Ministry of Education and Science of Ukraine
Ternopil Ivan Puluj National Technical University

Computer Information Systems and Software Engineering
(full name of faculty)

Computer Science
(full name of department)

QUALIFYING PAPER

For the degree of

Bachelor
(degree name)

topic: Creating a standard music website

Submitted by: fourth year student 4 , group ISN-42

specialty 122 «Computer Science»

(code and name of specialty)

	(signature)	Mbaezue Kyrian (surname and initials)
Supervisor	(signature)	Zolotyi R. (surname and initials)
Standards verified by	(signature)	Matsiuk O. (surname and initials)
Head of Department	(signature)	Bodnarchuk I. (surname and initials)
Reviewer	(signature)	Stadnyk N. (surname and initials)

Ternopil, 2022

Ministry of Education and Science of Ukraine
Ternopil Ivan Puluj National Technical University

Faculty Computer Information Systems and Software Engineering
(full name of faculty)

Department Computer Science
(full name of department)

APPROVED BY
Head of Department

(signature) (surname and initials)

« » 20___

ASSIGNMENT

for QUALIFYING PAPER

for the degree of Bachelor
(degree name)

specialty 122 «Computer Science»
(code and name of the specialty)

student Mbaezue Kyrian
(surname, name, patronymic)

1. Paper topic Creating a standard music website

Paper supervisor Ph. D., Assoc. Prof., department Computer Sience, Roman Zolotyi

(surname, name, patronymic, scientific degree, academic rank)

Approved by university order as of «*17* » *12* 2021 № *4/7-1068*

2. Student's paper submission deadline 23.06.2022

3. Initial data for the paper Technical requiremen for patient card

4. Paper contents (list of issues to be developed)
Introduction,
CHAPTER 1
CHAPTER 2
CHAPTER 3
Conclusions

5. List of graphic material (with exact number of required drawings, slides)

6. Advisors of paper chapters

Chapter	Advisor’s surname, initials and position	Signature, date	
		assignment was given by	assignment was received by
Life safety, basics of labor protection	Ph.D. (Engineering), Assoc. Prof. department Valeriy Lazaryuk		

7. Date of receiving the assignment ______________________________

TIME SCHEDULE

LN	Paper stages	Paper stages deadlines	Notes
1	Acquaintance with the assignment for the qualification work	24.01.2022	done
2	Analysis of literary sources	04.01.2022-30.01.2022	done
3	Justification of the relevance of the research	31.01.2022-06.02.2022	done
4	Analysis of the research subject and subject area	07.02.2022-13.02.2022	done
5	Design section 1	14.02.2022-06.03.2022	done
6	Design section 2	07.03.2022-03.04.2022	done
7	Completion of the task for the unit "Life safety"	04.04.2022-17.04.2022	done
8	Completion of the task for the unit "Basics of labor protection"	18.04.2022-01.05.2022	done
9	Completion of qualification work	02.05.2022-15.05.2022	done
10	Standard control	16.05.2022-22.05.2022	done
11	Check for plagiarism	06.06.2022	done
12	Preliminary defense of qualifying work	07.06.2022	done
13	Protection of qualification work	6.07.2022	

Student ____________ Mbaezue Kyrian
(signature) (surname and initials)

Paper supervisor ____________ Roman Zolotyi
(signature) (surname and initials)

ABSTRACT

Creating a standard music website // Qualification work of the educational level "Bachelor" // Mbaezue Kyrian // Ternopil Ivan Puluj National Technical University, Faculty of Computer Information Systems and Software Engineering, Department of Computer Science, group ISN-42 // Ternopil, 2022 // P. , Fig – , Table – .

Keywords: INFORMATION SYSTEM, WEBSITE, DATABASE, APACHE.

As a result of this diploma project, the website of the Ukrainian theater was developed. During the simulation, a diagram of components was developed, which shows all the pages of the site and their interaction. The deployment diagram shows the hardware and software components that interact during the operation of this site. A logical and, on its basis, a physical database scheme for the MySQL DBMS was developed, which reflects the essence of the subject area "Ukrainian Theater".

An admin panel was developed especially for the site, which allows you to easily manage the news of the site. The main sections developed for the site are: contacts with feedback form, theater poster, news system, site gallery. The use of the PHP language for processing information and requests to the database, creating a site design and using the jQuery library for site animation is shown.

LIST OF CONVENTIONAL SYMBOLS OF ABBREVIATIONS AND TERMS

APACHE - a cross-platform web server

CGI - an interface standard used to organize the interaction of a web server program with an external program.

CSS - cascading style sheets.

HTML is a hypertext markup language.

HTTP is an application-level data transfer protocol (initially in the form of hypertext documents in HTML format, now it is used to transfer arbitrary data).

JavaScript - a scripting programming language for the client part of web applications.

jQuery - a library of graphical components for JavaScript.

MySQL - a lightweight DBMS for small web projects.

PC - personal computer.

PHP - a server-side programming language.

UML - a unified modeling language.

CONTENT

INTRODUCTION

In my work I want develop the information website of the Ukrainian Music and Drama Theater because its use is an urgent issue today, since millions of people use the Internet every day without leaving home.

The Internet brings together, the reaction to any event comes practically immediately, and the distances disappear.

From all of the above, the relevance of the issue is clearly understood. Within the framework of the diploma project commissioned by the Ukrainian Theater, the task was set to develop an information web-site of the "Ukrainian Music and Drama Theater".

The aim of the work is to develop a web-site in, which has differences from ordinary theaters, to consider dream remedies for the development of the theater site.

To achieve this goal, it is necessary to solve the following tasks:

1) selection of means of development;

2) the score of the programming language;

3) Selection of DBMS.

1 CHARACTERISTICS OF THE SUBJECT AREA

1.1 Description of sections and placement of information on the theater's website

To date, there is no benchmark for a theater site, but there is a standard set of information that must be posted [2]:

1) news;

2) repertoire. List of all repertoire performances, pages of each performance;

3) the performance page must contain a brief description of the performance, a list of actors and performers, media materials, excerpts from the press. Ideally - links to the pages of the director, actors, links to the pages of the press;

4) poster. Schedule of performances for the current and next month with links to the performance pages;

5) theater people (management, troupe, theater employees). Personal pages with text, media materials, links to the press;

6) contact information;

7) communication;

8) feedback form.

Let's consider in detail how to first think about the best option for creating a theater website, to make it as convenient and effective as possible.

When developing the concept of information content of the theater website

the following factors should be considered:

1) type and format of information presentation;

to determine the type and format of information presentation, it is necessary to decide whether the use of only plain text and graphics will be sufficient. It may be necessary to introduce videos, sound files, Flash presentations into the content of the site. Will the site consist of static Html pages, or will the user receive information from databases.

2) structuring of information;

it should be decided how information can be structured, what sections the site will consist of. Segmentation of information can be done according to various criteria:

- by the type of audience for which this or that information will be interesting or useful;
- according to the time of receipt of information;
- according to the degree of importance of information;
- according to the structure adopted within the organization itself;
- by the type of information itself.

It is necessary to decide on the total amount of information provided on the site and ways to expand the content in the future. Will the responsible employees be able to prepare all the information themselves, or will it be necessary to involve outside specialists. How long will the process of preparing materials take? Will external sources of information be used, and if so, under what conditions.

The choice of navigation tools depends on the volume and segmentation of information on the site. The most important requirement for a navigation system is that it should be intuitive for users with any level of training. It should not only allow the user to quickly find exactly what he needs, but also immediately give an idea of what the site consists of and what other information can be found on it.

There is a whole set of navigation tools. It is considered a good tone to have several different navigation tools duplicating each other on the site at once, each of which will be more convenient both for a certain range of users and for different types of searches on the server.

On the Internet, as in no other mass media, it is important to capture the attention of users as quickly as possible, because at the distance of one click of the mouse there are other sites of a similar subject or the sites of competitors. Already on the main page, the user should understand what the site is and what will be interesting and useful. Right here, on the main page, you should place news announcements, inform about any changes in the repertoire, expected events. Such information will be useful for repeated or regular visitors: if they are met with something new and interesting every time, this is a good incentive to return to the site. In addition, such mobility is an indicator of the active life of the theater and the dynamic development of the site.

Due to a lack of attention, the network often practices a pyramidal structure of information presentation: the main thing first, details below. The news feed also consists of a headline and a short announcement - if you are interested, you can click on the "detail" link and read the full text.

It is advisable to break the information into small portions. Very few users read very long files from the screen. At the same time, it is recommended to make "printable versions" of normal size, and not to break them into small parts.

The volume of pages is very important. You can't allow a visitor to be lost just because he couldn't wait for the site to finish loading. There is no need to abuse graphics and other elements that have a large volume. And the graphics used should be optimized as much as possible.

The perception of the site as a whole, as well as directly of the information posted on it, is significantly influenced by the design - the graphic embodiment of the Internet resource. The design should organically complement and strengthen the overall concept. It is important to have a theater logo and a corporate color scheme. Color is part of the image of the theater and evokes bright associations, so the user will always recognize the site by the color solution.

The design of musical theaters is usually designed to create an elegant, classical atmosphere, while experimental plays are often designed in an emphatically simple way to create an atmosphere of ease and spontaneity.

Speaking about design, one cannot fail to mention an important term - brand. The theater, its name, logo, corporate color, history are a brand that must be promoted, made visible and recognizable. There is a formula: "Perception is more important than reality, and the brand is more important than the product." Therefore, the design of the site should be organic for the proposed context.

A site rarely provides value to visitors if the information on it does not change. This is especially important for resources that seek to form a permanent audience.

News should be regularly published on the site, old information should be kept up-to-date and corrected in a timely manner, and archives should be created. New sections should appear, existing ones should be expanded. Therefore, it is necessary to immediately decide who will support the informational component of the site, and how the updates will take place. The choice of tools will depend on the nature, scope, regularity of the updates and the qualifications of the staff performing the updates.

On small resources, adjustments can be made at the level of text changes in the html code of given pages. This can be done by the web studio that created the site. But Internet projects with daily updates cannot do without a special publishing system that allows you to quickly change the content of given pages. At the same time, almost no knowledge of HTML is required from the personnel performing the update. All leading web developers have their own publishing system modules with varying levels of functionality. Sometimes their installation is included in the cost of creating a site, sometimes it is necessary to pay a separate fee.

Audience engagement tools are an integral part of a modern website. Gone are the days when a website was simply an online advertising brochure. Today, the site solves many more tasks and does it more efficiently.

Gathering important information online is much easier and faster. For this purpose, questionnaires, voting, and guest books are placed on the site.

Questionnaires for visitors - collection of demographic information about site visitors. Depending on the demographic profile of the site's permanent audience, its administration can make appropriate adjustments to the content, sections, and even positioning, thereby adapting to the interests of the audience.

Analysis of user behavior on the site - resource owners can monitor server traffic, the most popular routes on the site, entry and exit points of visitors, time spent on each of the pages.

This information is used to determine the site's performance and to optimize its structure and navigation.

Using the guest book, you can maintain communication with users, and the absence of the need to respond immediately allows you to prepare your response more carefully.

It is also necessary to indicate the schedule of theater box offices.

A theater in a city with a large flow of tourists is very useful to have a version of the site in a foreign language. Usually, the button to switch to a foreign version is located on the main page of the site in the upper corner. Most often, the site is translated into English. But, for example, the website of the Maly Theater is presented in Russian, English and French.

The method of information sponsorship can be a very effective way to declare yourself. Being mentioned on other sites can generate additional interest among Internet users.

Many cultural institutions are afraid of this kind of cooperation, believing that the mention of competitors may arouse interest in them from potential viewers, which is not desirable. But practice shows that the more often some information about the organization appears, its logo is remembered, the more actively interest in its activities increases. "This approach serves as a means of attracting a large part of the non-target audience, creates the company's popularity, popularity, stable image, that is, it opens up new market segments for it."

It is important to identify and promptly eliminate problems and errors in the functioning of web pecypca. Spelling errors in the text may appear due to a malfunction in the program. It is necessary to provide protection against viruses, as today the global network is overloaded with them.

One of the main methods of fighting viruses is, as in medicine, timely prevention. In order to determine the basic rules of computer hygiene, it is necessary to find out the ways of virus penetration into computers and computer networks. This is the Internet itself, e-mail, electronic conferences, local networks, pirated software, repair services. It is extremely important to take care of protection against viruses, to ensure the safety of the site for both administrators and its visitors. It must be remembered that if the user detects a virus on the site, he will not return to this site.

Visitors are no less worried about the possibility of intrusion into the process of interactive transactions - interception of credit card numbers. Many do not decide to post data about themselves, fearing misuse of this information.

Monitoring must be carried out prior to launch. It is important to assess how fully the site realizes the possibilities of using the Internet, and how effectively it solves the tasks set before it, whether the following components correspond to the plan:

1) information content (information sufficiency, quality of materials, adaptation of materials for www, effective use of various forms and formats of information presentation, support and accuracy of the presented information, adequacy of the site's information structure, arrangement of informational accents);

2) website functionality (modules for working with information, feedback);

3) usability – ergonomics and ease of use (efficiency of navigation, adequacy of site interfaces, site map, rules of good tone, usual form of fields and buttons, help);

4) design - visual embodiment (design complements and strengthens the information and functionality embedded in the site, design supports and fully reflects the company's brand, attractiveness of design solutions, adequate use of multimedia tools, uniqueness and memorability);

5) technical implementation;

6) marketing component (addresses, search engines, links to the site and citations, means of collecting information about site visitors, attendance and behavior line on the site, work with the site audience);

7) Providing feedback. From time to time, any organization needs to evaluate the attractiveness and usefulness of the site. You can find out the opinion of web design experts. But the most important and reliable source of information is feedback

from users, who can explain what they like and what they don't like, and make suggestions for its improvement.

1.2 Stages of website development

The task of the bachelor thesis is the development of the informational website of the Ukrainian Music and Drama Theater. The system development process includes the following stages:

1) selection of development software;

it is necessary to determine which software will be used to build the system. At the same time, it should be able to implement the functionality required by the customer (enterprise management) and be quite easy to use for the programmer.

2) interface development.

It is necessary to determine which page design will attract customers and ensure the formation of the company's image. At the same time, the interface should correspond to modern industry trends.

3) database development

It is necessary to determine the structure of information that will be stored, updated and used during the system's operation. Determine the number of tables, relationships, etc. To justify the choice of DBMS.

4) development of the administrative part of the site.

1.3 Comparison of existing site analogues

Criteria for comparing sites

For comparison, four sites of the relevant subject area of drama theater were chosen, such as:

1) Donetsk National Academic Ukrainian Music and Drama Theater;

2) Kharkiv State Academic Drama Theater;

3) Mariinsky Theater;

4) National Opera of Ukraine.

The general structure for all selected sites is presented in fig. 1.1.

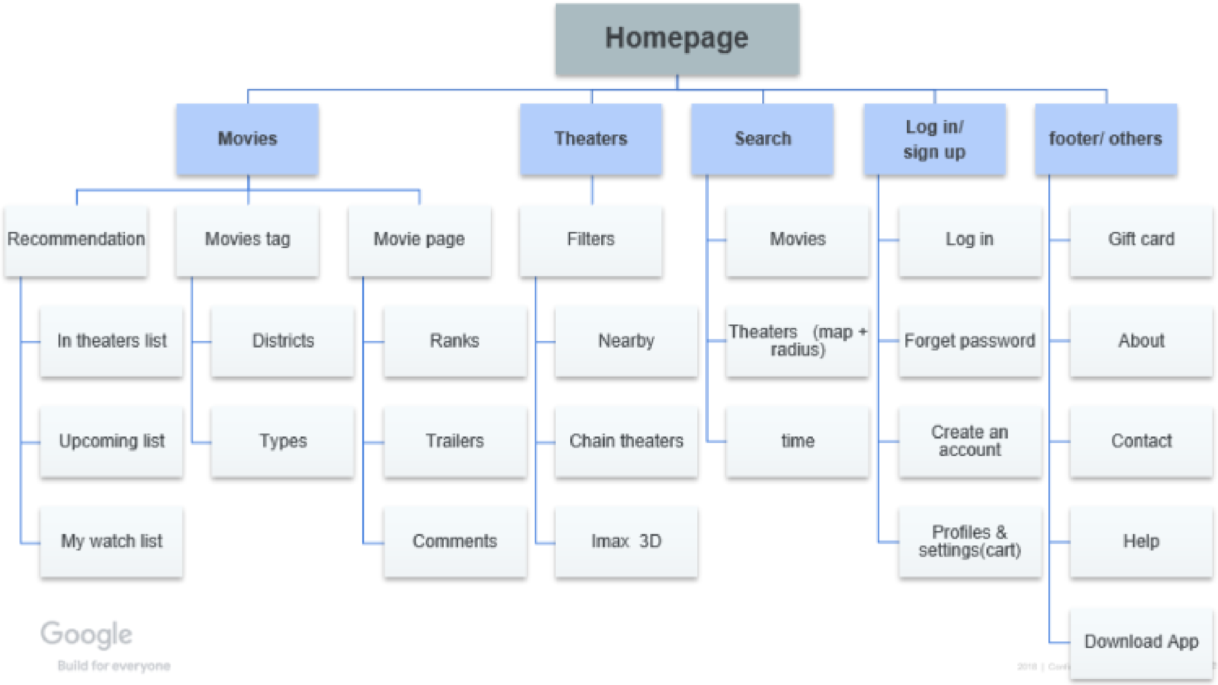


Figure 1.1 – Structure of the theater website

It is proposed to use the following comparison criteria [3]:

- visibility of the site by search engines;

- ease of use of the site;

- website design.

- Functionality

The main task of any site is to provide visitors with relevant information. For example, view information about the performance.

The visibility of the site by search engines stands out as the main criterion by which we can evaluate the site. If the site is not found by search engines (and most of the natural traffic is created by search engines), then they will not be visited, they will not be seen, and in fact it will not matter whether there is a site or not.

The main and mandatory conditions for ensuring the visibility of the site are its placement on the server and filling with the necessary content. Then the robot of the search engine will record this content in the database and will issue it according to the corresponding request.

Usability is the second most important evaluation criterion. It is important to find on the site the material that interests the visitor. There are sites that contain huge amounts of content, but are still simple and easy to use. And there are those where there are five sections, but you can't understand anything.

Site design is often brought to the fore. From the user's point of view, the design of the site is the main and often the only criterion for its quality. Obviously, appearance always plays an important role. Of course, the site should look beautiful and neat, but this work can be done already during the existence of the site, because the appearance does not affect the way search engines see the site. they can only read the texts, or rather, the HTML markup of the document. And the sooner they can do it, the sooner the site will start to pay off. As a rule, the design of the site requires a lot of time, a large part of which goes to the selection of colors, creation and editing of graphics. All this time, the site is "under development", its searches are not indexed, no one is involved in its content, and all employees are passionate about design, although they are not specialists in this.

Of course, you shouldn't put a completely raw site on the Internet either. The ideal option is to quickly create a competent and professional design (it will take a few days), then place the site on the Internet, and fill it with content that corresponds to the topic of the site as quickly as possible.

After that, you can safely start testing the site, adjusting its appearance (up to a complete redesign), without worrying about losing potential customers.

In some cases (for companies that do not want to show the site before its full design approval), a stub should be made, where all the company's keywords and contacts will be present. It is good if this page is designed in the style of the company, or at least has its logo.

The "functionality" of the site means the implementation of the content management system (content) of the site (CMS), as well as various services (if they are necessary, for example, an order basket, a form for searching for goods or services, user registration, management of votes, polls, publications and etc.).

Of course, the most important part is the site content management system (CMS), as it has to be used quite often. The CMS should be clear, understandable and convenient, since simple PC users who may not know anything about Web design and HTML will work with it.

In addition, a good CMS should correctly form the structure of the site so that the site is quickly indexed by search engines, in particular, provide the ability to edit meta tags.

Navigation on the site should be simple, clear and convenient, and the site itself should open quickly and behave predictably. For example, if some time-consuming process is taking place, such as loading images or searching, it will be correct to display a status bar or a preloader so that the user can see that the site is not hanging, but doing something.

1.4 Modeling of the subject field

Usually, websites are complex and dynamic structures. A short time is allocated for their development, so that the site is up and running as soon as possible and gives an effect. Often, developers immediately sit down to write code without even thinking about what they are going to create and how they are going to do it. Code for the server is often written from scratch, tables in databases are created as needed, and as a result, sometimes the system architecture begins to develop in a completely unexpected direction. However, with the help of simple modeling and a strict approach to programming, you can make the development process much smoother and ensure that the created web system will be easy to maintain in the future.

Development of a component diagram

The UML language [4] is a graphical modeling language designed for specification, visualization, design and documentation of all artifacts created during the development of software systems. The main purpose of UML is to provide, on the one hand, a fairly formal, and on the other hand, a fairly convenient and universal tool, which allows to reduce to some extent the risk of differences in the interpretation of specifications.

A component diagram [5] was chosen for site modeling because it generally reflects the parts of the developed system and the relationships between them. This makes it possible to design the architecture of the system and see all its main and auxiliary parts as a single structure.

The component diagram describes the features of the physical representation of the system. It allows you to determine the architecture of the developed system by establishing dependencies between software components, which can be the source and executable code. The main graphic elements of the component diagram are components, interfaces and dependencies between them.

The component diagram is developed for the following purposes:

1) visualization of the general structure of the source code of the software system;

2) specifications of executable versions of the software system;

3) provision of multiple use of individual program code fragments;

4) presentation of conceptual and physical schemes of databases.

The component diagram provides a coherent transition from a logical representation to a concrete implementation of the project in the form of software code. Some components can exist only at the stage of program code compilation, others at the stage of its execution. This diagram displays the general dependencies between components, treating the latter as classifiers.

To represent physical entities in the UML language, a special term is used - "component". The component implements a certain set of interfaces and serves for the general designation of elements of the physical representation of the model. A special symbol is used for the graphic representation of the component - a rectangle with two smaller rectangles inserted on the left. Inside the large rectangle, write the name of the component and, if necessary, some additional information. The image of this symbol may vary slightly depending on the nature of the information associated with the component.

The component name follows the general rules for naming model elements in the UML language and can consist of any number of letters, numbers and some punctuation marks.

An individual component can be represented at the type level or at the instance level. The graphic representation is the same in both cases, but the rules for writing the component name are different. If the component is represented at the type level, then only the type name with a capital letter is written as its name. If the component is presented at the instance level, then <component name> ':' <type name> is written as its name. In this case, the entire line of the name is underlined.

As simple names, it is customary to use the names of executable files (with the extension exe after the semicolon), dynamic libraries (DLL extension), web pages (HTML extension), text files (TXT extension or DOC) or help files, database files (DB) or files with source program texts (h extension, cpp for the C ++ language, .java extension for the Java language), scripts (pi, asp) and others.

Since the specific implementation of the logical representation of the system model depends on the used software tools, the names of the components are determined by the syntax features of the corresponding programming language.

In some cases, information about the name of the package and about the specific version of the implementation of this component can be added to the simple name of the component. In this case, the version number is written as a marked value in curly brackets. In other cases, the component symbol can be divided into sections to explicitly indicate the names of the interfaces implemented in it.

Since the component as an element of the physical implementation of the model represents a separate code module, it is sometimes commented with the indication of additional graphic symbols illustrating specific features of its implementation. These additional notations for annotations are not specified in the UML language, but their use simplifies the understanding of the component diagram, increasing the visibility of the physical representation.

In UML, three types of components are distinguished:

1) deployments that ensure the system performs its functions directly. Such components can be shared libraries with the DLL extension, web pages in the hypertext markup language with the HTML extension, and help files with the HLP extension;

2) work products. As a rule, these are files with the source texts of programs, for example, with extensions h or srr for the C ++ language;

3) implementations, which are executable modules - files with the extension exe.

Fig. 2.2 shows a diagram of the components of the website, which allows you to see how the pages of the website interact with each other. Packages are used as entities that group pages of the same type. If modeling large systems or systems with many parts, there will be many different classifiers in the model. Management of all classes in this case will be a difficult task. Therefore, UML provides an organizational element called a package. Packages allow you to organize classifier modelers in model namespaces as if they were folders in a file system. Dividing the system into packages makes the system more understandable, especially if each package represents a certain part of the system. Packages are great for organizing the parts of a model, but it's important to remember that the schematic class should easily communicate information about the parts being modeled. In cases where packages have many components, it is better to use several specific diagrams, rather than just producing one large diagram.

There are two ways to draw packets on diagrams. There is no rule to determine which designations are used. The drawing begins with a large rectangle with a smaller rectangle (tab) above its upper left corner. But the modeler must decide how to display the pack membership:

1) if the modeler decides to show package users within a large rectangle, then all these members must be placed inside the rectangle;

2) if the modeler decides to show the users of the package outside the large rectangle, then all the members that will be shown on the diagram must be placed outside the rectangle. To show that the classifiers belong to a package, a line is drawn from each classifier to a circle that has a plus sign inside the circle attached to the package.

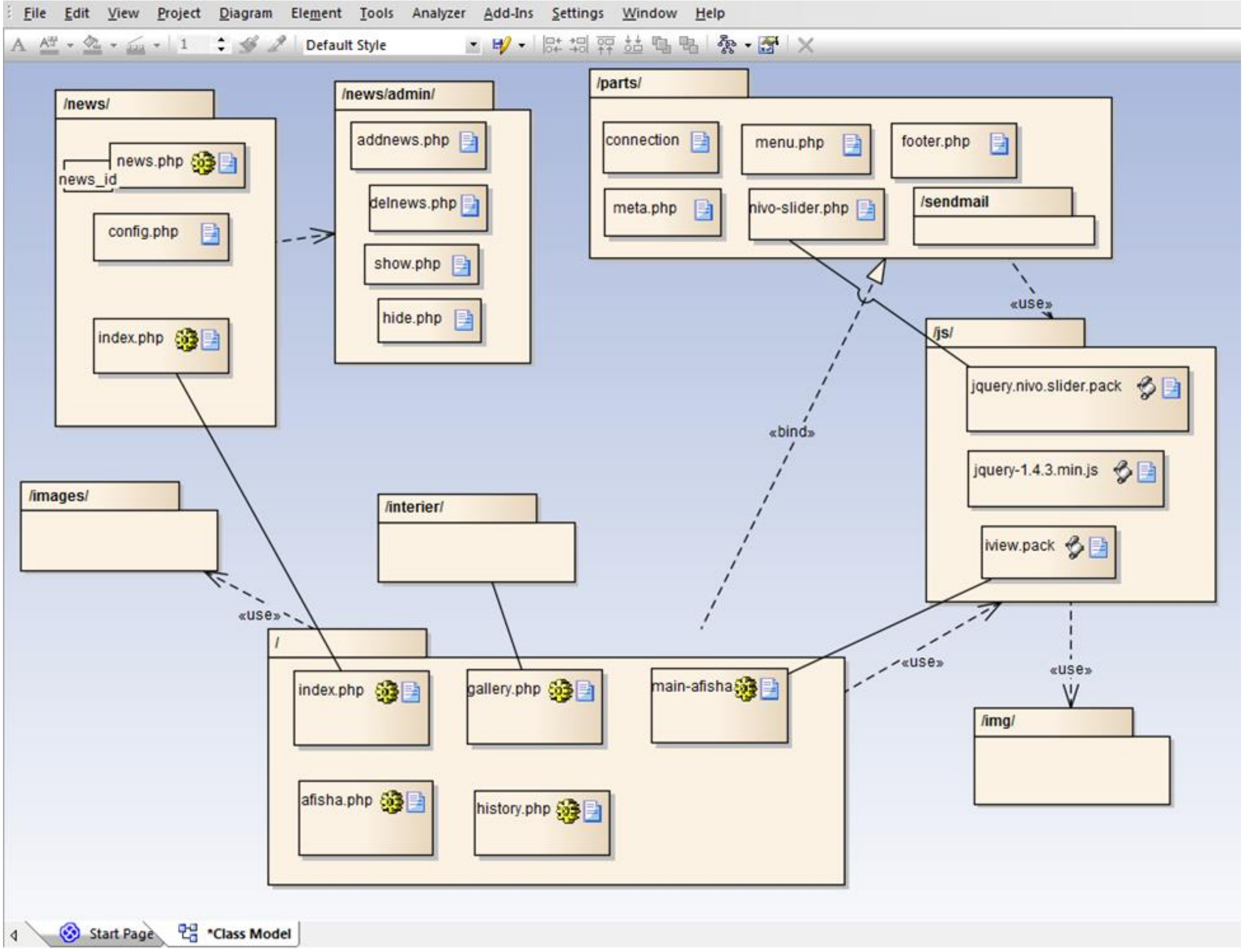


Figure 1.2 – Diagram of web application components

The following main entities were used in the diagram: packages, artifacts and relationships between them, such as association and dependency with the keywords “use” (use by one component of the components of another) and “bind” (connection between a template and a component) . An artifact is any artificially created element of a software system. Artifacts include web pages of the site (they are marked with the stereotype "webpage") and scripts of libraries (stereotype "library"), which are connected to each other by certain relationships. Packages were used as directories for diagrams. There is a main directory of the site (“root” or “/”), which is connected to all other directories. In the main directory there are pages: index.php, gallery.php, afisha.php, main–afisha.php, history.php. index.php is the main page that users see when they log in. On the main page, news is displayed on the right side in the content area, so index.php should be linked to index.php of the news directory.

The /news/ directory includes the configuration and database connection file config.php, as well as the php page responsible for displaying the current news and news on the main page. For greater flexibility of news management, an admin news management panel was created, which includes php-scripts for adding, deleting and changing news on the site through the admin interface.

All files from the root directory use images from the /images package, just as gallery.php uses the /interior package. Therefore, it makes sense to specify a relationship of the dependency type with the "use" parameter, which indicates use. Components from the /parts/ package are used for all pages of the root directory. Moreover, they all apply equally to all pages in which they were included. Therefore, to exclude repetition of code and the same actions, it is necessary to place them in the same package and specify the dependency with the keyword "bind", which will indicate that the used pages in the package are templates.

Components that require the use of sliders, windows with animation, use artifacts from the /js/ package. This package contains JavaScript library files that are suitable for effects and animations on pages. The nivo–slider.php template uses the jquery.nivo.slider.pack.js library, which is responsible for the image transition slider with various effects. The poster on the main site is displayed in a window view with various animation effects. The iview.pack script is used to reproduce the effects. All other packages that require user interface support on the site use the jquery-1.4.3.min.js library.

The site has a completely logical and detailed structure, shown in the figure above. It is expandable with the help of new components.

1.4 Development of a deployment diagram

Deployment diagram [5] is a diagram in UML that displays computing nodes during program operation, components, and objects executed on these nodes. Components correspond to the representation of working instances of units of code. Components that do not have a representation during the operation of the program are not displayed on such diagrams; but they can be displayed on component diagrams. A deployment diagram displays working instances of components, and a component diagram, in turn, displays the relationships between component types.

The developing website of the theater is launched on a computer that has a certain hardware configuration and works under the control of a certain operating system. Web-oriented programs often require some IT infrastructure for their work, store information in databases located on servers, call web services, use public resources, etc. In such cases, it would be good to have a graphic presentation attitude of the infrastructure on which the program will be deployed. That's what deployment charts, sometimes called placement charts, are for. I think it is obvious that it makes sense to build such diagrams only for hardware and software systems, while UML allows you to build models of any systems, not necessarily computer ones. First of all, a graphical representation of the IT infrastructure can help to more rationally distribute system components across network nodes, which, as is known, also depends on system performance. Secondly, such a diagram can help solve many auxiliary tasks related, for example, to ensuring security. The deployment diagram shows the system topology and the distribution of system components by its nodes, as well as the connections-routes of information transmission between hardware nodes. This is the only diagram on which "three-dimensional" notations are used: nodes of the system are marked with cubes. All other notations in UML are flat figures.

In the deployment diagram, there are two stereotypes for the "executionEnvironment" and "device" nodes. The node with the "executionEnvironment" stereotype allows you to model the hardware and software platform on which the program is executed. Examples of the execution environment are: operating system, database management system, etc. A node with the "device" stereotype also models a hardware-software platform, but allows the possibility of nesting one node into another. If the nodes are connected to each other by an association relation, then this means the same as in other contexts: the possibility of exchanging messages. In relation to computing networks consisting of nodes, association means the presence of a communication channel. If you need to specify additional information about the properties of the channel, then this can be done using general mechanisms: stereotypes, restrictions and named values.

Figure 2.3 shows a deployment diagram that shows the distribution of system components. A personal computer (PC) with the stereotype "device" acts as a hardware device, on which Chrome, Firefox or Opera browsers are installed, which are marked as components on the diagram. All of them send a GET request to the Apache HTTP server. After reception on the server, a PHP interpreter acts as a processor, which processes the code and makes appropriate requests to the database. After that, the server returns the response to the client and the corresponding representation is generated. This is the main task of Apache, which is handled by the so-called web server core. For each HTTP request, the content generator must be started. Some modules can register their own content generators, defining a handler function

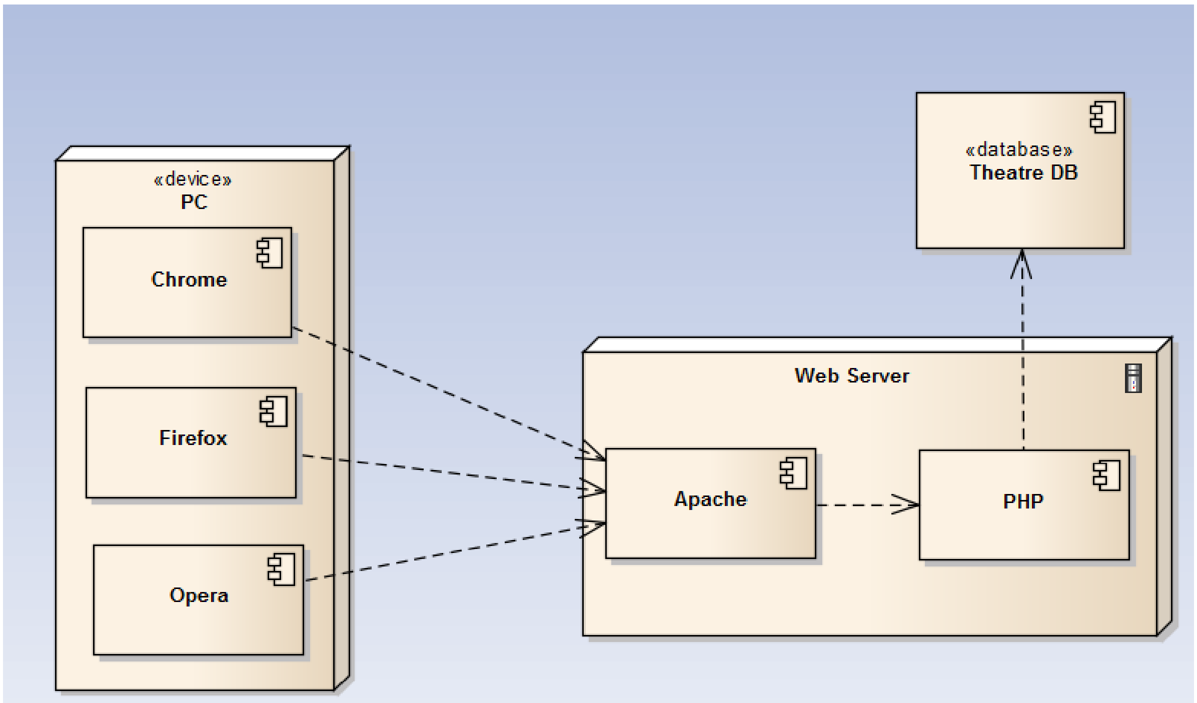


Figure 1.3 - Deployment diagram

1.5 Diagram of precedents (Use Case)

UML diagram of precedents [5] is a diagram that depicts the relationship between actors and precedents in the system. Also translates as use case diagram.

A precedent diagram is a graph consisting of a set of actors, precedents (use cases) bounded by a system boundary (directive), associations between actors and precedents, relations among precedents, and generalization relations between actors. Case diagrams represent the elements of the use case model.

The essence of this diagram is as follows: the designed system is presented in the form of many entities or actors interacting with the system with the help of so-called use cases. A use case is used to describe the services that the system provides to the actor. In other words, each use case defines some set of actions that the system performs during dialogue with the actor. At the same time, nothing is said about how the actors' interaction with the system will be implemented.

In the UML language, there are several standard types of relationships between actors and use cases:

1) associations (English association relationship);

2) inclusion (English include relationship);

3) extension (eng. extend relationship);

4) generalization (eng. generalization relationship);

At the same time, the general properties of the use cases can be represented in three different ways, namely, with the help of inclusion, extension and generalization relations.

The association relationship is one of the fundamental concepts in the UML language and is used to one degree or another in the construction of all graphical models of systems in the form of canonical diagrams.

Inclusion in the UML language is a type of dependency relationship between a basic use case and its special case. At the same time, dependency is a relationship between two elements of the model in which a change in one (independent) element leads to a change in another (dependent) element.

The extension relation determines the relationship of the basic usage option with another usage option, the functional behavior of which is not always used by the basic one, but only when additional conditions are present.

A rational unified process of developing a model of a complex system is its division into component parts with a minimum of mutual connections based on the allocation of packages. In the UML language itself, the "Usage Options" package is a subpackage of the "Elements of Behavior" package. The latter specifies the concepts used to determine the functionality of the simulated systems. The elements of the package of use cases are primary in relation to those that can be used to describe entities such as systems and subsystems. However, the internal structure of these entities is not described in any way. The basic elements of this package are the use case and the actor.

In fig. 1.4 shows a diagram of options for using website pages that provide the most important information for visitors - news and theater posters. When the "Visitor" actor requests a page, one or another page is returned to him, which is processed by rendering HTML pages, from which renderings for each page are inherited separately.

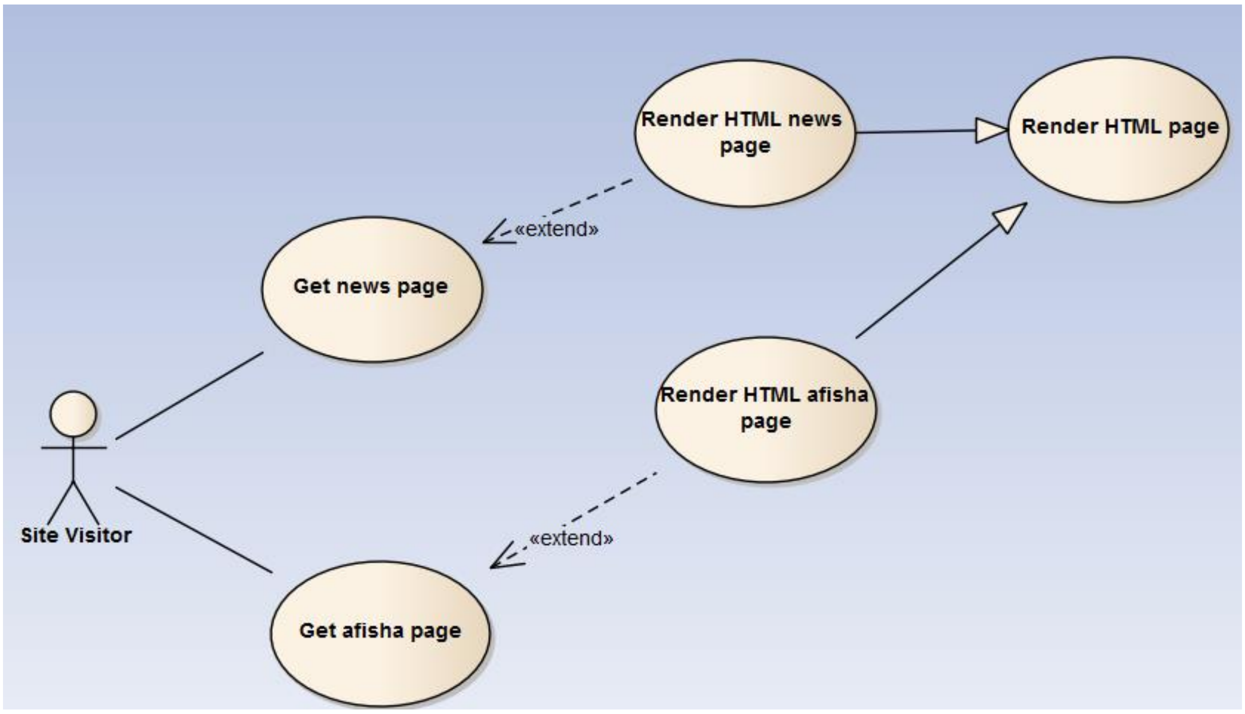


Figure 1.4 – Use case diagram

Since, in the general case, the actor is always outside the system, its internal structure is not defined in any way. For the actor, only his external presentation is important, that is, how he is perceived by the system. Actors interact with the system by sending and receiving messages from use cases. The message is a request by the actor for a service from the system and the receipt of this service. This interaction can be expressed with the help of associations between individual actors and options for use. In addition, actors can be associated with interfaces that define how other elements of the model interact with these actors.

2 SELECTION AND JUSTIFICATION OF THE SYSTEM

2.1 Architecture of the system

Great attention should be paid to the development of the interface when creating a site. An interface, in a broad sense, is a standard-defined boundary between interacting independent objects. The interface sets the parameters, procedures and characteristics of the interaction of objects.

User interface - elements and components of the program that can influence the user's interaction with the software. In particular:

– means of displaying information, displayed information, formats and codes;

– command modes, user-interface language;

- data entry devices and technologies;

– dialogues, interaction and transactions between the user and the computer;

– user feedback;

– support for decision-making in a specific field of vision;

- the procedure for using the program and its documentation.

The simpler, the better. This does not mean that there should be only text and contact information, but the site should not contain useless information, the font should be easy to read. Graphic elements should be clear, expressive and load quickly.

The human eye scans the pages of the site from top to bottom. Most attention is focused on the upper left side of the page. Therefore, the most important information is usually placed at the top of the site page: company name, logo, the site name itself, and so on. The simpler the upper part of the page is made, the easier it is to remember the name of the site and the company itself.

A typical mistake of many sites is the accumulation of an avant-garde and complex composition of many pictures in the upper part of the site page, among which the company name and the site name are barely noticeable. As a result, the human eye, scanning this part of the website page, does not have time to recognize and remember the necessary information. There is oversaturation. Instead of increasing memorability, on such sites, there is a decrease in memorability of the main inscriptions (pictures) of the site, the name of the company and the name of the site.

Ease of finding information. The user should easily find the data that interests him and be able to get comprehensive information about it (description in the form of text plus several photos).

Guided by these principles of interface development, it was decided to bet on simplicity and informativeness so that the user, visiting the site, could receive clear information about theater performances. You can also see the theater poster with the dates and times of the performance and so on.

So, we present the functioning structure of the theater as follows (Fig. 2.1):

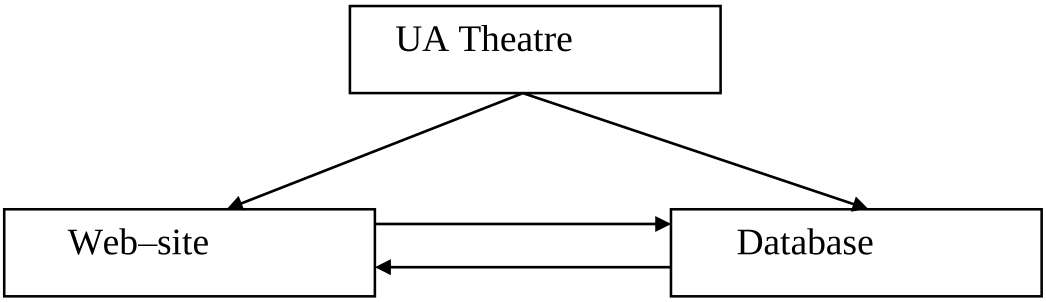


Figure 2.1 – Functional structure

2.2 Advantages of using PHP

PHP (Fig. 2.2) is an open source object-oriented programming language, a server scripting language embedded in HTML, which is interpreted and executed on the server [6].

Figure 2.2 – PHP language logo

PHP is an HTML preprocessor. His work is built according to the scheme shown in fig. 2.3.

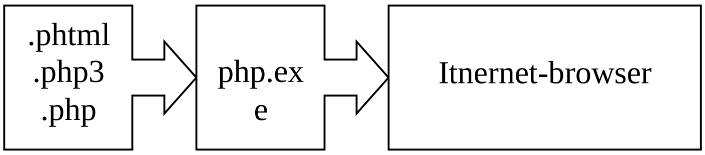


Figure 2.3 – Schematic of the PHP preprocessor

Before the server "gives" the file to the browser, it is viewed by a pre-processor - an interpreter. In order for this to happen, the files that are processed by the preprocessor must have a certain extension (usually it is . phtml or . php, but these values can be changed) and contain (although this is not a mandatory requirement) the code for preprocessor. Before sending the page, the PHP code is played on the server, and the browser receives the result in the form of HTML pages, which can be very different from the one stored on the server. Ordinary pages with extensions. html/htm the web server will send to the browser without any processing.

The main difference from CGI scripts written in other languages, such as Perl or C, is that in CGI programs you write the outputted HTML code yourself, and when using PHP, you embed your script program into the finished HTML page using one that opens and closes tags (for example, <? php ? >).

PHP is called a server-side scripting language, unlike JavaScript/Jscript/VBScript, which are client-side scripting languages. This means that the PHP script is executed on the server, and the result of its work is transferred to the client, while in JavaScript, the code is completely transferred to the client machine and is only executed there by the browser.

MS InternetInformation Server users will find that PHP is very similar to Active Server Pages (ASP), while Java users will say that PHP is similar to Java Server Pages (JSP). Some more analogues of PHP are ColdFusion and embPerl languages. All these languages allow you to place the code executed on the Web server inside HTML pages. There is no trace of PHP code in the resulting web page.

Everything you can do with CGI programs can be done with PHP. For example, processing data from forms, generating dynamic pages, receiving and sending cookies.

In addition, support for many databases is included in PHP, which makes writing web-applications using a database impossibly simple.

Here is a partial list of supported databases: Adabas D, InterBase, Solid, dBase, mSQL, Sybase, Empress, MYSQL, Velocis, FilePro, Oracle, Unixdbm, Informix, POSTGRESQL, ODBC.

In addition to everything, PHP understands IMAP, SNMP, NNTP, POP3 and even HTTP protocols, and also has the ability to work with sockets and communicate using other protocols.

Developers of web applications do not need to be told that a web page is not just text and images. A site worthy of attention should support a certain level of interactivity with the user: searching for information, selling products, conferences, and the like. Until recently, all this was traditionally implemented by CGI scripts written in Perl. But it turned out that CGI scripts scaled very poorly. Each new CGI script call requires the kernel to spawn a new process, which takes up processor time and consumes RAM. PHP offers another option - it works as part of a web server, and thus is similar to ASP from Microsoft or ColdFusion from Allaire.

PHP syntax is very similar to C or Perl syntax. Professionals who are familiar with programming will be able to start writing PHP programs very quickly. In this language, there is no strict data typing and there is no need for memory allocation/freeing operations.

Programs written in PHP are quite easy to read. Unlike Perl programs, PHP code is easy to understand visually.

In addition to being free (although MYSQL requires a license for commercial use), the PHP-MySQL connection is cross-platform. This means that you can develop Unix applications while working on Windows. In addition, PHP can work as an external CGI process, or as a regular script interpreter, or as a module that connects to the Apache or IIS web server.

And finally, since this product is developed by a joint effort, there is a huge amount of documentation and mailing lists that you can refer to in case of any questions.

2.3 Features of using the MYSQL database server

MySQL is a small compact multi-threaded database server. It (Fig. 2.4) is characterized by high speed, stability and lightness.

Figure 2.4 – MySQL database logo

MYSQL [7] was developed by TCX for internal needs, which consisted in the fast processing of very large databases. The company claims to have been using MYSQL since 1996 on a server with more than 40 databases containing 10,000 tables, of which more than 500 have more than 7 million rows. MYSQL is an ideal solution for small and medium-sized applications. Server outputs are compiled on many platforms. The most complete capabilities of the server are found on Unix servers, where there is support for multi-threading, which gives a significant increase in productivity. In the Windows version, MYSQL can be run as a Windows NT service or as a normal process on Windows 95/98.

MySQL Server is free for non-commercial use.

MYSQL supports the SQL query language in the ANSI 92 standard, and in addition has many extensions to this standard that are not found in any other DBMS.

A short list of MYSQL capabilities:

An unlimited number of users simultaneously working with the database is supported;

The number of rows in tables can reach 50 million;

Quick execution of commands. Maybe MYSQL is the fastest server out there;

A simple and effective security system.

MYSQL is indeed a very fast server, but to achieve this, the developers had to sacrifice some requirements for relational DBMS. MYSQL lacks:

- Transaction support is not implemented. Instead, it is suggested to use LOCK/UNLOCK TABLE;

- No support for triggers and stored procedures.

According to the creators, these points made it possible to achieve high performance. Their implementation significantly reduces the speed of the server. These capabilities are not critical when creating web applications, which, in combination with high speed and low price, allowed the server to gain great popularity.

One of the great features of MySQL, other than free, wide support and speed, is the choice of different storage engines for different tables.

A well-designed application that actively uses MySQL should use different storage engines for different tables.

Each engine stores data completely differently and is designed for specific application needs. MySQL offers 7 data storage engines:

1) MyISAM: Default engine. Does not support transactions, average data storage reliability. Excellent data read and write performance for intensive applications. Most web services and data stores heavily use MyISAM;

2) HEAP: everything in memory. Very fast data search, but all of them are stored only in memory and will be lost when the server stops. Great for temporary tables;

3) Archive: used to store large volumes of data without indexes, occupying a smaller size;

4) Merge: MyISAM collection of tables logically joined together for a single view;

5) InnoDB: Transactional type of engine, used for intensive write operations, thanks to the ability to block the level of table rows. Excellent recoverability and high reliability of data storage. The InnoDB engine was acquired by Oracle in 2005;

6) NDB: cluster engine. Data is automatically split and replicated across different machines called data nodes. It is used for applications that require high performance with the highest degree of availability. NDB works well on systems that require high throughput on read operations. From the point of view of an NDB cluster, a MySQL server instance is an API process connected to the cluster. NDB Cluster can simultaneously support access from other types of processes, including the Memcached API, JavaScript / Node.js.

A complex of programs is used to develop the system. Apache version 2.0 is selected as the web server, PHP 5.3.2 as the programming language, and MYSQL 4.0.21 as the database server.

2.4 Development of the database for the theater website

Definition of requirements

The requirements for the client database are as follows:

1) structuredness. The main requirement is that it should be formed according to a single principle: by organizations, employees, and industries. A multi-purpose client database should be broken down into topics or sections;

2) ease of use. It will be convenient to use the database if it is compiled in the form of a table. Visual perception of the table will make the work easier. The list of clients or partner organizations should be alphabetical. This will save time in searching for the right person or organization;

3) maximum completeness of information. The information contained in the database must necessarily include the following information elements: news, performances, posters, dates and times of performances, information about actors, composers, directors.

4) regular renewal. The database must be updated from time to time. Performance dates are renewed and new performances appear. It is necessary to ensure that the user receives the latest information from the theater's website, which exactly corresponds to the actual performances and dates. It is recommended to update the databases once every one and a half to two months, but it can be done more often if the need arises.

5) regular replenishment. Unlike updating databases, there are simply no specific regulations regarding this process. Depending on the circumstances, new entries can be made to the database almost every day, news appears on the website almost every day, new performances can be added, the theater's photo gallery can be replenished, and the theater's latest news and number of views can be monitored. It is only necessary to preserve its initial structure, or to develop a new one, if newly introduced additions require it.

Logical database model

A logical model [8] is a general view of data, a data model in a certain subject area. It is sometimes called a conceptual data model. Such a model is an entity-relationship type model. The DBDesigner online generator (Fig. 4.1) allows you to create a logical entity-relationship data model in the IDEF1x standard and, based on it, build a physical database for almost any DBMS. Moreover, it allows you to make an inverse transformation and build a logical one from a physical model. This is quite a valuable quality. Nowadays, databases are widely distributed and developed in various software environments. Using the specified property, you can transform existing databases developed in the environment of one DBMS into a logical model, and later make a reverse conversion into the environment of a completely different DBMS from those supported by DBDesigner. That is, existing databases can be recoded into a new software environment almost automatically, without using a programming language.

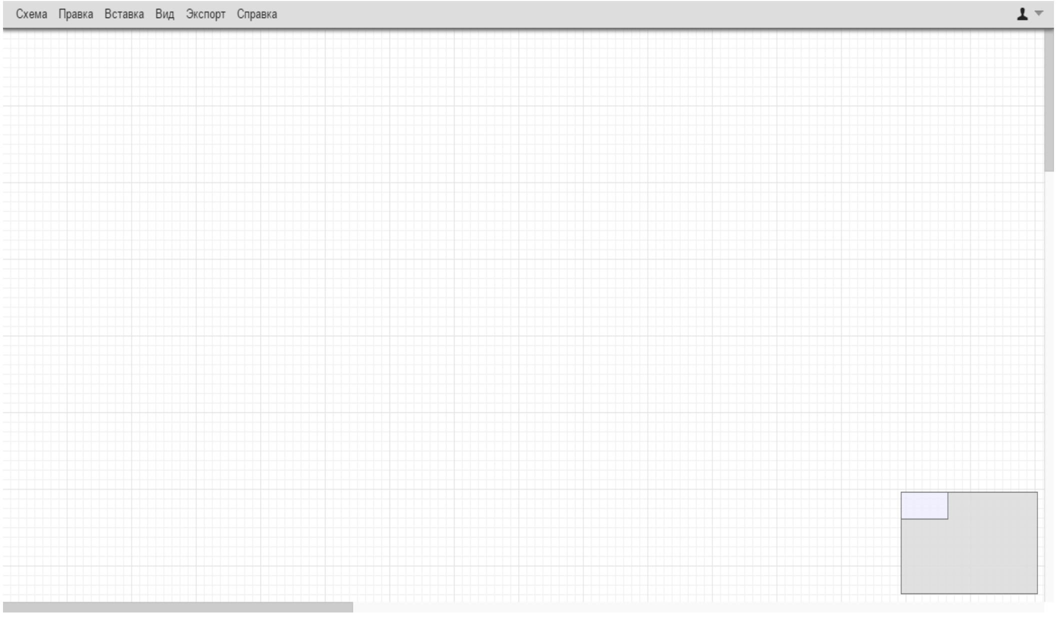


Figure 2.5 – DBDesigner program interface

In fig. 2.6 presents the logical model of the theater database. Tables page, news, contacts, about, gallery, afisha are used as entities. The structure of the two most important tables in the database is considered. The page table has the following fields:

- page_id – unique identifier of the page, which is of type varchar (variable-length string). The name of the page is used as an identifier;
- the title, keywords, description fields are used for the title, keywords, and description of the page, respectively. The title appears in browser tabs when you navigate to the page. Keywords are used for search engines. And the description serves for brief information about the subject and the main thing on the page;
- HTML code and content of all theater pages is stored in the text field of the page table. This is quite a convenient way to save. After all, there is no need to store many HTML pages as the site expands in the project directory. It is enough to have a corresponding field in the table.

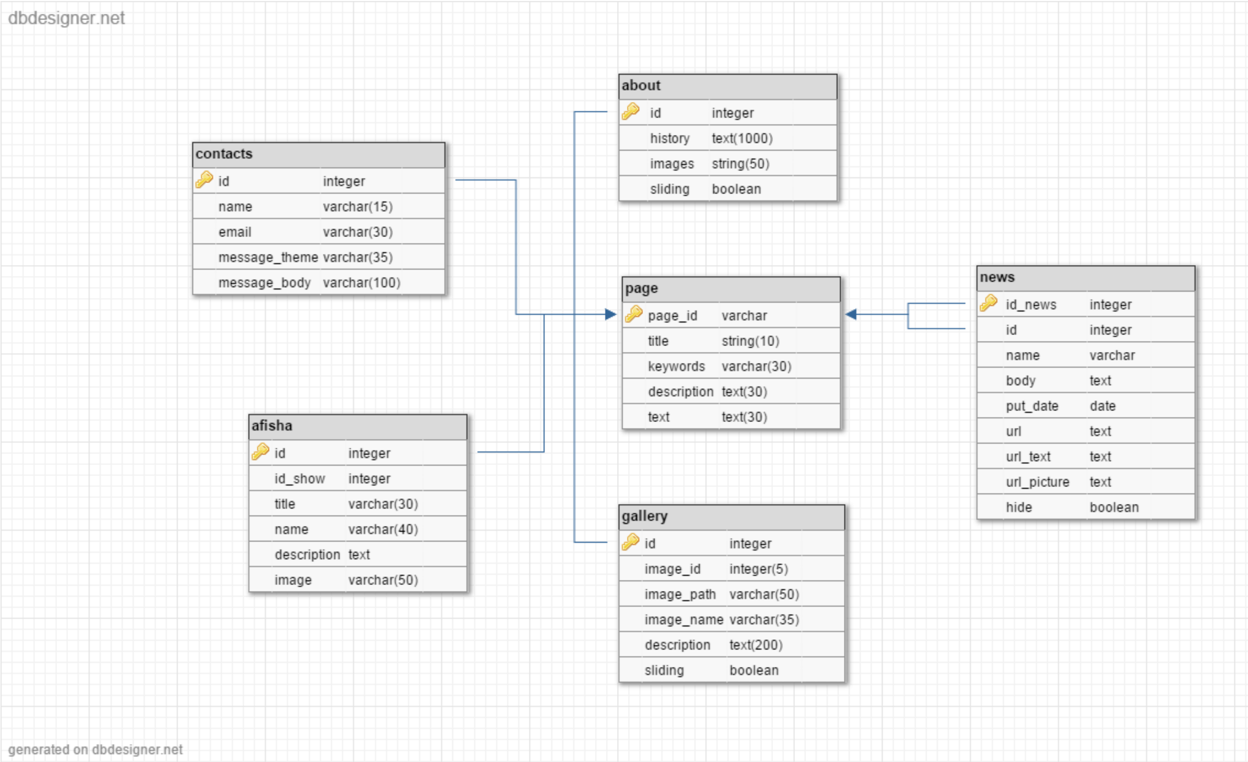


Figure 2.6 – Logical model of the theater database

Physical model of the database

The physical model [8] determines the placement of data in external memory. It is also called the internal model of the system and the form of its presentation depends on the selected DBMS. If a DBMS that supports the relational data model is selected, then the tables along with attributes and relationships between the tables must be transferred to the DBMS environment, taking into account the requirements for the corresponding DB objects. Thus, identifiers (names) of tables and fields must meet the requirements of the DBMS, data types, field sizes, restrictions must also be brought into line with those accepted in this DBMS. Next, indexing strategies are determined, as well as relationships between tables, primary and external keys based on those defined in the data model and taking into account the methods of their assignment in the selected DBMS. At the stage of physical design, special attention should be paid to ensuring the integrity of the database. In DBMS, data integrity is ensured by integrity constraints, i.e., a set of rules that establish the admissibility of data and relationships between them. Thus, the sequence of work when creating a physical model based on a relational data model can be as follows:

− analysis and selection of DBMS;

- development of the structure (physical model) of the database

by means of the selected DBMS, taking into account data types and attribute integrity restrictions;

− development of the database interconnection scheme by means of the selected DBMS, taking into account referential integrity restrictions;

- development of a control example of filling the database. The final result of the work is the presentation of the tables and relationships between them, described in the data model of the data, in the environment of the selected DBMS. At the same time, integrity constraints must be defined (programmed in the data description language), both for attributes and for referential integrity.

The MySQL DBMS was chosen to save information on the theater website. The database stores the content of the site (images or the path for downloading them, text in normal form or with appropriate HTML design tags), which will be loaded when clicking on the corresponding link in the menu or on the site page.

According to the logical scheme of the database, a physical model of the database is created, which will fully reflect the logical structure. The code for creating DB tables is given in the appendices.

After the tables were created, they were filled with the SQL command "INSERT INTO". The code for filling out the tables is given in the appendices.

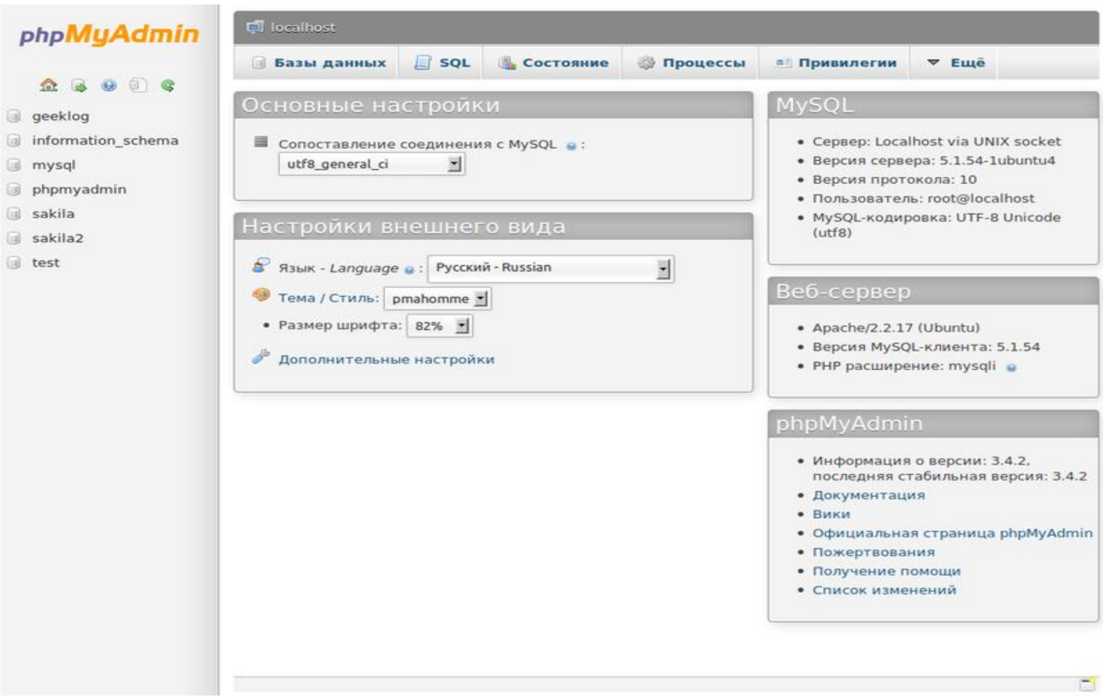
Creating a database using the phpMyAdmin environment

According to the logical model and the physical model created on its basis, the database of the theater is created using the MySQL DBMS in the phpMyAdmin environment [9].

phpMyAdmin (Fig. 5.3) is an open-source web application written in the PHP language and is a web interface for MySQL database administration. phpMyAdmin allows you to administer the MySQL server, run SQL commands and view the contents of tables and databases through a browser. The application is very popular among web developers.

Figure 2.7 – Logo of the phpMyAdmin environment

phpMyAdmin allows you to manage the MySQL DBMS without directly entering SQL commands, providing a friendly interface (Fig. 2.8).



Основные настройки

Сопоставление соединения с MySQL: utf8_general_ci

Настройки внешнего вида

Язык - *Language*: Русский - Russian

Тема / Стиль: pmahomme

- Размер шрифта: 82%

Дополнительные настройки

MySQL

- Сервер: Localhost via UNIX socket
- Версия сервера: 5.1.54-1ubuntu4
- Версия протокола: 10
- Пользователь: root@localhost
- MySQL-кодировка: UTF-8 Unicode (utf8)

Веб-сервер

- Apache/2.2.17 (Ubuntu)
- Версия MySQL-клиента: 5.1.54
- PHP расширение: mysqli

phpMyAdmin

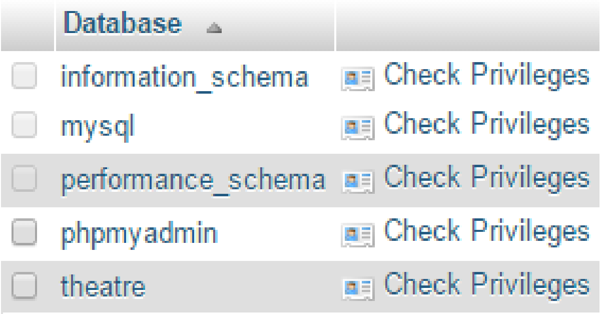
- Информация о версии: 3.4.2, последняя стабильная версия: 3.4.2
- Документация
- Вики
- Официальная страница phpMyAdmin
- Пожертвования
- Получение помощи
- Список изменений

Figure 2.8 – The main window of the phpMyAdmin environment

Today, PHPMyAdmin is widely used in practice. The latter is due to the fact that the developers are intensively developing their product, taking into account all the innovations of the MySQL DBMS. The vast majority of Russian providers use this application as a control panel in order to provide their customers with the ability to administer the databases allocated to them.

The application is distributed under the GNU General Public License and therefore many other developers integrate it into their developments, for example XAMPP, Denver, AppServ, Open Server.

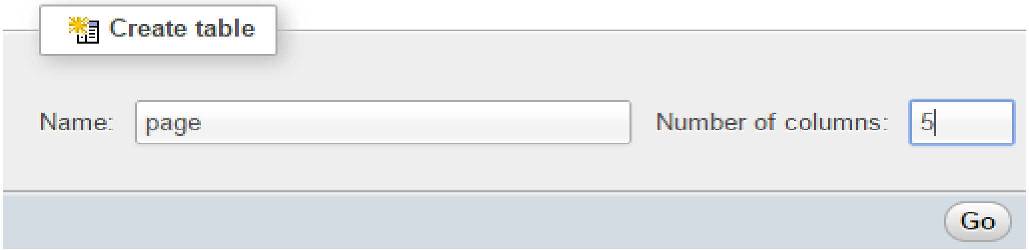
To create a new database, go to the tab from the top menu of the browser, enter the name and encoding type of the new database in the fields and click the "Create" button. After the database is created, it will be visible in the list of available databases (Fig. 2.9) on the local host (127.0.0.1).



	Database ▲	
□	information_schema	Check Privileges
□	mysql	Check Privileges
□	performance_schema	Check Privileges
□	phpmyadmin	Check Privileges
□	theatre	Check Privileges

Figure 2.9 – List of available databases

After creating the theater database, the page and news tables are created. To create tables, select the created database from the list of databases and click the button. After that, a window appears (Fig. 2.10), in which the name of the table and the number of columns are entered in the "name" and "Number of columns" fields, respectively, and the "Go" button is clicked.



Create table

Name: page

Number of columns: 5

Figure 2.10 – Table creation window

This is how tables are created. After pressing the "Go" button, a dialog box appears in the browser window (Fig. 5.7) in which you are asked to select parameters for the table columns. The parameters are:

- column name ("Name");

– data type ("Type") from available in MySQL;

- the number of digits or the length of the term (“Length/Values”);

- default value ("Default");

– column encoding (in MySQL, it is possible to specify a different type of “Collation” encoding for each column);

- the “Index” parameter allows you to set the field type (“Primary Key” – the primary key, “UNIQUE” – a unique value, “INDEX” – the possibility of having duplicate entries in the table and faster execution of SQL queries, “FULLTEXT” – indexes of are useful for full-text searches);

− Parameter A_I (AUTOINCREMENT) – automatically increases the value of the identifier by one when adding a new record to the table, if the field has an integer data type.

Other parameters are not important enough for these tables.

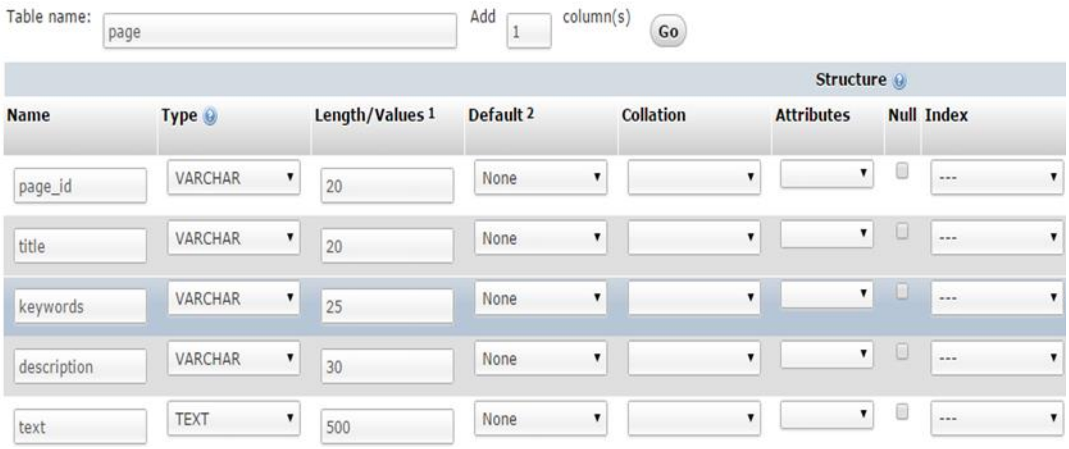


Table name: page Add 1 column(s) Go

Structure

Name	Type	Length/Values 1	Default 2	Collation	Attributes	Null	Index
page_id	VARCHAR	20	None			□	---
title	VARCHAR	20	None			□	---
keywords	VARCHAR	25	None			□	---
description	VARCHAR	30	None			□	---
text	TEXT	500	None			□	---

Figure 2.11 – Table column options selection window

After clicking the Go button, a new table is created in the database. Available tables in the database are in the list on the left (Fig. 2.12). Some basic tables are shown. Others are displayed in the same way. This section on the page is functional as it allows you to execute SQL queries to the created database by clicking on the icon, read documentation and information about working with phpMyAdmin by clicking on , update information about tables and database status by clicking on .

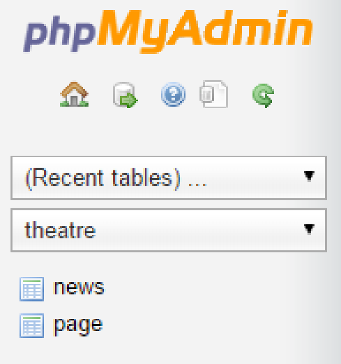
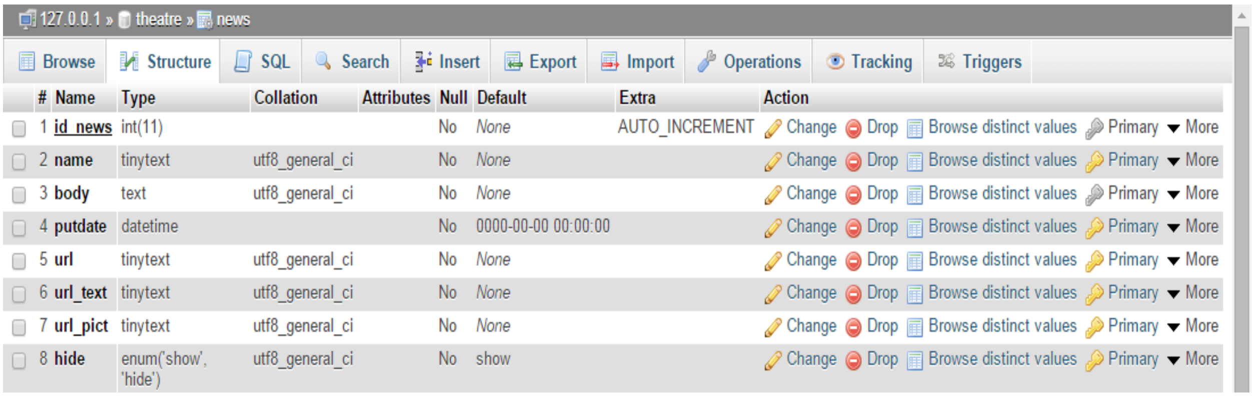


Figure 2.12 – List of database tables

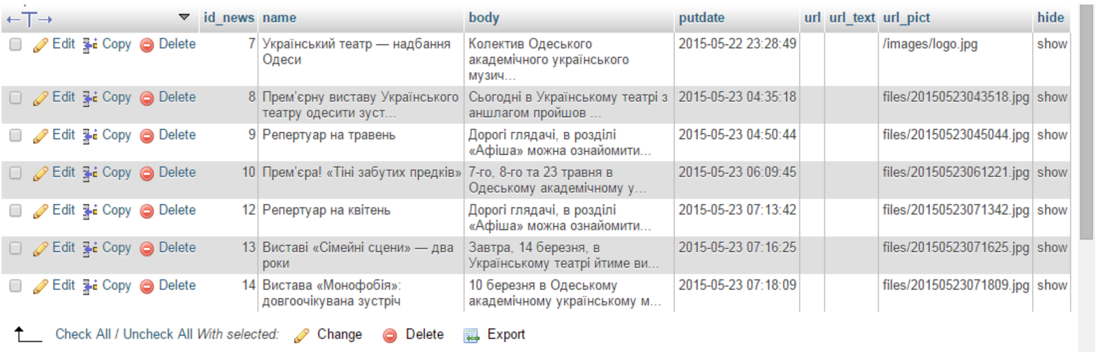
After selecting the appropriate table, it is possible to see the structure of the table by switching to the tabs (Fig. 2.13).



	#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
□	1	id_news	int(11)			No	None	AUTO_INCREMENT	Change Drop Browse distinct values Primary More
□	2	name	tinytext	utf8_general_ci		No	None		Change Drop Browse distinct values Primary More
□	3	body	text	utf8_general_ci		No	None		Change Drop Browse distinct values Primary More
□	4	putdate	datetime			No	0000-00-00 00:00:00		Change Drop Browse distinct values Primary More
□	5	url	tinytext	utf8_general_ci		No	None		Change Drop Browse distinct values Primary More
□	6	url_text	tinytext	utf8_general_ci		No	None		Change Drop Browse distinct values Primary More
□	7	url_pict	tinytext	utf8_general_ci		No	None		Change Drop Browse distinct values Primary More
□	8	hide	enum('show', 'hide')	utf8_general_ci		No	show		Change Drop Browse distinct values Primary More

Figure 2.13 – Table structure in phpMyAdmin

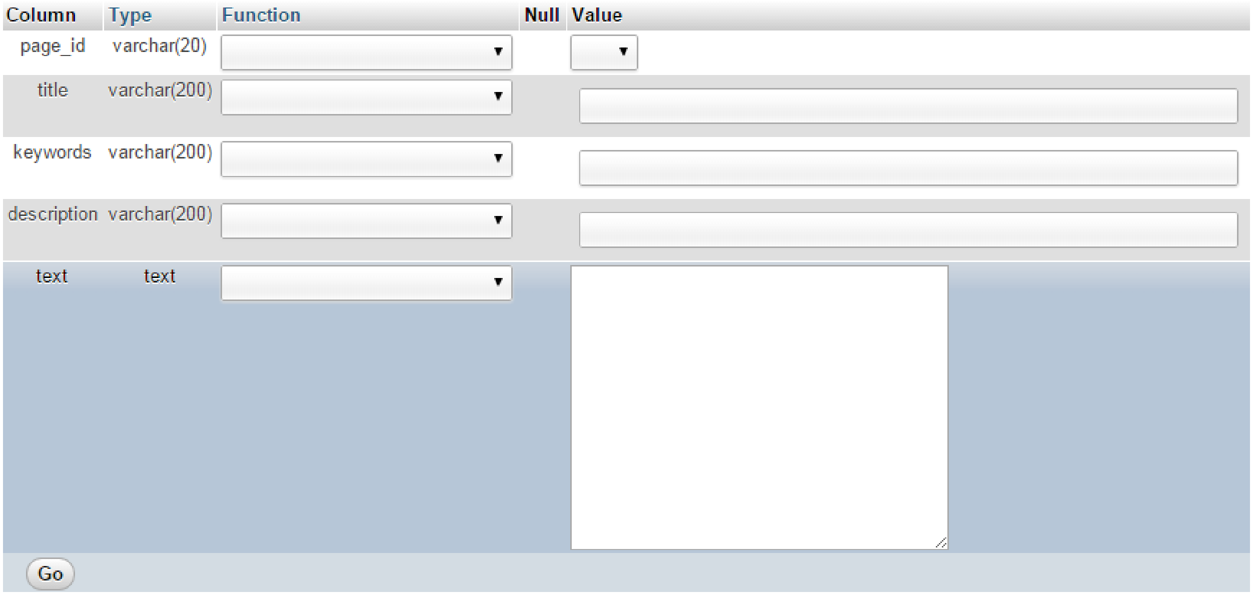
The tab allows you to manage entries in the table through the interface (change, add, delete them) if necessary (Fig. 2.14)



←T→	id_news	name	body	putdate	url	url_text	url_pict	hide
□ Edit Copy Delete	7	Український театр — надбання Одеси	Колектив Одеського академічного українського музич...	2015-05-22 23:28:49			/images/logo.jpg	show
□ Edit Copy Delete	8	Прем'єрну виставу Українського театру одесити зуст...	Сьогодні в Українському театрі з аншлагом пройшов ...	2015-05-23 04:35:18			files/20150523043518.jpg	show
□ Edit Copy Delete	9	Репертуар на травень	Дорогі глядачі, в розділі «Афіша» можна ознайомити...	2015-05-23 04:50:44			files/20150523045044.jpg	show
□ Edit Copy Delete	10	Прем'єра! «Тіні забутих предків»	7-го, 8-го та 23 травня в Одеському академічному у...	2015-05-23 06:09:45			files/20150523061221.jpg	show
□ Edit Copy Delete	12	Репертуар на квітень	Дорогі глядачі, в розділі «Афіша» можна ознайомити...	2015-05-23 07:13:42			files/20150523071342.jpg	show
□ Edit Copy Delete	13	Виставі «Сімейні сцени» — два роки	Завтра, 14 березня, в Українському театрі йтиме ви...	2015-05-23 07:16:25			files/20150523071625.jpg	show
□ Edit Copy Delete	14	Вистава «Монофобія»: довгоочікувана зустріч	10 березня в Одеському академічному українському м...	2015-05-23 07:18:09			files/20150523071809.jpg	show

Figure 2.14 – Table content management interface

Inserting a new entry in the table is performed by clicking on the tab on the menu bar of the environment. After that, a dialog box appears (Fig. 2.15), which allows you to select column values for inserting a new record.



Column	Type	Function	Null	Value
page_id	varchar(20)	▾		▾
title	varchar(200)	▾		
keywords	varchar(200)	▾		
description	varchar(200)	▾		
text	text	▾		

Figure 2.15 - Dialog box for inserting a new column

2.5 Theater website development

After creating and filling the database with information, the web application is created. Website creation is divided into several stages:

1) Creating the style and design of the site using HTML CSS binding.

2) Creation of graphical user windows using effects and animations of the jQuery library of the JavaScript language.

3) Creation of information processing code and database queries using PHP language and SQL query language.

2.6 Creation of site design

The design of the site begins with the creation of the structure of the site template (Fig. 2.16)

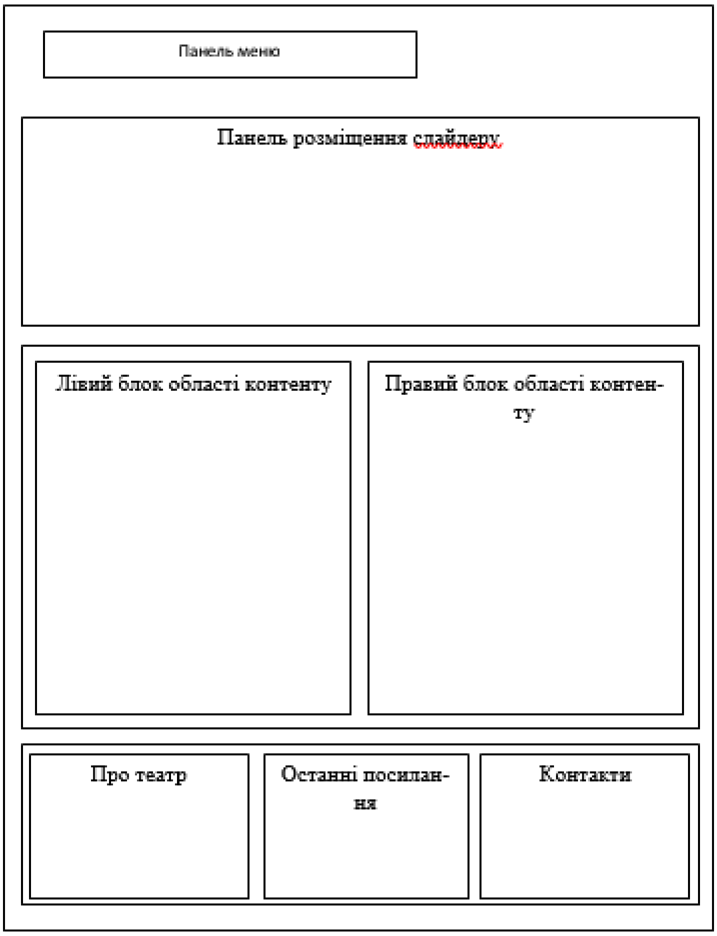

Панель меню
Панель розміщення слайдеру
Лівий блок області контенту
Правий блок області контенту
Про театр
Останні посилання
Контакти

Figure 2.16 – Site design template

After creating a site template, you can move on to creating all its components. A style file styles.css was developed for the template, which is connected to the pages of the site where it is needed. It contains global styles. The file code is given in the attachments. The main code snippets are explained for each component separately.

Creating a menu means writing the following styles for the menu:

In the menu, the user moves through the main sections of the site. After clicking on the corresponding menu button, a transition to one or another section of the site is performed. After applying the created styles, the site menu will look as shown in fig. 2.17.

```
#buttons{
      text-align:center;
      height: 36px;
      margin-top: 20px;
      padding-left: 0px;
      background: url(images/buttons.png) left top no-repeat;
      width: 612px;
      float: left;}
#buttons a {
      font-family: Arial, Helvetica, sans-serif;
      font-size: 14px;
      display: block;
      float: left;
      text-decoration: none;
      color: #ffffff;
      text-align: center;
      padding-top: 4px;
      height: 21px;}
.but {      width: 118px;}
#buttons .but:hover {  background:  url(images/but_hover.gif)  center  bottom
no-repeat;}
```

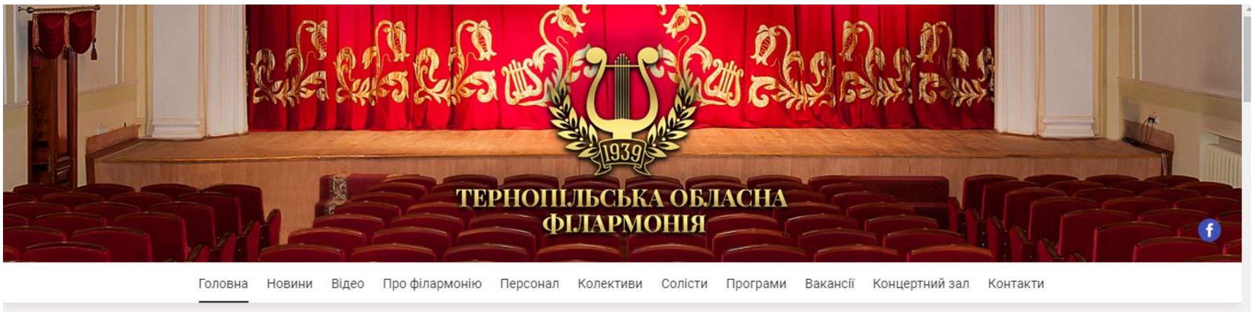


Figure 2.17 – Theater website menu

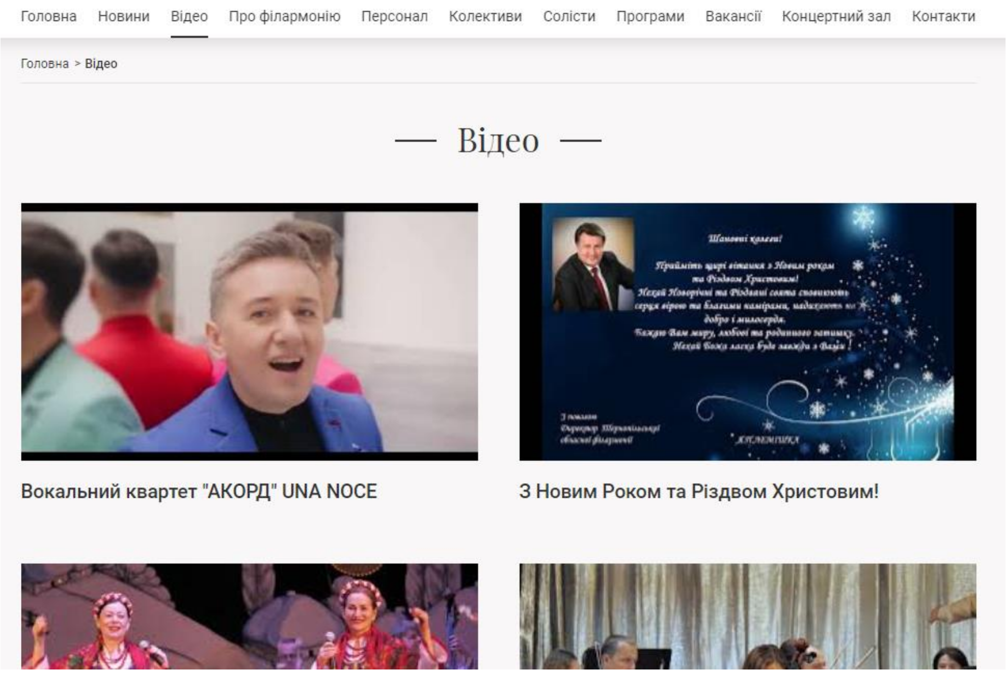
To place a slider with animation effects, it is necessary to determine its placement and on the page in pixels. The following code sets the area of the slider on the page:

```
#slider-wrapper {
    width: 998px;
    height: 332px;
    margin: 0px auto;
      padding-top: 0px;}
#slider {
      position:relative;
    width: 998px;
    height: 332px;
      background:url(images/loading.gif) no-repeat 50% 50%;}
#slider img {
      position:absolute;
      top:0px;
      left:0px;
      display:none;}
```

Figure 2.18 – Slider placement area on the page

The content area will contain the main content of the site. For ease of use, it is divided into blocks: left and right. The left block will contain brief information about the theater, a poster of the theater, which is executed in a jQuery window, and a link to more detailed information about the poster of performances and the history of the theater. The content area code is below..

After styling, the content area looks as follows (Fig. 2.19)

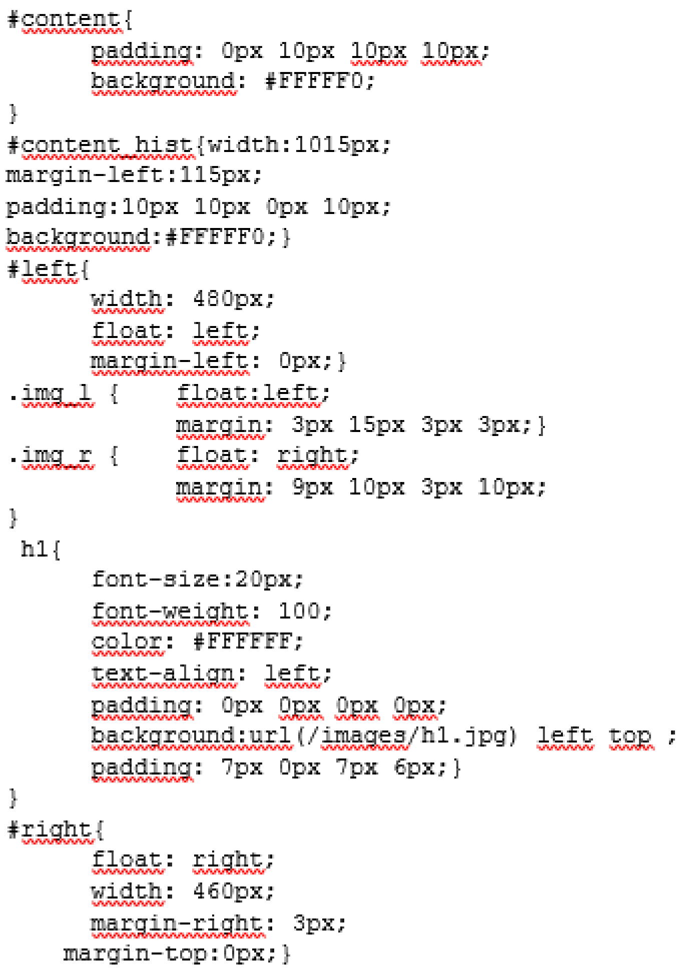


— Відео —

Вокальний квартет "АКОРД" UNA NOCE

З Новим Роком та Різдвом Христовим!

Figure 2.19 – Content area with applied styles



```
#content{
      padding: 0px 10px 10px 10px;
      background: #FFFFF0;
}
#content_hist{width:1015px;
margin-left:115px;
padding:10px 10px 0px 10px;
background:#FFFFF0;}
#left{
      width: 480px;
      float: left;
      margin-left: 0px;}
.img_l {    float:left;
            margin: 3px 15px 3px 3px;}
.img_r {    float: right;
            margin: 9px 10px 3px 10px;
}
 h1{
      font-size:20px;
      font-weight: 100;
      color: #FFFFFF;
      text-align: left;
      padding: 0px 0px 0px 0px;
      background:url(/images/h1.jpg) left top ;
      padding: 7px 0px 7px 6px;}
}
#right{
      float: right;
      width: 460px;
      margin-right: 3px;
    margin-top:0px;}
```

2.7 Creation of graphic windows based on jQuery

For the theater website, the Nivo-Slider [12] slider was adapted using the jQuery library and the iView plugin [13] was connected, which acts as a theater poster. The slider and plugin code is in the attachments.

Nivo-Slider was created specifically so that in the upper part of the site there was an opportunity to view a slideshow about the most famous theater performances. The slider automatically switches images using animation effects to transition from one image to another. It is possible to manually view the slideshow using .

Special CSS styles have been created for the slider (Fig. 6.8), JavaScript code has been written, which is responsible for animation, transitions, and a transition timer. The slider uses the jquery-1.4.3.min.js library and is connected to the site page with the jquery.nivo-slider.pack.js file. Connection code:

```
<script type="text/javascript" src="lib/jquery-1.4.3.min.js"> </script>
<script type="text/javascript"src="lib/jquery.nivo.slider.pack.js">
</script>
    <script type="text/javascript">

    $(window).load(function() {
        $('#slider').nivoSlider();
    });
    </script>
```

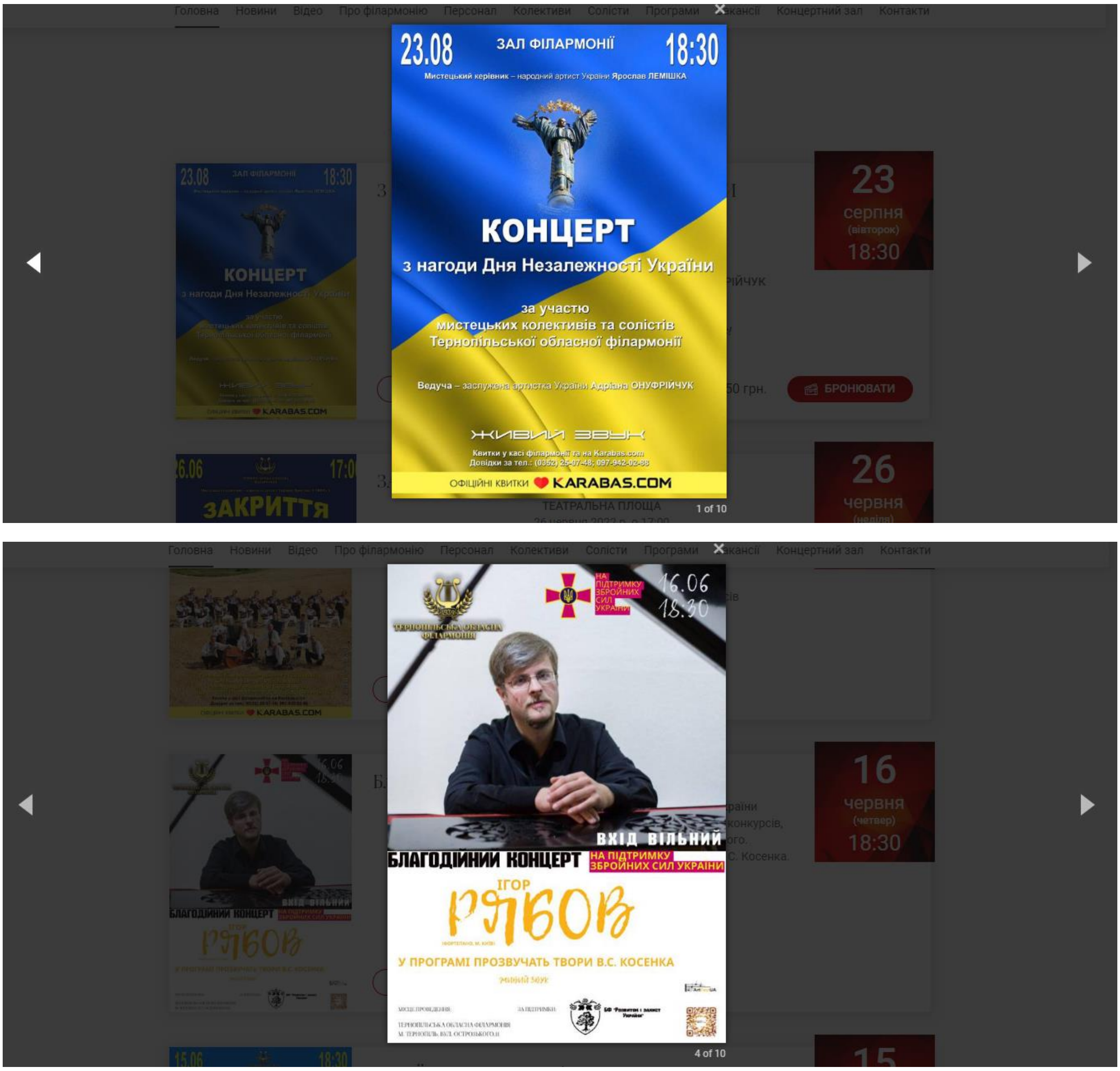


Figure 2.20 – Slider demonstration

For the convenience of viewing the theater poster, the IView plugin is used (Fig. 2.21). The plugin has a lot of animation effects and makes the poster viewing process more interesting. The plugin uses the same version of the jquery library as nivo-Slider, so there are no conflicts when using several jquery components on the same page. The file iview.pack, jquery.easing and raphael.min, which act as animation processing scripts, are connected to the site. Connecting and installing the plug-in for the corresponding style is performed as follows:

```
      <script type="text/javascript" src="/js/raphael-min.js"></script>
                 <script type="text/javascript"
src="/js/jquery.easing.js"></script>

      <script src="/js/iview.js"></script>
      <script>
      $(document).ready(function(){
      $('#iview').iView({
      pauseTime: 7000,
      directionNav: false,
      controlNav: true,
      tooltipY: -15});
      });</script>
      var iView = function (el, options) {
      //Get slider holder
      var iv = this;

      iv.options = options;

      iv.sliderContent = el, iv.sliderInner = iv.sliderContent.html();

      iv.sliderContent.html("<div class='iviewSlider'>" + iv.sliderInner +
"</div>");

      //Get slider
      iv.slider = $('.iviewSlider', iv.sliderContent);
      iv.slider.css('position', 'relative');}
```

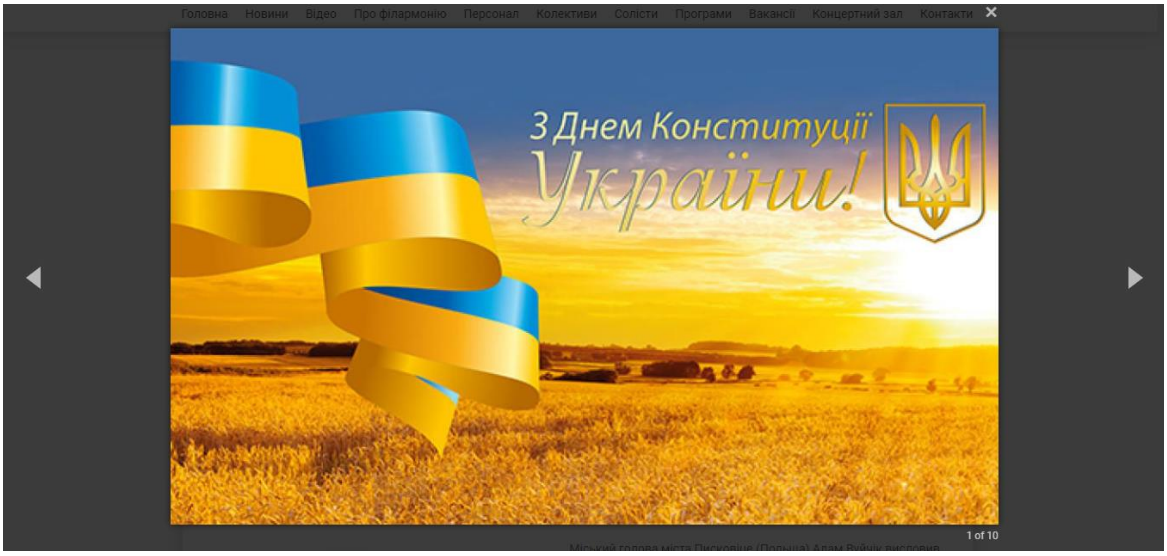


Figure 2.21 – Theater poster using the IView plugin

3 LIFE SAFETY, BASICS OF LABOR PROTECTION

3.1 Introduction

Occupational health or occupational safety, is a multidisciplinary field concerned with the safety, health, and well-being of workers. These terms also refer to the field's objectives.

Word related wellbeing bargains with all components of wellbeing and security within the work environment, with a solid center on essential avoidance of dangers," concurring to the World Health Organization (WHO). "A condition of add up to physical, mental, and social well-being, not as it were the nonattendance of affliction or inability," as characterized by the World Health Organization. Occupational wellbeing could be a multidisciplinary teach of medicine that centers on permitting individuals to do their occupations within the most beneficial conceivable way. It is in line with the advancement of word related wellbeing and security, which centers on anticipating harm from threats.

Given the appeal in the public eye for wellbeing and security arrangements at work dependent on solid data, word related security and wellbeing (OSH) experts should find their underlying foundations in proof-based practice. Another term is "proof informed independent direction". A functioning meaning of proof-based practice could be: proof-based practice is the utilization of proof from writing, and other proof-based sources, for exhortation and choices that favor the wellbeing, security, prosperity, and work capacity of laborers. Lastly, proof-based data should be coordinated with proficient ability and the laborers' esteems. Context oriented variables should be viewed as connected with regulation, culture, monetary, and specialized potential outcomes. Moral contemplations ought to be noticed.

Managers led their ventures how they figured fit to produce a benefit in the late nineteenth and mid-20th hundreds of years. Worker security and wellbeing were immaterial to them. In actuality, in conventional terms, absolutely no part of this made a difference. In the United States, injured specialists needed to go to court to get pay for their wounds. Workers were effectively deterred from going to court because of the significant expense of doing as such. Moreover, workers were seldom fruitful in light of the fact that, under customary law, assuming the representative knew about the perils of the gig or then again assuming that the wounds were brought about by the representative's or a colleague's carelessness, the business was not responsible.

Work environments under the locale are administered by your commonplace regulation. The regulation spots obligations on proprietors, bosses, laborers, providers, the independently employed and project workers, to build up and keep up with protected and solid working conditions. The regulation is managed by your commonplace regulation. Your authorities are liable for observing consistence.

3.2 Rights, duties and responsibilities of employers and workers during outbreak and emergencies

Maintaining an appropriate and functional workforce, as well as ensuring the continuity of emergency response and vital health services, requires protecting the health and safety of health-care employees and other emergency responders. Employers must be prepared to adapt their usual practice in consultation with workers, their representatives, and technical experts in an emergency situation such as an outbreak, chemical spill, or radiation release, where workplace risk changes rapidly, in order to achieve a reasonable balance of safety versus obligation to work.

Workers in the emergency response field, particularly health-care professionals, have a contractual obligation and a duty of care to deliver services that may expose them to infections, toxins, injuries, and diseases. Despite their duty of care, emergency response professionals may have the right, depending on the national context, scenario, and practice, to remove themselves from a work situation that they have sufficient grounds to believe poses an urgent and serious threat to their lives or health.

Employers of emergency responders have a responsibility to ensure safe working conditions as well as the resources needed to execute acceptable OSH procedures. Employers have a responsibility to:

– these personnel must be suitably trained, equipped, and protected;

– give them the ability and knowledge to execute OSH methodologies;

– give clear guidance on the working circumstances for these workers, what is expected of them, and the inherent risks of the scenario;

– give adequate psychological assistance, as well as put in place measures to promote healthy behaviors;

– offer fair compensation for the services supplied by these individuals in the form of risk premiums and insurance for them and their families, as well as disability benefits for those infected;

– collect information in a systematic manner to support ongoing monitoring and evaluation of the effectiveness of OSH programs in providing protection.

3.3 Strategies and tools for protecting occupational safety and health in emergencies and outbreaks

During scourges and crises, the administration frameworks approach gives a general structure to overseeing OSH hazards. Certain strategies, methodology, and instruments exist for the avoidance and control of OSH dangers and dangers inside this structure. These devices can be custom fitted to the flare-up or crisis circumstance within reach. The "stepping stool of controls," the ICS, and disease counteraction and control measures are totally shrouded in this segment.

The Incident Command System (ICS) is a standardized on-scene incident management approach that allows responders to adopt an integrated organizational structure that is equal to the complexity and demands of any single incident or numerous incidents, regardless of jurisdictional boundaries. By establishing a reasonable span of control, the ICS enables integrated communication and planning. An ICS splits a disaster into five controllable functions: command, operations, planning, logistics, and finance and administration.

Organization structure of the ICS. The minimum ICS should consist of the following and can be expanded according to requirements:

– The Public Information Officer, Safety Officer, and Liaison Officer make up the command staff. They are directly responsible to the Incident Commander.

– The organizational level with functional responsibility for the key segments of incident management (operations, planning, logistics, finance/administration) is represented by sections. The section level is located between the branch and the Incident Commander in terms of organizational structure.

– Each sector is made up of smaller and smaller organizational units, such as a branch, division, group, unit, task force, strike force, and finally a single resource. A single resource is a single piece of equipment and its staff complement, or a pre-established crew or team of people with a designated work supervisor that can be deployed to respond to an incident.

– The ICS is created by identifying the primary tasks or functions that must be performed in order to respond to incidents efficiently. The demand for an organizational manager has grown as occurrences have become more complicated, challenging, and costly. The Incident Commander handles the organization, not the incident, in the ICS, especially in larger incidents.

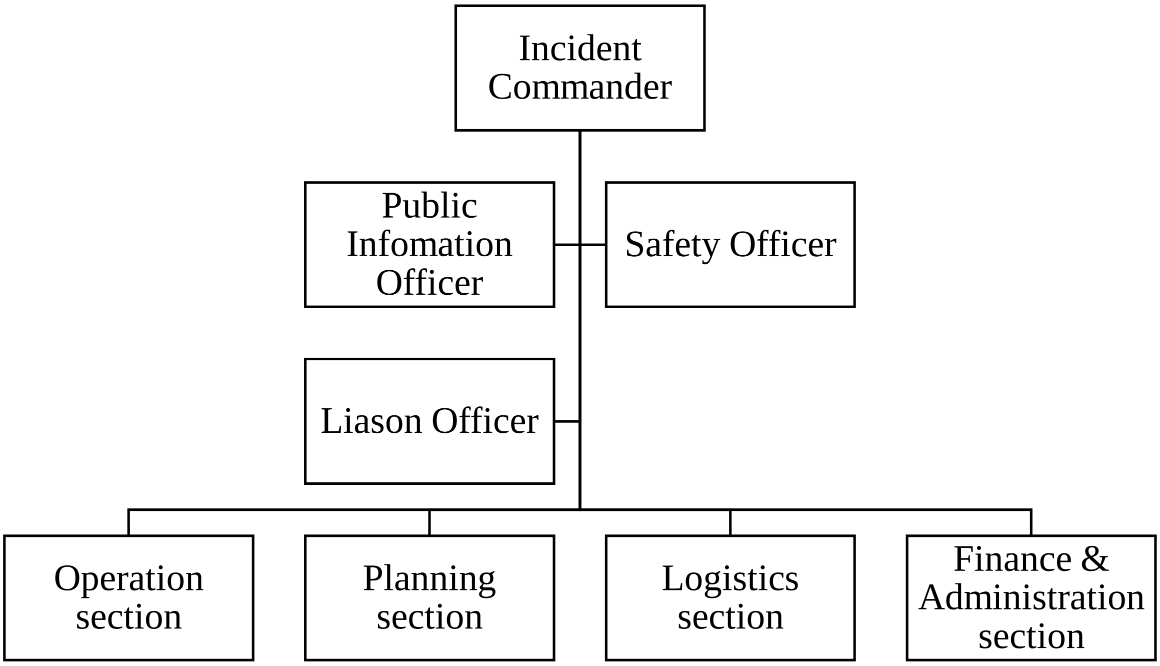


Figure 3.1 – Structure of Incident Command System

Roles and responsibilities:

– Command staffs are tasked with carrying out the staff responsibilities required to assist the Incident Commander. Interagency liaison, incident management, safety, and public communication are among these functions. Command staff jobs are created to delegate responsibility for critical operations that are not clearly listed in the general staff functions. In addition to those as required and assigned by the Incident Commander, these posts may include the Public Information Officer, Safety Officer, and Liaison Officer.

– The general staff is in charge of the ICS's operational aspects. The operations, planning, logistics, and finance/administration sectors are often part of the general staff.

The key responsibilities of the safety/OSH officer include:

– hazardous conditions must be identified and mitigated;

– guaranteeing that security messages are imparted and briefings are given;

– using emergency powers to halt and prevent dangerous activity;

- safety consequences are reviewed in the Incident Action Plan;
- appointing qualified specialists to assess unique threats.

3.4 Occupational safety and health controls

To manage the health and safety concerns posed by diverse hazards, strategies for both prevention and mitigation must be in place. The hierarchy of controls in occupational safety and health refers to the desired order of selecting control measures, from the most effective to the least effective. The basic principle is that it is always preferable to attempt to eliminate the hazard first. Where this is not possible, the hazard should be confined first at the source, then along the path, and finally at the person. Because each environment is unique, a workplace evaluation is required to identify hazards and propose management strategies.

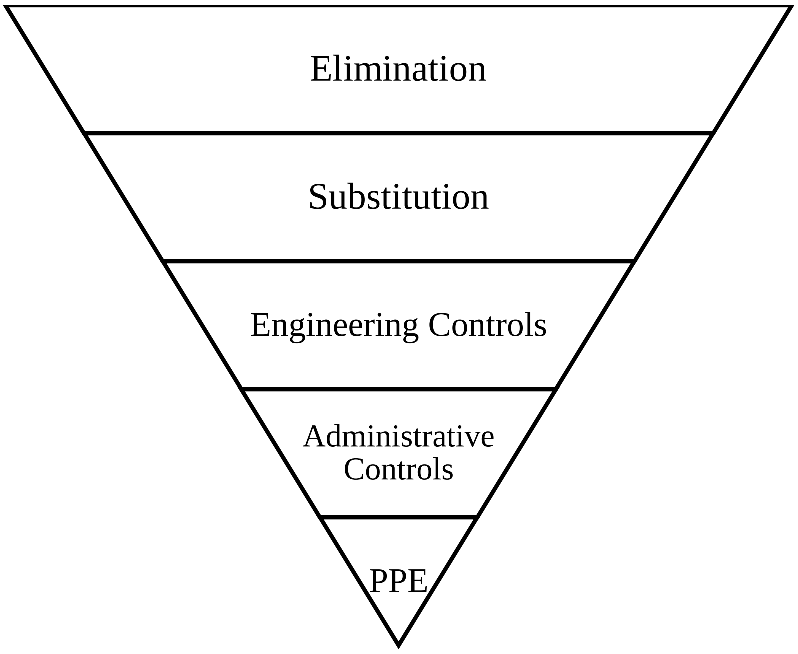


Figure 3.2 – Hierarchy of occupational safety and health controls

Where:

- Elimination; Physically remove the hazard;

- Substitution; Replace the hazard;
- Engineering Controls; Isolate people from the hazard;
- Administrative Controls; Change the way people work;
- PPE (Personal Protective Equipment); Protect the worker with personal protective equipment.

3.5 Conclusion of the chapter 4

Occupational safety and health (OSH), sometimes known as occupational health or occupational safety, is a multidisciplinary area that is concerned with worker safety, health, and well-being. Occupational health is a multidisciplinary branch of medicine that focuses on enabling people to perform their jobs in the most beneficial way possible. It is consistent with the advancement of online safety and security, which focuses on preventing harm from dangers.

Finally, evidence-based facts should be coordinated with professional ability and laborers' esteems. Contextual elements should be considered in relation to regulatory, cultural, monetary, and specialized potential outcomes. In reality, in traditional terms, none of this made a difference.

Places of work under the authority of the area are protected under the law. Maintaining an appropriate and functional workforce, as well as guaranteeing the continuity of emergency response and important health services, necessitates the protection of health-care professionals and other emergency responders' health and safety. Workers in the emergency response area, particularly health-care professionals, are contractually obligated and have a duty of care to provide services that may expose them to infections, poisons, injuries, and diseases. Employers of emergency responders are responsible for providing safe working conditions as well as the resources necessary to carry out approved OSH practices. The administration frameworks technique provides a fundamental structure for supervising OSH dangers during scourges and crises. These gadgets can be tailored to the specific flare-up or crisis situation at hand. The Incident Command System (ICS) is a standardized on-scene incident management approach that enables responders to adopt an integrated organizational structure that is capable of handling the complexity and demands of any single incident or a series of incidents, regardless of jurisdictional boundaries. The ICS provides integrated communication and planning by establishing a suitable span of control.

The following should be included in the minimal ICS and can be expanded if needed:

– The command staff consists of the Public Information Officer, the Safety Officer, and the Liaison Officer.

– Sections represent the organizational level with functional responsibility for the essential segments of incident management (operations, planning, logistics, finance/administration). In terms of organizational structure, the section level is placed between the branch and the Incident Commander.

– Each sector is composed of progressively smaller organizational units, such as a branch, division, group, unit, task force, strike force, and, lastly, a single resource. The requirement for an organizational manager has increased as events have become more sophisticated, difficult, and expensive etc.

The safety/OSH officer's main responsibilities are as follows: To deal with the health and safety risks caused by various hazards, preventative and mitigation techniques must be in place. The core premise is that attempting to eliminate the hazard first is always preferable.

CONCLUSION

As a result of this diploma project, the website of the Ukrainian theater was developed. During the simulation, a diagram of components was developed, which shows all the pages of the site and their interaction. The deployment diagram shows the hardware and software components that interact during the operation of this site. A logical and, on its basis, a physical database scheme for the MySQL DBMS was developed, which reflects the essence of the subject area "Ukrainian Theater". An admin panel was developed especially for the site, which allows you to easily manage the news of the site. The main sections developed for the site are: contacts with feedback form, theater poster, news system, site gallery. The use of the PHP language for processing information and requests to the database, creating a site design and using the jQuery library for site animation is shown.

REFERENCES

1. RBC Ukraine. [Electronic resource] – Access modein: http://www. rbc. ua/rus/news/

1. Information content of the site. [Electronicresource] – Access modein: https://uk.wikipedia.org/wiki/ Complete site

2. Criteria for comparing a site http://habrahabr.ru/post/120989/

[Electronresource] – Access mode:http://ukrteatr.odessa.ua/

3. Unified Modeling Language™ . [Electronicresource] – Access modein: http://www.uml.org/

4. Plan your site with UML. [Elektronresource] – Access mode: http://www.i2r.ru/static/512/

5. PHP: What is PHP? [Електроний ресурс] – Access mode: http://php. net/manual/en/intro-whatis. php

6. MySQL: The world most popular opensource database. [Електроний ресурс] – Access mode: https://www. mysql. com/

7. Logical and physical model of the database. [Electronicresource] – Access modein:http://semestr.com.ua/book/

8. phpMyAdmin Bringing MySQL to the web. [Електроний ресурс] – Режим доступу: http://www. phpmyadmin. net/homepage/

9. CSS (Cascading Style Sheets). [Elektronresource] – Access mode: https://ru.wikipedia.org/wiki/CSS

10. jQuery: write less, do more). [Електроний ресурс] – Режим доступу: https://jquery. com/

11. iView v2.0. [Електроний ресурс] – Режим доступу: http://www. ymic3dworld. com/lightwindow2/iprodev-iView/documentation/