

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Сәтбаев университеті

Кибернетика және ақпараттық технологиялар институты

Киберқауіпсіздік, ақпаратты өңдеу және сақтау кафедрасы

Бибітова Динара Ақылжанқызы

Стоматологиялық емхананың ақпараттық жүйесін құру

ДИПЛОМДЫҚ ЖҰМЫС

5B070300 - «Ақпараттық жүйелер» мамандығы

Алматы 2020

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Сәтбаев университеті

Кибернетика және ақпараттық технологиялар институты

Киберқауіпсіздік, ақпаратты өңдеу және сақтау кафедрасы

ҚОРҒАУҒА ЖІБЕРІЛДІ

КАӨЖС кафедрасы меңгерушісі
техн. ғыл. канд., ассистент-
профессор

_____ Н.А. Сейлова
«__» _____ 2020 ж.

ДИПЛОМДЫҚ ЖҰМЫС

Тақырыбы: «Стоматологиялық емхананың ақпараттық жүйесін құру»

5B070300 - «Ақпараттық жүйелер» мамандығы

Орындаған
Бибітова Д.А.

Ғылыми жетекші
т.ғ.к., доцент, сениор-лектор
_____ Ш. М. Байматаева
«25» мамыр 2020 ж.

Алматы 2020

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Сәтбаев университеті

Кибернетика және ақпараттық технологиялар институты

Киберқауіпсіздік, ақпаратты өңдеу және сақтау кафедрасы

5B070300 - Ақпараттық жүйелер

БЕКІТЕМІН

КАӨЖС кафедрасы меңгерушісі
техн. ғыл. канд., ассистент-
профессор

_____ Н.А. Сейлова
«__» _____ 2020 ж.

**Дипломдық жұмысты орындауға
ТАПСЫРМА**

Білім алушы: Бибітова Динара Ақылжанқызы

Жобаның тақырыбы: «Стоматологиялық емхананың ақпараттық жүйесі»

Университет Ректорының «27» қаңтар 2020 жылғы №762-б бұйрығымен бекітілген.

Орындалған жұмыстың өткізу мерзімі «__» _____ 2020 ж.

Дипломдық жұмыстың бастапқы мәліметтері: берілген тақырып бойынша әдебиеттерге шолу кезінде жиналған мәліметтер, теориялық материалдар жинау.

Дипломдық жұмыста қарастырылатын мәселелер тізімі:

- а) пәндік аймақты талдау және есептің қойылымы;
- ә) стоматологиялық емхананың бағдарламалық қамтамасын құру;
- б) бағдарламалық қамтаманың баяндалу бөлімі.

Графикалық материалдардың тізімі (міндетті түрде қажет сызбалар көрсетілген): жұмыстың ___ слайдтан тұратын презентациясы көрсетіледі.

Негізгі әдебиет: 14 амау

Протокол анализа Отчета подобия

заведующего кафедрой / начальника структурного подразделения

Заведующий кафедрой / начальник структурного подразделения заявляет, что ознакомился(-ась) с Полным отчетом подобия, который был сгенерирован Системой выявления и предотвращения плагиата в отношении работы:

Автор: Бибітова Динара

Название: Стоматологиялық емхананың ақпараттық жүйесін құру

Координатор: Шолпан Байматаева

Коэффициент подобия 1:20,3

Коэффициент подобия 2:9,3

Замена букв:655

Интервалы:0

Микропробелы:0

Белые знаки:0

После анализа отчета подобия заведующий кафедрой / начальник структурного подразделения констатирует следующее:

- обнаруженные в работе заимствования являются добросовестными и не обладают признаками плагиата. В связи с чем, работа признается самостоятельной и допускается к защите;
- обнаруженные в работе заимствования не обладают признаками плагиата, но их чрезмерное количество вызывает сомнения в отношении ценности работы по существу и отсутствием самостоятельности ее автора. В связи с чем, работа должна быть вновь отредактирована с целью ограничения заимствований;
- обнаруженные в работе заимствования являются недобросовестными и обладают признаками плагиата, или в ней содержатся преднамеренные искажения текста, указывающие на попытки сокрытия недобросовестных заимствований. В связи с чем, работа не допускается к защите.

Обоснование:

.....
.....
.....
.....
.....
.....

Дата

Подпись заведующего кафедрой /

начальника структурного подразделения

Окончательное решение в отношении допуска к защите, включая обоснование:

.....
.....
.....
.....
.....

Дата

Подпись заведующего кафедрой /
начальника структурного подразделения

Протокол анализа Отчета подобия Научным руководителем

Заявляю, что я ознакомился(-ась) с Полным отчетом подобия, который был сгенерирован Системой выявления и предотвращения плагиата в отношении работы:

Автор: Бибітова Динара

Название: Стоматологиялық емхананың ақпараттық жүйесін құру

Координатор: Шолпан Байматаева

Коэффициент подобия 1: 20,3

Коэффициент подобия 2: 9,3

Замена букв: 655

Интервалы: 0

Микропробелы: 0

Белые знаки: 0

После анализа Отчета подобия констатирую следующее:

- обнаруженные в работе заимствования являются добросовестными и не обладают признаками плагиата. В связи с чем, признаю работу самостоятельной и допускаю ее к защите;
- обнаруженные в работе заимствования не обладают признаками плагиата, но их чрезмерное количество вызывает сомнения в отношении ценности работы по существу и отсутствием самостоятельности ее автора. В связи с чем, работа должна быть вновь отредактирована с целью ограничения заимствований;
- обнаруженные в работе заимствования являются недобросовестными и обладают признаками плагиата, или в ней содержатся преднамеренные искажения текста, указывающие на попытки сокрытия недобросовестных заимствований. В связи с чем, не допускаю работу к защите.

Обоснование:

.....

.....
Дата

.....
Подпись Научного руководителя

Дипломдық жұмысты даярлау
КЕСТЕСІ

Бөлім атаулары, қарастырылатын мәселелер тізімі	Ғылыми жетекшіге, кеңесшілерге өткізу мерзімі	Ескерту
Пәндік аймақты талдау және есептің қойылымы		
Стоматологиялық емхананың бағдарламалық қамтамасын құру		
Бағдарламалық қамтаманың баяндалу бөлімі		

Дипломдық жұмыс бөлімдерінің кеңесшілері мен
норма бақылаушының аяқталған жұмысқа қойған
қолтаңбалары

Бөлімдердің атауы	Кеңесшілер (аты-жөні, тегі, ғылыми дәрежесі, атағы)	Қол қойылған мерзімі	Қолы
БҚ жасау	Ш. М. Байматаева т.ғ.к., доцент, сениор- лектор		
Норма бақылаушы	А.Н.Дуйсенбаева техн.ғыл.магистр., лектор		

Ғылыми жетекшісі _____ Ш. М. Байматаева

Тапсырманы орындауға қабылдаған білім алушы _____ Д.А. Бибітова

Күні «25» мамыр 2020 ж.

АНДАТПА

Дипломдық жұмыстың мақсаты стоматологиялық емхананың тіркеу бөліміндегі қызметкерге тіркеудің ыңғайлылығы болып табылады.

Құрылған ақпараттық жүйе стоматологиялық емхананың қызметін пайдалануға тілек білдірген клиенттерді тіркеу, қажетті процедураға тіркеу жұмыстарын атқаратын бөлім қызметкерінің жұмысын автоматтандыруға бағытталған қосымша болып табылады. Бұл программа тіркеуші қызметкердің жұмысын жеңілдетіп қана қоймай, оның уақытын ұтымды пайдалануға да көмектеседі.

Бұл дипломдық жұмысты жасау барысында Microsoft SQL Server мәліметтер базасы мен C# программалау тілі және Microsoft Visual Studio ортасы қолданылды.

АННОТАЦИЯ

Целью дипломной работы является удобство регистрации работника в регистратуре стоматологической поликлиники.

Созданная информационная система является дополнением, направленным на автоматизацию работы сотрудника отдела, осуществляющего регистрацию клиентов, желающих воспользоваться услугами стоматологической поликлиники, на необходимую процедуру. Эта программа поможет не только облегчить работу сотрудника регистратора, но и рационально использовать его время.

При разработке данной дипломной работы использовалась база данных Microsoft SQL Server и язык программирования C# и среда Microsoft Visual Studio.

ANNOTATION

The purpose of the thesis is the convenience of registering an employee at the dental clinic registry.

The created information system is an addition aimed at automating the work of a Department employ who registers clients who want to use the services of a dental clinic for the necessary procedure. This program will not only facilitate the work of the Registrar's employee, but also make rational use of their time.

When developing this thesis, we used the Microsoft SQL Server database and the C# programming language and the Microsoft Visual Studio environment.

МАЗМҰНЫ

Кіріспе	9
1 Пәндік аймақты талдау және есептің қойылымы	10
1.1 Пәндік саланың сипаттамасы	10
1.2 Автоматтандыру объектісінің сипаттамасы	11
1.3 Есептің қойылымы	13
2 Стоматологиялық емхананың бағдарламалық қамтамасын құру	16
2.1 Мәліметтер қорына жалпы шолу	16
2.2 Мәліметтер қорын басқару жүйесі және оның негізгі функциялар	16
2.3 МҚБЖ құрылымы, артықшылығы мен кемшілігі	18
2.4 Мәліметтер базасының физикалық моделін әзірлеу	20
2.5 «Denta+» стоматологиялық емханасының ақпараттық жүйесінің құрылымы	27
2.6 ER-диаграмманы құру	28
2.7 Мәліметтер базасының кестелерін сипаттау	30
2.8 Жүйенің интерфейсі	32
2.9 Бағдарламалау тілін таңдауды негіздеу	33
3 Бағдарламалық қамтаманың баяндалу бөлімі	36
3.1 Жалпы мағлұматтар	36
3.2 Функционалдық тағайындалуы	36
3.3 Логикалық құрылымын сипаттау	36
3.4 Шақыру және жүктеу	36
3.5 Қолданылған техникалық жабдықтар	37
3.6 Кіріс мәлімет	37
3.7 Шығыс мәлімет	37
3.8 Жүйенің жұмыс істеуін сипаттау	37
Қорытынды	45
Пайдаланған әдебиеттер	46
А қосымшасы	47

КІРІСПЕ

Ақпараттық жүйе – бұл пайдаланушылардың ақпараттық қажеттіліктерінің жиынтығы компьютерлік және коммуникациялық жабдықтар, бағдарламалық қамтамасыз ету, сатып алуды және ақпараттық ресурстарды қамтитын кешен.

Стоматологиялық емханалардың ақпараттық жүйелерінің негізгі міндеттері кез келген жағдайда адамдарға ақпараттық стоматологиялық көмек көрсету, стоматологиялық мекемелерді басқару мен денсаулық сақтау жүйесінің өзін ақпараттық қамтамасыз ету.

Стоматологиялық емханаларда ақпараттық жүйені құрудың өзектілігі бүгінгі таңда диагностикалық, терапиялық, статистикалық, басқарушылық және басқа да міндеттерді шешу кезінде үлкен және үнемі өсіп келе жатқан ақпарат көлемін пайдалану қажеттілігінен туындап отыр.

Денсаулық сақтау мекемелерінің қызметін ақпараттандыру бұрыннан бері қазіргі заман ағымдарының данасына ғана емес, маңызды қажеттілікке айналды. Барлық уақытта өсіп келе жатқан ақпарат массивтерін өңдеу қазіргі заманғы компьютерлік технологияларды пайдалана отырып ғана мүмкін болды.

Стоматология қызметтерін есепке алу бойынша ақпараттық жүйені әзірлеу мақсаты:

- медициналық мекемелердің заманауи талаптарына жауап беретін және осы саладағы басқа жүйелермен салыстырғанда өз артықшылықтарына ие болатын жүйені құру;

- бағдарлама базасындағы мәліметтер негізінде есеп беретін және стоматологиялық клиниканың ұсынылатын қызметтерінің сапасына талдау жасайтын жүйені құру.

Қойылған мақсаттарды шешу үшін жүзеге асыру қажет: экрандық нысандарды пайдалана отырып, жеке стоматология базасына мәліметтерді енгізу, бақылау және жүктеуді автоматтандыру, қызметті енгізуді және оның бағасын автоматтандыру, берілген ұйымдар бойынша есептілікті қалыптастыру, қызметті және диагнозды енгізуді автоматтандыру, мәліметтер негізінде есеп жасау, көрсетілетін қызметтердің сапасын талдау жүргізу.

Ақпараттық жүйенің кіші жүйесінің бағдарламалық қамтамасыз етуі C# тілінде Microsoft Visual Studio ортасында әзірленген. Microsoft Visual Studio– бұл .net платформасында жұмыс істейтін қосымшаларды жасауға мүмкіндік беретін Microsoft компаниясының әзірлемесі. Бұл жағдайда сервистер базалық архитектураға тәуелді емес аралық код түрінде іске асырылады. Мұндай платформаны құрудың басты мақсаты әзірлеушілерді дербес компьютерден бастап мобильді құрылғымен аяқталатын кез келген платформада жұмыс істей алатын арнайы сервистік-бағдарлы қосымшалармен жабдықтау болды.

1 ПӘНДІК АЙМАҚТЫ ТАЛДАУ ЖӘНЕ ЕСЕПТІҢ ҚОЙЫЛЫМЫ

1.1 Пәндік саланың сипаттамасы

Стоматологиялық емханаларда ақпараттық технологияларды пайдалану уақытты "қағаз" жұмыстан қысқартуға мүмкіндік береді. Пациенттің электрондық карточкаларын жасау әрбір қызметкерге пациенттің барлық аурулары мен жарақаттары, көптеген көрсеткіштердің өзгеруі туралы толық ақпаратты дереу алуға және аталған препараттың емделушіге қалай әсер ететінін қадағалауға мүмкіндік береді. Егер адамға басқа қалада медициналық көмек қажет болса, және жоғарыда аталған ақпаратты білуге ешқандай мүмкіндік болмаса, бұл әсіресе ыңғайлы.

Денсаулық сақтау жүйесінде АТ қолдану мүмкіндіктері:

- қазіргі заманғы стоматологиялық емханалардың алдында тұрған жоғарыда аталған проблемаларға сүйене отырып, стоматологиялық емханалардағы және денсаулық сақтаудағы ақпараттық технологиялар: пациенттердің оңтайлы және ұтымды есебін жүргізу;

- олардың жай-күйін қашықтан бақылау (әсіресе тіс импланттары немесе басқа органдар болған кезде ыңғайлы);

- аурудың толық тарихын, диагностика нәтижелерін және тағайындалған препараттарды сақтау;

- тағайындалған емнің дұрыстығын бақылау, бұл диагнозды қате қою және қолайсыз ем тағайындау тәуекелін айтарлықтай төмендетеді;

- стоматологиялық емханаларға жаңа зерттеулер, әзірлемелер және технологиялар туралы ақпарат алу;

- жұмысты тиімді жоспарлау және олардың іске асырылуын бақылау, сондай-ақ жоспардан тыс міндеттерді шешу, денсаулық сақтау мекемесі, жоспарлау-экономикалық бөлімі және кадрлар бөлімі әкімшілігі;

- қажетті дәрі дәрмектер қоймаларында медициналық тауарлардың есебін жүргізу, кіріс-шығыс операцияларын тіркеу, белгілі бір препараттардың қажеттілігін талдау және болжау;

- есеп беру құжаттамасын бақылаушы органдарға беруге.

Жоғарыда аталған ақпараттық технологияларды қолдану тәсілдерінен көрініп тұрғандай, олар стоматологиялық емханалардың барлық салаларын қамтиды. АТ шешетін міндеттерге байланысты стоматологиялық емханалардағы ақпараттық технологиялардың келесі сыныптамасын бөледі:

- жүйені медициналық әкімшілендіру;

- стоматологиялық емханалар ақпараттық жүйелер;

- іздеу жүйелері;

- диагностикалық зерттеулерді есепке алу жүйелері;

- телемедиа жүйелері және т.б.

Ақпараттық технологиялар қазіргі өмірдің барлық салаларында берік нығайып келеді, олар стоматология саласында да өз қолдануын таба алды. Тіпті

«стоматологиялық информатика», «компьютерлік стоматология» және басқалар да бар.

Сандық технологиялар стоматологиялық емдеудің барлық кезеңдерінде – медициналық құжаттама нысандарын толтырудан және жүргізуден бастап емханалық жағдайларды және ұсынылатын емдеу жоспарын үлгілеуге дейін және т.б. пайдаланылуы мүмкін.

Тіс протездерін автоматтандыру арқылы жобалау және жасалуы бұл осы технологияның теориялық негіздері ХХ ғасырдың 70-ші жж. басында пайда болған. CAD CAD (автоматтандырылған жобалау), ал өндірісті автоматтандыру жүйесі үшін-Cam (автоматтандырылған өндіріс).

Технология екі бағытта дамиды.

Біріншісі – бұл жеке САПР/САМ жүйесі, ол бір медициналық мекеме шеңберінде жұмыс істеуге мүмкіндік береді, кейде тіпті пациенттің қатысуымен тіс дәрігерінің кабинетінде. Жеке жүйелердің негізгі артықшылығы - дайындау жеделдігі, бірақ толыққанды жұмыс істеу үшін әлі де көп тұрған барлық жабдық кешені қажет.

Екінші нұсқа – әртүрлі жұмыс станциялары үшін конструкциялардың үлкен ассортиментін дайындайтын өндірістік орталықтың болуын болжайтын АЖЖ орталықтандырылған модульдері. Бұл опция әрбір стоматологқа дайындаушы модульді сатып алмауға мүмкіндік береді. Алайда, оның кемшілігі-барлық іс-шаралар кешені бір бару үшін жүргізілуі мүмкін емес, сондай-ақ дайын конструкцияны жеткізу қиындап, қымбатқа түседі. Себебі өндірістік орталық басқа қалада немесе тіпті елде орналасуы мүмкін.

Стоматологиялық емхана саласындағы тұрмыстық қызметтерді автоматтандыру және басқару тұтынушылармен өзара іс-қимыл жасау үшін, сондай-ақ ішкі процестерде ақпараттық технологияларды пайдалану, стоматологиялық өнеркәсіпті дамыту бұл факторлардың бірі. Стоматологиялық желілер санының өсуімен жоғары бәсекелестік, стоматологиялық емханада жұмыс істеуде ақпараттық технологиялар сияқты құрал пайда болғанша тиімді болмайды.

1.2 Автоматтандыру объектісінің сипаттамасы

«Denta+» – стоматологиялық емхана Алматы қаласындағы тіс емдеу емханаларының көшбасшы емханалар желісі. Емхана 2015 жылдан бері жұмыс жасауда. «Denta+» стоматологияқ емханасы қазіргі уақытта Алматы қаласында жұмысын жасауда.

Ал менің бұл дипломдық жұмысымда Алматы қаласы, Төле би көшесі, 221 үйде орналасқан «Denta+» стоматологиялық емханасының автоматтандырылған ақпараттық жүйесін құру қарастырылды. Емхана толықтай автоматтандырылған жүйені қолданбағандықтан, бұл қызметкерлерге тиімсіз болып, жұмыс өнімділігін төмендетуде.

«Denta+» – бұл әр отбасы тіс пен ауыз қуысын сапалы және қауіпсіз ем алатын отбасылық стоматология. «Denta+» стоматологиялық емхана дамуының стратегиясын өзімен бірге, ғылыми айқындамалардың және тәжірибелі-экспериментальды медицинаның берік қорғаны ретінде анықтауға болады. Стоматологиялық емхананың жұмысы, стоматологиялық емхананың маңайындағы тұрғындарды қолайлы стоматологиялық емханамен қамтып, қаланың экономикалық жағдайын көтеруге, қарапайым халықты жұмыспен қамтып, арнайы мамандардың көмегімен қарапайым жұмысшылардың медициналық біліктілігін арттыруға зор үлесін қосуда. Бұл стоматологиялық емхана Алматы қаласының ортасында болғандықтан, сауда – саттық аясы кең, жол қатынасында қолайлы, осындай шарт нәтижесінде күннен – күнге даму деңгейі көтерілуде.

Стоматологиялық емхана қызметінің негізі келесідей:

- Қазақстанды осы заманғы техникалық стоматологиялық жүйелермен камсыздандыруға өз үлесін қосу;

- тұтынушының талаптарына сай, сұранысын сенімді жеткізушінің болуы.

Стоматологиялық емхана ұстанымдары:

- клиенттің мұқтаждығын қанағаттандыру;

- медициналық технологияларын және басқарудың жаңа пішіндерін енгізу;

- персоналдың әлеуметтік шарттарын жақсарту.

Клиентпен және бәсекелестермен қарым-қатынасы:

- тұтынушыға арқашанда есік ашық;

- қоғамның ой-пікірін құрметтеу;

- адал бизнес жүргізу;

- клиенттің сенімін ақтау;

- бәсекелестермен кәсіби және этикалық қағида қатынастарды сақтау.

Стоматологиялық емхана негізгі міндеттері мен құндылықтары – тұрғындарды әлемдік сапа стандартына сәйкес еммен қамтамасыз ету, Қазақстанда стоматологияны дамыту болып табылады.

Жалпы бұл жұмыста «Denta+» стоматологиялық емханасының қызметінің автоматтандырылған жүйесі құрылып стоматологиялық емхана клиент және жұмысшылар арасындағы қызмет байланысы бір жүйеге келіп халыққа пайдалы қолайлы компьютерлік қызмет көрсету қатынасы жасалуы керек, процедуралар жүйесі құрылып, жасалған әрбір процедура дер кезінде компьютер арқылы тіркеуге алып стоматологиялық емхананың тауарды дер кезінде компьютер арқылы тіркеуге алып сауда үйінің күндік, айлық, жылдық табысын тез есептеп шығару, сол арқылы бухгалтер жұмысын жеңілдету, сонымен қатар компьютер арқылы стоматологиялық емханадағы кез-келген процедураны тез іздеп тауып, ол туралы егжей-тегжейлі ақпаратқа қол жеткізу мүмкіндігі қарастырылып, клиенттерге сапалы қызмет көрсету, бұл программа тауарларды тіркеп отыратын маманға арналып жасалынады.

Жүйенің дұрыс жұмысын қамтамасыз ету үшін оның жақсы ойластырылған, ұйымдастырылған және түсінікті мәзірлік, құрылымдық жүйесі болуы қажет.

1.3 Есептің қойылымы

Бұл дипломдық жұмыстың негізгі мақсаты – Алматы қаласында орналасқан «Denta+» стоматологиялық емханасының ақпараттық жүйесін құру. Стоматологиялық емхананың қызметін автоматтандырып клиенттер мен жұмысшыларға қолайлы және тез қызмет көрсететін жүйе жасау қажет.

Жүйеде келесідей функциялар орындалуы керек:

- кестелер туралы формалар құру және оны өзгерту, мәліметтер қосу, өшіру, сақтау мүмкіндіктері қарастырылуы керек;

- істелген жұмыс бойынша әр процедура туралы ақпарат, қажетті процедураның нақты анықтамасын көрсету, сол процедураға жауапты қызметкерлер туралы ақпарат алу мүмкіндігін қарастыру. Сонымен қатар жасалған процедураларды, клиенттерді тіркеп алу ортасын жасау, жаңа процедураларды базаға қосу сияқты жұмыс жүйесін қалыптастыру;

- анықтамаларда әрбір процедура туралы анықтама, қызметкерлер және дәрігерлер туралы анықтама, клиент туралы анықтама формалары құрылуы тиіс;

- есеп беруде әрбір клиенттің қабылдаған процедура санын, жалпы бағасын шығару жылдық ішкі табыс, жалпы процедура арасындағы процедура түрлерінің саны бойынша салыстыру жүргізіп диаграммалар құру, стоматологиялық емханада жұмыс істейтін адам санын есепке ала отырып диаграмма құрастыру, процедуралардың түрі бойынша сатылым мөлшерін салыстырып ең көп қажет ететін процедураның түрін анықтау, жалпы процедуралар санына байланысты есеп жүргізу керек;

- мәліметтер базасымен жұмыс істеуді ыңғайлы ету үшін, формалар және статистикалық мәліметтері бойынша есеп беруді құру;

- есеп берулерді экранға шығару.

«Denta+» стоматологиясындағы мәліметтерді сақтау ортасы ретінде мәліметтер базасын пайдаланады. Мәліметтер базасы операциялық жүйенің файлдарында ұйымдастырылған кез келген басқа ақпарат қоймасынан ерекшеленеді, МБ құрамына кіретін мәліметтерді мәліметтер базасын басқару жүйесі (МБЖ) деп аталатын арнайы бағдарламаны басқарады.

МҚБЖ нарығында өнімдердің көптүрлілігі арасында біздің АЖ-ға қойылатын талаптарға барынша толық SQL Server МҚБЖ сәйкес келеді.

МҚБЖ таңдау. Берілген тапсырманы іске асыру үшін желілік МҚБЖ ретінде Microsoft компаниясының SQL Server МҚБЖ таңдалған. Жоғарғы деңгей жүйесі MS SQL Server базасында құрылған. Таңдау сол МБЖ іске асыру үшін кіші ақталды, ең алдымен, экономикалық тұрғыда. Әрине, MS SQL Server өзінің бәсекелестеріне әлемдік нарықта ең қуатты және жан-жақты дамыған деп санауға болатын жиілікте Oracle МБЖ жиілікте өз бәсекелестерінен ұтады деп жоққа шығармауға болмайды. Тиісінше, Oracle құны біршама жоғары. Сонымен қатар MS SQL Server мәліметтер қорын әзірлеу мен қолдаудың толыққанды кәсіби жүйесі болып табылады, яғни оның бәсекеге қабілеттілігіне ешқандай күмән жоқ.

Осы бағдарламаны келесі әрекеттер арқылы оңтайландыруға болады:

- әлеуетті клиенттерден сұраныстары бар бағдарлама беттеріндегі мәтіндер тізімі талаптарға сәйкестігін бақылау;

- бұл қызметтерді сипаттайтын бірнеше негізгі комбинацияларды таңдау.

Түсті таңдау бағдарламаны жасау кезінде маңызды рөл атқарады. Кірушілер оны қабылдаудан, жалпы бағдарламада адамның психикалық және физиологиялық жай-күйі мен оқу қабілетіне, түсі көңіл-күйіне және көңіл-күйіне әсер етеді.

Сондықтан бұл бағдарламаны құру кезінде өте маңызды:

- бағдарлама негізінде жасалған фирмалық стильдің ерекшеліктері;

- түс спектрін қабылдау және физиологиялық тұлғаларға психологиялық мінездеме;

- монитор алдында ұзақ отырғанда, адам шаршайды, сондықтан бағдарламаның түс гаммасы ешқашан тітіркендірмеуі керек, керісінше, ол жеңіл болып, онда кедергі болмауы керек. Дәл осылай тек адамға қажетті ақпаратпен шатастырмайды. Бағдарлама пайдаланушыға ұнауы керек, ол жағымды әсер алуы керек;

- бағдарлама үлгісінің дизайны оның тақырыбына сәйкес келуі керек. Бағдарлама тақырыпқа сәйкес түспен қатар адамның психологиялық мінез-құлқының ажырамас бөлігі болып табылады. реакция мен тұрақты үйлесімге түс әсерінің негізгі заңдарының бұзылуына жол бермеу үшін түс үйлесімі сәйкес келуі керек. Әрине, түстерді қабылдау ол жалпы сыртқы жағдайына байланысты субъективті.

Пайдалануға және техникалық қызмет көрсетуге қойылатын талаптар жабдықты бұзу кезінде жүйені қалпына келтіру үшін ұйымдастыру қажет етпейтін жұмыс болып табылады.

Стоматологиялық емханадық жүйені автоматтандыруға арналған бағдарламалық кешендер таңдау кезінде келесі факторларды ескеру қажет:

- нысаны функционалды. Бағдарламалық қамтамасыз етуді пайдалану бірнеше функциялар мен әрекеттерді орындау қажет;

- жұмысты орындау. Тіркеуде автоматтандыру жүйесін жүргізу кезінде көптеген меліттер қоры құрылды, сондықтан жоғары сапалы ақпараттың көп бөлігі сізге көлемді тез өңдей алатын жүйе қажет. Бұл жағдайда бағдарлама және мәліметтер қорын басқару жүйесі жазбаша платформаға байланысты, себебі ол сипаттамалары жүйенің өнімділігін анықтайды;

- пайдалану оңай болуы. Бағдарламалық қамтамасыз ету жасақтамасы қаншалықты әдемі болса да адамдар оны пайдаланады екенін ұмытпаңыз. Бұдан басқа, есеп секторында жұмыс істейтін қызметкерлердің бағдарламалық қамтамасыз етуі болуы тиіс емес. Стоматологиялық емхананың қызметкерлері ол автоматтандырылған жүйемен жұмыс істеуді үйренгенше технологияларды қолданудан ешқандай оң нәтижелер мен пайда болмайды;

- тіркеудегі автоматтандыру үшін бағдарламалық қамтамасыз ету пайдалану және жүйелік әкімшілендіру үшін осы параметрді таңдағанда ерекше назар аудару керек;

- техникалық қолдаудың болуы. Әрбір бағдарламалық жүйе тұрақты жаңарту мен толықтыруды қажет етеді, ағымдағы жүйе жауап беруден бас тартылуы мүмкін және қойылатын талаптарға сәйкес келмеуі мүмкін. Осылайша, өзіңіз басқаруға мүмкіндік беретін бағдарламалық жасақтаманы таңдағанда нұсқаларға назар аудару керек. Кейбір өндірушілер бағдарламаларды қабылдауды бағдарламашыларға ұсынады, одан кейін сонда олар бар жүйені өз бетінше ұстай алады;

- кәсіпорынның мақсаттары мен міндеттеріне бейімделу қабілеті. Бұл мәселеде кейбір бағдарламалық қамтамасыз етуді басқару жүйелері міндеттеу, көп уақыт пен белгілі бір дағдыларды талап етеді. Көп сарапшылардың қымбат жобалары жетекші компаниялардың тәжірибесіне негізделген өз бизнес-сценарийлерін ұсынады;

- жобаның құны. Әрине, көптеген компаниялар үшін бұл фактор өте маңызды болады. Бөлшек бағдарламалық қамтамасыз етуді Автоматтандыру жүйесі өте әртүрлі. Жеке ерекшеліктерін ескере отырып авторлық жобалар, ол бағалануы мүмкін ондаған немесе жүздеген мың доллар. Рас, шағын егер дүкен немесе жергілікті бөлшек желі болса, онда әдеттегі өтінімдер бойынша болуы керек, ол да бар.

Кейбір авторлардың пікірінше, стоматологияны компьютерлеу міндеті төрт кезеңнен тұрады негізгі кезеңдерге бөлуге болады:

- қажетті компьютерлік жабдықтар мен жергілікті желі сатып алынып, орнату;

- компьютерлік бағдарламаларды енгізу;

- компьютерлік бағдарламалардың енгізілуі мен бағдарламаларды іске асыру, кәсіби дайындық;

- аппараттық - бағдарламалық кешенді қолдану.

2 СТОМАТОЛОГИЯЛЫҚ ЕМХАНАНЫҢ БАҒДАРЛАМАЛЫҚ ҚАМТАМАСЫН ҚҰРУ

2.1 Мәліметтер қорына жалпы шолу

МҚ үшін маңызды қосымшалардың бірі – адам функцияларының әртүрлілігінде қолданылады. Мысалы: банк, көлік, ғылым, сауда, медицина, экономика және т. б.

Мәліметтерді сақтау және беруге арналған қазіргі заманғы ақпараттық жүйелер үлкен көлемді өңдеу, пайдаланушылардың әр түрлі талаптары бар қажеттіліктерді қанағаттандыру қажеттілігімен және күрделі ұйыммен сипатталады.

Мәліметтерді жинауға арналған автоматтандырылған ақпараттық жүйелер, манипуляциялау және өңдеу, бағдарламалық қамтамасыз ету және қызметкерлердің қолдауын, сондай-ақ мәліметтерді өңдеуді қамтамасыз ететін техникалық құралдар кіреді.

Бұл сөздің ең кең мағынасында мәліметтер базасы кез келген пәндік салада шынайы әлемдік объектілер туралы ақпарат компендиумы.

Мәліметтер қорын жасау кезінде пайдаланушы белгілі бір сипаттамаларды көрсетеді, ақпаратты ұйымдастыру және сипаттамалардың кез келген үйлесімі, тез таңдауды қалайды. Сондықтан дұрыс мәліметтер үлгісін таңдаңыз. Мәліметтер моделі - оның объектілері, байланыстары, қасиеттері, сондай-ақ олардың нақты әлеммен өзара әрекеттесуі негізгі категориялардың ресми өкілдігі.

Мәліметтер қоры – бірдей қасиеттері бар нысандар тобы ұйымның қызметі туралы ақпаратты сақтай алатын мүмкін болатын ақпараттық модель.

Мәліметтер базасындағы мәліметтер реттелген тәртіпте сақталады.

2.2 Мәліметтер қорын басқару жүйесі және оның негізгі функциялары

МББЖ – бірнеше пайдаланушының мәліметтер базасын енгізу, жасау және жүргізіп жаңартылулармен бағдарламалық қамтамасыздандыру және ортақ пайдаланудың тіл саймандар жинағы.

МББЖ келесі функцияларды орындайды.

Қойманы басқару. Бұл функция пайдаланушыларға ақпаратты сақтауға мүмкіндік беріп, жаңарту және жою сияқты мәліметтердегі операцияларды орындау мүмкіндігін береді. Сыртқы жадтың қажетті құрылымы офистік мақсаттарда да қолданылады, ол сондай-ақ мәліметтер базасына байланысты мәліметтерді сақтау үшін қолданылады.

Жад буферлерін басқару. Мәліметтер қорын басқару жүйесі мәліметтер базасымен жұмыс істейді. Мәліметтер элементіне кіргенде, ол сыртқы көрініс үшін қол жетімді болады, сыртқы жад құрылысының барлық жүйесі ауыстырылып, мәліметтерді беру жылдамдығымен жұмыс істейді. Буферизация–

жылдамдықты арттырудың басты жолы. Алдыңғы қатарлы мәліметтер қорын басқару жүйелері аралық буферлер ауыстырғыш буфердің көмегімен өз жинақтарын қолдайды. Аралықтарды басқару, тізбекті тіркеу, синхрондау және буферлендіру алгоритмдері оны пайдалану және дамыту қажет. МББЖ барлық мәліметтер базасын қамтитын жеке бағытқа ие тұрақты негізде. Бұл бағыт болашақта компьютерлік жадыға, мысалы, ол буферизация туралы өте үлкен болуы мүмкін бұл сізге алаңдамауға мүмкіндік береді. Бірақ бұл жұмыстар әлі де зертеу сатысында тұр.

Мәліметтер қорын басқару. Әрбір мәліметтерді басқару жүйесі бағдарлама кодтауыдың болмауын, күрделі қатынастары бар мәліметтерді, олардың жүйелік сөздігін, сондай-ақ қарымқатынасына көмектесетін қажетті мәліметтерді іздеу және құрылымдау үшін каталогты пайдаланады, өйткені кез келген бағдарламалар мәліметтер базасын басқарады жүйелік мәліметтерге қол жеткізу арқылы жүзеге асырылады.

Мәліметтер сөздігі мәліметтер базасындағы мәліметтерді сипаттайтын ақпаратты қамтиды. Каталогтар, МҚБЖ пайдаланушылары ол сондай-ақ өз функциялары үшін қол жетімді болып саналады.

Мәліметтер сөздігі келесі ақпаратты қамтиды: сілтемелердің атаулары, тұтастығы, мәліметтер элементтерінің өлшемдерін, аттарын және түрлерін сақтау, осы мәліметтерді пайдалануға рұқсат етілген пайдаланушыларға шектеу енгізумен, атаулар, сыртқы, тұжырымдамалық және ішкі схемалар және олардың мәндерін көрсету, статистика.

Мәліметтер қорын басқару жүйесі мәліметтер абстракциясын қамтамасыз етеді осылайша, мәліметтерге тәуелділік және жүйедегі құрылымдық тәуелділікті жояды.

Транзакцияларды басқару. Транзакция – бұл мәліметтер базасындағы операциялар тізбегі бұл дегеніміз, ол мәліметтер қорын басқару жүйесін біртұтас ретінде қарастырады. Сыртқы жадта мәліметтер қорын басқару жүйесімен құрылған мәліметтер базасын өзгерту немесе мәліметтер қорының күйіндегі осы өзгерістердің бірі жоқ. Мәліметтер базасының логикалық тұтастығын алу үшін транзакциялар қажеттілігі.

Кез келген нысан туралы мәліметтер базасындағы қарапайым транзакциялар ақпаратты жаңарту, қосу немесе жою үшін оны пайдаланыңыз. Күрделі операция дереу мәліметтер базасына бірнеше өзгерістер енгізу қажет болған кезде орындалады. Транзакцияны орнату бағдарлама немесе жеке пайдаланушы арқылы шақырылады.

Барлық транзакциялар мәліметтер қорының толық күйінен басталады және осы жағдай аяқталғаннан кейін мәліметтер қорына қатысты бәрін қалдырыңыз пайдаланушы әрекеті ретінде транзакция ұғымын пайдалану өте қолайлы.

Мәліметтер қорын басқару жүйесіне қойылатын негізгі талаптардың бірі сыртқы көрініс болып табылады жадтағы ақпараттың тұтастығы. Мәліметтер қорын басқару жүйедегі бағдарламалық немесе аппараттық қамтамасыз ету зақымдалғаннан кейін мәліметтер базасының соңғы дәйекті күйін қалпына келтіру мүмкіндігі сақтау кезінде бұл сенімді.

Әдетте біз аппараттық ақаулардың екі түрін қарастырамыз: сыртқы жад құрылғысында мәліметтерді жоғалту және жұмсақ ақаулар сипатталған қатты қателермен – компьютерді күтпеген тоқтауына әкеледі.

Сенімді мәліметтерді сақтау және мәліметтер базасында артық мәліметтерді сақтау бұл қалпына келтіру үшін қажет, бірақ қауіпсіз сақтау дұрыс.

Қажетті ақпаратты сақтаудың ең көп таралған тәсілі мәліметтер базасын өзгертуге мүмкіндік береді, тіркеуі болса.

Журнал – мәліметтер қорын басқару жүйесінің пайдаланушыларына қол жеткізу жоқ мәліметтердің арнайы бөлігі, мәліметтердің басындағы өзгерістер олардың ұқыпты сақталғаны туралы жазбалар.

Қауіпсіздікті басқару. Мәліметтер қорын басқару жүйесі қауіпсіздік жүйесі олар пайдаланушыларды қалыптастырады және мәліметтер базасында ақпаратты қорғайды құпиялылықты қамтамасыз ету.

Қауіпсіздік ережелері пайдаланушыға мәліметтерге қолжетімді операциялар және қандай элементтері пайдаланушыға қол жетімді жүйелерден артық болуы мүмкін, бұл функция бірнеше пайдаланушыға бір уақытта мәліметтерге қол жеткізуге мүмкіндік береді.

Тілдік мәліметтер базасын басқару. Мәліметтер базасымен жұмыс істеу үшін мәліметтер қорының тілдері деп аталатын арнайы тілдер қолданылады. Мәліметтер қоры басқару жүйелері бірыңғай интеграцияланған тілді қолдайды, ол мәліметтер базасымен жұмыс істеу үшін барлық қажетті құралдар бар және мәліметтерді қамтитын интерфейс, негізгі мәліметтер базасын ұсынады. Қазіргі уақытта стандартты тіл - SQL тілі. Ол екі негізгі компоненттен тұрады: мәліметтерді манипуляциялау және мәліметтерді сипаттау. SQL мәліметтерін басқару және салыстырмалы мәліметтер қорының сызбасын анықтауға мүмкіндік береді.

2.3 МҚБЖ құрылымы, артықшылығы мен кемшілігі

МББЖ пайдалану ыңғайлылығы және оңай дамып келе жатқан.

Жүйелік бағдарламалық қамтамасыз етудің МББЖ – файлдық жүйелері мүмкіндіктерін кеңейтетін жүйелік функцияларды орындау тапсырмаларды орындауға арналған бағдарламалық қамтамасыз ету.

МББЖ келесі функцияларды орындайды:

- мәліметтерді ұйымдастыру үлгілері (иерархиялық, желілік, реляциялық) үлгілермен қамтамасыз ететін мәліметтердің логикалық құрылымын қолдау мен ұйымдастық;

- сыртқы жадыдағы мәліметтердің физикалық құрылымын ұйымдастыру мәліметтер қоры файлдарының ішкі құрылымын қоса алғанда, мәліметтер қоры сақтау және ұйымдастыру, сондай-ақ мәліметтерге жүйелі және тиімді қол жеткізу жеткізу үшін арнайы құрылымдарды (индекстер, беттер) сақтау және құру;

Мәліметтер қоры құрылымының сипаттамасы және процессорды қолдау. Бұл процессор мәліметтерді ұйымдастыру моделін іске асырады, ал конструктор оны тәуелсіз іске асырады тақырыптық схемаға сәйкес келетін қаражат есебінен инфрақұрылым мәліметтер базасының логикалық құрылымын, ішкі мәліметтер базасын жасайды бұл схема жүйені құруға және жұмыс жағдайында ұстап тұруға кепілдік береді. Мәліметтер процессоры мәліметтер қорының құрылымы және мәліметтер қорының құрылымы мәліметтер базасының ішкі схемасы (физикалық мәліметтер) конструкциялары) қамтамасыз етеді, монтаждау.

Мәліметтер қорының сұраныстарын өңдеуші. Процессор сұраулардан туындайтын мәліметтерді пайдаланады өңдеу тілінде және процессормен бірге мәліметтер базасының құрылымын түсіндіреді, сұрауларды сипаттайды, сақтайды және орындайды.

МББЖ кіріс интерфейсі. Оның функциясы – ақпараттық жүйеге ақпаратты енгізу тілін іске асыру, ақпараттық жүйелерге және ақпаратты сипаттау құралдарына ақпаратты енгізу.

Сұраныс интерфейсі. Пайдаланушы туралы ақпаратты сұрау интерфейсі сіздің қажеттіліктеріңізді көрсететін стандартты сұрау салуларда бұл пайдалану үшін тұжырымдамалық модель.

Транзакциялар мониторингі функциялары – жалпы мәліметтер ортақ желіде транзакцияларды орындайтын пайдаланушы мәліметтердің бүтіндігін ұйымдастыру және қамтамасыз ету.

Интерфейстің шығуы туралы ақпарат. Процессор сұрауын сұрау орындау нәтижесі ақпарат беру интерфейсіне жіберіледі, ол пайдаланушыға нәтижелерді беру және қабылдау үшін ыңғайлы нысан тізімдер.

Бухгалтерлік генератор. Есептерді қалыптастыруға арналған арнайы шығарылым нысандар толтыруға жатады. Мәліметтер базасынан мәтіндік құжаттарды пішімдеу үшін есептер ақпаратты таңдау және іздеу нәтижелерін ұсынады.

Қазіргі заманғы МҚБЖ функционалдық компоненттері бар:

- экранда мәліметтер қоры файлдарының құрылымын өңдеу және құру мәліметтерді көрсету және жаңарту;

- жаңартылған мәліметтер мен үлгілерді құру және сұрау;

- енгізу-шығару экрандық формаларын әзірлеу;

Ақпаратты ұсыну және басып шығару үшін бланкілерді дайындау:

- есептілік нысандарын әзірлеу және жариялау;

- қосымшаларды әзірлеу.

Файлдық жүйелерімен және МББЖ-мен салыстырғанда артықшылықтар бар.

Артықшылықтары:

- мәліметтердің артық болуын бақылау;

- тізімдегі мәліметтер тізбегі;

- осы бөлімде өз мәліметтеріңізді бөлісіңіз;

- мәліметтердің бүтіндігін сақтау;

- қауіпсіздікті арттыру;

- конфигурацияда стандарттарды қолдану;
- жүйе ауқымының ұлғаюына қарай тиімділік мәні;
- қарама-қайшы талаптармен ымыраға келу мүмкіндігі;
- мәліметтердің қолжетімділігі мен қолжетімділігін арттыру;
- өнімділікті жақсарту;
- мәліметтер тәуелсіздігі есебінен жүйеге қызмет көрсетуді жеңілдету;
- жақсартылған жылдамдықты басқару;
- бұл кеңейтілген сақтық көшірме және сақтық көшірме қызметі.

Кемшіліктер:

- күрделілік;
- бағдарламалық қамтамасыз ету өлшемі;
- МББЖ құны;
- жабдыққа арналған қосымша шығыстар;
- өтінімді өңдеуге арналған шығындар;
- өнімділік.

2.4 Мәліметтер базасының физикалық моделін әзірлеу

Әрбір мәліметтер қоры бірінші кезекте белгілі бір тақырып саласына арналған мәліметтерді көрсетуге арналған. Кез келген аумаққа қатысты тақырып пішіндер туралы ақпаратты көрсетеді. Әр объектінің күрделі құрылымы және белгілі бір қасиеттермен сипатталуы мүмкін, бұл қасиеттер сандық болуы мүмкін осы субъектінің басқа объектілерімен, яғни осы субъектінің басқа объектілерімен қарым-қатынас сипаты болып табылады. Бұл қасиеттердің барлығы әдетте мәліметтер базасында объект бар бұл анықтаушы атрибуттар.

Мәліметтер базасында көрсетілген барлық мәліметтердің құрылымы бұл маңызды элемент болуы керек, өйткені ол пәндік аймақты көрсетеді. Барлық мәліметтер қасиеттерінің құрылымы арасында қисынды байланыс. Бұл құрылым тұжырымдамалық схема, жалпы схема немесе схеманы атады.

МБ – белгілі бір кезеңге енгізілген нақты объектілер туралы барлық мәліметтер физикалық мәліметтер түрінде сақталады.

Мәліметтерді тиімді өңдеу үшін бастапқы бірнеше схема аз бөліктерге бөлінген. Әрбір бөлік түпнұсқа схеманың белгілі бір мағынасы бар қамтиды. Осы бөліктердегі жекелеген ішкі схемалар арасындағы байланыс үшін сол өрістерді (аты мен мәліметтері бірдей) пайдалану қажет.

Мәліметтер құрылымы формальды түрде сипатталуы тиіс. Логикалық және мәліметтер базасы бар мәліметтер базасының физикалық құрылымдарының сипаттамасы ақпарат алу және дерекқорларды басқару талаптарын редакциялау бағдарламалық қамтамасыз ету үшін пайдаланылады. Мәліметтер қоры жалпы логикалық құрылымның сипаттамасы сызба деп аталады. Бұл кейде болады мәліметтердің жалпы моделі, тұжырымдамалық модель немесе тұжырымдамалық схема деп аталатын. Схема - бұл қолданылатын мәліметтер түрлерінің кестесі. Ол нысандар аттары мен олардың атрибуттары, сондай-ақ

қарым-қатынастар, өзіңіз анықтаңыз. Мәліметтер элементтерінің сұлбасы мәндер орналасуы мүмкін құрылым. Диаграммалар мен жазбалар. Суреттер әрбір мәліметтер элементінің барлық мәнін анықтайды. Мерзімі мәліметтер қорында сақталған барлық мәліметтер элементтері мен жазбалары бағдарламада толық қарау кестесін анықтау үшін пайдаланылатын мәліметтердің сипаттамасын анықтайды. Көптеген түрлі ішкі схемалар бір схемаға негізделуі мүмкін.

Мәліметтер қорының қолданбалы бағдарламасы немесе пайдаланушылық схемасы туралы толық білім міндетті емес, өйткені ол әдетте өте күрделі. Кейде мұндай ақпараттың болмауы қауіпсіздікке байланысты болады. Бағдарламалаушы немесе пайдаланушы қажетті арнайы бағдарламаларды және сіз тек жазбаларды жасау керек.

Мәліметтер қорын басқару бағдарламалары схемада сипатталған мәліметтерге сәйкес келеді қосымша мәліметтер негізінде тиісті сұрау автоматты түрде өңделеді басып шығарыңыз, бағдарламаңыз, бағдарламаға жіберіңіз. Схемалар және ішкі схемалар мәліметтерді сақтаудың физикалық әдістерін көрсетпейді. Көрсетілген логикалық құрылым үшін мүмкін, мәліметтерді физикалық ұйымдастырудың түрлі нысандары бар сіз көрсете аласыз. Осылайша, үш түрлі мәліметтер сипаттамалары бар:

Қосымша дискілер - бағдарлама қажеттіліктеріне арналған бір немесе бірнеше бағдарлама жіберілген мәліметтердің бөлігін сипаттайтын кесте. Бағдарламаларда пайдаланушы сұраулары.

Мәліметтер қорының немесе схеманың логикалық құрылымының құрылымы сипаттама-барлық мәліметтер базасын логикалық сипаттайтын кесте. Бұл мәліметтер әкімшісі жүйесі немесе барлық мәліметтер базасымен жұмыс істейтін та талдаушылардың мәліметтері.

Физикалық ұйымның сипаттамасы мәліметтер базасы – қоймалар мәліметтерді бұқаралық ақпарат құралдарында физикалық орналастыру кестесі.

Бұл мәліметтер идеясы жүйенің өнімділігі, медиа мәліметтер орналасу, индекстеу немесе іздеу, сондай-ақ мәліметтерді қысу әдістері қолдануға байланысты жүйелерді әзірлеу үшін

Реляциялық мәліметтер үлгілерінде кестелер арнайы стандарт бойынша жасалады туыстық: басты және бағынышты. Анықтама үшін, байланысты кесте байланыстырылған бірдей бағандар болуы керек. Сол кесте олардың біреуіне де қолданылады баған және басқа кесте бағанға байланысты бағанға бағынуы мүмкін.

Мәліметтерді кестелік көрсету тек мәліметтер логикасына жатады, физикалық олар басқа принциптерге сәйкес орналастырылуы мүмкін.

Кестеге сәйкес жазбаларды бірдей сәйкестендіретін мәліметтер базасының әрбір кестесінде. бір немесе бірнеше өрістерді көрсететін бастапқы кілттер болуы керек. Кестелер бастапқы кілттің мәні бірегей болуы керек.

Кейде бастапқы кілт ретінде арнайы құрылған өріс индексі деп аталады автоматты түрде жасалған нәрсені таңдаңыз, әдетте бұл бір реттік жазба сан сан болып саналады.

Бастапқы кілт арнайы жазбаға қол жеткізуге мүмкіндік береді (ол шынайы жазбаны анықтайды). Сонымен қатар, бастапқы кілт кестелер олардың арасында байланыс орнату үшін пайдаланылады.

Қосымша кілттер мәліметтерді іздеу үшін пайдаланылады өрістер жиі мәліметтерді іздейтін атрибуттар үшін қойылады.

Қосалқы кілттердің мәні бірегей емес және анық жазба болмауы мүмкін. бірегей анықтау үшін пайдаланылмайды. Олар екінші кілттің мәні бар жазбаларды іздеудің арнайы шарттарын қанағаттандыратын жазбаларды іздеу ол үшін қолданылады.

МББЖ қолдану сипаты бойынша дербес, бірнеше және арнайы бөліктерге бөлінген.

Жергілікті компьютерде жұмыс істейтін жеке МҚБЖ, мәліметтер базасын жасауға мүмкіндік береді. Paradox, dBase, FoxPro және т. б. ол жеке МҚБЖ тағайындалуы мүмкін.

Архитектурада жұмыс істейтін ақпараттық жүйе МББЖ клиент-сервері жүйе мүмкіндік береді. Oracle, Informix, SyBase, Interbase, DB2 МББЖ бірнеше пайдаланушыларды тасымалдауға болады.

Мәліметтерді өңдеу технологиясына арналған орталықтандырылған МББЖ және ыдырады.

Бір компьютерлік жүйенің жадындағы орталықтандырылған мәліметтер базасы сақталған. Егер бұл компьютерлік жүйе компьютерлік желінің бөлігі болса егер бар болса, онда мұндай мәліметтер базасы, яғни осы желіде түрлі компьютерлер үшін. пайдаланушылардың қол жетімділігі.

Компьютерлік желідегі түрлі компьютерлерде бөлінген мәліметтер базасы бірнеше сақталған, мүмкін, жабылатын немесе тіпті қайталанатын ол бөліктерден тұрады. Мұндай мәліметтер базасымен жұмыс. мәліметтер қорын басқарудың үлестірілген жүйелері.

Қазіргі тілдік құралдардың құрамына МББЖ кіреді:

- логикалық мәліметтердің құрылымын сипаттау үшін мәліметтер сипаттама тілі;

- мәліметтерді жазу, редакциялау, іріктеу бойынша негізгі операцияларды іске асыру мәліметтерді манипуляциялау тілі,;

- мәліметтер қорының құрылымын басқару және мәліметтерді басқару, бұл сондай-ақ қашықтағы мәліметтер қорына қол жеткізу үшін стандартты құрал. құрылымдық сұрау тілі (SQL structured query language);

- сұрау үлгісі үлгі (мысалы, QBE сұрау), мәліметтер қоры сұраныстардың визуалды дизайнын қамтамасыз етеді.

Қолданбалы бағдарламалар немесе мәліметтер базасындағы бағдарламалар мәліметтерді өңдеу үшін қолданылады. Пайдаланушының мәліметтер базасын басқару мәліметтер қорының бағдарламалары деп аталатын қосымшаларды пайдалану мәліметтермен жұмыс істейді.

Желілік хабарламасы бар орталықтандырылған мәліметтер базасы ұқсас ол әртүрлі жүйелік архитектура болып табылады:

- файлдық сервер;

- клиент сервері.

Файлдық сервер. Сервер ретінде бір компьютер желісін таңдаңыз. Пайдаланушы жергілікті мәліметтер қорының көшірмесімен жұмыс істейді мәліметтер әр кестемен кез келген кестеге жаңартылады. Бұл жағдайда серверден мәліметтерді сұрайтын барлық кестенің жаңа көшірмесі жіберіледі. Осылайша, егер пайдаланушы бірнеше кесте жазбасын талап етсе, онда бүкіл кесте толығымен серверден желі бойынша жіберілді. Көптеген қажетсіз ақпараттық желілерде айналымның нәтижесінде желідегі жүктеме күрт өседі, бұл жалпы ақпараттық жүйе жұмысының жылдамдығы мен өнімділігі тәуекелдерді азайту.

Клиент сервері. Осы сәулетте мәліметтерді сақтауға қосымша, орталықтандырылған мәліметтер базасы тек пайдаланушылардың сұраныстарын өңдей алады пайдаланушы сұралған мәліметтерді жіберуі керек. Бұл желілік жүктеме ол қажетті ақпараттың желі арқылы өтуіне қарай азаяды.

Мәліметтерді өңдеу жүйелерінде ең жақсы мәліметтерді басқару әдісін іздеу негізгі мәселелерді шешудің бірнеше жолы бөлінді. Зерттеу нәтижесі дерекқорлардың түрлі үлгілерін жасау болды.

Мәліметтер қорының үлгісі мәліметтер қорының құрылымы мен құрылымын қамтиды. арасындағы өзара байланысты көрсету үшін жинағы конструкциялар. Іріктемелі мәліметтер базасын екі санатқа бөлу онда: тұжырымдамалық модельдер мен іске асыру үлгілері ұсынылады.

Мәліметтерді ұсынудың тұжырымдамалық моделі логикалық кейіпкерге назар аударды. Осылайша, тұжырымдамалық модель мәліметтер базасында ұсынылған емес, оның қалай көрсетілгеніне назар аударыңыз тізімдер. Концептуалды модельдегі субъект – қатынастар моделі (ER-модель). Объектілі-бағытталған модель бар.

Модель концепциясынан айырмашылығы, іске асыру үлгісі мәліметтер болып табылады мәліметтер базасы негізінде мәліметтерді ұсынуды немесе құрылымдауды моделдеу олар мұны қалай туралы түсінік алу үшін. Мәліметтер қоры іске асыру модельдері иерархиялық модель, желілік модель, реляциялық модель модель мен объектілі-бағытталған модельді қамтиды.

Мәліметтер қорын жобалау үшін негіз ретінде мәліметтер қорын әзірлеушілер тұжырымдамалық модель қабылдайды. Мәліметтердің тұжырымдамалық модельдері үш түрлі қатынастар арасындағы байланысты сипаттау үшін пайдасы: бір-бірінен артық, көп және бір-бірінен артық. Мәліметтер қоры дизайнерлер әдетте осы мақсатқа арналған аббревиатуралар: 1: M, M: N және 1:1

Тәжірибе көрсеткендей, әрбір мәліметтер үлгісінің құрылымы мәліметтер қорын басқарудың заманауи жүйелерін жобалаудың негізгі концепцияларын зерттеу әдістері мен мәселелері. Шын мәнінде, көптеген «жаңа» мәліметтер қорының тұжырымдамалары кейбір «ескі» идеялар және құрылыстар. Мысалы, кейбір иерархиялық мәліметтер базасының үлгілері тұжырымдамалар мен объектілі-бағытталған құрылымды іске асыру арасында күмәнсіз, сіз ұқсастықты таба аласыз.

Қазіргі заманғы ақпарат тиімді ақпарат жоқ. Нақты қоршаған ортаны қабылдау алуан түрлі, бірақ кейде өзара байланысты құбылыстармен қатар мүмкін. Адамдар бұл құбылыстарды бұрыннан сипаттайды Мен тырыстым (тіпті түсінбеймін). Бұл сипаттама мәліметтер деп аталады. Бүгін күні бағдарламалық қамтамасыз етудің маңызды санаты мәліметтер базасына негізделген ақпаратты өңдеу жүйесі. Негізінде, ақпараттық жүйенің мәліметтерінен немесе тиімділігіне қарай, мекеменің жұмыс. Мұндай жүйе: жалпы және немесе толық орындау есептерді алуды қамтамасыз етеді. индикаторлар бойынша трендтерді оңай анықтауға мүмкіндік береді; уақыт өте келе маңызды, маңызды ақпарат кідіріссіз алынуын қамтамасыз ету.

Бірнеше пайдаланушылар (бағдарламалық бағдарламалар)бір уақытта ұқсас мәліметтерді сұрау кезінде туындайтын мәселелер;

Қате жаңартулар және (немесе) рұқсат етілмеген қол жеткізу мәліметтері қауіпсіздікті қамтамасыз ету тәсілдері;

MySQL мәліметтерді енгізу үшін мәліметтер қоры ретінде қолданылады. MySQL – жоғарыда аталған функцияларды орындау үшін барлық қажетті құралдары бар контактілердің мәліметтер базасының үстелдік жүйесі. MySQL мәліметтер қорын басқару жүйе жұмыс үстеліндегі МББЖ тобының ең танымал қосымшаларының бірі болып табылады. MySQL барлық нұсқаларына мәліметтерді енгізу және өңдеу, мәліметтерді өңдеу кіреді, кестелер, кестелер және есептер түрінде ақпаратты іздеу және ұсыну бұл әлдеқайда оңай болатын құралдардың толық арсеналы бар. MySQL нұсқасынан пайдаланушыға көрінетін Internet Explorer браузерінен бастап бұл мәліметтер веб-беттерді қамтиды.

Сонымен қатар, MySQL - қосымша кестелер мен кестелермен басқа жұмыс үстел және Серверлік мәліметтер қорынан қолданба үшін қажетті ақпаратты сақтау үшін оны пайдалануға мүмкіндік береді. MySQL сыртқы кестелерді қосу арқылы MySQL кестелері сияқты осы кестелердегі мәліметтер қоры бар бұл жұмыс істейді. Сонымен қатар, осы мәліметтермен басқа пайдаланушылар жұмыс істей алады сіз мұны жалғастыра аласыз, олар жасалды. MySQL Артықшылықтары. қарапайым және күрделі қосымшаларды әзірлеу кезінде ол қарапайым графикалық интерфейс бар. Тағы бір үстел Сонымен қатар, MySQL барлық мәліметтерді бір файлда сақтайды, бірақ бұл олай емес. олар әртүрлі кестелерге бөлінген.

Ғылыми зерттеу саласы – басқаруды ұйымдастыру және сайып келгенде автоматтандыру мақсатында зерттелетін нақты әлемнің бір бөлігі. Сабәк олар фрагменттер жиынтығымен ұсынылған, мысалы, бизнес-семинарлар, басқарма, бухгалтерлік есеп және аудит ол әртүрлі нысандармен және процестермен сипатталады. тақырып аясындағы түрлі дисплейлер бар пайдаланушылар барлық. Дипломдық жұмыста стоматологиялық емхананың мәліметтер базасын құру міндеті. Сондықтан осы тақырып бойынша оны іске асыру барысында зерттеу жүргізілді.

Бұл дипломдық жұмыс медицина саласындағы мәліметтер базасы болып табылады құру және даму үдерісі жүріп жатыр. Демек, қорытынды мәліметтер базасы ол осы тақырып саласындағы барлық бизнес - логиканы көрсетпейді, тек

нақты мәліметтер үлгісі ретінде "стоматология" пән аймағының осы бөлігімен шектелмей мұны сипаттауға болады. Мәліметтерді сақтау ортасында қолданылатын мәліметтер үлгісі. толық тәуелсіз адам кейбір параметрлер бойынша Тирумалға түсті нақты әлемді көрсетеді. Мұндай модельдерді құрудың көптеген жолдары бар: графикалық модельдер, семантикалық желілер, субъективті қатынастар модельдері және т. б. зерттеу тақырыбын сипаттау үшін субъект – қатынас нысанының мәліметтерін ұсыну қажет. Ұйым басқа объектілерден өзгеше нәрсе бұл нысан анықталуы мүмкін.

Әрекет – бірнеше субъектілер арасында орнатылатын қауымдастық. N (және $n \geq 2$ ден кем емес), олардың әрқайсысы белгілі бір жиынтыққа жатады. $N = 2$ жағдайда, яғни, байланыс екілік деп аталады, ол екі бірлікті қосады. Әрине, N -Ари әрқашанда екілік сандардың жиынтығы бар қатынастар жиыны ($n > 2$) ауыстырылуы мүмкін, бірақ олардың біріншісі пән аймағының семантикасы үшін жақсы. Басқа субъектімен байланыс жиынтығы арқылы байланысты болуы мүмкін субъектілер саны байланыс дәрежесі деп аталады.

Деңгейлерді көру, әсіресе екілік қосылыстар үшін пайдалы. Екілік қарым-қатынастар мынадай дәрежелерге ие болуы мүмкін: бір (1: 1) белгіленген).

Бұл мұндай қатынастарда бір рөлі бар субъект әрқашан басқа рөлдердің бірінен артық емес. Біреуден артық (1: n). Бұл жағдайда бір рөлі бар субъектілер басқа рөлге ие кез келген дәрежеге сәйкес келуі мүмкін. Көп (n : 1). Бұл қатынасы 1: n картасына ұқсас. Көп адамдар (n : n). Бұл жағдайда байланысты мәндер олардың әрқайсысы кез келген даналармен ұсынылуы мүмкін.

Параметрлермен сұраулар. Параметрлермен сұрау – ол орындалғанда, оның диалогтық терезесінде мәліметтерді енгізу үшін шақыруды көрсетеді. Бірнеше мәліметтер бірлігін енгізу сұрауын жасауға болады.

Сұраныстарды талдауды жеңілдететін құрылымдағы мәліметтерді есепке алу және талдау көрсету үшін пайдаланылады. Айқас сұраулар сомасы, орташа мәндер, мәндер санын есептейді немесе басқа статистикалық есептеулерді есептейді осыдан кейін нәтижелер екі кесте бойынша топтастырылады, олар біреуі баған тақырыптарын, ал екіншісі – жол тақырыптарын анықтайды.

Өзгерістер енгізуге сұрау салу.

Өзгертуді сұрау – бір әрекет үшін бірнеше жазбаларды өзгерткен немесе өзгерткен құжат ауыстыруға сұраныс. Өзгертуге сұраудың жазбаны жою үшін, жазбаны жаңарту кезінде, мәтін жазбаларын қосу. кестені құру деген төрт түрі бар.

Мәліметтер қоры жергілікті бағдарламаға (клиент) бөлінеді. бағдарлама іске қосылған компьютерде) және алыстан (серверде) қашықтан компьютерде орнатылады). Мәліметтер базасының екінші түрін көрсетіңіз бұл мақсат үшін клиент-сервер немесе желілік мәліметтер қорының аты қолданылады, өйткені мәліметтерді өңдеу бағдарламасы клиентте орналасқан және мәліметтер серверінде орналасқан. Егер клиент-серверлік мәліметтер базасы болса, мәліметтер серверде сақталады. Клиенттік қосымша арнайы SQL тілінде немесе оның тілінде мәліметтерді қамтиды. Кеңейтілімінде – Transact-SQL немесе PL / SQL) мәліметтерді пайдалану қол жеткізу. Бағдарлама тек қажетті мәліметтерді

алып, серверде өзгертілген мәліметтерді жүктеу. Желілік мәліметтер базасы серверде сақталады, бірақ олардың әрқайсысы клиент осы мәліметтердің жеке көшірмесін алады. Бұл жағдайда екі негізгі кемшіліктер де бар: желіге үлкен жүктеме және тұтастықты қамтамасыз ету мәліметтерді ұйымдастыру түріне байланысты кешенді процедура мәліметтер базасында мәліметтерді ұсынудың келесі иерархиялық, желі, коммуникация, объективті бағытталған негізгі нысандары көрсетілген.

Қазіргі уақытта ең кең тараған реляциялық мәліметтер қоры, мәліметтер кестелерде жинақталады, олар өз кезегінде Графиктер мен жолдарды қамтиды. қиылысатын ұяшықтардан тұрады. Мұндай дерекқорларға сұрау салу келесі сұрауда қайтадан қатыса алатын кестені қайтарады. Кейбір кестелерде мәліметтер әдетте басқа кестелердегі мәліметтермен байланысты, Алын реляционный аты шыққан. Реляциялық мәліметтер қорының ерекшеліктері сіз қысқаша мәлімдей аласыз:

- дера бағандар мен жолдардан тұрады;
- әрбір графика мен жолдардың қиылысында ол тек бір мағынаға ие;
- әрбір баған өз аты бар және бір бағанда барлығы бар мәндер бірдей көрініс;
- бағандар кездейсоқ тәртіппен орналасқан кестені жасау кезінде жолдан қарағанда берілген белгілі бір тәртіппен.

Кестеде бір жол емес, кем дегенде бір баған болуы керек мәліметтер базасына сұраныс нәтижелері кестесінің нысаны олар сондай-ақ сұрау салу объектісі ретінде әрекет ете алады. Бұл жағдайда, мәліметтер базасы бір кестеден тұруы мүмкін, бірақ әдеттегі мәліметтер базасы өзара байланысты үстел жиынтығынан тұрады. Кестелер арасындағы тұлға аралық қатынастар ұйым кестені байланыстыру деп аталады. Сілтемелердің өрістері кестелерді байланыстыру үшін қолданылады (кейде «сәйкес өрістер» термині қолданылады). Өрістер индекстелуі керек. Ұсынымдар келесі түрлері бар:

- бір-бірімен қарым-қатынас;
- бір-көптеген қатынастар;
- көптеген қатынастар;
- көп қарым-қатынас.

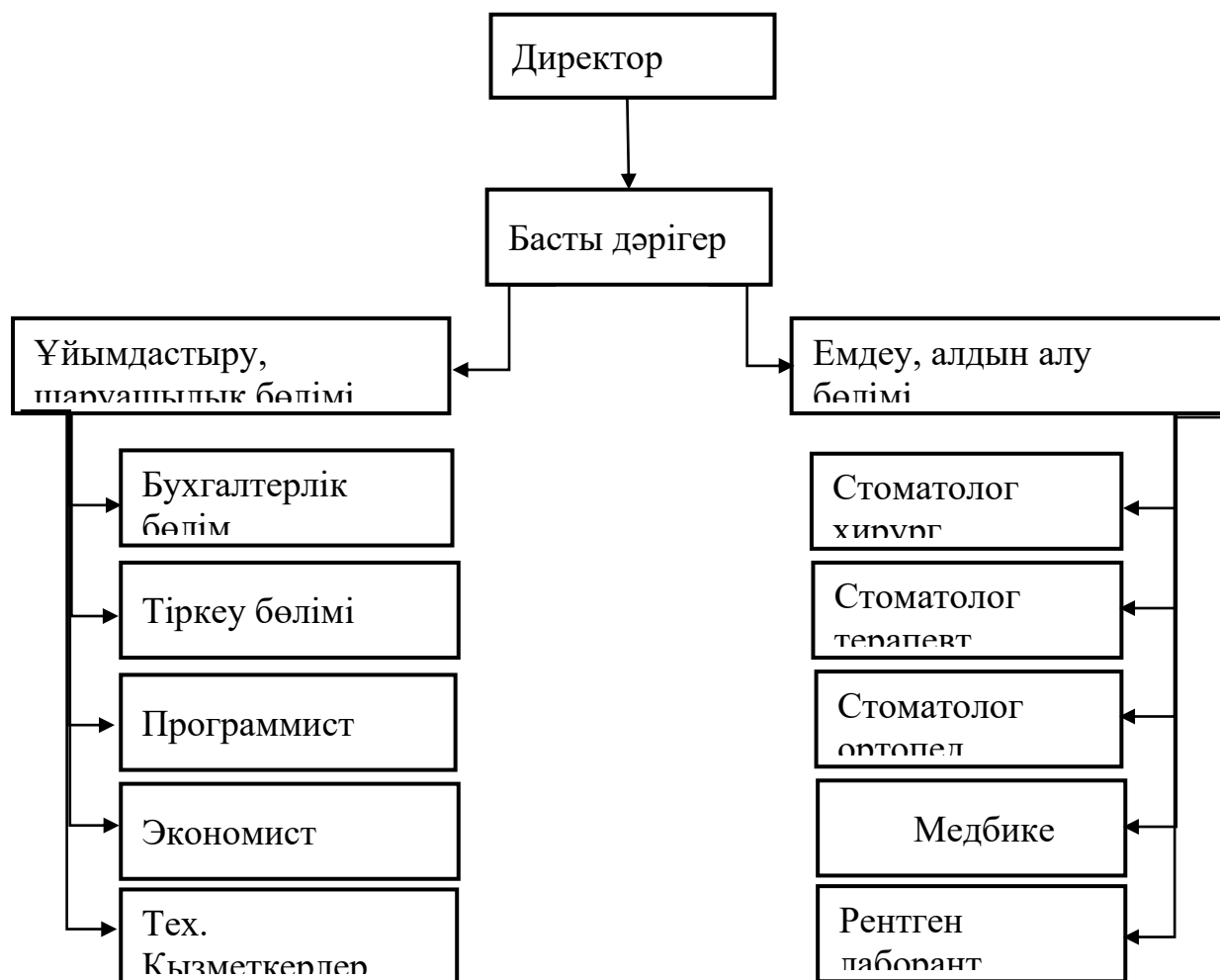
Көптеген қатынастар көбінесе қолданылады, яғни төмен кестедегі негізгі кестенің бір жазбасы бірнеше жазбадан тұрады, соның ішінде: ол олардың біріне сәйкес келуі мүмкін.

МББЖ – көптеген мәліметтер қорын пайдаланушылар (бағдарлама) жасау, қолдау және пайдалану мәліметтер қорын интеграцияланған қолдау үшін бағдарламалық кешен. МББЖ ерекшеліктері:

- жаңа мәліметтер базасын жасау;
- мәліметтер түрлендіру;
- мәліметтерді толтыру;
- іздеу;
- мәліметтер базасындағы құпия ақпарат;
- дискіге жаңартылған мәліметтер жазбалау;
- мәліметтерді және сұрауларға жауаптарды басып шығару.

2.5 «Denta+» стоматологиялық емханасының ақпараттық жүйесінің құрылымы

«Denta+» стоматологиялық емханасының ақпараттық жүйесінің құрылымы:



2.1 Сурет – «Denta+» стоматологиялық емханасының ақпараттық жүйесінің құрылымының сұлбасы

2.6 ER-диаграмманы құру

МБ құру көзқарасынан жүйелік талдау аясында бірінші кезенді жүзеге асыру қажет, яғни пәндік облыс объекті мен сипатталатын объектілер арасындағы шын байланыстар сөзбен сипаттауды жүргізу. Бұл сипаттама пәндік облыстардың объектілер арасындағы барлық өзара байланысын дұрыс анықтауға мүмкіндік бергені жөн.

Жалпы жағдайда қазіргі кезде пәндік облыстың құрамы мен құрылымын таңдаудың екі әдісі бар:

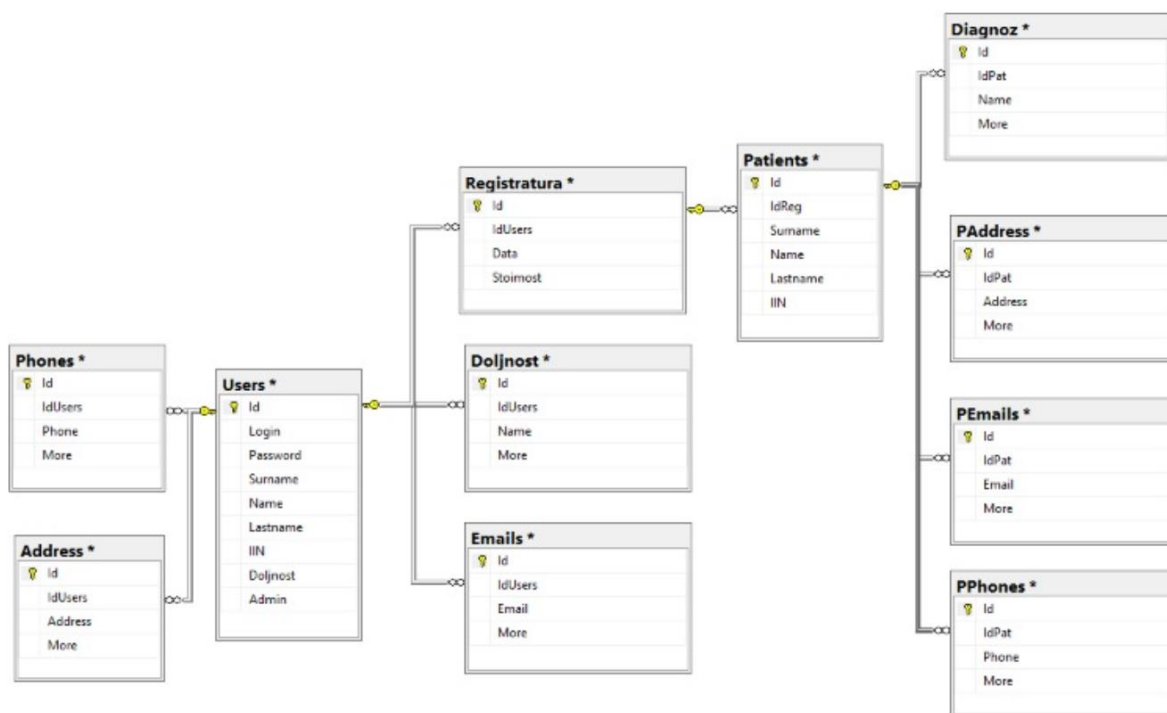
- функционалдык әдіс – ол «тапсырмадан» қозғалыс қағидасын жүзеге асырады және қарастырылатын МБ құрылытатын ақпараттық қажеттілік үшін кейбір тұлға тобы мен есеп кешенінің функциялары алдын ала анықталғанда қолданылады;

- пәндік әдіс – МБ-ның болашақ қолданушыларының ақпараттық қажеттіліктері нақты кездеспегенде. Олар көп аспектілі және әбден динамикалық болуы мүмкін. Осы кезде құрылған МБ пәндік деп аталады, яғни ол әртүрлі алдын ала анықталмаған есептерді шешу кезінде қолданыла алады.

Қазіргі кезде «мән-байланыс» немесе «Entity Relationship» Чен моделі мәліметтер базасын инфологиялық модельдеу кезінде факттік стандарт болады. Жалпыға бірдей ER-модель қысқарған аты қабылданды; заманауи CASE-құралдардың көбісі мәліметтерді осы модель формасында сипаттауға арналған құрал сайманды қамтиды. Сондай-ақ, МБ жобасын ER-модельден реляциялыққа автоматты түрлендіру әдістері жасалған; осы түрлендіру арқылы сәйкес нақты МББЖ-н датологиялық модельге орындалады. Барлық CASE-жүйелерден МБ жасаудың процестерін құжаттаудың дамыған құралдары бар; есеп берудің автоматты генераторлары МБ жобасының ағымдағы күйі туралы есеп МБ объекттерін сипаттауымен және оның қатынасын графикалық түрде де, стандартты дайын басылған есеп беру түрінде де дайындауға мүмкіндік береді, бұл жобаны енгізуді айтарлықтай жеңілдетеді.

Сонымен қатар, бұл жүйе әр түрлі "мәндер" жүйеге байланысты екенін көрсететін блок-схеманың түрі болып табылады. ER-диаграммалар көбінесе білім берудегі, бағдарламалық қамтамасыз ету мен бизнес-ақпараттық жүйелердегі реляциялық деректер базасын жобалау және жөндеу үшін пайдаланылады. ER-диаграммалар тіктөртбұрыштар, ромбалар, овалдар және байланыстырғыш сызықтарды қоса алғанда, мәндерді, олардың атрибуттарын және қатынастарын көрсету үшін символдардың стандартты жиынтығын пайдаланады.

ER-диаграммалар мәліметтерді өңдеу үшін қолданылады және мәліметтер мен олардың арасындағы қатынасты аықтаудың стандартты әдісін көрсетеді. ER-диаграмма жүйе мәні мен олардың өзара әрекеттесуінің әдістері туралы ақпаратты қамтиды, пәндік облыс үшін маңызды объекттер идентификациясын, осы объекттер (атрибуттар) қасиетін және олардың басқа объекттермен қатынасын (байланысын) қамтиды. Көп жағдайда ақпараттық модель өте күрделі және көптеген объекттерді қамтиды.



2.2 Сурет – Стоматологиялық емхананың мәліметтер базасын құрудағы ER диаграммасының сұлбасы

2.7 Мәліметтер базасының кестелерін сипаттау

Мәлімет базасының кестелерінің құрылымы жасалу үшін кестелердің қатынастары және олардың арасындағы байланыстары анықталады. Жұмысқа қатысты элементтер кесте құрылымында қамтылады. Бұл кезде қатынастардың аттары кестелер аттарына сәйкес келеді.

2.1 Кесте – Phones

Кесте аты	Өріс аты	Мәліметтер типі
Phones	Id	int AUTOINCREMENT
	IdUsers	Int
	Phone	Int
	More	varchar(100)

2.2 Кесте – Address

Кесте аты	Өріс аты	Мәліметтер типі
Address	Id	int AUTOINCREMENT
	IdUsers	Int
	Address	varchar(100)
	More	varchar(100)

2.3 Кесте – Users

Кесте аты	Өріс аты	Мәліметтер типі
Users	Id	int AUTOINCREMENT
	Login	varchar(100)
	Password	varchar(100)
	Surname	varchar(100)
	Name	varchar(100)
	Lastname	varchar(100)
	IIN	Int
	Doljnost	varchar(100)
	Admin	varchar(100)

2.4 Кесте – Registratura

Кесте аты	Өріс аты	Мәліметтер типі
Registratura	Id	int AUTOINCREMENT
	IdUsers	Int
	Data	Date
	Stoimost	Int

2.5 Кесте – Doljnost

Кесте аты	Өріс аты	Мәліметтер типі
Doljnost	Id	int AUTOINCREMENT
	IdUsers	Int
	Name	varchar(100)
	More	varchar(100)

2.6 Кесте – Emails

Кесте аты	Өріс аты	Мәліметтер типі
Emails	Id	int AUTOINCREMENT
	IdUsers	Int
	Emails	varchar(100)
	More	varchar(100)

2.8 Кесте – Patients

Кесте аты	Өріс аты	Мәліметтер типі
Patients	Id	int AUTOINCREMENT
	IdReg	Int
	Surname	varchar(100)
	Name	varchar(100)
	Lastname	varchar(100)
	IIN	Int

2.8.1 Кесте – Diagnoz

Кесте аты	Әріс аты	Мәліметтер типі
Diagnoz	Id	int AUTOINCREMENT
	IdPat	Int
	Name	varchar(100)
	More	varchar(100)

2.8.2 Кесте – PAddress

Кесте аты	Әріс аты	Мәліметтер типі
PAddress	Id	int AUTOINCREMENT
	IdPat	Int
	Address	varchar(100)
	More	varchar(100)

2.8.3 Кесте – PEmails

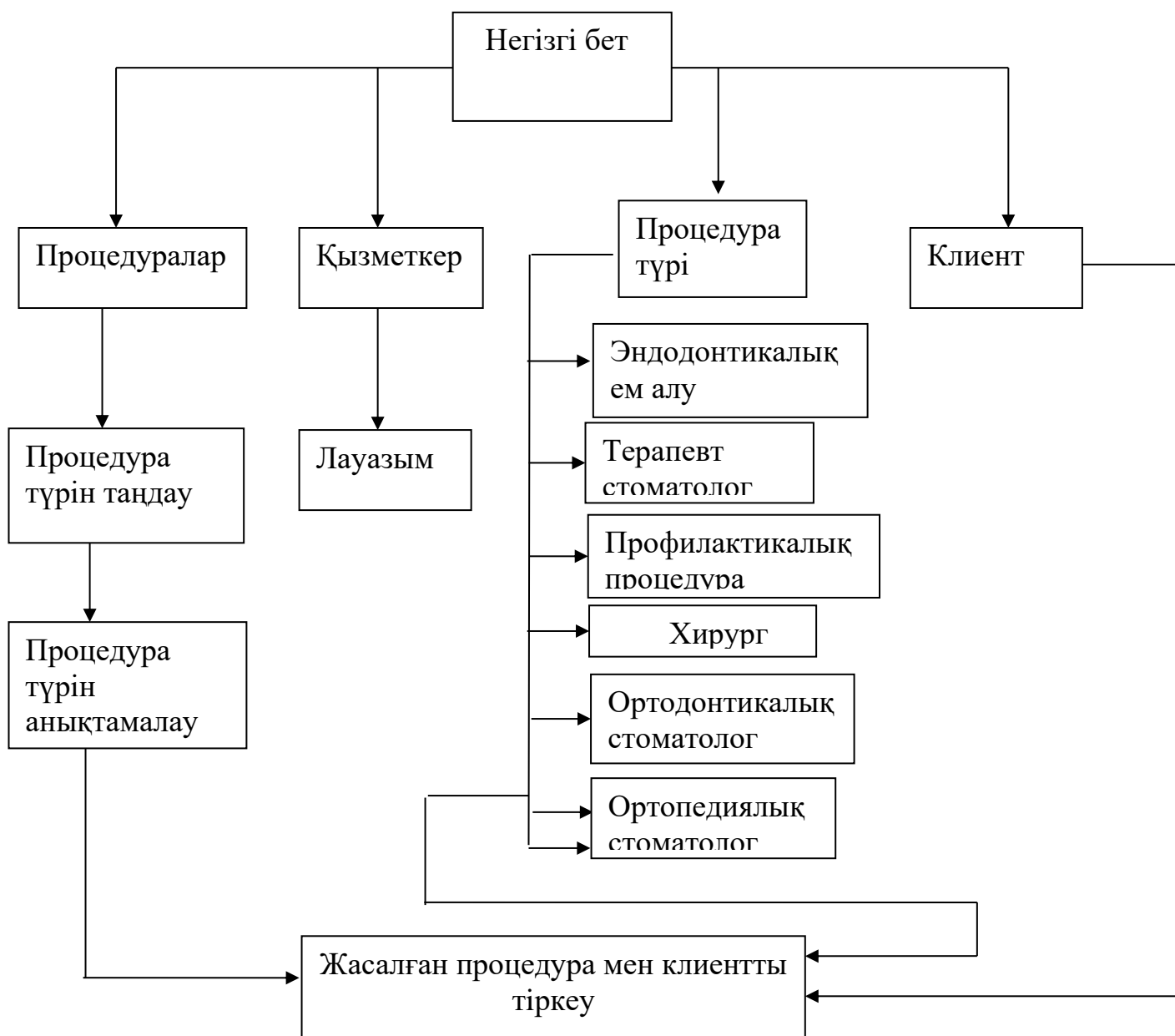
Кесте аты	Әріс аты	Мәліметтер типі
PEmails	Id	int AUTOINCREMENT
	IdPat	Int
	Emails	varchar(100)
	More	varchar(100)

2.8.4 Кесте – PPhones

Кесте аты	Әріс аты	Мәліметтер типі
PPhones	Id	int AUTOINCREMENT
	IdPat	Int
	Phones	Int
	More	varchar(100)

2.8 Жүйенің интерфейсі

Қолданушының батырма қолдану арқылы функция жүйесін таңдай алады. Қолданушы мәліметтер базасының құрылымын экрандық құжат түрінде көре алады (2.3-сурет).



2.3 Сурет – Жүйенің интерфейсі

2.9 Бағдарламалау тілін таңдауды негіздеу

Бүгінгі күні Visual Studio 2017 құралдары:

- дайындық ортасы;
- бірлескен жұмысты ұйымдастыруға арналған сервис-Visual Studio Team Services;
- оның жергілікті нұсқасы, TFS ретінде белгілі;
- мобильді қосымшаларды әзірлеудің толық циклін іске асыру кешенді шешім – Visual Studio Mobile Center;

- Visual Studio Code кодының Кросс-платформалық редакторы (Mac, Linux, Windows үшін қол жетімді) - бұл жақында 1.10 нұсқасына дейін жаңартылған;
- Mac үшін Visual Studio, Microsoft өзінің IDE және macOS нұсқасын қолдайды шығарылған.

Әрбір нұсқасы Microsoft корпорациясының веб-сайтында қол жетімді, кез келген платформа үшін қосымшаларды жасау кезінде жеңіл опцияны жасауға тырысады.

Ең алдымен, кез-келген әзірлеуші Visual Studio 2017 толық нұсқасын жүктеп, C# тілінде кроссплатфорлы Мобильді қосымшаларды жасау бойынша оқыту сервисіне Xamarin University қол жеткізуге 60 күндік тегін жазылым ала алады. Тегін жазылым 14 наурызға дейін жүктейтін барлық адамдар үшін қолжетімді болады.

Екіншіден, жасаушылар тек кодты жазуға назар аудару үшін барлық жағдайларды жасай отырып, әзірлеушілердің өнімділігін арттыруға қамқорлық жасауды жалғастыруда. Мысалы, барлық қолдау көрсетілетін тілдер үшін код бойынша шарлауда, рефакторингте, түзетулерде және баптауда жақсартулар бар. Сонымен қатар, жаңа нұсқа модульдік тестілеу және тәуелділіктерді тексеру үшін нақты уақыттың жаңа функциялары арқылы команда әзірлеу жылдамдығын арттыруға мүмкіндік береді.

Үшінші маңызды өзгеріс құралды орнату процесіне қатысты. Жаңа, жеңіл модульдік тәсіл қажетті ортаның компоненттерін ғана орнатуға мүмкіндік береді және құралды басынан аяғына дейін орнатуды тездетеді. Сонымен қатар, әзірлеушілер кез келген қажетті код фрагментін қалпына келтіру үшін жобалар мен шешімдер жасаудың қажеті жоқ.

Visual Studio соңғы презентациялары Azure бұлтты платформасы қызметтерімен интеграцияны жақсартусыз аяқталмады. Бұл саладағы Microsoft әзірлемелері Ide ортасынан тікелей Azure бұлт қолданбаларын жасауды, баптауды, орналастыруды және жариялауды жеңілдетеді.

Ұялы құрылғыларды әзірлеу саласындағы тағы бір маңызды өзгеріс. Әзірлеушілер жақсартылған жөндеу және профильдеу құралдарын, сондай-ақ модульдік тестілер жасауға арналған құралдарды алды. Егер сіз кросс-платформалық қосымшаны жасауды жоспарласаңыз, онда қазір Visual Studio 2017 және Xamarin көру немесе Apache Cordova және, бәлкім, Visual C++ балама тәсілді пайдалану, бірақ бір құрал-Visual Studio 2017 аясында кросс – платформалық кітапханаларды құру.

C# – заманауи объектілі-бағытталған және типтік бағдарламалау тілі. C# тілдердің кең танымал тобына жатады C, және C, C++, Java немесе JavaScript жұмыс істеген кез келген жақсы таныс болып көрінеді.

Мұнда C# тілінің 8 негізгі компоненттеріне және ерте нұсқаларына шолу жасалған. Егер сіз интерактивті мысалдар арқылы тілді үйренгіңіз келсе, c#кіріспе нұсқауларымен жұмыс істеуді ұсынамыз.

C# объектілі-бағытталған тіл болып табылады, бірақ компонентті-бағытталған бағдарламалауды да қолдайды. Қазіргі заманғы қосымшаларды әзірлеу жеке функционалдық мүмкіндіктерді іске асыратын автономды және

өзін-өзі жазатын пакеттер түрінде бағдарламалық компоненттерді құруға көбірек тартуда. Мұндай компоненттердің басты ерекшелігі-олар қасиеттері, әдістері мен оқиғалары бар бағдарламалау моделі болып табылады. Оларда компонент туралы декларациялық мәліметтерді беретін атрибуттар бар. Олар өз құжаттарын қамтиды. С# мұндай жұмыс тұжырымдамасын қолдайтын тілдік құрылымдарды ұсынады. Осы С# бағдарламалық компоненттерді жасау және қолдану үшін жарамды.

Мұнда қосымшалардың сенімділігі мен тұрақтылығын қамтамасыз ететін С# тілінің бірнеше функциялары ғана. Қоқыс жинау автоматты түрде пайдаланылмайтын объектілермен қамтылған жақты босатады. Ал кейбіреулер қателерді табу және оларды қалпына келтіру үшін құрылымдалған және кеңейтілген тәсіл береді. Тілдің типтік қауіпсіз құрылымы безинициализированных айнымалылардан оқу, массивтердің шекараларынан тыс индекстеу немесе тексерілмеген түрлерді орындау мүмкін емес етеді.

С# - да түрлердің бірыңғай жүйесі бар. С# барлық түрлері, Int және double сияқты примитивтерді қоса алғанда, объект бір тамыр түрінен мұра. Осылайша, барлық түрлер операциялардың жалпы жиынтығын қолданады және кез келген түрдегі мәндерді сақтауға, тасымалдауға және өңдеуге болады. Сонымен қатар, С# нысандар үшін жақты динамикалық бөліп шығаруға, сондай-ақ стекте қарапайым құрылымдарды сақтауға мүмкіндік беретін теңшелетін сілтемелер түрлері мен мән түрлерін қолдайды.

Одан әрі даму кезінде с# бағдарламалар мен кітапханалардың үйлесімділігін қамтамасыз ету үшін, С# әзірлеу кезінде нұсқаларды басқаруға көп көңіл бөлінді. Көптеген бағдарламалау тілдері бұл мәселені назарға алады. Бағдарлама нәтижесінде бұл тілдерде тәуелді кітапханалардың жаңа нұсқаларын шығарғанда қалағанына қарағанда жиі бұзылады. Нұсқаларды басқару мәселелері virtual және override бөлек модификаторлары, әдістерді қайта жүктеуді шешу ережелері және интерфейс мүшелерін анық хабарландыруды қолдау сияқты с# әзірлеу аспектілеріне елеулі әсер етті.

Соңғы с# нұсқаларында басқа бағдарламалау парадигмалары қолданылды. С# лямбда-өрнектер сияқты функционалдық бағдарламалау тәсілдерін қолдайтын функцияларды қамтиды. Басқа жаңа мүмкіндіктер деректер мен алгоритмдердің бөлінуін қолдайды, мысалы, үлгілерді салыстыру.

С# типті жүйені тез және визуальды құру ортасы деп атайды. Себебі ол өзіне компилятормен қоса редактор және көптеген дайын ібағдарламалар жиынтығын қамтиды. С#-та шамамен 200 - ге жуық дайын компоненттер бар.

С# қолданушыға өте ыңғайлы, тез бағдарлама құруға мүмкіндік береді. Бұл ортаның құрал-саймандары өте көп және керек кезінде актив элемент үшін F1 пернесін басу арқылы көмек алуға болады.

С# ортасы – бағдарламаушының жоғары әрі тиімді жұмысын қамтамасыз ететін күрделі механизм. Ол визуальды түрде бір уақытта бірнеше ашылған терезелермен жұмыс атқарады. Терезелер экранда толық немесе бөлшектелген түрде бір- бірін жаба отырып жылжи алады, бұл Word тексттік процессор немесе

Excel кестелік процессор ортасының «қатаңдығына» үйренген қолданушының дисккомфорт сезімін оятуы мүмкін.

C#-тың негізгі ерекшелігі – онда қосымша құруда компоненттік және объектілік тәсілдер пайдаланылады. Компоненттік тәсілдің мәнісі жеңіл: әр қосымша арнайы іс-әрекеттерді орындайтын компоненттерден жинақталады. Ол жеткіліксіз болса, объектіні өңдеуге арналған үстеме бағдарлама құрылады. Компоненттер визуальды компоненттер кітапханасында (VCL – Visual Component Library) жинақталған. Компоненттер панелінде түрлі кластарға тиісті стандартты компоненттер өте көп. Пайдаланушы жаңа компонент дайындап, оны осы панельге қосуына да болады.

C#-та бағдарлама дайындау, бағдарлама мәзірін құру, анимация, мультимедиа үрдістерін ұйымдастыру, OLE технологиясын пайдаланып, басқа офистік қосымшаларды шақыру, олармен жұмыс істеу және т.б. іс-әрекеттерді орындауға болады. C# көптеген есептерді шешуге мүмкіндік береді, оның ішінде:

- әртүрлі бағыттағы аяқталған қосымшаларды Windows үшін құру;
- кез келген тілде кез келген қосымшалар үшін профессионалды көрінетін интерфейсін тез құру. Интерфейс Windows-тың барлық талаптарын қанағаттандырады және қолданушы компьютерінде орнатылған жүйеге автоматты түрде ыңғайланады. Өйткені Windows-тың көптеген функцияларын, кітапханаларын қолданады;
- бағдарламалаудың басқа түрлерінде қолдануға болатын функция, форма, компоненттердің динамикалы біріктірілетін кітапханасын (DDL) құру;
- кез келген типтегі жергілікті және жойылған мәліметтер базасымен жұмыс істеудің қуатты жүйесін құру;
- кестелері, графиктері бар күрделі есеп берулерді құру және баспаға шығару;
- қосымшалардан Windows арқылы жұмыс істеуге болатын өз қосымшалары үшін анықтамалық жүйелерді (hlp файлдары) құру;
- Windows-тың барлық талаптарын және спецификасын ескеретін Windows-тың қосымшалары үшін орнатудың профессионалды бағдарламаларын құру.

Көптеген операторлардың жазылуы Турбо Паскальдағы сияқты. Бірақ, мұнда прогаммалауды үйрену үшін объект, оқиға, қасиет, әдіс, класс ұғымдарымен еркін танысып, компоненттерді пайдалану және түрлі командалардың жазылу түрлерін білу қажет.

3 БАҒДАРЛАМАЛЫҚ ҚАМТАМАНЫҢ БАЯНДАЛУ БӨЛІМІ

3.1 Жалпы мағлұматтар

«Denta+» стоматологиялық емханасының ақпараттық автоматтандырылған жүйесін құру Visual Studio және SQL Server құрылымды сұраныстар арқылы жазылған.

«Denta+» стоматологиялық емханасының ақпараттық автоматтандырылған жүйесін құру – процедура қабылдаушы клиенттерге таңдамалы қызметтерді, ұсыныстарды және кеңестерді пайдалану мүмкіндігін береді.

Біздің автоматтандырылған жүйе жұмысын жасау үшін Microsoft Windows 10 операциялық жүйесі, Visual Studio, SQL ServэКСЕOr қажет.

3.2 Функционалдық тағайындалуы

Visual Studio және Microsoft SQL Server бағдарламалау құралдарын пайдалана отырып стоматологиялық емхананың автоматтырылған жүйесін құру. Клиенттердің сұрауларын сипаттау.

Мәліметтер базасының көмегімен қолданушы мәліметтерге қатынаудың жолын қысқартады. Бағдарлама «Denta+» стоматологиялық емханасының тіркеуші маманына арналып жасалған.

3.3 Логикалық құрылымын сипаттау

UnMain басты модулі:

Модульдің аты;

Модульдің ашық интерфейсі;

Қосылатын модульдер тізімін баяндау;

Компоненттер типін баяндау;

Процедураларды баяндау;

Кластың ашық және жабық бөлімі;

Сақтандыру түрі туралы есеп беру функциясы;

Программаның соңы.

3.4 Шақыру және жүктеу

Мәліметтер базасын Visual Studio ортасында ашу үшін Пуск → Программы → Visual Studio жолын ашамыз. Өз папкамызды тауып, проектімізді ашамыз. Оны F9 пернесі арқылы немесе папкадағы exe файлын тауып жүктейміз. Бұл жұмысты орындау үшін Visual Studio жүктеп, қарап шыға аласыз.

3.5 Қолданылған техникалық жабдықтар

Стоматологиялық емхананың ақпараттық жүйесіне құру үшін Intel(R) Core(TM) i5-4200M CPU 2.50 GHz / RAM 8192 Mb / HDD 697 Gb/ VideoCard 2176 Mb / HP ENVY m6 ноутбугын қолданылды.

Сонымен қатар HP Laser-jet 1010 принтерін қолдандым.

3.6 Кіріс мәлімет

Кіріс мәліметтер ретінде клиенттердің тапсырыс үшін ақпараттарын енгізу, клиенттер, қызметкерлер, процедуралар мен жаңа қызметтер туралы енгізілген мәліметтері бойынша тіркеу.

3.7 Шығыс мәлімет

Шығыс мәліметтерге қолданушы туралы енгізілген мәліметтер бойынша сұраныстары қабылданған жағдайда оларға керекті мәліметтерді іздеу жүйесі бойынша шығару. Қолданушы қажетіне енгізілген мәліметтер бойынша есептеме алуға болады.

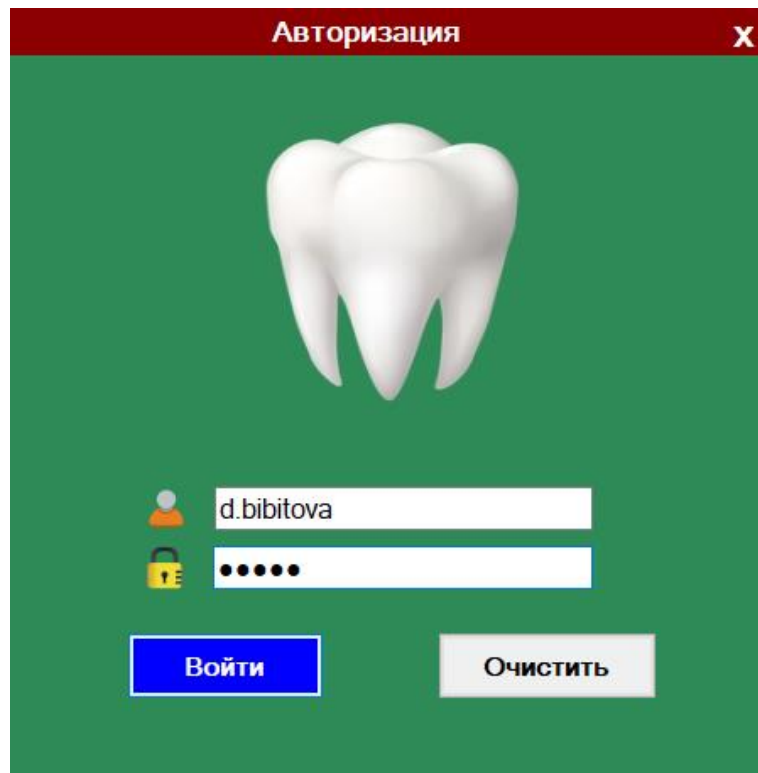
3.8 Жүйенің жұмыс істеуін сипаттау

Құрылып отырған жұмыс Visual Studio ортасында бағдарламалау ортасында құрастырылған. C# бағдарламасын Borland Softwar Corporation құрастырып жасаған.

Құрылған мәліметтер базасының дұрыстығын тексеру үшін F9 пернелері арқылы жүктейміз.

Бағдарламаны іске қосу – RUN батырмасы арқылы жүзеге асады. Бағдарламаны іске қосқанда «Негізгі меню» терезесі ашылады.

Негізгі меню терезесінен базаға қызметкер өз фамилиясын таңдап, паролі арқылы кіреді (3.1-сурет).



3.1 Сурет – Жүйеге ену диалог терезесі

Парольды тергеннен кейін «Войти» батырмасын басамыз. Парольды дұрыс енгізгеннен кейін ғана бағдарламаға кіруге мүмкіндік аламыз (3.2-сурет).



3.2 Сурет – Негізгі бет

Келесі 3.3-суретте қалыптасқан ақпарат бойынша қызметкерлер туралы жалпы мәлімет шығару терезесі келтірілген.

Сотрудники

Выборка

Электронная почта Телефоны

Сведения о пользователе:

	Login	Password	Surname	Name	Lastname	IIN	Admin
▶	u.amir	1234	Әмір	Ұлжан	Қайратқызы	980901400561	No
*							

Электронные почты сотрудника:

	Электронная почта	Вид
▶	ulzh_a@kaznitu.kz	Корпоративная
	ulzh_a@gmail.com	Дополнительная
	ulzh_a@mail.ru	Личная
*		

Добавить

Удалить

3.3 Сурет – Қызметкерлер терезесі

Қолданушылар батырмасын басқан кездегі терезедегі енгізілген ақпарат беріледі. Енгізілген қолданушыларды өшіріп, мәліметтерді өзгерте аламыз.

Пользователи

Выборка

Поиск данных: Выберите вид данных

Количество записей: 3

Электронная почта Телефоны Адрес Должность Регистратура

Сотрудники

	Login	Password	Surname	Name	Lastname	IIN	Admin
	d.bitova	12345	Бібітова	Динара	Ақылжанқызы	990111450972	Yes
▶	u.amir	123	Әмір	Ұлжан	Ақылжанқызы	980901400561	No
	b.aubakirov	B123T	Аубакиров	Бахтияр	Талғатұлы	990119300630	No

Добавить пользователя

Удалить пользователя

Адрес сотрудника:

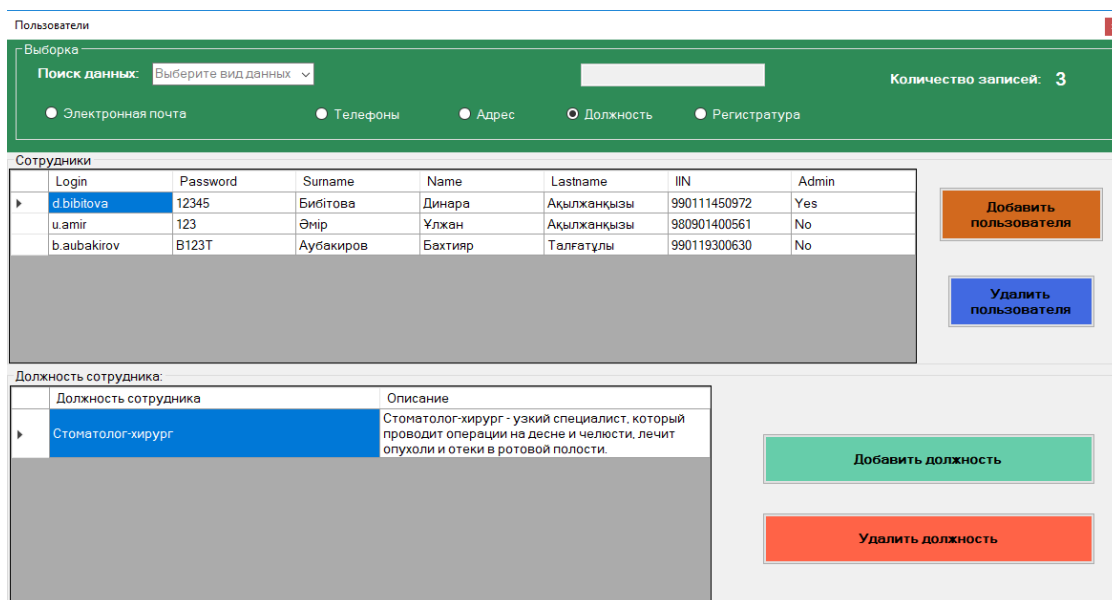
	Адрес сотрудника	Вид
▶	г.Алматы, ул.Желтоқсан, д.5, кв.77	Временная прописка
	г.Талғар, ул.Грушевая, дом.35	Домашний

Добавить адрес сотрудника

Удалить адрес сотрудника

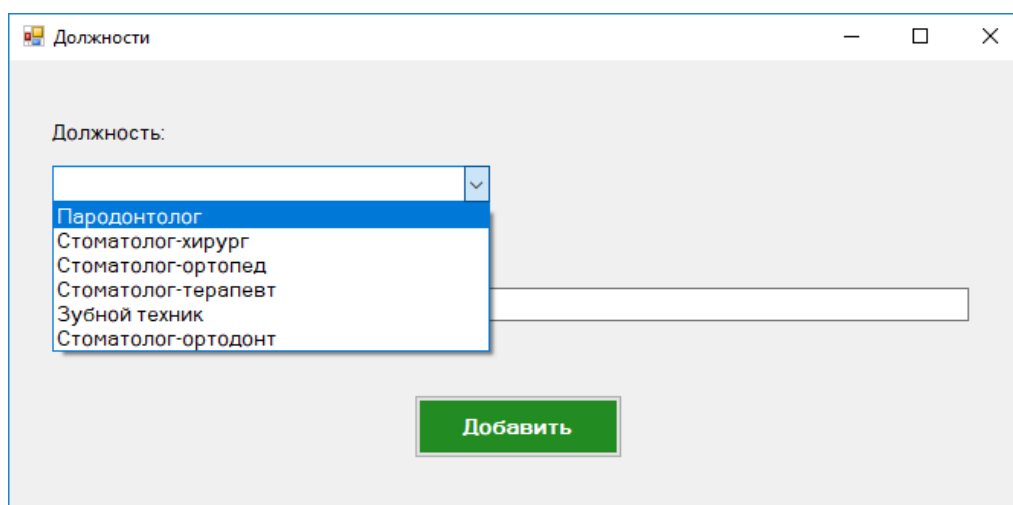
3.4 Сурет – Қолданушылар батырмасы

Қолданушылар батырмасын басқан кездегі терезедегі енгізілген ақпарат беріледі және олар туралы ақпаратты қарай аламыз.



3.5 Сурет – Қолданушылар терезесі

Егер админ деген құқығы болса, жалпы қолданушылар туралы ақпарат енгізе аламыз.



3.6 Сурет – Должность терезесі

Клиенттерді тіркеу процесі. Мәліметтерді енгізу 3.7 Суретінде көрсетілген.

Регистрация пациента X

25.05.2020

Стоимость услуги

Введите фамилию

Введите имя

Введите отчество

Введите ИИН

Зарегистрировать

3.7 Сурет – Регистрация пациентов терезесі

Регистрация пациента X

25.05.2020

7300

Машанов

Талгат

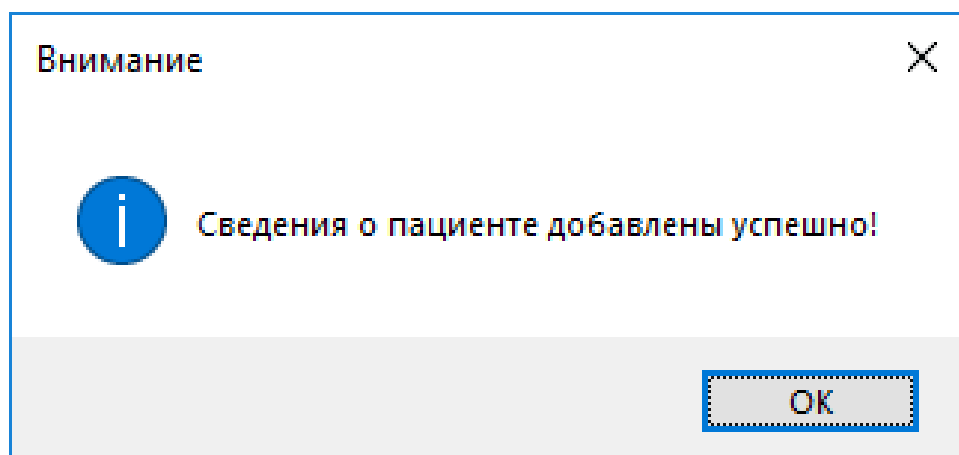
Садуакасович

810215300457

Зарегистрировать

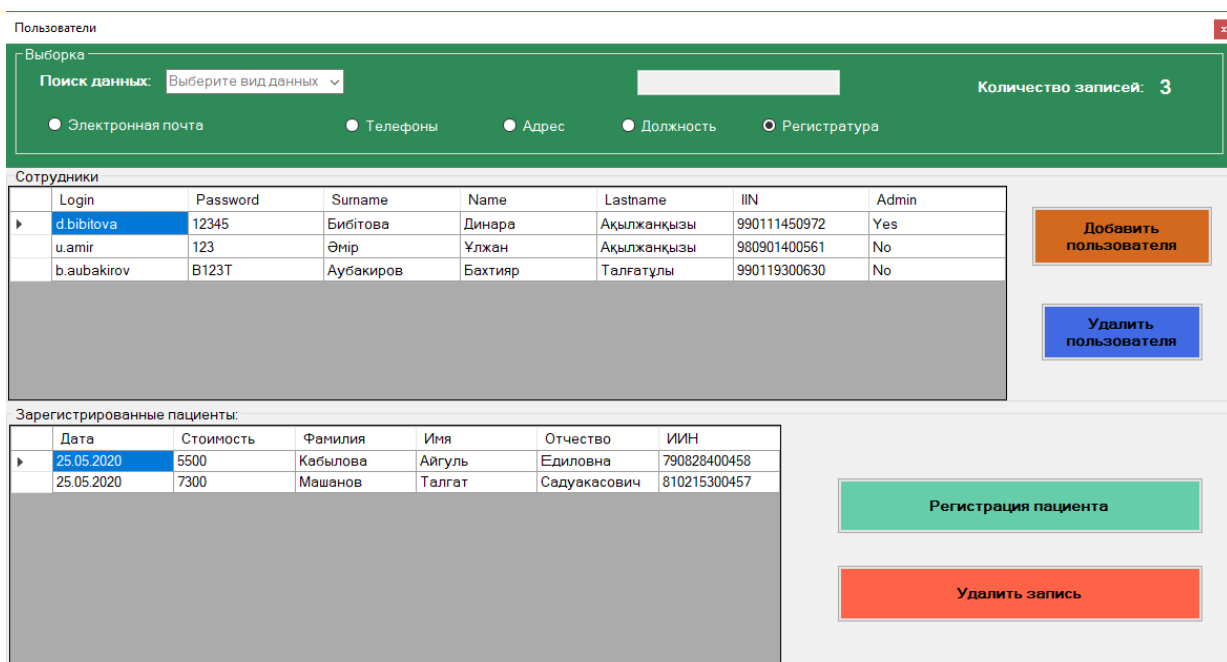
3.9 Сурет – Мәліметтерді енгізу терезесі

Енгізілген мәліметтерді «Зарегистрировать» батырмасын басып сақтай аламыз. Енгізілген мәліметтер сақаталады.



3.10 Сурет – Енгізілген мәліметтерді сақтау

Қызметкерлердің атына жазылған клиенттер туралы енгізілген ақпаратты алу 4.1 Суретте көрсетілген



Сотрудники	Login	Password	Surname	Name	Lastname	IIN	Admin
▶	d.bibitova	12345	Бибітова	Динара	Ақылжанқызы	990111450972	Yes
	u.amir	123	Әмір	Ұлжан	Ақылжанқызы	980901400561	No
	b.aubakirov	B123T	Аубакиров	Бахтияр	Талғатұлы	990119300630	No

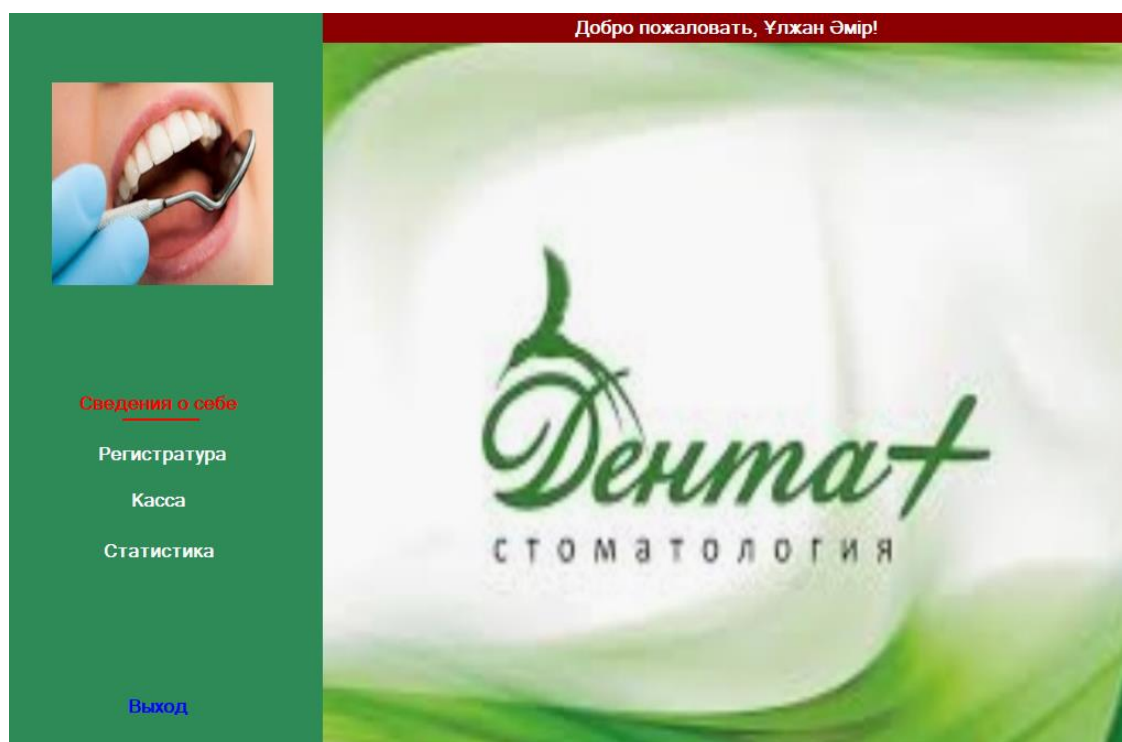
Зарегистрированные пациенты:	Дата	Стоимость	Фамилия	Имя	Отчество	ИИН
▶	25.05.2020	5500	Кабылова	Айгуль	Едиловна	790828400458
	25.05.2020	7300	Машанов	Талгат	Садуакасович	810215300457

3.11 Сурет – Қолданушылар терезесі

3.12 Суретте қолданушы админ құқығы болмаған кезде, жүйеге ену процессі. Админ құқығы болмаған жағдайда қолданушылар туралы ақпаратты қарай алмайды. Тек өзі туралы ақпаратты енгізіп өшіре алады.



3.12 Сурет – Қолданушы жүйеге ену терезесі



3.13 Сурет – Меню терезесі

«Сведения о себе батырмасын басып өзі туралы ақпаратты қарастыра алады.
3.14 Сурет.

Сотрудники

Выборка

Электронная почта Телефоны

Сведения о пользователях:

	Login	Password	Surname	Name	Lastname	IIN	Admin
▶	u.amir	1234	Әмір	Ұлжан	Қайратқызы	980901400561	No
*							

Электронные почты сотрудника:

	Электронная почта	Вид
▶	ulzh_a@kaznitu.kz	Корпоративная
	ulzh_a@gmail.com	Дополнительная
	ulzh_a@mail.ru	Личная
*		

Добавить

Удалить

3.14 Сурет – Сотрудники терезесі

Сотрудники

Выборка

Электронная почта Телефоны

Сведения о пользователях:

	Login	Password	Surname	Name	Lastname	IIN	Admin
▶	u.amir	1234	Әмір	Ұлжан	Қайратқызы	980901400561	No
*							

Телефоны сотрудника:

	Номер телефона	Вид
▶	+77077041598	Мобильный
	+7(727)-390-63-21	Рабочий
*		

Добавить

Удалить

3.15 Сурет – Сотрудники терезесі

ҚОРЫТЫНДЫ

Құрылған «Denta+» стоматологиялық емханасының мәліметтер базасы уақыттың өтуімен өзгеруі немесе толықтырылуы мүмкін және мәліметтер база бөлімдік үлестірілген қосымша түрінде, Visual Studio объектілі-бағытталған бағдарламалау тілінде жазылды. Visual Studio қуатты құрылғыларын пайдалану негізінен Windows операциялық жүйесімен жұмыс істейтін қосымшаларды құруды және кейбір жағдайларда мәліметтер базасының қосымшаларын құру кезінде пайдаланылады. Нәтижесінде пайдаланушыға максималды бағытталған программалық өнім шығарылады.

Бұл дипломдық жұмыста «Denta+» стоматологиялық емханасының автоматтандырылған жүйесі құрастырылып, келесідей тапсырмалар орындалды :

- қызметкер процедураларға тапсырыс алуы;
- жасалған процедураларды тіркеу, баспаға шығару;
- қызметкерлер өзінің құпия сөзін өзгерту;
- жүйе қолданушыларының қызмет көрсетуі.

Сонымен қатар, бұл жұмыста «Denta+» стоматологиялық емханасының қызметінің қызметімен клиент және жұмысшылар арасындағы қызмет байланысы бір жүйеге келіп халыққа пайдалы қолайлы компьютерлік қызмет көрсету қатынасы жасалды, процедуралардың жасалу жүйесі құрылып, жасалған әрбір процедураны дер кезінде компьютер арқылы тіркеуге алып стоматологиялық емхананың күндік, айлық, жылдық табысын тез есептеп шығару, сол арқылы бухгалтердің жұмысын жеңілдету, компьютер арқылы стоматологиялық емханадағы кез-келген процедураны тез іздеп тауып, ол туралы егжей – тегжейлі ақпаратқа қол жеткізу мүмкіндігі қарастырылып, клиенттерге сапалы қызмет көрсету, бұл бағдарлама клиенттерді тіркеп отыратын маманға арналып жасалынады.

«Denta+» стоматологиялық емханасының автоматтандырылған басқару жүйесін құру қазіргі кездегі электронды үкімет жүйесін жалпырастырып жатқан кезде өте маңызды жоба, сондықтан алдағы уақытта осы сияқты жобаны стоматологиялық емхана өз қажеттеріне пайдалануына болады. Барлық стоматологиялық емханаларда автоматтандырылған басқару жүйесі құрылар болса, ол жұмыс өнімділігін арттырып стоматологиялық емхананы басқару қызметін сапаландырады.

Бағдарламалық кешен Visual Studio ортасында жасалып, мәліметтер базасын басқару жүйесі ретінде Microsoft SQL Server таңдалды. Жасалған дипломдық жұмысты жобалау үрдісінде алға қойылған мақсаттардың бәріне қол жеткізілді және бұл жүйе стоматологиялық емхана қызметкерлерінің жұмысын айтарлықтай жеңілдетеді деген үміттемін.

ПАЙДАЛАНҒАН ӘДЕБИЕТТЕР ТІЗІМІ

- 1 Варфоломеева А.О. Информационные системы предприятия: Учебное пособие / А.О. Варфоломеева, А.В. Коряковский, В.П. Романов. - М.: НИЦ ИНФРА-М, 2013. - 283 с. 8.
- 2 Холзнер С. Программирование на С#. Учебный курс; Питер, 2001. - 576 с.
- 3 Кузин А.В. Базы данных: Учебное пособие для студ. высш. учеб. заведений / А.В. Кузин, С.В. Левонисова. - М.: ИЦ Академия, 2012. - 320 с.
- 4 Бажанов Н. Н. Стоматология / Н.Н. Бажанов. - М.: Медицина, 2013. - 270 с.
- 5 Лобовкина, Л. А. Современные технологии реставрации зубов / Л.А. Лобовкина, А.М. Романов. - М.: МЕДпресс-информ, 2015. - 128 с.
- 6 Информатика /под редакцией С.В.Симоновича. СПб, Питер 2016- 400 с.
- 7 Пирогов В.Ю. Информационные системы и базы данных: организация и проектирование: Учебное пособие / В.Ю. Пирогов. - СПб.: БХВ-Петербург, 2009. - 528 с.
- 8 Норенков И.П. Автоматизированные информационные системы: Учебное пособие / И.П. Норенков. - М.: МГТУ им. Баумана, 2011. - 342 с.
- 9 <https://tproger.ru/news/vs-2017/>
- 10 Робинсон С.; Корнес, О.; Глинн, Д. и др. С# для профессионалов; М.: Лори, 2005. - 396 с.
- 11 <https://stom-academy.com/blog/it-v-mire-stomatologii-tsifrovyye-tehnologii-mogut-by-t-ispol-zovany-na-vseh-e-tapah-stomatologicheskogo-lecheniya/>
- 12 <https://nsportal.ru/ap/library/nauchno-tekhnicheskoe-tvorchestvo/2019/12/02/avtomatizirovannye-meditsinskie-sistemy>
- 13 Информатика негіздері / Жапарова Гүлжамал Әлкенқызы, - Алматы : Экономика, 2006 . - 296 б.
- 14 Байжұманов М.Қ., Жапсарбаева Л.Қ. Информатика: - Астана, 2004. – 232

А ҚОСЫМШАСЫ

```
using System;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Windows.Forms;

namespace Autorization
{
    public partial class Autorization : Form
    {
        public Autorization()
        {
            InitializeComponent();
            textBox1.Text = "Введите логин";
            textBox1.ForeColor = Color.Gray;
            textBox2.Text = "Введите пароль";
            textBox2.ForeColor = Color.Gray;
            textBox2.UseSystemPasswordChar = false;
        }

        DB db = new DB();
        DataTable table = new DataTable();
        SqlDataAdapter adapter = new SqlDataAdapter();

        String name, surname, pass, admin, login;

        public Boolean isUsersExists()
        {
            SqlCommand cmd_login = new SqlCommand("SELECT Login FROM
Users WHERE Login = @ul", db.GetConnection());
            cmd_login.Parameters.Add("@ul", SqlDbType.VarChar).Value =
textBox1.Text;
            adapter.SelectCommand = cmd_login;
            adapter.Fill(table);
            if (table.Rows.Count > 0)
            {
                db.openConnection();
                login = cmd_login.ExecuteScalar().ToString();
                db.closeConnection();
                return true;
            }
            else

```

```

        {
            MessageBox.Show("Такой пользователь не существует!",
                "Внимание!", MessageBoxButtons.OK, MessageBoxIcon.Information);
            textBox1.Text = "";
            textBox2.Text = "Введите пароль";
            textBox2.ForeColor = Color.Gray;
            textBox2.UseSystemPasswordChar = false;
            textBox1.Focus();
            return false;
        }
    }

private void button1_Click(object sender, EventArgs e)
{
    try
    {
        if (textBox1.Text == "" || textBox1.Text == "Введите логин")
        {
            MessageBox.Show("Введите логин", "Внимание!",
                MessageBoxButtons.OK, MessageBoxIcon.Information);
            textBox1.Focus();
            return;
        }

        if (textBox2.Text == "" || textBox2.Text == "Введите пароль")
        {
            MessageBox.Show("Введите пароль", "Внимание!",
                MessageBoxButtons.OK, MessageBoxIcon.Information);
            textBox2.Focus();
            return;
        }

        db.openConnection();

        SqlCommand cmd_name = new SqlCommand("Select Name From
        Users WHERE Login = @ul", db.GetConnection());
        cmd_name.Parameters.Add("@ul", SqlDbType.VarChar).Value =
        textBox1.Text;
        adapter.SelectCommand = cmd_name;
        adapter.Fill(table);
        if (table.Rows.Count > 0)
            name = cmd_name.ExecuteScalar().ToString();
    }
}

```

```

        SqlCommand cmd_surname = new SqlCommand("Select Surname
From Users WHERE Login = @ul", db.GetConnection());
        cmd_surname.Parameters.Add("@ul", SqlDbType.VarChar).Value =
textBox1.Text;
        adapter.SelectCommand = cmd_surname;
        adapter.Fill(table);
        if (table.Rows.Count > 0)
            surname = cmd_surname.ExecuteScalar().ToString();

        SqlCommand cmd_pass = new SqlCommand("Select Password From
Users WHERE Login = @ul", db.GetConnection());
        cmd_pass.Parameters.Add("@ul", SqlDbType.VarChar).Value =
textBox1.Text;
        adapter.SelectCommand = cmd_pass;
        adapter.Fill(table);
        if (table.Rows.Count > 0)
            pass = cmd_pass.ExecuteScalar().ToString();

        SqlCommand cmd_adm = new SqlCommand("Select Admin From
Users WHERE Login = @ul", db.GetConnection());
        cmd_adm.Parameters.Add("@ul", SqlDbType.VarChar).Value =
textBox1.Text;
        adapter.SelectCommand = cmd_adm;
        adapter.Fill(table);
        if (table.Rows.Count > 0)
            admin = cmd_adm.ExecuteScalar().ToString();

        db.closeConnection();

        if (isUsersExists())
        {
            if (pass.Equals(textBox2.Text))
            {
                Menu m = new Menu();
                m.Show();
                m.label4.Text = "Добро пожаловать, " + name + " " + surname +
"!";

                m.label5.Text = admin;
                m.label6.Text = login;
                if (!admin.Equals("Yes"))
                {
                    m.label1.Visible = false;
                    m.label2.Text = "Сведения о себе";
                    m.label2.Location = new Point(60, 344);

```

```

        m.panel5.Location = new Point(96, 368);
    }
    else
        m.label2.Text = "Сторудники";
    return;
}
else
{
    MessageBox.Show("Пароль введён неверно!", "Внимание!",
    MessageBoxButtons.OK, MessageBoxIcon.Information);
    textBox2.Text = "";
    textBox2.Focus();
    return;
}
}
}
catch (Exception ex)
{
    MessageBox.Show(ex.Message, "Внимание!",
    MessageBoxButtons.OK, MessageBoxIcon.Information);
}
}

```

```

private void button2_Click(object sender, EventArgs e)
{
    textBox1.Text="";
    textBox1.Text = "Введите логин";
    textBox1.ForeColor = Color.Gray;
    textBox2.Text="";
    textBox2.Text = "Введите пароль";
    textBox2.UseSystemPasswordChar = false;
    textBox2.ForeColor = Color.Gray;
}

```

```

private void button1_MouseMove(object sender, MouseEventArgs e)
{
    button1.BackColor = Color.Blue;
    button1.ForeColor = Color.White;
}

```

```

private void button2_MouseMove(object sender, MouseEventArgs e)
{
    button2.BackColor = Color.Blue;
    button2.ForeColor = Color.White;
}

```

```

}

private void label1_MouseMove(object sender, MouseEventArgs e)
{
    label1.ForeColor = Color.Blue;
}

private void label1_MouseLeave(object sender, EventArgs e)
{
    label1.ForeColor = Color.White;
}

private void button1_MouseLeave(object sender, EventArgs e)
{
    button1.BackColor = SystemColors.Control;
    button1.ForeColor = Color.Black;
}

private void button2_MouseLeave(object sender, EventArgs e)
{
    button2.BackColor = SystemColors.Control;
    button2.ForeColor = Color.Black;
}

private void textBox1_Enter(object sender, EventArgs e)
{
    if (textBox1.Text == "Введите логин")
    {
        textBox1.Text = "";
        textBox1.ForeColor = Color.Black;
    }
}

private void textBox1_Leave(object sender, EventArgs e)
{
    if (textBox1.Text == "")
    {
        textBox1.Text = "Введите логин";
        textBox1.ForeColor = Color.Gray;
    }
}

private void textBox2_Enter(object sender, EventArgs e)
{

```



```

        if (textBox2.Text == "Введите пароль")
        {
            textBox2.Text = "";
            textBox2.UseSystemPasswordChar = true;
            textBox2.ForeColor = Color.Black;
        }
    }

    private void textBox2_Leave(object sender, EventArgs e)
    {
        if (textBox2.Text == "")
        {
            textBox2.Text = "Введите пароль";
            textBox2.UseSystemPasswordChar = false;
            textBox2.ForeColor = Color.Gray;
        }
    }

    private void label1_Click_1(object sender, EventArgs e)
    {
        Application.Exit();
    }

    Point lastPoint;
    private void label2_MouseMove(object sender, MouseEventArgs e)
    {
        if (e.Button == MouseButton.Left)
        {
            this.Left += e.X - lastPoint.X;
            this.Top += e.Y - lastPoint.Y;
        }
    }

    private void label2_MouseDown(object sender, MouseEventArgs e)
    {
        lastPoint = new Point(e.X, e.Y);
    }
}
}

using System;
using System.Drawing;

```

```

using System.Windows.Forms;

namespace Autorization
{
    public partial class Menu : Form
    {
        public Menu()
        {
            InitializeComponent();

            Autorization autor = new Autorization();
            private void label7_Click(object sender, EventArgs e)
            {
                this.Close();
                //Application.Exit();
            }

            private void label1_MouseMove(object sender, MouseEventArgs e)
            {
                label1.ForeColor = Color.Red;
                panel5.BackColor = Color.Red;
                this.panel5.Location = new Point(108, 321);
            }

            private void label2_MouseMove(object sender, MouseEventArgs e)
            {
                label2.ForeColor = Color.Red;
                panel5.BackColor = Color.Red;
                this.panel5.Location = new Point(103, 368);
            }

            private void label3_MouseMove(object sender, MouseEventArgs e)
            {
                label3.ForeColor = Color.Red;
                panel5.BackColor = Color.Red;
                this.panel5.Location = new Point(103, 413);
            }

            private void label7_MouseMove(object sender, MouseEventArgs e)
            {
                label7.ForeColor = Color.Red;
            }
        }
    }
}

```

```
private void label1_MouseLeave(object sender, EventArgs e)
{
    label1.ForeColor = Color.White;
    panel5.BackColor = Color.White;
}
```

```
private void label2_MouseLeave(object sender, EventArgs e)
{
    label2.ForeColor = Color.White;
    panel5.BackColor = Color.White;
}
```

```
private void label3_MouseLeave(object sender, EventArgs e)
{
    label3.ForeColor = Color.White;
    panel5.BackColor = Color.White;
}
```

```
private void label7_MouseLeave(object sender, EventArgs e)
{
    label7.ForeColor = Color.White ;
}
```

```
private void label1_Click(object sender, EventArgs e)
{
    Registration reg = new Registration();
    reg.Show();
}
```

Point lastPoint;

```
private void label4_MouseMove(object sender, MouseEventArgs e)
{
    if (e.Button == MouseButtons.Left)
    {
        this.Left += e.X - lastPoint.X;
        this.Top += e.Y - lastPoint.Y;
    }
}
```

```
private void label4_MouseDown(object sender, MouseEventArgs e)
{
    lastPoint = new Point(e.X, e.Y);
}
```

```

private void label2_Click(object sender, EventArgs e)
{
    Users_Admin adm = new Users_Admin();
    Users us = new Users();
    if (label5.Text.Equals("Yes"))
    {
        adm.login = label6.Text;
        adm.Show();
    }
    else
    {
        us.login = label6.Text;
        us.Show();
    }
    this.panel5.Location = new Point(103, 368);
}

private void label3_Click(object sender, EventArgs e)
{
    Registratura reg = new Registratura();
    reg.Show();
}

private void label9_Click(object sender, EventArgs e)
{
}
}
}
}

```