UDC 004.223.2

«SMART RECIPE BOOK» APP

A. SKUKOVSKAYA, T. RUDKOVA
Polotsk State University, Belarus

The article discusses an application for storing recipes with the function of counting calories and selecting the optimal dishes. The analysis of the technologies used to develop this application is carried out. The relevance of the development of this application is considered.

Introduction. Currently, a large number of people are switching from paper to electronic media. The popularity of using electronic media is determined not only by the convenience of adding and storing information, but also by the presence of additional functions for processing large amounts of data. Electronic storage is not limited to professional use, it is also popular in everyday life, for example, in the form of notebooks.

Along with information technology, sports and healthy nutrition are actively popularized, which makes the introduction of information technology in the field of a healthy lifestyle relevant. At the moment, a large number of applications for sports, calorie counting and menu compilation are being created, which confirms the relevance of the created application.

The Smart Recipe Book app is convenient data storage, as well as a tool for creating a useful and tasty menu, and selecting dishes based on users' physical characteristics.

This article is aimed at describing the database, its design methods and the technologies used to create the developed application.

Main section. Designing a software product begins with designing the database needed to store all the application information. In accordance with the subject area and purpose of the application, the database stores information about users, recipes, products, cooking technologies and the values of proteins, fats and carbohydrates (PFC), both for recipes and for individual users, based on their physical characteristics.

Database development is performed using data modeling. The purpose of data modeling is to develop a conceptual database schema in the form of one model or several local models that can be relatively easily mapped to any database system. The most common data modeling tool is entity-relationship diagrams (ERDs). Using ERD, the DFD data storage devices are detailed, diagrams are documented, informational aspects of the business system are documented, including the identification of objects, entities, the properties of these objects — attributes and their relationships with other objects (relationships).

Entity is a set of instances of real or abstract objects (people, events, states, ideas, objects, etc.) that share common attributes or characteristics. Each entity must have a unique identifier. Each entity instance must be uniquely identified and distinguished from all other instances of this entity type. Each entity must have some properties:

- a unique name;
- one or more attributes that either belong to the entity or are inherited through communication;
- one or more attributes that uniquely identify each entity instance.
- any number of relationships with other model entities.

Relationship is an association between entities in which each instance of one entity is associated with an arbitrary (including zero) number of instances of the second entity, and vice versa.

An attribute (Attribute) represents a type of characteristic or property associated with a variety of real or abstract entities - objects (people, places, events, states, ideas, objects, etc.). An attribute instance is a specific characteristic of an individual element of a set. An attribute instance is determined by the type of characteristic and its value, called the attribute value. An entity instance must have a single specific value for the associated attribute [1].

In the course of the analysis of knowledge and database development, the main entities were identified, which are described below.

The Dish essence describes dishes that can have a large number of recipes, characterized by name, type (for example: soup, salad), belonging to the national cuisine of a country, type of event for which this dish is being prepared (for example: breakfast, New Year).

The recipe essence describes one of the possible recipes for a particular dish, characterized by its name, number of servings, calorie content, attachment to a specific dish.

The Products essence describes the products characterized by the name, calorie content and PFC indicators.

Ingredients essence describes the ingredients of a recipe characterized by the number of grams and linked to the recipe and product.

The technology of preparation eesence describes the steps for preparing the recipe characterized by the description, duration, number of action.

The user entity contains the usernames and passwords of users registered in the application.

The User's characteristics essence describes the parameters of the user (weight, height, age), BJU indicators for the user, the required number of calories and the date the characteristics were added.

Favorite recipes essence describes recipes that the user liked and is characterized by priority.

Based on the selected basic entities, we distinguish under the entities to the main entities:

For the Dish entity, we distinguish the entity by type, event type, and national cuisine.

For the Recipe entity, select the type of recipe for the entity.

For the Technology of cooking essence, we select an action for the essence.

To implement the application, it was necessary to establish all the relationships between the entities: it was necessary to consider the entire information system in aggregate and determine the mutual influence of the entities.

The relational model represents a database in the form of a set of interrelated relationships (tables) that are used to store information about objects represented in the database.

One of the most important advantages of relational databases is that it is possible to store logically grouped data in different tables and specify the relationships between them, combining them into a single database. Such data organization allows us to reduce the redundancy of stored data, simplifies their input and organization of queries and reports.

In each database table, a primary key may exist. The primary key is understood as a field or a set of fields that uniquely (uniquely) identify an entry. The primary key should be minimally sufficient: it should not contain fields whose removal from the primary key will not affect its uniqueness.

To organize relationships between two or more tables, secondary keys are used. They serve as constraints on the integrity of the relationships of several tables, because the subordinate table cannot refer to non-existent records of the main table (which allows the construction of integral data models).

There are three types of relationships between tables:
– one-to-many,
– one-to-one,
– many-to-many.

One-to-many relationships are the most common for relational databases. To ensure referential integrity, a foreign key is created in the child table through which a link to the child table is organized. A prerequisite is that the foreign key must match the composition of the fields with the primary key of the parent table [2].

The primary and foreign keys of the relationships of the designed database are presented in table 1.

Table 1. - Primary and foreign keys of relations

№	Table name	Primary key	Foreign key
1	Dish	IDDish	IDTypeOfDishes, IDNatianality, IDEventType
2	TypeOfDishes	IDTypeOfDishes	-
3	NationalCuisine	IDNationality	-
4	EventTypeDishes	IDEventType	-
5	Recipe	IDRecipe	IDDish
6	Products	IDProduct	-
7	Ingridients	IDIngridient	IDProduct, IDRecipe
8	TypeOfRecipe	IDTypeOfRecipe	-
9	RelRecipeType	IDRel	IDRecipe, IDTypeOfRecipe
10	CookingTechnology	IDTechnology	IDRecipe, IDAction
11	Action	IDAction	-
12	User	IDUser	-
13	UserFeatures	IDFeatures	IDUser
14	FavouriteRecipe	IDFavouriteRecipe	IDRecipe, IDUser

MSSQL 2014 was used to create the information database. To create the application, the Microsoft Visual Studio 2017 development environment was used.

To create the application, C # was chosen. C # is a programming language that combines object-oriented and context-oriented concepts. C # belongs to a family of languages with C-like syntax, of which its syntax is closest to C ++ and Java. The language has strict static typing, supports polymorphism, operator overloading, pointers to class member functions, attributes, events, properties, exceptions, comments in XML format [1].

MSSQL Server was used to create the database due to its good compatibility with the selected programming language and ease of use.

Connection to the database was carried out using the Entity Framework, this framework provides quick and easy connection to the database and convenient work with it using the built-in functions. Entity Framework presents a special object-oriented technology based on the .NET framework for working with data. If traditional ADO.NET tools allow you to create connections, commands and other objects for interacting with databases, the Entity Framework represents a higher level of abstraction, which allows you to abstract from the database itself and work with data regardless of the type of storage. If at the physical level the programmer operates with tables, indexes, primary and foreign keys, then at the conceptual level that the Entity Framework offers, he already works with objects [2].

To ensure the functionality and ease of use of the information system, an application was created that allows you to add new recipes, add, delete, edit information about yourself, about your favorite recipes, display information about recipes and select the best dishes for each application user.

When creating databases, one of the important tasks is to ensure data integrity. Integrity (from the English integrity - intactness, inviolability, safety, integrity) - is understood as the correctness of the data at any time. But this goal can be achieved only within certain limits: the DBMS cannot control the correctness of each individual value entered into the database. To do this, there are a number of tools that help the developer minimize the possibility of violating the integrity of the database data: triggers, checks («check»), uniqueness («unique»), etc. A trigger is a stored procedure of a special type that is automatically executed when a given event occurs [3].

In this application, triggers are also used to automatically calculate the calorie content of dishes and user characteristics. Table 2 summarizes the main uses of constraints and triggers.

Table 2. – Table Constraints and Triggers

№	Table name	Description
1	Ingridients	When adding a new ingredient, the calorie content of the whole dish is recalculated
		When adding a new ingredient, the UNIQUE restriction applies to the combination of the recipe and the product
2	Recipe	When a recipe is deleted, all tables associated with it are deleted
3	CookingTechnology	When adding new technology, the calorie content of the entire dish is recalculated depending on the action
4	UserFeatures	When changing weight, age and height, the values of PFC and calories are recounted
5	RelRecipeType	When adding a new type of recipe, the UNIQUE restriction applies to the combination of recipe and type

In addition, to ensure data integrity, there are triggers for adding, deleting and modifying tables.

Using Windows Forms, an intuitive and easy-to-use user interface was created.

Conclusion. The developed application meets all the requirements of the subject area, the tables of the created database meet the requirements of normalization, which ensures the integrity and consistency of information.

The most suitable and modern programming languages and development environments were used to create the application.

REFERENCES

1. Progoredia [Electronic resource]. Access mode: http://progopedia.ru/language/csharp/. Access date: 02/17/2020.
2. METANIT.COM. Programming site [Electronic resource]. Access Mode: https://metanit.com/sharp/entityframework/1.1.php. Access date: 02/17/2020.
3. Studme.org [Electronic resource]. Access Mode: https://studme.org/93824/informatika/triggery. Access date: 02/20/2020.