrient EQUALS 'carbs' OR nutrient EQUALS 'proteins' THEN
      conversion_factor = 4  // kcal/g for carbs and proteins
   ELSE IF nutrient EQUALS 'fat' THEN
      conversion_factor = 9  // kcal/g for fat
   ELSE
      RAISE ValueError("Invalid nutrient type")
   END IF

   RETURN calories / conversion_factor
END FUNCTION
```

Figure 5.10: Conversion of macronutrients from to grams

The code snippets, figure B. 3 to figure B.6, show the raw code used to create these rules. Figure 5.11 shows the result of a test case range used.

```
Enter your age: 26
Enter your height in cm: 167
Enter your weight in kg: 70
Enter your gender (male/female): male
Enter your activity level (sedentary/lightly active/moderately active/very active/physical job/professional athlete): lightly active
Total Daily Energy:  2309.9766250000002
Meal Macronutrients in Grams:
Meal: Breakfast
carbs: 54.6g - 72.8g
proteins: 18.2g - 24.3g
fat: 10.8g - 18.9g
Meal: Lunch
carbs: 80.6g - 107.4g
proteins: 26.9g - 35.8g
fat: 15.9g - 27.8g
Meal: Dinner
carbs: 91.0g - 121.3g
proteins: 30.3g - 40.4g
fat: 18.0g - 31.4g
Meal: Snacks
carbs: 33.8g - 45.0g
proteins: 11.3g - 15.0g
fat: 6.7g - 11.7g
```

Figure 5.11: Result from the rule based representation model

5.2.4 System Development Environment

The application was developed on hardware, software and cloud services that were selected to ensure the smooth operation of the application. These are listed below.

5.2.5 Hardware requirements

Hardware	Description
Lenovo Thinkpad X1 Yoga	This system is running on an Intel® Core™ i7- -7600U CPU @ 2.80GHz 2.90 GHz, with 16.0 GB RAM on a 64-bit operating system, x64-based processor

Table 5.1: Hardware Requirements:

5.2.6 Software requirements

Software	Description
Operating System	Windows 10 and above
IDE	Pycharm Community Edition 2023.3.5/ R studio
Programming language	Python 3.11

Table 5.2: Software Requirements

5.2.7 Cloud Service Requirements

Service	Description
Google Firebase Database	This is a real-time dynamic NoSQL database that is provided and hosted by Google

Table 5.3: Cloud Service Requirements

5.3 System Implementation

The National Diabetes Prevention and Control Program (2018) recommended various tools to be used as meal planning in nutrition therapy. Among them is the Plate Model. The plate model is a visual meal planning tool that uses a literal plate to advise on nutrient intake for diabetic patients (USDA Center for Nutrition Policy and Promotion (CNPP), 2020). This tool is a commonly used in various nations. The rules created are used to implement this tool. The result contains a list of food items that are in the ranges calculated for each individual. For the test range in figure 5.11, a list of items was picked. Figure 5.12 show a snapshot of these items. These items were satisfying the rules and the ranges calculated for carbohydrates and proteins.

```
  Cereals and Cereal products, Legumes and pulses: Amaranth, whole grain, dry, raw, Amaranth, whole grain, flour, Biscuit, Savoury,
 Biscuit, Sweet, Breakfast cereal, wheat biscuits, Weetabix type, Cake, Fruit, Maize, grain, white variety, whole, dry, raw, Maize,
 grain, yellow variety, whole, dry, raw, Maize meal, sifted, fortified, packaged, raw, Maize, White, Degermed (Muthokoi), Maize, wh
ole, flour, raw, Millet, bulrush, flour, Millet, bulrush, grain, dry, raw, Oat bran, unprocessed, uncooked, Pasta, spaghetti, dry,
raw-imported, Semolina, Sorghum, Grain, Red, Dried, Raw, Sorghum, Grain, Red, Flour, Sorghum, grain, white, dry, raw, Sorghum, grai
n, white, flour, Teff, raw, Wheat Flour-Atta, Wheat Flour (refined/fortified/sifted packaged), raw, Wheat, whole, flour, raw, Wheat
, whole, grain, dry, raw
  Legumes and pulses, Meat, poultry and Eggs: Beans, kidney, dry, raw, Beans, lima, dry, raw, Bonavist, dry, raw, Chick peas, whole
, dry, raw, Garden peas, dry, raw, Gram, Flour, Pigeon peas, dry, raw, Beef, high fat, w/o bones, raw     , Beef, lean, raw, Beef,
liver, raw, Beef, medium fat, w/o bones, raw, Camel Meat, Chicken, unespecified part, w/o bone, meat&skin, raw, Chicken, unspecifie
d part, w/o bone, meat&skin, boiled (without salt), Chicken, unspecified part, w/o bone, meat&skin, grilled (without salt and fat),
 Goat, lean, raw, Goat liver, raw, Goat, medium fat, raw, Guinea fowl, meat, with skin, Lamb liver, raw, Pork, meat, raw (unspecifi
ed part), Quail, flesh & skin, raw
Lunch:
  Vegetable and vegetable products: No food items selected
  Cereals and Cereal products, Legumes and pulses: Breakfast cereal, flakes of corn, fortified (iron, thiamine, riboflavin, niacin
and folate), Breakfast cereal, flakes of corn, Cornflour, from maize starch
  Legumes and pulses, Meat, poultry and Eggs: Gram, green, dry, raw, Lentils, whole, dry, raw, Soybean, dry, raw, Beef, high fat, w
/o bones, boiled (without salt), Beef, high fat, w/o bones, stewed (without salt), Beef, lean, grilled (without salt and fat), Beef
, liver, boiled (without salt), Beef, medium fat, without bones, boiled (without salt), Beef, medium fat, without bones, grilled (w
ithout salt and fat), Beef, medium fat, without bones, stewed (without salt), Goat, lean, boiled (without salt), Goat, lean, grille
d (without salt and fat), Goat, liver, boiled (without salt), Goat, medium fat, boiled (without salt), Goat, medium fat, grilled (w
ithout salt and fat), Lamb, liver, boiled (without salt), Pork, meat, unspecified part, boiled (without salt), Pork, meat, unspecif
ied part, grilled (without salt and fat), Rabbit meat, stewed (without salt), Rabbit meat, roasted (without salt and fat)
Dinner:
  Vegetable and vegetable products: No food items selected
  Cereals and Cereal products, Legumes and pulses: No food items selected
  Legumes and pulses, Meat, poultry and Eggs: Flour, soya, full fat, Soybean, dry, raw, Beef, high fat, w/o bones, boiled (without
salt), Beef, lean, stewed (without salt), Beef, medium fat, without bones, boiled (without salt), Beef, medium fat, without bones,
stewed (without salt), Goat, liver, boiled (without salt), Lamb, liver, boiled (without salt), Rabbit meat, stewed (without salt)
Snacks:
  Vegetable and vegetable products: No food items selected
  Cereals and Cereal products, Legumes and pulses: Beans, broad, dry, raw, Beans, kidney, dry, raw, Beans, lima, dry, raw, Chick pe
as, whole, dry, raw, Cowpeas, dry, raw, Gram, black, dry, raw, Gram, green, dry, raw, Lentils, whole, dry, raw, Pigeon peas, dry, r
aw
  Legumes and pulses, Meat, poultry and Eggs: Bean, red, fresh, raw, Soybean, dry, unsoaked, boiled, drained (without salt), Soybea
```

Figure 5.12: Snapshot of Nutritional Information

The total intake caloric values are divided into the portions guided by the plate method, where half the meal calories are vegetables or fruits, one quarter is foods rich in carbohydrates and the other is foods rich in protein (USDA Center for Nutrition Policy and Promotion (CNPP), 2020). Additionally, the data was categorized so that the right items are picked as shown in figure 5.13.

```
FUNCTION categorize_food_item(code):
    code = int(code)

    WHILE True DO
        IF 1001 <= code <= 2000 THEN
            RETURN "Cereals and Cereal products"
        ELSE IF 2001 <= code <= 3000 THEN
            RETURN "Starchy roots, tubers and bananas"
        ELSE IF 3001 <= code <= 4000 THEN
            RETURN "Legumes and pulses"
        ELSE IF 4001 <= code <= 5000 THEN
            RETURN "Vegetable and vegetable products"
        ELSE IF 5001 <= code <= 6000 THEN
            RETURN "Fruits"
        ELSE IF 6001 <= code <= 7000 THEN
            RETURN "Milk and dairy products"
        ELSE IF 7001 <= code <= 8000 THEN
            RETURN "Meat, poultry and Eggs"
        ELSE IF 8001 <= code <= 9000 THEN
            RETURN "Fish and sea food"
        ELSE IF 9001 <= code <= 10000 THEN
            RETURN "Oils and fat"
        ELSE IF 10001 <= code <= 11000 THEN
            RETURN "Nuts and seeds"
        ELSE IF 12001 <= code <= 13000 THEN
            RETURN "Beverages"
        ELSE IF 13001 <= code <= 14000 THEN
            RETURN "Condiments and spices"
        ELSE IF 14001 <= code <= 15000 THEN
            RETURN "Insects"
        ELSE IF 15001 <= code <= 16000 THEN
            RETURN "Mixed dishes"
        ELSE
            RETURN "Sugar and sweetened products"
        END IF
    END WHILE
END FUNCTION
```

Figure 5.13: Categorizing food items

This helped in picking the right items with the right macronutrients for each particular portion. For each portion, only items in these categories were picked, as shown on the Appendix B, figure B.9 and figure 5.14.

```
categorized_items = {}

FOR EACH food, code IN food_items.items() DO
    categorized_items[food] = categorize_food_item(code)
END FOR

FOR EACH food, category IN categorized_items.items() DO
    food_categories[category].append(food)
END FOR

// Define portions and their corresponding categories and macronutrients
portions = [    {'categories': ['Vegetable and vegetable products'], 'macronutrients': None},
    {'categories': ['Cereals and Cereal products', 'Legumes and pulses'], 'macronutrients': 'carbs'},
    {'categories': ['Legumes and pulses', 'Meat, poultry and Eggs'], 'macronutrients': 'proteins'}
]
```

Figure 5.14: Categorized food items

After user details are input, the model picks various food items that is the nutritional information used to advise the patient on what to eat, as shown in Appendix B, figure B.8. and the pseudocode in figure 5.12. The caloric values of macronutrients are also displayed, and help ease the nutrition therapy process.

```
meal_plan = {}

FOR EACH meal, macronutrient_allocations IN meal_macronutrients_grams.items() DO
    meal_plan[meal] = {}

    FOR EACH portion IN portions DO
            selected_food_items = []

        IF portion['macronutrients'] IS NULL THEN
            portion_size = meal_calories[meal]['calories'] / 2
        ELSE
            nutrient_to_use = portion['macronutrients']
            min_grams, max_grams = macronutrient_allocations[nutrient_to_use]

            FOR EACH category IN portion['categories'] DO
                FOR EACH food_item IN food_categories[category] DO
                    food_nutrients = macronutrient_content.get(food_item, {})
                    IF min_grams <= food_nutrients.get(nutrient_to_use, 0) <= max_grams THEN
                        // Add the food item to the list of selected food items
                        selected_food_items.append(food_item)
                    END IF
                END FOR
            END FOR
        END IF
```

```
      meal_plan[meal][', '.join(portion['categories'])] = selected_food_items
    END FOR
END FOR
```

Figure 5.15: Portion pseudocode

5.4 System Testing and Validation

Evaluation and testing were done to ensure the seamless functionality of the nutrition delivery tool. These tests helped in resolving and in elimination of errors bypassed during the development stage.

5.4.1 Functional requirements

Test Cases	Pre-condition	Test Data	Expected Results	Pass/Fail
The user can input details		Age, weight, height, gender and activity level values	The values are saved in the database	Pass
TDEE can be calculated and output	The user has input their details	User details	The caloric value is displayed	Pass
The meal calories have been calculated	The TDEE has been calculated	User details	The meal calories are output for each of the four meals	Pass
Food items recommendations is based on nutrients	The TDEE and meal calories have been calculated	User details	The food items are displayed for each meal within a range of maximum and minimum values	Pass
Portions on my plate are displayed	The nutrient distribution has been done	User details	The portions return various food items	Pass
Portions meet the caloric values	The food items have been recommended	User details	The food items returned meet the caloric value	Pass

Table 5.4: Test Case for functional requirements

Chapter 6: Discussion

6.1 Overview

This chapter discusses the results of the study in relation to its objectives, as listed in section 1.3.2. The aim of the research was to develop a context-specific and customizable nutritional information delivery application for Type II diabetic patients that provides suitable information based on a user's needs.

6.2 Review of research objectives

6.2.1 The current methods for provision of nutrition information to Type II diabetic patients in Kenya

Non-pharmacological management methods were found to be the primary recommendation offered to type II diabetes patients. From the literature review, it was discovered that patients got their nutritional from the health workers. The information was minimal, scattered and not easily available (Kiberenge et al., 2010). According to the (The National Diabetes Prevention and Control Program, 2018), dieticians are to encouraged to factor in the patients such as culture and socio-economic factors. Equipping them with a tool to access and use to advise their patients.

6.2.2 Techniques used to design context-specific and customizable applications

The second objective was to analyse the techniques used to design context specific and customizable applications. The literature explored various used techniques and were used to inform the technique used in the study. Various methods to create the model were discussed, including key- value model, ontology-based models and markup scheme models. The analysis of these methods indicated that the rule based representation model had various advantages to it. These included the elements that help in its working, and the reasoning support it offers for its applications (Nalepa & Bobek, 2014). The literature stage revealed the advantages of customizable applications such as considering the environment and its ability to handle different users (Salber et al., 2004, Cuddy et al., 2005).

6.2.3 A context-specific and customizable nutritional information delivery application for Type II diabetic patients

This objective focused on the development of a context- specific and customizable nutritional information delivery application. The application is to help guide the nutritionist catering to diabetic patients on the delivery of nutritional information. The provision of information in a centralized and easily accessible would help in the management of type II diabetes. Existing literature were used to guide the researcher in understanding in a contextual manner how the information is provisioned, what kind of information is provisioned, the actors involved and any existing protocols that guide in the dissemination of the information.

6.2.4 Validation of the performance of the application

This objective was to validate the performance of the model. The performance was evaluated using performance metrics that gained insight on its overall effectiveness. The effectiveness in which it acquires input and its output was evaluated. The results of these tests, as evidence in section 5.4 show that this could be an effective tool to deliver nutritional information for type II diabetes.

Chapter 7: Conclusion and Recommendations

7.1 Conclusion

Management of type II diabetes in Kenya has been affected by factors such as the scarcity of nutritionists, an imbalance between the diabetic patients and their nutritionists and the minimal and unavailable needed nutritional information. This study has reviewed the methods that are currently used in provision nutritional information. As lifestyle modification remains the first and most recommended methodology used to manage type II diabetes, it is important that nutritional information be readily available to them. The tool developed aims to provide this information, factoring in the user's needs. The study made use of locally available data and report guidelines in development of the tool. This would help the local users access this information with food items that they are familiar with and have access to. This addresses the need for solutions that consider the socio-economic and cultural values of the community.

7.2 Recommendations

This study developed a tool that can acquire context factors such as time, the user's profile and needs. The study made use of local datasets, the Kenya Food Composition with 10,512 data items and the Kenya Recipe book with 2603 data points. An increase of the food items available locally would improve the recommendations offered to patients. This would also increase the number of food recommendations applications available with local food, not just for diabetic patients but for the entire population looking into living healthier.

References

Bansal, N. (2015). Prediabetes diagnosis and treatment: A review. *World Journal of Diabetes*, *6*(2), 296–296. https://doi.org/10.4239/wjd.v6.i2.296

Bene, B. A., O'Connor, S., Mastellos, N., Majeed, A., Fadahunsi, K. P., & O'Donoghue, J. (2019). Impact of mobile health applications on self-management in patients with type 2 diabetes mellitus: protocol of a systematic review. *BMJ Open*, *9*(6), e025714. https://doi.org/10.1136/bmjopen-2018-025714

Borse, S., Abu Sufiyan Chhipa, Sharma, V., Singh, D. P., & Manish Nivsarkar. (2020). Management of Type 2 Diabetes: Current Strategies, Unfocussed Aspects, Challenges, and Alternatives. *Medical Principles and Practice*, *30*(2), 109–121. https://doi.org/10.1159/000511002

Buthpitiya, S., Luqman, F., Griss, M. L., & Dey, A. K. (2012, March). *Hermes -- A Context-Aware Application Development Framework and Toolkit for the Mobile Environment*. ResearchGate; unknown. https://www.researchgate.net/publication/241629618_Hermes_--_A_Context-Aware_Application_Development_Framework_and_Toolkit_for_the_Mobile_Environment

Cuddy, S. M., Katchabaw, M., & Lutfiyya, H. (2005). Context-aware service selection based on dynamic and static service attributes. *IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*, *4*, 13–20. https://doi.org/10.1109/wimob.2005.1512944

Dey, A. K. (2001). Understanding and Using Context. *Personal and Ubiquitous Computing*, *5*(1), 4–7. https://doi.org/10.1007/s007790170019

Doros, R., Delcea, A., Mardare, L., & Petcu, L. (2015). *BASAL METABOLIC RATE IN METABOLIC DISORDERS*. https://www.academia.edu/download/101290953/art05Doros.pdf

FAO, & Government of Kenya. (2018). *FOOD COMPOSITION GOVERNMENT OF KENYA*. https://www.fao.org/3/I8897EN/i8897en.pdf

FAO, Ministry of Health , & Ministry of Agriculture and Irrigation. (2018). Kenyan Food Recipes. In *Ministry of Agriculture, Livestock, Fisheries and Cooperatives*. https://kilimo.go.ke/wp-content/uploads/2021/01/KENYA-FOOD-RECIPES-2018.pdf

Farinde , A. (2021, October 17). *Oral Hypoglycemic Agents: Oral Antihyperglycemic Drugs*. Medscape.com; Medscape. https://emedicine.medscape.com/article/2172160-overview#a1

Fedele, D. A., Cushing, C. C., Fritz, A. M., Amaro, C. M., & Ortega, A. E. (2017). Mobile Health Interventions for Improving Health Outcomes in Youth. *JAMA Pediatrics*, *171*(5), 461–461. https://doi.org/10.1001/jamapediatrics.2017.0042

Gervais, J. B. (2023, April 24). *Your Guide To The Best Diabetes Apps In 2023* (J. Newgent, Ed.). Forbes Health. https://www.forbes.com/health/conditions/diabetes/best-diabetes-apps/

Ghiani, G., Manca, M., Fabio Paternò, & Santoro, C. (2017, April 27). *Personalization of Context-Dependent Applications Through Trigger-Action Rules*. ResearchGate; Association for Computing Machinery. https://www.researchgate.net/publication/316570711_Personalization_of_Context-Dependent_Applications_Through_Trigger-Action_Rules

Golafshani, N. (2015). Understanding Reliability and Validity in Qualitative Research. *The Qualitative Report*, *8*(4). https://doi.org/10.46743/2160-3715/2003.1870

Gray, A., & Threlkeld, R. J. (2019a, October 13). *Nutritional Recommendations for Individuals with Diabetes*. Nih.gov; MDText.com, Inc. https://www.ncbi.nlm.nih.gov/books/NBK279012/

Gray, A., & Threlkeld, R. J. (2019b, October 13). *Nutritional Recommendations for Individuals with Diabetes*. Nih.gov; MDText.com, Inc. https://www.ncbi.nlm.nih.gov/books/NBK279012/

Hampton, J. A. (2003). Abstraction and context in concept representation. *Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences*, *358*(1435), 1251–1259. https://doi.org/10.1098/rstb.2003.1314

Harris, A. D., McGregor, J. C., Perencevich, E. N., Furuno, J. P., Zhu, J., Peterson, D., & Finkelstein, J. (2006). The Use and Interpretation of Quasi-Experimental Studies in Medical Informatics. *Journal of the American Medical Informatics Association*, *13*(1), 16–23. https://doi.org/10.1197/jamia.m1749

Hayes-Roth, F. (1985). Rule-based systems. *Communications of the ACM*, *28*(9), 921–932. https://doi.org/10.1145/4284.4286

Institute of Medicine (US) Committee to Review Child and Adult Care Food Program Meal Requirements, Murphy, S. P., Yaktine, A. L., West Suitor, C., & Moats, S. (2011). *Process for Developing Recommendations for Meal Requirements*. Nih.gov; National Academies Press (US). https://www.ncbi.nlm.nih.gov/books/NBK209815/

International Diabetes Federation. (2021). *IDF Diabetes Atlas 10th edition* . https://diabetesatlas.org/idfawp/resource-files/2021/07/IDF_Atlas_10th_Edition_2021.pdf

Jones, T. (2013, August). *Diabetes Mellitus: the increasing burden of disease in Kenya*. Southsudanmedicaljournal.com. http://www.southsudanmedicaljournal.com/archive/august-2013/diabetes-mellitus-the-increasing-burden-of-disease-in-kenya.

Kanafani, K., Regitze Kjær Zimmermann, Freja Nygaard Rasmussen, & Harpa Birgisdóttir. (2021). Learnings from Developing a Context-Specific LCA Tool for Buildings—The Case of LCAbyg 4. *Sustainability*, *13*(3), 1508–1508. https://doi.org/10.3390/su13031508

Karega, L. A. (2022). *A Location-aware nutritional needs prediction tool for type II Diabetic patients: case Kenya*. Strathmore.edu. https://su-plus.strathmore.edu/items/5d5fc28d-64fd-433b-8a5c-55950f814397

Kariuki, E. M. (2021). Food recommender system for Diabetes Type 2 patients. In *su-plus.strathmore.edu*. http://hdl.handle.net/11071/12750

Katambo, D. (2021, June 23). *Diabetes Statistics in Kenya - Afyacode Diabetes Management*. Afyacode LTD. https://afyacode.com/diabetes-statistics-in-kenya/

Khardori, R. (2023, May). *Type 2 Diabetes Mellitus: Practice Essentials, Background, Pathophysiology*. Medscape.com; Medscape. https://emedicine.medscape.com/article/117853-overview

Kiberenge, M. W., Ndegwa, Z. M., Njenga, E. W., & Muchemi, E. W. (2010). Knowledge, attitude and practices related to diabetes among community members in four provinces in Kenya: a cross-sectional study. *The Pan African Medical Journal*, *7*, 2. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3172632/

König, L. M., Attig, C., Franke, T., & Renner, B. (2021). Barriers to and Facilitators for Using Nutrition Apps: Systematic Review and Conceptual Framework. *JMIR MHealth and UHealth*, *9*(6), e20037–e20037. https://doi.org/10.2196/20037

Kothari, C. R. (2014). Research Methodology Methods and Techniques. In *Scirp.org* (2nd ed., pp. 58–65). New Age International Publishers. https://ccsuniversity.ac.in/bridge-library/pdf/Research-Methodology-CR-Kothari.pdf

Kumar, R., Maheshwary, P., & Malche, T. (2019). Inside Agile Family Software Development Methodologies. *International Journal of Computer Sciences and Engineering*, *7*(6), 650–660. https://doi.org/10.26438/ijcse/v7i6.650660

Liu, Z., Chen, C., Wang, J., Xing, C., Huang, Y.-K., Hu, J., & Wang, Q. (2023). Fill in the Blank: Context-aware Automated Text Input Generation for Mobile GUI Testing. *ArXiv (Cornell University)*. https://doi.org/10.1109/icse48619.2023.00119

Makkonen, J., Avdouevski, I., Kerminen, R., & Visa, A. (2009). Context Awareness in Human-Computer Interaction. *InTech EBooks*. https://doi.org/10.5772/7743

Marín-Peñalver, J. J., Martín-Timón, I., Sevillano-Collantes, C., & Javier, F. (2016). Update on the treatment of type 2 diabetes mellitus. *World Journal of Diabetes*, *7*(17), 354–354. https://doi.org/10.4239/wjd.v7.i17.354

Ministry of Health. (2021). National Strategic Plan for the Prevention and Control of Non-Communicable Diseases 2021/22 - 2025/26. In *Guidelines, Standards & Policies Portal*. http://guidelines.health.go.ke:8000/media/National_Strategic_Plan_NCD_Prevention_and_Control_2021-22__2025-26.pdf

Ministry of Public Health and Sanitation. (2010a). KENYA NATIONAL DIABETES STRATEGY 2010-2015. In *World Diabetes Foundation*. https://www.worlddiabetesfoundation.org/sites/default/files/WDF09-436%20Kenya%20National%20Diabetes%20Strategy%202010-2015%20%20-%20Complete.pdf

Ministry of Public Health and Sanitation. (2010b). REPUBLIC OF KENYA NATIONAL CLINICAL GUIDELINES FOR MANAGEMENT OF DIABETES MELLITUS FIRST EDITION. In *World Diabetes Foundation*. https://www.worlddiabetesfoundation.org/sites/default/files/WDF09-436%20National%20Clinical%20Guidelines%20for%20Management%20of%20Diabetes%20Melitus%20-%20Complete.pdf

Mohamed, S. F., Mwangi, M., Mutua, M. K., Kibachio, J., Hussein, A., Ndegwa, Z., Scholastica Owondo, Gershim Asiki, & Kyobutungi, C. (2018). Prevalence and factors associated with pre-diabetes and diabetes mellitus in Kenya: results from a national survey. *BMC Public Health*, *18*(S3). https://doi.org/10.1186/s12889-018-6053-x

Mohsen Tavakol, & Dennick, R. (2011). Making sense of Cronbach's alpha. *International Journal of Medical Education*, *2*, 53–55. https://doi.org/10.5116/ijme.4dfb.8dfd

Mpa, S. (2023). *Blood Glucophytes Levels*. S Glucotypes Data.

Musumba, G., & Nyongesa, H. (2013, May 23). *Context awareness in mobile computing: A review*. ResearchGate; AOSIS. https://www.researchgate.net/publication/272898383_Context_awareness_in_mobile_computing_A_review

Nalepa, G., & Bobek, S. (2014). Rule-based solution for context-aware reasoning on mobile devices. *Computer Science and Information Systems, 11*(1), 171–193. https://doi.org/10.2298/csis130209002n

Narayan, K. V. (2016). The mysteries of type 2 diabetes in developing countries. *Bulletin of the World Health Organization*, *94*(4), 241–242. https://doi.org/10.2471/blt.16.030416

NATIONAL FOOD AND NUTRITION SECURITY POLICY. (2011). http://www.nutritionhealth.or.ke/wp-content/uploads/Downloads/NFNSP%20SESSIONAL%20PAPER%2022nd%20June%202011.pdf

NHS Choices. (2023). *Eating a balanced diet*. NHS. https://www.nhs.uk/Live-well/eat-well/how-to-eat-a-balanced-diet/eating-a-balanced-diet/

NIDDK. (2023). *Diabetes Diet, Eating, & Physical Activity*. National Institute of Diabetes and Digestive and Kidney Diseases; NIDDK - National Institute of Diabetes and Digestive and Kidney Diseases. https://www.niddk.nih.gov/health-information/diabetes/overview/diet-eating-physical-activity

Nutritionists and Dieticians Act. (2012). *NUTRITIONISTS AND DIETICIANS ACT CHAPTER 253B*. https://faolex.fao.org/docs/pdf/ken128724.pdf

Omwenga, V. (2016). *Context-Centred Mobile Applications Development For Effective Adoption Of Mobile Technology*. *18*(1), 1-07. https://doi.org/10.9790/0661-18140107

Ostendorf, D. M., Caldwell, A. E., Creasy, S. A., Pan, Z., Lyden, K., Bergouignan, A., MacLean, P. S., Wyatt, H. R., Hill, J. O., Melanson, E. L., & Catenacci, V. A. (2019). Physical Activity Energy

Expenditure and Total Daily Energy Expenditure in Successful Weight Loss Maintainers. *Obesity*, *27*(3), 496–504. https://doi.org/10.1002/oby.22373

Phrashiah Githinji, Dawson, J. A., Appiah, D., & Rethorst, C. D. (2022). A Culturally Sensitive and Theory-Based Intervention on Prevention and Management of Diabetes: A Cluster Randomized Control Trial. *Nutrients*, *14*(23), 5126–5126. https://doi.org/10.3390/nu14235126

Poppendieck, M., & Cusumano, M. A. (2012). Lean Software Development: A Tutorial. *IEEE Software*, *29*(5), 26–32. https://doi.org/10.1109/ms.2012.107

Puska, P. (2002). Nutrition and global prevention on non-communicable diseases. *Asia Pacific Journal of Clinical Nutrition*, *11*(s9), S755–S758. https://doi.org/10.1046/j.1440-6047.11.s.5.x

Salber, D., Dey, A. K., & Abowd, G. (2004, February 16). *Designing and Building Context-Aware Applications*. ResearchGate; unknown. https://www.researchgate.net/publication/2905237_Designing_and_Building_Context-Aware_Applications

Schilit, B. N., Hilbert, D. M., & Trevor, J. (2002). Context-aware communication. *IEEE Wireless Communications*, *9*(5), 46–54. https://doi.org/10.1109/mwc.2002.1043853

Shahzeydi, M., & Gandomani, T. J. (2016, June). *Adding lean principles to agile software development: A case study report*. ResearchGate; unknown. https://www.researchgate.net/publication/311322764_Adding_lean_principles_to_agile_software_development_A_case_study_report

Shannon, G., Haghparast-Bidgoli, H., Chelagat, W., & Kibachio, J. (2019). Innovating to increase access to diabetes care in Kenya: an evaluation of Novo Nordisk's base of the pyramid project. *Global Health Action*, *12*(1), 1605704–1605704. https://doi.org/10.1080/16549716.2019.1605704

Shiroya, V., Neuhann, F., Müller, O., & Deckert, A. (2019). Challenges in policy reforms for non-communicable diseases: the case of diabetes in Kenya. *Global Health Action*, *12*(1), 1611243. https://doi.org/10.1080/16549716.2019.1611243

Sridevi, S., Bhattacharya, S., & Pitchiah, R. (2012). Context Aware Framework. *Springer EBooks*, *73*, 358–363. https://doi.org/10.1007/978-3-642-29154-8_40

Thais Steemburgo, Lazzari, C., Juliano Boufleur Farinha, Pedroso, T., Luciana Vercoza Viana, Reischak, A., & de, J. (2019). Basal metabolic rate in Brazilian patients with type 2 diabetes: comparison between measured and estimated values. *Archives of Endocrinology and Metabolism (Impresso)*, *63*(1), 53–61. https://doi.org/10.20945/2359-3997000000103

Thanasegaran, G. (2009). Reliability and Validity Issues in Research. *Integration & Dissemination*, *4*, 35–40. Business Source Complete.

The National Diabetes Prevention and Control Program. (2018a). KENYA NATIONAL CLINICAL GUIDELINES FOR THE MANAGEMENT OF DIABETES MELLITUS. In *Non-communicable Diseases Alliance Kenya (NCDAK)* (pp. 43–60). https://ncdak.org/wp-content/uploads/2021/07/Kenya-National-Clinical-Guidelines-for-the-Management-of-Diabetes-2nd-Editiion-2018-1.pdf

The National Diabetes Prevention and Control Program. (2018b). KENYA NATIONAL CLINICAL GUIDELINES FOR THE MANAGEMENT OF DIABETES MELLITUS. In *Non-communicable Diseases Alliance Kenya (NCDAK)* (pp. 43–60). https://ncdak.org/wp-content/uploads/2021/07/Kenya-National-Clinical-Guidelines-for-the-Management-of-Diabetes-2nd-Editiion-2018-1.pdf

Thyer, B. A. (2012). Quasi-Experimental Research Designs. In *Google Books*. Oxford University Press. https://books.google.com/books?hl=en&lr=&id=MBcSDAAAQBAJ&oi=fnd&pg=PP1&dq=quasi+experimental+research+design&ots=j8t5hVjZxY&sig=4Lul-08hf4oOuxKwy6AJ9NVNH7o

Toshi, N. (2022, January 5). *Diabetes Food Chart-The best food for diabetes | PharmEasy* (R. Budania & A. Shah, Eds.). Diabetes. https://pharmeasy.in/conditions/diabetes/diabetes-food-chart-indian/#:~:text=Generally%2C%20experts%20suggest%20a%201200,their%20calories%20from%20complex%20carbs.

Unai Gangoiti, Lopez, A., Aintzane Armentia, Elisabet Estévez, Oskar Casquero, & Marcos, M. (2022). A Customizable Architecture for Application-Centric Management of Context-Aware Applications. *IEEE Access*, *10*, 1603–1625. https://doi.org/10.1109/access.2021.3138586

UNESCO. (2014, November 28). *Peri-Urban Landscapes; Water, Food and Environmental Security*. UNESCO. https://en.unesco.org/events/peri-urban-landscapes-water-food-and-environmental-security

UNESCO. (2023). *Kenya launches five-year plan to control non-communicable diseases*. Unesco.org. https://www.unesco.org/en/articles/kenya-launches-five-year-plan-control-non-communicable-diseases

UNICEF. (2021, April). *Non-communicable diseases* . UNICEF DATA. https://data.unicef.org/topic/child-health/noncommunicable-diseases/

USDA Center for Nutrition Policy and Promotion (CNPP). (2020). *What Is MyPlate?* Myplate.gov. https://www.myplate.gov/eat-healthy/what-is-myplate

Vahdat-Nejad, H. (2019). *Context-Aware Middleware: A Review*. https://arxiv.org/ftp/arxiv/papers/1905/1905.11365.pdf

Vu, T.-M.-H., Minh, C., Le Dinh, T., Anh, N., Dam, K., Pham-Nguyen, C., & Minh, H. (n.d.). *Context-aware Knowledge-based Systems: A Literature Review*. https://scholarspace.manoa.hawaii.edu/server/api/core/bitstreams/cb9096f8-0488-4417-8c25-0d33f7fcf6a7/content

Wainaina, P. (2022, December 25). *System design, UML diagrams(A summary) - Peter Wainaina - Medium*. Medium; Medium. https://medium.com/@wainaina.pierre/system-design-uml-diagrams-a-summary-39cf6c9ab63c

Walker, K. Z., O'Dea, K., Gomez, M., Girgis, S., & Colagiuri, R. (2010). Diet and exercise in the prevention of diabetes. *Journal of Human Nutrition and Dietetics*, *23*(4), 344–352. https://doi.org/10.1111/j.1365-277x.2010.01061.x

Walliman, N. (2021). *Research Methods* (3rd ed.). https://doi.org/10.4324/9781003141693

Wang, H., Li, Y., Du, M., Li, Z., & Jin, D. (2021). App2Vec: Context-Aware Application Usage Prediction. *ACM Transactions on Knowledge Discovery from Data*, *15*(6), 1–21. https://doi.org/10.1145/3451396

Watts, M. (2022, September 8). *Diabetes Nutrition*. Diabetes. https://www.diabetes.co.uk/nutrition.html

WHO. (2020). *NONCOMMUNICABLE DISEASES PROGRESS MONITOR 2020*. https://apps.who.int/iris/bitstream/handle/10665/330805/9789240000490-eng.pdf?sequence=1&isAllowed=y

WHO. (2023, April 5). *Diabetes*. Who.int; World Health Organization: WHO. https://www.who.int/news-room/fact-sheets/detail/diabetes

World Health Organization. (2011). *Glycated haemoglobin (HbA1c) for the diagnosis of diabetes*. Nih.gov; World Health Organization. https://www.ncbi.nlm.nih.gov/books/NBK304271/

Wu, Y., Ding, Y., Tanaka, Y., & Zhang, W. (2014). Risk Factors Contributing to Type 2 Diabetes and Recent Advances in the Treatment and Prevention. *International Journal of Medical Sciences*, *11*(11), 1185–1200. https://doi.org/10.7150/ijms.10001

Zhang, D., Adipat, B., & Mowafi, Y. (2009). User-Centered Context-Aware Mobile Applications—The Next Generation of Personal Mobile Computing. *Communications of the Association for Information Systems*, *24*. https://doi.org/10.17705/1cais.02403

Appendix A: Ethical Clearance Certificate

Strathmore UNIVERSITY

27th March 2024

Ms Muti Zainabu
zinabu.muti@strathmore.edu

Dear Ms Muti,

RE: A Context-Specific and Customizable Nutritional Information Delivery Application for Type II Diabetic Patients

This is to inform you that SU-ISERC has reviewed and **approved** your above **SU-masters** research proposal. Your application reference number is **SU-ISERC2007/24.** The approval period is from **27th March 2024 to 26th March 2025.**
This approval is subject to compliance with the following requirements:

i. Only approved documents including (informed consents, study instruments, MTA) will be used.
ii. All changes including (amendments, deviations, and violations) are submitted for review and approval by SU-ISERC.
iii. Death and life-threatening problems and serious adverse events or unexpected adverse events whether related or unrelated to the study must be reported to SU-ISERC within 72 hours of notification.
iv. Any changes anticipated or otherwise that may increase the risks or affected safety or welfare of study participants and others or affect the integrity of the research must be reported to SU-ISERC within 72 hours.
v. Clearance for the export of biological specimens must be obtained from relevant institutions.
vi. Submission of a request for renewal of approval at least 60 days prior to the expiry of the approval period. Attach a comprehensive progress report to support the renewal.
vii. Submission of an executive summary report within 90 days of completion of the study to SU-ISERC.

Before commencing your study, you will be expected to obtain a research license from National Commission for Science, Technology, and Innovation (NACOSTI) https://research-portal.nacosti.go.ke/ and obtain other clearances needed.

Yours sincerely,

Mr Ambrose Rachier,
Chairperson; SU-ISERC

STRATHMORE UNIVERSITY INSTITUTIONAL SCIENTIFIC AND ETHICAL REVIEW COMMITTEE
(SU-ISERC)
27-Mar-2024
Email:ethicsreview@strathmore.edu
P.O BOX 59857-00200
NAIROBI-KENYA

Ole Sangale Rd, Madaraka Estate. PO Box 59857-00200, Nairobi, Kenya. Tel +254 (0)703 034000
Email admissions@strathmore.edu www.strathmore.edu

Appendix B: System Implementation Code Snippets.

```r
library(readxl)
FCT <- read_excel("C:/Users/USER/PycharmProjects/RBSrules/content/FCT.xlsx")
View(FCT)
library(tidyverse)
library(fs)
food_dir <- '~/School/Thesis code/Plan D'
food_items <- as.data.frame(read_excel(paste(food_dir, "FCT.xlsx", sep = "/")))
head(food_items)
View(food_items)
wfood_items <- food_items[!apply(food_items, 1, function(row) any(row == "SD or min-max")), ]
wfood_items <- food_items[c(1,2,5:12,14,16,17,20,29,30)]
View(wfood_items)
colnames(wfood_items)<-c('code', 'Food Item', 'Energy in Kcal', 'Water', 'Protein','fat', 'carbohydrate available'
w_food_items <- wfood_items[-c(3,2,1),]
View(w_food_items)
sum(is.na(w_food_items$`Food Item`) | (w_food_items$`Food Item`==""))
wfood_items <- w_food_items[!(is.na(w_food_items$`Food Item`) | w_food_items$`Food Item` == ""), ]
sum(is.na(w_food_items$`Food Item`) | (w_food_items$`Food Item`==""))
View(wfood_items)
sum(duplicated(wfood_items$`Food Item`))
wfood_items <- wfood_items[!duplicated(wfood_items$`Food Item`), ]
rm(w_food_items)

str(wfood_items)
library(dplyr)
wfood_items <- wfood_items %>% mutate_at(c('Energy in Kcal', 'Water', 'Protein','fat', 'carbohydrate available', '
str(wfood_items)

food_items_ <- wfood_items[!rownames(wfood_items) %in% c("NA","NA.1", "NA.2", "NA.3", "NA.4", "NA.5", "NA.6", "NA.
sum(is.na(food_items_))
food_items_[is.na(food_items_)] <- 0

write.csv(food_items_, paste(food_dir, "Food_items.csv", sep = "/"), row.names = FALSE)
```

Figure B. 1: Preprocessing Data Snippet

Source on Save

```r
# descriptive analysis
summary(wfood_items)

# box plot
boxplot(wfood_items[,3:16])

# box plot for macro nutrients
boxplot(wfood_items[,5:7])

#box plot for micros
boxplot(wfood_items[,10:13])

# correlation-- inferential
corr_matrix <- cor(wfood_items)
corr_matrix <- cor(wfood_items[,3:16])
library(corrplot)
corrplot(corr_matrix, method = "color")
corrplot(corr_matrix, type="lower", method = "square")
corrplot(corr_matrix,na.label=" ")

hist(food_items_$`Energy in Kcal`)
```

Figure B. 2: Data Analysis snippet

```python
2 usages   new *
def calculate_tdee(gender, age, height, weight, activity_levels):
    if gender.lower() == 'male':
        # Harris-Benedict equation for men
        bmr = 88.362 + (13.397 * weight) + (4.799 * height) - (5.677 * age)
    elif gender.lower() == 'female':
        # Harris-Benedict equation for women
        bmr = 447.593 + (9.247 * weight) + (3.098 * height) - (4.330 * age)
    else:
        raise ValueError("Gender should be 'male' or 'female'.")

    # Calculate TDEE based on activity level
    tdee = bmr * activity_levels
    return tdee
```

Figure B. 3: Calculation of Energy Intake

```python
1 usage   new *
def calculate_meal_calories(tdee, meal_distribution, macronutrients_distribution):
    meal_calories = {}
    for meal, percentage in meal_distribution.items():
        meal_calories[meal] = {'calories': tdee * percentage}
        # Add macronutrient percentages for the meal
        for nutrient, (min_percentage, max_percentage) in macronutrients_distribution.items():
            meal_calories[meal][nutrient] = (min_percentage, max_percentage)
    return meal_calories


# Macronutrient Components for Each Meal
new *
def calculate_macronutrient_components(meal_calories, macronutrients_distribution):
    meal_macronutrients = {}
    for meal, calories in meal_calories.items():
        macronutrients = {}
        for nutrient, (min_percentage, max_percentage) in macronutrients_distribution.items():
            min_calories = calories * min_percentage
            max_calories = calories * max_percentage
            macronutrients[nutrient] = (min_calories, max_calories)
        meal_macronutrients[meal] = macronutrients
    return meal_macronutrients
```

Figure B. 4: Calculation of meal calories and nutrient distribution

```python
def recommend_food_items(tdee, meal_distribution, food_categories, meal_categories, food_dataset_file):
    recommended_food_items = {}
    lunch_items = set()  # Keep track of items selected for lunch
    dinner_items = set()  # Keep track of items selected for dinner

    for meal, categories in meal_categories.items():
        print(f"Recommended Food Items for {meal}:")
        recommended_food_items[meal] = []
        energy_per_meal = tdee * meal_distribution[meal]

        # Keep track of selected items from exclusion categories
        first_excluded = False
        second_excluded = False

        # Keep track of selected categories and items to ensure only one item per category is selected
        selected_categories = set()
        selected_items = set()

        # Exclude items already selected for the other meal when recommending for the current meal
        if meal == 'Lunch':
            categories = [cat for cat in categories if cat not in dinner_items]
        elif meal == 'Dinner':
            categories = [cat for cat in categories if cat not in lunch_items]
```

Figure B. 5: Food Item recommendation

```python
1 usage  new *
def calculate_macronutrient_components_in_grams(meal_calories):
    meal_macronutrients_grams = {}
    for meal, nutrients in meal_calories.items():
        macronutrients_grams = {}
        for nutrient, percentages in nutrients.items():
            if nutrient == 'calories':
                continue  # Skip 'calories' key
            min_percentage, max_percentage = percentages
            min_calories = nutrients['calories'] * min_percentage
            max_calories = nutrients['calories'] * max_percentage
            min_grams = calories_to_grams(min_calories, nutrient)
            max_grams = calories_to_grams(max_calories, nutrient)
            macronutrients_grams[nutrient] = (round(min_grams, 1), round(max_grams, 1))
        meal_macronutrients_grams[meal] = macronutrients_grams
    return meal_macronutrients_grams

2 usages  new *
def calories_to_grams(calories, nutrient):
    if nutrient == 'carbs' or nutrient == 'proteins':
        conversion_factor = 4  # kcal/g for carbs and proteins
    elif nutrient == 'fat':
        conversion_factor = 9  # kcal/g for fat
    else:
        raise ValueError("Invalid nutrient type")
    return calories / conversion_factor
```

Figure B. 6: Caloric conversions

```python
# Initialize meal plan dictionary
meal_plan = {}
for meal, macronutrient_allocations in meal_macronutrients_grams.items():
    meal_plan[meal] = {}

    for portion in portions:
        selected_food_items = []

        if portion['macronutrients'] is None:
            #   portion_size = meal_calories[meal]/ 2 change this
                portion_size = meal_calories[meal]['calories'] / 2

        else:
            nutrient_to_use = portion['macronutrients']
            min_grams, max_grams = macronutrient_allocations[nutrient_to_use]
            for category in portion['categories']:

                for food_item in food_categories[category]:
                    food_nutrients = macronutrient_content.get(food_item, {})

                    if min_grams <= food_nutrients.get(nutrient_to_use, 0) <= max_grams:
                        selected_food_items.append(food_item)

        # print("Selected food items:", selected_food_items)
        meal_plan[meal][', '.join(portion['categories'])] = selected_food_items
```

Figure B. 7:Portion snippet

```python
# recommended nutrient distribution
nutrient_distribution = {
    'carbs': (0.45, 0.60),
    'proteins': (0.15, 0.20),
    'fat': (0.20, 0.35)
}

# meal distribution percentages
meal_distribution = {
    'Breakfast': 0.21,
    'Lunch': 0.31,
    'Dinner': 0.35,
    'Snacks': 0.13
}

# Activity levels
activity_levels = {
    'sedentary': 1.2,
    'lightly active': 1.375,
    'moderately active': 1.55,
    'very active': 1.725,
    'physical job': 1.9,
    'professional athlete': 2.4
}
```

Figure B. 8: Dictionaries used

```python
# Grouping food items into categories
categorized_items = {food: categorize_food_item(code) for food, code in food_items.items()}

# Append food items to their respective categories
for food, category in categorized_items.items():
    food_categories[category].append(food)

# Define portions and their corresponding categories and macronutrients
portions = [
    {'categories': ['Vegetable and vegetable products'], 'macronutrients': None},
    {'categories': ['Cereals and Cereal products', 'Legumes and pulses'], 'macronutrients': 'carbs'},
    {'categories': ['Legumes and pulses', 'Meat, poultry and Eggs'], 'macronutrients': 'proteins'}
]
```

Figure B. 9: Categorizing the portions.