

COMMUNICATION AND INFORMATION AGENCY OF UZBEKISTAN

TASHKENT UNIVERSITY OF INFORMATION TECHNOLOGIES

To defense

Head of Department

« » 2010.

BACHELOR'S FINAL QUALIFYING WORK

**Theme: DECISION SUPPORT SYSTEM FOR SELECTING
APPROPRIATE LOCATION FOR CAFE**

Graduate _____ Safabayev U.S.
(signature) (surname)

Supervisor _____ Hachaturova E.M..
(signature) (surname)

Reviewer _____
(signature) (surname)

Consultant _____
on SLA (signature) (surname)

Tashkent - 2012

SUMMARY

The main goal of this work is development the software for selecting the most appropriate location for café. By exploring business processes and chooses factors are identified that helps to successfully choose appropriate places for café at the certain area.

INTRODUCTION

The main goal of this work is selecting *the most appropriate location for café*.

This task can be solved by means of:

- Finding required information about Decision Support Systems.
- Exploring business processes and chooses which factors may cause success for café at the certain area.
- Create comfortable user interface.
- By means of special mathematical computations, external factors, user`s preferences find and choose relevant location for café.

I am going to solve this problem by Decision Support System. And my work consists of 4 parts, and each of it explores certain problem, and devotes to creating application, which help to find appropriate location for café.

The main reason of my choice is because this field is able to offer to a lot of people different opportunities to start our own business, develop our ideas and gain revenue. By means of this system, my project can solve the main problem, which faces a lot of beginners in this sphere. That is why, I decided to devote my work for exploring one of the main tasks for café establishment.

It is very difficult field, because there are so many external factors, which we can not predict or avoid, that is why I decided to devote this work for exploration this task by means of math algorithms, mental inferences and life experience in this sphere like customer and café`s visitor .

We must know, that this system cannot provide success in this sphere, instead, it proposes several possible ways, which are based on the customer preferences and special math algorithms. My project, includes the most important and influencing factors which are able to help to achieve success.

Moreover, my system is able to choose location for café using user`s preferences, because in this case, I will be able to choose one of the ways: Build new building for café, buy it or rent it from another entrepreneur.

The law on information

Article 1. The purpose of the present Law

The purpose of the present Law is regulation of relations in the field of information, uses of information resources and information systems.

Article 2. The legislation on information

The legislation on information consists of the present Law and other certificates of the legislation. If the international contract of Republic Uzbekistan establishes other rules, than what are provided by the legislation of Republic Uzbekistan on information rules of the international contract are applied.

Article 3. The basic concepts

In the present Law following basic concepts are applied: information - organizational social and economic and scientific and technical process of creation of conditions for satisfaction of requirements legal and physical persons in the information with use of information resources, information technology and information systems; an information resource - the information, a databank, a database in the electronic form as a part of information system; the proprietor of information resources or information systems - legal or the physical person who is carrying out possession, using and the order information resources or information systems; the owner of information resources or information systems - legal or the physical person who is carrying out possession, using and the order information resources or information systems within the rights established by the law or the proprietor of information resources, information systems; Information technology - set of methods, devices, ways and the processes used for gathering, storage, search, processing and information distribution; the information system - organizational ordered set of information resources, information technology and the communication facilities, allowing to carry out gathering, storage, search, processing and using the information.

Article 4. The State policy in the field of information

The state policy in the field of information is directed on creation of national information system taking into account modern world tendencies of development

and perfection of information resources, information technology and information systems. The basic directions of a state policy in the field of information are: realization of constitutional laws of everyone on free reception and information distribution, providing of access to information resources; creation of uniform information field of Republic Uzbekistan on the basis of information systems of state structures, branch and territorial information systems, and also information systems legal and physical persons; creation of conditions for access to the international information networks and the world information network the Internet; formation of the state information resources, creation and development of information systems, maintenance of their compatibility and interaction; The organization of manufacture of modern means of information technology; assistance to formation of the market of information resources, services and information technology; stimulation of development of manufacture of software products; support and business stimulation, creation of favorable conditions for attraction of investments; preparation and improvement of professional skill of shots, stimulation of scientific researches.

Uzbekistan on a threshold of achievement of independence.

The edition of the book of President Islam Karimov «Uzbekistan on a threshold of achievement of independence» became considerable event in public life of our country. In educational institutions, labor collectives, blocks pass the actions devoted to deep studying and wide propagation of given product, being an important source in studying of history of our independence. In one of such actions, №5 cities which have taken place at comprehensive school of Dzhizak, deputies of city convention, the teacher, pupils, activists of the blocks have taken part.

On it was noticed that the given book of the President of the country has huge educational and scientifically-practical value. In it on the basis of concrete events and documents it is told about the difficult period of historical development of our state. The book of the President is an authentic and valuable source in studying of

history of our independence. In it deeply well-founded historical data on political strike and various events of those years are collected, it is told about tests and difficulties which were courageously overcome by our people thanks to deeply thought over wise policy of President Islam Karimov.

To construct the independent state which takes today a worthy place in the world community where the people lead happy and safe life, was uneasy business, – the deputy of Dzhizak city convention of People's Deputies of Nosir Siddikov has told. The Given book has great value for deep acquaintance of people with those difficulties which should be overcome, tells about courage and the courage shown by the President of the country in the first years of independence, about our recent history. Deep studying of this product is very important for that comprehension, what invaluable riches are freedom and independence, and also for youth education by worthy sons and daughters of our free Native land.

DECISION SUPPORT SYSTEM

Summary

Information Systems researchers and technologists have built and investigated computerized Decision Support Systems (DSS) for approximately 40 years. This article chronicles and explores the developments related to building and deploying DSS. The journey begins with building model-driven DSS in the late 1960s, theory developments in the 1970s, and implementation of financial planning systems, spreadsheet-based DSS and Group DSS in the early and middle 1980s. Data warehouses, Executive Information Systems, OLAP and Business Intelligence evolved in the late 1980s and early 1990s. Finally, the chronicle ends with knowledge-driven DSS and the implementation of Web-based DSS beginning in the mid-1990s. The field of computerized decision support is expanding to use new technologies and to create new applications.

History

According to Keen (1978), the concept of decision support has evolved from two main areas of research: The theoretical studies of organizational decision making

done at the Carnegie Institute of Technology during the late 1950s and early 1960s, and the technical work on interactive computer systems, mainly carried out at the Massachusetts Institute of Technology in the 1960s. It is considered that the concept of DSS became an area of research of its own in the middle of the 1970s, before gaining in intensity during the 1980s. In the middle and late 1980s, executive information systems (EIS), group decision support systems (GDSS), and organizational decision support systems (ODSS) evolved from the single user and model-oriented DSS.

According to Sol (1987) the definition and scope of DSS has been migrating over the years. In the 1970s DSS was described as "a computer based system to aid decision making". Late 1970s the DSS movement started focusing on "interactive computer-based systems which help decision-makers utilize data bases and models to solve ill-structured problems". In the 1980s DSS should provide systems "using suitable and available technology to improve effectiveness of managerial and professional activities", and end 1980s DSS faced a new challenge towards the design of intelligent workstations.

In 1987 Texas Instruments completed development of the Gate Assignment Display System (GADS) for United Airlines. This decision support system is credited with significantly reducing travel delays by aiding the management of ground operations at various airports, beginning with O'Hare International Airport in Chicago and Stapleton Airport in Denver Colorado.

Beginning in about 1990, data warehousing and on-line analytical processing (OLAP) began broadening the realm of DSS. As the turn of the millennium approached, new Web-based analytical applications were introduced.

The advent of better and better reporting technologies has seen DSS start to emerge as a critical component of management design. Examples of this can be seen in the intense amount of discussion of DSS in the education environment.

DSS also have a weak connection to the user interface paradigm of hypertext. Both the University of Vermont PROMIS system (for medical decision making) and the Carnegie Mellon ZOG/KMS system (for military and business decision making)

were decision support systems which also were major breakthroughs in user interface research. Furthermore, although hypertext researchers have generally been concerned with information overload, certain researchers, notably Douglas Engelbart, have been focused on decision makers in particular.

Theory Development

In the mid- to late 1970s, both practice and theory issues related to DSS were discussed at academic conferences including the American Institute for Decision Sciences meetings and the ACM SIGBDP Conference on Decision Support Systems in San Jose, CA in January 1977 (the proceeding were included in the journal Database). The first International Conference on Decision Support Systems was held in Atlanta, Georgia in 1981. Academic conferences provided forums for idea sharing, theory discussions and information exchange.

At about this same time, Keen and Scott Morton's DSS textbook (1978) provided the first broad behavioral orientation to decision support system analysis, design, implementation, evaluation and development. This influential text provided a framework for teaching DSS in business schools. McCosh and Scott-Morton's (1978) DSS book was more influential in Europe.

In 1980, Steven Alter published his MIT doctoral dissertation results in an influential book. Alter's research and papers (1975; 1977) expanded the framework for thinking about business and management DSS. Also, his case studies provided a firm descriptive foundation of decision support system examples. A number of other MIT dissertations completed in the late 1970s also dealt with issues related to using models for decision support.

Alter concluded from his research (1980) that decision support systems could be categorized in terms of the generic operations that can be performed by such systems. These generic operations extend along a single dimension, ranging from extremely data-oriented to extremely model-oriented. Alter conducted a field study of 56 DSS that he categorized into seven distinct types of DSS. His seven types include:

- π File drawer systems that provide access to data items.

- π Data analysis systems that support the manipulation of data by computerized tools tailored to a specific task and setting or by more general tools and operators.
- π Analysis information systems that provide access to a series of decision-oriented databases and small models.
- π Accounting and financial models that calculate the consequences of possible actions.
- π Representational models that estimate the consequences of actions on the basis of simulation models.
- π Optimization models that provide guidelines for action by generating an optimal solution consistent with a series of constraints.
- π Suggestion models that perform the logical processing leading to a specific suggested decision for a fairly structured or well-understood task.

Donovan and Madnick (1977) classified DSS as institutional or ad hoc. Institutional DSS support decisions that are recurring. An ad hoc DSS supports querying data for one time requests. Hackathorn and Keen (1981) identified DSS in three distinct yet interrelated categories: Personal DSS, Group DSS and Organizational DSS.

In 1979, John Rockart of the Harvard Business School published a ground breaking article that led to the development of executive information systems (EISs) or executive support systems (ESS). Rockart developed the concept of using information systems to display critical success metrics for managers.

CHAPTER №1. Main information.

This chapter devotes to main concepts, which will be used in this work. I will introduce you with major terms of Decision Support Systems, its benefits and structure. Also, I will give my own opinion about café idea, its location, and influencing factors to this area. In addition, this part is devoted to environment, which was used during creating application. I will give a little information about programming language I chose, Data Base, which I used, and relevant programs.

1.1 Terms and major concepts in DECISION SUPPORT SYSTEM.

A decision support system (**DSS**) is a computer-based information system that supports business or organizational decision-making activities. DSSs serve the management, operations, and planning levels of an organization and help to make decisions, which may be rapidly changing and not easily specified in advance.

A decision support system (**DSS**) is a computer program application that analyzes business data and presents it so that users can make business decisions more easily. It is an "informational application" (to distinguish it from an "operational application" that collects the data in the course of normal business operation). Typical information that a decision support application might gather and present would be:

- Comparative sales figures between one week and the next
- Projected revenue figures based on new product sales assumptions
- The consequences of different decision alternatives, given past experience in a context that is described
- Accessing all of your current information assets, including legacy and relational data sources, cubes, data warehouses, and data marts

A decision support system may present information graphically and may include an expert system or artificial intelligence (AI). It may be aimed at business executives or some other group of knowledge workers.

DSSs include knowledge-based systems. A properly designed DSS is an interactive software-based system intended to help decision makers compile useful

information from a combination of raw data, documents, personal knowledge, or business models to identify and solve problems and make decisions.

1.2 Taxonomies

A Decision Support System (DSS) is an interactive computer-based system or subsystem intended to help decision makers use communications technologies, data, documents, knowledge and/or models to identify and solve problems, complete decision process tasks, and make decisions. Decision Support System is a general term for any computer application that enhances a person or group's ability to make decisions. Also, Decision Support Systems refers to an academic field of research that involves designing and studying Decision Support Systems in their context of use. In general, Decision Support Systems are a class of computerized information system that support decision-making activities. Five more specific Decision Support System types include:

- A communication-driven DSS supports more than one person working on a shared task; examples include integrated tools like Microsoft's NetMeeting or Groove
- A data-driven DSS or data-oriented DSS emphasizes access to and manipulation of a time series of internal company data and, sometimes, external data.
- A document-driven DSS manages, retrieves, and manipulates unstructured information in a variety of electronic formats.
- A knowledge-driven DSS provides specialized problem-solving expertise stored as facts, rules, procedures, or in similar structures.
- A model-driven DSS emphasizes access to and manipulation of a statistical, financial, optimization, or simulation model. Model-driven DSS use data and parameters provided by users to assist decision makers in analyzing a situation; they are not necessarily data-intensive.

Components

Three fundamental components of a DSS architecture are:

- the database (or knowledge base),
- the model (i.e., the decision context and user criterion), and
- the user interface.

The users themselves are also important components of the architecture

1.3 Classification

There are several ways to classify DSS applications. Not every DSS fits neatly into one of the categories, but may be a mix of two or more architectures.

Holsapple and Whinston classify DSS into the following six frameworks: Text-oriented DSS, Database-oriented DSS, Spreadsheet-oriented DSS, Solver-oriented DSS, Rule-oriented DSS, and Compound DSS.

A compound DSS is the most popular classification for a DSS. It is a hybrid system that includes two or more of the five basic structures described by Holsapple and Whinston.

The support given by DSS can be separated into three distinct, interrelated categories: Personal Support, Group Support, and Organizational Support.

DSS components may be classified as:

- Inputs: Factors, numbers, and characteristics to analyze
- User Knowledge and Expertise: Inputs requiring manual analysis by the user
- Outputs: Transformed data from which DSS "decisions" are generated
- Decisions: Results generated by the DSS based on user criterion

DSSs which perform selected cognitive decision-making functions and are based on artificial intelligence or intelligent agents technologies are called Intelligent Decision Support Systems (IDSS).

The nascent field of Decision engineering treats the decision itself as an engineered object, and applies engineering principles such as Design and Quality assurance to an explicit representation of the elements that make up a decision.

Benefits:

1. Improves personal efficiency
2. Speed up the process of decision making
3. Increases organizational control
4. Encourages exploration and discovery on the part of the decision maker
5. Speeds up problem solving in an organization
6. Facilitates interpersonal communication
7. Promotes learning or training
8. Generates new evidence in support of a decision
9. Creates a competitive advantage over competition
10. Reveals new approaches to thinking about the problem space
11. Helps automate managerial processes

Destination of the Decision Support System. This system doesn't offer the single and the best choice, instead of it, this system is able to suggest to you several alternatives, which are differed by different criterion. And I think, it is the most valuable feature of this system, because in this case, we are able to look to our task from various sides. What is more, each person has diverse point of view and Decision Support System helps satisfy desire of each of them.

1.4 Information about CAFÉ

My task is selecting appropriate area for café. Relevant location presumes the successful future and developing your café at the certain place. The future of your café, its success depends on several main factors. And the next part of this work explains the importance of my work, its value.

Selecting the Best Location for Bar and Café

Almost half of the success of a bar and café business depends on its location and area. If a bar and café is located at a place where lots of people pass or come there for business, it is bound to attract more customers during the day and make good profits. Thus, it would not be wrong to say that selecting the best location for a bar

and café can contribute immensely in making it more popular and more profitable venture.

Locality and area matter a lot in successful running of a bar and café as a busy place with lots of people passing by offers opportunities for people to grab a bite to eat when they feel hungry instead of ordering or going some place else when they have got a bar and café nearby serving scrumptious food and drinks.

Before selecting location for our café, we must solve some questions about this problem.

Tips on Selecting the Best Location for Bar and Café

First of all decide on the type of bar and café you want to establish. It is going to be a neighborhood bar, a sports bar, a specialty bar or a brew pub and beer bar. If you have decided the type of bar and café, it will be very easy to find the best location.

For a neighborhood bar, a residential or a mix type area near to house would be good where people can bring in their friends or guests for lunch dinner or snacks. It is a good place to bring children too where they can get fast food and drinks; thus a sober and clean environment is important for a neighborhood bar.

For sports bar, specialty bar or brewpub and bar, select a commercial place near to shopping mall, commercial area or a busy area where all kinds of people come. Opening in a busy place is always beneficial as it can bring in more customers and it can also advertise your business by just being there in the midst of things.

Keep in mind the price range and the kind of food you will serve when searching for the best location to open a bar and café. Fast food, simple dinners and lunches and drinks are best as they attract all kinds of people from young to old and kids. However, as a more sophisticated platter which is made of Italian or French food is definitely going to be expensive so look for a posh area to open such a bar and café.

Open your bar and café in a place where no such bar or café exists or even if they do exist, try to introduce something unique, something better or something entirely different from it to make your business successful. For example, if the other bar

and café within a locality offers only lunch and dinner, you can start with breakfast too. This will be a good motivating point for customers. Make sure to do something different that your customers will like.

If you are opening a bar and café in a simple and middle class area, keep the prices to match. If the prices are too high, the local will not be able to afford which means loss of business. Thus it is important to make the price list according to the locality of the bar and café.

Follow these tips on selecting the best location for a bar and café to ensure your business does well and gets you good returns on investment. Opening a bar and café at the right location is very important for its success and popularity among the customers so make sure you reach your target market by opening in the best locality.

Influencing factors.

In order to choose the most appropriate location for café, we must regard several factors very properly which are able to define the future of our café and its success.

- The first factor is the price. My system offer several ways for café establishing at the certain address. We can take already exist building for a rent or we are also able to build new building and of course we can just buy exist café and open our doors. In the most cases, the price is crucial factor which influences to our decision. That is why, we must refer to it carefully.
- The second one is presence the enterprise in the neighbor area. It implies that a lot of workers after the work or during the break will visit our café and bring income to us. The presence of enterprise presumes a lot of people which will work in this area and will be hungry and will be needed in food.
- The third factor is habitat level. It connotes that a lot of people will live near this area and it implies that they will go for walk and see this café and like the curious people they will visit it to test the food and spend good time there with family, friends and may be just to eat. Also it presumes the possibility, that they will be able to order the food to their houses.

- The next one is location quality. We can say that it is not correctly to establish new café in the non-comfortable sites. People will not visit such kind of places where they can not receive satisfaction from spent time, where difficult to get to such place. That is why, it is very important to establish the new café in the correct place with sufficient quality of environment.
- The fifth factor is the presence of stops in this area for example for taxi-drivers or bus-drivers. It presumes that such kind of people after or before the work will visit this place to eat or just to have a break, for a rest. As the outcome, they will bring prominent revenue for our business.
- And the last one is the location of entertainment near this area. It is very important factor, because parks, trade streets or other entertainment establishments where people walk with their family, friends, where they enjoy and buy gifts, such kind of places attract a lot of people and after this activity they prefer to eat very often, especially during some holidays or other galas.

And in addition I want to say, that these factors are evaluated by 100 present rate and we are able to insert new addresses with these features in the Data Base. I think it is the most appropriate way to choose the best location because in this case, we will be able to regard more alternatives for our café. We must assess these factors if 100% is the highest mark for each feature and 1% is the worst mark for it. This factors can not be computed, we just are able to use it from special documents, where they have already been calculated by special staff and individual user`s opinion. Also, one of the virtues of this project is that the user is able to change the rate of these factors if he is sure that they are not correct when the system is running or by changing the information in Data Base.

Also, I must emphasize, that in the Data Base of my system, we are able to see the risk column. This column, shows the part and share of the risk, when we want to

select certain area for establishing our cafe. It is not calculated by special computations, no, instead, we can measure it by special statistical data, which we face in special documentation about this area. This risk rate, expresses the digit feature of flood, earthquake or other possibility that this location may be busy.

1.5 Information environment.

1.5.1 C Sharp

Here I must introduce you with my work environment. First of all, I must emphasize, that, in order to realize my idea and implement my application, I used C#(C Sharp programming language). And of course I have to explain my decision. Further I will give some additional information about C#, but now I want to tell you, why I chose C#. The C# is the next version after another programming language C++. C# is its descendant and C++ is its ancestor. C# give us opportunity to create a lot of various applications from different spheres like Console Applications, Windows Applications, Web applications and others. It has understandable syntax(it is almost the same with C++ syntax) but at the same time, it offers more opportunity and easier ways to solve your problems during writing a code, because of its very useful functions, which simplify your task by writing less code and let to perform your task faster. Another words, these functions simplify programmer`s work.

And as I said, now I want to give some information about C# to broad your view point about it.

C# (pronounced see sharp) is a multi-paradigm programming language encompassing strong typing, imperative, declarative, functional, generic, object-oriented (class-based), and component-oriented programming disciplines. It was developed by Microsoft within its .NET initiative and later approved as a standard by Ecma (ECMA-334) and ISO (ISO/IEC 23270:2006). C# is one of the programming languages designed for the Common Language Infrastructure.

C# is intended to be a simple, modern, general-purpose, object-oriented programming language.[6] Its development team is led by Anders Hejlsberg. The most recent version is C# 4.0, which was released on April 12, 2010.

- **Paradigm(s)** multi-paradigm: structured, imperative, object-oriented, event-driven, functional, generic, reflective
- **Appeared in** 2001
- **Designed by** Microsoft
- **Developer** Microsoft
- **Stable release** 4.0 (April 12, 2010; 23 months ago)
- **Typing discipline** static, dynamic,[1] strong, safe, nominative
- **Major implementations** Visual C#, .NET Framework, Mono
- **Dialects** C ω , Spec#, Polyphonic C#
- **Influenced by** C++,Eiffel, Java, Modula-3, Object Pascal
- **Influenced** D, F#, Java 5,[5] Nemerle, Vala
- **Platform** Common Language Infrastructure
- **License** CLR is proprietary, Mono compiler is dual GPLv3, MIT/X11 and libraries are LGPLv2, DotGNU is dual GPL and LGPLv2
- **Usual filename extensions** .cs

Design goals

The ECMA standard lists these design goals for C#:

- C# language is intended to be a simple, modern, general-purpose, object-oriented programming language.
- The language, and implementations thereof, should provide support for software engineering principles such as strong type checking, array bounds checking, detection of attempts to use uninitialized variables, and automatic garbage collection. Software robustness, durability, and programmer productivity are important.

- The language is intended for use in developing software components suitable for deployment in distributed environments.
- Source code portability is very important, as is programmer portability, especially for those programmers already familiar with C and C++.
- Support for internationalization is very important.
- C# is intended to be suitable for writing applications for both hosted and embedded systems, ranging from the very large that use sophisticated operating systems, down to the very small having dedicated functions.
- Although C# applications are intended to be economical with regard to memory and processing power requirements, the language was not intended to compete directly on performance and size with C or assembly language.

1.5.2 SQL.

In my application I used the Data Base, which reflects certain properties of the location. These features help me to do the correct choice and select certain location. C# offers several ways for reflection and working with data base. The easiest way is the next, I could just connect special form “Data Grid View”(which reflects data base in table form) from “Microsoft Visual Studio” with my Data Base and work with it, make some computations without any queries. But, instead of it, I decided to use SQL requests to show my knowledge. And now, I am going to give you a little information about SQL and Microsoft Visual Studio.

First of all, I had to create my data base. To do it, I had to work with relational database server, relational database management system. Relational database management system- is the main system of relational database server, which allows to manage with data in relational data base.

Microsoft SQL Server is a relational database server, developed by Microsoft: it is a software product whose primary function is to store and retrieve data as requested by other software applications, be it those on the same computer or those running on another computer across a network (including the Internet). There are at least a dozen different editions of Microsoft SQL Server aimed at different audiences and for different workloads (ranging from small applications that store

and retrieve data on the same computer, to millions of users and computers that access huge amounts of data from the Internet at the same time). True to its name, Microsoft SQL Server's primary query languages are T-SQL and ANSI SQL.

SQL (referred to as Structured Query Language) is a programming language designed for managing data in relational database management systems (RDBMS). Originally based upon relational algebra and tuple relational calculus, its scope includes data insert, query, update and delete, schema creation and modification, and data access control.

SQL was one of the first commercial languages for Edgar F. Codd's relational model, as described in his influential 1970 paper, "A Relational Model of Data for Large Shared Data Banks". Despite not adhering to the relational model as described by Codd, it became the most widely used database language. Although SQL is often described as, and to a great extent is, a declarative language, it also includes procedural elements. SQL became a standard of the American National Standards Institute (ANSI) in 1986, and of the International Organization for Standards (ISO) in 1987. Since then, the standard has been enhanced several times with added features. However, issues of SQL code portability between major RDBMS products still exist due to lack of full compliance with, or different interpretations of, the standard. Among the reasons mentioned are the large size and incomplete specification of the standard, as well as vendor lock-in.

Queries

The most common operation in SQL is the query, which is performed with the declarative SELECT statement. SELECT retrieves data from one or more tables, or expressions. Standard SELECT statements have no persistent effects on the database. Some non-standard implementations of SELECT can have persistent effects, such as the SELECT INTO syntax that exists in some databases.

Queries allow the user to describe desired data, leaving the database management system (DBMS) responsible for planning, optimizing, and performing the physical operations necessary to produce that result as it chooses.

A query includes a list of columns to be included in the final result immediately following the SELECT keyword. An asterisk ("*") can also be used to specify that the query should return all columns of the queried tables. SELECT is the most complex statement in SQL, with optional keywords and clauses that include:

- The FROM clause which indicates the table(s) from which data is to be retrieved. The FROM clause can include optional JOIN subclauses to specify the rules for joining tables.
- The WHERE clause includes a comparison predicate, which restricts the rows returned by the query. The WHERE clause eliminates all rows from the result set for which the comparison predicate does not evaluate to True.
- The GROUP BY clause is used to project rows having common values into a smaller set of rows. GROUP BY is often used in conjunction with SQL aggregation functions or to eliminate duplicate rows from a result set. The WHERE clause is applied before the GROUP BY clause.
- The HAVING clause includes a predicate used to filter rows resulting from the GROUP BY clause. Because it acts on the results of the GROUP BY clause, aggregation functions can be used in the HAVING clause predicate.
- The ORDER BY clause identifies which columns are used to sort the resulting data, and in which direction they should be sorted (options are ascending or descending). Without an ORDER BY clause, the order of rows returned by an SQL query is undefined.

The SQL language consists of 3 language subsets-The Data manipulation Language, The Data Definition Language and The Data Control Language.

Data manipulation

The Data Manipulation Language (DML) is the subset of SQL used to add, update and delete data:

- INSERT adds rows (formally tuples) to an existing table
- UPDATE modifies a set of existing table rows
- DELETE removes existing rows from a table

- **MERGE** is used to combine the data of multiple tables. It combines the **INSERT** and **UPDATE** elements. It is defined in the SQL:2003 standard; prior to that, some databases provided similar functionality via different syntax, sometimes called "upsert"

Data definition

The Data Definition Language (DDL) manages table and index structure. The most basic items of DDL are the **CREATE**, **ALTER**, **RENAME**, **DROP** and **TRUNCATE** statements:

- **CREATE** creates an object (a table, for example) in the database
- **ALTER** modifies the structure of an existing object in various ways, for example, adding a column to an existing table or a constraint
- **TRUNCATE** deletes all data from a table in a very fast way, deleting the data inside the table and not the table itself.
- **DROP** deletes an object in the database, usually irretrievably

Data control

The Data Control Language (DCL) authorizes users and groups of users to access and manipulate data. Its two main statements are:

- **GRANT** authorizes one or more users to perform an operation or a set of operations on an object.
- **REVOKE** eliminates a grant, which may be the default grant.

1.5.3 Visual Studio

To be able to work with C# and use Data Base, reflect it and create subscriber's interface I used "Visual Studio 2008".

Microsoft Visual Studio includes native support for data programming with Microsoft SQL Server. It can be used to write and debug code to be executed by SQL CLR. It also includes a data designer that can be used to graphically create, view or edit database schemas. Queries can be created either visually or using code. SSMS 2008 onwards, provides intelligence for SQL queries as well.

1.5.4 SQL Server Management Studio

In order to create my Data Base with all necessary information I used SQL Server Management Studio. SQL Server Management Studio is a GUI tool included with SQL Server 2005 and later for configuring, managing, and administering all components within Microsoft SQL Server. The tool includes both script editors and graphical tools that work with objects and features of the server. SQL Server Management Studio replaces Enterprise Manager as the primary management interface for Microsoft SQL Server since SQL Server 2005. A version of SQL Server Management Studio is also available for SQL Server Express Edition, for which it is known as SQL Server Management Studio Express (SSMSE).

A central feature of SQL Server Management Studio is the Object Explorer, which allows the user to browse, select, and act upon any of the objects within the server. It can be used to visually observe and analyze query plans and optimize the database performance, among others. SQL Server Management Studio can also be used to create a new database, alter any existing database schema by adding or modifying tables and indexes, or analyze performance. It includes the query windows which provide a GUI based interface to write and execute queries.

Conclusion

In this chapter, I made the deep analysis in sphere of Decision Support Systems and selecting area by destination, regarded the main concepts, which we must know during working with Decision Support System and its major goals. Also, it were deliberated the most important factors, which are influenced on selecting area for café. I expressed my own ideas about this problem and way for its solution. What is more, I used information about tools, which helped me to find the solution of this assignment.

Chapter №2. Mathematical criterion and explorations.

This chapter of my work is dedicated mathematical components of my application. I used a lot of formulas, which were invented by famous scientists, others, were invented by me during analytic process. First of all, I want to explain the “Decision matrix” idea. And now, I want to give you some additional information about it.

Do you ever find yourself in a quandary when trying to choose between several options? Do you ever make a list of pros and cons to help you decide which path to take in life? Do you ever wish you could be sure you were making objective decisions and not just being driven by emotional preferences?

Now, I use it to make important decisions like:

- Which vacation location to visit
- Which candidate to hire for a job
- Which job offer to accept
- Which apartment to rent
- Which software to purchase
- Whether to lease or purchase a car

You can use the decision matrix to make any decision you face in your personal or business life. You can use it alone, with your partner, or with a business team. The only limit to how you can harness the power of the decision matrix is your own imagination. If you have an important decision to make and you want to reduce or eliminate the chances of second guessing yourself later, you should try the decision matrix.

2.1 Decision Matrix

A decision matrix is a list of values in rows and columns that allows an ologist to systematically identify, analyze, and rate the performance of relationships between sets of values and information. Elements of a decision matrix show decisions based on certain decision criterion. The matrix is useful for looking at large masses of decision factors and assessing each factor’s relative significance. Decision matrix is used to describe a multi-criterion decision analysis (MCDA) problem.

Decision matrix is one of the simplest decision making techniques. You identify the key elements of your decision making situation. You assign each such element a special number (score). You then put those numbers into a special table and use a simple calculation to tell you which choice is the best.

This allows you to process a complex decision on paper, one byte at a time, without choking your mind with too many details at the same time. A decision making matrix also makes it easier for you to comprehend the bigger picture of your situation.

Decision Matrix Definition

A decision matrix allows decision makers to structure, then solve their problem by:

- specifying and prioritizing their needs with a list a criterion; then
- evaluating, rating, and comparing the different solutions; and
- selecting the best matching solution.

Description

A decision matrix evaluates and prioritizes a list of options. The team first establishes a list of weighted criterion and then evaluates each option against those criterion.

When to Use a Decision Matrix

- When a list of options must be narrowed to one choice.
- When the decision must be made on the basis of several criterion.
- After the list of options has been reduced to a manageable number by list reduction.

Typical situations are:

- When one improvement opportunity or problem must be selected to work on.
- When only one solution or problem-solving approach can be implemented.
- When only one new product can be developed.

To build a decision matrix you first need to decompose your decision making context into its basic elements, its building blocks. There are two main types of such building blocks. The first type is your options or **alternatives** from which you

are choosing. Write those down. You may want to extend your options further by taking another sheet of paper and going through a brainstorming exercise.

The second type of decision elements is the **selection criterion** you can use for judging your options. For example, various kinds of costs or losses, as well as benefits (in terms of money, time savings, health, fun, and so on). Think carefully about those and list all of them on paper. Look at them again. Can you make them more clear and specific? Are you missing any other relevant factors?

The benefits

Here are some benefits you can expect from using the decision matrix:

- You understand how and why you choose one option over another.
- Your decision fits with your priorities.
- You evaluate each option on its own merit without directly comparing it to other options.
- You use logic rather than emotion to make the decision.
- You get a chance to think about what you really want.

2.2 Data Table

Before computing expressions, we must know which data to use. That is why, first of all, I have to show my Data Table which includes all necessary factors, columns for results and required information. Further I will show you my Data Table I used in this application.

All information I used is situated at one single table. The reason of my decision is very simple. I wanted to simplify my work and make the creating process easier.

Table 2 "Café Data Table "

Address	nvarchar(50)
Risk	int
Rest_rent	int
Rest_build	int
Rest_purchase	int
Enterprice	int
Habitat_level	int
Location_quality	int
Stops	int
Entertainment	int
Pref_LPR	nvarchar(50)
Opt_variant	nvarchar(50)

2.3 Using of the Decision Matrix within practice.

In my case, I used just 3 criterion, which are the rent price of the café, the price we need to build it and the price we need to buy the building for our café. And like alternatives, I used 9 addresses in my data base, which are relevant for our café.

So, my matrix consists of 3 columns(criterion) and 9 alternatives(addresses). And the each cell of my matrix can be formed by multiplication of the risk of certain address to the price of café at this address.

$$A_{ij} = R_i * S_j$$

where A_{ij} – the value of the certain cell of decision matrix

R_i – the value of the risk

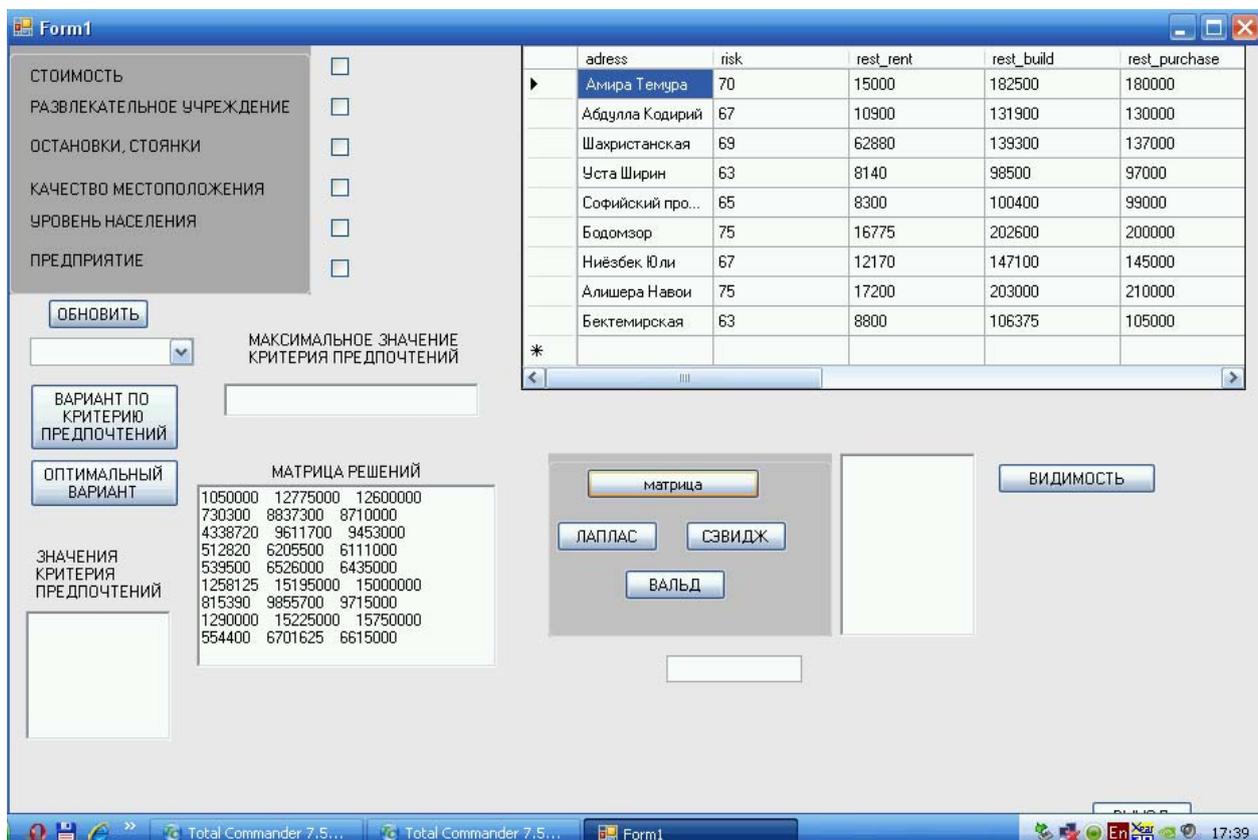
S_j – certain price(for rent, buying or building café at the relevant address)

The next my task is the expressing this formula by programming language:

```
Z = new double[10, 10];

    for (int i = 0; i < 9; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            Z[i, j] = Convert.ToDouble(dataGridView1[j + 3, i].Value) *
                Convert.ToDouble(dataGridView1[2, i].Value);
        }
    }
```

This strings of code reveals the building of the decision matrix, where Z is our matrix.



Pic.1 "Decision matrix"

Now, we have formed our decision matrix, but to finish our task and choose the most appropriate alternative for us, we need to use special formulas. In my application I used Laplace, Wald and Savage criterions. Now, I will tell you about this criterion.

2.4 Laplace criterion.

His formula calculates for each string of decision matrix his arithmetical mean. And the most optimal solution is the maximum value from these arithmetical means. It looks like this one:

$$F = \max \frac{1}{n} \sum_{i=1}^n a_{ij}$$

And now I want to show you how I expressed this formula in my application by the programming language.

```
double[] sr = new double[100];

for (int i = 0; i < 9; i++)
{
    for (int j = 0; j < 3; j++)
    {

        sr[i] += Z[i, j];

    }
    sr[i] = sr[i] / 3;
    richTextBox4.Text += sr[i] + "\n";
}
double max = sr[0];
for (int i = 0; i < 9; i++)
{
    if (max < sr[i])
    {
        max = sr[i];
    }
}
}
```

adress	risk	rest_rent	rest_build	rest_purchase
Амира Темура	70	15000	182500	180000
Абдулла Кодирий	67	10900	131900	130000
Шахристанская	69	62880	139300	137000
Уста Ширин	63	8140	98500	97000
Софийский про...	65	8300	100400	99000
Бодомзор	75	16775	202600	200000
Нийэзбек Юли	67	12170	147100	145000
Алишера Навои	75	17200	203000	210000
Бектемирская	63	8800	106375	105000

МАТРИЦА РЕШЕНИЙ

1050000	12775000	12600000
730300	8837300	8710000
4338720	9611700	9453000
612820	6205500	6111000
539500	6526000	6435000
1258125	15195000	15000000
815390	9855700	9715000
1290000	15225000	15750000
554400	6701625	6615000

Видимость

8808333.33333333
6092533.33333333
7801140
4276440
4500166.66666667
10484375
6795363.33333333
10755000
4623675

10755000

Pic.2”Laplace criterion”

I think this criterion is the most appropriate personally for me, because it is like the "gold middle", it regards the risk and positive factors together, and provides lucky selection.

2.5 Wald's criterion

This criterion computes for each string of the matrix the minimal value. And the optimal solution is the maximal value from these minimal values. It looks like this:

$$F = \max(\min a_{ij})$$

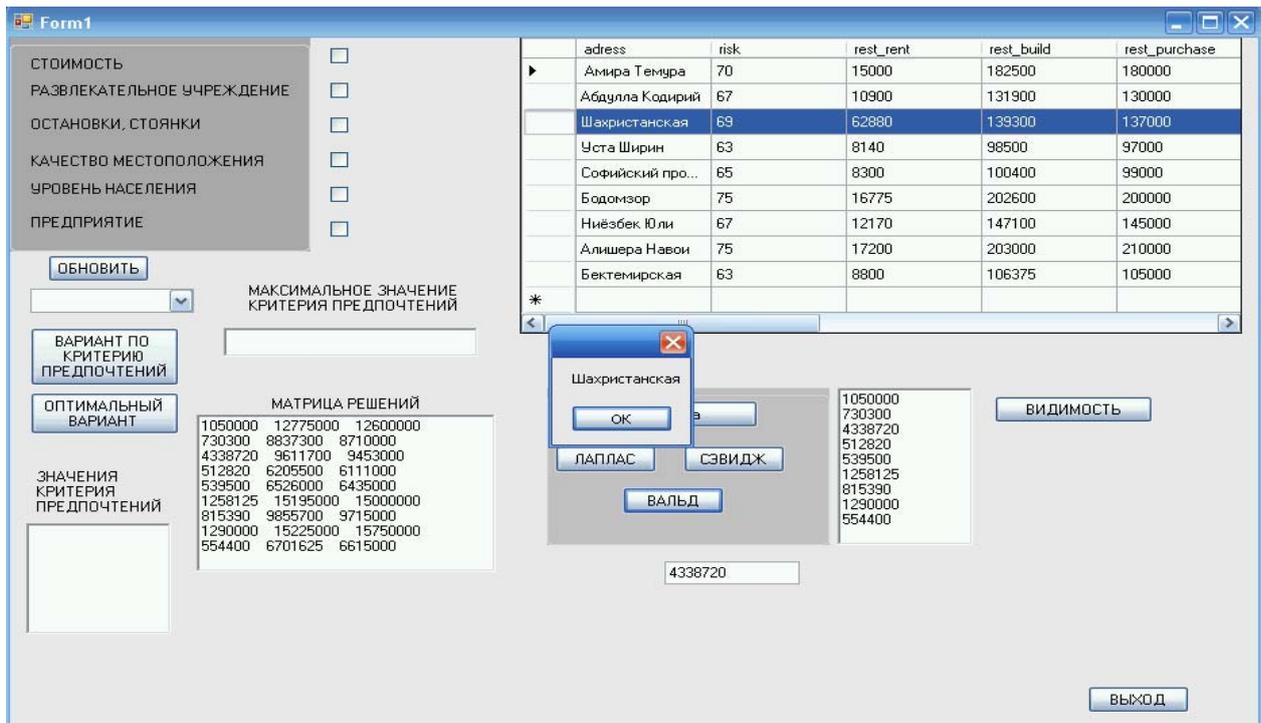
And in my application it is programmed like this one:

```
double[] min = new double[100];

    for (int i = 0; i < 9; i++)
    {
        double m = Z[i, 0];
        for (int j = 0; j < 3; j++)
        {
            if (m > Z[i, j])
            {
                m = Z[i, j];
            }
        }

        min[i] = m;
        richTextBox4.Text += min[i] + "\n";
    }

double max1 = min[0];
for (int i = 0; i < 9; i++)
{
    if (max1 < min[i])
    {
        max1 = min[i];
    }
}
```



Pic.3”Wald `s criterion”

If we speak about this criterion, I think it is not bad, but also not good choice, because it deliberates high positive factors but at the same moment, it regards high risk rate. That is why, I do not like it.

2.6 Savage`s criterion

His formula selects the maximal value for each column of the matrix. Then, we have to create another matrix by subtracting this maximal value from each column of primary matrix and value from each cell of this primary matrix . After, we must find the maximal value from each string. And the optimal solution is the minimal value from these selected marks.

$$R = (\max a_{ij}) - a_{ij}$$

$$F = \min(\max R_{ij})$$

```
double[] max = new double[10];

for (int i = 0; i < 3; i++)
{
    double m = Z[i, 0];
    for (int j = 0; j < 9; j++)
    {
```

```

        if (m < Z[j, i])
        {
            m = Z[j, i];
        }

    }
    max[i] = m;
    richTextBox4.Text += max[i] + "\n";

}
richTextBox4.Clear();

double[,] A = new double[10, 10];

for (int i = 0; i < 9; i++)
{
    for (int j = 0; j < 3; j++)
    {
        A[i, j] = max[j] - Z[i, j];
        richTextBox4.Text += A[i, j].ToString() + "  ";

    }
    richTextBox4.Text += "\n";
}

richTextBox4.Clear();

double[] min = new double[100];

for (int i = 0; i < 9; i++)
{
    double m = A[i, 0];
    for (int j = 0; j < 3; j++)
    {
        if (m < A[i, j])
        {
            m = A[i, j];
        }

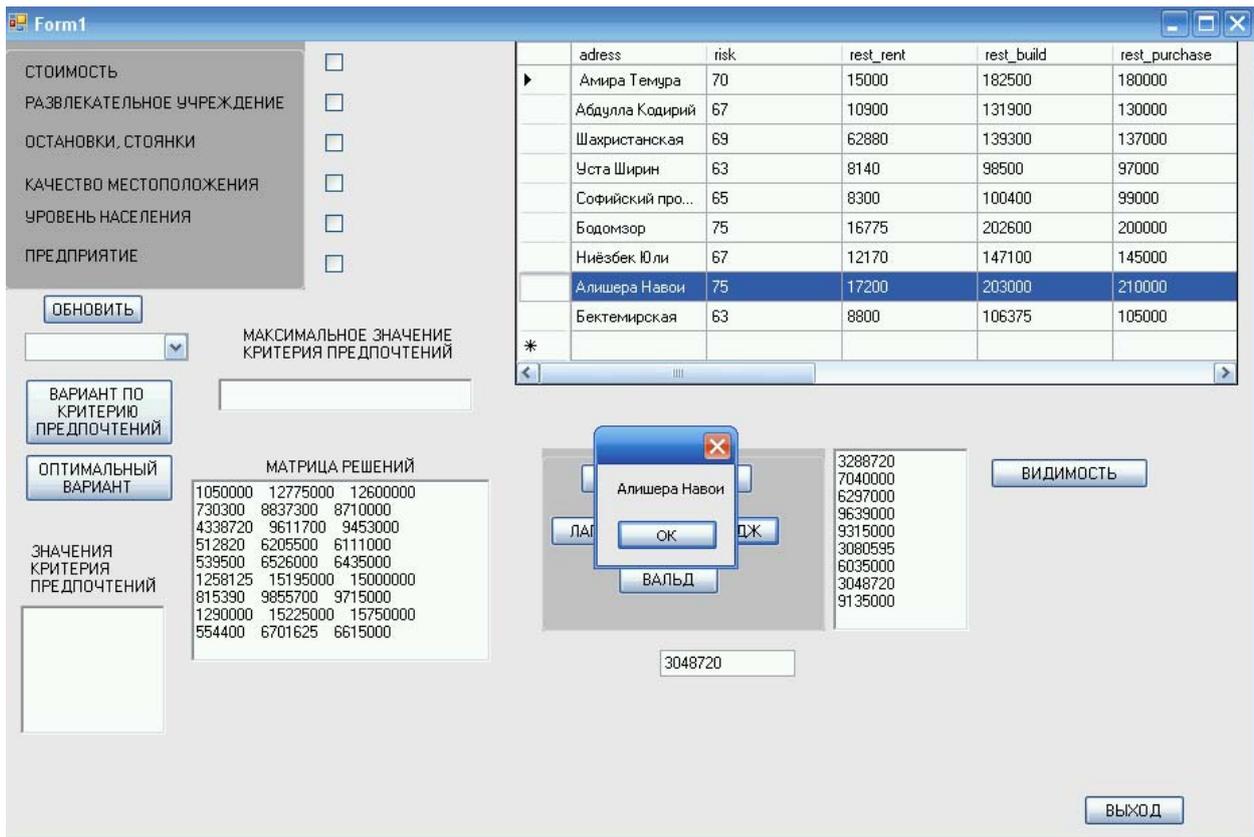
    }

    min[i] = m;
    richTextBox4.Text += min[i] + "\n";
}

double max1 = min[0];
for (int i = 0; i < 9; i++)
{
    if (max1 > min[i])
    {
        max1 = min[i];
    }

}

```



Pic.4 "Savage's criterion"

And eventually about this last criterion. I think this way is the most appropriate for users, which do not want to lose a lot of money and time, and want to find the location with the least risk rate.

2.7 User's preferences

If we use the Decision Support System, we have to emphasize, that it must use the preferences of the user. It is very important criterion of the application, because it helps to select the best location by means of user's view point. In my application, I used the most significant factors which admit to choose the best location. It are: the price of the café(its building or buying) at this area, presence of entertainments or other establishments, stops for the drivers, quality location, population rate and the presence of enterprises.

So, my application uses these factors and the user is able to choose which criterion are the most important personally for him. What is more, the user is able to select

the price mode: he has the opportunity to select the price for the rent, buying or building café and it is very useful feature of this application. Further I will show the program code which performs these actions.

First of all, to select the relevant alternative, user must select required factors on the application interface, then, program will sum the values of these factors together.

$$F = \sum_{i=1}^n A_i$$

where

F-our total value,

A_i -selected factors

But, if the price is like the required factor, the program will sum all another factors and after these actions, it will divide this total value on price value, which depends on price mode(building, rent or buying the café at the certain area).

$$F = \left(\sum_{i=1}^n A_i \right) / S$$

where

F-our total value,

A_i -selected factors

S-relevant price value

```
double[] A = new double[100];
switch (comboBox1.Text)
{
    case "АРЕНДА":
        for (int j = 0; j < 9; j++)
        {
            for (int i = 0; i < 6; i++)
            {
```

```

        if (c[5].Checked == false)
        {
            if (c[i].Checked == true && c[5].Checked == false)
            {
                A[j] += Convert.ToDouble(dataGridView1[i + 6, j].Value);
            }
        }
    }
    if (c[5].Checked == true)
    {
        for (int i = 0; i < 5; i++)
        {
            if (c[i].Checked == true)
            {
                A[j] += Convert.ToDouble(dataGridView1[i + 6, j].Value);
            }
        }
        A[j] = A[j] / Convert.ToDouble(dataGridView1[3, j].Value);
    }
}

```

I must underscore that price mode also depends on user, he chooses it. And this property of application makes it very comfortable and adaptive to user's preferences.

	adress	risk	rest_rent	rest_build	rest_purchase
▶	Амира Темура	70	15000	182500	180000
	Абдулла Кодирий	67	10900	131900	130000
	Шахристанская	69	62880	139300	137000
	Уста Ширин	63	8140	98500	97000
	Софийский про...	65	8300	100400	99000
	Бодомзор	75	16775	202600	200000
	Ниязбек Юли	67	12170	147100	145000
	Алишера Навои	75	17200	203000	210000
	Бектемирская	63	8800	106375	105000
*					

МАТРИЦА РЕШЕНИЙ		
1050000	12775000	12600000
730300	8837300	8710000
4338720	9611700	9453000
512820	6205500	6111000
539500	6526000	6435000
1258125	15195000	15000000
815390	9855700	9715000
1290000	15225000	15750000
554400	6701625	6615000

1050000
730300
4338720
512820
539500
1258125
815390
1290000
554400

Pic.5 "User's preferences"

2.8 Optimal variant

And eventually about optimal variant. This way proposes the most optimal variant for user, because it includes his preferences, in addition the user is able to evaluate required factors by 10 grade rate. By means of this way, application calculates the most relevant solution especially for each subscriber.

First of all, application multiplies factor values from the Data Base with this evaluated factors by subscriber. After this, we multiply computed value to data which we can be received by subtraction of 100 and the risk value from Data Base.

Mathematical formula looks like that:

when price is not selected factor

$$A_i = a_i * x$$

A_i -selected factors

a_i -value from Data Base for relevant factor

x -subscriber's rate

$$F = \left(\sum_{i=1}^n A_i \right) * (100 - R)$$

F -total value

R - risk value from Data Base for certain area

But when price is selected factor, we calculate relevant multiplication for factors from Data Base and evaluated factors by subscriber. Then, application computes multiplication for the price value from Data Base and evaluated rate by subscriber especially for price factor. After, we sum both of these values. And then, we multiply total value with data which we can be received by subtraction of 100 and the risk value from Data Base.

$$A_i = a_i * x$$

A_i-selected factors

a_i-value from Data Base for relevant factor

x-subscriber`s rate

$$B = b_i * y$$

B-price rate

b_i-value especially for price from Data Base

y-subscriber`s price rate

$$F = \left(\sum_{i=1}^n A_i + B \right) * (100 - R)$$

F-total value

R- risk value from Data Base for certain area

Now, I want to show you how these actions work and afford to you the code of these computations:

```
double[] opt = new double[100];
double[] A = new double[10];
double[] B = new double[10];
switch (comboBox1.Text)
{
    case "АРЕНДА":

        for (int i = 0; i < 9; i++)
        {
            for (int j = 0; j < 5; j++)
            {
                if (c[j].Checked == true && c[5].Checked == false)

                    A[i] += Convert.ToDouble(t[j].Text) *
                    Convert.ToDouble(dataGridView1[j + 6, i].Value);

            }

            if (c[5].Checked == true)
            {
                if (c[j].Checked == true)

                    A[i] += Convert.ToDouble(t[j].Text) *
                    Convert.ToDouble(dataGridView1[j + 6, i].Value);
            }
        }
    }
}
```

```

        B[i] = Convert.ToDouble(t[5].Text) *
Convert.ToDouble(dataGridView1[3, i].Value);

    }

    opt[i] = (A[i] + B[i]) * (100 -
Convert.ToDouble(dataGridView1[2, i].Value));

```

address	risk	rest_rent	rest_build	rest_purchase
Амира Темура	70	15000	182500	180000
Абдулла Коидрий	67	10900	131900	130000
Шахристанская	69	62880	139300	137000
Уста Ширин	63	8140	98500	97000
Софийский про...	65	8300	100400	99000
Бодомзор	75	16775	202600	200000
Ниезбек Юли	67	12170	147100	145000
Алишера Навои	75	17200	203000	210000
Бектемирская	63	8800	106375	105000

МАТРИЦА РЕШЕНИЙ		
1050000	12775000	12600000
730300	8837300	8710000
4338720	9611700	9453000
512820	6205500	6111000
539500	6526000	6435000
1258125	15195000	15000000
815390	9855700	9715000
1290000	15225000	15750000
554400	6701625	6615000

Pic.6 "Optimal variant"

Conclusion

In this chapter, it were deliberated mathematical spine of solution our problem. I used not only works and explorations of famous scientists in this sphere, what is more, I tried to invent new formulas and adapt it to our task by using important factors of location. In addition, I showed the its solution by programming and provided all necessary pictures, which prove the correct work this application.

Chapter №3. User`s manual.

This chapter of my work shows the rules of using my application. It includes all buttons and results of actions, which were explained in previous chapter.

First of all, I have to say, that, for success running our application, we must install to our computer “Microsoft Visual Studio 2008” and “SQL Express Manager” for working our Data Base. After this actions, we can launch our application. So, we have launched it, at the window we can see a lot of buttons and other forms. Our Data Base reflects at the form “Data Grid View”, and this form helps to us to underline certain alternative for solution of our task after several computations. We have several ways to find appropriate location for our café.

3.1 Decion Matrix buuton.

Firstly, we are able to form the Decision Matrix by pushing on the button “Матрица”, then, we can use three mathematical criterion to find the solution for our task(Laplace criterion, Savage criterion and Waldes criterion by pushing in the relevant buttons «Лаплас», «Севидж» и «Вальд»). Application show appropriate address on the screen and underline relevant string in Data Base at the “Data Grid View” form. In addition, we can see all necessary computations at our form by pushing on the “Видимость” button. In this case, we will be able to see our Decision Matrix and other computations.

The screenshot displays the application's interface. At the top, a data grid shows a list of addresses and their associated costs. The row for 'Алишера Навои' is highlighted. Below the grid, a dialog box titled 'Алишера Навои' is open, containing buttons for 'ОК', 'ЛАПЛАС', 'СЭВИДЖ', and 'ВАЛЬД'. To the left, a window titled 'МАТРИЦА РЕШЕНИЙ' displays a 3x3 matrix of numerical values. To the right, a window titled 'ВИДИМОСТЬ' displays a list of numerical values. At the bottom, a text box shows the value '10755000'.

address	risk	rest_rent	rest_build	rest_purchase
Амира Темура	70	15000	182500	180000
Абдулла Кодирий	67	10900	131900	130000
Шахристанская	69	62880	139300	137000
Уста Ширин	63	8140	98500	97000
Софийский про...	65	8300	100400	99000
Бодомзор	75	16775	202600	200000
Ниезбек Юли	67	12170	147100	145000
Алишера Навои	75	17200	203000	210000
Бектемирская	63	8800	106375	105000

МАТРИЦА РЕШЕНИЙ

12600000	12775000	12600000
8763600	8837300	8710000
52064640	9611700	9453000
6153840	6205500	6111000
6474000	6526000	6435000
15097500	15195000	15000000
9784680	9855700	9715000
15480000	15225000	15750000
6652800	6701625	6615000

Алишера Навои

ОК

ЛАПЛАС

СЭВИДЖ

ВАЛЬД

ВИДИМОСТЬ

8808333.33333333
6092533.33333333
7801140
4276440
4500168.66666667
10484375
6795363.33333333
10755000
4623675

10755000

Fig.7”Decision Matrix and computations”

3.2 User's preferences button

Secondly, we can find interested to us address by pushing on the “Вариант по критерию предпочтений” button. This button reflects computations of user's preferences. But firstly, we should select appropriate price mode (rent, buying or building) in combo box form, which is located at the left side of our window. After this, we must denote interested to us factors.

The screenshot shows a window titled "Form1" with a list of factors and checkboxes on the left. The factors are: СТОИМОСТЬ, РАЗВЛЕКАТЕЛЬНОЕ УЧРЕЖДЕНИЕ, ОСТАНОВКИ, СТОЯНКИ, КАЧЕСТВО МЕСТОПОЛОЖЕНИЯ, УРОВЕНЬ НАСЕЛЕНИЯ, and ПРЕДПРИЯТИЕ. The checkboxes for "КАЧЕСТВО МЕСТОПОЛОЖЕНИЯ" and "ПРЕДПРИЯТИЕ" are checked. Below the list are buttons "ОБНОВИТЬ" and "ПОСТРОЙКА" (a dropdown menu). To the right is a label "МАКСИМАЛЬНОЕ ЗНАЧЕНИЕ КРИТЕРИЯ ПРЕДПОЧТЕНИЙ" and a text input field containing "177". At the bottom is a large button labeled "ВАРИАНТ ПО КРИТЕРИЮ ПРЕДПОЧТЕНИЙ".

Pic.8”Selection price mode and required factors”

Then, we must just push on the button “Вариант по критерию предпочтений” to receive the result. Relevant area will be shown on the screen like a Message Box and will be underlined at the “Data Grid View” window's form which reflects our Data Base in the table form.

The screenshot shows a "Data Grid View" window with a table containing the following data:

address	risk	rest_rent	rest_build	rest_purchase
Амира Темура	70	15000	182500	180000
Абдулла Кодирий	67	10900	131900	130000
Шахристанская	69	62880	139300	137000
Уста Ширин	63	8140	98500	97000
Софийский про...	65	8300	100400	99000
Бодомзор	75	16775	202600	200000
Ниезбек Юли	67	12170	147100	145000
Алишера Навои	75	17200	203000	210000
Бектемирская	63	8800	106375	105000

Overlaid on the table is a message box titled "Амира Темура" with an "OK" button. Below the message box are three buttons: "ПАТНАС", "СЭВИДЖ", and "ВАЛЬД".

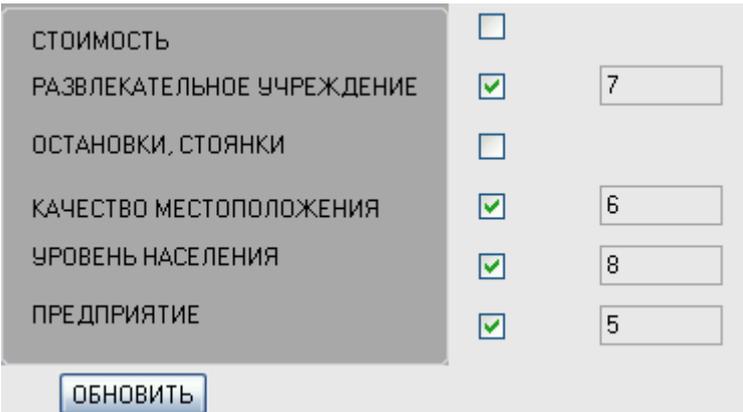
Pic.9”Result the user's preferences criterion”

3.3 Optimal variant button

And eventually, we may choose the Optimal Variant criterion by pushing on the “Оптимальный вариант” button.

To receive the result of this criterion, first of all, we have to select required to us factors and choose the price mode. Then we must push on the button “Обновить” to be able to evaluate these factors by 10 grade rate. You must remember that 1 grade is unimportant factor and 10 grade is the most important factor for subscriber.

By means of pushing on the button “Обновить”, text boxes will show in front of those factors we chose earlier. After, we should fill in these text boxes by digits from 1 to 10.

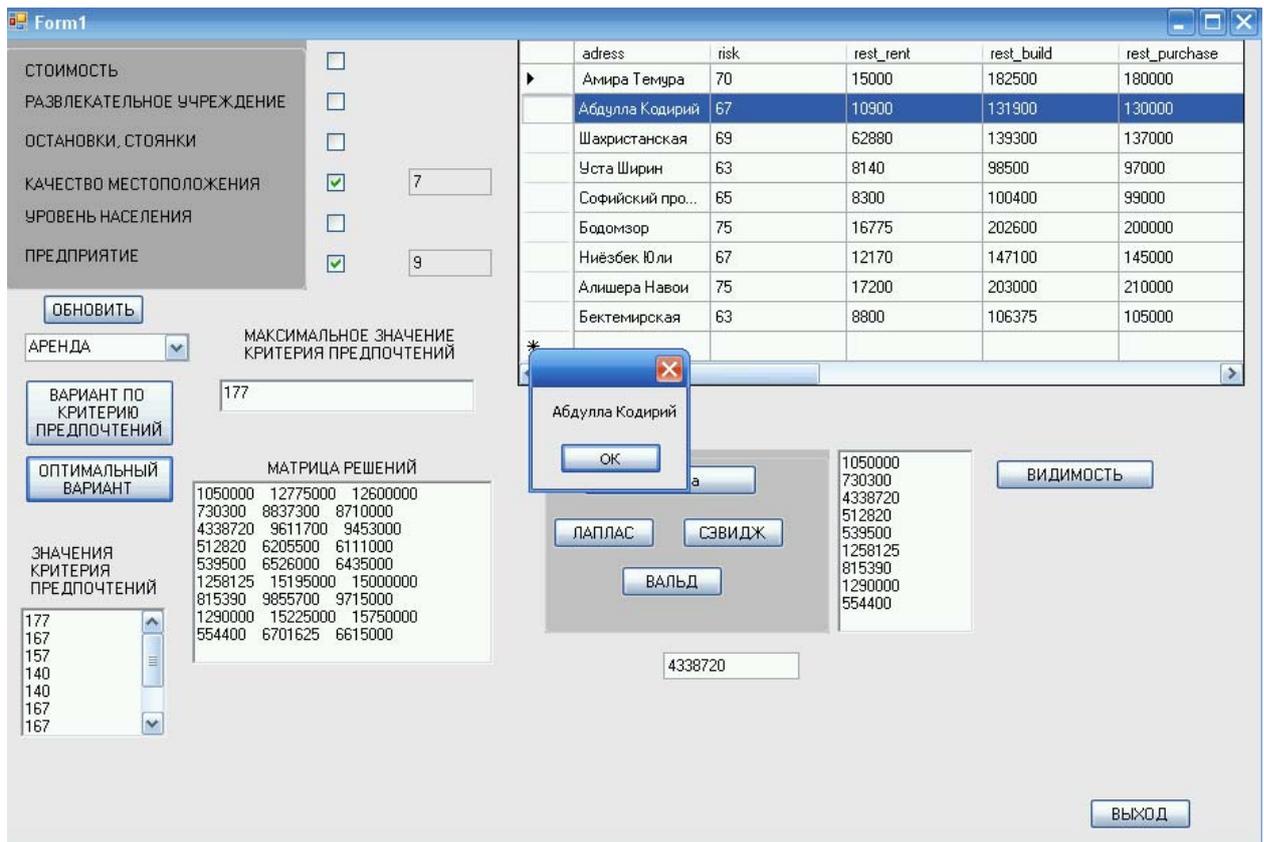


СТОИМОСТЬ	<input type="checkbox"/>	
РАЗВЛЕКАТЕЛЬНОЕ УЧРЕЖДЕНИЕ	<input checked="" type="checkbox"/>	7
ОСТАНОВКИ, СТОЯНКИ	<input type="checkbox"/>	
КАЧЕСТВО МЕСТОПОЛОЖЕНИЯ	<input checked="" type="checkbox"/>	6
УРОВЕНЬ НАСЕЛЕНИЯ	<input checked="" type="checkbox"/>	8
ПРЕДПРИЯТИЕ	<input checked="" type="checkbox"/>	5

ОБНОВИТЬ

Fig. 10 “Evaluation required factors”

And at the finish, we must push on the “Оптимальный вариант” button to get relevant area for our café. Our result will be also underscored on the table and will be shown on the screen like a message box.



Pic.11 “Optimal variant result”

Conclusion

This part of my work, implies the manual for user. I tried to show all required actions for correct work of this application with all necessary pictures, images and explanations.

CHAPTER 4. OCCUPATIONAL SAFETY AND SAFETY ENGINEERING

This part is intended for subscribers, they safety during working with computer and this application.

The working at personal computer is characterized by effecting of the following harmful factors to human's organism:

- The boosted value of a voltage;
- X-radiation originating at braking of an electron ray on an inside surface of a kinescope of a monitor;
- The dark blue luminophore of the monitor screen has partial radiation in ultra-violet domain of a spectrum;
- Electromagnetic waves of a low frequencies concerned with the operation of sweep circuits of a cathode ray tube of the monitor;
- Electromagnetic fields (effect of reflection);
- The intensive noise level;
- Electric waves (radiofrequency);
- Electrostatic field.

For provision of best conditions for effective and safe operation at PC it's necessary to establish such working conditions, which would be comfortable and reduce the effects of the given harmful factors as much as possible. It is essential, that the listed harmful factors match with the set rules and norms.

4.1. Safety of work

The noise is an aggregate of sounds effecting on human organism, and interfering to its operation and rest.

The researches show that in conditions of noise aural functions suffer first of all. But the effect of noise is not limited to influence only on hearing. It provokes noticeable shifts of a number of physiological and mental functions. The noise harmfully influences a nervous system and reduces velocity and accuracy of sense-motor processes, the number of errors at the solving of the intellectual tasks increases. The noise renders noticeable influence on human's attention and

provokes negative emotions.

The fundamental noisemaker in rooms where computer is situated is the air conditioning equipment, print and copy techniques, and fans of cooling systems in computers. According to the specifications the noise level on a work station should not exceed 59 dB. The normalized noise levels are provided by usage of sound-proof materials for rooms facing.

The fundamental measures of noise control are:

- Elimination or attenuation of noise factors in its source during design and maintenance of an equipment;
- Isolation of noisemakers from an environment particularly by a means of deafening and acoustical absorption;
- Rational design of product engineering.

4.1.1. Protection from electric radiation

According to the specifications, normalized parameters in a frequency band 60 kHz – 300 MHz are E and H intensities of an electromagnetic field. At frequency from 60 kHz up to 3 MHz the electric field intensity may be up to 50 V/m, and magnetic intensity can be up to 5A/m.

During the operation of the monitor the electric waves of very low frequencies are generated. The intensity of an electromagnetic field near the monitor is from 4 up to 7 milligauss. The magnetic intensity higher than 4 milligauss is harmful to human. Such radiation is a reason of anomalies at pregnancy and causes cellular level variations. Magnifying of quantity of positively charged ions in air near to the switched on monitor also negatively affects the human organism. The medical research, which have been carried out in USA, shows that long-time staying in deionized atmosphere effects on metabolism and results in variation of biochemical reaction in blood on cellular level, which quite often results in stresses.

The charge of a static electricity, resulting in originating of an electrostatic discharge is accumulated on the surface of a monitor during its work. The

protection from electromagnetic effect may be provided using screens with electro conductive surface. In a construction of these screens possibility of grounding, as a rule, is stipulated. These screens provide reliable protection against electric waves.

There are following types of screens:

- “Polaroid” type screens;
- Screens of a film type;
- Glass screens;
- Cancellated screens.

Among the representatives of above listed types there are screens with a conducting surface and grounding possibility. These screens protect the owner from electrostatic and electric waves of the monitor.

4.1.2. Protection from ionizing radiation

Ionizing radiation is a radiation, whose interaction with substance results in derivation of different types of ions in this substance.

Ionizing radiation consists of charged and not charged particles, to which photons also concern. Power of particles of ionizing radiation is measured in off-system units - electron - volts.

The source of ionizing radiation is a device which may let out ionizing radiation. With the beginning of application display engineering, in which cathode ray tubes are used, there has been a problem of protection of the operator from a harmful X-radiation, which source they are. At the development of modern displays the safe dozes of radiation are taken into account and everything is made to secure the human from harmful effect of radiation.

4.1.3. Protection from the effect of an electrostatic field

The fact of that the cathode ray tube in the monitor is charged negatively is a reason of electrostatic charge originating, which results in accumulation of positive aero ions. At the surplus of positive aero ions their positive charge starts to repel micro particles, which always present in air. They become dispersed and bombard the human’s face and eyes. It makes very negative effect (irritation of a nervous system and skin).

The contents of easy aero ions of both signs in a breath area of the working man practically can change in limits from 1,5-10 up to 5-10 in 1 cm of an air. The intensity of an electrostatic field should not increase to exceed these values.

It's rather preferable to use grounded protective filter for the protection from effect of an electrostatic charge.

Also, as protection from static electricity in rooms with PC it is possible to use neutralizers and humidifiers, and floors should have an antistatic coating.

For maintenance of normalized values of positive and negative ions concentration, it is recommended to install conditioners, devices for air ionization in work rooms with PCs, or to carry out natural airing lasting not less than 10 minutes after every 2 work hours.

For the purposes of preventing the harmful influence of motes with aero ions to an organism of the operating personnel it is necessary to carry out daily damp cleanup of rooms, and not less often than 1 time in scheduled work period delete a dust from screens at switched off monitor.

4.2. Working conditions

Appropriate working conditions in workplaces are significant factor of efficiency of business.

4.2.1. An industrial microclimate

The most significant physical factor is the industrial microclimate, which is characterized by a level of temperature and air humidity, and also intensity of a radiation level.

The used computers do not require the creation of the special microclimatic conditions for operation and normally functions within the limits of values of temperature and humidity, allowed for the man. In sort that computers are sources of heat releases, there is a possibility of rising of temperature and descending of air humidity on work stations promoting a skin irritation. The microclimatic conditions in a room with a computer should meet the following requirements:

- Temperature of an environment in cold period of year 20 - 22°C, and 22 - 25°C in warm period;

- Relative humidity of air 30 - 60 %;
- The contents of a dust - max. 0.0001 kg/m at the dimension of particles max. 3 microns.

Another one of conditions of healthy and high-efficiency work is the provision of cleanness of air. Atmospheric air contains in its structure these in percentage terms:

- Nitrogen..... 78,8 %
- Oxygen 20,25 %
- Argon, neon and other inert gases 0,93 %
- Carbonic gas 0,03 %

Air of such structure is most favorable for breath of the man.

Network equipment and the workstations, considered in the given degree work do not produce any harmful substances during their operation. Thus, aerial environment in a room where they work does not render harmful effects on human organism and meets the requirements of first category of works.

The optimum norms of temperature, relative humidity and rate of movement of air in a working area of industrial rooms are normalized as given in the Table 4.1.

Table 4.1. Normalized parameters of microclimate in industrial rooms

Year season	Category of works	Temperature, °C	Relative humidity, %	Rate of air movement, m/s
Cold period	I	22 - 24	40 – 60	0,1
Warm period	I	23 - 25	40 – 60	0,1

4.2.2. Industrial lighting

The lighting is one of the major factors influencing to the productivity of work. The rationally arranged lighting on work stations of operators is an essential metric of high level labor culture, integral part of scientific organization of work

and aesthetics of production. The requirements to rational room lighting are reduced to the following:

- Correct choice of light sources and lighting systems;
- Creation of a necessary level of lighting of working surfaces;
- Limiting of blinding action of light, elimination of patches of reflected light;
- Provision of uniform lighting.

The acceptable level of lighting in a room can be found if we sequentially solve two tasks:

1. To determine a required level of lighting of operator's work station by external light sources.

2. If the required level of lighting appears unacceptable for other operators working in considered room, it is necessary to find a way of saving of required contrast of the representation by other means. For example, it is possible to arrange light flow taking into account the location of workstations and means of displaying of information.

At designing and organization of computer operator's workstation it is necessary to undertake actions on preventing the direct and reflected patches of light. Direct patches of light occur as a result of presence of light sources directly in sight of the operator, reflected patches of light appear as a result of presence of reflecting surfaces inside of field of view. The direct patches of light can be reduced by any of the following ways: to apply reflected lighting; to use several light sources of smaller power instead of one of high power; to use means of screening of direct light from eyes of the operator.

The reflected patches of light can be reduced by the following ways:

- to use diffused light;
- to apply matted surfaces;
- to allocate direct light sources so that a visual angle of working square by the operator do not concur with an angle of incidence of light from source.

The important task is the choosing the sort of lighting (natural or artificial) and choosing a working room according to it (with windows or without windows). Natural lighting is most favorable for the working personnel. The productivity of work at natural lighting is higher than at artificial one. For sufficient natural lighting the square of windows should take not less than 1/3 part of the total square of walls. However, it is necessary to take into account that the application of natural lighting has many disadvantages: as a rule arrival of light from only one side, the space non-uniformity of illumination, etc. For elimination of these disadvantages is necessary to apply extras. The application of double light (combination of natural and artificial lighting) physiologically non-effectively and negatively affects the sight, promotes early tiredness.

The application of artificial lighting helps to avoid many of the examined weaknesses and to establish an optimum lighting mode. However, the application of rooms without windows in a number of cases establishes feeling of constraint and uncertainty for working personnel. It appears especially strongly in small size rooms. In the large rooms the given disadvantage is practically absent, so it's preferable to use here the artificial lighting - filament lamps and luminescent lamps.

Accordingly to standards, the value of lighting by luminescent lamps should not be below 300 lx in a horizontal plane for aggregate system of lighting. In view of visual operation of high accuracy the value of illuminance can be increased up to 1000 lx. Apart from the illuminance, color of coloring of a room and spectral characteristics of used light renders the large influence on the operator's activity. It is recommended, that the ceiling should reflect 80-90 %, wall - 50-60 %, and floor - 15-30 % of light, falling on them. A room, where PC is located should be light and clean. The ceilings and walls are recommended to be colored tint. In rooms where the computer equipment is placed, the conditions fit to the given requirements must be created.

4.2.3. Technical measures of protection from electric shock

All technical measures can be conditionally divided into two groups. The technical protective measures of the first group provide protection of the personnel from electric shock in case of their touch to current-carrying parts. These include:

- The supervision over conditions of isolation of electro technical devices and sections of power supply network;
- Blocking and protective guards;
- Optimum arrangement of equipment, providing severance between current-carrying parts;
- Trouble signaling (light, acoustic), marking and preventive placards;
- Protection against junction of a high voltage to the side of a low voltage;
- Application of low (42 V and 12 V) voltages;
- Use of an individual protective insulating means.

The technical measures of the second group provide protection from electric shock at a touch to carcass of electro installation in case of breakdown of insulation of current-carrying parts, followings concern to them:

- Protective grounding;
- Protective zeroing;
- Protective cutoff (disconnection);
- Double isolation;
- Application of isolation transformers.

4.2.4. Electric isolation of current-carrying parts

It is known that the reliability and longevity of electro technical equipment in many respects depend on a state of electric isolation of current-carrying parts. The insulation failure is frequently a main reason of many electric traumas, crashes and fires. The physical meaning of isolation, as protective measure consists in limiting down of current, passing through a body of a man, to safe value. The reliable isolation depends on many factors and is provided with application of its certain type (operational, strengthened and double), appropriate insulating

materials, rational construction of an electric equipment, standard states of the industrial environment and, at last, by correct organization of preventive maintenance in the process of technical maintenance.

As a rule, the electro technical equipment has operational isolation, which should endure extreme mechanical, electric and thermal loads, which are possible under operational conditions.

The protective ground is a deliberate junction of noncurrent-carrying metal parts of electric equipment, lighting rods and dischargers with ground. The designation of protective grounding is to lower to safe value the voltage, which originates on noncurrent-carrying parts of electro installations in case of a fault to field at an insulation failure of conductors, carrying an operational current for the equipment.

Conclusion to chapter 4

Safety engineering is one of the most important parts of the final work. Because of utilization given project must be safety and security. In general every human should learn rules of occupation and safety engineering. Here is given how to use equipment and said that hand shouldn't be bare when working with electricity devices. Also was written how to get safety working with links and cable.

CONCLUSION

First of all, I want to say, that I chose this sphere, because I wanted to show connection information technologies with other fields of life with other sciences. This work includes knowledge from different subjects like mathematics, economy, programming, information technologies, system analysis and others.

Decision support systems are very interesting for me personally, because it does not provide one single solution of your task, what is more, it finds several ways to solve your problem, which depend on subscriber preferences and other criterion. In addition, they must be adaptive, be universal and be appropriate for almost all similar assignments.

Also, I must accentuate, that the main goal is correct data filling in Data Base. It is very significant, because all further computations and solutions depend on this primary information.

Over more, this application is appropriate not only for subscribers who are looking for area for café or restaurant, moreover, it is relevant for people, who want to find location for other similar establishments like clubs, internet cafes, cinemas because this application includes approximately all factors which may cause successful developing these establishments at the certain location in future.

In my work, I used not only mathematical expressions and criterion of famous scientists, what is more, I tried to create and use my own knowledge in this sphere and apply it in practice by means of user's preferences and optimal variant solutions.

List of the literature.

- <http://www.CrowInfoDesign.com/ebooks/decisionmatrix.pdf>
- http://ru.wikipedia.org/wiki/Microsoft_SQL_Server
- <http://ru.wikipedia.org/wiki/SQL>
- <http://dssresources.com/>
- http://ru.wikipedia.org/wiki/C_Sharp
- http://ru.wikipedia.org/wiki/Decision_Support_System
- http://en.wikipedia.org/wiki/Microsoft_SQL_Server
- [http://en.wikipedia.org/wiki/C_Sharp_\(programming_language\)](http://en.wikipedia.org/wiki/C_Sharp_(programming_language))
- Lectures conspectus of Decision Support System
- [http://en.wikipedia.org/wiki/Regret_\(decision_theory\)](http://en.wikipedia.org/wiki/Regret_(decision_theory))
- <http://www.doc.ic.ac.uk/~frk/frank/da/1.%20decision%20analysis%20intro.pdf>
- http://en.wikipedia.org/wiki/Wald's_maximin_model
- <http://businessmanagementcourses.org/Lesson35SavageMinimaxRegretCriterion.pdf>
- http://www.businessownersideacafe.com/starting_business/index.php
- http://www.ehow.com/how_2323724_start-small-cafe.html
- The law on information of Republic of Uzbekistan
- “Uzbekistan on a threshold of achievement of independence”. I.A.Karimov

Application code

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.Data.SqlClient;

namespace Venera
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();

            SqlConnection con = new SqlConnection(@"Data
Source=SABIR\SQLEXPRESS; Initial Catalog=1; Integrated
Security=true");
            SqlDataAdapter A_adap;
            DataTable A_table;

            SqlDataAdapter B_adap;
            DataTable B_table;

            private void button3_Click(object sender, EventArgs e)
            {
                Application.Exit();
            }

            public CheckBox[] c = new CheckBox[6];
            public TextBox[] t = new TextBox[6];

            private void Form1_Load_1(object sender, EventArgs e)
            {
                button1.Visible = false;
                button11.Visible = false;
                label7.Visible = false;
                textBox2.Visible = false;
                button6.Visible = false;
                textBox3.Visible = false;

                // button2.Enabled = false;
                button8.Enabled = false;
                button9.Enabled = false;
                button10.Enabled = false;
                button11.Enabled = false;
            }
        }
    }
}
```

```

int k = 0;
int k1 = 0;

richTextBox1.Visible = false;
richTextBox2.Visible = false;
richTextBox3.Visible = false;
richTextBox4.Visible = false;
textBox1.Visible = false;
label8.Visible = false;
label9.Visible = false;
label10.Visible = false;

for (int i = 0; i < 6; i++)
{
    c[i] = new CheckBox();
    c[i].SetBounds(237, 155 - 5 * i - k * 25, 51, 23);
    c[i].Tag = i + 1;
    c[i].BackColor = SystemColors.ButtonFace;
    this.Controls.Add(this.c[i]);
    k++;
}

for (int i = 0; i < 6; i++)
{
    t[i] = new TextBox();
    t[i].SetBounds(297, 155 - 5 * i - k1 * 25, 61, 23);
    t[i].Tag = i + 1;
    t[i].BackColor = SystemColors.ButtonFace;
    this.Controls.Add(this.t[i]);
    k1++;
    t[i].Visible = false;
}

con);

A_adap = new SqlDataAdapter("select * from [street]",
A_table = new DataTable();
A_adap.Fill(A_table);
dataGridView1.DataSource = A_table;
dataGridView1.Columns["id_street"].Visible = false;

for (int i = 9; i < 475; i++)
{
    dataGridView1.Rows[i].Visible = false;
}

// dataGridView1.Columns["risk"].Visible = false;
for(int i=11; i<14; i++)
{

```

```

        dataGridView1.Columns[i].Visible = false;
    }

}

private void button2_Click(object sender, EventArgs e)
{
    richTextBox1.Clear();
    richTextBox2.Clear();
    for (int i = 0; i < 9; i++)
    {
        dataGridView1.Rows[i].Selected = false;
    }

    /*double[] A = new double[100];
    for (int j = 0; j < 9; j++)
    {
        for (int i = 3; i < 7; i++)
        {
            A[j] += Convert.ToDouble(dataGridView1[i,
j].Value);

        }
        richTextBox1.Text += A[j].ToString() + "\n";
    }*/

    double[] A = new double[100];
    switch (comboBox1.Text)
    {

        case "АРЕНДА":
            for (int j = 0; j < 9; j++)
            {
                for (int i = 0; i < 6; i++)
                {
                    if (c[5].Checked == false)
                    {
                        if (c[i].Checked == true &&
c[5].Checked == false)
                            {
                                A[j] +=
Convert.ToDouble(dataGridView1[i + 6, j].Value);
                            }
                        }
                    }
                }

            }

            if (c[5].Checked == true)

```

```

        {
            for (int i = 0; i < 5; i++)
            {
                if (c[i].Checked == true)
                {
                    A[j] +=
Convert.ToDouble(dataGridView1[i + 6, j].Value);
                }
            }
            A[j] = A[j] /
Convert.ToDouble(dataGridView1[3, j].Value);
        }

        richTextBox1.Text += A[j].ToString() + "\n";
    }

    for (int i = 0; i < 9; i++)
    {
        dataGridView1[14, i].Value = A[i].ToString();
    }

    Array.Sort(A);
    Array.Reverse(A);

    richTextBox2.Text += A[0].ToString();

    for (int i = 0; i < 9; i++)
    {
        if (Convert.ToDouble(dataGridView1[14,
i].Value) == A[0])
        {
            dataGridView1.Rows[i].Selected = true;
        }
    }

    for (int i = 0; i < 9; i++)
    {
        if (Convert.ToDouble(dataGridView1[14,
i].Value) == Convert.ToDouble(richTextBox2.Text))
            MessageBox.Show(dataGridView1[1,
i].Value.ToString());
    }
    break;

    case "ПОКУПКА":
        for (int j = 0; j < 9; j++)

```

```

        {
            for (int i = 0; i < 6; i++)
            {
                if (c[5].Checked == false)
                {
                    if (c[i].Checked == true &&
c[5].Checked == false)
                        {
                            A[j] +=
Convert.ToDouble(dataGridView1[i + 6, j].Value);
                        }
                    }
                }
            }
            if (c[5].Checked == true)
            {
                for (int i = 0; i < 5; i++)
                {
                    if (c[i].Checked == true)
                    {
                        A[j] +=
Convert.ToDouble(dataGridView1[i + 6, j].Value);
                    }
                }
            }
            A[j] = A[j] /
Convert.ToDouble(dataGridView1[5, j].Value);
        }

        richTextBox1.Text += A[j].ToString() + "\n";
    }

    for (int i = 0; i < 9; i++)
    {
        dataGridView1[14, i].Value = A[i].ToString();
    }

    Array.Sort(A);
    Array.Reverse(A);

    richTextBox2.Text += A[0].ToString();

    for (int i = 0; i < 9; i++)
    {
        if (Convert.ToDouble(dataGridView1[14,
i].Value) == A[0])
        {
            dataGridView1.Rows[i].Selected = true;

```

```

        }
    }

    for (int i = 0; i < 9; i++)
    {
        if (Convert.ToDouble(dataGridView1[14,
i].Value) == Convert.ToDouble(richTextBox2.Text))
            MessageBox.Show(dataGridView1[1,
i].Value.ToString());
    }
    break;

    case "ПОСТРОЙКА":
        for (int j = 0; j < 9; j++)
        {
            for (int i = 0; i < 6; i++)
            {
                if (c[5].Checked == false)
                {
                    if (c[i].Checked == true &&
c[5].Checked == false)
                    {
                        A[j] +=
Convert.ToDouble(dataGridView1[i + 6, j].Value);
                    }
                }
            }

            if (c[5].Checked == true)
            {
                for (int i = 0; i < 5; i++)
                {
                    if (c[i].Checked == true)
                    {
                        A[j] +=
Convert.ToDouble(dataGridView1[i + 6, j].Value);
                    }
                }
                A[j] = A[j] /
Convert.ToDouble(dataGridView1[4, j].Value);
            }

            richTextBox1.Text += A[j].ToString() + "\n";
        }

        for (int i = 0; i < 9; i++)
        {
            dataGridView1[14, i].Value = A[i].ToString();

```

```

    }

    Array.Sort(A);
    Array.Reverse(A);

    richTextBox2.Text += A[0].ToString();

    for (int i = 0; i < 9; i++)
    {
        if (Convert.ToDouble(dataGridView1[14,
i].Value) == A[0])
        {
            dataGridView1.Rows[i].Selected = true;
        }
    }

    for (int i = 0; i < 9; i++)
    {
        if (Convert.ToDouble(dataGridView1[14,
i].Value) == Convert.ToDouble(richTextBox2.Text))
            MessageBox.Show(dataGridView1[1,
i].Value.ToString());
        break;
    }
}

```

```
public double[,] Z;
```

```
private void button7_Click(object sender, EventArgs e)
{
    richTextBox3.Clear();
    button8.Enabled = true;
    button9.Enabled = true;
    button10.Enabled = true;
    if (push == 1)
    {
        button11.Enabled = true;
    }
    else
    {
        button11.Enabled = false;
    }

    Z = new double[10, 10];

```

```

        for (int i = 0; i < 9; i++)
        {
            for (int j = 0; j < 3; j++)
            {
                Z[i, j] = Convert.ToDouble(dataGridView1[j + 3,
i].Value) *
                    Convert.ToDouble(dataGridView1[2, i].Value);
                //Z[i, 0] *= 12;
                richTextBox3.Text += Z[i, j].ToString() + "    ";
            }
            richTextBox3.Text += "\n";
        }
    }

private void button8_Click(object sender, EventArgs e)
{
    for (int i = 0; i < 9; i++)
    {
        dataGridView1.Rows[i].Selected = false;
    }
    richTextBox4.Clear();
    textBox1.Clear();
    double[] sr = new double[100];

    for (int i = 0; i < 9; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            sr[i] += Z[i, j];
        }
        sr[i] = sr[i] / 3;
        richTextBox4.Text += sr[i] + "\n";
    }
    double max = sr[0];
    for (int i = 0; i < 9; i++)
    {
        if (max < sr[i])
        {
            max = sr[i];
        }
    }
    textBox1.Text = max.ToString();
    for (int i = 0; i < 9; i++)
    {
        if (sr[i] == max)
        {
            dataGridView1.Rows[i].Selected = true;
            MessageBox.Show(dataGridView1[1,
i].Value.ToString());
        }
    }
}

```

```

}

private void button9_Click(object sender, EventArgs e)
{
    for (int i = 0; i < 9; i++)
    {
        dataGridView1.Rows[i].Selected = false;
    }
    richTextBox4.Clear();
    textBox1.Clear();
    double[] min = new double[100];

    for (int i = 0; i < 9; i++)
    {
        double m = Z[i, 0];
        for (int j = 0; j < 3; j++)
        {
            if (m > Z[i, j])
            {
                m = Z[i, j];
            }
        }

        min[i] = m;
        richTextBox4.Text += min[i] + "\n";
    }

    double max1 = min[0];
    for (int i = 0; i < 9; i++)
    {
        if (max1 < min[i])
        {
            max1 = min[i];
        }
    }

    textBox1.Text = max1.ToString();
    for (int i = 0; i < 9; i++)
    {
        if (min[i] == max1)
        {
            dataGridView1.Rows[i].Selected = true;
            MessageBox.Show(dataGridView1[1,
i].Value.ToString());
        }
    }
}

private void button10_Click(object sender, EventArgs e)
{
    for (int i = 0; i < 9; i++)
    {
        dataGridView1.Rows[i].Selected = false;
    }
}

```

```

}
richTextBox4.Clear();
textBox1.Clear();

double[] max = new double[10];

for (int i = 0; i < 3; i++)
{
    double m = Z[i, 0];
    for (int j = 0; j < 9; j++)
    {
        if (m < Z[j, i])
        {
            m = Z[j, i];
        }
    }
    max[i] = m;
    richTextBox4.Text += max[i] + "\n";
}
richTextBox4.Clear();

double[,] A = new double[10, 10];

for (int i = 0; i < 9; i++)
{
    for (int j = 0; j < 3; j++)
    {
        A[i, j] = max[j] - Z[i, j];
        richTextBox4.Text += A[i, j].ToString() + " ";
    }
    richTextBox4.Text += "\n";
}

richTextBox4.Clear();

double[] min = new double[100];

for (int i = 0; i < 9; i++)
{
    double m = A[i, 0];
    for (int j = 0; j < 3; j++)
    {
        if (m < A[i, j])
        {
            m = A[i, j];
        }
    }

    min[i] = m;
    richTextBox4.Text += min[i] + "\n";
}

double max1 = min[0];

```

```

        for (int i = 0; i < 9; i++)
        {
            if (max1 > min[i])
            {
                max1 = min[i];
            }
        }
        textBox1.Text = max1.ToString();
        for (int i = 0; i < 9; i++)
        {
            if (min[i] == max1)
            {
                dataGridView1.Rows[i].Selected = true;
                MessageBox.Show(dataGridView1[1,
i].Value.ToString());
            }
        }
    }

private void button12_Click(object sender, EventArgs e)
{
    richTextBox1.Visible = true;
    richTextBox2.Visible = true;
    richTextBox3.Visible = true;
    richTextBox4.Visible = true;
    textBox1.Visible = true;
    label8.Visible = true;
    label9.Visible = true;
    label10.Visible = true;
}

public int push;
private void textBox2_TextChanged(object sender, EventArgs e)
{
    push = 1;
}

public int check;
private void button14_Click(object sender, EventArgs e)
{
    /* check = 0;
    for (int i = 0; i < 6; i++)
    {
        if (c[i].Checked == true)
            check++;
    }
    */

    for (int i = 0; i < 6; i++)

```

```

    {
        t[i].Visible = false;
        if (c[i].Checked == true)
        {
            t[i].Visible = true;
        }
    }
}

```

```

private void button13_Click(object sender, EventArgs e)
{
    double[] opt = new double[100];
    double[] A = new double[10];
    double[] B = new double[10];
    double[] f_opt = new double[10];
    //double x = Convert.ToDouble(textBox3.Text);
    double max = 0;
    switch (comboBox1.Text)
    {
        case "АРЕНДА":
            for (int i = 0; i < 9; i++)
            {
                for (int j = 0; j < 5; j++)
                {
                    if (c[j].Checked == true && c[5].Checked
== false)
                        {
                            A[i] += Convert.ToDouble(t[j].Text) *
Convert.ToDouble(dataGridView1[j + 6, i].Value);
                        }
                    if (c[5].Checked == true)
                    {
                        if (c[j].Checked == true)
                            A[i] +=
Convert.ToDouble(t[j].Text) * Convert.ToDouble(dataGridView1[j + 6,
i].Value);
                        B[i] = Convert.ToDouble(t[5].Text) *
Convert.ToDouble(dataGridView1[3, i].Value);
                    }
                }
                opt[i] = (A[i] + B[i]) * (100 -
Convert.ToDouble(dataGridView1[2, i].Value));
                /*f_opt[i] = check * opt[i] *
Convert.ToDouble(dataGridView1[2, i].Value) +
                check * opt[i]; // *x;
                */
            }
        }
    }
}

```

```

        /* f_opt[i] = opt[i] *
Convert.ToDouble(dataGridView1[2, i].Value) +
        opt[i]; // *x;
        */
        // dataGridView1[15, i].Value =
f_opt[i].ToString();

        dataGridView1[15, i].Value =
opt[i].ToString();

        if (max < Convert.ToDouble(dataGridView1[15,
i].Value))
        {
            max = Convert.ToDouble(dataGridView1[15,
i].Value);
        }
    }
    for (int i = 0; i < 9; i++)
    {
        dataGridView1.Rows[i].Selected = false;
        if (Convert.ToDouble(dataGridView1[15,
i].Value) == max)
        {
            dataGridView1.Rows[i].Selected = true;
            MessageBox.Show(dataGridView1[1,
i].Value.ToString());
        }
    }
    break;

    case "ПОСТРОЙКА":

        for (int i = 0; i < 9; i++)
        {
            for (int j = 0; j < 5; j++)
            {
                if (c[j].Checked == true && c[5].Checked
== false)
                {
                    A[i] += Convert.ToDouble(t[j].Text) *
Convert.ToDouble(dataGridView1[j + 6, i].Value);
                }

                if (c[5].Checked == true)
                {
                    if (c[j].Checked == true)
                        A[i] +=
Convert.ToDouble(t[j].Text) * Convert.ToDouble(dataGridView1[j + 6,
i].Value);
                    B[i] = Convert.ToDouble(t[5].Text) *
Convert.ToDouble(dataGridView1[4, i].Value);
                }
            }
        }
    }

```

```

        }
        opt[i] = (A[i] + B[i]) * (100 -
Convert.ToDouble(dataGridView1[2, i].Value));

        /* f_opt[i] = check * opt[i] *
Convert.ToDouble(dataGridView1[2, i].Value) +
        check * opt[i] ;/** x;
        */
        /* f_opt[i] = opt[i] *
Convert.ToDouble(dataGridView1[2, i].Value) +
        opt[i];/** x;
        */
        // dataGridView1[15, i].Value =
f_opt[i].ToString();

        dataGridView1[15, i].Value = opt[i].ToString();

        if (max < Convert.ToDouble(dataGridView1[15,
i].Value))
        {
            max = Convert.ToDouble(dataGridView1[15,
i].Value);
        }
    }
    for (int i = 0; i < 9; i++)
    {
        dataGridView1.Rows[i].Selected = false;
        if (Convert.ToDouble(dataGridView1[15,
i].Value) == max)
        {
            dataGridView1.Rows[i].Selected = true;
            MessageBox.Show(dataGridView1[1,
i].Value.ToString());
        }
    }

    break;

    case "ПОКУПКА":
        for (int i = 0; i < 9; i++)
        {
            for (int j = 0; j < 5; j++)
            {
                if (c[j].Checked == true && c[5].Checked
== false)
                {
                    A[i] += Convert.ToDouble(t[j].Text) *
Convert.ToDouble(dataGridView1[j + 6, i].Value);
                }

                if (c[5].Checked == true)

```

```

        {
            if (c[j].Checked == true)
                A[i] +=
Convert.ToDouble(t[j].Text) * Convert.ToDouble(dataGridView1[j + 6,
i].Value);
                B[i] = Convert.ToDouble(t[5].Text) *
Convert.ToDouble(dataGridView1[5, i].Value);
        }
    }
    opt[i] = (A[i] + B[i]) * (100 -
Convert.ToDouble(dataGridView1[2, i].Value));

    /* f_opt[i] = check * opt[i] *
Convert.ToDouble(dataGridView1[2, i].Value) +
        check * opt[i] ;// * x;
    */
    /*f_opt[i] = opt[i] *
Convert.ToDouble(dataGridView1[2, i].Value) +
        opt[i];// * x;
    */
    //dataGridView1[15, i].Value =
f_opt[i].ToString();

    dataGridView1[15, i].Value =
opt[i].ToString();

    if (max < Convert.ToDouble(dataGridView1[15,
i].Value))
    {
        max = Convert.ToDouble(dataGridView1[15,
i].Value);
    }
}
for (int i = 0; i < 9; i++)
{
    dataGridView1.Rows[i].Selected = false;
    if (Convert.ToDouble(dataGridView1[15,
i].Value) == max)
    {
        dataGridView1.Rows[i].Selected = true;
        MessageBox.Show(dataGridView1[1,
i].Value.ToString());
    }
}
break;

}

}

private void dataGridView1_CellContentClick(object sender,
DataGridViewCellEventArgs e)
{

```

```
    }  
    private void groupBox1_Enter(object sender, EventArgs e)  
    {  
    }  
} }
```