



INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Nutrition Informatics App

1Dr.A.V.Senthil Kumar, 2Mr.siva.A,

1Director,MCA,M.Phil,PGDCA,Ph.D, 2Final MCA Student

1PG & Research Department Of Computer Applications(MCA),

1Hindusthan College of Arts and Science(Autonomous), Coimbatore, India

ABSTRACT

The project entitled “**Nutrition Informatics App**” is a mobile application which has been designed by using Android Studio with Java as front end tool and Google Firebase as back end tool. Nutrition informatics is the intersection of information, nutrition and technology. This Food Nutrition Informatics app is mainly used to evaluate the daily foods intake by the user and compare the food intake nutritive value including minerals and vitamins details. These details will get compared with the Recommended Dietary Allowance (RDA). The RDA is the average daily level of intake sufficient to meet the nutrient requirements of nearly all (97%-98%) healthy people.

The proposed mobile app has been designed with several modules such as User Registration, User Login, User Daily intake of food, Comparison with RDA and display the higher and lower level of minerals and vitamins. The app calculates the nutritive value of Energy, Carbohydrate, Protein, Fat, Total Fibre, Soluble fibre and Insoluble fibre. And also calculate the Vitamins level such as Thiamine(B1), Riboflavin (B2), Niacin (B3), Pantothenic Acid (B5), Pyridoxine (B6), Biotin(B7), Folate (B9), Vitamin A(Retinol), Vitamin C(Ascorbic Acid), Vitamin D(Calciferol), Vitamin E(tocopherol) and Vitamin K (Phylloquinone). Furthermore the app also calculate the minerals value such as Calcium, Phosphorus(P), Iron (Fe), Sodium(Na), Potassium(K), Zinc(Zn), Magnesium(Mg), Copper (cu), Chromium(Cr) and Selenium.

The calculated nutritive value will get compared with the standard RDA according to the age group and gender standard values. The comparative result will be displayed to the user with the interpretation and inference result. The mobile app has been designed with two roles such as user role and administrator role. The administrator will update the RDA details and nutritive details of selected food details on Indian food. These details will be more helpful to the user to control the food habit and turn into a healthy dietary chart for a long healthy and happy life.

CHAPTER -1

1. INTRODUCTION

1.1 PROBLEM STATEMENT

In a family there may be infants, young children, adolescent, adults, and elderly all living under one roof. It is always a big challenge to provide a good nutritious diet for each of the member, particularly when their needs, preferences vary. Humans need a wide range of nutrients to lead a healthy and active life. The amount of each nutrient needed for an individual depends on age, body weight, physical activity, physiological state (pregnancy, lactation) etc. So basically the requirement for nutrients varies from individual to individual. A proposed app is needed which calculates the level of the nutrient consumed daily by the user and suggest dietary suggestions by comparing with Recommended Dietary Allowances (RDA) values.

1.3 MODULES DESCRIPTION

1. Administrator – Nutritive Value

a. Once the administrator get login to the main page, the main portion of the app is to fillup the nutritive values. The nutritive values for all types of foods including South Indian foods will be entered in this form. Each nutritive value includes the nutritive value of Energy, Carbohydrate, Protein, Fat, Total Fibre, Soluble fibre and Insoluble fibre. And also calculate the Vitamins level and mineral levels.

2. Administrator - Recommended Dietary Allowances (RDA)

a. Recommended Dietary Allowances (RDAs) are the levels of intake of essential nutrients that, on the basis of scientific knowledge, are judged by the Food and Nutrition Board to be adequate to meet the known nutrient needs of practically all healthy persons. In the RDA, the values of vitamins, general nutritive value and minerals will be maintained according to gender wise, BMI wise, age group wise. This form includes the special fields such as babies, pregnant ladies also.

3. User Signup and Login

a. In the User Signup and Login form, the user has been allowed to simply enter the email id and password to signup their details. Once the details entered, the user account will get created. Later, the registered user can get log into the main form by just entering the registered email id and password.

4. User – Input Daily Food Intake

a. The Registered users can input the daily breakfast, lunch, dinner and snacks intaken during the whole day in this form. These details will evaluated with the RDA and nutritive value calculation. The evaluated data has been stored in the database.

5. User – Food Recall

- a. In the Food Recall system, the user can see the present nutritive value position of the whole day food intaken. This form will display which is highest and lowest comparing to the nutritive value and RDA values. This form will also displays the interpretation result with suggestions.

6. User - View Nutritive Values

- a. The user can view the nutritive values of the entire foods stored by the administrator. Each nutritive value includes the nutritive value of Energy, Carbohydrate, Protein, Fat, Total Fibre, Soluble fibre and Insoluble fibre. And also calculate the Vitamins level and mineral levels.

7. User – View Recommended Dietary Allowances (RDA)

- a. The users can also check their Recommended Dietary Allowances (RDAs) such as the values of vitamins, general nutritive valu and minerals will be maintained according to gender wise, BMI wise, age group wise. This form includes the special fields such as babies, pregnant ladies also.

8. Administrator – View Registered User List

- a. The Administrator can also see the registered user list in a list format which includes their height, weight, BMI, age and gender details.

9. Administrator – View User wise Food Recall

1.4 HARDWARE CONFIGURATION

- **Random Access Memory (RAM)** : 4 GB RAM minimum,
 - 8 GB RAM recommended.
- **Free disk space** : 2 GB of available disk space minimum,
 - 4 GB Recommended (500 MB for IDE + 1.5 GB for Android SDK and emulator system image).

The Android Emulator has additional requirements beyond the basic system requirements for Android Studio, which are described below:

- SDK Tools 26.1.1 or higher
- 64-bit processor
- Windows: CPU with UG (unrestricted guest) support
- Intel Hardware Accelerated Execution Manager (HAXM) 6.2.1 or later (HAXM 7.2.0 or later recommended)
- **Minimum required JDK version** : Java Development Kit 8
- **Minimum screen resolution** : 1280 x 800

1.5 SOFTWARE CONFIGURATION

Operating System : Microsoft Windows 7/8/10 (32- or 64-bit)

Front End : Android Studio 4.1

Back End : SQLite 2.1.0

1.6 SOFTWARE JUSTIFICATION

1.6.1 FRONT END – ANDROID STUDIO

Android Studio is the official integrated development environment (IDE) for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. It is available for download on Windows, macOS and Linux based operating systems or as a subscription-based service in 2020. It is a replacement for the Eclipse Android Development Tools (E-ADT) as the primary IDE for native Android application development.

Android Studio was announced on May 16, 2013 at the Google I/O conference. It was in early access preview stage starting from version 0.1 in May 2013, then entered beta stage starting from version 0.8 which was released in June 2014. The first stable build was released in December 2014, starting from version 1.0. On May 7, 2019, Kotlin replaced Java as Google's preferred language for Android app development. Java is still supported, as is C++.

FEATURES

The specific feature of the Android Studio is an absence of the possibility to switch autosave feature off. The following features are provided in the current stable version:

- Gradle-based build support
- Android-specific refactoring and quick fixes
- Lint tools to catch performance, usability, version compatibility and other problems
- ProGuard integration and app-signing capabilities
- Template-based wizards to create common Android designs and components
- A rich layout editor that allows users to drag-and-drop UI components, option to preview layouts on multiple screen configuration
- Support For Building Android Wear Apps
- Built-in support for Google Cloud Platform, enabling integration with Firebase Cloud Messaging (Earlier 'Google Cloud Messaging') and Google App Engine.
- Android Virtual Device (Emulator) to run and debug apps in the Android studio.

Android Studio supports all the same programming languages of IntelliJ (and CLion) e.g. Java, C++, and more with extensions, such as Go; and Android Studio 3.0 or later supports Kotlin and "all Java 7 language features and a subset of Java 8 language features that vary by platform version." External projects backport some Java 9 features. While IntelliJ states that Android Studio supports all released Java versions, and Java

12, it's not clear to what level Android Studio supports Java versions up to Java 12 (the documentation mentions partial Java 8 support). At least some new language features up to Java 12 are usable in Android. Once an app has been compiled with Android Studio, it can be published on the Google Play Store. The application has to be in line with the Google Play Store developer content policy.

Android SDK :

The Android software development kit (SDK) includes a comprehensive set of development tools. These include a debugger, libraries, a handset emulator based on QEMU, documentation, sample code, and tutorials. Currently supported development platforms include computers running Linux (any modern desktop Linux distribution), Mac OS X 10.5.8 or later, and Windows 7 or later. As of March 2015, the SDK is not available on Android itself, but software development is possible by using specialized Android applications.

Until around the end of 2014, the officially-supported integrated development environment (IDE) was Eclipse using the Android Development Tools (ADT) Plugin, though IntelliJ IDEA IDE (all editions) fully supports Android development out of the box, and NetBeans IDE also supports Android development via a plugin. As of 2015, Android Studio, made by Google and powered by IntelliJ, is the official IDE; however, developers are free to use others, but Google made it clear that ADT was officially deprecated since the end of 2015 to focus on Android Studio as the official Android IDE. Additionally, developers may use any text editor to edit Java and XML files, then use command line tools (Java Development Kit and Apache Ant are required) to create, build and debug Android applications as well as control attached Android devices (e.g., triggering a reboot, installing software package(s) remotely).

Enhancements to Android's SDK go hand-in-hand with the overall Android platform development. The SDK also supports older versions of the Android platform in case developers wish to target their applications at older devices. Development tools are downloadable components, so after one has downloaded the latest version and platform, older platforms and tools can also be downloaded for compatibility testing.

Android applications are packaged in .apk format and stored under /data/app folder on the Android OS (the folder is accessible only to the root user for security reasons). APK package contains .dex files (compiled byte code files called Dalvik executables), resource files, etc.

1.6.2 BACKEND - SQLITE

SQLite is a relational database management system (RDBMS) contained in a C library. In contrast to many other database management systems, SQLite is not a client–server database engine. Rather, it is embedded into the end program.

SQLite is ACID-compliant and implements most of the SQL standard, generally following PostgreSQL syntax. However, SQLite uses a dynamically and weakly typed SQL syntax that does not guarantee the domain integrity. This means that one can, for example, insert a string into a column defined as an integer.

SQLite will attempt to convert data between formats where appropriate, the string "123" into an integer in this case, but does not guarantee such conversions and will store the data as-is if such a conversion is not possible.

SQLite is a popular choice as embedded database software for local/client storage in application software such as web browsers. It is arguably the most widely deployed database engine, as it is used today by several widespread browsers, operating systems, and embedded systems (such as mobile phones), among others. SQLite has bindings to many programming languages.

DESIGN

Unlike client–server database management systems, the SQLite engine has no standalone processes with which the application program communicates. Instead, the SQLite library is linked in and thus becomes an integral part of the application program. Linking may be static or dynamic. The application program uses SQLite's functionality through simple function calls, which reduce latency in database access: function calls within a single process are more efficient than inter-process communication.

SQLite stores the entire database (definitions, tables, indices, and the data itself) as a single cross-platform file on a host machine. It implements this simple design by locking the entire database file during writing. SQLite read operations can be multitasked, though writes can only be performed sequentially.

Due to the server-less design, SQLite applications require less configuration than client–server databases. SQLite is called zero-conf because it does not require service management (such as startup scripts) or access control based on GRANT and passwords. Access control is handled by means of file-system permissions given to the database file itself. Databases in client–server systems use file-system permissions that give access to the database files only to the daemon process.

Another implication of the serverless design is that several processes may not be able to write to the database file. In server-based databases, several writers will all connect to the same daemon, which is able to handle its locks internally. SQLite, on the other hand, has to rely on file-system locks. It has less knowledge of the other processes that are accessing the database at the same time. Therefore, SQLite is not the preferred choice for write-intensive deployments. However, for simple queries with little concurrency, SQLite performance profits from avoiding the overhead of passing its data to another process.

SQLite uses PostgreSQL as a reference platform. "What would PostgreSQL do" is used to make sense of the SQL standard. One major deviation is that, with the exception of primary keys, SQLite does not enforce type checking; the type of a value is dynamic and not strictly constrained by the schema (although the schema will trigger a conversion when storing, if such a conversion is potentially reversible). SQLite strives to follow Postel's rule.

FEATURES

SQLite implements most of the SQL-92 standard for SQL, but lacks some features. For example, it only partially provides triggers and cannot write to views (however, it provides INSTEAD OF triggers that provide this functionality). While it provides complex queries, it still has limited ALTER TABLE function, as it cannot modify or delete columns. This changed in version 3.25.0 with support for ALTER TABLE RENAME COLUMN and version 3.35.0 with ALTER TABLE DROP COLUMN.

SQLite uses an unusual type system for a SQL-compatible DBMS: instead of assigning a type to a column as in most SQL database systems, types are assigned to individual values; in language terms it is dynamically typed. Moreover, it is weakly typed in some of the same ways that Perl is: one can insert a string into an integer column (although SQLite will try to convert the string to an integer first, if the column's preferred type is integer). This adds flexibility to columns, especially when bound to a dynamically typed scripting language. However, the technique is not portable to other SQL products. A common criticism is that SQLite's type system lacks the data integrity mechanism provided by statically typed columns in other products. The SQLite web site describes a "strict affinity" mode, but this feature has not yet been added. However, it can be implemented with constraints like CHECK (typeof(x)='integer').

Tables normally include a hidden rowid index column, which gives faster access. If a database includes an Integer Primary Key column, SQLite will typically optimize it by treating it as an alias for rowid, causing the contents to be stored as a strictly typed 64-bit signed integer and changing its behavior to be somewhat like an auto-incrementing column. Future[when?] versions of SQLite may include a command to introspect whether a column has behavior like that of rowid to differentiate these columns from weakly typed, non-autoincrementing Integer Primary Keys.

SQLite with full Unicode function is optional. Several computer processes or threads may access the same database concurrently. Several read accesses can be satisfied in parallel. A write access can only be satisfied if no other accesses are currently being serviced. Otherwise, the write access fails with an error code (or can automatically be retried until a configurable timeout expires). This concurrent access situation would change when dealing with temporary tables. This restriction is relaxed in version 3.7 when write-ahead logging (WAL) is turned on, enabling concurrent reads and writes. Version 3.6.19 released on October 14, 2009 added support for foreign key constraints.

SQLite version 3.7.4 first saw the addition of the FTS4 (full-text search) module, which features enhancements over the older FTS3 module. FTS4 allows users to perform full-text searches on documents similar to how search engines search webpages. Version 3.8.2 added support for creating tables without rowid, which may provide space and performance improvements. Common table expressions support was added to SQLite in version 3.8.3. In 2015, with the json1 extension and new subtype interfaces, SQLite version 3.9 introduced JSON content managing. As per version 3.33.0 the maximum supported database size is 281 TB.

1.6.3 JAVA DEVELOPMENT KIT

The Java Development Kit (JDK) is an implementation of either one of the Java Platform, Standard Edition, Java Platform, Enterprise Edition, or Java Platform, Micro Edition platforms released by Oracle Corporation in the form of a binary product aimed at Java developers on Solaris, Linux, macOS or Windows. The JDK includes a private JVM and a few other resources to finish the development of a Java application. Since the introduction of the Java platform, it has been by far the most widely used Software Development Kit (SDK).

The JDK is available for 64-bit x64 macOS (and that version also works with Rosetta 2), while an early access build (developer preview) from Microsoft is also available to support recent Apple M1 Macs. The JDK also comes with a complete Java Runtime Environment, usually called a private runtime, due to the fact that it is separated from the "regular" JRE and has extra contents. It consists of a Java Virtual Machine and all of the class libraries present in the production environment, as well as additional libraries only useful to developers, such as the internationalization libraries and the IDL libraries. Copies of the JDK also include a wide selection of example programs demonstrating the use of almost all portions of the Java API.

CHAPTER - 2

2. SYSTEM STUDY

2.1 EXISTING SYSTEM

Being mindful of the food you eat can promote better digestion, keep you full with less food, and influence wiser choices about what you eat in the future. It can also help you free yourself from unhealthy habits around food and eating. There are plenty of nutrition informatics related app are available in the android apps market. But there is no diet app available for south Indian foods related. Our south Indian foods is entirely different pattern when compared to north Indian food styles and foreign countries food styles. Most of the diet apps are especially designed for foreign foods and very rare North Indian foods. Little cases in rare apps covered South Indian foods but there is no user friendly interaction with the apps.

2.1.1 DRAWBACKS OF THE EXISTING SYSTEM

1. There is no Gender wise and Age wise Recommended Dietary Allowances (RDA) available in the existing apps.
2. There is no nutritive value information available for South Indian Foods such as General Values, Mineral Values and Vitamin Values.
3. There is no user friendly interactions in the apps.
4. Diet Chart Inference and suggestions are not there.
5. Food wise Nutritive value details cannot be seen in listwise.

2.2 PROPOSED SYSTEM

The proposed mobile app has multiple modules, including User Registration, User Login, User Daily Food Intake, RDA Comparison, and Display of Higher and Lower Mineral and Vitamin Levels. Energy, Carbohydrate, Protein, Fat, Total Fibre, Soluble Fibre, and Insoluble Fibre are all calculated in the app. Calculate the vitamin levels as well. The computed nutritive value will be compared to the recommended daily allowance (RDA) for the age group and gender. The user will see the comparison result together with the interpretation and inference result. These facts will aid the user in controlling their eating habits and developing a good dietary chart for a long, healthy, and happy life.

2.2.1 ADVANTAGES OF THE PROPOSED SYSTEM

1. Recommended Dietary Allowances (RDA) chart is available in this app.
2. Nutritive value information available for South Indian Foods such as General Values, Mineral Values and Vitamin Values.
3. More user friendly front end designs to interact with the user.
4. Users can update their daily food intake into this app and identify their excess and below nutrients values.
5. List wise nutritive value details are available for the South Indian Foods.

2.3 FEASIBILITY STUDY

Feasibility study is a test of the system proposal according to the workability impact of the organization, ability to meet user's needs and effective use of resources. The feasibility study is to serve as a decision document: it must satisfy the following factors:

- 1) User demonstrable needs
- 2) Problem worth solving
- 3) Method of solving problem.

❖ **Economic Feasibility:**

Economic feasibility is the most frequently used method of evaluating the effectiveness of a candidate system. The procedure is to determine the savings and benefits from the candidate system and compare the costs. If the benefits outweigh the costs then it is decided to go ahead with the project. Otherwise, further justification or alternations in the proposed system will have to be made if it to have a chance of being approved. It is an on-going effort that improves the accuracy at each phase of the system life cycle.

In the economic feasibility study, the following points were found out:

- The automated system will cost bit as the initial expenditure.
- Maintenance also involves some investment in terms of money.

Once the computerized system is installed, it can cater to the needs of the customer and the business without much of manual work, which is more cost-effective for the management. In the sense, the automated system is economically feasible.

❖ **Technical Feasibility:**

Technical feasibility centers on the existing system and to what extent it can support the proposed system. It involves financial considerations to accommodate technical enhancements. If the budget is a serious constraint, then the project will be judged not feasible. Now considering the proposed system, and company planning to implement

So the owner having realized the advantages, benefits and economic feasibility of the new system is ready to afford the extra expense that may arise for the satisfaction of all the hardware and software requirements.

❖ **Operational Feasibility:**

People are inherently resistant to change and computers have been known to facilitate change. An estimate should be made how strong a reaction the general public is likely to have towards a new computer system. It is common knowledge that computer installations have a lot to do with the turnover transfer retaining and changes to employee job status. Therefore it is understandable that the introduction of the candidate system requires special effort to educate and train the staff on a new way of conducting business. But since ultimately the introduction of a new system will only reduce the staff's workload, staff's may have no objection to install a computerized system and of course will be eager to extend their co-operation.

❖ **Behavioral Feasibility:**

Behavioral feasibility deals with how the developed software behaves in different scenarios when deployed. It is also a very important part in the different stages of software development.

ER DIAGRAM:

3.2 DATA FLOW DIAGRAM

The data flow diagram (DFD) is a graphical tool used for expressing system requirements in a graphical form. The DFD also known as the "bubble chart" has the purpose of clarifying system requirements and identifying major transformations that will become programs in system design. Thus DFD can be stated as the starting point of the design phase that functionally decomposes the requirements specifications down to the lowest level of detail. The DFD consists of series of bubbles joined by lines. The bubbles represent data transformations and the lines represent data flows in the system

Rules Used For Constructing a DFD

Process should be named and numbered for easy reference. Each name should be representative of the process. The direction of flow is from top to bottom and from left to right. That is data flow should be from source to destination. When a process is exploded into lower level details, they are numbered. The name of the data stores, sources and destinations are written in capital letters. Process and data flow names have the first letter of each word capitalized. The DFD is particularly designed to aid communication. If it contains dozens of process and data stores it gets too unwieldy. The rule of the thumb is to explode the DFD into a functional level. It is best to take each function

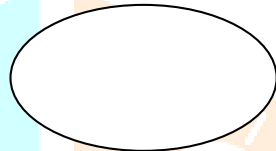
separately and expand it to show the explosion in a single process. If a user wants to know what happens within a given process, then the detailed explosion of that process may be shown. A data flow diagram is a graphical representation or technique depicting information flow and transform that are applied as data moved from input to output. The DFD are partitioned into levels that represent increasing information flow and functional details. The processes, data store, data flow, etc are described in Data Dictionary.

Data flow:



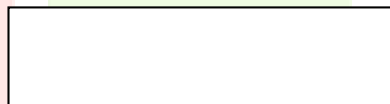
Data moves in a specific direction from an origin to destination

Process:



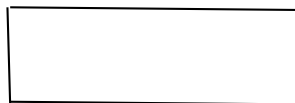
Procedure s people or devices that use or transform data

External entity:

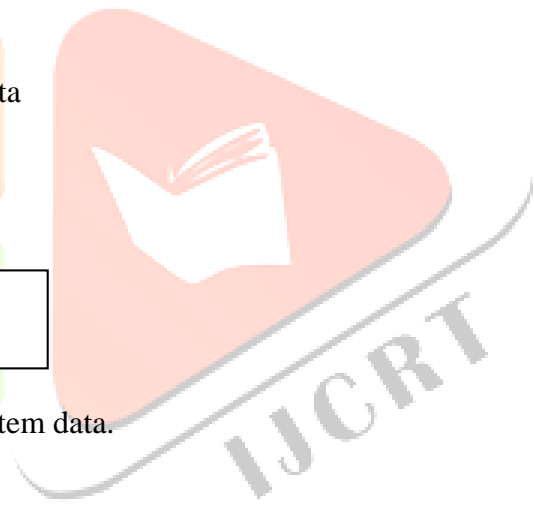


This defines a source (originator) or destination of system data.

Data Store:



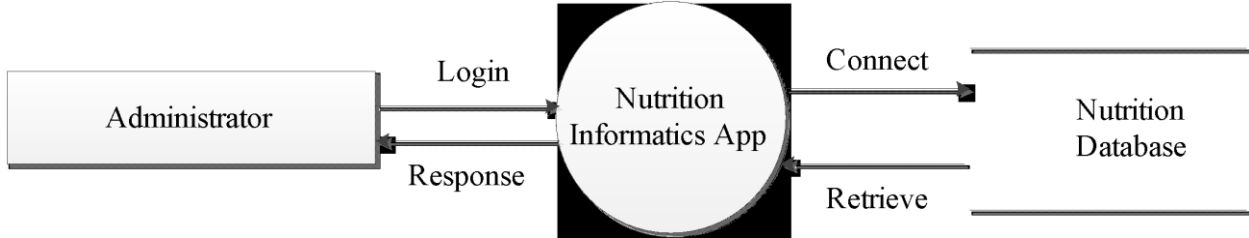
This indicates where data is stored in the system.



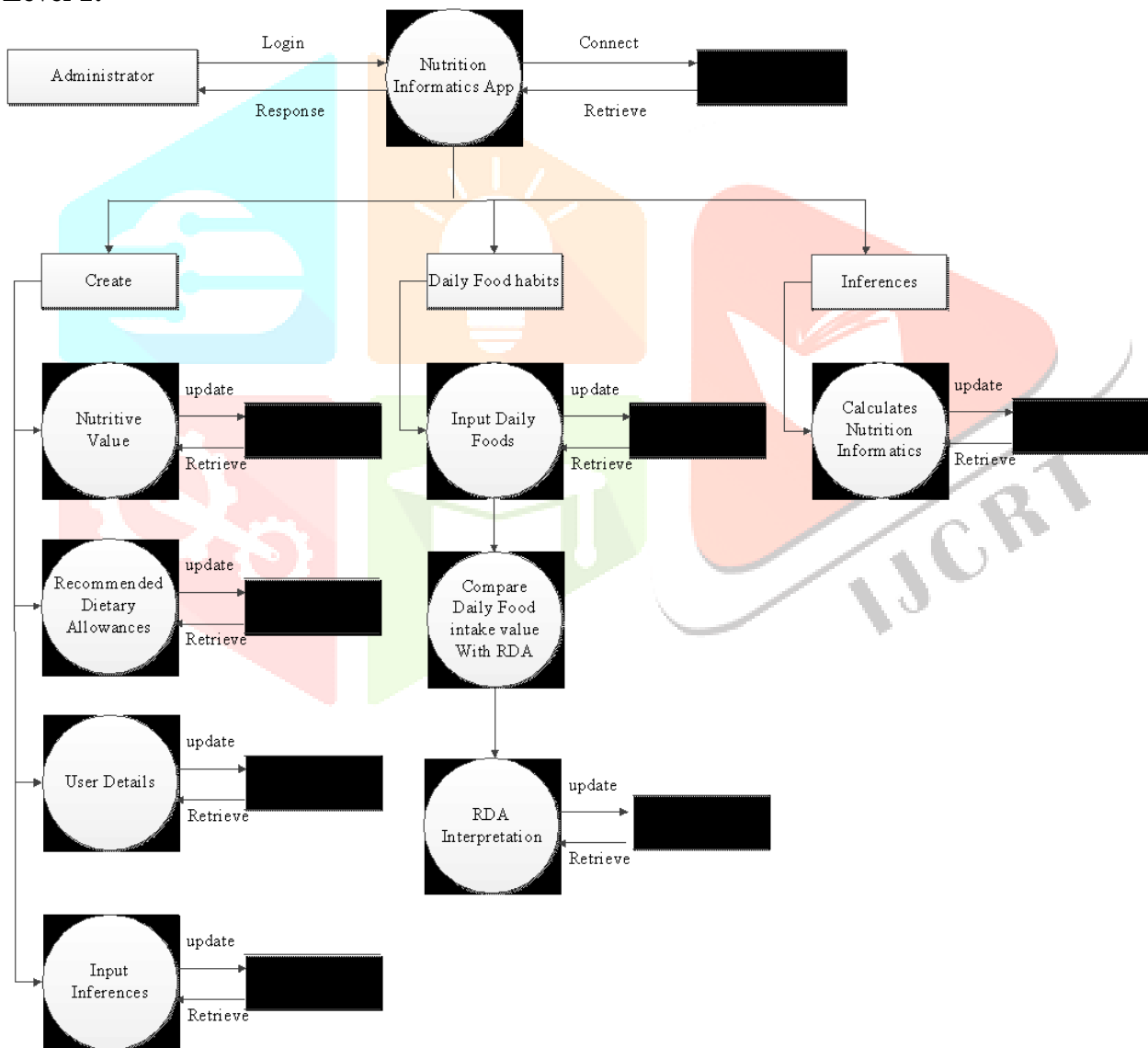
DATA FLOW DIAGRAM

DATA FLOW DIAGRAM

Level 0:



Level 1:



3.3 INPUT DESIGN

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy.

Input Design considered the following things:

- What data should be given as input?
- How the data should be arranged or coded?
- The dialog to guide the operating personnel in providing input.
- Methods for preparing input validations and steps to follow when error occur.

1. Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.

2. It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.

3. When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user

will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow

The input design in the Nutrition Informatics Apps are as follows,

- Food Master
- Nutrient Values
- Vitamin Values
- Mineral Values
- Food and Nutrient Values
- Recommended Dietary Allowances

3.4 DATABASE DESIGN

Data Constraints

All business in the world runs on business data being gathered stored and analyzed. Business managers determine a set of rules that must be applied to the data being stored to ensure its integrity.

Types of Data Constraints

There are two types of data constraints that can be applied to data being inserted into a database table. One type of constraint is called an I/O constraint. The other type of constraint is called a business rule constraint.

❖ I/O Constraints

The input /output data constraint is further divided into two distinctly different constraints.

The Primary Key Constraint

Here the data constraint attached to a column ensures:

- That the data entered in the table column is unique across the entire column.
- That none of the cells belonging to the table column are left empty.

The Foreign Key Constraint

Foreign constraint establishes a relationship between records across a master and a detail table. The relationship ensures.

- Records cannot be inserted in a detail table if corresponding records in the master table does not exist.
- Records of the master table cannot be deleted if corresponding records in the detail table exist.

❖ Business Rule Constraints

The Database allows the application of business rules to table columns. Business managers determine business rules.

The Database allows programmers to define constraints at:

- Column Level
- Table Level

Column Level Constraints

If data constraints are defined along with the column definition where creating or altering a table structure, they are column level constraints.

Table Level Constraints

If data constraints are defined after defining all the table columns when creating or altering a table structure, it is a table level constraint.

Null Value Concepts

A NULL value is different from a blank or zero. NULL values are treated specially by the database. A NULL value can be inserted into the columns of any data type.

Not Null Constraint Defined at the Column Level

When a column is defined as not null, then that column becomes a mandatory column. It implies that a value must be entered into the column if the record is to be accepted for storage in the table.

The Primary Key Constraint

Primary Key Concepts

A primary key in a table used to uniquely identify each row in the table. A primary key column in a table has special attributes.

- It defines the column as a mandatory column i.e. the column cannot be left blank. The NOT NULL attribute is active.

The data held across the column MUST BE UNIQUE

TABLE DESIGN

Table Name : Foodmaster

Primary Key : Food_id

Foreign Key : -

Column Name	Datatype	Description
Food_id	numeric(4,0)	Food id
Food_name	varchar(20)	Food Name
Food_Description	varchar(20)	Food Description
Dining_Type	varchar(20)	Dining_Type

Table Name : General_Value

Primary Key : General_id

Foreign Key : Food_id

Column Name	Datatype	Description
General_id	numeric(4,0)	General_id
Food_id	numeric(4,0)	Food_id
Energy	varchar(20)	Energy
Carbohydrate	varchar(20)	Carbohydrate
Protein	varchar(20)	Protein
Fat	varchar(20)	Fat
Total Fibre	varchar(20)	Total Fibre
Soluble fibre	varchar(20)	Soluble fibre
Insoluble fibre	varchar(20)	Insoluble fibre

CHAPTER - IV

4. SYSTEM TESTING

4.1 SYSTEM TESTING

Testing of the software as a mean of accessing or measuring the software to determine its quality. The area of testing is one of the key process areas in ensuring the quality of the software known as Software Quality Assurance (SQA).

Testing is done with one primary objective to ensure the quality of the software before it is actually implemented. The main purpose of testing from developer's point of view is to gain confidence. If no error are found at least he is sure that the product under development is meeting it required goals in terms of quality.

There is no way to find when to stop testing however people have followed certain norms and guidelines over the course of time. They are as follows:

- When the rate of finding errors has reached an acceptable level.
- Based on the size of the system, "enough" errors have been found where "enough" can be a quantitative parameter based on historical data.
- Measuring coverage of testing as percentage of line executed divided by total number of lines in the system. If the percentage is below acceptable limits, then the test plans have to be enhanced to test areas where code not been tested.

4.2 VALIDATION TESTING

In validation testing we ensure that all the data entered was displayed properly on the view screen and that we got the expected results.

In the Nutrition Informatics App, the following validation has been done such as mobile number validation, Email Validation, password validation and other criteria check validation. The validation check is done using java script inside HTML script.

4.3 INTEGRATION TESTING

The last step was to integrate the various modules in the software. The proposed system is implemented top-down integration. The integration was performed in five steps:

- The main control module was used as test driver and stubs were substituted for all components directly subordinate to the main control module.
- Depending on depth first search subordinate stubs was replaced with actual components.
- Tests were conducted as each component with real component.
- On completion of each set of tests, another stub is replace with real component.
- Integration testing was conducted to ensure execution of some subset of tests have already been conducted had not propagated unintended side effects.

The integration testing is done in the following segments,

- Integrate General Nutrient Value Details to the Daily Food Intake Calculation.
- Integrate Mineral and Vitamin Value Details to the Daily Food Intake Calculation.
- Recommended Dietary Allowances with the Daily Food Intake.

4.4 ACCEPTANCE TESTING

Acceptance testing involves planning and execution of functional tests, performance tests and stress tests in order to demonstrate that the implemented system satisfies its requirements. When custom software is built for one customer, a series of acceptance tests are conducted to enable the customer to validate all requirements.

In fact acceptance cumulative errors that might degrade the system over time will incorporate test cases developed during integration testing. Additional testing cases are added to achieve the desired level functional, performance and stress testing of the entire system.

4.5 UNIT TESTING

Unit testing focuses verification effort on the smallest unit of the software. Using the detailed design description as design a guide, important control path are tested to uncover errors within the boundary of the module. This testing was carried out during programming stage itself. After testing each every field in the modules, the modulus of the project is tested

4.6 TEST CASES

User Login:

Test ID	Test Type	Description	Test Steps	Expected Result	Actual Result	Status
1.	Validation Testing in User Login Form	Test the 'Login' button with valid values in the username and password	Step1: Enter the values in textbox fields Step2 :Click 'Login'	Login should be verified and allowed to next main form	Login form verified and navigate to main form	Success

User Registration Form:

Test ID	Test Type	Description	Test Steps	Expected Result	Actual Result	Status
1.	Unit Testing In User Registration form	Test the 'Submit Now' button with valid values in the Text boxes.	Step1: Enter the values in textbox fields Step2 :Click 'Submit'	Valid all the textboxes and save the data into database	Data validated and stored in database	Success
2.	Validation Testing in User Registration form	Test the 'Submit Now' button with valid	Step1: Enter the mobile number Step2 :Click 'Submit'	Valid Mobile number and if valid save the	Data validated and stored in database	Success

		Mobile Number		data into database		
3.	Validation Testing in User Registration form	Test the 'Submit Now' button with valid Email ID	Step1: Enter the Email ID Step2 :Click 'Submit'	Valid Email ID and if valid save the data into database	Data validated and stored in database	Success

Food Master :

Test ID	Test Type	Description	Test Steps	Expected Result	Actual Result	Status
1.	Unit Testing In Food Master	Test the 'Submit Now' button with valid values in the Text boxes.	Step1: Enter the values in textbox fields Step2 :Click 'Submit'	Valid all the textboxes and save the data into database	Data validated and stored in database	Success

Nutrient Value :

Test ID	Test Type	Description	Test Steps	Expected Result	Actual Result	Status
1.	Unit Testing In Nutrient Value	Test the 'Submit Now' button with valid values in the Text boxes.	Step1: Enter the values in textbox fields Step2 :Click 'Submit'	Valid all the textboxes and save the data into database	Data validated and stored in database	Success

Mineral Value:

Test ID	Test Type	Description	Test Steps	Expected Result	Actual Result	Status
1.	Unit Testing In Mineral Value	Test the 'Submit Now' button with valid values in the Text boxes.	Step1: Enter the values in textbox fields Step2 :Click 'Submit'	Valid all the textboxes and save the data into database	Data validated and stored in database	Success

Vitamins Value:

Test ID	Test Type	Description	Test Steps	Expected Result	Actual Result	Status
1.	Unit Testing In Vitamins Value	Test the 'Submit Now' button with valid values in the Text boxes.	Step1: Enter the values in textbox fields Step2 :Click 'Submit'	Valid all the textboxes and save the data into database	Data validated and stored in database	Success

Recommended Dietary Allowances:

Test ID	Test Type	Description	Test Steps	Expected Result	Actual Result	Status
1.	Unit Testing In Recommended Dietary Allowances	Test the 'Submit Now' button with valid values in the Text boxes.	Step1: Enter the values in textbox fields Step2 :Click 'Submit'	Valid all the textboxes and save the data into database	Data validated and stored in database	Success

CHAPTER – 5**5 SYSTEM IMPLEMENTATION AND MAINTENANCE**

System Implementation is the stage of the project when the theoretical design is tuned into working system. If the implementation system stage is not carefully controlled and planned, it can cause chaos. Thus it can be considered to be the most critical stage in achieving a successful new system and in giving the users a confidence that the system will work and be effective.

The implementation stage in a project involves,

- Careful Planning investigation of the current system, checking constraints and the implementation.
- Training the staffs in the newly developed system.

A software application in general is implemented after navigating the complete life cycle method of a project. Various life cycle processes such as requirement analysis, design phase, verification, testing and finally followed by the implementation phase results in a successful project management. The software application which is basically a Windows based application has been successfully implemented after passing various life cycle processes mentioned above.

As the software is to be implemented in a high standard industrial sector, various factors such as application environment, user management, security, reliability and finally performance are taken as key factors through out the design phase. These factors are analyzed step by step and the positive as well as negative outcomes are noted down before the final implementation.

Security and authentication is maintained in both user level as well as the management level. The data is stored in Sqlite as RDBMS, which is highly reliable and simpler to use, the user level security is

managed with the help of password options and sessions, which finally ensures that all the transactions are made securely.

The application's validations are made, taken into account of the entry levels available in various modules. Possible restrictions like number formatting, date formatting and confirmations for both save and update options ensures the correct data to be fed into the database. Thus all the aspects are charted out and the complete project study is practically implemented successfully for the end users.

End Users

- No specific knowledge or skills are required from the end user.
- End user should have basic idea about computer operations and database.

Administrator

- Administrator must be having good knowledge of database management system.
- Administrator must be capable to manage user rights.
- If the network connection does not work properly than our system should not work as intended.
- Also that is assumed that the product is installed properly at windows server

Assumptions and Dependencies

Assumptions:

- The code should be free with compilation errors/syntax errors.
- The product must have an interface which is simple enough to understand.

Dependencies:

- All necessary hardware and software are available for implementing and use of the tool.
- The proposed system would be designed, developed and implemented based on the software requirements specifications document.
- End users should have basic knowledge of computer and they also assure that the users will be given software training documentation and reference material.
- The system is not required to save generated reports.

5.3 SYSTEM MAINTENANCE

Security and authentication is maintained in both user level as well as the management level. The data is stored in SQL Lite as RDBMS, which is highly reliable and simpler to use, the user level security is managed with the help of password options and sessions, which finally ensures that all the transactions are made securely.

5.4 SYSTEM SECURITY MEASURES

The application's validations are made, taken into account of the entry levels available in various modules. Possible restrictions like number formatting, date formatting and confirmations for both save and update options ensures the correct data to be fed into the database. Thus all the aspects are charted out and the complete project study is practically implemented successfully for the end users.

5.5 SYSTEM ENHANCEMENT

Security and authentication is maintained in both user level as well as the management level. The data is stored in SQLite as RDBMS, which is highly reliable and simpler to use, the user level security is managed with the help of password options and sessions, which finally ensures that all the transactions are made securely.

A software application in general is implemented after navigating the complete life cycle method of a project. Various life cycle processes such as requirement analysis, design phase, verification, testing and finally followed by the implementation phase results in a successful project management. The software application which is basically a web based application has been successfully implemented after passing various life cycle processes mentioned above.

As the software is to be implemented in a high standard industrial sector, various factors such as application environment, user management, security, reliability and finally performance are taken as key factors throughout the design phase. These factors are analyzed step by step and the positive as well as negative outcomes are noted down before the final implementation.

CONCLUSION

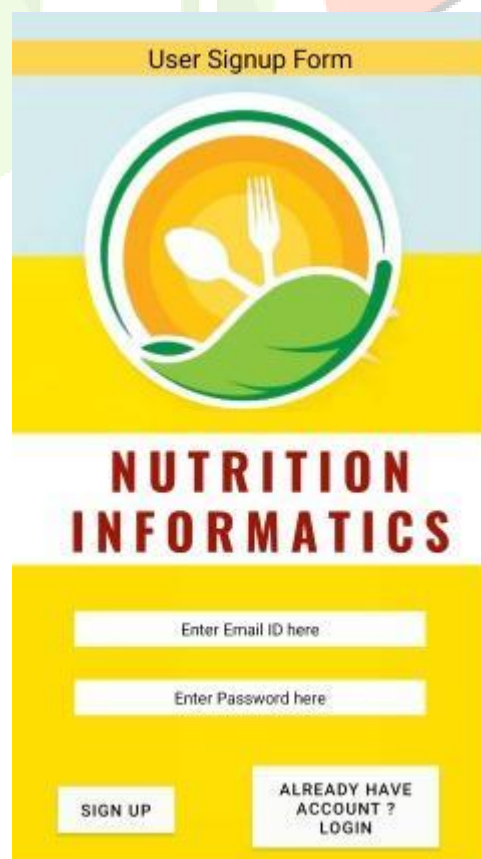
The developed "**Nutrition Informatics App**" is primarily used to assess the user's daily food consumption and compare the nutritive content of the food consumed, including minerals and vitamins. Infants, young children, adolescents, adults, and the elderly may all live under the same roof in a family. It's never easy to create a balanced, healthy meal for each family member, especially when their demands and tastes differ. To live a healthy and active life, humans require a wide variety of nutrients. Age, body weight, physical activity, physiological status (pregnancy, breastfeeding), and other factors influence the quantity of each nutrient required by an individual. So, in general, nutritional requirements differ from one person to the next. A suggested app is needed that assesses the amount of nutrient the user consumes on a daily basis and makes dietary recommendations based on RDA values.

The proposed mobile app automated the modules encompasses User Registration, User Login, User Daily Food Intake, RDA Comparison, and Display of Higher and Lower Mineral and Vitamin Levels. Energy, Carbohydrate, Protein, Fat, Total Fibre, Soluble Fibre, and Insoluble Fibre are all calculated in the app. Also, determine the levels of vitamins such as Thiamine (B1), Riboflavin (B2), Niacin (B3), Pantothenic Acid (B5), Pyridoxine (B6), Biotin (B7), Folate (B9), Vitamin A (Retinol), Vitamin C (Ascorbic Acid), Vitamin D (Calciferol), Vitamin E (tocopherol), Vitamin K (tocopherol), and Vitamin K (tocopherol (Phylloquinone). Calcium, Phosphorus(P), Iron (Fe), Sodium(Na), Potassium(K), Zinc(Zn), Magnesium(Mg), Copper (cu), Chromium(Cr), and Selenium are among the minerals that the app calculates.

The computed nutritive value will be compared to the recommended daily allowance (RDA) for the age group and gender. The user will see the comparison result together with the interpretation and inference result. The user and administrator roles in the mobile app have been created separately. The administrator will update the RDA and nutritional information for a variety of Indian foods. These facts will aid the user in controlling their eating habits and developing a good dietary chart for a long, healthy, and happy life.

SCOPE FOR FUTURE ENHANCEMENT

1. In the future enhancement, the present Nutrition Informatics App will be integrated with more Nutrient values.
2. SMS Notifications will be integrated in order to alert the user exceeding daily intake limit.
3. Exercise time, sleeping time and other workouts will be integrated and compare with daily food intake.



User Signup Form

NUTRITION INFORMATICS

Enter Email ID here

Enter Password here

SIGN UP

ALREADY HAVE ACCOUNT? LOGIN





