ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Сәтбаев университеті

Ақпараттық және телекоммуникациялық технологиялар институты

Киберқауіпсіздік, ақпаратты өңдеу және сақтау кафедрасы

Есенқызы Ақбота

«Университетке талапкерлерді тіркеудің ақпараттық жүйесін құру»

ДИПЛОМДЫҚ ЖҰМЫС

5В070300 – «Ақпараттық жүйелер» мамандығы

Алматы 2020

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Сәтбаев университеті

Ақпараттық және телекоммуникациялық технологиялар институты

Киберқауіпсіздік, ақпаратты өңдеу және сақтау кафедрасы

ҚОРҒАУҒА ЖІБЕРІЛДІ
КАӨжС кафедра меңгерушісі,
тех.ғыл.канд, ассоц.
профессор
_____ Н.А.Сейлова
«___» _____2020 ж.

ДИПЛОМДЫҚ ЖҰМЫС

Тақырыбы: «Университетке талапкерлерді тіркеудің ақпараттық жүйесін құру»

5В070300 – «Ақпараттық жүйелер» мамандығы

Орындаған:
Есенқызы А.
Ғылыми жетекші:
сеньор-лектор, т.ғ.к., доцент
_____Ш.М.Байматаева
«___» _____2020 ж.

Алматы 2020

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Сәтбаев университеті

Ақпараттық және телекоммуникациялық технологиялар институты

Киберқауіпсіздік, ақпаратты өңдеу және сақтау кафедрасы

БЕКІТЕМІН
КАӨжС кафедра меңгерушісі,
тех.ғыл.канд, ассоц.
профессор
_____ Н.А.Сейлова
«___»_____2020 ж.

Дипломдық жұмысты орындауға
ТАПСЫРМА

Білім алушы: Есенқызы Ақбота
Тақырыбы: «Университетке талапкерлерді тіркеудің ақпараттық жүйесін құру»

Университет Ректорының 2020 жылғы «27» 01 №762 -б бұйрығымен бекітілген
Аяқталған жұмысты тапсыру мерзімі 2020 жылғы « 20» мамыр
Дипломдық жұмыстың бастапқы берілістері: диплом алдындағы практикалық жұмыс қорытындысы, тақырып бойынша әдебиеттерге шолу нәтижелері,теориялық мәліметтердің жиыны
Дипломдық жұмыста қарастырылатын мәселелер тізімі:
а)қойылған мәселенің қазіргі жағдайын пайымдау
ә)ақпараттық қамтаманы құру
б)программалық қамтаманы құру
Сызбалық материалдар тізімі: Power Point бағдарламасындағы слайдтар
Сызба материалдар: 15 слайдпен көрсетілген
Ұсынылатын негізгі әдебиет: 15 атау

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: Есенкызы Акбота

Название: Разработка информационной системы учета абитуриентов вуза

Координатор: Шолпан Байматаева

Коэффициент подобия 1:6,4

Коэффициент подобия 2:1,8

Замена букв:5

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: Есенкызы Акбота

Название: Разработка информационной системы учета абитуриентов вуза

Координатор:Шолпан Байматаева

Коэффициент подобия 1:6,4

Коэффициент подобия 2:1,8

Замена букв:5

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 дайындау

КЕСТЕСІ

Бөлімдер атауы, қарастырылатын мәселелер тізімі	Ғылыми жетекші мен кеңесшілерге көрсету мерзімдері	Ескерту
Мәселенің қазіргі жағдайына шолу және оны талдау		
Ақпараттық қамтаманы құру		
Программалық қамтаманы құру		

Дипломдық жұмысының бөлімдерінің кеңесшілері мен норма бақылаушыларының аяқталған жобаға қойған **қолтаңбалары**

Бөлімдер атауы	**Кеңесшілер, аты, әкесінің аты, тегі (ғылыми дәрежесі,атағы)**	**Қол қойылған күні**	Қолы
Норма бақылаушы	Дүйсенбаева А.Н. Лектор, тех-ғыл магистрі		
Программалық қамтама	Кабдуллин М.А., ассистент		

Ғылыми жетекші _____Ш.М.Байматаева

Тапсырманы орындауға алған білім алушы_____Есенқызы Ақбота

Күні « 20 » мамыр 2020 ж.

АҢДАТПА

Дипломдық жұмыста веб технологиялар негізінде талапкерлерді тіркеудің ақпараттық жүйесін әзірлеу карастырылады. Осыған байланысты қойылған мақсатқа жету үшін келесі міндеттер орындалуы тиіс:

– ұқсас жүйелерге шолу жасау;
– ақпараттық жүйелер негізін зерттеу;
– мәліметтер базасын жобалау;
– бағдарламалық қамтаманы құру;

Осы мақсатқа сәйкес C# негізінде талапкерлерді тіркеудің ақпараттық жүйесі әзірленеді.

Зерттеу объектісі. C# тілінде жазылған талапкерлерді тіркеудің ақпараттық жүйесі.

Мәселені талдау дәрежесі. Дипломдық жобаның тақырыбын зерттеу барысында отандық және шетелдік ғылыми еңбектері мен оқулықтар, электронды басылымдардағы ғылыми мақалалар қолданылды.

Зерттеу тәжірибесі: Салыстырмалық, аналитикалық, логикалық, құрылымдық жүйе, функционалдық жүйе, жалпылау тәжірибесі.

Дипломдық жұмыстың жаңалығы: C# негізінде жазылған талапкерлерді тіркеудің ақпараттық жүйесін әзірлеу тиімді құрылымы.

АННОТАЦИЯ

Дипломная работа предусматривает разработка информационной системы учета абитуриентов на основе веб-технологий. В связи с этим для достижения этой цели необходимо выполнить следующие задачи:

- обзор подобных систем;
- изучить основы информационных систем;
- дизайн базы данных;
- разработка программного обеспечения;

Для этого будет разработана информационная система учета абитуриентов на основе C#.

Объект исследования. Написанная C# информационная система учета абитуриентов.

Степень анализа проблемы. При изучении темы дипломного проекта использованы отечественные и зарубежные научные труды и учебники, научные статьи в электронных изданиях.

Опыт исследования: сравнительная, аналитическая, логическая, структурная система, функциональная система, опыт обобщения.

Новизна дипломной работы: эффективная структура для разработки информационной системы регистрации заявителей на основе C#.

ABSTRACT

Thesis includes the development of an information system for the registration of applicants based on web technologies. In this regard, to achieve this goal, it is necessary to perform the following tasks:

- review of such systems;
- learn the basics of information systems;
- database design;
- software development;

For this, an information system for the registration of applicants based on C# will be developed.

Object of study. Written by C# information system for accounting applicants.

The degree of analysis of the problem. When studying the topic of the graduation project, domestic and foreign scientific works and textbooks, scientific articles in electronic publications were used.

Research experience: comparative, analytical, logical, structural system, functional system, generalization experience.

The novelty of the thesis: an effective framework for developing an information system for registration of applicants based on C#.

МАЗМҰНЫ

Кіріспе 8
1 Ақпараттық жүйені талдау 9
1.1 Ақпараттық жүйеге жалпы шолу 9
1.2 Дизайн технологиясын таңдау негіздемесі 9
1.3 Қолдау түрлері бойынша жобалық шешімдерді негіздеу 10
1.4 Есеп қойылымы 11
2 Ақпараттық жүйені дамыту 12
2.1 Ақпараттық жүйенің жобалау әдіснамасының негіздері 12
2.2 Деректер базасының физикалық құрылымы 17
3 Программалық қамтаманы құру 19
3.1 Программалау тілін таңдауды негіздеу 19
3.2 Жалпы мағлұматтар 19
3.3 Функционалдық тағайындалуы 19
3.4 Қажетті техникалық жабдықтар 23
3.5 Шақыру және жүктеу 20
3.6 Кіріс мәліметтер 20
3.7 Шығыс мәліметтер 20
3.8 Бакылау мысалын сипаттау 20
Қорытынды 28
Пайдаланылған әдебиеттер тізімі 29
А Қосымшасы 30

КІРІСПЕ

Соңғы уақытта компьютерлер жоғары жылдамдықпен тарала бастады. Талапкерлердің ақпараттық іздестіру жүйесі жоғары оқу орындарының қабылдау комиссияларының мүшелеріне арналған. Мектеп бітірушілердің 75% - ы жыл сайын жоғары оқу орындарына түсуге тырысады. Талапкерлердің әрқайсысы үшін жеке іс жүргізу керек, мәліметтерді рәсімдеу, барлық қажетті құжаттарды қабылдау қажет. Содан кейін барлық талапкерлердің істері таратылып, жүйелі түрде ұйымдастырылады. Қабылдау комиссиясы өз жұмысында бұл құжаттарға бірнеше рет сілтеме жасайды, бұл уақытты қажет етеді. Абитуриенттің іздеу жүйесі оны үнемдейді. Бұл жүйе студенттерге, мамандыққа қатысты ақпаратты осы оқу орнында алуға, тез қарауға және басып шығаруға мүмкіндік береді. Олар файлға автоматты түрде жазылады және құжатты басып шығара алады. іздеу жүйесі диалогтық режимде жұмыс істейді. Барлық енгізулерді редакциялауға, толықтыруға, жоюға болады. Барлық әрекеттер өте ыңғайлы мәзірдің көмегімен жүзеге асырылады, ол жүйені пайдалану кезінде пайдаланушыға қажеттінің барлығын қамтамасыз етеді.

Компьютерлік технологияның жетілуімен бағдарламашыдан деректерді өңдеудің неғұрлым күрделі құралдарын тез әзірлеуді және сол мәліметтерді өңдей білуді талап ететін көптеген жүйелер пайда болады.

Microsoft SQL Server дерекқоры өте жоғары жылдамдыққа ие. Пәрмендер мен функциялар жиынтығы мәліметтерді ұсыну мен өңдеудің барлық заманауи талаптарына сәйкес келеді. Ең ыңғайлы интерфейсті қолдануға болады. Түрлі көп деңгейлі мәзірлер, тінтуір мен терезе қолдауы бар. Деректер кесте түрінде ұсынылуы мүмкін. FoxPro бағдарламалау тілдерінің болуы мәліметтерді өңдеудің күрделі, тапсырмаға негізделген жүйелерін құруға мүмкіндік береді.

Тиімді басқарусыз қазіргі өмірді елестету мүмкін емес. Оның маңызды категорияларының бірі - ақпаратты өңдеу жүйесі. Бүгінгі таңда бірде-бір кәсіпорын немесе мекеме жаңа компьютерлік технологияларды енгізусіз жасай алмайды. Әрбір мекеменің өз бөлімшелері бар, олардың қызметтері автоматтандыруды қажет етеді.

Сәтбаев университетінде негізгі құрылымдық бөлімдер студенттерді күндізгі жұмысқа қабылдайтын және студенттерді мамандықтар мен топтар бойынша бөлетін және студенттер туралы жеке ақпаратты сақтайтын қабылдау комиссиясы және бөлімдер болып табылады. Дипломдық жұмыстың мақсаты автоматтандырылған ақпараттық жүйесін құру, тиімділігін арттыру және жоғары кәсіптік оқу орындарындағы қабылдау комиссиясының жұмысын жеңілдету болып табылады.

Іс-әрекеттерді оңтайландыру және автоматтандыру көбінесе автоматтандырылған жүйелерді енгізу жүзеге асырылатын бөлімшелердің құрылымдық және логикалық қайта құрылуына әкеледі, нәтижесінде олардың қызметінің тиімділігі артады.

1 Ақпараттық жүйені талдау

1.1 Ақпараттық жүйеге жалпы шолу

Таңдалған жоғары оқу орнына талапкерлермен қабылдау қабылдау комиссиясына құжаттарды тапсырудан басталады.

Қабылдау комиссиясы - келесі оқу жылына арналған іріктеу науқанын жүзеге асыратын жоғары немесе орта оқу орнының бөлімшесі. Ол студенттер жиынтығын ұйымдастырады, талапкерлерді оқу орнының ережелері мен ережелерімен, облыстар мен мамандықтарды аккредиттеу туралы құжаттамамен, қабылдау критерийлерімен таныстырады.

Үміткер құжаттарды қабылдау кезінде қабылдау комиссиясымен тікелей байланысқа түседі: құжаттарды тапсырған кезде, емтихан парағын алған кезде, өтініш берген кезде, сондай-ақ құжаттарды қабылдау немесе қайтару процедуралары кезінде. Қазіргі уақытта көптеген университеттерде өтініштер мен құжаттарды онлайн режимінде тапсыру мүмкіндігі бар, бірақ бұл ерекше.

Комиссия құрылымдалған және белгіленген міндеттерді орындайтын белгілі бір бөлімдерден тұрады. Олар оқу орнының сипаттамаларын - оның географиясын қамтуды, кафедралардың санын, көлемін, мүмкіндіктерін ескере отырып өзгеруі мүмкін.

Көптеген жағдайларда университеттің осы бөлімшесінің құрамына кіретіндердің тізімі келесідей: қабылдау комиссиясының төрағасы, оның қызметіне, заңдылығына, белгіленген мерзімге және тексеру сандарының сақталуына жауап береді, сонымен қатар жауапкершілікті бөледі.

- көбінесе университет ректоры;
- комиссия төрағасының орынбасары - қабылдау комиссиясын басқара алатын, бірақ төрағадан гөрі проректор;
- бөлім басшылары;
- бұйрықтар дайындайтын және отырыстар хаттамаларын жүргізетін, өтініш берушілердің рейтингін бақылайтын, барлық ақпаратты жариялап, жариялайтын хатшылық;
- пәндік емтихан комиссияларының төрағалары, түсу емтихандарын құрастырушылар, емтиханға дайындалу және бағалау критерийлері бойынша әдістемелік әдебиеттер жасау;
- профессор-оқытушылар құрамының өкілдері.

Сондай-ақ композицияға мыналар кіруі мүмкін:

- Қазақстан азаматтарын қабылдауды ұйымдастыру бөлімі;
- шетелдік азаматтарды қабылдауды ұйымдастыру бөлімі;
- кәсіби бағдар беру бөлімі;
- ЖОО дейінгі дайындық бөлімі;
- ұйымдастыру мәселелері жөніндегі мамандар;
- апелляциялық комиссия.

Қабылдау комиссиясының төрағасы оқу орнының директоры немесе ректорымен тағайындалуы мүмкін және комиссияның бүкіл жұмысына толық

жауап береді. Ол қабылдау және апелляциялық комиссияларды басқарады, студенттерді ауыстыру және қалпына келтіруді басқарады, барлық түсу емтихандарының нысандарын бекітеді, ақпаратты бақылайды және оны өз қолымен растайды. Қабылдау комиссиясының қосымша құрамы көбінесе жоғары оқу орнының көлеміне, оның ерекшеліктеріне және оқу жылына қабылдау жоспарланған үміткерлер санына байланысты. Оқу орнында шетелдік азаматтардың білім алу мүмкіндіктерінің болуы қабылдау комиссиясына қосымша мамандар енгізу қажеттілігін тудырады. Хатшылық құрамын кеңейту қажет, ал оқу процесі қазақ, ағылшын және орыс тілдерінде жүргізілсе де, аудармашы керек.

Қабылдау комиссиясы тиісті жоғары оқу орны мен Білім министрлігіне есеп береді. Қызметінің соңында ол жұмыс нәтижелерін талдау және статистикалық өңдеу үшін құжаттаманы ұсынады.

Мұндай құжаттар ЖОО-ға түсу ережелері мен жоспары, комиссия құрамы мен оның ішкі құрылымдары туралы бұйрықтар, қабылдау комиссиясының отырыстарының хаттамалары, емтихан парақтары, жеке құжаттар және талапкерлердің тізімі, қабылдау туралы бұйрық.

Іріктеу комиссиясы түрлі жұмыстарды атқарады, ондағы мамандар білімді және жан-жақты, пікірлес және көмектесуге дайын болуы керек. Комиссияның міндеттеріне:

- қабылдау үшін ақпараттық қолдауды құру;
- буклеттер;
- анықтамалық материалдар;
- Үміткерлерге арналған тізімдер мен ақпараттық стендтер.
- білім беру қызметін ұйымдастыруға, оқуға түсуге байланысты мәселелер бойынша кеңес алуға келген үміткерлердің қоңыраулар, электрондық пошталар, ауызша сұрақтарына жауаптар;
- келесі оқу жылына талапкерлерді қабылдау ережелері мен ережелерін әзірлеу. Құжаттарды қабылдау, қабылдау кезінде қажет пакеттің құрамы туралы кеңестер.
- түсу емтихандарын немесе тестілеу нәтижелерін ұйымдастыру және өңдеу;
- талапкерлерді жарғымен, аккредиттеуімен, білім беру қызметін жүзеге асыру құқығына лицензиямен, осы оқу орнына қабылдау шарттарымен таныстыру;
- Кәсіби бағдар беру жұмыстарын жүргізу. ЖОО-ға түсу немесе қабылданбау туралы шешім.

Оқуға түсу емтихандары алдында қабылдау комиссиясы талапкерлерді олардың кестесі, қабылдау тестілері, осы ЖОО-ға қабылдау ережелері, аккредитацияға сәйкес облыстар мен мамандықтардың тізімі, емтихан нәтижелерін бағалау жүйесі мен өлшемдері, өтініштерді беру және қарау шарттары туралы ақпараттандырады.

Сондай-ақ, қабылдау комиссиясы талапкерлердің жазбаша емтихандарын шифрлайды және шешеді, өз отырыстарының хаттамаларын жасайды және жеке

іс жүргізудің дұрыстығын бақылайды. Комиссияның міндетіне сонымен қатар қабылдау немесе түсу емтихандарына дайындық курстарын ұйымдастыру, осы оқу орнына қатысты олимпиадалар мен жарыстардың нәтижелерін өңдеу кіреді және қабылдау туралы қорытынды шешімге әсер ете алады.

Қабылдау комиссиясының тағы бір міндеті - құжаттарды қабылдау мен емтихандық тестілерді қабылдаудың басталу және аяқталу мерзімдерін белгілеу. Үміткерді білім беру ұйымына қабылдау немесе тіркеу туралы шешімді қабылдау комиссиясы тең көпшілік дауыспен қабылдайды. Маңызды аспект объективтілік пен конструктивтілік, рейтингке сәйкестік, қабылдаудың дәйектілігі және заңды құжаттардың сақталуы болып табылады.

Қабылдау комиссиясы байланыстырушы буын болып табылады, өйткені ол өтініш берушіні жаңа бұйрықтармен, адамдармен және білім алатын оқу орны туралы ақпаратпен таныстырады. Іріктеу комиссиясында өтініш беруші қызығушылықты білдіретін мамандық бойынша бюджеттік орындардың саны, келісім-шарт түрінде оқу құны, келісімшарттан бюджеттік оқыту түріне өту шарттары туралы ақпарат алады.

Комиссияға қатысушылардың көмегімен өтініш берушіге кафедралар мен бөлімдердің қызметкерлерімен, қызығушылық танытатын мамандықтардың профессорлық-оқытушылық құрамымен танысуға, сонымен қатар олардың біліктіліктері, сіңірген еңбектері мен ерекшеліктері туралы білуге мүмкіндігі бар.

Ең тиімді әдіс - бұл құжаттар жиынтығы туралы алдын-ала веб-сайтта немесе жеке-жеке ақпарат алу, сондықтан кейінірек келген кезде сіз берілген құжаттар мен мүмкін болатын қателіктер туралы ойламайсыз, бірақ келесі әрекеттер туралы ұсыныстар алуыңыз керек.

Күнделікті әр мамандыққа сәйкес келіп түскен өтінімдер саны туралы ақпаратты жаңарту, қабылдау комиссиясы үміткерге бәсекелестіктің динамикасын бақылау, олардың жинаған баллдары бойынша қабылдау мүмкіндігін есептеп отыру мүмкіндігін береді.

Автоматтандыру және ақпараттық жүйелерді құру қазіргі кезде техногендік қоғамның көп ресурстарды қажет ететін салаларының бірі болып табылады. Осы саланың белсенді дамуының себептерінің бірі - автоматтандыру адам мен қоғам өмірінде маңызды рөл атқаратын процестерді түбегейлі өзгерту үшін негіз болып табылады. Ақпараттық жүйелердің көптеген түрлері бар: мәліметтерді өңдеу жүйелері, басқарудың ақпараттық жүйелері, маркетинг жүйелері, бухгалтерлік жүйелер және әртүрлі ұйымдарда қолданылатын басқалары.

Ақпараттық жүйелердің көптеген түрлері оларды жасауға арналған көптеген әдістемелер мен технологиялардың пайда болуына себеп болды.

Ақпараттық жүйе - бұл қоршаған орта, оның құрамдас бөліктері болып компьютерлер, компьютерлік желілер, бағдарламалық өнімдер, мәліметтер базасы, адамдар, әртүрлі техникалық және бағдарламалық коммуникациялар болып табылады. Ақпараттық жүйенің негізгі мақсаты - ақпаратты сақтау мен беруді ұйымдастыру.

Ақпараттық қолдаудың анықтамасы автоматтандырылған басқару жүйелерін құру процесінде пайда болды.

Ақпараттық қамтамасыздандыруға енгізу, шығару, аралық, есептерді шешуге арналған бағдарламалар кіреді. Нормативті-анықтамалық ақпараттарды жіктеуге және кодтауға бағытталған жүйелерден тұратын қосымша аппарат болып келеді.

Ақпараттық қамтамасыз етудің негізгі талаптарының бірі - ақпараттық базадағы мәліметтердің сенімділігі.

Ақпараттық деректер базасындағы талап етілетін сенімділік деңгейі мәліметтермен жұмыс істеудің барлық кезеңдеріне қолданылатын жоғары бақылау деңгейімен қамтамасыз етіледі.

Деректерді өңдеу технологиясының ерекшелігі осындай факторларға тікелей байланысты: ақпаратты сақтау құрылғыларының болуы, қолданушымен диалог жағдайында жұмыс істеу, ақпаратты өңдеуде қағаз технологияларын пайдалануды болдырмау.

Технологиялық операцияларға мыналар жатады:

– мәліметтерді енгізу, шығару;
– бағдарламаны жүктеу;
– ақпаратты бақылау және түзету мүмкіндігі;
– ақпараттық массивтерді қалыптастыру;
– анықтамалық-ақпараттық қолдау.

Бастапқы ақпаратты тіркеудің бірнеше әдісі бар:

– деректі;
– автоматты;
– машина тасымалдаушысында тіркеумен бірге деректі.

Деректер арасындағы қатынасты қалыптастыру әдісіне сәйкес желілік, реляциялық және иерархиялық модельдер бар. Желілік және иерархиялық модельдер жалпы сипаттамалары бар мәліметтер арасындағы байланысты қажет етеді. Иерархиялық модель аясында осы түрдегі байланыстарды ағаш сызбасы түрінде бейнелеуге болады, желілік модельде «барлығымен» қарым-қатынастың болуы мүмкін. Реляциялық модель - бұл байланыстырылған кесте түрінде мәліметтерді ұсынудың қарапайым және ең танымал түрі.

Қазіргі уақытта реляциялық жүйелер дербес компьютерлердің техникалық мүмкіндіктеріне барынша сәйкес келеді. Осы ДҚБЖ-нің жылдамдық сипаттамалары ақпаратқа жедел қол жетімділіктің арнайы әдістерімен, атап айтқанда мәліметтер базасын индекстеумен қамтамасыз етілген. Ақпараттық жүйені құру үшін Microsoft SQL Server DBMS-ті қарастыруға болады, оның құрамында дерекқорды басқаруға арналған маңызды құралдар бар.

Бағдарламалық жасақтама - мәліметтерді өңдеу жүйесінің жұмыс істеу процесіне байланысты бағдарламалар, құжаттар, процедуралар мен ережелер жиынтығы болып табылады.

Бағдарламалық жасақтама әдетте келесі өлшемдерге сәйкес жіктеледі:

Бағдарламалық жасақтаманы пайдалану және тарату әдісі бойынша мыналарға бөлуге болады:

– ашық;
– жабық;
– тегін.

Тағайындау бойынша бағдарламалық жасақтама келесіге бөлінеді:

– қолданбалы;
– жүйелік;
– аспаптық.

Тегін бағдарламалық қамтамасыздандыру үшін оны үйде, мектептерде, университеттерде, кеңселерде, сондай-ақ мемлекеттік және коммерциялық мекемелерде ешқандай шектеусіз таратуға, орнатуға және қолдануға болады.

1.2 Дизайн технологиясын таңдау негіздемесі

Дизайн технологиясын таңдаудағы негізгі факторларды келесі деп атауға болады:

– қосымшаны жасау жылдамдығы;
– әзірленген қосымшаның жоғары өнімділігі;
– әзірленген қосымшаның компьютерлік ресурстарға төмен талаптары;
– өз ресурстарыңыздың көмегімен жаңа компоненттер мен құралдарды жасау мүмкіндігі;
– масштабталу;
– нысандар иерархиясын ыңғайлы зерттеу.

Ақпараттық жүйені жобалау құралы ретінде Microsoft Visio таңдалды, ал оны жүзеге асыру үшін - SQL-де кеңінен қолданылатын жалпыға бірдей ашық көздер бағдарламалау тілі және реляциялық мәліметтер базасы.

Бұл ақпараттық жүйе клиент-сервер технологиясын қолданады. Сервер жағында Microsoft визуалдық студиясы және DBMS Microsfot SQL сервері орнатылуы керек. Қажет болған жағдайда МҚБЖ тізімін кеңейтуге болады.

1.3 Қолдау түрлері бойынша жобалық шешімдерді негіздеу

Ақпараттық жүйенің құрылымы - бұл оның ішкі бөліктері деп аталатын ішкі бөліктерінің жиынтығы. Шағын жүйе - бұл белгілі бір ерекшелігімен ерекшеленетін жүйенің бөлігі.

Ақпараттық жүйенің жалпы құрылымын, көлеміне қарамастан, ішкі жүйелер жиынтығы ретінде қарастыруға болады. Бұл жағдайда олар жіктеудің құрылымдық ерекшелігі туралы айтады, ал ішкі жүйелер тірек деп аталады. Осылайша, кез-келген ақпараттық жүйенің құрылымын қосалқы ішкі жүйелер жиынтығымен ұсынуға болады. Қосалқы ішкі жүйелер арасында, әдетте, ақпараттық, техникалық және бағдарламалық қамтамасыздандыру ерекшеленеді.

Техникалық қолдау дегеніміз - ақпараттық жүйенің жұмысына арналған техникалық құралдардың жиынтығы, сонымен қатар осы құралдар мен технологиялық процестерге қатысты құжаттама.

Техникалық құралдар кешеніне мыналар кіреді:

– кез-келген модельдегі компьютерлер;

– ақпаратты жинауға, жинақтауға, өңдеуге, беруге және шығаруға арналған құрылғылар;

– деректерді беру құрылғылары мен байланыс желілері;

– кеңсе жабдықтары және автоматты ақпарат іздейтін құрылғылар;

– пайдалану материалдары және т.б.

Құжаттама техникалық жабдықты алдын-ала таңдауды, олардың жұмысын ұйымдастыруды, мәліметтерді өңдеудің технологиялық процесін және технологиялық жабдықты жасайды. Құжаттарды үш топқа бөлуге болады:

– техникалық қолдаудың мемлекеттік және салалық стандарттары;

– техникалық қолдауды дамытудың барлық кезеңдеріне арналған техниканың жиынтығы;

– нормативті - анықтамалық, техникалық қолдау үшін есептеулер жүргізілуі.

Бүгінгі таңда техникалық қолдауды ұйымдастырудың екі негізгі формасы бар: орталықтандырылған және жартылай немесе толық орталықтандырылмаған.

Орталықтандырылған техникалық қолдау ақпараттық жүйеде ірі компьютерлер мен компьютерлік орталықтарды пайдалануға негізделген.

Техникалық құралдарды орталықсыздандыру тікелей жұмыс орындарында дербес компьютерлерге функционалды ішкі жүйелерді енгізуді қамтиды. Сәтпаев университетіндегі ақпараттық жүйе арнайы сервермен жергілікті желіге біріктірілген.

Жергілікті желі мыналарды қамтамасыз етеді: желіге қосылған компьютерлердің пайдаланушылары деректерді, ұжымдық өңдеуді және осы пайдаланушылар арасында деректер алмасуды; бағдарламалық жасақтаманы, принтерлерді, модемдерді және басқа құрылғыларды ортақ пайдалану.

Бұл мәселені шешу үшін алдымен жергілікті желіге қосылған компьютерлер, ортақ принтерлер, сонымен қатар сканер қажет.

Техникалық құралдарды таңдау өлшемдері:

– жүйенің сенімділігі;

– жүйенің функционалды толықтығы;

– шығындарды азайту: аппараттық құралдар, қолданбалы жүйелер, жүйеге техникалық қызмет көрсету, жүйені құру.

Ақпараттық қолдаудың ішкі жүйесінің мақсаты - мәселелерді шешу үшін сенімді ақпаратты уақытылы қалыптастыру және жеткізу.

Ақпараттық қамтамасыз етудің негізі базалық процестердің мәліметтер базасы және мәліметтер сақтау болып табылады. Деректер базасында әр түрлі

ақпараттық қорлар болуы керек - логикалық, семантикалық және физикалық деңгейде өзара байланысты жүйелердің құрылымдық жиынтығы ретінде.

Ақпараттық қолдау - ақпаратты жіктеу мен кодтаудың бірыңғай жүйесінің жиынтығы, құжаттаманың біртұтас жүйелері, білім беру мекемесінде таратылатын ақпарат ағындарының схемалары, сонымен қатар мәліметтер базасын құру әдістемесі болып табылады.

Ақпараттық қолдауды құру үшін келесі қадамдарды іске асыру қажет:

– бүкіл ұйымды басқару жүйесінің мақсаттары, міндеттері, функциялары туралы нақты түсінік;

– ақпарат ағымының сызбалары түрінде талдау үшін ұсынылған, пайда болған сәттен бастап оны басқарудың әртүрлі деңгейлерінде қолдануға дейінгі қозғалысын анықтау;

– жұмыс процесінің жүйесін жетілдіру;

– жіктеу және кодтау жүйесінің болуы және қолданылуы;

– ақпараттың байланысын көрсететін тұжырымдамалық ақпараттық-логикалық модельдерді құру әдістемесін білу;

– заманауи техникалық қолдаудың болуын талап ететін электрондық тасымалдағыштарда ақпараттық массивтер құру.

Информациялық қамтамасыз ету машинада индикаторлардың толық жиынтығын, құжаттарды, жіктеушілерді, файлдарды, мәліметтер базасын, білім негіздерін, оларды пайдалану әдістерін, сондай-ақ кез-келген ақпаратты қанағаттандыру үшін белгілі бір жүйеде қабылданған ақпаратты ұсыну, жинақтау, сақтау, конверсиялау, беру тәсілдерін қамтиды. Ақпараттық қолдауды ұйымдастырудың жетекші бағыты мәліметтер базасы технологиясы болып табылады. Іс-шараларды ақпараттық қолдауды ұйымдастыруға бірқатар талаптар ұсынылады.

Олардың ішіндегі ең маңыздылары:

– нақты уақыт режимінде жұмыс жасайтын көптеген пайдаланушыларға қызмет көрсету;

– ақпарат алмасуды қамтамасыз ету;

– қабылдау комиссиясының ақпаратын сақтау және беру қауіпсіздігі;

– аппараттық құралдың істен шығуы жағдайында ақпараттың тұтастығын сақтау.

Бағдарламалық жасақтама - басқару мәселелерін шешуге қатысатын әр түрлі функциялардың өзара байланысты бағдарламаларының жиынтығы және олардың жұмысына қажетті бағдарламалық құжаттар болып саналады.

Бағдарламалық жасақтама ақпараттық жүйенің маңызды құрамдас бөлігі болып табылады. Дамыған ақпараттық жүйенің тағдырындағы шешуші рөл жүйені және қолданбалы бағдарламалық жасақтаманы орынды және сәтті таңдау болып табылады.

Әзірленген бағдарламалық өнім платформадан тәуелсіз қосымша болып табылады, яғни ол тек бір қатаң операциялық жүйемен ғана жұмыс істемейді. Бүкіл даму кезеңі windows операциялық жүйесінде жүзеге асырылады. Бұл

операциялық жүйені үйде де, желілік шешімдер үшін де әртүрлі басылымдарда алуға болады.

Бағдарламалық жасақтама негізгі және қолданбалы бола алады. Ең бастысы, бастапқыда операциялық жүйемен орнатылған бағдарламалық жасақтама барлық жұмыс станциялары үшін стандартты болып табылады. Қолданбалы - бұл арнайы, мамандандырылған, оны белгілі бір пайдаланушылар ғана қолданады және оның сипаттамасында ерекшеленеді.

Деректерді технологиялық өңдеу деп бастапқы деректерді тіркеуден бастап нәтижені шығаруға дейінгі барлық кезеңдерді қамтитын өңдеудің белгілі бір әдістері мен құралдарын қолдана отырып, белгілі бір жүйелі жұмыстар жиынтығы жатады.

Осы жобаны іске асыра отырып, жоғары кәсіптік оқу орындарының қабылдау комиссиясының жұмысын жеңілдету жоспарлануда.

Орталықтандырылған ақпараттық жүйені өтініш берушілер, бөлімдер және т.б. туралы ақпаратты қабылдау және өңдеу мәселелерін шешу үшін қолдану, өңдеу жылдамдығы мен сапасын арттырып қана қоймайды, сонымен қатар осы тапсырмаларға кететін уақытты үнемдейді.

Ақпарат мұнай, газ, пайдалы қазбалар және т.б. сияқты ресурстардың дәстүрлі материалдық түрлерімен бірге қоғамның ең маңызды ресурстарының бірі болып табылады, демек оны өңдеуді материалдық ресурстарды өңдеу процестерімен ұқсастық технология ретінде қабылдауға болады. Ақпараттық технологиялар дегеніміз - бұл объектінің, процестің немесе құбылыстың жай-күйі туралы жаңа сапалы ақпарат алу үшін мәліметтерді жинау, өңдеу және таратудың құралдары мен әдістерінің жиынтығын қолданатын процесс.

Ақпараттық технологиялар - компьютерлерде сақталатын мәліметтерге қатысты әр түрлі деңгейдегі операцияларды, әрекеттерді, күрделілік деңгейлерін орындау үшін нақты реттелген ережелерден тұратын процесс. Ақпараттық технологияның негізгі мақсаты - бастапқы ақпаратты өңдеудің мақсатты әрекеттері нәтижесінде пайдаланушыға қажетті ақпаратты алу.

1.4 Есептің қойылымы

«Талапкер» автоматтандырылған ақпараттық жүйесі студенттердің қозғалысы мен қол жетімділігі туралы ақпаратты қорытындылауға арналған. Бағдарламаны пайдаланушылар - кадрлар бөлімінің қызметкерлері. Студенттер оқуға қабылдау туралы өтініштер негізінде қабылданады, онда оқу шарттары көрсетілген. Өтініштер түскен кезде алғашқы құжаттардың деректері өтінішті қабылдау журналында атауы, туған жылы, жынысы, үйінің мекен-жайы, ұлты, таңдаған мамандығы, қаржыландыру түрі, аймақ, бөлім, елді мекен және сертификаттың орташа белгісі көрсетілген жинақталады. Кадр бөлімі аптасына кемінде бір рет студенттердің жазбаларын тексереді.

Мәселенің тұжырымы:

– үміткердің өтініші туралы мәліметтер (коды, аты, туған жылы, таңдаған мамандығының коды, бірінші және екінші емтихандардағы бағалау);

– мамандықтар тізімі (бірінші емтиханның атауы, атауы, бірінші емтиханның түрі, бірінші емтиханның күні, екінші емтиханның тақырыбы, екінші емтиханның түрі, екінші емтиханның күні);

Ақпаратты іздестіру жүйесі осындай сұраулардың қорытындысын қамтамасыз етуі керек:

– берілген күні емтихан тапсырған талапкерлердің тізімі;

– бір немесе бірнеше мамандыққа үміткерлердің тізімі, алдымен мамандық бойынша (индекстелген), содан кейін фамилиясы бойынша сұрыпталуы;

Нұсқаулар:

– қамтамасыз ететін бағдарламалар жасау;

– экранда терезелер мен диалог құралдарын таңдай отырып, дерекқорды өңдеу режимдерінің мәзір жүйесін құру және қолдану;

– деректерді енгізу, өзгерту және қарау;

– шығарылатын құжаттарды басып шығару және файлға шығару.

Сәтпаев университетінің қабылдау комиссиясының қызметін талдау нәтижесінде әзірленген «абитуриент» автоматтандырылған жүйесін пайдалануды ескере отырып, оның жұмысын ұйымдастырудың келесі қағидаттары ұсынылады, онда келесі мәліметтер келтірілуі керек:

– оқу орны лицензияға сәйкес құжаттарды қабылдау туралы жариялайтын оқыту бағыттары мен мамандықтарының тізімі;

– келісім-шарт негізінде оқу ақысын төлеумен, әр бағытта және әр мамандық бойынша бірінші және кейінгі курстарға қабылдау орындарының саны;

– оқудың әр бағыты мен мамандықтары бойынша түсу тестілерінің тізімі, олардың бағдарламалары, сонымен қатар талапкерлердің білімін бағалау жүйесі, оның ішінде тестілеудің қысқартылған нәтижелері бойынша оқуға түсуге құқығы бар үміткерлер үшін кіру тесттерін өткізу атауы мен нысаны;

– республикалық бюджеттен қаржыландырылатын орындарға конкурс ұйымдастыру;

– келісім-шарт негізінде оқу ақысын төлейтін орындарға конкурс ұйымдастыру;

– әр өтініш беруші үшін жеке карта құру;

– студенттерді қабылдау құжаттарын толтыру;

– студенттерді қабылдау.

Мүмкін болатын қателіктерді жою және қабылдау процесін жедел тіркеу үшін автоматтандырылған жүйеде жоғары оқу орнына түсудің барлық мүмкін ережелерін ескере отырып, әртүрлі мүмкін өлшемдер бойынша үлгілерді алуға мүмкіндік беретін қуатты аналитикалық құрал болуы керек. Сондай-ақ, қабылдаудың тапсырыстарын қоса алғанда, есептің әртүрлі нысандарын автоматтандырылған құру мүмкіндігі болуы керек. Сонымен қатар, ұсынылған аналитикалық жүйе өтініш берушіге көп мүмкіндіктері болатын мамандықтарды анықтауға көмектесуі керек.

2 Ақпараттық жүйені дамыту

2.1 Ақпараттық жүйенің жобалау әдіснамасының негіздері

Білім беру ұйымдары мен реттеуші органдар желісінің ойдағыдай дамуының айқындаушы факторы басқару жүйесіндегі сапалы өзгеріс болып табылады. Заманауи ақпараттық технологиялардың даму тенденциялары экономиканың әртүрлі салаларында құрылған ақпараттық жүйелер күрделілігінің үнемі жоғарылауына алып келеді. Қазіргі заманғы ірі жобалар, ақпараттық жүйелер, әдетте, келесі ерекшеліктермен сипатталады:

– мәліметтер мен процестерді мұқият модельдеуді және талдауды қажет ететін сипаттаманың күрделілігі;

– жергілікті міндеттері мен функционалды мақсаттары бар тығыз өзара әрекеттесетін компоненттер жиынтығының болуы;

– кез-келген типтік жобалық шешімдер мен қолданбалы жүйелерді пайдалану мүмкіндігін шектейтін тікелей аналогтардың болмауы;

– қолданыстағы және жаңадан әзірленген қосымшаларды біріктіру қажеттілігі;

– бірнеше аппараттық платформаларда жұмыс жасау;

– біліктілік деңгейіне және белгілі бір құралдарды қолданудың қалыптасқан дәстүріне сәйкес әзірлеушілердің жекелеген топтарының бөлінбеуі және біртектілігі;

– жобаның ұзақ уақыттың біршама ұзақтылығы, бір жағынан, әзірлеу тобының мүмкіндіктері шектеулі, екінші жағынан, тапсырыс беруші ұйымның ауқымы және оның жеке бөлімшелерінің ақпараттық жүйелерді енгізуге әр түрлі дайындығы.

Жобаны сәтті жүзеге асыру үшін жобалау объектісі, ең алдымен, толық сипатталған, толық және дәйекті функционалды және ақпараттық жүйелердің ақпараттық модельдерін құру керек. Бүгінгі таңда ақпараттық жүйелерді жобалауда жинақталған тәжірибе көрсеткендей, бұл логикалық тұрғыдан күрделі, көп уақытты және уақытты қажет ететін жұмыс, оған тартылған мамандардың жоғары біліктілігін талап етеді. Алайда, осы уақытқа дейін ақпараттық жүйелерді жобалау негізінен интуитивті деңгейде өнерге, практикалық тәжірибеге, сараптамалық бағалауларға және ақпараттық жүйелер жұмысының сапасын эксперименттік тексерулерге негізделген бейресми әдістерді қолдана отырып жүргізілді. Сонымен қатар, ақпараттық жүйелерді құру және жұмыс істеу процесінде пайдаланушылардың ақпараттық қажеттіліктері өзгертілуі немесе нақтылануы мүмкін, бұл мұндай жүйелердің дамуы мен қызмет етуін одан әрі қиындатады.

Қазіргі жағдайда кәсіпорындарды тиімді басқару компьютерлік технологияны пайдаланбай мүмкін емес. Бағдарламалық жасақтама өнімі мен әзірлеуші компанияны дұрыс таңдау бухгалтерлік есепті автоматтандырудың бірінші және анықталатын кезеңі болып табылады. Қазіргі уақытта белгілі бір

тапсырмадан ақпараттық жүйені таңдау мәселесі стандартты процедураға ауысуда.

Ең тиімді компьютерлік жүйені таңдау туралы шешім қабылдау процедурасы көптеген отандық көшбасшылар үшін жаңа болып табылады және оның салдары бірнеше жыл бойы кәсіпорынға айтарлықтай әсер етеді. Кәсіпорынның талаптарына сәйкес келетін интеграцияланған ақпараттық жүйені қолдану менеджерге шығындарды азайтуға және жалпы алғанда кәсіпорын менеджментінің тиімділігін арттыруға мүмкіндік береді.

Деректерді өңдеу процесінің жекелеген компоненттері белгілі бір экономикалық объектіде ақпаратты өңдеудің барлық құралдарын «экономикалық ақпараттық жүйе» ұғымымен біріктіруге мүмкіндік беретін жоғары деңгейдегі ұйымдастырушылық және өзара байланысты деңгейге жетті.

Өтініш берушілердің есебін жүргізу бағдарламасын жасаңыз. Университетте жұмыс істеу үшін жүйені жобалау.

Нұсқаны көрсете отырып, басқару функциялары белгілерді қолдану арқылы жүзеге асырылады.

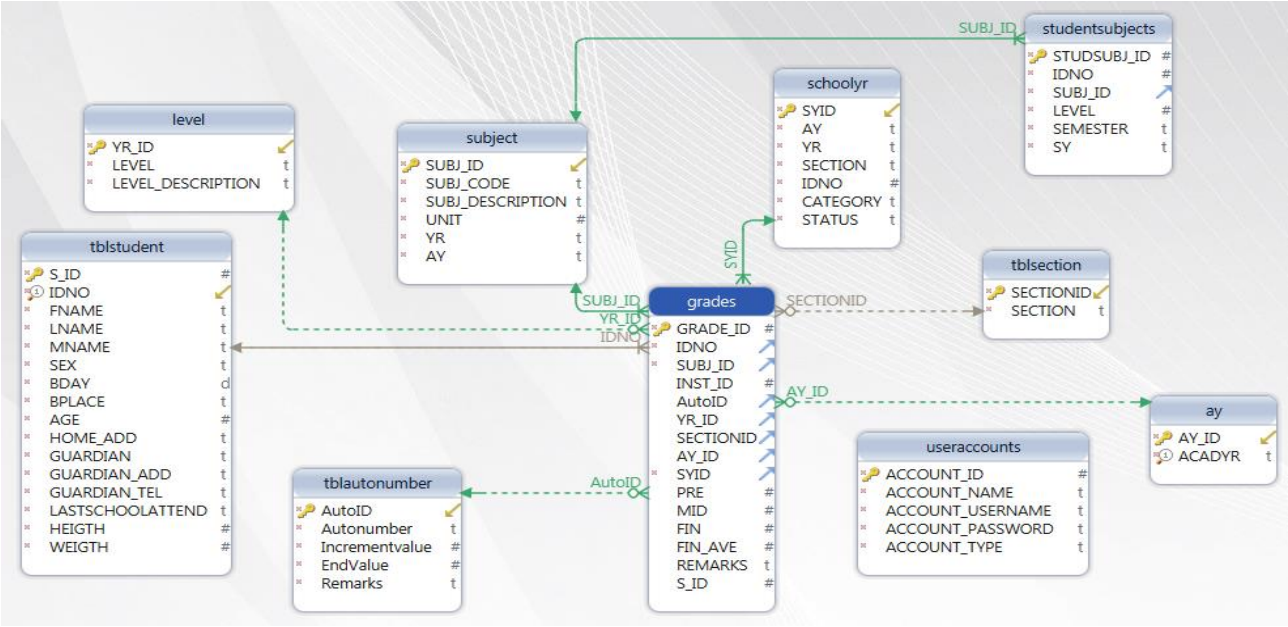
Қосымшаның интерфейсі - ақпаратты енгізу, редакциялау және шығару үшін ыңғайлы және бірнеше кестелерден қажетті мәліметтерді таңдауға арналған сұраныс жиынтығы үшін интерфейсті басқару элементтері бар форма.

Пішіндер - деректерді енгізудің қарапайым нысандары, сонымен қатар есептер шығару және диаграмма құру нысандары.

Қосымшаның функционалдылығы этикеткалар мен басқару батырмаларының болуымен қамтамасыз етіледі.

Қате жағдайлардың көпшілігі тиісті ерекшелік өңдеушілері арқылы қабылданады және өңделеді.

2.2 Деректер базасының физикалық құрылымы



2.1 сурет- Деректер базасының ER құрылымы

1 кесте – Ақпараттық жүйенің талапкерлердің оқу жылы ақпараттары орналасқан

Индекс	Өріс атауы	Деректер типі
*	AY_ID	int AUTOINCREMENT
*1	ACADYR	varchar(40)
Индекстер		
	pk_ay	ON AY_ID
1	Acadyr	ON ACADYR

AY кестесінде ақпараттық жүйенің талапкерлердің оқу жылы ақпараты сақталған. Кесте «AY_ID, ACADYR» деген өрістерінен тұрады. AY кестесінде «pk_ay, acadyr» деген индекс құрылған.

2 кесте – Ақпараттық жүйенің талапкерлердің арнайы код ақпараттары орналасқан

Индекс	Өріс атауы	Деректер типі
*	GRADE_ID	int AUTOINCREMENT
*	IDNO	int
*	SUBJ_ID	int
	INST_ID	int
	AutoID	int
	YR_ID	int
	SECTIONID	int
	AY_ID	int
*	SYID	int
	PRE	int
	MID	int
	FIN	int
Индекстер		
	pk_grades	ON GRADE_ID
Сыртқы кілттер		
	fk_grades_schoolyr	(SYID) ref schoolyr (SYID)
	fk_grades_subject	(SUBJ_ID) ref subject (SUBJ_ID)
	fk_grades_tblautonumber	(AutoID) ref tblautonumber (AutoID)
	fk_grades_tblsection	(SECTIONID) ref tblsection (SECTIONID)
	fk_grades_level	(YR_ID) ref level (YR_ID)

2 кестенің жалғасы

	fk_grades_ay	(AY_ID) ref ay (AY_ID)
	fk_grades_tblstudent	(IDNO) ref tblstudent (IDNO)

Grades кестесінде ақпараттық жүйенің әр талапкердің арнайы код мәліметтері сақталған. Кесте «GRADE_ID, IDNO, SUBJ_ID, INST_ID, AutoID, YR_ID, SECTIONID, AY_ID, SYID, PRE, MID, FIN, FIN_AVE, REMARKS, S_ID» деген өрістерінен тұрады. Grades кестесінде «pk_grades» деген индекс құрылған. Tblbranch кестесі fk_grades_schoolyr, fk_grades_subject, fk_grades_tblautonumber, fk_grades_tblsection, fk_grades_level, fk_grades_ay, fk_grades_tblstudent деген сыртқы кілтімен байланысқан.

3 кесте – Ақпараттық жүйенің талапкерлердің дайындық деңгейі ақпараттары орналасқан

Индекс	Өріс атауы	Деректер типі
*	YR_ID	int AUTOINCREMENT
*	LEVEL	varchar(30)
*	LEVEL_DESCRIPTION	varchar(255)
Индекстер		
	pk_level	ON YR_ID

Level кестесінде ақпараттық жүйенің әр талапкердің дайындық деңгейі мәліметтері сақталған. Кесте «YR_ID, LEVEL, LEVEL_DESCRIPTION» деген өрістерінен тұрады. Level кестесінде «pk_level» деген индекс құрылған.

4 кесте – Ақпараттық жүйенің әр талапкердің жоғарғы оқу орнына түскен жылы ақпараттары орналасқан

Индекс	Өріс атауы	Деректер типі
*	SYID	int AUTOINCREMENT
*	AY	varchar(30)
*	YR	varchar(30)
*	SECTION	varchar(10)
*	IDNO	int
*	CATEGORY	varchar(30) DEFAULT 'INROLLED'
*	STATUS	varchar(30) DEFAULT 'New'
Индекстер		
	pk_schoolyr	ON SYID

Schoolyr кестесінде ақпараттық жүйенің әр талапкердің жоғарғы оқу орнына түскен жылы мәліметтері сақталған.

Кесте «SYID, AY, YR, SECTION, IDNO, CATEGORY, STATUS» деген өрістерінен тұрады. Schoolyr кестесінде «pk_schoolyr» деген индекс құрылған.

5 кесте – Ақпараттық жүйенің әр оқылатын пәндердің ақпараттары орналасқан

Индекс	Өріс атауы	Деректер типі
*	STUDSUBJ_ID	int AUTOINCREMENT
*	IDNO	int
*	SUBJ_ID	int
*	LEVEL	int
*	SEMESTER	varchar(30)
*	SY	mediumtext
Индекстер		
	pk_studentsubjects	ON STUDSUBJ_ID
Сыртқы кілттер		
	fk_studentsubjects_subject	(SUBJ_ID) ref subject (SUBJ_ID)

Studentsubjects кестесінде ақпараттық жүйенің әр талапкердің оқылатын пәндердің мәліметтері сақталған. Кесте «STUDSUBJ_Id, IDNO, SUBJ_ID, LEVEL, SEMESTER, SY» деген өрістерінен тұрады. Studentsubjects кестесінде «pk_studentsubjects» деген индекс құрылған. Studentsubjects кестесі fk_studentsubjects_subject деген сыртқы кілтімен байланысқан.

6 кесте – Ақпараттық жүйенің пәндердің жалпы мәліметтері ақпараттары орналасқан

Индекс	Өріс атауы	Деректер типі
*	SUBJ_ID	int AUTOINCREMENT
*	SUBJ_CODE	varchar(30)
*	SUBJ_DESCRIPTION	varchar(255)
*	UNIT	int
*	YR	varchar(30)
*	AY	varchar(30)
Индекстер		
	pk_subject	ON SUBJ_ID

Subject кестесінде ақпараттық жүйенің пәндердің жалпы мәліметтері сақталған. Кесте «SUBJ_ID, SUBJ_CODE, SUBJ_DESCRIPTION, UNIT, YR, AY» деген өрістерінен тұрады. Subject кестесінде «pk_subject» деген индекс құрылған.

7 кесте – Ақпараттық жүйенің әр талапкердің автоматты түрде құрылатын кездейсоқ идентификатор ақпараттары орналасқан

Индекс	Өріс атауы	Деректер типі
*	AutoID	int AUTOINCREMENT
*	Autonumber	varchar(90)
*	Incrementvalue	int
*	EndValue	int
*	Remarks	varchar(90)
Индекстер		
	pk_tblautonumber	ON AutoID

Tblautonumber кестесінде ақпараттық жүйенің әр талапкердің автоматты түрде құрылатын кездейсоқ идентификатор мәліметтері сақталған. Кесте «AutoID, Autonumber, Incrementvalue, EndValue, Remarks» деген өрістерінен тұрады. Tblautonumber кестесінде «pk_tblautonumber» деген индекс құрылған.

8 кесте – Ақпараттық жүйенің талапкердің оқылатын бөлімдер ақпараттары орналасқан

Индекс	Өріс атауы	Деректер типі
*	SECTIONID	int AUTOINCREMENT
*	SECTION	
Индекстер		
	pk_tblsection	ON SECTIONID

Tblsection кестесінде талапкердің оқылатын бөлімдер мәліметтері сақталған. Кесте «SECTIONID, SECTION» деген өрістерінен тұрады. Tblsection кестесінде «pk_tblsection» деген индекс құрылған.

9 кесте – Ақпараттық жүйенің әр талапкердің жалпы ақпараттары орналасқан

Индекс	Өріс атауы	Деректер типі
*	S_ID	int AUTOINCREMENT
*1	IDNO	int
*	FNAME	varchar(40)
*	LNAME	varchar(40)
*	MNAME	varchar(40)
*	SEX	varchar(10) DEFAULT 'Male'
*	BDAY	date
*	BPLACE	mediumtext
*	AGE	int
*	HOME_ADD	mediumtext
*	HEIGTH	int
*	WEIGTH	int
Индекстер		
	pk_tblstudent	ON S_ID
1	IDNO	ON IDNO

Tblstudent кестесінде ақпараттық жүйенің әр талапкердің жалпы мәліметтері сақталған. Кесте «S_ID, IDNO, FNAME, LNAME, MNAME, SEX, BDAY, BPLACE, AGE, HOME_ADD, GUARDIAN, GUARDIAN_ADD, GUARDIAN_TEL, LASTSCHOOLATTEND, HEIGTH, WEIGTH» деген өрістерінен тұрады. Tblstudent кестесінде «pk_tblstudent, IDNO» деген индекс құрылған.

10 кесте – Ақпараттық жүйенің пәндердің жалпы мәліметтері ақпараттары орналасқан

Индекс	Өріс атауы	Деректер типі
*	UPLEVEL_ID	int AUTOINCREMENT
*	IDNO	int
*	LEVEL	varchar(30)
*	SECTION	varchar(10)
*	AY	varchar(30)
Индекстер		
	pk_tblstudupdatelevel	ON UPLEVEL_ID

Tblstudupdatelevel кестесінде ақпараттық жүйенің әр талапкердің оқу дейгейі мәліметтері сақталған. Кесте «UPLEVEL_ID, IDNO, LEVEL, SECTION, AY » деген өрістерінен тұрады.

11 кесте – Ақпараттық жүйенің пайдалунышының ақпараттары орналасқан

Индекс	Өріс атауы	Деректер типі
*	ACCOUNT_ID	int AUTOINCREMENT
*	ACCOUNT_NAME	varchar(255)
*	ACCOUNT_USERNAME	varchar(255)
*	ACCOUNT_PASSWORD	mediumtext
*	ACCOUNT_TYPE	varchar(30)
Индекстер		
	pk_useraccounts	ON ACCOUNT_ID

Useraccounts кестесінде ақпараттық жүйе пайдалунышының мәліметтері сақталған. Кесте «ACCOUNT_ID, ACCOUNT_NAME, ACCOUNT_USERNAME, ACCOUNT_PASSWORD, ACCOUNT_TYPE» деген өрістерінен тұрады. Useraccounts кестесінде «pk_useraccounts» деген индекс құрылған.

3 Программалық қамтаманы құру

3.1 Программалау тілін таңдауды негіздеу

C# - объектілі-бағдарланған және контекстік-бағдарланған ұғымдарды біріктіретін бағдарламалау тілі. 1998-2001 жж. Microsoft корпорациясының Anders Heilсберг басшылығымен инженерлер тобы Microsoft корпорациясының Microsoft .NET платформасына арналған қосымшаларды әзірлеудің негізгі тілі ретінде жасалды. C # құрастырушысы .NET-дің стандартты орнатылымына қосылған, сондықтан оған арналған бағдарламалар Visual Studio сияқты құралдарсыз құрастырылып, құрастырылуы мүмкін.

C# тілі C тіліндегі синтаксисі бар C тіліндегі синтаксисі бар, оның синтаксисі C++ және Java-ге ең жақын. Тілде қатаң статикалық теру, полиморфизмді, оператор жүктемесін қолдайды, сыныптың мүшелік функцияларына, атрибуттарға, оқиғаларға, қасиеттерге, ерекшеліктерге, XML пішіміндегі түсініктемелерге нұсқайды. C++, Delphi, Modula және Smalltalk - C # бағдарламаларын қолдану тәжірибесіне сүйене отырып, бағдарламалық қамтамасыз етуді әзірлеу кезінде қиындықтар туындаған кейбір модельдерді жоққа шығарады: мысалы, C# бірнеше сыныпты иеленуді қолдамайды немесе Haskell-қа қарағанда басқаша түрін енгізіңіз.

3.2 Функционалдық тағайындалуы

«Талапкер» автоматтандырылған ақпараттық жүйесі студенттердің қозғалысы мен қол жетімділігі туралы ақпаратты қорытындылауға арналған. Бағдарламаны пайдаланушылар - кадрлар бөлімінің қызметкерлері. Студенттер оқуға қабылдау туралы өтініштер негізінде қабылданады, онда оқу шарттары көрсетілген. Өтініштер түскен кезде алғашқы құжаттардың деректері өтінішті қабылдау журналында атауы, туған жылы, жынысы, үйінің мекен-жайы, ұлты, таңдаған мамандығы, қаржыландыру түрі, аймақ, бөлім, елді мекен және сертификаттың орташа белгісі көрсетілген жинақталады. Кадр бөлімі аптасына кемінде бір рет студенттердің жазбаларын тексереді.

3.3 Шақыру және жүктеу

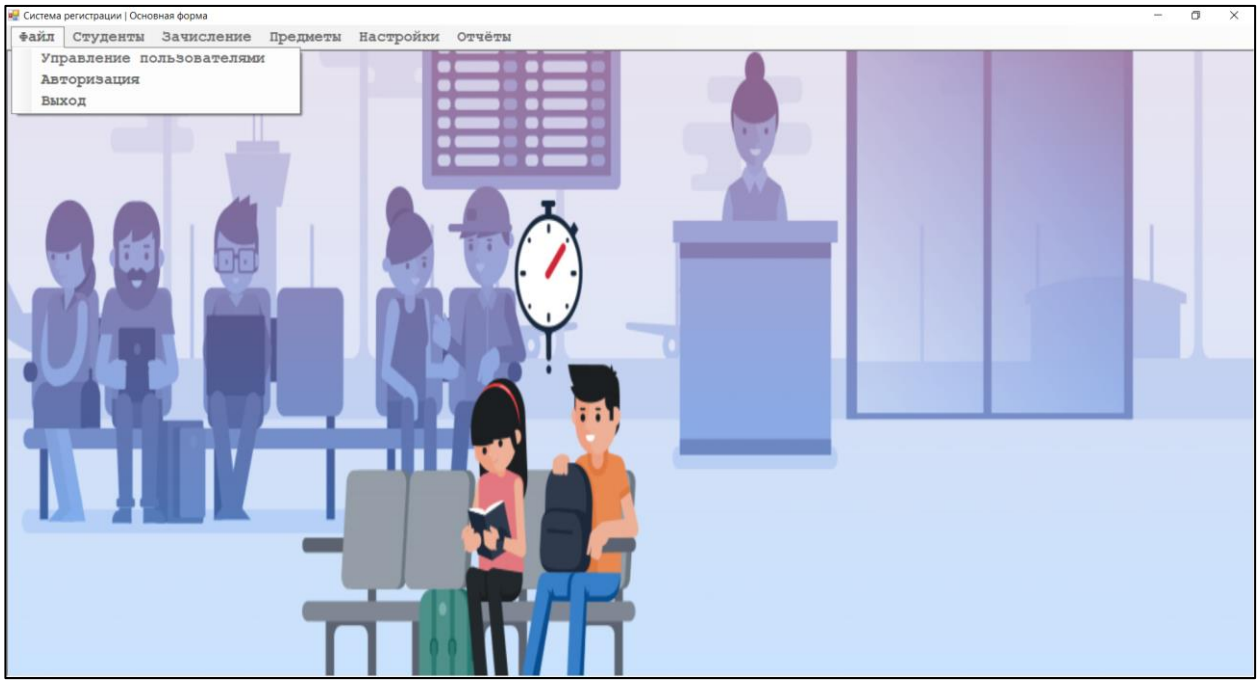
Талапкер тіркеу ақпараттық жүйесін Microsoft visual srudio көмегімен ашу үшін Пуск → Программы → Microsoft visual studio 2019 жолын ашамыз. Содан кейін өз жобамызды ашып, «пуск» батырмасын басамыз.

3.4 Кіріс және шығыс мәліметтер

Бағдарламаның шығыс мәліметтері арқылы мәліметтерді беру, нәтиже ретінде алуға болады.

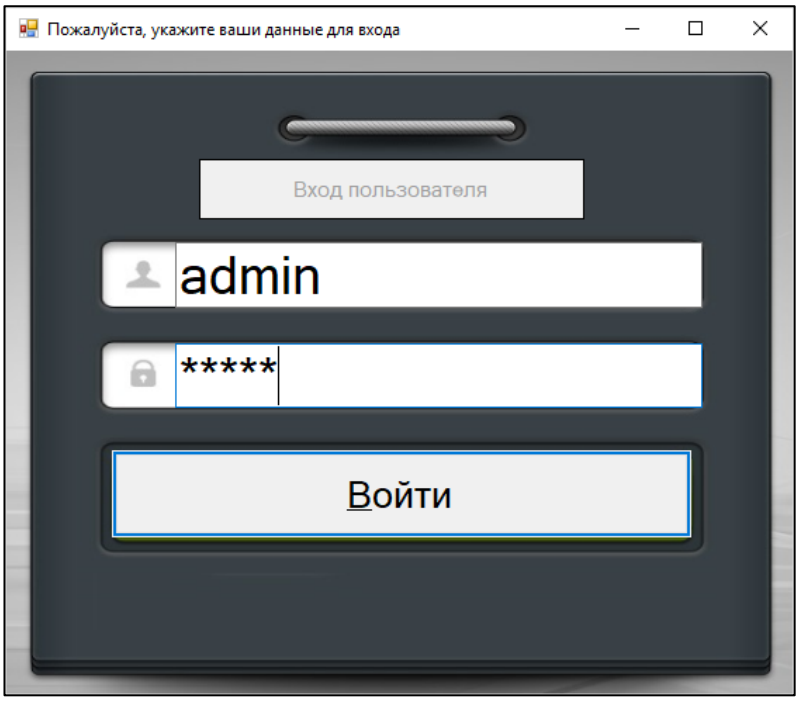
3.5 Бакылау мысалын сипаттау

Талапкер тіркеу ақпараттық жүйесіне бағдарламасымен жұмыс бастау үшін ең алдымен Microsoft visual studio бағдарламалау ортасы бағдарламаны іске қосамыз. Содан кейін 3.1 суретте көрсетілгендей бағдарламаның негізгі терезесі пайда болады.



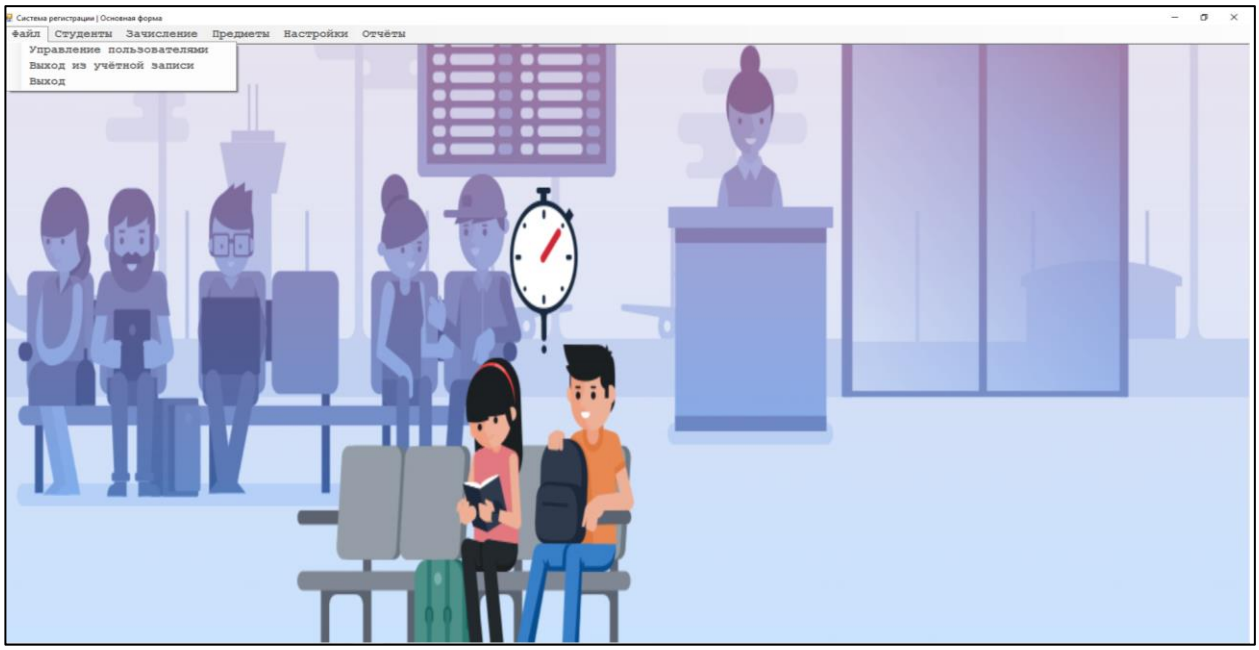
3.1 сурет- Бағдарламалық негізгі бөлімі

Талапкер тіркеу ақпараттық жүйесі іске қосылғаннан кейін «авторизация» бөлімі арқылы ақпараттық жүйеге «администратор» болып кіреміз. 3.2 – суретте көрсетілгендей «логин» мен «құпия сөзді» енгізіп «войти» батырмасын басамыз.



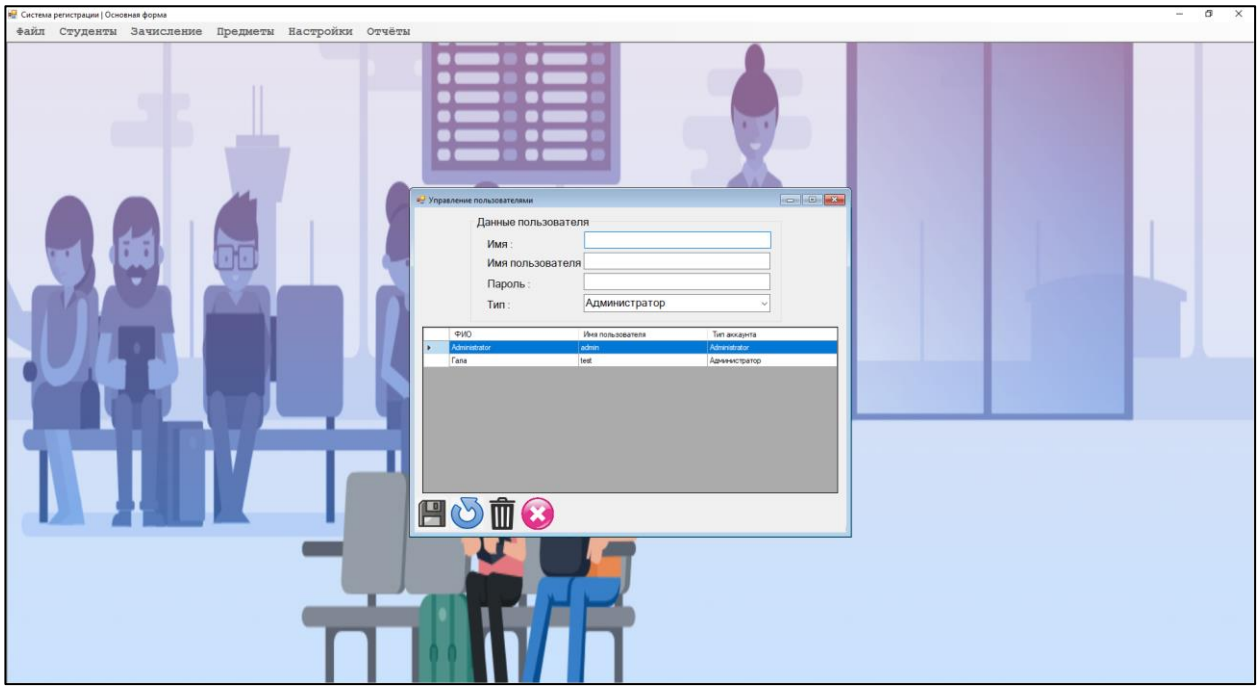
3.2 сурет-Бағдарламаның «авторизация» бөлімі

Талапкер тіркеу ақпараттық жүйесіне «администратор» болып кіргеннен кейін, көптеген ақпараттық функциялары пайда болып, оларды қолдануға болады.



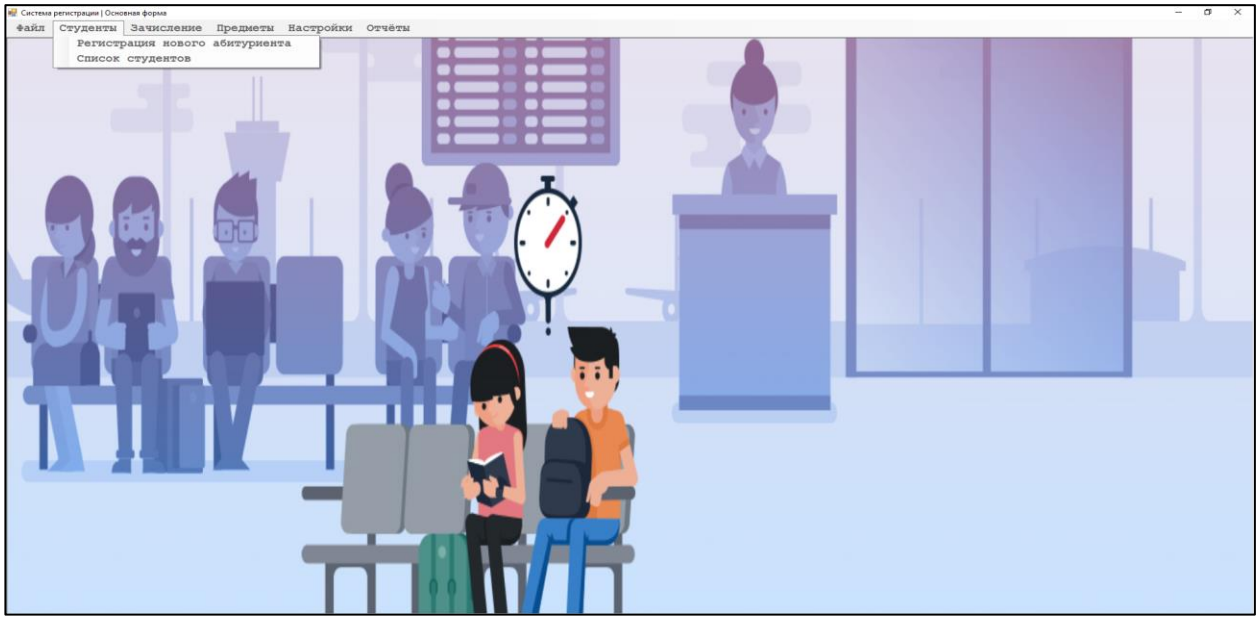
3.3 сурет- Бағдарламаның «админ» болып кіру бөлімі

Талапкер тіркеу ақпараттық жүйесінің «пайдаланушылар басқару» бөлімінде келесі 3.4 суретте көрсетілгендей жалпы пайдаланушылардың аты, құпия сөзін, пайдалунышы түрін өзгертуге болады.



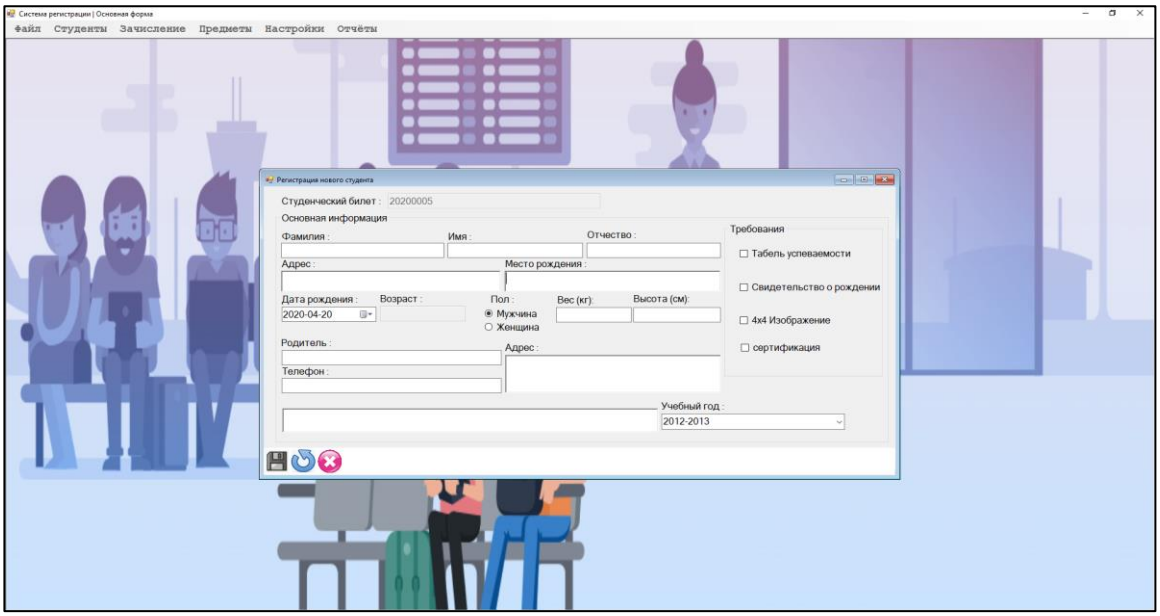
3.4 сурет- Бағдарламаның «управление пользователями» бөлімі

Талапкер тіркеу ақпараттық жүйесінің «студенты» мәзірінде келесі 3.5 – суретте көрсетілгендей, жаңа талапкерді енгізу және талапкер тізімін көруге болатын бөлімдері орналасқан.



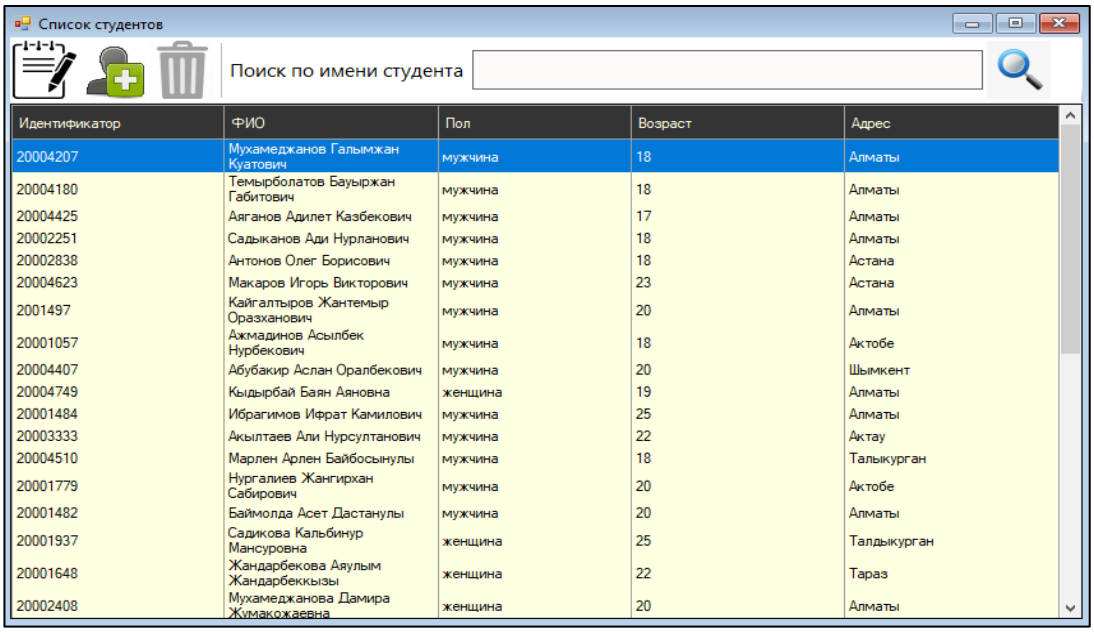
3.5 сурет- Бағдарламаның «студенты» бөлімі

Жаңа таларкерді қосу үшін ең алдымен «студенты» мәзірін басып «регистрация нового абитуриента» бөлімін басамыз. Содан кейін 3.6 – суретте көрсетілгендей жаңа талапкер бөлімі пайда болады.



3.6 сурет- Бағдарламаның «регистрация нового студента» бөлімі

Жаңа талапкер қосу бөлімінде талапкердің жалпы ақпараты сақталады: Талапкердің аты жөні, туған жылы, жасы, байланыс нөмірі, оқу жылы, құжаттары. Жоғарғы оқу орнына түскен студенттерді көру үшін «список студентов» батырмасыз басамыз. Содан кейін 3.7 – суретте көрсетілгендей студенттер тізімі көрсетіледі.

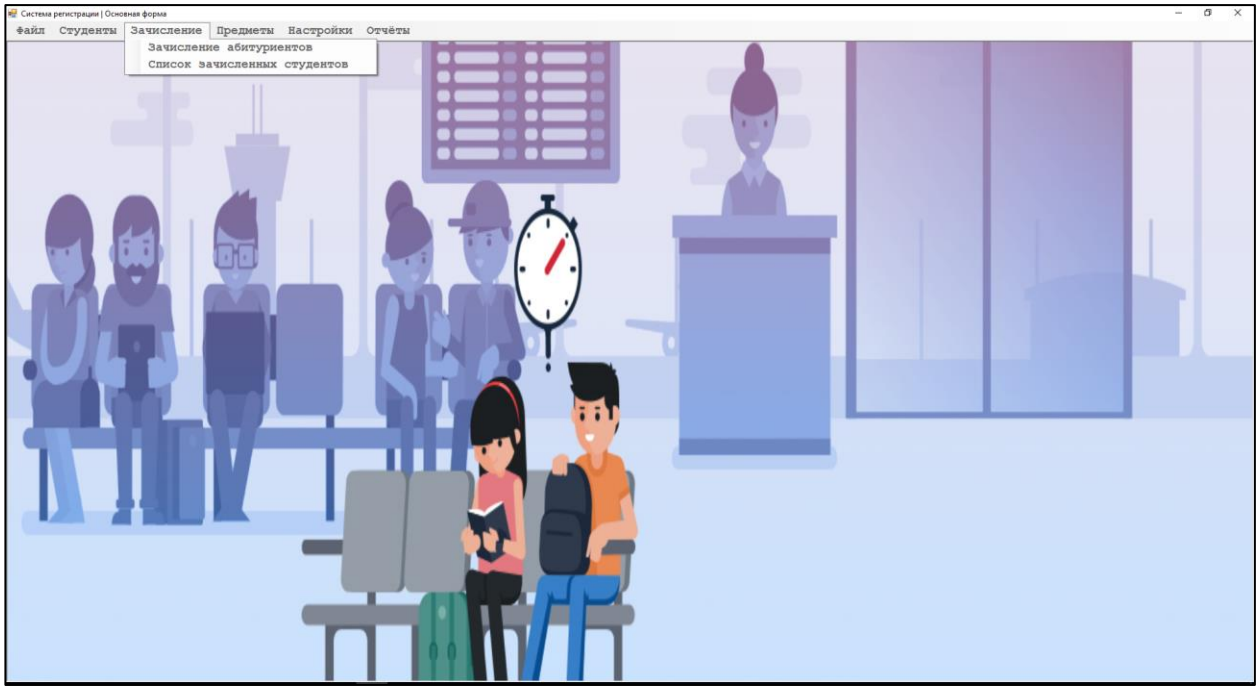


Идентификатор	ФИО	Пол	Возраст	Адрес
20004207	Мухамеджанов Галымжан Куатович	мужчина	18	Алматы
20004180	Темырболатов Бауыржан Габитович	мужчина	18	Алматы
20004425	Аяганов Адилет Казбекович	мужчина	17	Алматы
20002251	Садыканов Ади Нурланович	мужчина	18	Алматы
20002838	Антонов Олег Борисович	мужчина	18	Астана
20004623	Макаров Игорь Викторович	мужчина	23	Астана
2001497	Кайгалтыров Жантемыр Оразханович	мужчина	20	Алматы
20001057	Ажмадинов Асылбек Нурбекович	мужчина	18	Актобе
20004407	Абубакир Аслан Оралбекович	мужчина	20	Шымкент
20004749	Кыдырбай Баян Аяновна	женщина	19	Алматы
20001484	Ибрагимов Ифрат Камилович	мужчина	25	Алматы
20003333	Акылтаев Али Нурсултанович	мужчина	22	Актау
20004510	Марлен Арлен Байбосынулы	мужчина	18	Талыкурган
20001779	Нургалиев Жангирхан Сабирович	мужчина	20	Актобе
20001482	Баймолда Асет Дастанулы	мужчина	20	Алматы
20001937	Садикова Кальбинур Мансуровна	женщина	25	Талдыкурган
20001648	Жандарбекова Аяулым Жандарбеккызы	женщина	22	Тараз
20002408	Мухамеджанова Дамира Жумакожаевна	женщина	20	Алматы

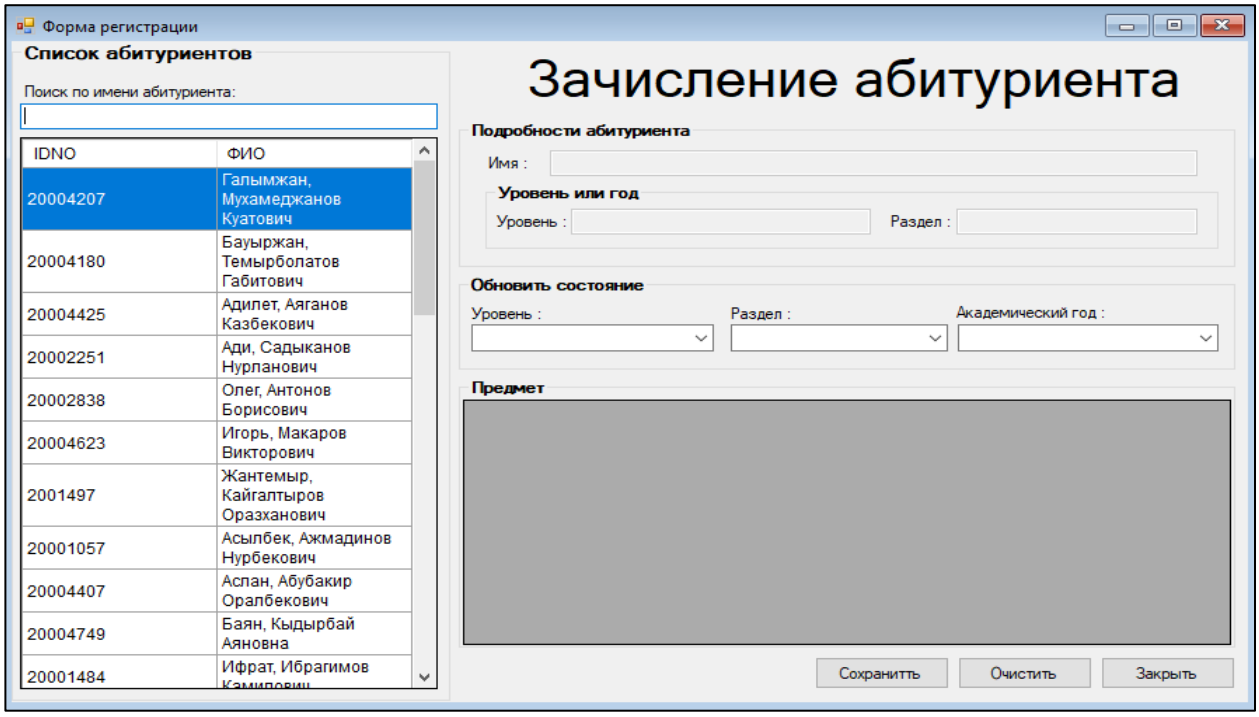
3.7 сурет- Бағдарламаның «список студентов» бөлімі

Студенттер тізімі бөлімінде студентті қосу, өзгерту және жою функциялары бар. Егер студентті аты бойынша табу керек болса «поиск» бөліміне студент атын жазып «enter» батырмасын басамыз.

Талапкер тіркеу ақпараттық жүйеге талапкерді жоғарғы оқу орнына қабылдау үшін «зачисление абитуриент» мәзірін басамыз.



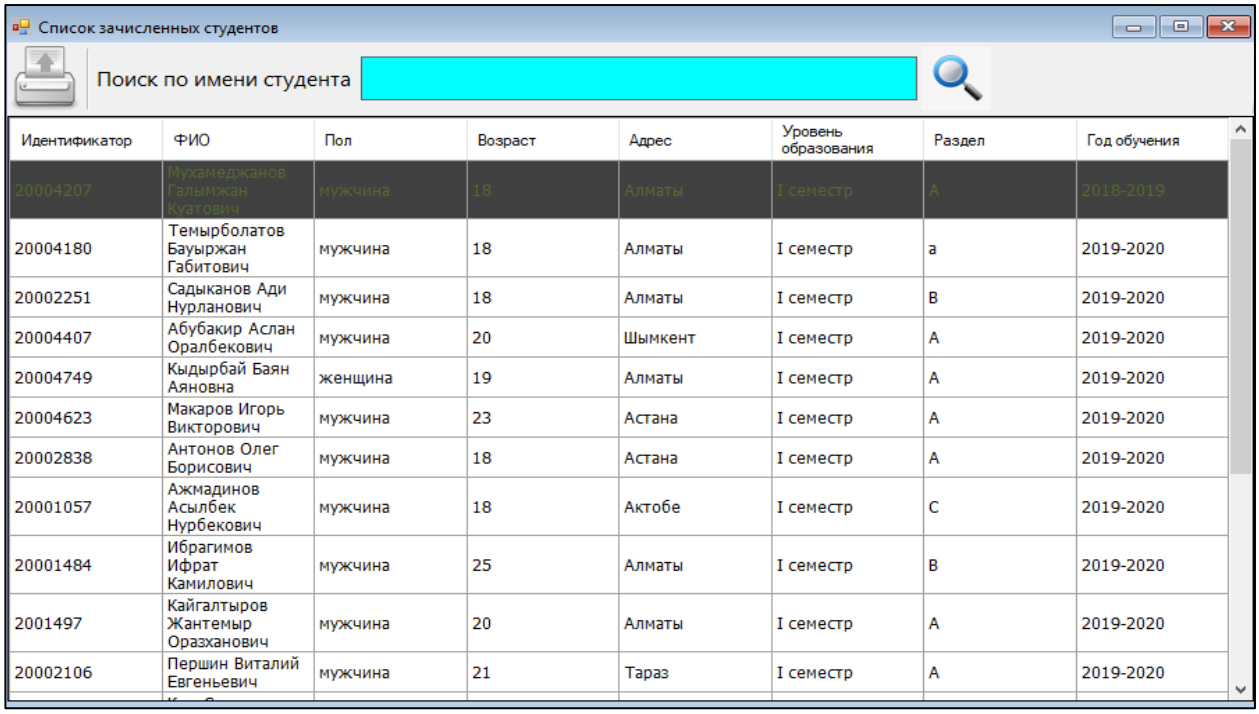
3.8 сурет- Бағдарламаның «зачисление» бөлімі



3.9 сурет-Бағдарламаның «зачисление абитуриента» бөлімі

Талапкер тіркеу ақпараттық жүйенің «зачисление абитуриента» бөлімінде талапкерді оқу жылы, оқылатын пән, тобын белгілеп деректер базасына енгізуге

болады. Студенттердің тізімін көру үшін «зачисление» мәзірін басып, «список зачисленных студентов» батырмасын басамыз. Содан кейін 3.9 суретте көрсетілгендей жаңа терезе пайда болады.



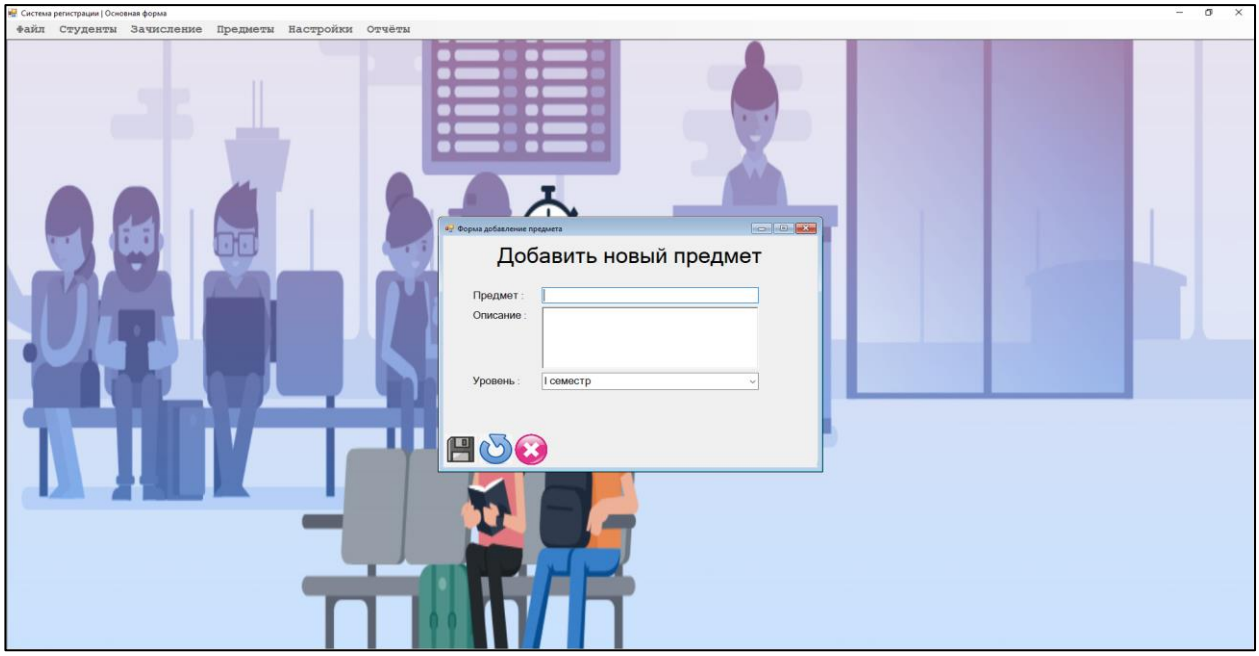
Идентификатор	ФИО	Пол	Возраст	Адрес	Уровень образования	Раздел	Год обучения
20004207	Мухамеджанов Галымжан Куатович	мужчина	18	Алматы	I семестр	А	2018-2019
20004180	Темырболатов Бауыржан Габитович	мужчина	18	Алматы	I семестр	а	2019-2020
20002251	Садыканов Ади Нурланович	мужчина	18	Алматы	I семестр	В	2019-2020
20004407	Абубакир Аслан Оралбекович	мужчина	20	Шымкент	I семестр	А	2019-2020
20004749	Кыдырбай Баян Аяновна	женщина	19	Алматы	I семестр	А	2019-2020
20004623	Макаров Игорь Викторович	мужчина	23	Астана	I семестр	А	2019-2020
20002838	Антонов Олег Борисович	мужчина	18	Астана	I семестр	А	2019-2020
20001057	Ажмадинов Асылбек Нурбекович	мужчина	18	Актобе	I семестр	С	2019-2020
20001484	Ибрагимов Ифрат Камилович	мужчина	25	Алматы	I семестр	В	2019-2020
2001497	Кайгалтыров Жантемыр Оразханович	мужчина	20	Алматы	I семестр	А	2019-2020
20002106	Першин Виталий Евгеньевич	мужчина	21	Тараз	I семестр	А	2019-2020

3.10 сурет- Бағдарламаның «список зачисленных студентов» бөлімі

Студенттер тізімі бөлімінде, студентті «аты» бойынша іздеуге болады. Тізімді принтер арқылы басып шығару мүмкіндігі орналасқан.

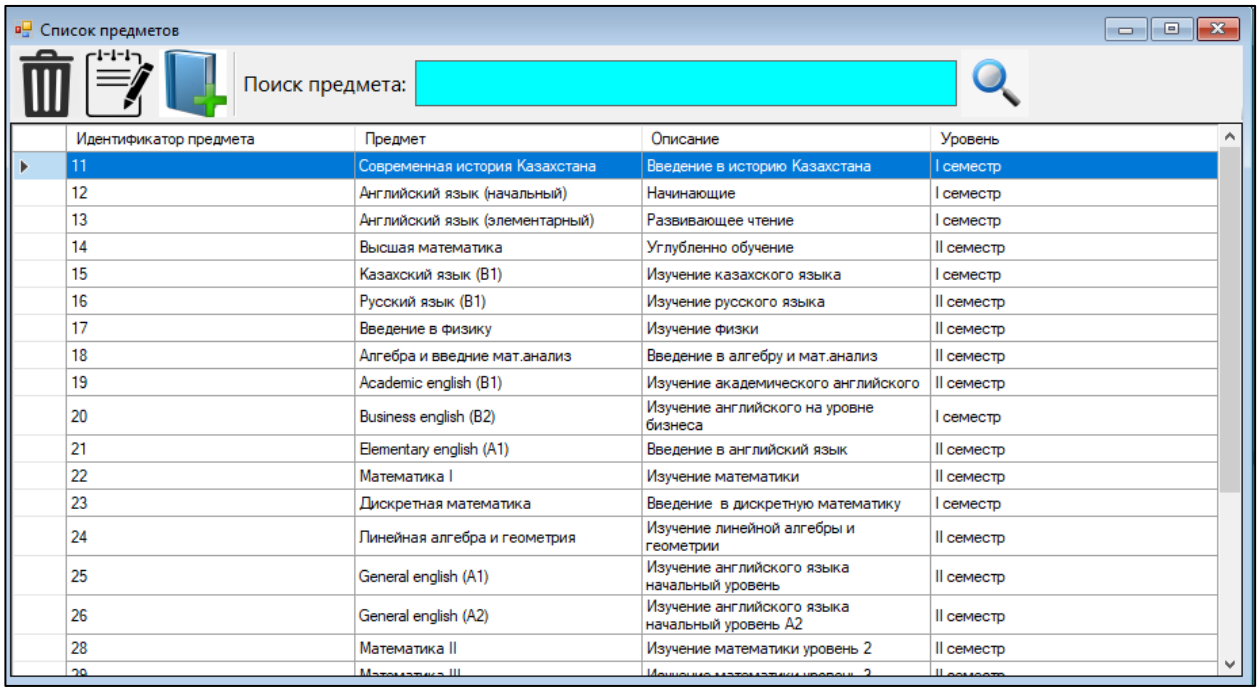
3.11 сурет- Бағдарламаның «предметы» бөлімі

Ақпараттық жүйеге жаңа пәнді енгізу үшін «предметы» мәзірін басып, онын ішіндегі «зарегистрировать новый предмет» батырмасын басу қажет.



3.12 сурет- Бағдарламаның «добавить новый предмет» бөлімі

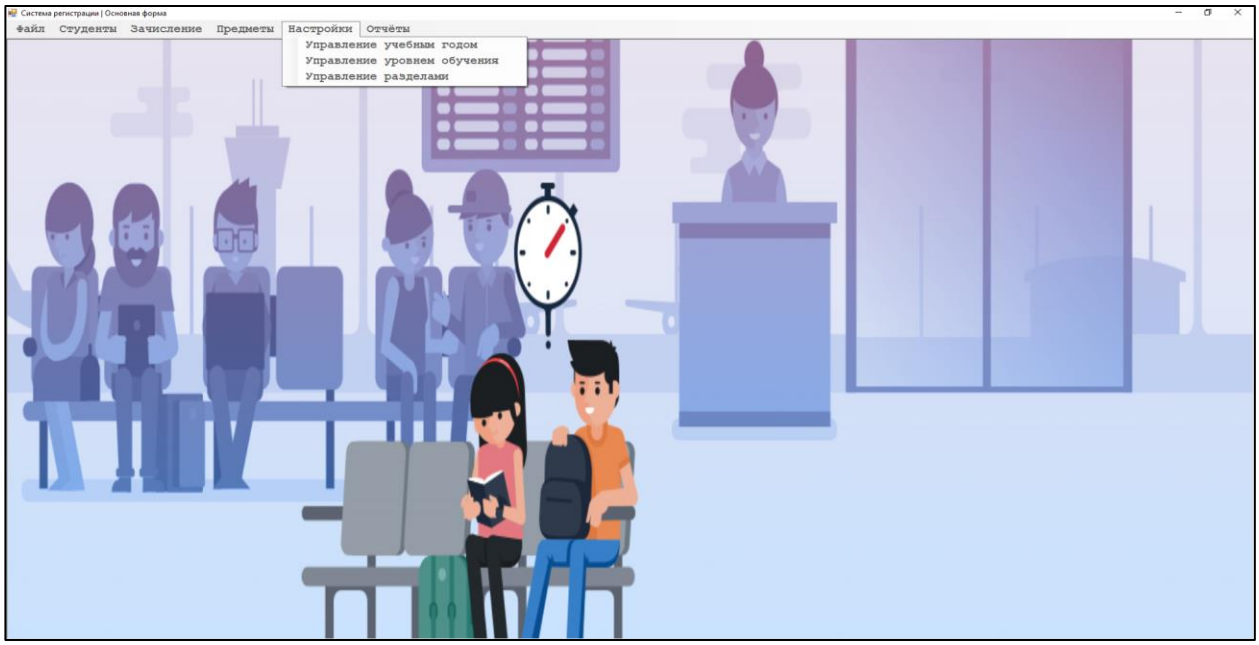
Жаңа пәнді енгізген кезде міндетті түрді пәннің атын, сипаттамасын және оқу деңгейін толтыру керек. Пәннің тізімі көру үшін «предметы» мәзірін басып, «список предметов» батырмасын басу қажет. Содан кейін 3.13 – суретте көрсетілгендей пәндердің тізімі шығады.



Идентификатор предмета	Предмет	Описание	Уровень
11	Современная история Казахстана	Введение в историю Казахстана	I семестр
12	Английский язык (начальный)	Начинающие	I семестр
13	Английский язык (элементарный)	Развивающее чтение	I семестр
14	Высшая математика	Углубленно обучение	II семестр
15	Казахский язык (B1)	Изучение казахского языка	I семестр
16	Русский язык (B1)	Изучение русского языка	II семестр
17	Введение в физику	Изучение физки	II семестр
18	Алгебра и введние мат.анализ	Введение в алгебру и мат.анализ	II семестр
19	Academic english (B1)	Изучение академического английского	II семестр
20	Business english (B2)	Изучение английского на уровне бизнеса	I семестр
21	Elementary english (A1)	Введение в английский язык	II семестр
22	Математика I	Изучение математики	II семестр
23	Дискретная математика	Введение в дискретную математику	I семестр
24	Линейная алгебра и геометрия	Изучение линейной алгебры и геометрии	II семестр
25	General english (A1)	Изучение английского языка начальный уровень	II семестр
26	General english (A2)	Изучение английского языка начальный уровень A2	II семестр
28	Математика II	Изучение математики уровень 2	II семестр

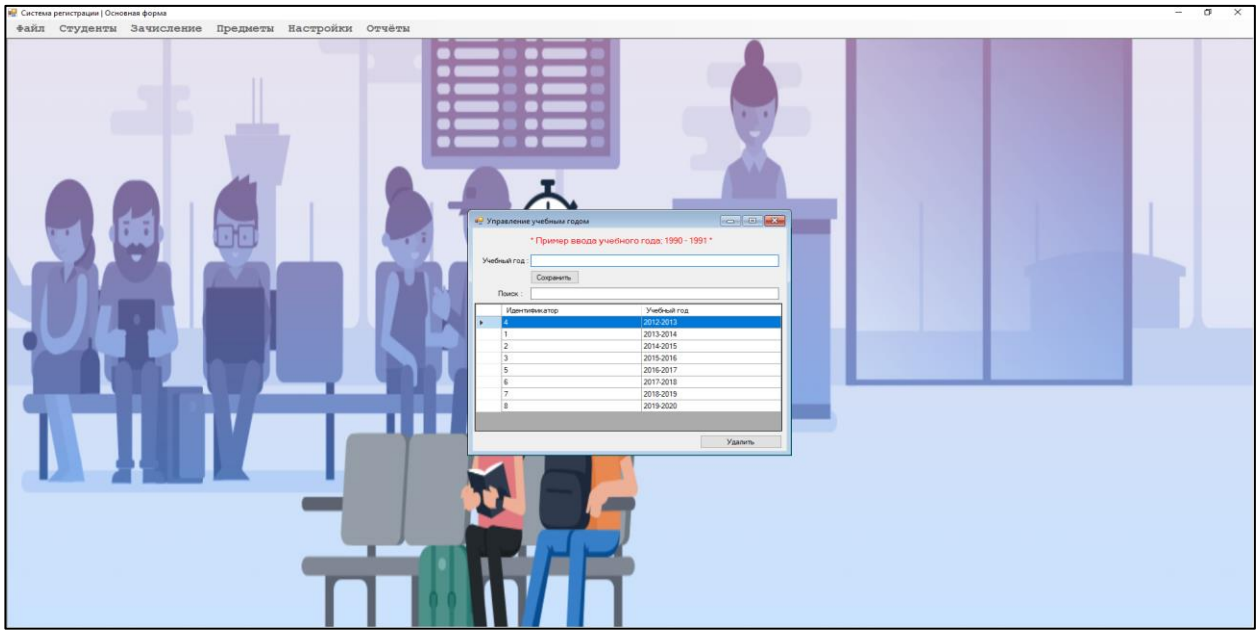
3.13 сурет-Бағдарламаның «список предметов» бөлімі

Пән тізімі бөлімінде пәнді жаңадан қосуға, жоюға, өңдеу функциялары бар. Пәнді аты бойынша іздеуге болады. Ақпараттық жүйенің «настройки» мәзірінде «управление учебным годом», «управление уровнем обучения» және «управление разделами» бөлімдері орналасқан.



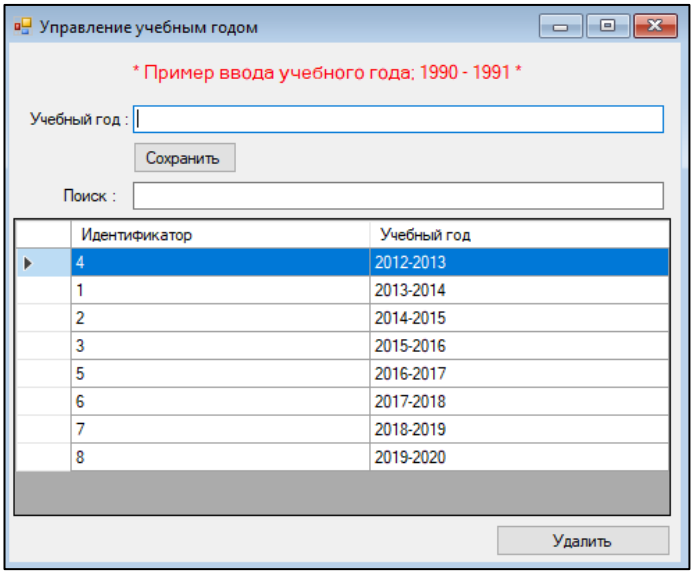
3.14 сурет- Бағдарламаның «настройки» бөлімі

Жаңадан оқу жылын қосу үшін «настройки» мәзірі ішіндегі «управление учебным годом» батырмасын басу қажет. Содан кейін 3.15 – суретте көрсетілгендей жаңа терезе пайда болады.

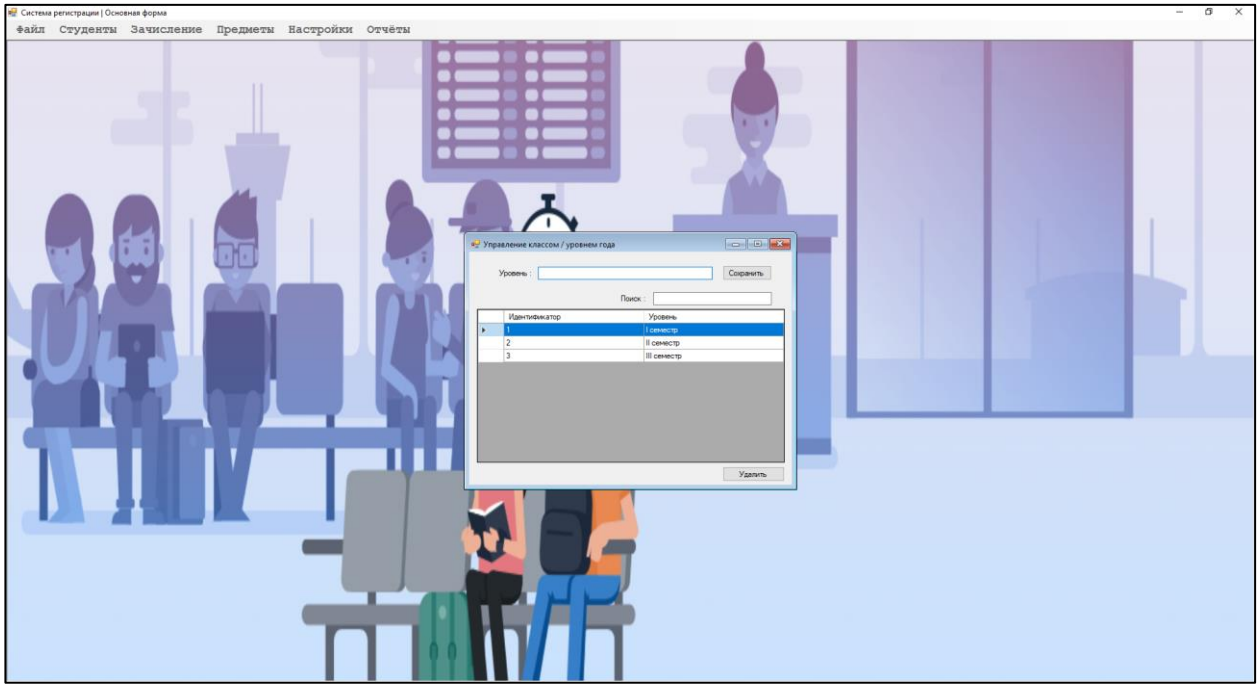


3.15 сурет-Бағдарламаның «управление учебным годом» бөлімі

«Управление учебным годом» бөлімінде жаңадан оқу жылын қосуға, оқу жылын іздеуге немесе оқу жылын жою мүмкіндіктері бар.

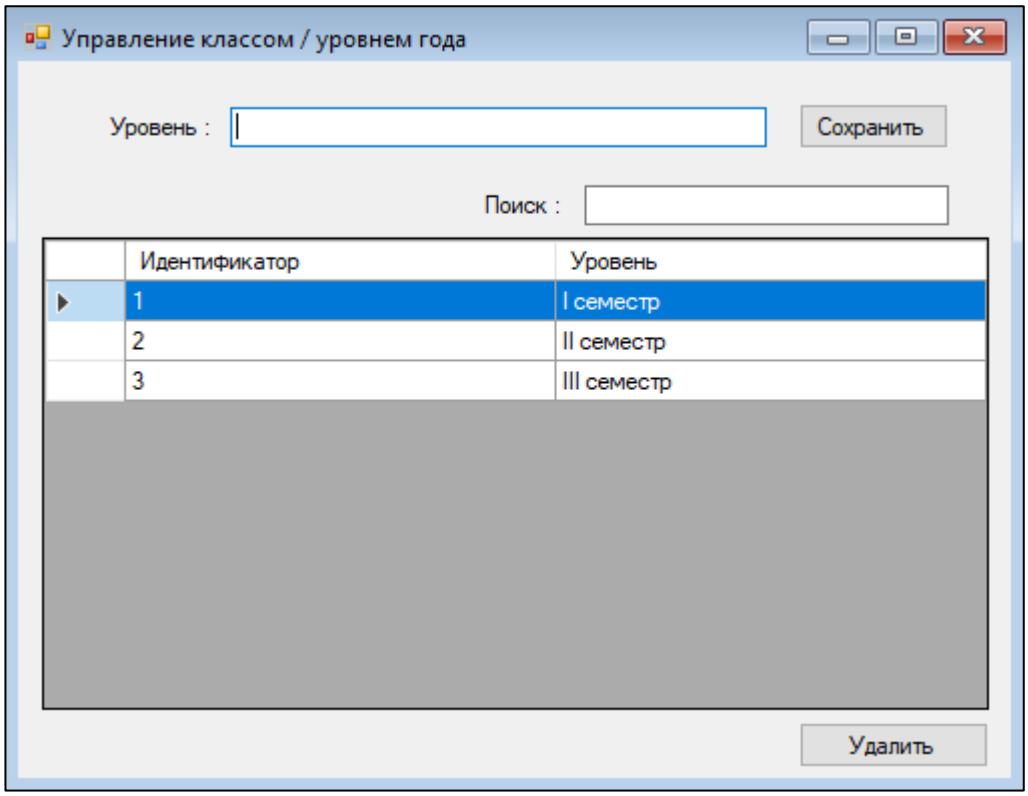


3.16 сурет- Бағдарламаның «управление учебным годом» бөлімі

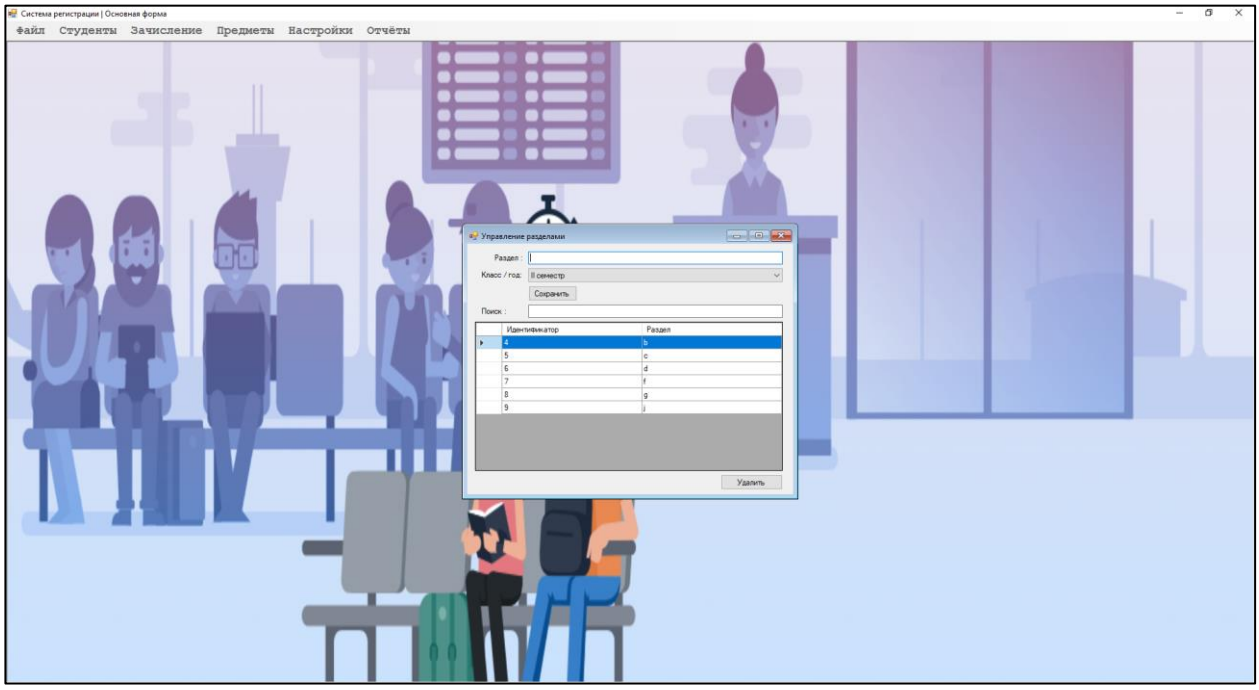


3.17 сурет- Бағдарламаның «управление классом» бөлімі

Талапкер тіркеу ақпараттық жүйесіне жаңадан оқу деңгейін қосу үшін «настройки» мәзіріне өтіп, онын ішіндегі «управление уровнем обучения» батырмасын басу қажет. Содан кейін 3.18 – суретте көрсетілгендей жаңа терезе ашылады.

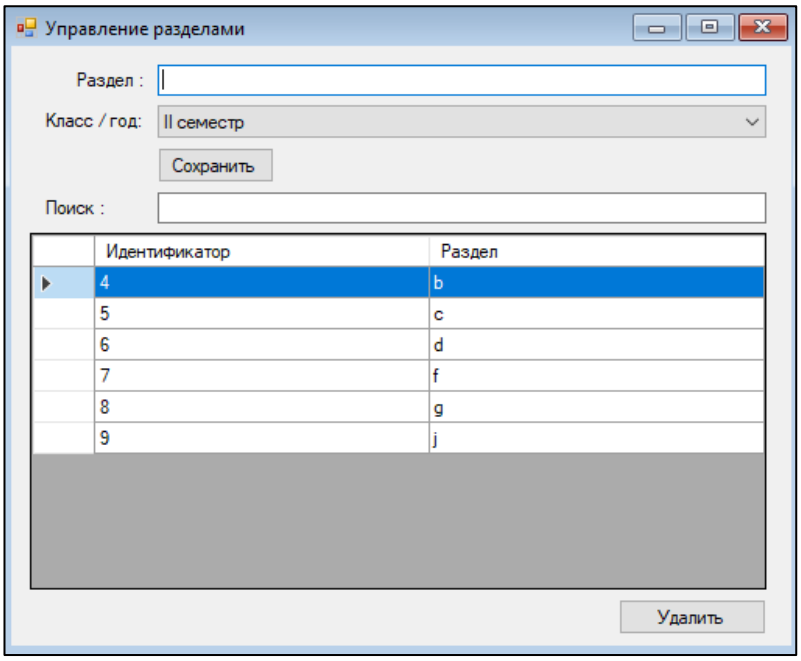


3.18 сурет- Бағдарламаның «управление классом уровнем года» бөлімі



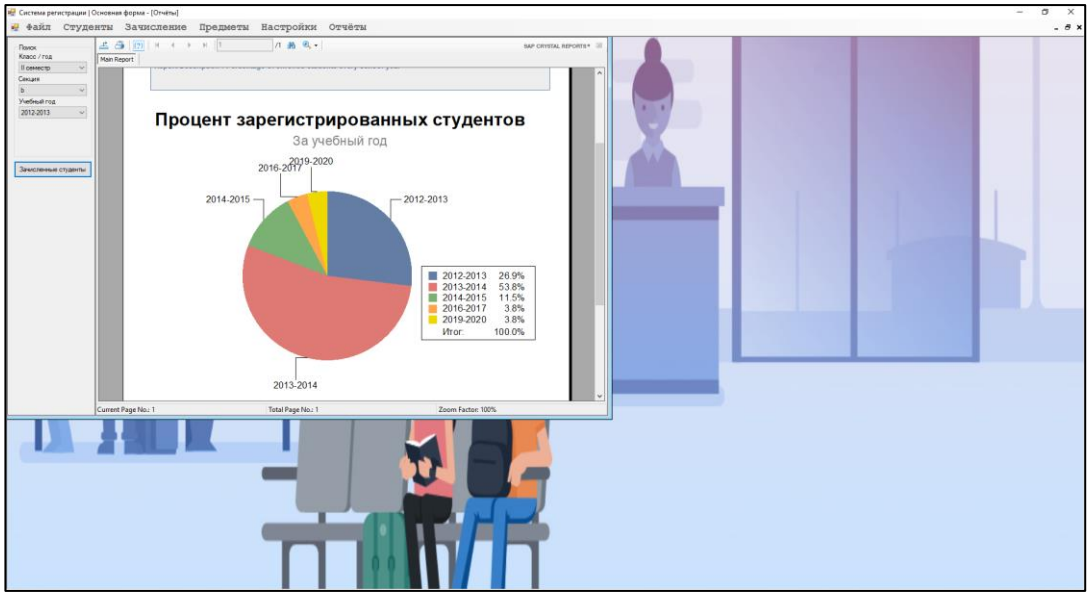
3.19 сурет- Бағдарламаның «управление разделами» бөлімі

Талапкер тіркеу ақпараттық жүйесіне жаңа топты қосу үшін «настройки» мәзіріне өтіп, «управление разделами» батырмасын басу қажет. Содан кейін 3.20 – суретте көрсетілгендей жаңа терезе ашылады.



3.20 сурет- Бағдарламаның «управление разделами» бөлімі

Бағдарламаның «управление разделами» бөлімінде әр топ бойынша «семестрді» қосуға болады. Әр топтың идентификаторы бойынша іздеу және жою мүмкіндіктері бар.



3.21 сурет- Бағдарламаның «отчет» бөлімі

Бағдарламаның «отчет» бөлімінде қанша талапкер жоғарғы оқу орнына тіркелгенін, әр жыл бойынша пайызбен шығарып беретін диаграммасы бар. Есеп шығару кезінде топ, секция және жылы таңдалады. Таңдалған шарттар бойынша диаграмма көрсетіледі.

ҚОРЫТЫНДЫ

Бұл дипломдық жобаның тақырыбы - «студенттің ақпараттық жүйесін жобалау». Бұл тақырыпты таңдау тек өзектілігімен ғана емес, сонымен қатар зерттелетін мәселеге жеке қызығушылығымен де анықталды.

Қазіргі заманғы дерекқор технологиялары кез-келген салада сәттіліктің анықтаушы факторларының бірі болып табылады, ақпаратты сақтауды, деректерді әлеуетті пайдаланушыларға ұсынуды және көптеген басқа процестерге қолдау көрсетуді қамтамасыз етеді. Сонымен қатар, мәліметтер базасы әртүрлі ғылыми жобалардың негізін құрайды, олар ақпарат жинауға мүмкіндік береді.

Деректер базасының қуаты соңғы бірнеше онжылдықта алынған зерттеулер мен технологиялық әзірлемелердің нәтижелеріне негізделген және мамандандырылған бағдарламалық өнімдерде қоршалған, оларды әдетте дерекқорды басқару жүйесі (ДББЖ) немесе жай дерекқор жүйелері деп атайды. ДҚБЖ - ақпараттың үлкен бөлігін жинаудың және оларды тиімді басқарудың тиімді құралы, бұл ұзақ уақыт бойы деректерді қауіпсіз және сенімді сақтауға мүмкіндік береді. ДҚБЖ қазіргі уақытта нарықта қол жетімді ең күрделі бағдарламалық өнімдер санатына жатады. Мәліметтер базасы - деректер жиынтығын ДҚБЖ бақылауымен белгілеу әдеттегі болып табылады.

Қабылдау комиссиясы және күндізгі және сырттай бөлімдер - орта кәсіптік білім беру мекемесінің негізгі оқу және ғылыми бөлімшелерінің бірі, колледжге қабылдау ережелерін әзірлейді, жастарға мансаптық кеңес берумен және оларды қабылдау емтихандарына қатысумен айналысатын университеттің барлық бөлімдерінің қызметін үйлестіреді, құжаттарды қабылдауды ұйымдастырады, түсу емтихандарын өткізеді, оқушылардың үлгерімі мен сабаққа қатысуын бақылауды ұйымдастыру, оқу процесін ұйымдастыру және тағы басқалар жатады.

Дипломдық жобаның мақсаты кәсіби білім беру мекемесінде күндізгі және сырттай бөлімдер сияқты құрылымдық бөлімшелердің жұмысын автоматтандыруға, жеңілдетуге және тиімді етуге мүмкіндік беретін ақпараттық жүйені құру болды. Осындай ақпараттық жүйемен дипломдық жобаны құрудың тағы бір мақсаты осы жобаның көмегімен кафедралардың жұмысы қарапайым, тиімді және сапалы болатындығын дәлелдеу болды. Іске асыру үшін қабылдау комиссиясының жұмысы үшін әр түрлі ақпараты бар мәліметтер базасы жасалды.

Қазіргі уақытта көптеген білім беру мекемелері заманауи телекоммуникациялар мен компьютерлік жүйелерді енгізу бойынша белгілі бір жұмыстарды жүргізуде, бірақ көптеген білім беру ұйымдарында ақпараттандыру саласындағы белсенді жұмыс ақпараттық және телекоммуникациялық технологияларды тек оқу процесінде қолдануға бағытталған және басқару процестеріне әсер етпейді.

ҚЫСҚАРТУЛАР МЕН БЕЛГІЛЕУЛЕР

ДББЖ - Деректер базасын басқару жүйесі
ДБ - Деректер базасы
ТҚК - Техникалық қызмет көрсету
БЖ - Бағдарламалық жасақтама
ААҚ - автоматтандырылған ақпараттық жүйе

ПАЙДАЛАНҒАН ӘДЕБИЕТТЕР ТІЗІМІ

1 Глинський Я.М., Анохін В. Е., Ряжська В.А. "Паскаль. Turbo Pascal і Delphi", третье изд., - Львов: "Деол", 2015 р.

2 Сердюченко В.Я. Разработка алгоритмов и программирования на языке Turbo Pascal. - Х., 2015.

3 Джаррод Холингворт, Боб Сворт, Марк Кешмен, Поль Густавсон Borland C++ Builder 6. Руководство разработчика = Borland C + + Builder 6 Developer's Guide. - М .: «Вильямс», 2016. - С. 976. - ISBN 0-672-32480-6

4 Джерод Холлингворс, Дэн Баттерфилд, Боб СВОТ C + + Builder 5. Руководство разработчика = C + + Builder 5 Developer's Guide. - М .: «Диалектика», 2015. - С. 884. - ISBN 0-672-31972-1

5 Краснов М. Графика в проектах Borland C builder. - Санкт-Петербург: БХВ, 1998. - 356 с.

6 Проценко С. С., Чаленко П. И., СТАВРОВСКИЙ А. Б. Техника программирования на языке Си. - К: Лебедь, 2017 - 224 с.

7 Шелест С. Д. Программирование: материал технической информации. Санкт-Петербург: БХВ, 2016. - 592 с.

8 «Информатика»: Учебник под ред. проф. Н.В. Макаровой, М .: "Брабус», 2016 г.. -37 с.

9 «Информационные системы»: под ред. Панфилова и Турковского, М .: ЗАО «Издательство БИНОМ», 2014. -67 с.

10 В.В. Кириллов CitForum «Основы проектирования реляционных баз данных» »- М .: ЗАО« Издательство Тризор », 2017. -121 с.

11 Эрик Дж. Брауде. Технология разработки программного обеспечения, 2018. .Липаев, В. В. Программная инженерия. Методологические основы [Текст]: Учеб. / В.

12 Липаев; Гос. ун-т - Высшая школа экономики. - М.: ТЕИС, 2006. - 608 с. - 1000 экз. - ISBN 5-7598-0424-3 УДК 004,41 (075,8), ББК 32.973.26-018я73, Липаев, Высшая школа экономики, ТЕИС, 2019, PDF, экономика, программирование.

13 Лапланте, Фил (2017). Требования Техника для программного обеспечения и систем (изд. 1). Редмонд, штат Вашингтон: CRC Press. ISBN 1-42006-467-3.

14 Макконнелл Стив (2015). Быстрое развитие: Укрощение Дикого Software Расписания (. Вид 1). Редмонд, штат Вашингтон: Microsoft Press. ISBN 1-55615-900-5. .Виджерс, Карл Е. (2015). Требования к программному обеспечению (изд. 2). Редмонд, штат Вашингтон: Microsoft Press. ISBN 0-7356-1879-8.

15 Брайан Беренбач, Даниэль Паулиш, Юрген Катзмеер, Арнольд Рудофер (2016). Программное обеспечение и системы технических требований: на практике. Нью-Йорк: McGraw-Hill Профессиональный. ISBN 0-07-1605479.

А ҚОСЫМШАСЫ

Form1.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
namespace EnrollmentSystem
{
public partial class Form1 : Form
{
public Form1()
{
InitializeComponent();
}
public void closeForm()
{
foreach (Form frm in this.MdiChildren)
{
frm.Close();
}
}
public void enabled_menu()
{
ts_Enrollment.Enabled = true;
ts_Maintenance.Enabled = true;
ts_students.Enabled = true;
ts_Subject.Enabled = true;
ts_User.Enabled = true;
ts_Reports.Enabled = true;
ts_Login.Text = "Выход из учётной записи";
}
public void disabled_menu()
{
ts_Enrollment.Enabled = false;
ts_Maintenance.Enabled = false;
ts_students.Enabled = false;
ts_Subject.Enabled = false;
ts_User.Enabled = false;
```

```
ts_Reports.Enabled = false;
}
public void showForm(Form frm)
{
closeForm();
this.IsMdiContainer = true;
frm.MdiParent = this;
frm.Show();
}
private void registerNewStudentsToolStripMenuItem_Click(object sender, EventArgs
e)
{
Form frm = new frmAddStudent();
showForm(frm);
}
private void listOfStudentsToolStripMenuItem_Click(object sender, EventArgs e)
{
Form frm = new frmListofStudents();
showForm(frm);
}
private void enrollmentFormToolStripMenuItem_Click(object sender, EventArgs e)
{
Form frm = new frmEnrollStudents();
showForm(frm);
}
private void listOfEnrolledStudentsToolStripMenuItem_Click(object sender,
EventArgs e)
{
Form frm = new frmEnrolledStudents();
showForm(frm);
}
private void registerNewSubjectsToolStripMenuItem_Click(object sender, EventArgs
e)
{
Form frm = new frmSubjects();
showForm(frm);
}
private void listOfSubjectsToolStripMenuItem_Click(object sender, EventArgs e)
{
Form frm = new frmListofSubject();
showForm(frm);
}
```

```
private void manageAcademicYearToolStripMenuItem_Click(object sender, EventArgs e)
{
Form frm = new frmSy();
showForm(frm);
}
private void manageGradeLevelToolStripMenuItem_Click(object sender, EventArgs e)
{
Form frm = new frmLevel();
showForm(frm);
}
private void manageSectionsToolStripMenuItem_Click(object sender, EventArgs e)
{
Form frm = new frmSection();
showForm(frm);
}
private void ts_Reports_Click(object sender, EventArgs e)
{
Form frm = new frmReports();
showForm(frm);
}
private void ts_User_Click(object sender, EventArgs e)
{
Form frm = new frmUser();
showForm(frm);
}
private void exitToolStripMenuItem_Click(object sender, EventArgs e)
{
this.Close();
}
private void Form1_Load(object sender, EventArgs e)
{
disabled_menu();
}
private void ts_Login_Click(object sender, EventArgs e)
{
if (ts_Login.Text == "Авторизация")
{
Form frm = new frmLogin(this);
frm.ShowDialog();
}
else
{
```

```
disabled_menu();
ts_Login.Text = "Авторизация";
closeForm();
}
}
}
}
```

frmAddStudents.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using EnrollmentSystem.Includes;
using EnrollmentSystem.Properties;
namespace EnrollmentSystem
{
public partial class frmAddStudent : Form
{
public frmAddStudent()
{
InitializeComponent();
}
SQLConfig config = new SQLConfig();
usableFunction func = new usableFunction();
string sql, radioSEX, Ryesno, Nyesno, Pyesno, Cyesno;
private void ts_Close_Click(object sender, EventArgs e)
{
this.Close();
}
private void dtpDbirth_ValueChanged(object sender, EventArgs e)
{
int age;
age = int.Parse(DateTime.Now.ToString("yyyy")) -
int.Parse(dtpDbirth.Value.ToString("yyyy"));
txtAge.Text = age.ToString();
}
private void tsNew_Click(object sender, EventArgs e)
{
func.clearTxt(GroupBox1);
```

```
txtStudentId.Clear();
rdoMale.Checked = true;
rdoFemale.Checked = false;
chkBirthCert.Checked = false;
chkPic.Checked = false;
chkReportCard.Checked = false;
chkCertification.Checked = false;
sql = "SELECT ACADYR FROM ay";
config.fiil_CBO(sql, cboAddSy);
}
private void frmAddStudent_Load(object sender, EventArgs e)
{
tsNew_Click(sender, e);
config.autonumber("StudentID", txtStudentId);
txtLName.Focus();
}
private void tsSave_Click(object sender, EventArgs e)
{
if(txtStudentId.Text == "")
{
MessageBox.Show("Одна ячейка пуста на детали студента, вы должны заполнить
его", "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);
return;
}
//foreach(Control txt in GroupBox1.Controls)
//{
//    if(txt is TextBox)
//    {
//        MessageBox.Show("One of the box is empty on the student details,You must
fill it", "Invalid", MessageBoxButtons.OK, MessageBoxIcon.Error);
//        return;
//    }
//}
if (rdoMale.Checked)
{
radioSEX = "Мужчина";
}
else
{
radioSEX = "Женщина";
}
Nyesno = func.checkOption(chkBirthCert);
Cyesno = func.checkOption(chkCertification);
Pyesno = func.checkOption(chkPic);
```

```
Ryesno = func.checkOption(chkReportCard);
sql = "INSERT INTO `tblrequirements` (`NSO`, `PICID`, `REPORTCARD`, `CERTIFICATE_OF_TRANSFER`, `IDNO`) "
+ "VALUES ('" + Nyesno + "','" + Pyesno + "','" + Ryesno
+ "','" + Cyesno + "','" + txtStudentId.Text + "')";
config.Execute_Query(sql);
sql = "INSERT INTO tblstudent(`IDNO`, `FNAME`, `LNAME`, `MNAME`, `SEX`, `BDAY`, `BPLACE`, `AGE`, `HOME_ADD`, `GUARDIAN`,`GUARDIAN_ADD`,`GUARDIAN_TEL`,`LASTSCHOOLATTEND`,`HEIGTH`,`WEIGTH`) "
+ "VALUES ('" + txtStudentId.Text + "','" + txtFName.Text + "','" + txtLName.Text + "','" + txtMName.Text
+ "','" + radioSEX + "','" + dtpDbirth.Text + "','" + txtPBirth.Text + "','" + txtAge.Text + "','"
+ txtAddress.Text + "','" + txtguardian.Text + "','" + txtguardianadd.Text + "','" + txtguardiantel.Text
+ "','" + txtLastSchoolAttend.Text + "','" + txtHeight.Text + "','"
+ txtWeigth.Text + "')";
config.Execute_CUD(sql, "ошибка при сохранении данных", "Студент был зарегистрирован.");
sql = "Update tblautonumber set EndValue = EndValue + IncrementValue WHERE Remarks='StudentID'";
config.Execute_Query(sql);
tsNew_Click(sender, e);
}
}
}
```

frmEnrolledStudents.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using EnrollmentSystem.Includes;
namespace EnrollmentSystem
{
public partial class frmEnrolledStudents : Form
{
public frmEnrolledStudents()
{
```

```
InitializeComponent();
}
SQLConfig config = new SQLConfig();
string sql;
private void tsFind_Click(object sender, EventArgs e)
{
sql = "SELECT s.`IDNO` as 'Идентификатор', CONCAT(`FNAME` ,' ', `LNAME`,'
', `MNAME`) as 'ФИО', `SEX` as 'Пол' , `AGE` as 'Возраст', `HOME_ADD` as
'Адрес' ,`LEVEL` as 'Уровень образования' , `SECTION` as 'Раздел' , `AY` as 'Год
обучения' " +
" FROM `tblstudent` s, `tblstudupdatelevel` sl WHERE s.`IDNO`=sl.`IDNO` AND (
CONCAT(`FNAME` ,' ', `LNAME`,' ', `MNAME`) LIKE '%" + tsSearch.Text + "%'
OR FNAME LIKE '%" + tsSearch.Text +
"' OR LNAME LIKE '%" + tsSearch.Text + "' OR LEVEL LIKE '%" + tsSearch.Text
+ "' OR SECTION LIKE '%" + tsSearch.Text + "' OR AY LIKE '%" + tsSearch.Text
+
"' OR CONCAT(`LEVEL` ,' ', `AY`) LIKE '%" + tsSearch.Text + "' OR
CONCAT(`LEVEL` ,' ', `SECTION`) LIKE '%" + tsSearch.Text + "' OR
CONCAT(`LNAME` ,' ', `AY`) LIKE '%" + tsSearch.Text + "')";
config.Load_DTG(sql, dtgList);
}
private void frmEnrolledStudents_Load(object sender, EventArgs e)
{
tsFind_Click(sender, e);
}
private void tsPrint_Click(object sender, EventArgs e)
{
Form frm = new
frmPrintEnrolledSubject(dtgList.CurrentRow.Cells[0].Value.ToString());
frm.ShowDialog();
}
}
}
```

frmEditStudent.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using EnrollmentSystem.Includes;
```

```
using EnrollmentSystem.Properties;

namespace EnrollmentSystem
{
public partial class frmEditStudent : Form
{
public frmEditStudent(string studentid)
{
InitializeComponent();
txtStudentId.Text = studentid;
}
SQLConfig config = new SQLConfig();
usableFunction func = new usableFunction();
string sql, radioSEX, Ryesno, Nyesno, Pyesno, Cyesno;

private void tsSave_Click(object sender, EventArgs e)
{
if (rdoMale.Checked)
{
radioSEX = "Мужчина";
}
else
{
radioSEX = "Женщина";
}
Nyesno = func.checkOption(chkBirthCert);
Cyesno = func.checkOption(chkCertification);
Pyesno = func.checkOption(chkPic);
Ryesno = func.checkOption(chkReportCard);
//'student(requirements)
sql = "UPDATE  `tblrequirements` SET `NSO`='" + Nyesno + "', `PICID`='" + Pyesno
+ "', `REPORTCARD`='" + Ryesno
+ "', `CERTIFICATE_OF_TRANSFER`='" + Cyesno + "' WHERE IDNO='" +
txtStudentId.Text + "'";
config.Execute_Query(sql);
//'student basic info
sql = "UPDATE tblstudent SET   `FNAME`='" + txtFName.Text + "', `LNAME`='" +
txtLName.Text + "', `MNAME`='" + txtMName.Text
+ "', `SEX`='" + radioSEX + "', `BDAY`='" + dtpDbirth.Text + "', `BPLACE`='" +
txtPBirth.Text + "', `AGE`='" + txtAge.Text
+ "', `HOME_ADD`='" + txtAddress.Text + "',GUARDIAN='" + txtguardian.Text +
"',GUARDIAN_ADD ='" + txtguardianadd.Text
+ "',GUARDIAN_TEL ='" + txtguardiantel.Text + "',`LASTSCHOOLATTEND`='" +
txtLastSchoolAttend.Text
```

```
+ "', WEIGTH='" + txtWeigth.Text + "',HEIGTH='" + txtHeight.Text + "' WHERE
IDNO='" + txtStudentId.Text + "'";
config.Execute_CUD(sql, "ошибка при обновлении данных", "Информация о
студентах была обновлена.");
}
private void ts_Close_Click(object sender, EventArgs e)
{
this.Close();
}
private void tsNew_Click(object sender, EventArgs e)
{
txtStudentId_TextChanged(sender, e);
}
private void txtStudentId_TextChanged(object sender, EventArgs e)
{
sql = "SELECT ACADYR FROM ay";
config.fiil_CBO(sql, cboAddSy);
sql = "SELECT * FROM tblstudent WHERE IDNO = '" + txtStudentId.Text + "'";
config.singleResult(sql);
if(config.dt.Rows.Count > 0)
{
txtFName.Text = config.dt.Rows[0].Field<string>("FNAME");
txtLName.Text =config.dt.Rows[0].Field<string>("LNAME");
txtMName.Text =config.dt.Rows[0].Field<string>("MNAME");
txtAddress.Text =config.dt.Rows[0].Field<string>("HOME_ADD");
txtguardian.Text =config.dt.Rows[0].Field<string>("GUARDIAN");
txtguardianadd.Text =config.dt.Rows[0].Field<string>("GUARDIAN_ADD");
txtguardiantel.Text =config.dt.Rows[0].Field<int>("GUARDIAN_TEL").ToString();
txtPBirth.Text =config.dt.Rows[0].Field<string>("BPLACE");
dtpDbirth.Text =config.dt.Rows[0].Field<DateTime>("BDAY").ToString();
txtAge.Text =config.dt.Rows[0].Field<int>("AGE").ToString();
txtWeigth.Text =config.dt.Rows[0].Field<int>("WEIGTH").ToString();
txtHeight.Text =config.dt.Rows[0].Field<int>("HEIGTH").ToString();
txtLastSchoolAttend.Text
=config.dt.Rows[0].Field<string>("LASTSCHOOLATTEND");
if (config.dt.Rows[0].Field<string>("SEX") == "Женщина")
{
rdoFemale.Checked = true;
}
else
{
rdoMale.Checked = true;
}
```

```
//    'students requirements
sql = "SELECT * FROM `tblrequirements`  WHERE IDNO = '" + txtStudentId.Text +
"'";
config.singleResult(sql);
if (config.dt.Rows.Count > 0)
{
if(config.dt.Rows[0].Field<string>("NSO") == "Yes")
{
chkBirthCert.Checked = true;
}
else
{
chkBirthCert.Checked = false;
}
if(config.dt.Rows[0].Field<string>("PICID") == "Yes")
{
chkPic.Checked = true;
}
else
{
chkPic.Checked = false;
}
if(config.dt.Rows[0].Field<string>("REPORTCARD") == "Yes")
{
chkReportCard.Checked =true;
}
else
{
chkReportCard.Checked = false;
}
if(config.dt.Rows[0].Field<string>("CERTIFICATE_OF_TRANSFER") == "Yes")
{
chkCertification.Checked = true;
}
else
{
chkCertification.Checked = false;
}
```