

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Қ.И. Сәтбаев атындағы Қазақ ұлттық техникалық зерттеу университеті

Автоматика және ақпараттық технологиялар институты

Киберқауіпсіздік, ақпаратты өңдеу және сақтау кафедрасы

Жаниева Айгерим

Персоналды басқарудағы ақпараттық жүйенің компонентін жасау

ДИПЛОМДЫҚ ЖҰМЫС

5B070300 – «Ақпараттық жүйелер» мамандығы

Алматы 2022

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Қ.И. Сәтбаев атындағы Қазақ ұлттық техникалық зерттеу университеті

Автоматика және ақпараттық технологиялар институты

Киберқауіпсіздік, ақпаратты өңдеу және сақтау кафедрасы



ҚОРҒАУҒА ЖІБЕРІЛДІ

КАӨЖС кафедрасы меңгерушісі

тех.ғыл.канд., ассоц. профессор

Р.Ж.Сатыбалдиева

«18» мамыр 2022 ж.

ДИПЛОМДЫҚ ЖҰМЫС

Тақырыбы: «Персоналды басқарудағы ақпараттық жүйенің компонентін жасау»

5B070300 – «Ақпараттық жүйелер» мамандығы

Орындаған:

Жаниева Айгерим

Рецензент

PhD докторы, "Ақпараттық жүйелер және киберқауіпсіздік" кафедрасының аға оқытушысы, Алматы энергетика және байланыс университеті

Бимұрат Ж.

«18» мамыр 2022 ж.

Ғылыми жетекші

техн.ғыл.канд., ассоц.профессор

Жумағалиев Б.И

«16» мамыр 2022ж.

Алматы 2022

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Қ.И. Сәтбаев атындағы Қазақ ұлттық техникалық зерттеу университеті

Автоматика және ақпараттық технологиялар институты

Киберқауіпсіздік, ақпаратты өңдеу және сақтау кафедрасы

5B070300 – Ақпараттық жүйелер



БЕКІТЕМІН

КАӨЖС кафедрасы менгерушісі

тех.ғыл.канд., ассоц. профессор

Р.Ж.Сатыбалдиева

«18» мамыр 2022 ж.

**Дипломдық жұмысты орындауға
ТАПСЫРМА**

Білім алушы Жаниева Айгерим

Тақырыбы: «Персоналды басқарудағы ақпараттық жүйенің компонентін жасау»

Университет Ректорының 2021 жылғы «24» желтоқсан №_489-П/Ө бұйрығымен бекітілген.

Орындалған жұмыстың өткізу мерзімі «14» мамыр 2022 ж.

Дипломдық жұмыстың бастапқы мәліметтері: диплом алдындағы практикалық жұмыс қорытындысы, тақырып бойынша деректерге шолу, теориялық мәліметтер.

Дипломдық жұмыста қарастырылатын мәселелер тізімі:

1. қойылған мәселенің қазіргі жағдайын пайымдау
2. ақпараттық қамтаманы құру
3. программалық қамтаманы құру

Графикалық материалдардың тізімі: Power Point, Draw.io сызбалары

Жұмыстың 20 слайдтан тұратын презентациясы көрсетіледі.

Ұсынылған негізгі әдебиет 10 әдебиет тізімінен тұрады.

Дипломдық жұмысты даярлау

КЕСТЕСІ

Бөлім атауы, қарастырылатын мәселелер тізімі	Ғылыми жетекші мен кеңесшілерге көрсету мерзімі	Ескерту
Тақырып аясын талдау және шолу	18.01.2022	
Жүйені жобалау және жасау технологияларын таңдау	15.02.2022	
Бағдарламалық қамтаманы құру	20.03.2021	

Дипломдық жұмыс бөлімдерінің кеңесшілері мен
норма бақылаушының аяқталған жұмысқа қойған
қолтаңбалары

Бөлімдердің атауы	Кеңесшілер (аты-жөні, тегі, ғылыми дәрежесі, атағы)	Қол қойылған мерзімі	Қолы
Норма бақылаушы	Аристомбаева М.Т. (тех.ғыл.магистрі, лектор)	17.05.2022	
Негізгі бөлім	Жумағалиев Б.И. (техн.ғыл.канд, ассоц.профессор)	16.05.2022	

Ғылыми жетекшісі  Жумағалиев Б.И.

Тапсырманы орындауға қабылдаған білім алушы  Жаниева А.М

Күні «13» қаңтар 2022 ж.

АҢДАТПА

«Персоналды басқарудағы ақпараттық жүйенің компонентін жасау» жобасының негізгі мақсаты персоналдарды басқаруды автоматтандыруға негізделген WEB-қосымша жасауға бағытталған. Қазіргі кезде бұл жүйе көбінде қолмен жүзеге асырылатындықтан, WEB-қосымша кәсіпорынды қағазбастылықтан арылтуға септігін тигізеді.

Дипломдық жұмыста жалпы тақырыптың өзектілігі қарастырылып, жобаның басты мақсаты мен тапсырмасы, жобаны дайындау барысында қолданылған технологиялар таңдалынып, бағдарламалау ортасы мен әзірлеу барысы сипатталды және жасалынатын веб-қосымшаның түрі анықталып, жұмыс жасау принципі жобаланды. Жоба нәтижесінде WEB-қосымша жасалынып, оның интерфейстері көрсетілді.

Жоба “Asp.Net Core” платформасында “с#” бағдарламалау тілінде жасалды. Деректерді сақтау жүйесіне “MS SQL Server” таңдалып алынды.

АННОТАЦИЯ

Проект "Разработка компонента информационной системы в управлении персоналом" направлен на создание WEB-приложения, основной целью которого является автоматизация управления персоналом. Поскольку в настоящее время эта система в большинстве случаев реализуется вручную, WEB-приложение способствует избавлению предприятия от бумажной волокиты.

В дипломной работе рассмотрена актуальность общей темы, выбраны основные цели и задачи проекта, технологии, использованные в ходе подготовки проекта, описана среда программирования и ход разработки, определен тип создаваемого веб-приложения, разработан принцип работы. В результате проекта было разработано WEB-приложение и продемонстрированы его интерфейсы.

Проект разработан на платформе "Asp.Net Core" на языке программирования "с#". Для системы хранения данных выбран "MS SQL Server".

THE ANNOTATION

The project "Development of an information system component in personnel management" is aimed at creating a WEB application, the main purpose of which is the automation of personnel management. Since this system is currently implemented manually in most cases, the WEB application helps to rid the enterprise of paperwork.

In the thesis, the relevance of the general topic is considered, the main goals and objectives of the project are selected, the technologies used during the preparation of the project are described, the programming environment and the course of development are described, the type of web application being created is determined, the principle of operation is developed. As a result of the project, a WEB application was developed and its interfaces were demonstrated.

The project is developed on the platform "Asp.Net Core" in the programming language "c#". "MS SQL Server" is selected for the data storage system.

МАЗМҰНЫ

	Кіріспе	9
1	Ақпараттық жүйені құруға жалпы шолу	10
1.1	Персоналды басқарудағы ақпараттық жүйеге шолу	10
1.2	Ақпараттық жүйені құру негіздемесі	11
1.3	Есептің қойылымы	12
2	Web-қосымшаны жобалау	13
2.1	Жүйенің концептуалды моделі	13
2.1.1	Прецеденттер диаграммасы	13
2.1.2	ER диаграммасы	14
2.2	Мәліметтер базасының кестелерін жобалау	15
3	Web-қосымшаны құру	21
3.1	Web-қосымшаны құруда пайдаланылған технологиялар	21
3.1.1	C# бағдарламалау тілі	21
3.1.2	“ASP.NET Core MVC” фреймворкы	21
3.1.3	Жобаны жасау ортасы	22
3.1.4	Мәліметтер базасын басқару жүйесін таңдау	23
3.2	Web-қосымшаның функционалды құрылымын жасау	26
3.3	Web-қосымша жұмысын сипаттау	27
	Қорытынды	34
	Пайдаланылған әдебиеттер тізімі	35
	А қосымшасы	36
	Б қосымшасы	37

КІРІСПЕ

Қазіргі таңда ақпараттық технологиялар адамзат өмірінде үлкен рөл атқаруда. Біз күн сайын, сағат, тіпті минут сайын сан мыңдаған ақпарат аламыз, жібереміз және оны өңдейміз. Сол себептенде жалпы адамзаттың, қызметкердің, жалпы ұйымның тиімділігін арттыруға арналған ақпараттық технологиялар жасалынуда. Кез келген компанияны компьютерсіз немесе интернетсіз елестету қиын. Бүгінгі күні тіпті шағын бизнестің бөлшек саудагерлері автоматтандырылған қосымшаларды пайдаланып қаржылық есептерін жүргізеді.

Ақпараттық жүйелер кәсіпорын қызметінің кез келген саласын дерлік автоматтандыруға, қажетті ақпаратты дұрыс формада беру үшін ресурстарды тиімді пайдалануға мүмкіндік береді. Кез келген кәсіпорынның ең маңызды ресурсы оның персоналдары болып табылады, сондықтан персоналды басқару мәселелері кәсіпорынның жалпы даму стратегиясында негізгі орындардың бірін алуы керек. Персоналды басқарудың тиімділігі көбінесе ұйымның персоналды басқарудағы таңдаған жүйесіне байланысты, мұнда кадр қызметкерлері пайдаланатын ақпараттық жүйелер маңызды рөл атқарады.

Қазіргі кезде қағазбастылықты азайту, персоналды басқару жүйесін автоматтандыру жағдайы өзекті мәселелердің бірі болып отыр.

Дипломдық жұмыс негізінде ақпараттық жүйе келесідей мәселелерді шешуі керек:

- персоналдар туралы барлық мәліметтерге оңай қол жеткізу;
- персоналдардың жұмыс кезеңдерін оңтайлы белгілей алу;
- табельдік есеппент қамтамасыздандыру;
- персоналдарды басқаруды автоматтандыру;

Дипломдық жұмысты жасаудың ғылыми негізгі міндеті – персоналды басқаруды автоматтандырылған түрде жүзеге асыратын ақпараттық жүйені құру.

1 АҚПАРАТТЫҚ ЖҮЙЕНІ ҚҰРУҒА ЖАЛПЫ ШОЛУ

1.1 Персоналды басқарудағы ақпараттық жүйеге шолу

Заманауи персоналды басқарудың ақпараттық жүйелері, ең алдымен, кәсіпорындардың персоналды басқару бөлімшелерінің (бухгалтерлік және кейбір басқа да бөлімдерді есептемегенде) жұмысын оңтайландыруға, олардың жұмысының өнімділігін арттыруға арналған. Атап айтқанда, персоналды басқарушылар осындай жүйелердің көмегімен персоналдармен жұмыс жасау, бұйрықтарды дайындау және есепке алу кезінде күнделікті операциялардан құтылады (персоналдар бөлімінің қызметкерлері жұмыс уақытының 60% -на дейін тек құжаттамамен жұмыс істеуге жұмсайды деген есеп бар). Персоналдар туралы толық ақпаратты автоматтандырылған сақтау және өңдеу сонымен қатар қызметкерлерді тиімді жұмысқа қабылдауға және ауыстыруға мүмкіндік береді. Сонымен қатар, автоматтандырылған еңбекақыны есепке алу штаттық лауазымдар, еңбек демалысы, ауру бойынша демалыстар, сыйлықақылар және іссапарлар туралы ақпаратты ескере отырып, бухгалтерия қызметкерлеріне жалақыны дәл және жылдам есептеуге мүмкіндік береді. Және бұл персоналды басқарудың қазіргі заманғы ақпараттық жүйелерінің кейбір функциялары ғана. [1,2]

Ақпараттық жүйелер беретін деректер, әдетте, белгілі бір процесте, өткен ағымдағы оқиғаларды тұрақты немесе бір реттік басқару есептері түрінде қалыптастыруға болады. Осы мәліметтерді салыстыра отырып, кәсіпорынның жақын болашақтағы даму бағытына жеткілікті дәл болжам жасауға және оның ресурстарын дұрыс бөлуге болады.

Ірі кәсіпорындар күрделі ақпараттық жүйелерді пайдаланады. Ақпараттық жүйе- ол ақпаратты сақтауға, өңдеуге және пайдалануға негізделген жүйе болып табылады. Қазіргі кезде ірі кәсіпорындарды және ұйымдарды басқаруды заманауи ақпараттық жүйелерді қолданбай елестету мүмкін емес. Персоналды басқарудың қазіргі заманғы ақпараттық жүйелері, ең алдымен, кәсіпорынның кадр бөлімшелерінің басшылығы мен персоналының жұмысын оңтайландыруға арналған.[3]

Персоналды басқару процесінде ақпараттық жүйелер жүзеге асыратын негізгі функциялар шешімдерді әзірлеу және персоналдардың осы шешімдерді уақтылы орындауын бақылау болып табылады. Осылайша, персоналды басқарудың ақпараттық жүйесі персонал мәселелерін шешудің ұйымдастырушылық процестерін, жұмыс уақытының кестесін, жұмыс процесін, ішінара еңбекақыны есептеуді автоматтандыруға мүмкіндік беретін технологиялары бар қажетті бағдарламалық қамтаманы әзірлеуді және пайдалануды қамтиды. Бұл ретте кәсіпорын басшылығы іссапар мерзімін ұйымдастыру, еңбек демалысын уақтылы беру, ынталандыру төлемдері және түптеп келгенде кәсіпорынның бизнесін табысты жүргізу үшін әрбір қызметкер туралы ақпаратқа қол жеткізе алады.

Персоналды басқару – бұл кәсіпорындағы қызметкерлерді үйлестіруді және реттеуді жүзеге асырады. Бұл салаға қызметкерлерді басқару, жұмысқа алу, лауазымын жоғарылату немес жұмыстан шығару кіреді. Жақсы жұмыс істейтін персоналды басқарушымен кәсіпорын көбіне тиімді және бәсекеге қабілетті кәсіпорындаы жасай алады, сол себептенде кәсіпорын үшін білікті персоналды басқарушыны таңдау өте маңызды мәселелердің бірі болып табылады.

Әртүрлі кәсіпорындарда жақсы таңдалған қызметкерлер тиімділікке және жақсы табыс көзіне жетелейді. Сол себептенде, персоналды басқарушының ең негізгі мақсаты – барлық персоналдарды жақсы жұмыс жасауға және берілген тапсырманы жақсы орындайтындай етіп ұйымдастыру, бұл тұрғыда персоналды басқарушының рөлі зор.

Персоналды басқарушы әр бөлімде қанша қызметкер болуы керек, қызметкерлердің ақпараттарымен жұмыс жасай отырып, басшылықпен қызметкер арасындағы байланыс орнататын және администрациямен қарым-қатынасты қамтамасыз етеді.

Қызметкерлерді басқарудың маңыздылығы оның тиімді жұмыс күшін ұйымдастырудағы маңыздылығымен байланысты болып келеді. Кез-келген ұйымның қызметкерлерді басқару бөлімі қызметкерлердің мәліметтеріне және олардың құндылықтарына байланысты сұрақтарға жауап береді, мысалы , жұмыс орнындағы жұмыс лауазымына қажеттілікті анықтау, белгіленген жұмыс күшін қанағаттандыру үшін қажетті қызметкерлерді жалдау және таңдалған адамдардың мүмкіндіктеріне сәйкес келетін лауазымдарға сәйкес келуіне бағытталған жаттығулар.[4]

1.2 Ақпараттық жүйені құру негіздемесі

Персоналды басқарудың ақпараттық жүйелері – бұл жұмыс процесі, табельдік есеп, персоналды басқару, жалақы төлеу және есептеу сияқты салалардағы бизнес-процестерді жетілдіруге және автоматтандыруға мүмкіндік беретін белгілі бір технологиялар мен бағдарламалық қамтамасыздандыру жиынтығы.

Бұл бағдарламалық жасақтама әртүрлі бөлімдердің әрекеттерін, сондай-ақ қызметкерлер мен басшылықтың өзара әрекеттесу арналарын үйлестіруге мүмкіндік береді. Бұл бағдарламалық жасақтама қызметкерлерге демалыстар, оқыту, іссапарлар бойынша бюджеттерді ең жақсы бақылау және жоспарлау үшін қажетті қызметкер туралы толық ақпаратқа қол жеткізуге мүмкіндік береді. Шешімдерді әзірлеу және олардың орындалуын бақылау сияқты функциялар ұйымды басқару жүйесінің әртүрлі деңгейлерінде жүзеге асырылады.

Осы функциялардың нақты орындалуы бізге кадрларды басқаруды ақпараттық процесс ретінде және ақпаратты сақтауды, өңдеуді және

пайдалануды функционалды түрде, ал басқару жүйесінің өзі ақпараттық жүйе ретінде қарастыруға мүмкіндік береді.

Персоналды басқару – бұл ұйымды "сапалы" қызметкерлермен қамтамасыз етуге бағытталған және оны оңтайлы пайдалануды көздейтін практикалық қызмет саласы.

Бүгінгі таңда персоналды басқару ақпараттық технологиялармен тығыз байланысты. Компьютерлік техника, интернет, сондай - ақ ақпараттық технологиялармен жұмыс істеу дағдыларын меңгерген персонал бәсекеге қабілетті кәсіпорынның қажетті атрибуттары.

Ақпараттық жүйені құру арқасында кәсіпорындарда персоналдарды басқарудағы мынадай өзекті мәселелер шешіледі:

- кәсіпорын қызметкерлері үшін арнайы нұсқаулықтар мен жұмыс кестелерін қалыптастыру;
- кәсіпорын қызметкерлерінің дербес деректерін тіркеу;
- жаңа қызметкерлерді іздеу немесе олардың лауазымын жоғарылату;
- еңбек ресурстарын пайдалану, оларды жоспарлау және есепке алу;
- еңбек демалысы, іссапар, ауру бойынша демалыс кезеңін бақылау;
- құжаттармен басқару жүйесін ұйымдастыру;

Сонымен қатар, бағдарламалық жасақтама кәсіпорынның персоналын басқаруда құпия ақпаратты қорғауда өте маңызды рөл атқарады.

Кәсіпорында персоналды басқарудың ақпараттық жүйелерін енгізудің әсерін үш аспектіде қарастыруға болады:

1) Ұйымдастырушылық – кәсіпорынды басқарудың барлық деңгейлерінде шешім қабылдау уақытын қысқарту және қызметкерлер шешімдерінің сапасын арттыру;

2) Экономикалық – персоналды басқару шығындарын азайту, қызметкерлердің еңбек өнімділігін арттыру және кәсіпорынның нақты қызметкерінің кәсіби сапасын оңтайлы пайдалану.

3) Әлеуметтік – кәсіпорын қызметкерлерінің еңбек өтілімі жинақтарын дербес есепке алу және кәсіпорын персоналының толық жеке еңбек тарихын жүргізу, басшылық резервті дайындау және кәсіпорынның неғұрлым перспективалы қызметкерлерін қызмет лауазымы бойынша жоғарылату.[5,6,7]

Осылайша, персоналды басқарудың ақпараттық жүйесін құру маңызды және қазіргі заман талабы.

1.3 Есептің қойылымы

Жобаның негізгі мақсаты, персоналды басқарудағы ақпараттық жүйені жасап шығу, персоналдарды басқарушының персоналдар туралы ақпараттарға қол жеткізуі, олардың іссапар кезеңін және еңбек демалысы уақытын бақылап отыруы. Осы әрекеттердің барлығының автоматтандырылған түрде іске асырылуы.

Ұсынылған веб-қосымшаның негізгі міндеттері:

- Жобалау және әзірлеу құралдарын яғни технологияларды таңдау;
- Ақпараттық жүйені жобалау;
- Ақпараттық жүйені әзірлеу;
- Персоналдар туралы барлық мәліметтерді көре алу;
- Персоналдардың іссапар кезеңін белгілей алу;
- Персоналдардың еңбек демалысы уақытын белгілей алу;
- Персоналдарды жұмысқа алу және жұмыстан шығару

күзiреттiлiгi;

Жобаны құру үшін келесі орталар қолданылады:

- Visual Studio ортасы;
- C# бағдарламалау тілі;
- Microsoft SQL Server Management Studio мәліметтер базасы;
- UML объектілі модельдеуге арналған ортасы.

2 WEB-ҚОСЫМШАНЫ ЖОБАЛАУ

2.1 Жүйенің концептуалды моделі

Ақпараттық жүйені жасау және жобалау оны қолданудың концептуалды моделін құрып алудан басталады. Ақпараттық жүйені концептуалды моделін құруда, ең алдымен, ақпараттық жүйені құру мен пайдалануды қамтамасыз ететін нақты тапсырмалар мен функциялардың шеңберін, сондай-ақ ақпаратты жинау, жинақтау және беру жүйесін анықтайды.

Ақпараттық жүйенің концептуалды моделі – жүйеге келтірілген, жүйенің мазмұнды сипаттамасы. Дайындалған модель сипатталатын жүйенің негізгі басты нысандарды, олардың сипаттамаларын және нысандар арасындағы өзара әрекеттесуді анықтауды қамтиды. Модельді құру жағдайында кестелер, графиктер, диаграммалар және т.б. пайдаланылуы мүмкін. Біздің жағдайда біз UML диаграммаларды, оның ішінде прецеденттер, яғни қолданушылар арасындағы байланысты көрсететін диаграмма және ER, яғни мәліметтер базасы арасындағы байланысты көрсететін диаграммаларды пайдаланатын боламыз.

UML диаграммалар — кез-келген бағдарламалық жасақтама жасау үшін маңызды құжаттама модельдерін қолданылатын жобалау диаграммалар. Ол кез-келген бағдарламалық қосымшалар мен аппараттық жүйелердің жұмысын көркемдейтін модельдерді жасауға көп септігін тигізеді.

UML диаграммалары кез-келген жүйені әзірлеудің таптырмас бөлігі болып табылады және кәсіби модель құруда тамаша тәсіл қолданылады. Ол оңай түсінуге болатын қосымшалар үшін модельдер мен құрылымдарды құруға көп септігін тигізеді.[8]

2.1.1 Прецеденттер диаграммасы

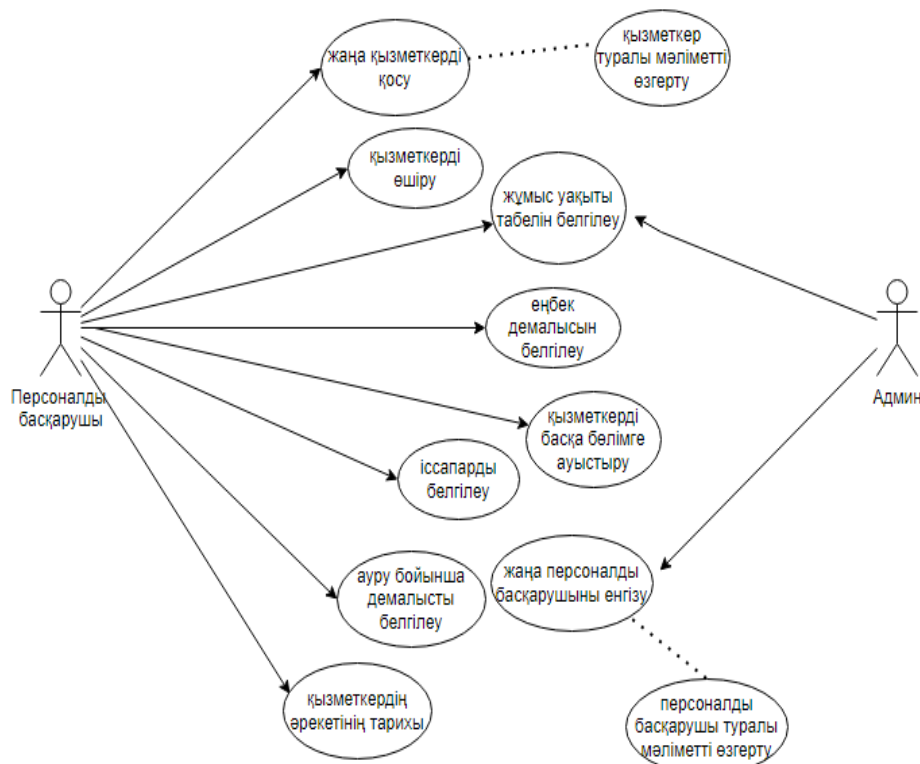
Прецеденттер диаграммасы – актерлер мен прецеденттер арасындағы байланысты және жүйеде әр түрлі рөлдерді және сол рөлдердің жүйемен қалай әрекеттесетінін көрсетуге мүмкіндік беретін диаграмма. Пайдалану жағдайларының диаграммасындағы бұл нұсқаулық келесі тақырыптарды қамтиды және пайдалану жағдайларын жақсартуға көмектеседі.

Диаграмма бағдарламалардың ішкі жүйесіне қажет әрекеттер мен функцияларды суреттейді. Прецеденттер диаграммасы негізгі бірнеше нысаннан құралады. Ол актер, яғни қатысушы және прецедент, яғни актер орындайтын іс-әрекет.

- Біздің жағдайда 2 актер бар. Ол персоналды басқарушы және админ. Персоналды басқарушы құзырында мынадай прецеденттер бар. Олар:
- Жаңа қызметкерді қосу және өшіру;
 - Жұмыс уақыты табелін белгілеу;
 - Іссапар кезеңін белгілеу;

- Ауру бойынша демалыс кезеңін белгілеу;
 - Еңбек демалысы кезеңін белгілеу;
 - Қызметкерді басқа бөлімге ауыстыру;
- Ал админ құзырында мынадай прецеденттер бар:
- Жаңа персоналды басқарушыны енгізу және өшіре алу;
 - Жұмыс уақыты табелін көре және белгіле алу;

Төмендегі 1-суретте Web-қосымшаның прецеденттер диаграммасы көрсетілген.



Сурет 1. Прецеденттер диаграммасы

2.1.2 ER диаграммасы

Жобада реляциялық мәліметтер базасы пайдаланылатындықтан, web-қосымшаның объектілерін жобалау үшін ER диаграммасы қолданылады, ол нысандарды кесте түрінде сипаттайды және олардың байланысын білдіреді.

ER диаграмма – бұл мәліметтер базасының ішіндегі құрылымы қандай жолмен байланысқанын көрсететін қатынас диаграммасы болып табылады. ER диаграммалары мәліметтер базасының деректерін жобалап көрсетуде және ақпараттық жүйені дайындауда қолданылады.

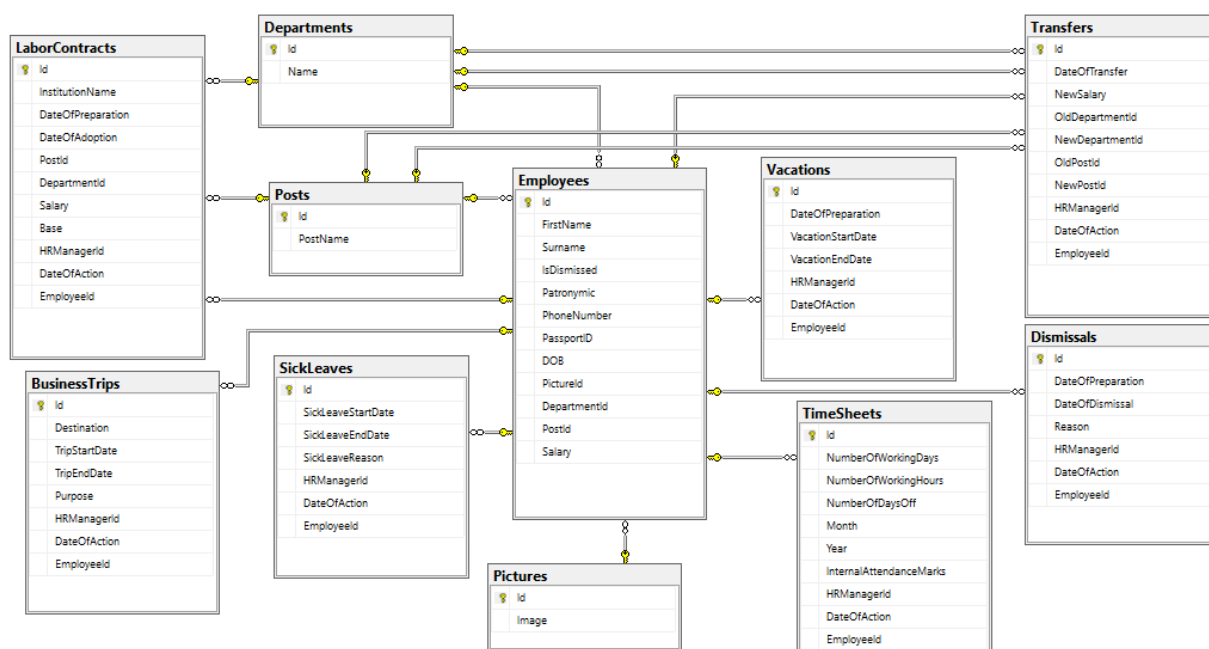
ER диаграммаларының мәліметтер базасы арасындағы байланыстарды және қатынастарды көрсетеді, сондай-ақ олар тек реляциялық деректерқорын

құрылымды көрсетеді. Мәліметтер өрістерге мен бағандарға және жолдарға нақты бөлінуі қажет, әйтпесе ER диаграммасының пайдасы аз болады. Бұл ішкі құрылымдалған мәліметтерге де қатысты.

ER диаграммасында жобаның мәліметтер базасындағы мынадай кестелер көрсетілген:

- Employees;
- Departments;
- LaborContracts;
- Posts;
- Pictures;
- SickLeaves;
- BusinessTrips;
- Vacations;
- Timesheets;
- Dismissals;
- Transfers;

Төмендегі 2-суретте Web-қосымшаның ER-диаграммасы көрсетілген.



Сурет 2. ER диаграммасы

2.2 Мәліметтер базасының кестелерін жобалау

Web-қосымшаны жасау барысында құрылған мәліметтер базасының кестелері төменде көрсетілген. Әрбір кестелерде id өрісі «бастапқы кілтке» ие және ол қайталанбайды(уникальный), бастапқы кілттен бөлек «сыртқы кілт» болады, ол бір кестемен екінші бір кестені байланыстырады.

«Employees» кестесінде персоналдар туралы толық ақпараттар, аты-жөні, ұялы телефон нөмірі, жеке куәлігінің нөмірі, суреті, жұмысқа кіру уақыты, қай бөлімде, қандай мамандықта жұмыс жасайтындығы және жалақысы сынды ақпараттар сақталатын болады.(1-кесте)

1-кесте – Персоналдар кестесінің құрылымы

Атауы	Өрістің атауы	Типі	Кілт
Employees	Id	int	бастапқы(первичный)
	FirstName	nvarchar(MAX)	
	Surname	nvarchar(MAX)	
	IsDismissed	bit	
	Patronymic	nvarchar(MAX)	
	PhoneNumber	nvarchar(MAX)	
	PassportID	nvarchar(MAX)	
	DOB	datetime2(7)	
	PictureId	int	сыртқы (внешний)
	DepartmentId	int	сыртқы (внешний)
	PostId	int	сыртқы (внешний)
	Salary	real	

«BusinessTrips» кестесінде персоналдың іссапар кезеңінің басталу және аяқталу уақыты, іссапар кезеңі өтетін жер сынды ақпараттар сақталатын болады.(2-кесте)

2-кесте – Персоналдардың іссапар кезеңі кестесінің құрылымы

Атауы	Өрістің атауы	Типі	Кілт
BusinessTrips	Id	int	бастапқы
	Destination	nvarchar(MAX)	
	TripStartDate	datetime2(7)	
	TripEndDate	datetime2(7)	
	Purpose	nvarchar(MAX)	
	HRManagerId	nvarchar(MAX)	сыртқы
	DateOfAction	datetime2(7)	
	EmployeeId	int	сыртқы

«Departments» кестесінде персонал жұмыс жасайтын бөлімі ақпараттары сақталатын болады.(3-кесте)

3-кесте – Персоналдардың жұмыс бөлімі кестесінің құрылымы

Атауы	Өрістің атауы	Типі	Кілт
Departments	Id	int	бастапқы
	Name	nvarchar(MAX)	

«Pictures» кестесінде персоналдың суреттері сақталатын болады.(4-кесте)

4-кесте – Персоналдардың суреттері сақталатын кесте құрылымы

Атауы	Өрістің	Типі	Кілт
Pictures	Id	int	бастапқы
	Image	varbinary(MAX)	

«Dismissals» кестесінде персоналдың жұмысқа кіру мен жұмыстан шығу күндері және шығу себебі ақпараттары сақталатын болады.(5-кесте)

5-кесте – Персоналдың жұмыстан шығу мәліметі сақталатын кесте құрылымы

Атауы	Өрістің атауы	Типі	Кілт
Dismissals	Id	int	бастапқы
	DateOfPreparation	datetime2(7)	
	DateOfDismissal	datetime2(7)	
	Reason	nvarchar(MAX)	
	HRManagerId	nvarchar(MAX)	сыртқы
	DateOfAction	datetime2(7)	
	EmployeeId	int	сыртқы

«LaborContracts» кестесінде персоналдың жұмысқа алынған күні, қандай бөлімге, қандай мамандыққа алынғандығы және жалақысы туралы ақпараттары сақталатын болады.(6-кесте)

6-кесте – Персоналдың еңбек шарты сақталатын кесте құрылымы

Атауы	Өрістің атауы	Типі	Кілт
Labor Contracts	Id	int	бастапқы
	InstitutionName	nvarchar(MAX)	
	DateOfPreparation	datetime2(7)	
	DateOfAdoption	datetime2(7)	
	PostId	int	сыртқы
	DepartmentId	int	сыртқы
	Salary	real	
	Base	nvarchar(MAX)	
	HRManagerId	nvarchar(MAX)	сыртқы
	DateOfAction	datetime2(7)	
	EmployeeId	int	сыртқы

«Posts» кестесінде персонал жұмыс жасайтын мамандық туралы ақпараттар сақталатын болады.(7-кесте)

7-кесте – Персоналдың мамандығы сақталатын кесте құрылымы

Атауы	Өрістің атауы	Типі	Кілт
Posts	Id	int	бастапқы
	PostName	nvarchar(MAX)	

«Vacations» кестесінде персоналдың демалыс кезеңінің басталу және аяқталу уақыты ақпараттары сақталатын болады.(8-кесте)

8-кесте – Персоналдың еңбек демалысы уақыты сақталатын кесте құрылымы

Vacations	Id	int	бастапқы
	DateOfPreparation	datetime2(7)	
	VacationStartDate	datetime2(7)	
	VacationEndDate	datetime2(7)	
	HRManagerId	nvarchar(MAX)	сыртқы
	DateOfAction	datetime2(7)	
	EmployeeId	int	сыртқы

«SickLeaves» кестесінде персоналдың ауру бойынша демалыс кезеңінің басталу және аяқталу уақыты және себебі сияқты ақпараттары сақталатын болады.(9-кесте)

9-кесте – Персоналдың ауру бойынша демалыс кезеңі сақталатын кесте құрылымы

Атауы	Өрістің атауы	Типі	Кілт
SickLeaves	Id	int	бастапқы
	SickLeaveStartDate	datetime2(7)	
	SickLeaveEndDate	datetime2(7)	
	SickLeaveReason	nvarchar(MAX)	
	HRManagerId	nvarchar(MAX)	сыртқы
	DateOfAction	datetime2(7)	
	EmployeeId	int	сыртқы

«Timesheets» кестесінде персоналдың жұмыс кестесі ақпараттары, атап айтқанда жұмыс сағаттары, күндері ақпараттары сақталатын болады. (10-кесте)

10-кесте – Персоналдың жұмыс кестесі сақталатын кесте құрылымы

Атауы	Өрістің атауы	Типі	Кілт
Timesheets	Id	int	бастапқы
	NumberOfWorkingDays	int	
	NumberOfWorkingHours	int	
	NumberOfDaysOff	int	
	Month	int	
	Year	int	
	InternalAttendanceMarks	nvarchar(MAX)	
	HRManagerId	nvarchar(MAX)	сыртқы
	DateOfAction	datetime2(7)	
	EmployeeId	int	сыртқы

«Transfers» кестесінде персоналдың бір бөлімнен, екінші бөлімге ауысуы ақпараттары, атап айтқанда екі бөлім атаулары, ауысу күні және жаңа жалақысы сақталатын болады. (11-кесте)

11-кесте – Персоналдың басқа бөлімге ауысу жағдайы сақталатын кесте құрылымы

Transfers	Id	int	бастапқы
	DateOfTransfer	datetime2(7)	
	NewSalary	int	
	OldDepartmentId	int	сыртқы
	NewDepartmentId	int	сыртқы
	OldPostId	int	сыртқы
	NewPostId	int	сыртқы
	HRManagerId	nvarchar(MAX)	сыртқы
	DateOfAction	datetime2(7)	
	EmployeeId	int	сыртқы

3 WEB-ҚОСЫМШАНЫ ҚҰРУ

3.1 Web-қосымшаны құруда пайдаланылған технологиялар

3.1.1 C# бағдарламалау тілі

C# — объектілерге бағытталатын бағдарламалау тілінің бірі. C# бағдарламашыларға(яғни, программисттерге) .NET жүйесінде жұмыс істейтін қауіпсіз және сенімді web-қосымшалардың сантүрлілерін жасауға мүмкіндік беретін тіл.

C# синтаксисі басқа бағдарламалау тілдерінен едәуір ерекшеленеді. Ол автоматты түрде жадыдан қолданылмайтын немесе өшіріп тасталған қосымшалардың объекттерін жойып отыратындығымен ерекшеленеді. Сонымен қатар, ол көптеген дайын кітапханаларымен және web-қосымшаларды жасайтын фреймворктарды қолдануға қуаттылығы жетеді.[9]

3.1.2 “ASP.NET Core MVC” фреймворкы

Қолданушы интерфейсына "ASP.NET Core MVC" таңдалып алынды.

"ASP.NET Core MVC"- әртүрлі web-қосымшалар жасау үшін арналған фреймворк.

MVC үш құрамдас бөлік, яғни компоненттерден тұрады. MVC құрылымының ең басты артықшылығы бір компонентін өзгерту барысында, ол құрамдас басқа екі компонентке азырақ әсерін тигізеді, яғни бір компонентін бөліп қарастырып өзгерте беруге болады. MVC компоненттері:

Модель (Model) – контроллердің сұрауларына жауап ретінде көріністерге деректерді жібереді және қосымшаның бизнес логикасын қамтиды.

MVC-де модельдер екі негізгі типтен құралады: деректерді көрсету және беру үшін пайдаланатын көріністердің модельдерінен және деректерді басқаруды сипаттайтын домен модельдерінен.

Көрініс (View) – пайдаланушы интерфейсінің сұрауларына жауап береді және модельден алынатын деректерді көрсетеді.

Контроллер (Controller) – пайдаланушы мен жүйенің арасын байланыстырады, пайдаланушының енгізуін басқарады және қажетті жауапты пайдаланушыға беру үшін модель мен көріністі пайдаланады.

Контроллердің немесе көріністердің кез-келген өзгерістері модельге ешқандай әсерін тигізбейді. Контроллер мен көрініс салыстырмалы түрде модельге қарағанда тәуелсіз компоненттері болып табылады.[10]

Қосымшаның компоненттерін бұла бөлу, әр компоненттің өзінің белгіленген ғана жұмысына жауап беруіне әкеліп соқтырады. Осы себепті жеке компоненттермен жұмыс жасау оңайырақ болып саналады. Осының арқасында қосымшаны жасау, жобалау және жеке компоненттеріне сынақ

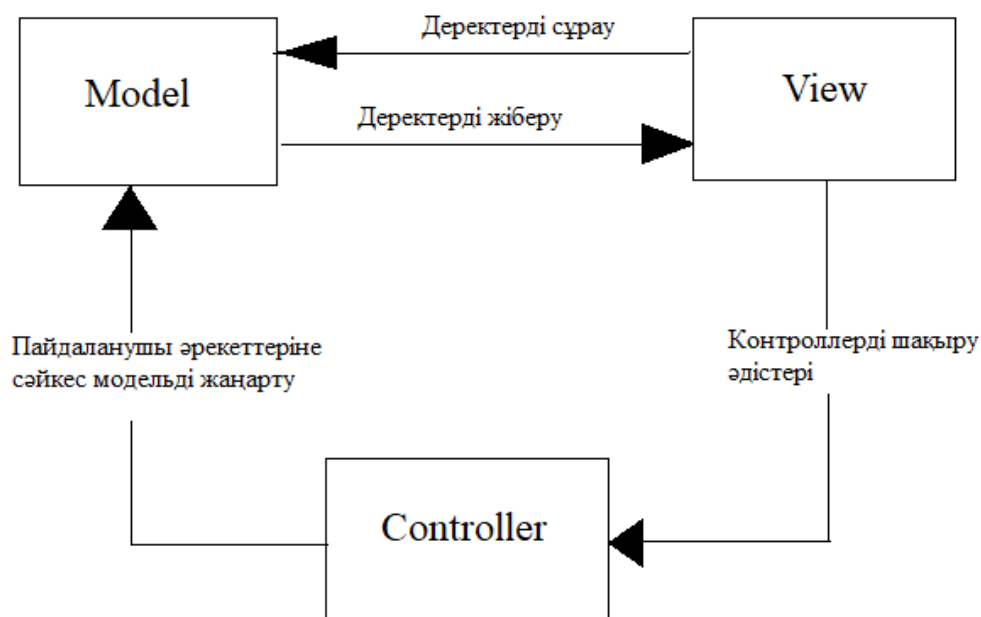
жүргізу оңайырақ болып келеді. Мысалы, егер визуалды бөлігі немесе фронтендті тестілеу қажет болса, онда біз контроллерге тәуелсіз көрініске ғана тексеріс жүргізе аламыз. Немесе біз бэккендке назарды аударып, контроллерге тексеріс жүргізе аламыз.

ASP.NET Core модульді болып келеді, яғни компоненттерді бөлек "NuGet" пакеттік менеджері арқылы жобаға қажетті пакеттерді жүктей аламыз.

Жобада келесідей пакеттік компоненттер қолданылды:

- “Microsoft.AspNetCore.Identity.EntityFrameworkCore(6.0.0)”;
- “Microsoft.AspNetCore.Mvc.Razor.RuntimeCompilation(6.0.0)”;
- “Microsoft.EntityFrameworkCore(6.0.0)”;
- “Microsoft.EntityFrameworkCore.SqlServer(6.0.0)”;
- “Microsoft.EntityFrameworkCore.Proxies(6.0.0)”;
- “Microsoft.EntityFrameworkCore.Sqlite(6.0.0)”;
- “Microsoft.EntityFrameworkCore.Tools(6.0.0)”;

Осылайша “MVC” компоненттері арасындағы байланысты 3 -суреттен көруге болады



Сурет 3. MVC архитектурасы

3.1.3 Жобаны жасау ортасы

Кез-келген web-қосымшасын болсын, windows-қосымшасын болсын жасау үшін жобаны жасау ортасы (IDE) қажет болады. IDE – бұл әртүрлі қосымшаларды жасауды жеңілдете түсетін бағдарламалық жасақтама.

IDE мынадай құрылымдардан тұрады:

- компилятор;

- мәтіндік редактор;
- құрастыруды автоматтандыру құралдары;
- реттеуші.

Ауқымды ASP.NET веб-сайттарының барлығы дерлігі Visual Studio бағдарламасымен жасалынады, сол себептен де IDE ретінде Visual Studio бағдарламасы таңдалынып алынған болатын.

Бұл кәсіби бағдарлама өзінің қателіктерді жөндеу құралдары мен жазу кезінде қателер мен ұсыныстарды көрсете алатын IntelliSense механизмімен өте танымалдылыққа ие. Сонымен қатар, Visual Studio жасалған .NET кодын және веб-парақшаның тегтерін бөлуге мүмкіндік беретін қуатты кодтық модельді қолдайды. Сонымен, жобаны жасау ортасы ретінде Microsoft Visual Studio таңдалынып алынды.

Өзірленген жүйенің негізінде "клиент-сервер" архитектурасы жатыр, онда тапсырмалар немесе желілік жүктеме серверлер деп аталатын қызмет провайдерлері мен клиенттер деп аталатын қызметтерге Тапсырыс берушілер арасында бөлінеді. Интернет клиенттің сервермен өзара әрекеттесу ортасы ретінде қолданылады.

Клиент-сервер архитектурасының негізгі артықшылықтары:

- Көп жағдайда желідегі бірнеше тәуелсіз компьютерлер арасында есептеу жүйесінің функцияларын тарату мүмкіндігі. Бұл есептеу жүйесіне техникалық қызмет көрсетуді жеңілдетеді. Атап айтқанда, серверді ауыстыру, жөндеу, жаңарту немесе жылжыту клиенттерге әсер етпейді.
- Барлық деректер серверде сақталады, ол көптеген клиенттерге қарағанда әлдеқайда жақсы қорғалған. Серверде деректерге тек тиісті қол жетімділік құқықтары бар клиенттерге қол жеткізуге рұқсат беру үшін бақылауды қамтамасыз ету оңайырақ.
- Әр түрлі клиенттерді біріктіруге мүмкіндік береді.

Қосымшаның клиенттік бөлігі келесі технологияларды қолдауы керек:

- Интернет желісіне қолжетімділік.
- HTTP хаттамасы бойынша жұмыс істеу мүмкіндігі.
- Деректерді енгізу үшін адаммен өзара әрекеттесу құрылғыларын қолдау.[10]

3.1.4 Мәліметтер базасын басқару жүйесін таңдау

Жасалынғалы отырған web-қосымша өте көп мәліметтерді сақтайтындығы себепті мәліметтер базасыз жұмыс жасауы мүмкін емес. Web-қосымша мен мәліметтер базасының өзара әрекеттесуі қолданушының сұрауы бойынша нақты уақыт режимінде жауап беру үшін қосымшаның беттерін жедел түрде құруға әкеледі. Ақпаратты толтыру, өзгерту және ұсыну үшін Мәліметтер базасын басқару жүйесі қолданылады. Мәліметтер базасы ұсынатын операциялар:

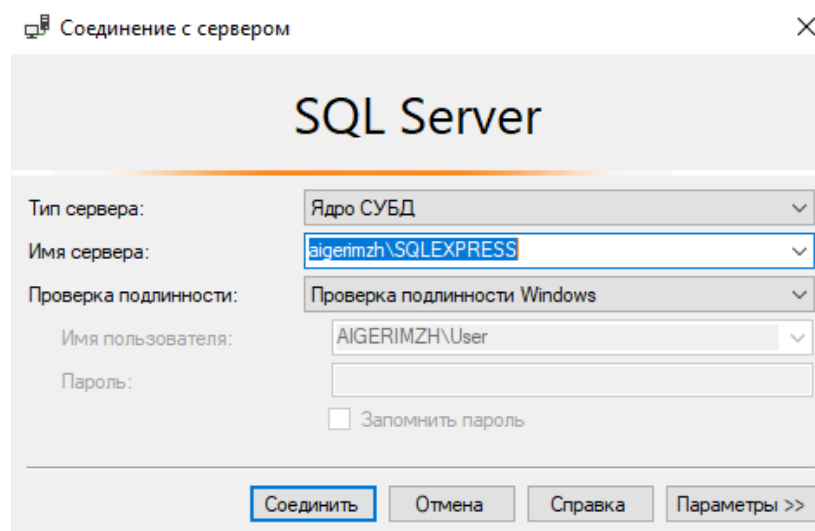
- сыртқы жадтағы деректерді басқару;
- жедел жадтағы деректерді басқару;
- Мәліметтер базасы резервтік көшіру және қалпына келтіру;

Мәліметтер базасының өзі, персоналдар мен олардың басқарушыларынан, персоналдардың еңбек демалысы, ауру бойынша демалысы, іссапар кезеңі сынды мәліметтерден және кәсіпорын туралы маңызды мәліметтерден тұратын кестелерді қамтиды.

Пайдаланушылар егер мәліметтерді қолдануға құқығы бар болатын болса, олармен жұмыс істеу мүмкіндігі болады. Мәліметтерді басқару жүйесінде мәліметтерге қолжетімділік мүмкіндігі бар, яғни персоналды басқарушы өз қол астындағы персонал туралы барлық мәліметтерге қол жеткізе алады және еңбек демалысы уақыттары, ауру бойынша демалысы, іссапар кезеңін белгілей алады және персонал туралы жаңа ақпарат енгізу немесе өзгерту құзыреттілігі бар. Мәліметтер базасын басқару жүйесі мәліметтермен жұмысты жасау кезеңін жылдамдатады және онымен жұмыс жасауды ыңғайлырақ етеді. Жүйеде мәліметтер қауіпсіздігі қатаң түрде сақталған.

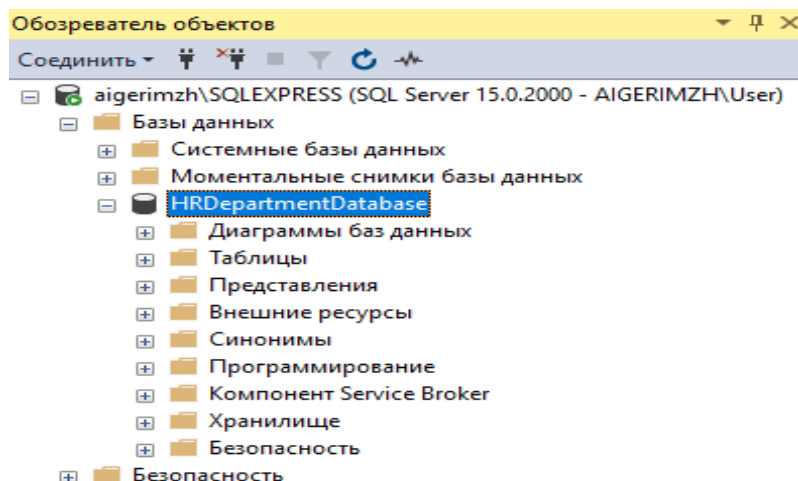
Мәліметтер базасының басқару жүйесі ретінде MS SQL Server бағдарламасы таңдалынып алынған болатын. Ол пайдаланушыларға SQL сұрауларын сұрауға және оларды орындауға мүмкіндік береді. MS SQL Server клиент-сервер архитектурасы негізінде жұмысын жасайды. Клиент– бұл компьютерде орнатылған MS SQL серверіне сұрауларды жібереді. Сервер–сұрау бойынша кіріс деректерін өңдеп жауап жібереді.

Web-қосымша толығымен жұмыс жасауы үшін мәліметтер базасының файлдарын компьютер серверімен байланыс орнату керек. Ол үшін MS SQL Server бағдарламасында, өз мәліметтер базасын қосу керек. (4- сурет).



Сурет 4. Сервермен байланыс орнату

Сервермен байланыстырған кейін, мәліметтер базасын қосу қажет. Ол төмендегі 5- суретте көрсетілген.



Сурет 5. Мәліметтер базасын қосу

Мәліметтер базасын қосқаннан кейін "appsettings" деп аталатын конфигурация файлында орналасқан дерекқорға қосылу параметрлерін өзгертіп, өз дерекқорымызды тіркейміз.

“Server=aigerimzh\SQLEXPRESS\\;Database=HRDepartmentDatabase;Trusted_Connection=True;”

Төмендегі 6-суретте мәліметтер базасымен байланыс орнату көрсетілген.

```
{
  "ConnectionStrings": {
    "DefaultConnection": "Server=aigerimzh\\SQLEXPRESS;Database=HRDepartmentDatabase;Trusted_Connection=True;MultipleActiveResultSets=true"
  },
  "Logging": {
    "LogLevel": {
      "Default": "Information",
      "Microsoft": "Warning",
      "Microsoft.Hosting.Lifetime": "Information"
    }
  },
  "AllowedHosts": "*"
}
```

Сурет 6. Мәліметтер базасын байланыстыру

Мәліметтер базасымен жұмыс келесідей түрде жасалды: қосымшаның серверлік бөлігінде (backend) "appsettings" деп аталатын конфигурация файлында мәліметтер базасына апаратын жолды көрсеттік, web-қосымшаны іске қосқан кезде, мәліметтер базасында кестелер құрылады, серверлік бөлігінде репозиторий көмегімен мәліметтер базасына қол жеткіземіз және басқара аламыз.

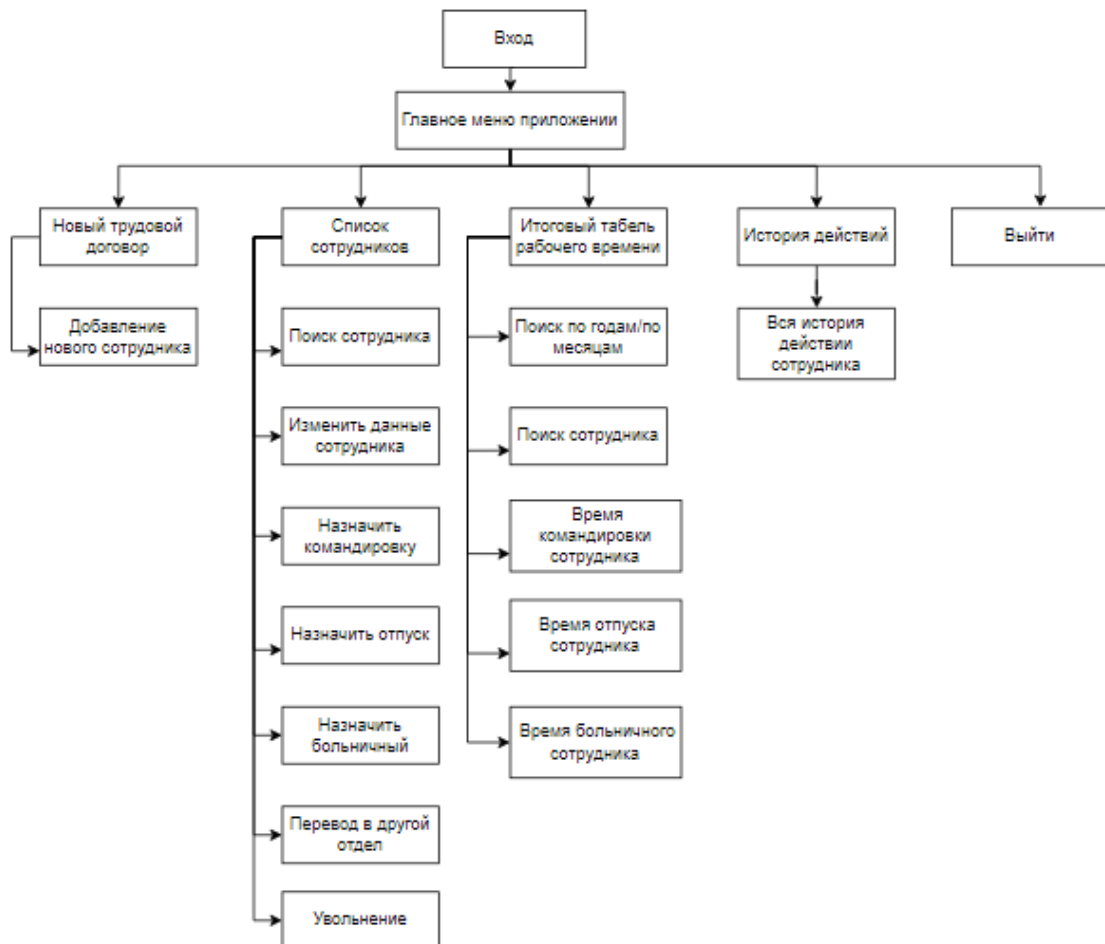
3.2 WEB-қосымшаның функционалды құрылымын жасау

WEB-қосымша функционалды құрылымын жасау – web-қосымшаның интерфейсі негізінде жасалынатын құрылым. Құрылымда қосымшаға кірген сәттен бастап, қосымшадан шығу сәтіне дейін көрсетілген.

Қосымшаға құпия сөз арқылы кіргеннен кейін персоналд басқарушының негізгі бетінде мынадай бөлімдерге бөлінген:

- Жаңа еңбек шарты;
- Қызметкерлердің тізімі;
- Жұмыс уақытының жиынтық тақтасы;
- Әрекеттер тарихы;
- Шығу;

Қосымшаның құрылымын төмендегі 7- суреттен көруге болады.



Сурет 7. WEB-қосымшаның функционалды құрылымы

3.3 WEB-қосымша жұмысын сипаттау

The screenshot shows the login interface of the 'Управление персоналом' (HR Management) application. At the top, there is a navigation bar with 'Управление персоналом' and 'Меню'. The main heading is 'Вход в приложение' (Login to the application). Below this, there are two input fields: 'Логин' (Login) with the value 'hr' and 'Пароль' (Password) with two asterisks. A 'ВОЙТИ' (Login) button is positioned below the password field. The footer contains the copyright notice '© Управление персоналом'.

Сурет 8. Web-қосымшаға кіру терезесі

«Войти» батырмасы арқылы қосымшаға кіре аламыз. Қосымшаға «Hr», яғни персоналды басқарушы немесе «Admin» ретінде кіруге болады.

The screenshot shows the main menu of the application after login. The navigation bar at the top includes 'Управление персоналом', 'Меню', and a user profile 'hr (Hr-Менеджер)'. The main heading is 'Меню' (Menu). Below it, there is a vertical list of five buttons: 'НОВЫЙ ТРУДОВОЙ ДОГОВОР' (New Labor Contract), 'СПИСОК СОТРУДНИКОВ' (List of Employees), 'ИТОГОВЫЙ ТАБЕЛЬ РАБОЧЕГО ВРЕМЕНИ' (Summary of Working Time), 'ИСТОРИЯ ДЕЙСТВИЙ' (History of Actions), and 'Выйти' (Logout). The 'Выйти' button is highlighted with a red border. The footer contains the copyright notice '© Управление персоналом'.

Сурет 9. Web-қосымшаның басты беті

Басты бет бірнеше бөлімге бөлінген:

- Жаңа еңбек шарты;
- Қызметкерлердің тізімі;
- Жұмыс уақытының жиынтық тақтасы;

- Әрекеттер тарихы;
- Шығу;

Заклучение нового трудового договора

Имя	Название учреждения
	университет
Фамилия	Отдел
	Администрация
отчество	Должность
	Менеджер
Фотография	Оклад
Выберите файл Файл не выбран	
Номер телефона	Основание

Сурет 10. Жаңа қызметкерді тіркеу

Бұл бөлімде жаңа қызметкерді тіркеу жүзеге асырылады. Тіркеу жасау үшін, қызметкер туралы толық ақпаратты жазып, толтыру міндетті.

Поиск

☒ Показывать уволенных

Аскарова Мадина Руслановна
Сагидуллаева Айнура Изтурганқызы
Жаналиева Гулжайна Берикқызы
Жаналиева Асемей Мухтарқызы
Жаниева Мафруза Мухитқызы
Ерпанова Аружан Нурланқызы
Владимиров Сергей Антонович
Сергеев Кирилл Русланович
Владимир Путин Владимирович
Жолдыбаев Болат Мырзабаевич
Сергеев Максим Павлович



Личные данные

Фамилия
Имя
Отчество
Дата рождения
Номер паспорта
Номер телефона

Профессиональные данные

Сурет 11. Қызметкерлер тізімі

Келесі бөлімде қызметкерлердің тізімін, олар туралы барлық мәліметті көруге және өзгертуге болады.

Ериланова Аружан Нурланқызы
Владимиров Сергей Антонович
Сергеев Кирилл Русланович
Владимир Путин Владимирович
Жолдыбаев Болат Мырзабаевич
Сергеев Максим Павлович
Павелов Павел Алексеевич
Болатов Мурат Аскарлович
Кулиев Серик Аскарбаевич
Максатов Самат Аскарбекович
Жаксылыков Бакыт Тимурович
Марат Жаксылыков Тимурович

КОМАНДИРОВКА
ОТПУСК
БОЛЬНИЧНЫЙ
ПЕРЕВОД
ТАБЕЛЬ РАБОЧЕГО ВРЕМЕНИ
УВОЛЬНЕНИЕ

Номер паспорта AA1234560
Номер телефона 87473144897

Профессиональные данные

Табельный номер 4
Отдел Разработка
Должность Стажер
Оклад 100000

Сурет 12. Қызметкерлер тізімінің жалғасы

Сонымен қатар бұл бөлімде қызметкердің іссапар кезеңін(13-сурет), еңбек демалысы кезеңін(14-сурет), ауру бойынша демалыс кезеңін(15-сурет), басқа бөлімге ауысуын(16-сурет) немесе жұмыстан шығаруды белгілеуге болады.

Командировка

Дата начала
дд.мм.гггг

Дата окончания
дд.мм.гггг

Пункт назначения

цель

СОХРАНИТЬ

Сурет 13. Қызметкердің іссапар кезеңін белгілеу

Отпуск

Дата составления

дд.мм.гггг --:--



Дата начала

дд.мм.гггг



Дата окончания

дд.мм.гггг



СОХРАНИТЬ

Сурет 14. Қызметкердің еңбек демалысы кезеңін белгілеу

Больничный

Дата начала

дд.мм.гггг



Дата окончания

дд.мм.гггг



Причина

СОХРАНИТЬ

Сурет 15. Қызметкердің ауру бойынша демалыс кезеңін белгілеу

Перевод сотрудника

Новая должность

Менеджер

Новый отдел

Администрация

Новый оклад

0

Дата перевода

01.01.0001



СОХРАНИТЬ

Сурет 16. Қызметкерді басқа бөлімге ауыстыру

Осы жасалған әрекеттердің барлығы қызметкердің әрекеттер тарихында, яғни «Табель рабочего времени» бөлімінде сақталатын болады. (17-сурет) Бұл бөлімде қызметкердің іссапар кезеңі, еңбек демалысы кезеңі, ауру бойынша демалыс кезеңі, басқа бөлімге ауысуы немесе жұмыстан шығу мерзімдерін көруге болады. Сонымен қатар, бұл бөлімде қызметкерлер, жылдар және айлар бойынша іздеу жүргізуге болады.

Составление итогового табеля рабочего времени

январь 2022 ВЫБРАТЬ

Табельный номер	Фамилия, инициалы, должность	Отметки о явках и неявках на работу по числам месяца																															Отраб. (дней)	Отраб. (часов)	Проп.			
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31						
1	Аскарова М.Р. Менеджер	К	К	К	К	К	К	К	К	К	К	К	К	К	К	К	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	31
2	Сагидуллаева А.И. Менеджер	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	31
3	Жаналиева Г.Б. Главный бухгалтер	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	31
4	Абипхайыр А.М.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	К	К	К	К	К	К	К	К	К	К	0	0	0	0	0	0	0	0	0	0	0	0	31

Сурет 17. Қызметкерлердің әрекеттер тарихы журналы

Действия отдела кадров				
Сотрудник				
Все	Действие	Сотрудник	Дата	Кадровик
Hr-Manager	Перевод	Сергеев К.Р.	28.03.2022 22:40:32	Имя И.И.
Все	Больничный	Владимиров С.А.	28.03.2022 22:39:38	Имя И.И.
Действие	Отпуск	Жаниева М.М.	28.03.2022 22:37:18	Имя И.И.
Все	Командировка	Абилхайыр А.М.	28.03.2022 22:35:44	Имя И.И.
Дата	Заключение трудового договора	Талгат М.Т.	28.03.2022 22:31:30	Имя И.И.
ДД.ММ.ГГГГ	Заключение трудового договора	Жаксылыков Б.Т.	28.03.2022 22:25:36	Имя И.И.
☒ За всё время	Увольнение	Курманов С.А.	28.03.2022 22:10:15	Имя И.И.
ПРИМЕНИТЬ				

Сурет 18. Қызметкерлердің әрекеттер тарихы

Персоналды басқарушыдан бөлек web-қосымшада «admin» қолданушысын енгіздім. Ол жаңа персоналды басқарушыны бекіте алады (19-сурет) және оны өшіріп немесе өзгерте алады. (20-сурет)

Управление персоналом	Меню	admin (администратор)
-----------------------	------	-----------------------

Меню

ДОБАВИТЬ НОВОГО ПОЛЬЗОВАТЕЛЯ

СПИСОК ПОЛЬЗОВАТЕЛЕЙ

ИСТОРИЯ ДЕЙСТВИЙ

ВЫЙТИ

© Управление персоналом

Сурет 19. «Админ» қолданушысының басты беті

Новый пользователь

Имя

Фамилия

Фамилия

Роль

HR-Manager

Логин

Сурет 20. Жаңа персоналды басқарушыны енгізу

Управление персоналом Меню

admin (администратор)

Пользователи приложения

[+Добавить пользователя](#)

Логин	ФИО	
admin	Имя Фамилия Отчество	 
hr	Имя Имя Имя	 

© Управление персоналом

Сурет 21. Жаңа персоналды басқарушыны өшіру немесе өзгерту

Қорытынды

Дипломдық жұмыстың нәтижесінде, персоналды басқарудағы ақпараттық жүйе, персоналдарды басқарушының әрекеттерінің барлығы автоматтандырылған түрде іске асыратын веб-қосымша жасалынған болатын. Web-қосымша болашақта өзгертілуі мүмкін қосымша болып саналды, яғни қажет болған кезде оның әртүрлі жаңа тың функциялармен толықтырылуы мүмкін. Web-қосымша автоматтандырылған түрде жұмыс жасайтын болғандықтан, қолданушының жұмысын оңтайландырып, уақыт тапшылығын азайтады. Қосымша барлық техникалық талаптарға сәйкес жасалынып шығарылды, түсінуге оңай пайдаланушы интерфейсіне ие. Көптеген пайдаланушыға бір мезетте әртүрлі жерде пайдалануға болатын клиент-сервер архитектурасына ие етіп жасалынды.

Жоба “Asp.Net Core” платформасында “с#” бағдарламалау тілінде жасалды. Деректерді сақтау жүйесіне “MS SQL Server” таңдалып алынды. Жобаның концептуалды моделі прецеденттер және ER диаграммалары арқылы жобаланды.

Ұсынылған веб-қосымшаның негізгі міндеттері орындалды:

- Ақпараттық жүйені әзірленді;
- Персоналдарды қашықтықтан басқару жүйесі жасалынды;
- Персоналдар туралы барлық мәліметтерді көре алу жүзеге асырылды;

Сонымен, дипломдық жұмыс барысында қойылған негізгі міндеттердің бәрі орындалды.

ПАЙДАЛЫНЫЛҒАН ӘДЕБИЕТТЕР ТІЗІМІ

1. Алавердов, А.Р. Управление персоналом: Учебное пособие / А.Р. Алавердов, Е.О. Куроедова, О.В. Нестерова. - М.: МФПУ Синергия, 2013.
2. Автоматизированные корпоративные системы - необходимый инструмент эффективного ведения бизнеса. /Автор: Малькова О.И [Электрондық ресурс]
Сілтеме: <http://info.tatcenter.ru/economy/16148.htm/>
3. Цели использования автоматизированной системы административного управления /Автор: Гогалев Е.Р. [Электрондық ресурс] Сілтеме:
4. Ивановская, Л.В. Управление персоналом: Теория и практика. Организация, нормирование и регламентация труда персонала: Оқу құралы / Автор: Л.В. Ивановская, 2013.
5. Норенков И.П. автоматизированные информационные системы: Оқу құралы. –М.: МГТУ им.Баумана, 2011.
6. Красильникова М.В. Проектирование информационных систем: Учебное пособие- М.:2004.
7. Кисляков Ю.Н., Слуднов А.В. Информационные технологии Управления персоналом: Учебно-методический комплекс для дистанционного обучения. - М.: Новосибирск: СибАГС, 2005. - 146 с
8. Что такое UML диаграммы? [Электрондық ресурс]. Сілтеме: <https://coderlessons.com/tutorials/kompiuternoe>
9. Язык программирования С# и платформа .NET. [Электрондық ресурс]. Сілтеме: URL: <https://metanit.com/sharp/>
10. ASP.NET и его архитектура [Электрондық ресурс]. Сілтеме: <https://www.guru99.com/what-is-asp-dot-net.html>

А ҚОСЫМШАСЫ

ManagementContrller

```
using HumanResourcesDepartment.Data;
using HumanResourcesDepartment.Models;
using HumanResourcesDepartment.ViewModels;
using Microsoft.AspNetCore.Authorization;
using Microsoft.AspNetCore.Identity;
using Microsoft.AspNetCore.Mvc;
using Microsoft.AspNetCore.Mvc.Rendering;
using Microsoft.EntityFrameworkCore;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;

namespace HumanResourcesDepartment.Controllers
{
    public class ManagementController : Controller
    {
        private readonly ApplicationDbContext _context;
        private readonly UserManager<User> _userManager;

        public ManagementController(ApplicationDbContext context,
UserManager<User> userManager)
        {
            _userManager = userManager;
            _context = context;
        }

        public IActionResult Index()
        {
            if (!User.Identity.IsAuthenticated)
                return RedirectToAction("Login", "Account");

            return View();
        }

        [Authorize]
        public async Task<IActionResult> History()
        {
            var employeesId = await _context.Employees.Select(e => new { Id =
e.Id, Name = e.GetFullName() }).ToListAsync();
            employeesId.Add(new { Id = 0, Name = "Bce" });
        }
    }
}
```

А ҚОСЫМШАСЫНЫҢ ЖАЛҒАСЫ

```
employeesId.Reverse();
ViewData["EmployeesId"] = new SelectList(employeesId, "Id",
"Name", 0);

var HRManagerId = (await
_userManager.GetUsersInRoleAsync("HR-Manager")).Select(u => new { Id = u.Id,
Name = u.GetFullName() }).ToList();
HRManagerId.Add(new { Id = "0", Name = "Bce" });
HRManagerId.Reverse();
ViewData["HRManagerId"] = new SelectList(HRManagerId, "Id",
"Name", 0);

var item= await LoadActionsFromDB();
return View(item);
}

[Authorize]
public async Task<IActionResult>
GetActions(ActionsFilterViewModel model)
{
    List<Models.Action> actions = await LoadActionsFromDB();

    if (model.HRManagerId != "0")
        actions = actions.Where(a => a.HRManager != null &&
a.HRManager.Id == model.HRManagerId).ToList();

    if (!model.AllTime)
        actions = actions.Where(a => a.DateOfAction.ToShortDateString()
== model.Date?.ToShortDateString()).ToList();

    if (model.EmployeeId != 0)
        actions = actions.Where(a => a.Employee.Id ==
model.EmployeeId).ToList();

    if (model.ActionId == 1)
        actions = actions.Where(a => a is LaborContract).ToList();

    if (model.ActionId == 2)
        actions = actions.Where(a => a is TimeSheet).ToList();

    if (model.ActionId == 3)
        actions = actions.Where(a => a is Dismissal).ToList();
}
```

А ҚОСЫМШАСЫНЫҢ ЖАЛҒАСЫ

```
if (model.ActionId == 4)
    actions = actions.Where(a => a is BusinessTrip).ToList();

if (model.ActionId == 5)
    actions = actions.Where(a => a is SickLeave).ToList();

if (model.ActionId == 6)
    actions = actions.Where(a => a is Transfer).ToList();

return PartialView("ActionsList", actions);
}

private async Task<List<Models.Action>> LoadActionsFromDB()
{
    List<Models.Action> actions = new List<Models.Action>();

    List<Department> departments = await
_context.Departments.ToListAsync();
    List<Post> posts = await _context.Posts.ToListAsync();
    List<Transfer> transfers = await _context.Transfers.Include(a =>
a.Employee).Include(a => a.HRManager).ToListAsync();
    transfers.ForEach(t => t.OldDepartment =
departments.FirstOrDefault(d => d.Id == t.OldDepartment.Id));
    transfers.ForEach(t => t.NewDepartment =
departments.FirstOrDefault(d => d.Id == t.NewDepartment.Id));
    transfers.ForEach(t => t.OldPost = posts.FirstOrDefault(p => p.Id ==
t.OldPost.Id));
    transfers.ForEach(t => t.NewPost = posts.FirstOrDefault(p => p.Id ==
t.NewPost.Id));
    actions.AddRange(transfers);

    actions.AddRange(await _context.BusinessTrips.Include(a =>
a.Employee).Include(a => a.HRManager).ToListAsync());
    actions.AddRange(await _context.Dismissals.Include(a =>
a.Employee).Include(a => a.HRManager).ToListAsync());
    actions.AddRange(await _context.LaborContracts.Include(a =>
a.Department)
.Include(a => a.Post).Include(a => a.Employee).Include(a =>
a.HRManager).ToListAsync());
    actions.AddRange(await _context.SickLeaves.Include(a =>
a.Employee).Include(a => a.HRManager).ToListAsync());
```

А ҚОСЫМШАСЫНЫҢ ЖАЛҒАСЫ

```
        actions.AddRange(await _context.Vacations.Include(a
a.Employee).Include(a => a.HRManager).ToListAsync());
        actions.AddRange(await _context.TimeSheets.Include(a
a.Employee).Include(a => a.HRManager).ToListAsync());
        actions.Sort();

        return actions;
    }
}
```

Б ҚОСЫМШАСЫ

Users

```
using HumanResourcesDepartment.Data;
using HumanResourcesDepartment.Models;
using HumanResourcesDepartment.ViewModels.Users;
using Microsoft.AspNetCore.Authorization;
using Microsoft.AspNetCore.Identity;
using Microsoft.AspNetCore.Mvc;
using Microsoft.AspNetCore.Mvc.Rendering;
using Microsoft.EntityFrameworkCore;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;

namespace HumanResourcesDepartment.Controllers
{
    public class UsersController : Controller
    {
        private readonly ApplicationDbContext _context;
        private readonly UserManager<User> _userManager;

        public UsersController(ApplicationDbContext context,
UserManager<User> userManager)
        {
            _userManager = userManager;
            _context = context;
        }

        //[Authorize(Roles = "admin")]
        public IActionResult Index() => View(_userManager.Users.Where(u =>
u.UserName != "unknow").ToList());

        // [Authorize(Roles = "admin")]
        public ActionResult Create()
        {
            ViewData["Roles"] = new SelectList(new List<string> { "admin",
"HR-Manager" });
            return View();
        }

        [HttpPost]
```

Б ҚОСЫМШАСЫНЫҢ ЖАЛҒАСЫ

```
// [Authorize(Roles = "admin")]
[ValidateAntiForgeryToken]
public async Task<IActionResult> Create(CreateUserViewModel
model)
{
    if (ModelState.IsValid)
    {
        User user = new User
        {
            UserName = model.UserName,
            FirstName = model.FirstName,
            Surname = model.Surname,
            Patronymic = model.Patronymic
        };

        var result = await _userManager.CreateAsync(user,
model.Password);

        if (result.Succeeded)
        {
            await _userManager.AddToRoleAsync(user, model.Role);

            return RedirectToAction(nameof(Index));
        }
        else
        {
            foreach (var error in result.Errors)
            {
                ModelState.AddModelError(string.Empty, error.Description);
            }
        }

        ViewData["Roles"] = new SelectList(new List<string> { "admin",
"HR-Manager" });
        return View(model);
    }

    // [Authorize(Roles = "admin")]
    public async Task<IActionResult> Edit(string id)
    {
        EditUserViewModel model;
        User user = await _userManager.FindByIdAsync(id);
```

Б ҚОСЫМШАСЫНЫҢ ЖАЛҒАСЫ

```
if (user == null) return NotFound();

var role = await _userManager.GetRolesAsync(user);

model = new EditUserViewModel
{
    Id = user.Id,
    UserName = user.UserName,
    FirstName = user.FirstName,
    Surname = user.Surname,
    Patronymic = user.Patronymic,
    Role = role.First()
};

ViewData["Roles"] = new SelectList(new List<string> { "admin",
"HR-Manager" }, model.Role);
return View(model);
}

[HttpPost]
// [Authorize(Roles = "admin")]
[ValidateAntiForgeryToken]
public async Task<IActionResult> Edit(EditUserViewModel model)
{
    if (ModelState.IsValid)
    {
        User user = await _userManager.FindByIdAsync(model.Id);

        if (user == null)
        {
            return NotFound();
        }

        user.FirstName = model.FirstName;
        user.Surname = model.Surname;
        user.UserName = model.UserName;
        user.Patronymic = model.Patronymic;

        var updateResult = await _userManager.UpdateAsync(user);

        if (updateResult.Succeeded)
```

Б ҚОСЫМШАСЫНЫҢ ЖАЛҒАСЫ

```
{
    if (model.Password != null)
    {
        await _userManager.RemovePasswordAsync(user);
        await _userManager.AddPasswordAsync(user,
model.Password);
    }

    await _userManager.AddToRoleAsync(user, model.Role);

    return RedirectToAction(nameof(Index));
}
else
{
    foreach (var error in updateResult.Errors)
    {
        ModelState.AddModelError(string.Empty, error.Description);
    }
}
}

ViewData["Roles"] = new SelectList(new List<string> { "admin",
"HR-Manager" });
return View(model);
}

// [Authorize(Roles = "admin")]
public async Task<ActionResult> Delete(string id)
{
    User user = await _userManager.FindByIdAsync(id);

    if (user == null)
    {
        return NotFound();
    }

    var actions = (await LoadActionsFromDB()).Where(a =>
a.HRManager == user).ToList();

    foreach (var item in actions)
        item.HRManager = null;
```

Б ҚОСЫМШАСЫНЫҢ ЖАЛҒАСЫ

```
_context.UpdateRange(actions);
    await _context.SaveChangesAsync();

    await _userManager.DeleteAsync(user);

    return RedirectToAction(nameof(Index));
}

private async Task<List<Models.Action>> LoadActionsFromDB()
{
    List<Models.Action> actions = new List<Models.Action>();
    actions.AddRange(await _context.Transfers.Include(a =>
a.HRManager).ToListAsync());
    actions.AddRange(await _context.BusinessTrips.Include(a =>
a.HRManager).ToListAsync());
    actions.AddRange(await _context.Dismissals.Include(a =>
a.HRManager).ToListAsync());
    actions.AddRange(await _context.LaborContracts.Include(a =>
a.HRManager).ToListAsync());
    actions.AddRange(await _context.SickLeaves.Include(a =>
a.HRManager).ToListAsync());
    actions.AddRange(await _context.Vacations.Include(a =>
a.HRManager).ToListAsync());
    actions.AddRange(await _context.TimeSheets.Include(a =>
a.HRManager).ToListAsync());
    return actions;
}
}
```


Қ. И. Сәтбаев атындағы Қазақ ұлттық техникалық зерттеу университетінің
5В070300 – «Ақпараттық жүйелер» мамандығының 4 курс студенті

Жаниева Айгерим Мухитқызы

«Персоналды басқарудағы ақпараттық жүйенің компонентін жасау»
тақырыбына жазылған дипломдық жұмысына

СЫН ПІКІР

Дипломдық жұмыста Персоналды басқару саласын талдау, олардың ерекшеліктері мен қызметтері анықталып, ақпараттық жүйенің компоненті жасалған. Жалпы деректер базасының моделі және жобаның моделі жасалған. Программалық қамтаманы құру, іске асыру қарастырылып, әр бөлімі тиянақты жасалған.

Кемшілік ретінде кейбір шағын стилистикалық және семантикалық қателерді атап кетуге болады. Алайда бұл кемшіліктер орындалған жұмыстардың жалпы деңгейін төмендетпейді.

Менің пікірімше, диплом жазушы алдына қойылған тапсырманы толығымен орындады және ақпараттық жүйелердің заманауи технологияларын меңгергендігін көрсетті.

Дипломдық жоба квалификациялық жұмыстарға қойылатын талаптарға сәйкес келеді және қорғауға ұсынылады, ал оның авторы Жаниева Айгерим 5В070300 – «Ақпараттық жүйелер» мамандығы бойынша бакалавр академиялық дәрежесін алуға лайықты.

Сын пікір беруші:

Ғұмарбек Дәукеев атындағы

Алматы энергетика және байланыс

университетінің Ақпараттық жүйелер және

киберқауіпсіздік кафедрасының PhD докторы

Бимұрат Жанар



Қолтаңбаны растаймын	
Подпись заверяю	
<i>Салтамағалинов</i>	
Қызметі	аты-жөні
« 20 »	05 2022 ж.

**Университеттің жүйе администраторы мен Академиялық мәселелер департаменті
директорының ұқсастық есебіне талдау хаттамасы**

Жүйе администраторы мен Академиялық мәселелер департаментінің директоры көрсетілген еңбекке қатысты дайындалған Плагиаттың алдын алу және анықтау жүйесінің толық ұқсастық есебімен танысқанын мәлімдейді:

Автор: Жанниева Айгерим

Тақырыбы: Персоналды басқарудағы ақпараттық жүйенің компонентін жасау

Жетекшісі: Жумагалнев Б.И.

1-ұқсастық коэффициенті (30): 0,67

2-ұқсастық коэффициенті (5): 0,00

Әріптерді ауыстыру: 5

Аралықтар: 0

Шағын кеңістіктер: 0

Ақ белгілер: 0

Ұқсастық есебін талдай отырып, Жүйе администраторы мен Академиялық мәселелер департаментінің директоры келесі шешімдерді мәлімдейді :

☒ Ғылыми еңбекте табылған ұқсастықтар плагиат болып есептелмейді. Осыған байланысты жұмыс өз бетінше жазылған болып санала отырып, қорғауға жіберіледі.

☐ Осы жұмыстағы ұқсастықтар плагиат болып есептелмейді, бірақ олардың шамадан тыс көптігі еңбектің құндылығына және автордың ғылыми жұмысты өзі жазғанына қатысты күмән тудырады. Осыған байланысты ұқсастықтарды шектеу мақсатында жұмыс қайта өңдеуге жіберілсін.

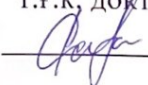
☐ Еңбекте анықталған ұқсастықтар жосықсыз және плагиаттың белгілері болып саналады немесе мәтіндері қасақана бұрмаланып плагиат белгілері жасырылған. Осыған байланысты жұмыс қорғауға жіберілмейді.

Негіздеме:

Күні:

КАӨЖС кафедрасы меңгерушісі

т.ғ.к, доктор, ассоц. профессор

 Р.Ж.Сатыбалдиева

**Университеттің жүйе администраторы мен Академиялық мәселелер департаменті
директорының ұқсастық есебіне талдау хаттамасы**

Академиялық жетекшінің ұқсастық туралы есебі

Жүйе администраторы мен Академиялық мәселелер департаментінің директоры көрсетілген еңбекке қатысты дайындалған Плагиаттың алдын алу және анықтау жүйесінің толық ұқсастық есебімен танысқанын мәлімдейді:

Автор: Жаниева Айгерим

Тақырыбы: Персоналды басқарудағы ақпараттық жүйенің компонентін жасау

Жетекшісі: Жумағалиев Б.И

1-ұқсастық коэффициенті (30): 0.7

2-ұқсастық коэффициенті (5): 0

Дәйексөз (35): 3.1

Әріптерді ауыстыру: 5

Аралықтар: 0

Шағын кеңістіктер: 0

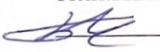
Ақ белгілер: 0

Есепті талдағаннан кейін :

- ☐ Ғылыми еңбекте табылған ұқсастықтар плагиат болып есептелмейді. Осыған байланысты жұмыс өз бетінше жазылған болып санала отырып, қорғауға жіберіледі.
- ☐ Осы жұмыстағы ұқсастықтар плагиат болып есептелмейді, бірақ олардың шамадан тыс көптігі еңбектің құндылығына және автордың ғылыми жұмысты өзі жазғанына қатысты күмән тудырады. Осыған байланысты ұқсастықтарды шектеу мақсатында жұмыс қайта өңдеуге жіберілсін.
- ☐ Еңбекте анықталған ұқсастықтар жосықсыз және плагиаттың белгілері болып саналады немесе мәтіндері қасақана бұрмаланып плагиат белгілері жасырылған. Осыған байланысты жұмыс қорғауға жіберілмейді.

Негіздеме: Жұмыста анықталған қарыздар шамалы және плагиат болып саналмайды. Сондықтан мен жұмысты тәуелсіз деп санаймын және оны қорғауға мүмкіндік беремін.

Күні

Ғылыми жетекші
техн.ғыл.канд., ассоц.профессор
 Жумағалиев Б.И

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Сәтбаев университеті

Ғылыми жетекшінің пікірі

Дипломдық жұмыс

Жаниева Айгерим Мухитқызы

5B070300 – Ақпараттық жүйелер

Тақырыбы: Персоналды басқарудағы ақпараттық жүйенің компонентін жасау

Бұл дипломдық жұмыс өзінің логикалық құрылымымен ерекшеленген. Түсіндірме жобаның құрамы кіріспеден, 3 бөлімнен, қорытындыдан, әдебиеттер тізімінен және қосымшадан тұрады.

Менің пікірімше, диплом жобалаушы алдына қойылған тапсырманы толығымен орындады және кейінгі технологияларын меңгергендігін көрсетті.

Жалпы дипломдық жоба профессионалдық деңгейде орындалған. Түсіндірме жазба сауатты бейнеленген, жоба бойынша барлық қажетті ақпараттар бар.

Кемшілік ретінде кейбір шағын стилистикалық қателерді атап кетуге болады.

Жоғарыда айтылғандарға байланысты, дипломдық жұмыс 5B070300 – «Ақпараттық жүйелер» мамандығының бітіру жұмыстарына қойылатын талаптарына сәйкес және дипломдық жұмыс қорғауға жіберіле алады, ал оның авторы Жаниева Айгерим Мухитқызы бакалавр академиялық дәрежесін алуға лайықты деп есептеймін.

Ғылыми жетекші

Ассоц.проф, т.ғ.к.

«17» мамыр 2022ж.



Жумағалиев Б.И.



Метаданные

Название

Антиплагиат Жаниева Айгерим.docx

Автор

Жаниева Айгерим

Научный руководитель

Подразделение

ИАИИТ

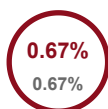
Список возможных попыток манипуляций с текстом

В этом разделе вы найдете информацию, касающуюся манипуляций в тексте, с целью изменить результаты проверки. Для того, кто оценивает работу на бумажном носителе или в электронном формате, манипуляции могут быть невидимы (может быть также целенаправленное вписывание ошибок). Следует оценить, являются ли изменения преднамеренными или нет.

Замена букв		5
Интервалы		0
Микропробелы		0
Белые знаки		0
Парафразы (SmartMarks)		2

Объем найденных подоби

Обратите внимание! Высокие значения коэффициентов не означают плагиат. Отчет должен быть проанализирован экспертом.



КП1

25

Длина фразы для коэффициента подобия 2



КП2

2688

Количество слов



КЦ

23791

Количество символов

Подобия по списку источников

Просмотрите список и проанализируйте, в особенности, те фрагменты, которые превышают КП №2 (выделенные жирным шрифтом). Используйте ссылку «Обозначить фрагмент» и обратите внимание на то, являются ли выделенные фрагменты повторяющимися короткими фразами, разбросанными в документе (совпадающие сходства), многочисленными короткими фразами расположенными рядом друг с другом (парафразирование) или обширными фрагментами без указания источника ("криптоцитаты").

10 самых длинных фраз

Цвет текста

ПОРЯДКОВЫЙ НОМЕР	НАЗВАНИЕ И АДРЕС ИСТОЧНИКА URL (НАЗВАНИЕ БАЗЫ)	КОЛИЧЕСТВО ИДЕНТИЧНЫХ СЛОВ (ФРАГМЕНТОВ)	
1	Мектеп оқушыларын жаппай кітап оқуға тарту мақсатында кітапхана жүйесінің веб-қосымшасын құру 5/28/2021 Satbayev University (ИКИИТ)	18	0.67 %

из базы данных RefBooks (0.00 %)



ПОРЯДКОВЫЙ НОМЕР	НАЗВАНИЕ	КОЛИЧЕСТВО ИДЕНТИЧНЫХ СЛОВ (ФРАГМЕНТОВ)
------------------	----------	---

из домашней базы данных (0.67 %)



ПОРЯДКОВЫЙ НОМЕР	НАЗВАНИЕ	КОЛИЧЕСТВО ИДЕНТИЧНЫХ СЛОВ (ФРАГМЕНТОВ)	
1	Мектеп оқушыларын жаппай кітап оқуға тарту мақсатында кітапхана жүйесінің веб-қосымшасын құру 5/28/2021 Satbayev University (ИКиИТ)	18 (1)	0.67 %

из программы обмена базами данных (0.00 %)



ПОРЯДКОВЫЙ НОМЕР	НАЗВАНИЕ	КОЛИЧЕСТВО ИДЕНТИЧНЫХ СЛОВ (ФРАГМЕНТОВ)
------------------	----------	---

из интернета (0.00 %)



ПОРЯДКОВЫЙ НОМЕР	ИСТОЧНИК URL	КОЛИЧЕСТВО ИДЕНТИЧНЫХ СЛОВ (ФРАГМЕНТОВ)
------------------	--------------	---

Список принятых фрагментов (нет принятых фрагментов)

ПОРЯДКОВЫЙ НОМЕР	СОДЕРЖАНИЕ	КОЛИЧЕСТВО ИДЕНТИЧНЫХ СЛОВ (ФРАГМЕНТОВ)
------------------	------------	---