

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Сәтбаев университеті

Ақпараттық және телекоммуникациялық технологиялар институты

Киберқауіпсіздік, ақпаратты өңдеу және сақтау кафедрасы

Төлен Бағдат Қасанәліұлы

Сақтандыру компаниясының ақпараттық жүйесін құру

ДИПЛОМДЫҚ ЖҰМЫС

5B070300 – «Ақпараттық жүйелер» мамандығы

Алматы 2019

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Сәтбаев университеті

Ақпараттық және телекоммуникациялық технологиялар институты

Киберқауіпсіздік, ақпаратты өндеу және сақтау кафедрасы

ҚОРҒАУҒА ЖІБЕРІЛДІ

КАӨЖС кафедра меңгерушісі,

техн. ғыл. канд. ,

ассистент профессор

Н. А. Сейлова

« 16 » мамыр 2019 ж.

ДИПЛОМДЫҚ ЖҰМЫС

Тақырыбы: Сақтандыру компаниясының ақпараттық жүйесін құру

5B070300 – «Ақпараттық жүйелер» мамандығы бойынша

Орындаған :

Төлен Б.Қ.

Пікір беруші:

PhD докторы, ассистент проф.

О. А. Баймуратов

« 16 » мамыр 2019ж.

Ғылыми жетекші :

Тьютор

М. Б. Бауыржан

« 13 » мамыр 2019ж.

Алматы 2019

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Сәтбаев университеті

Ақпараттық және телекоммуникациялық технологиялар институты

Киберқауіпсіздік, ақпаратты өңдеу және сақтау кафедрасы

5B070300 – «Ақпараттық жүйелер» мамандығы

БЕКІТЕМІН

КАӨЖС кафедра меңгерушісі,

техн. ғыл. канд. ,

ассистент профессор

 Н.А.Сейлова

« 16 » мамыр 2019 ж.

**Дипломдық жұмысты орындауға
ТАПСЫРМА**

Білім алушы: Төлен Бағдат Қасанәліұлы

Тақырыбы: Сақтандыру компаниясының ақпараттық жүйесін құру

Университет Ректоры: «16» қазан 2018 ж. № 1162-б бұйрығымен
бекітілген

Аяқталған жұмысты тапсыру мерзімі: «17» мамыр 2019 ж.

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: 20 слайдпен көрсетілген

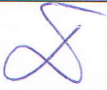
Ұсынылатын негізгі әдебиет: 19 атау

Дипломдық жұмысты дайындау

КЕСТЕСІ

Бөлімдер атауы, қарастырылатын мәселелер тізімі	Ғылыми жетекші мен кеңесшілерге көрсету мерзімдері	Ескерту
Мәселенің қазіргі жағдайына шолу және оны талдау	10.01.2019 – 08.03.2019	
Ақпараттық қамтаманы құру	05.02.2019 – 10.03.2019	
Программалық қамтаманы құру	11.03.2019 – 22.04.2019	

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	А.А.Кабдуллин, кафедра ассистенті	13.05.19	
Программалық қамтама	М.Б. Бауыржан ,тьютор	8.05/19	

Ғылыми жетекші  М. Б. Бауыржан

Тапсырманы орындауға алған білім алушы  Б. Қ. Төлен

Күні “ 08 ” 01 2019 ж.

ҚАЗАҚСТАН РЕСПУБЛИКАСЫНЫҢ БІЛІМ ЖӘНЕ ҒЫЛЫМ
МИНИСТРЛІГІ
СӨТБАЕВ университеті

Ғылыми жетекшінің сын пікірі

Дипломдық жұмысқа

Төлен Бағдат Қасанәліұлына

5B070300 – Ақпараттық жүйелер

Тақырыбы: Сақтандыру компаниясының ақпараттық жүйесін құру

Бұл дипломдық жұмыста “Visual Studio” бағдарламалау ортасында Сақтандыру компаниясының ақпараттық жүйесін құру мәселесі қарастырылған.

Дипломдық жұмысқа түсіндірме хат кіріспеден, қорытындыдан, пайдаланылған әдебиеттер тізімінен тұрады.

Кіріспеде мәселенің маңыздылығы, жұмыстың мақсаты және қолданылған құралдар қысқаша келтірілген. Бұл программа оператордың жұмысын жеңілдетіп қана қоймай, оның уақытын ұтымды пайдалануға да көмектеседі. Ол операторға қажетті мәліметтерді жүйелеп сақтап, оның керек кезінде іздеп электронды іздеу жүйесі, құжаттар бойынша іздеу, қайсыбір құжаттар бойынша есеп берулерді қағазға шығаруды ұйымдастыруды жеңілдетеді.

Төлен Бағдат дипломдық жұмыс орындау барысында өз беттілік көрсетіп, әдебиеттермен жұмыс істеген. Дипломдық жұмыс “Ақпараттық жүйелер” мамандығының бітіру жұмыстарына қойылатын талаптарға сәйкес.

Қорытындылай келе, Төлен Бағдаттың дипломдық жұмысын қорғауға жіберуге болады.

Ғылыми жетекші:

Техника ғылымдарының магистрі, тьютор

 М. Б. Бауыржан

«13» 05 2019 ж

РЕЦЕНЗИЯ

Дипломдық жұмыс

(жұмыс түрінің атауы)

Төлен Бағдат Қасанәліұлы

(білім алушының Т.А.Ә.)

5B070300 – Ақпараттық жүйелер

(мамандық атауы мен шифрі)

Тақырыбы: Сақтандыру компаниясының ақпараттық жүйесін құру

Орындалды:

түсініктеме 48 бет

Төлен Б.Қ. дипломдық жұмысы операторлардың жұмысын онайлатуға арналған. Осы тақырыпқа сәйкес ақпараттық жүйелер негізінде–сақтандыру компаниясының ақпараттық жүйесін құрудағы мәселелердің бірі болып табылады. Бұл жұмыс осы мәселені шешуге арналған. Сол үшін орындалған жұмыс практикалық маңызға ие.

ЖҰМЫСҚА ЕСКЕРТУ

Жұмысқа келесідей ескертулер жасалды:

- түсініктемелік жазбада/анықтамаларда грамматикалық және стилистикалық қателер кездеседі;
- жасалған бағдарламаның сипаттамасын мен талаптары келтірілмеген.

ЖҰМЫСТЫҢ БАҒАСЫ

Дипломдық жұмыс тапсырмаға сәйкес толық орындалған және «90» бағаға бағалап, ал жұмыстың авторы Төлен Бағдат Қасанәліұлы 5B070300 – «Ақпараттық жүйелер» мамандығы бойынша «бакалавр» академиялық дәрежесін алуға лайық деп санаймын.

РЕЦЕНЗЕНТ

PhD докторы, ассистент проф.

«16» мамыр 2019 ж.



О.А. Баймуратов

қолы

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: Төлен Бағдат

Название: Сактандыру компаниясының акпараттық жүйесін құру

Координатор: Мадина Бауыржан

Коэффициент подобия 1:24,4

Коэффициент подобия 2:9,5

Тревога:28

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:

Заимствования являются добросовестными
и не имеют признаков плагиата

Дата 16.08.19

Подпись заведующего кафедрой /

начальника структурного подразделения


К.В.И.И.

Окончательное решение в отношении допуска к защите, включая обоснование:

Допущен к защите

Дата

16.05.1992

Подпись заведующего кафедрой /

начальника структурного подразделения

HH
16.05.1992

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: Төлен Бағдат

Название: Сактандыру компаниясының ақпараттық жүйесін құру

Координатор: Мадина Бауыржан

Коэффициент подобия 1: 24,4

Коэффициент подобия 2: 9,5

Тревога: 28

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:

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

13.05.19^м

Дата



Подпись Научного руководителя

Краткий отчет



Университет:	Satbayev University
Название:	Сақтандыру компаниясының ақпараттық жүйесін құру
Автор:	Төлен Бағдат
Координатор:	Мадина Бауыржан
Дата отчета:	2019-05-04 14:52:44
Коэффициент подоби́я № 1:	24,4%
Коэффициент подоби́я № 2:	9,5%
Длина фразы для коэффициента подоби́я № 2:	25
Количество слов:	4 760
Число знаков:	39 711
Адреса пропущенные при проверке:	
Количество завершнных проверок:	25



К вашему сведению, некоторые слова в этом документе содержат буквы из других алфавитов. Возможно - это попытка скрыть позаимствованный текст. Документ был проверен путем замещения этих букв латинским эквивалентом. Пожалуйста, уделите особое внимание этим частям отчета. Они выделены соответственно.

Количество выделенных слов 28

>> Самые длинные фрагменты, определенные, как подобные

№	Название, имя автора или адрес гиперссылки (Название базы данных)	Автор	Количество одинаковых слов
1	Марал теория — копия.doc Satbayev University (ИИИТТ)	Төреқұлова М	65
2	URL_ http://kz1.fatwords.org/safia/r-trli-jimdardi-tabisti-jmis-jasaul-shin-apatatti-jeni-damian/main.html		59
3	Документ из базы НЭУ 55f67c88-2d04-4094-aa78-3b31c0a80d1d.docx NARXOZ (NEU) (Information Technology Center)	na	57
4	URL_ http://kz1.fatwords.org/safia/r-trli-jimdardi-tabisti-jmis-jasaul-shin-apatatti-jeni-damian/main.html		47
5	URL_ http://kz1.fatwords.org/safia/r-trli-jimdardi-tabisti-jmis-jasaul-shin-apatatti-jeni-damian/main.html		37
6	Документ из базы НЭУ 5447822a-eb8c-42d6-8711-06d5c0a80d1d.doc NARXOZ (NEU) (Information Technology Center)	na	37
7	ҚР-да сақтандыру нарығын бағалау .doc M.Auezov South Kazakhstan State University (Высшая школа Управления и Бизнеса)	ЭФ 16-5к3 тобының студенті Дилмагамбет Гүлім	32

8	Документ из базы НЭУ 5447822a-eb8c-42d6-8711-06d5c0a80d1d.doc NARXOZ (NEU) (Information Technology Center)	na	31
9	Марал теория — копия.doc Satbayev University (ИИИТТ)	Төреқұлова М	29
10	Документ из базы НЭУ 55f67c88-2d04-4094-aa78-3b31c0a80d1d.docx NARXOZ (NEU) (Information Technology Center)	na	28

>>

Документы, в которых найдено подобные фрагменты: из RefBooks



Не обнаружено каких-либо

заимствований

>>

Документы, содержащие подобные фрагменты: Из домашней базы данных

Документы, выделенные жирным шрифтом, содержат фрагменты потенциального плагиата, то есть превышающие лимит в длине коэффициента подобия № 2

№	Название (Название базы данных)	Автор	Количество одинаковых слов (количество фрагментов)
1	Марал теория — копия.doc Satbayev University (ИИИТТ)	Төреқұлова М	231 (19)
2	Almat_plagiat.docx Satbayev University (ИИИТТ)	Джолбарысов А	146 (22)
3	ZHANA55.docx Satbayev University (ИИИТТ)	Сыдыкова К	68 (9)

>>

Документы, содержащие подобные фрагменты: Из внешних баз данных

Документы, выделенные жирным шрифтом, содержат фрагменты потенциального плагиата, то есть превышающие лимит в длине коэффициента подобия № 2

№	Название (Название базы данных)	Автор	Количество одинаковых слов (количество фрагментов)
1	Документ из базы НЭУ 55f67c88-2d04-4094-aa78-3b31c0a80d1d.docx NARXOZ (NEU) (Information Technology Center)	na	148 (8)
2	Документ из базы НЭУ 5447822a-eb8c-42d6-8711-06d5c0a80d1d.doc NARXOZ (NEU) (Information Technology Center)	na	112 (6)
3	ҚР-да сақтандыру нарығын бағалау .doc M.Auezov South Kazakhstan State University (Высшая школа Управления и Бизнеса)	ЭФ 16-5к3 тобының студенті Дилмагамбет Гүлім	37 (2)
4	Документ из базы НЭУ 55f287e4-7960-434e-ad4d-3b43c0a80d1d.doc NARXOZ (NEU) (Information Technology Center)	na	26 (3)

>>

Документы, содержащие подобные фрагменты: Из интернета

Документы, выделенные жирным шрифтом, содержат фрагменты потенциального плагиата, то есть превышающие лимит в длине коэффициента подобия № 2

№	Источник гиперссылки	Количество одинаковых слов (количество фрагментов)
1	URL http://kzref.org/mazmni-kirispe-avtokolikti-satandirudi-aporatti-jjesin-trfizud.html	215 (33)
2	URL http://kz1.fatwords.org/safia/r-trli-jimdardi-tabisti-jmis-jasaui-shin-aporatti-jjeni-damian/main.html	178 (5)

АҢДАТПА

Дипломдық жұмыстың мақсаты клиенттерге сақтандыру түрлері бойынша қызмет көрсету болып табылады.

Құрылған ақпараттық жүйе сақтандыру қызметін пайдалануға тілек білдірген клиенттерді тіркеу, құжаттандыру жұмыстарын атқаратын бөлім қызметкерінің жұмысын автоматтандыруға бағытталған қосымша болып табылады. Бұл программа оператордың жұмысын жеңілдетіп қана қоймай, оның уақытын ұтымды пайдалануға да көмектеседі.

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

АННОТАЦИЯ

Целью дипломной работы является предоставление клиентам услуг по видам страхования.

Созданная информационная система является дополнением к автоматизации работы отдела регистрации и документирования клиентов, желающих воспользоваться страховой услугой. Эта программа не только упростит работу оператора, но и поможет вам использовать его более эффективно.

Во время этой работы использовались базы данных SQLite, язык программирования C # и среда Microsoft Visual Studio.

ANNOTATION

Certified diploma work is valid for clients on insurance coverage.

An extensive information system is available to support the automation of client registrations and documenting clients who are willing to use insurance. Whether the program or operator is just working, though, I can use it more efficiently.

During this time, SQLite has used base data files, C # programming languages, and Microsoft Visual Studio.

МАЗМҰНЫ

КІРІСПЕ	8
1 Сақтандыру қызметінің қазіргі кездегі жағдайына шолу	9
1.1 Қазақстанда сақтандыру қызметі нарығының қалыптасуы	9
1.1.1 Сақтандырудың түрлері, заңды және жеке тұлғалардың автокөліктерін сақтандыруды анықтау және есепке алу	11
1.1.2 Азаматтардың жеке автокөлігін ерікті сақтандыру кезеңдері	12
1.2 Ұқсас жұмыстарға шолу	14
1.2.1 «Цесна Гарант» сақтандыру компаниясының қызметіне шолу	14
1.2.2 «Нұрполис» сақтандыру компаниясы қызметіне шолу	15
1.3 Есептің қойылымы	17
2 Сақтандыру компаниясының ақпараттық қамтамасын құру	19
2.1 Мәліметтер базасын басқару жүйесін таңдауды негіздеу	19
2.2 Қосымшаның функционалдық сұлбасы	20
2.3 Программалау тілін таңдауды негіздеу	22
2.3.1 Visual Studio 2017 ортасын таңдауды негіздеу	22
2.3.2 C# программалау тілін таңдауды негіздеу	23
3 Программалық қамтаманы құру	26
3.1. Программалық қамтаманың құрылымы	26
3.2 Программаның баяндалуы	26
3.2.1 Жалпы мағлұматтар	26
3.2.2 Функционалдық тағайындалуы	27
3.2.3 Логикалық құрылымын сипаттау	27
3.2.4 Қажетті техникалық жабдықтар	27
3.2.5 Шақыру және жүктеу	27
3.2.6 Кіріс мәліметтер	27
3.2.7 Шығыс мәліметтер	28
3.3 Жүйенің жұмыс жасау барысын сипаттау	28
ҚОРЫТЫНДЫ	36
ПАЙДАЛАНЫЛҒАН ӘДЕБИЕТТЕР ТІЗІМІ	37
Қосымша А	38

КІРІСПЕ

Ақпараттандыру – дамыған елдердің қоғамдық қатынастар жүйесінің ең жарқын ерекшеліктерінің бірі. Адамзат ақпараттың адамдар қызметінің барлық саласында айрықша рөл ойнайтын мәдени даму кезеңіне қадам басты. Сонымен бірге ақпарат қазіргі қоғамда экономикалық дамудың ең маңызды факторына айналып келеді.

Ақпараттық технология – бұл айтарлықтай жиі қайталанатын қандай да бір ақпараттық процесті жүйелі түрде ұйымдастыруға мүмкіндік беретін ғылыми білімнің, мағлұматтардың және практикалық тәжірибенің жобалық түрде (практикада қолдануға болатындай құрылған түрде) келтірілген көрінісі.

Ақпараттық технологиялар бүгінгі күні адамдар арасындағы өзара ақпараттық қатынаспен қамтамасыз етуде, сонымен қатар бұқаралық ақпаратты дайындау мен тарату жүйелерінде ерекше маңызды рөлге ие. Бұл құралдар біздің қоғамның мәдениетіне тез сіңісіп кетеді, өйткені олар үлкен қолайлылық тудырып қана қоймай, әлемдік бірлестікті жаһандандыру мен интеграциялау процестерінен, ішкі және халықаралық экономикалық және мәдени байланыстарды кеңейтуден, халықтың қоныс аударуынан және планетада өте қарқынды орын ауыстырудан туындайтын көптеген өндірістік, әлеуметтік және тұрмыстық мәселелерді шешіп береді.

Кез келген басқарушы ақпараттық процестерге ақпаратты тіркеу, жинау, жіберу, сақтау, өңдеу, беру және басқару шешімдерін қабылдау процедуралары кіреді. Ақпараттық технологиялар – осы процедураларды әр түрлі ақпараттық жүйелерде жүзеге асыратын құралдар мен әдістер болып табылады.

Қазіргі заманғы ақпараттық жүйелер қамсыздандырушы жүйешелердің бірнеше түрінен құралады, оларға: техникалық, бағдарламалық, ақпараттық, ұйымдық, құқықтық және эргономикалық қамсыздандырулар жатады.

Әлеуметтік уақытты үнемдеудің қажеттілігі біздің назарымызды, біріншіден, бұқаралық ақпараттық процестермен байланысты технологияларға аудартады, олардың жүзеге асырылуы оларды кең көлемде және қайта-қайта қолданудың арқасында әлеуметтік уақытты барынша үнемдеуі тиіс.

Қазіргі заманғы ақпараттық технологияларды енгізу бірқатар талаптарға сай болу керек, оның ішінде қолайлы және достық интерфейстің болуы, ақпараттық ресурстарды әр түрлі бақылау әдістері мен оларға қатынас жасауға шек қою арқылы қауіпсіздікті қамтамасыз ету, жүйелерді құрудың модульді принципі, Интернет технологиясын қолдауы және т.б. бар.

Аталғандарды ескеріп, мен өзімнің дипломдық жұмысымда сақтандыру компаниясының ақпараттық қосымшасын құру моделін көрсеткім келеді. Жұмыстың бірінші сақтандыру компаниясы қызметінің даму мәселесінің қазіргі жағдайына шолу және оны талдау, екінші бөлімінде ақпараттық қамтаманы құру, үшінші бөлімінде сақтандыру компаниясының программалық қамтамасы қарастырылады.

1 Сақтандыру қызметінің қазіргі кездегі жағдайына шолу

1.1 Қазақстанда сақтандыру қызметі нарығының қалыптасуы

Сақтандыру түрлерінің барлық жиынтығын сақтандыру объектісі бойынша екі үлкен салаға бөлуге болады: автокөлікті сақтандыру және жеке тұлғаны сақтандыру.

Автокөлікті сақтандыру кезінде сақтандырушы қатынастар объектісі ретінде автокөліктердің барлық түрлері болады. Автокөлікті сақтандыру экономикалық негізі бұл сақтандырушы жағдайы салдарынан болған зиян орнын толтыру арқылы бекітіледі.

Сақтанатын болып сақтанушы меншігіндегі, яғни оның сенімінде болатын, өкілеттігінде және қолдануында болатын автокөлікті айтуға болады. Сақтандырушы дегеніміз – тек автокөліктердің ғана меншіктері емес, сонымен бірге автокөліктің сақталуына жауапты заңды жеке тұлғаларды айтамыз.

Кәсіби қызметті алатын болсақ кәсіпкер негіздемелері мен өз міндеттемелері бұзылуынан әйтпесе кәсіпкерден тәуелсіз жағдайлардан осындай қызметтің шараларының өзгеруінен күтілетін табыстарды ала алмауымен сақтануды айтады.

Азаматтық және құқықтық жауапкершілік түрін сақтандыру барысында басқа адамдар өміріне, денсаулығына зиянның келуінен пайда болатын сақтандыру объектісі болып, негізінен келісімдер болмаса басқа тәуекелдерден пайда болатын тапсырмалар бойынша жауапты болып табылады.

Тәжірибе жүзінде сақтандыру түрлерінің саны шексіз болады және әлеуметтік сақтанушылар мүдделерінің әртүрлігіне тәуелді болып келеді.

Мысал келтіретін болсақ, бізде мемлекетімізде ең көп таралған сақтандыру түрлерін қарастырайық.

Заңды болмаса жеке тұлғалардың автокөліктерін сақтандыру барысында көлік құралдарын және онымен байланысты материалдарды сақтандыру болып есептеледі. Жауапкершілік қызметін сақтандыру бойынша автокөлік иелерінің азаматтық және құқықтық жауапкершілік түрін сақтандыру мен тасымалдаушылар жауаптылығын сақтандыру түрі, жұмыс беруші ұйымдардың жауапкершілік түрін сақтандыру, үшінші жақ өкілдеріне қатысты зиянкестік үшін кәсіпорын азаматтық жауапкершілік қызметін сақтандыру (экологиялық сақтандыру түрі), кәсіби жауапты сақтандыру түрі және зейнеткерлік қор жауапты сақтандыру болып табылады. Кәсіптік тәуекелділіктерді сақтандыру түрлері өндіріс процесінде болмаса арнайы қызметтер берудегі анықталған тәуекелге байланысты болады. Оларға: өндірісте болатын үзіліс, жаңа техниканы иелену мен валюта бағаларының ауытқуы және басқалар жатады.

Негізі келетін қауіп сипаттамасы бойынша бөлшектеу автокөлікті ғана сақтандыруды қамтиды және бір-бірімен иерархиялық байланыста болмайтын 5 негізгі буындардан тұрады. Бұлар:

1. Төтенше жағдайлардан және оттан сақтандыру.

2. Зиянкестер мен табиғат апаттарынан сақтандыру.

3. Көліктің бәрлық түрін апаттардан, мініп кетуден және басқа да апаттардан сақтандыру.

4. Катастрофалар мен техногендік апаттардан сақтандыру.

5. Финанстық тәуекелдерден сақтандыру.

Қауіптер мінездемесі бойынша бөлшектеу зиянды анықтау мен сақтану орнын толтыру тәсілдерін дамыту барысында пайдаланылады. Көрсетілген буындардың әрбірі берілген сыныптама аясындағы сақтандырудың жекеше түрі болып келеді.

Сақтандыру төлемін төлеп болғаннан кейін сақтандырушы, егер оны төлеу кейінгі уақытқа қалдырылса – ең бірінші сақтандырушы салым уақытынан бастап сақтандыру келісімі күшіне енеді және қарама-қарсы жақтар үшін ол міндетті болады, бірінші сақтандырушы жағдайдан сақтандырушы орнын толықтыруды төлеу мерзімінен бастап міндетті сақтандыру жайлы келісіммен болмаса заң арқылы басқаша қарастырылмаса, өз қызметін тоқтату қажет болады.

Сақтандырушы қорғау қызмет беру кезеңі, егер міндетті сақтандыру жайлы келісіммен болмаса заңмен басқаша қарастырылмаған болса, онда келісім қызметі мерзімі сай келеді.

Жарлық арқылы сақтандыру келісімін мерзіміне дейін тоқтатуға болмаса оны жарамсыз деп тану тәртібі мақұлданған.

Азаматтық кодексі қарастырылған міндеттерді тоқтатудың жалпы негіздерінен басқаша сақтандыру келісімі келесідей жағдайларда тоқтатылу керек болады:

- сақтандыру объектісі жоғалған кезде;
- оны алмастыру болмаған кезде сақтандырушы болмайтын сақтанушы өлімі;
- егер де сақтандырушы сақтанушының ауысуына қарсы болса, ал келісім арқылы болмаса заң арқылы басқаша қарастырылмайтын болса, сақтандырушының мүлікті сақтандыру объектісінен бас тартуы кезінде;
- егер де басқа кезеңдер қарастырылмаса, сақтанушы арқылы кезекті сақтандыру салымдарын арнайы мерзімге дейін төлемей қоюы;
- сақтандырушы және сақтанушы талаптары, егер сақтандыру шарттарымен яғни қарсы жақтардың келісімімен қарастырылатын болса.

Келісімге отырған уақыт бойына сақтандырушының сақтандыру төлемін көбейту талабына сай, онымен бірге негізгі келісім аяқталу уақытына дейінгі қосымша келісімге қол қоюына мүмкіндігі бар. Мұндай жағдайда негізгі және қосымша келісім шарт бойынша сақтандырушы төлем берілген автокөлікті сақтандыру ережесі бойынша нақты мөлшерден жоғары болуы мүмкін емес.

Бір жылға келісім шартқа отырған келісімдер бойынша сақтандырушы төлемдері сақтанушы арқылы бекітілген мөлшерлеме бойынша төленеді.

Бір жылдан кем уақытқа қол қойылған келісім шарттар бойынша сақтандырушы төлемдері жылдық төлем сомасынан келесі мөлшерлеме бойынша төленеді: 2 айға – 30%, 3 айға – 40%, 4 айға – 50%, 5 айға – 60%, 6

айға – 70%, 7 айға – 75%, 8 айға – 80%, 9 айға – 85%, 10 айға – 90%, 11 айға – 95% мерзімдеріне сақтандыру.

Жоғарыда келтірілген мөлшерлемелерде және сақтандырушы сомасының өсуіне қарай келісілген қосымша келісім шарттар бойынша сақтандырушы төлемдері төленуі мүмкін. Мұндай жағдайда ай толық болмаса да төлем толық төленуі керек болады [3].

Тәжірибе жүзінде сақтандырушылар екі болмаса үш жылға келісілген уақытпен автокөлікті сақтандырудың жалпы келісімдері бойынша төленген соманың 10 пайызына дейін, ал төрт немесе бес жылға – 15 пайызға дейін жеңілдіктер ұсынады.

Ал егер 5 және одан көп жылдар бойы үзіліссіз автокөлікті сақтандыру келісім шарттарына қол қойған және осы уақыттың ішінде сақтандырушы орнын толтыра алмаған тұрақты сақтандырушыларға жаңа келісім шартқа отытру кезінде төленетін соманың 20 пайызына дейін жеңілдік беріледі [4].

1.1.1 Сақтандырудың түрлері, заңды және жеке тұлғалардың автокөліктерін сақтандыруды анықтау және есепке алу

Өзінің сақтанушыларына кездейсоқ оқиға болған жағдайда жоғалған, құртылған көліктер үшін материалдық өтем беретініне кепілдік беріп, сақтандыру ұйымдары олардың әлеуметтік жағынан қорғалуды іске асырады және оны барынша толық көлемде жасауға бар күшін салады. Осы жағынан алғанда азаматтар мен барлық жүйелердегі кәсіпорын автокөліктерін ерікті түрде сақтандыру қызметінің кейбір жеке түрін енгізу жайында ұсынысқа көңіл бөлуге болады.

Осылайша, жер сілкінісі – адамдардың кездейсоқ оқиғадан қорғана алуының бір әдісі деп айтсақ болады. Сондықтан мемлекеттік коммерциялық сақтандыру компаниясының басқармасы мемлекетімізде жер сілкіну жағдайында ерікті түрде сақтандыруды енгізіп отыр. Сонымен қатар жер сілкінісі зардабын жою жөнінде жасалатын жұмыстарды ұйымдастыру тәртібі, келетін зардаптарды анықтау және құрылыстарға, үй мүліктеріне, малға, көлік құралдарына жеке сақтандыру түрі бойынша сақтандыру өтемін төлеу жолдары жан-жақты түсіндірілетін болады.

Жер сілкінуден сақтандыру, шын мәнінде орасан зор апатты оқиғадан алдын ала сақтандыру болып табылады, сол себепті сақтандыру куәлігі франшиз арқылы берілуі тиіс, яғни оны сақтандыру сомасының белгілі проценті болмаса сақтандырушыға төленетін белгілі сомадан құралады. Сақтандыру өтілін уақытымен және толық төлеп отыру сақтанушыға бүлінген автокөлікті қысқа мерзім ішінде жөндеп немесе жойылған көліктің орнына жаңасын сатыпалуға мүмкіндік береді. Сондықтан жер сілкіну салдарынан болатын шығынды анықтау сақтандыру ұйымдарының өте күрделі және жауапты қызметтері болады. Бұның негізгісі етіп техникалық есепке алу

бюросының бағасын алуға болады. Мұнда шығынды анықтау және сақтандыру төлемін төлеу үшін мынадай кезеңдер алынады:

- сақтандыру оқиғасы арқылы барды түгендеу және автокөлік жойылуы мен зақымданып бүліну себебін анықтау;
- шығынды анықтау;
- сақтандыру өтілінің шығынын есептеу;
- сақтанушы есептелген өтілінің толық болмаса жартылай сомасын алуға құқығын белгілеу;
- сақтанушыға сақтандыру өтілін төлеу.

Келтірілген зиян мен сақтандыру өтілін жеке-жеке қараудың қажеттілігі, көптеген жағдайларда сақтандыру жеке түрлерінің белгіленген келісімдеріне сай болмаса сақтандыру түрінің жүйесін пайдалануға байланысты олардың мөлшері сәйкес келмеуі мүмкін. Бірақ та сақтандыру өтілінің сомасын есептеу үшін бастапқы көрсеткіш алынады.

Шығын құнына сақтандыру оқиғасы кезінде бар-жоғы құжаттармен немесе сақтандыру оқиғасы болған жер мен қалдықтарды көру кезінде дәлелденбеген болса, автокөліктердің құны есептелмейді. Сақтанушыға тиесілі автокөліктің жойылуы немесе бүлінуі сақтандыру өтілі автокөліктің сақтандырылған шығын сомасы процентінде яғни ерікті түрде сақтандыру туралы өтініште көрсетілген сақтандыру сомасынан асып кетпеуі керек.

Негізгі материалдар қатарына жататын автокөліктердің тозып кетуі бойынша жеңілдік беру амортизациялық жарна үстемесі негізіндегі заң бойынша белгіленген негізгі қор бойынша анықталады. Кооперативтік ұйымдардың сақтандырылған автокөліктері үшін сақтандыру өтілі, сақтандыру дәрежесінің мөлшеріне қарамастан толық көлемде төленуі қажет [3].

Шығынның жалпы түрдегі формуласы мынадай болады:

$$CC = \frac{100 + \text{Баллы}}{100} * \frac{\text{БНС}}{0,96 - \text{АШ}} \quad (1.1)$$

мұндағы,

СС – сақтандыру сыйақысы;

БНС – базалық нетто – сыйақы;

АШ – аквизициялық шығындардың коэффициенті (әр регионға жарлықпен қойылады);

0,96 – 4% салықты есептейтін коэффициент [4].

1.1.2 Азаматтардың жеке автокөлігін ерікті сақтандыру кезеңдері

Жеке тұлғалардың автокөлігін ерікті түрде сақтандыру шарты сақтандыру ұйымдары арқылы бір мерзімге құрылады. Бұндай шарт бойынша сақтандыру жеке тұлғалардың жеке меншігіне жататын автокөліктер және республика аймағында уақытша болатын кезінде (жұмыста, жол үстінде, іссапарда, демалыста, көшеде) қолданатын көліктер жатады.

Сақтандыру екі түрде жүзеге асырылады:

1. 0,5 мың теңгеден 1,0 мың теңгеге дейін сақтандыру сомасымен, яғни тариф бойынша 1% төлем төлеу арқылы сақтандыру.

2. 0,1 мың теңгеден 3,0 мың теңгеге дейін сақтандыру сомасына, яғни тариф бойынша 2 % төлем төлеу арқылы сақтандыру.

Жеке автокөлікті сақтандыру ұрланған жағдайда болмаса сол сияқты жойылып кеткенде (немесе одан әрі пайдалануға жарамсыз болған жағдайда), сонымен бірге басқа бір адамның қасақана жасаған жөнсіз тірлігінің салдарынан қорғау үшін болмаса аяқ-астынан болатын жазатайым оқиғалардан: жарылыстан, өрттен, көлік құралдарын қағып кетуден, басып кетуден, жылу жүйесі, су құбырлары, көрші үй жақтан су аққанда, құрылыстың басты конструкциясы күтпеген жерден бұзылған жағдайда жүргізіледі.

Құрылатын шарт сақтандыру куәлігінің бланкісі арқылы тіркеледі. Жеке автокөлікті сақтандыру келісім-шарты төмендегідей жағдайларда күшіне енеді:

- қолма-қол ақшамен төлеген жағдайда;
- ақшасыз есеп айырысқан жағдайда – еңбекақы беретін күн белгіленген күннен бастап сақтандыру төлемі аударылған кезде;
- салымнан алып ақша аударғанда – сақтандыру компанияларының есеп-шотына төлем түскен күннің ертесі.

Келісім-шарттың іс-әрекеті бір жыл өткен соң, яғни келісім-шарт күшіне енген күннен бір күн бұрын аяқталады. Осыдан бұрын үш болмаса одан көп жылдар бойына үздіксіз сақтандыру шартына отырған болса, сондай адамға жаңа мерзімді шартқа отыру барысында төлемнің 20 пайызы мөлшерінде жеңілдік алуға құқығы бар [2].

Сақтандыру оқиғасы басталған сәттегі сақтанушының міндеттері:

- көлік ұрланған болса, адамның қасақана жасаған жөнсіз әрекеті салдарынан ол жойылған болса бұл жайлы полиция органына, ал өрт шыққан болса - өртті қадағалау органдарына, жарылыс болмаса апат болған жағдайда – тиісті авария қызметтері органына хабарлау керек;
- біртәулік мерзімінде сақтандыру оқиғасы болған аймаққа қарайтын ауданның сақтандыру органдарына хабарласу қажет. Жазбаша өтініште мұндай жағдайлар толық көрсетіледі: қашан, қай жерде және қандай жағдайда сақтандырылған көлік бүлінді немесе ұрланды, тиісті органдарға бұл жөнінде хабарлап, өзінің тұратын жерін айтады;
- жойылған болмаса ұрланған автокөліктердің сақтандыру ұйымдарынан алған нысандары бойынша тізімдері құрылады;
- акт құру кезінде сақтандыру ұйымының өкіліне бүлінген заттардың қалғанын көру үшін барлық мүмкіндіктер келтіріледі.

Акт құру барысында сақтандыру ұйымы сақтанушы мен екі куәгердің міндетті қатысуы арқылы жүргізеді. Мұнымен бірге, акт құруға қоғамдық өкіл де шақырылуы керек. Сақтандыру өтілін төлеуді сақтандыру шартын жасаған сақтандыру ұйымы тиісті құжаттар менқұрылған актіні алысымен жүргізеді. Зардаптың мөлшері олар жасаған акті мен тиісті органдардың құжаттарының негізінде анықталады. Сақтандыру өтілі жойылған автокөліктің нақты құны болмаса сақтандыру шарты бойынша белгіленген мөлшерде төленуі тиіс.

Сақтандыру өтілі төленбейтін жағдайлар:

- сақтандыру оқиғасы сақтанушының қасақана жасаған әрекеті салдарынан шығынға ұшыраса, қасақана жасалған қылмысты тергеу органдары анықтаған болса;
- басқа бір адам автокөлікті ұрлап кетсе немесе жойғысы келсе, бірақ сақтанушы бұл жайында мүмкіндігі бола тұрып полиция органына айтпаған болса, сонымен бірге бұл фактіні полиция органы растамаса;
- тек қана құралдар немесе материалдар қалдықсыз жоғалып кеткеннен басқа кездегі бүлінген заттардың қалдықтары көрсетілмейді.

Егер сақтанушы жоғалып кеткен заттардың ішіне құрымаған затты әдейілеп қосып жіберсе, онда жоғарыда тұрған сақтандыру компаниясы мәселенің мән-жайын зерттеп алып, төленуге тиісті сақтандыру өтілінің мөлшерін 50 пайызға дейінгі шекте төмендету жайлы шешім қабылдауы да мүмкін.

Егер құртылған зат орнына келтірілсе, онда сақтанушы сақтандыру өтілін алу құқығынан айырылады. Егер сақтанушыға ұрланған зат қайтарылатын болса, онда ол сол зат үшін алған сақтандыру өтілін сақтандыру мекемесіне қайтаруға міндетті, бірақ бұдан ұрлануға байланысты жөндеуге болмаса қайтарылған затты орнына келтіруге кеткен шығындар ұсталынады.

1.2 Ұқсас жұмыстарға шолу

1.2.1 «Цесна Гарант» сақтандыру компаниясының қызметіне шолу

«Цеснабанк» АҚ-ның еншілес ұйымы «Цесна Гарант» Сақтандыру компаниясы АҚ нарықта 2009 жылдан бері жұмыс жасап келе жатқан Қазақстанның әмбебап сақтандыру компаниясы болып табылады [11].

Компания «Қазақстан Республикасының сақтандыру төлемдеріне кепілдік беру қоры» АҚ қатысушы болып табылады, бұл әрбір клиентке арналған сақтандыру төлемінің уақытымен төленуіне нақты кепілдік бере алады және де клиенттерге арналған сенімді түрде сақтандыру арқылы қорғауды қамтамасыз ете алады. Соңғы жылдардың ішінде компания өте үлкен жетістіктерге жетті. Компанияның жұмыс жасау ережесі, ол клиенттің барлық қажеттіліктерін толық қанағаттандыра білу мен клиентке қажетті сақтандырудан сапалы түрдегі қорғауы мен ең үлкен жылдамдықты қамтамасыз ету болып табылады.

Компанияның алатын орны мен дамуына қызмет көрсету стандарты құрылуына компания акционерлерінің үлесі өте жоғары деп айтуға болады. Мұндағы акционер Қазақстанда ең үлкен банктердің бірі – Компанияның 100% акционері – **«Цеснабанк» АҚ**.

Компанияның жарғылық капиталы негізінен 1 млн 364 мыңтеңгені құрайды және 100 пайыз ақшамен қамтамасыздандырылған. Компанияның өте үлкен көлемде сақтандыру сыйлықтары жеткілікті қорды жинақтауға мүмкіндік береді. Компания капиталының бірқалыпты ұлғаюының көлемі, кәсіби

менеджмент пен инвестициялық саясаттың компанияға Қазақстан Республикасында ірі институционалды инвестор болуына мүмкіндік бере алады. Сонымен қатар компания өз белсенділігін жеке қызмет көрсету саласында дамытып отырады, бөлшектенген бөліктегі сақтандыру өнімдерін жаңартып отыруға, оларды құруға өте үлкен көңіл бөлінеді, сақтандыру саласында заманауи технологияларды пайдалану мен қажетті қызметтердің тізімін кеңейтуге болады.

Қазіргі кезде сақтандырудың барлық қызметтері халыққа кең көлемде ұсынылады. Клиенттердің ұсынысына қарай сақтандырудың өте көп түрлері жасалған (1.1-сурет).



1.1-сурет – «Цесна Гарант» сақтандыру компаниясы негізгі парақшасы

1.2.2 «Нұрполис» сақтандыру компаниясы қызметіне шолу

Ең алғаш «Нұрбанк» акционерлік қоғамы Атырау қаласында 1992 жылдың 3-тамызында құрылған. 2003 жылы компанияның бас офісі Алматы қаласына көшті [12].

«Нұрполис» сақтандыру компаниясының құндылықтарына мыналар жатады: сапалы қызмет көрсету және клиенттер сұранысына оперативті жауап беру, еліміздегі сақтандыру шарттарына заманауи бизнес-процестерді енгізу, тұрақты клиенттермен ұзақ мерзімге келісім-шарт жасасу.

Міндетті сақтандыру түрлеріне:

- Көлік құралдарының иелерінің азаматтық және құқықтық жауапкершілік түрін сақтандыру;

- Тасымалдаушының жолаушылар алдындағы азаматтық және құқықтық жауапкершілік түрін сақтандыру;
- туроператор мен турагенттің азаматтық және құқықтық жауапкершілік түрін сақтандыру;
- үшінші тұлғаларға қауіп төндіретін нысандардың иелерінің азаматтық және құқықтық жауапкершілік түрін сақтандыру;
- міндетті түрдегі экологиялық сақтандыру.

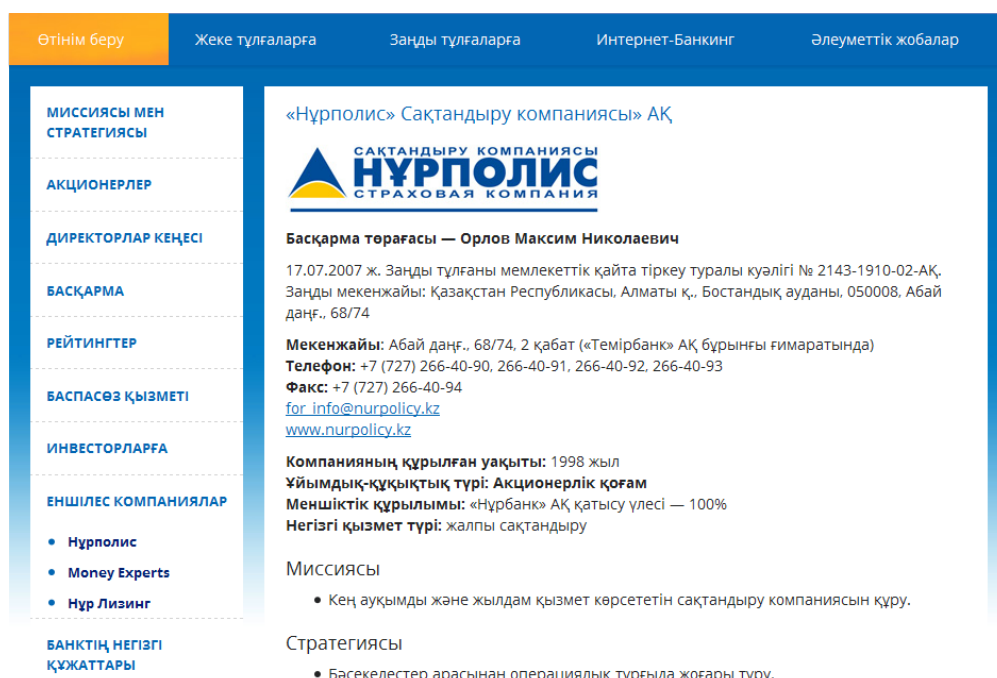
Еркін түрде сақтандыру түрлері:

- қайғылы оқиғалардан сақтандыру қызметі;
- ауырып қалу жағдайының сақтандыру қызметі;
- автокөлікті сақтандыру қызметі;
- әуе көлігін сақтандыру қызметі;
- ауыр жүкті сақтандыру қызметі;
- мүлікті қандай да қауіптен сақтандыру қызметі;
- қаржы шығындарынан сақтандыру қызметі;
- әуе көліктері иелерінің азаматтық және құқықтық жауапкершілік түрін сақтандыру қызметі;
- автокөлік иелерінің азаматтық және құқықтық жауапкершілік түрін сақтандыру қызметі;
- азаматтық және құқықтық жауапкершілік түрін сақтандыру қызметі;
- қарыздар алу мен беруді сақтандыру қызметі.

«Нұрполис» СК қызметі кең көлемдегі аймақтық жүйенің арқасында Қазақстанның барлық аумағында қол жетімді болады.

Қысқа мерзім ішінде «Нұрбанк» акционерлік қоғамы республикалық деңгейге аймақтық банктен көтеріліп, Қазақстан банктерінің алғашқы ондығына орналасты. Қазіргі кезде «Нұрбанк» акционерлік қоғамы кейбір көрсеткіштер бойынша сегізінші орында тұр, ол банктік және сақтандыру қызметтерінің барлық түрлерін ұсынуға дайын (1.2-сурет).

Шетелдік кредиторлар, халықаралық рейтингтік агенттіктер, аудиторлар бұл сақтандыру компаниясы активтерінің, тәуекел және менеджмент жүйесінің сапасының жоғары екендігін және банктің автоматтандырылған деңгейінің биік екенін мойындайды. Республикамыз барлық ірі аймақтарында он алты филиал, жетпіс бес әмбебап орталық сақтандыру бойынша және банктік қызмет көрсетуге дайын.



1.2-сурет – «Нұрполис» Сақтандыру компаниясы негізгі парақшасы

1.3 Есептің қойылымы

Жүйені құру үшін көптеген автоматтандырылған бағдарламалық жүйелермен құрылғылар қолданылады, олардың ішінде бухгалтерлік есептеулер жүйесі, электронды деканат, оқу жүктемесін есептеу, қызметкерлер, студенттер туралы электронды мәліметтер қоры және тағы да басқалар бар. Ал сақтандыру компанияларында жұмыс істейтін қызметкерлер сақтандыру туралы мәліметтерді, яғни өз автокөліктерін немесе өмірін сақтандыру жайлы ақпараттарды және т.с.с білу үшін осы бағдарламаны қолданса, керек мәліметтер алдында дайын тұрады. Жасалған жұмыстарға шолу жасай келіп сақтандыру ұйымы қызметкерлерінің жұмысын жеңілдету мақсатымен «Сақтандыру компаниясының ақпараттық қосымшасын құру» бағдарламасы құрылады.

Жұмыстың мақсаты клиенттерге әртүрлі қызметтер көрсету, соның ішінде сақтандырудың барлық түрлерін зерттеу болып табылады.

Қойылған мақсатқа жету барысында келесі есептер қарастырылады:

- сұраныстарды қабылдау және ерікті сақтандыру қызметтерінің жұмысын зерттеп, бағдарламалау;
- клиенттерді қабылдау және тіркеу технологиялық үрдісінің сұлбасын сараптап, оларды тіркеу;
- клиенттерге қосымша қызмет түрлерін таныстыру және оларды жұмыс аясына енгізу;
- көрсетілетін қосымша қызметтерге кез келген уақытта тапсырыс беру жүйесін бағдарламалау;

- сақталып тұрған тізім ішінен керекті клиентті жеке мәліметтері бойынша іздеу жүйесі арқылы тез тауып алу;
- сақталатын ақпарат қауіпсіздігін қамтамасыз ету, жеке аккаунт құру және мәліметтерге рұқсат беруге белгілі бір құқықтарды тағайындау;
- клиент жайлы енгізілген мәліметтер негізінде екіжақты сақтандырудың келісім-шартын құру;
- міндетті сақтандыру мен ерікті сақтандыру түрлерінің күрделі нарядтарын анағұрлым нақты, жылдам және сапалы өңдеу;
- бұрын қызмет көрсетілген және қызмет көрсетіліп жатқан клиенттер жайлы ақпаратты мұрағатқа сақтау.

Дипломдық жұмысты жасау барысында C# бағдарламалау ортасы мен мәліметтер қорын басқару жүйесі ретінде MS SQLite таңдалады.

2 Сақтандыру компаниясының ақпараттық қамтамасын құру

2.1 Мәліметтер базасын басқару жүйесін таңдауды негіздеу

Мәліметтер базасы ақпаратты алу, беру, сақтау мен мәліметтерге тез, тиімді қатынауды қамтамасыз етеді. Мәліметтер базасы – арнайы ережелерге сәйкес ұйымдастырылған әртүрлі сипатта берілетін мәліметтердің жиынтығы.

Мәліметтер базасын басқару жүйесі – мәліметтер базасын құру, енгізу және қолдануға арналған программалық және тілдік құралдар жиынтығы болып табылады. Қолдану саласына байланысты МББЖ дербес және көпқолданушылық болып бөлінеді. Дербес МББЖ тек қана бір компьютерде жұмыс жасайтын локальды мәліметтер базасын құру мүмкіндігін қарастырады.

Көпқолданушылық МББЖ "клиент-сервер" архитектурасы сипатында қызмет ететін ақпараттық жүйелерді құрудың мүмкіндігін береді. Қазіргі кездегі МББЖ тілдік құралдары құрамына кіреді:

- мәліметтердің логикалық құрылымын беруге арналған мәліметтерді сипаттау тілі;
- мәліметтерге амалдар енгізу, өзгерту және таңдауды орындауды қамтамасыз ететін мәліметтерді басқару тілі;
- МБ құрылымын басқаруды қамтамасыз ететін және мәліметтерді басқаратын, сондай-ақ өшірілген МБ-на қатынау стандартты құралы болып табылатын құрылымданған сұраныстар тілі (SQL-Structured Query Language);

Файл-серверлік мәліметтер базасының автономды мәліметтер базасынан ерекшелігі, олар желінің көмегімен көптеген клиенттерге қатынас жасауға мүмкіндік береді. Бұл мәліметтер базасында болатын өзгерістерді барлық қолданушылар көретін болғандықтан өте ыңғайлы. Мәліметтер базасы желілік файлдық-серверде бір түрде ғана сақталады, әрбір клиентке ол жұмыс жасау үшін жергілікті көшірмесі құрылады. Бұл жағдайда бірнеше қолданушының бір мәліметті өзгерту мәселесі туындайды. Файл-серверлік мәліметтер базасының кемшілігі желінің тиімсіз жүктелуінде. Клиент әрбір сұраныс тастаған кезде жергілікті көшірмедегі мәлімет серверде сақталған мәліметтер базасынан толығымен жаңартылады. Егер де келіп түсетін сұраныс бір жазбаға ғана қатысты болса да сақталған барлық жазбалар жаңартылады. Егер мәліметтер базасында сақталған мәліметтер өте көп болса, ал клиент саны аз болған жағдайда желі едәуір жүктелген болып тұрады да, бұл жағдай сұраныстардың жылдамдығына әсер етпей қоймайды.

Клиент-сервер мәліметтер базасының қолданушылары өте көп және үлкен мәліметтер базасына ие. Көбінде клиенттердің мәліметтер базасына қатынауын арнайы компьютер-сервер орындайды. Клиент серверге арнайы іздеу немесе мәліметтер базасын жаңарту операцияларын орындау үшін тапсырмалар береді. Сол кезде сұраныстар арқылы операцияларға келтірілген қуатты сервер тиімді жолмен сұраныстарды орындап, өзінің жұмысының нәтижесін клиентке хабарлайды. Мұндай жұмысты ұйымдастыру программаның орындалуын жылдамдатады, желіні босатады және мәліметтің

бірегейлігін жақсы бақылай алады. Клиент-сервер мәліметтер базасында қосымша сервердің ең төмен жұмыс жасауы мен ең аз мөлшерде ақпарат беру үшін желінің жүктелуін қамтамасыз ететіндей құрылуы қажет. Бұл серверде сақталатын процедуралардың көмегі арқылы жүзеге асырылады. Көп деңгейлі тармақталған мәліметтер базасы – бұл сәйкес түрде бекітілген тармақталған қосымшалар көмегімен желідегі мәліметті өңдеу. Ең көп таралғаны үшдеңгейлі түрі:

- төменгі деңгейде қолданушы компьютерінде клиенттердің қосымшасы орнатылған, ол қолданушы интерфейсін қамтамасыз етеді;
- екінші деңгейде қосымшалар сервері орнатылады, қолданушылар мен тармақталған мәліметтер базасы арасында мәлімет алмасуды қамтамасыз етеді. Қосымшалар сервері барлық клиенттерге қол жетерлік желі түйінінде орналасады;
- үшінші деңгейде мәліметтер базасының өшірілген серверінен қабылдап оларды өңдеумен айналысады.

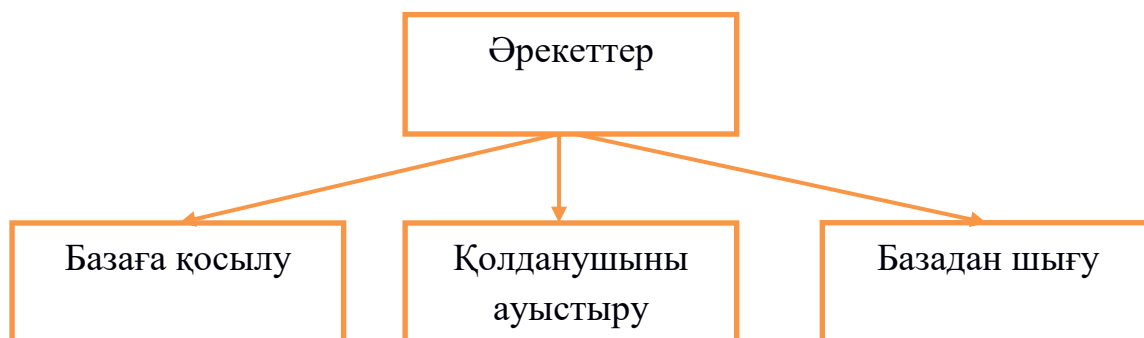
2.2 Қосымшаның функционалдық сұлбасы

Қосымшаның негізгі бетінің функционалды сұлбасы келесі 2.1-суретте келтірілген.



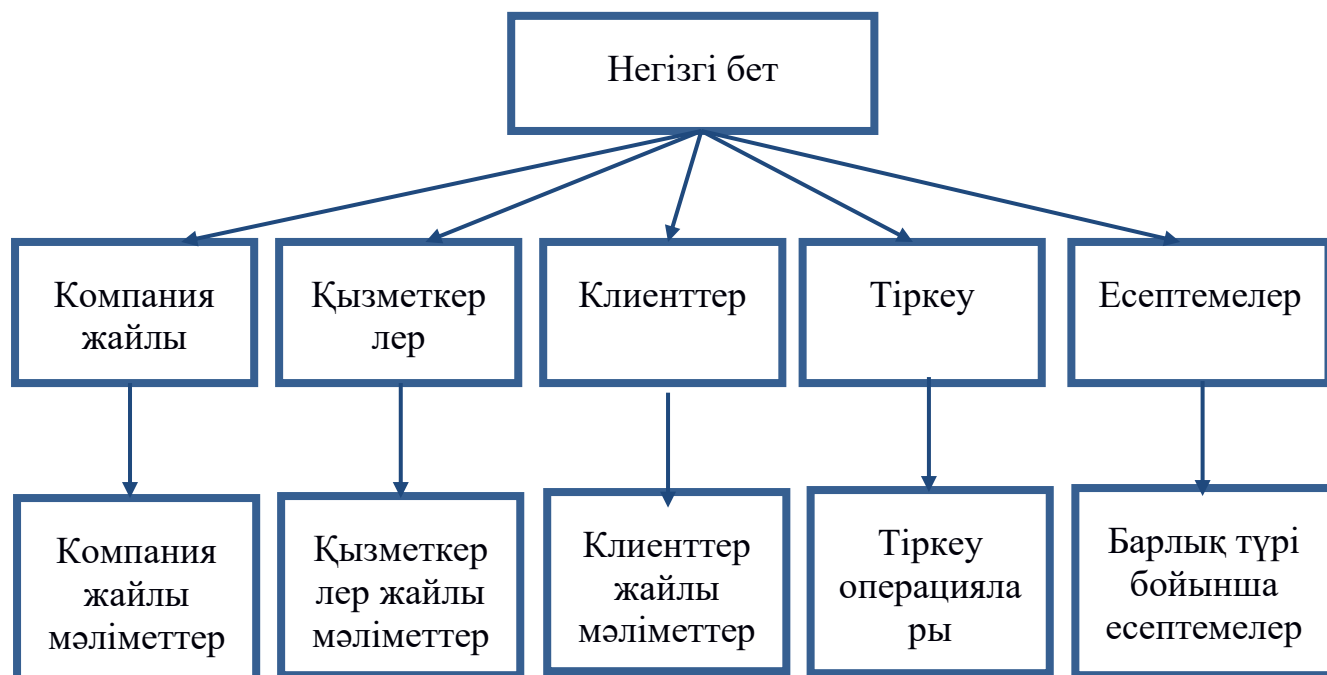
2.1-сурет – Қосымшаның негізгі бетінің сұлбасы

«Әрекеттер» жүйесінің функционалды сұлбасы келесі 2.2-суретте келтірілген.

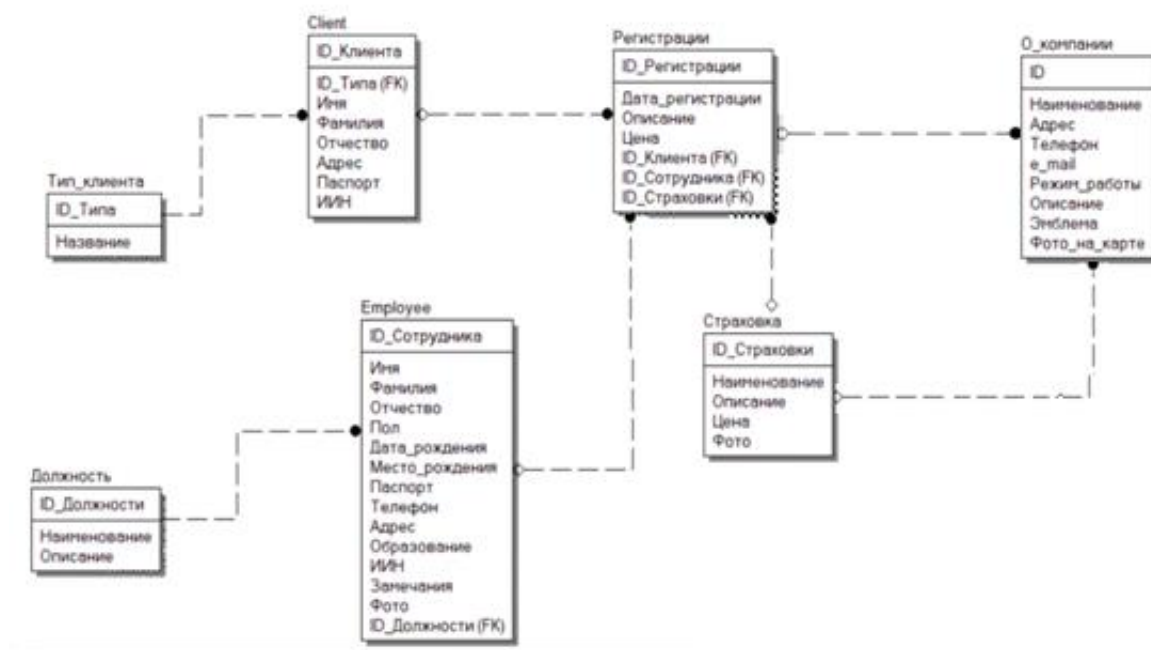


2.2-сурет – Қосымшаның әрекеттер бетінің сұлбасы

Сақтандыру компаниясының ақпараттық қосымшасын құрудың функционалды сұлбасы келесі 2.3-суретте келтірілген.



2.3-сурет – Сақтандыру компаниясының ақпараттық қосымшасын құрудың функционалды сұлбасы



2.4-сурет – Сақтандыру компаниясының ақпараттық қосымшасын құрудың кестелер байланысының сұлбасы

2.3 Программалау тілін таңдауды негіздеу

2.3.1 Visual Studio 2017 ортасын таңдауды негіздеу

Бүгінгі таңда Visual Studio 2017 құралдарына:

- даярлау ортасы;
- бірлескен жұмысты ұйымдастыруға арналған сервис - Visual Studio Team Services;
- оның on-premise нұсқасы, TFS ретінде белгілі;
- мобильді қосымшаларды әзірлеудің толыққанды циклін іске асыру үшін кешенді шешім — Visual Studio Mobile Center;
- Visual Studio Code кодының кроссплатформалы редакторы (Mac, Linux және Windows үшін қол жетімді) - ол жақында 1.10 нұсқасына дейін жаңартылды;
- Visual Studio for Mac-да, Microsoft өзінің IDE және macOS нұсқасын шығарды.

Microsoft құралдарының әрбір нұсқасымен әзірлеушілердің жағдайын ескеруге және кез келген платформаның астында қосымшаларды жасау үшін ыңғайлы кезекті нұсқаны жасауға тырысады.

Біріншіден, қазір кез-келген әзірлеуші Visual Studio 2017 толық нұсқасын жүктей алады және C#-та кроссплатформалы мобильді қосымшаларды құру туралы Xamarin University-оқыту сервисіне кіру үшін 60 күндік тегін жазылым ала алады.

Екіншіден, жасаушылар тек кодты жазуға зейінделу үшін барлық жағдайды жасай отырып, әзірлеушілердің өнімділігін арттыруға қамқорлық жасауды жалғастыруда. Мысалы, код бойынша навигацияның, рефакторингтің, түзетулердің және баптаудың тиімді мүмкіндіктерінде барлық қолдау көрсетілетін тілдер үшін жақсартулар болды. Жаңа нұсқада модульдік тестілеу және тәуелділіктерді тексеру жолында, жаңа real-time функцияларымен командалық әзірлеу және тиімділігін арттыру.

Үшінші маңызды өзгеріс құралды орнату процесіне әсер етті. Жаңа, жеңілдетілген модульдік тәсіл сізге қажетті ортаның компоненттерін орнатуға мүмкіндік береді және сол арқылы құралды орнатуды тездетеді. Сонымен қатар, енді әзірлеушілердің кез келген қажетті код фрагментін қалпына келтіру үшін жобалар мен шешімдер жасаудың қажеті жоқ.

Visual Studio соңғы презентациялары Azure бұлттық платформасы сервистерімен интеграцияны жақсартуды көрсетпей кете алмады. Microsoft әзірлемелері Ide-ден Azure бұлттарындағы қолданбаларыңызды жасауды, баптауды, орналастыруды және жариялауды жеңілдетуге мүмкіндік береді, осы қосымшалармен, сондай-ақ Docker-контейнерлермен,. NET Core қосымшаларымен және т. б. жұмыс істеуге арналған құрал-саймандарды ұсынады.

Ұялы өңдеу саласында басқа да маңызды өзгерістер. Әзірлеушілер модульдік тесттерді баптау, пішіндеу және генерациялау үшін жақсартылған

құралдарды алды. Егер сіз кроссплатформалы қосымшаны жасауды жоспарласаңыз, онда Visual Studio 2017 және Xamarin немесе Apache Cordova, тіпті Visual C++ қолдануға болады, бірақ сол құрал – Visual Studio 2017 шеңберінде кроссплатформалы кітапханаларды құру үшін баламалы тәсілді пайдалану қажет.

Бүгінгі күні Visual Studio TeamFoundation Server 2017 Update 1 жаңа нұсқасы ресми түрде қол жетімді болды. Командалардың бірлескен жұмысын ұйымдастыруға арналған on-premise платформасының бұл шығарылымына бұрыннан күтілген мүмкіндіктер енгізілді: мысалы, жаңа үрдістер үлгілері, репозиторийлерге, pull-реквестерге қатынауды жақсартылған басқару және т.с.с.

Жаңа мүмкіндіктер Visual Studio Mobile Center (Preview) қызметіне де қосылды. Visual Studio Mobile Center – бұл мобильді қосымшаларды құрастыру, тестілеу, тарату және мониторинг бойынша кешенді шешімді ұсыну үшін құрылған мобильді әзірлеушілерге арналған жаңа сервис. Бүгін Mobile Center жасаушылары Swift, ObjectiveC және Java-да, Xamarin және React Native қосымшаларымен бірге жазылған қосымшаларды қолдау туралы жариялады. Сондай-ақ Distribution Groups құру мүмкіндіктері, Espresso қолдауы және жақсартылған аналитика пайда болды.

Бүгін Microsoft-тың Mac-дағы бірінші IDE шығарылымы төртінші preview жарияланды. Қазіргі уақытта бұлт шешімдері мен mac OS негізіндегі қосымшаларды жасау, мобильді әзірлемелерге негізделген. Соңғы жылдары .NET Core жобаларының қолдауы, NuGet және мобильді әзірлеушінің құралдарын үнемі жақсартып отыру қосылды. Сондай-ақ, команда үнемі түзетулермен және өнімділікті оңтайландырумен жұмыс істейді.

Сондай-ақ, Enterprise әзірлеушілер өз ұйымының ішінде DevOps-циклін оңтайландыру және құру үшін қосымша сервистерге қол жеткізеді: мұндай бұлтты CI-сервер, бұлттағы жүктеме тестілеу және тіпті жеке DevOps-коучинг құралдары болып табылады. Бұл Visual Studio порталында қол жетімді корпоративтік жазылушылар үшін жаңа мүмкіндіктерді есептемегендегі талдау болып саналады.

Барлық тіркелген қатысушыларға жаңа қызметтер ұсынатын Visual Studio Dev Essentials тегін бағдарламасы да өз дамуын жалғастыруда. Бүгін Opsgility платформасының тренингтеріне 3 айлық жазылу жарияланды, ол DevOps бойынша курстардың толық каталогына және бұлтты шешімдерге қол жеткізуге мүмкіндік береді [5].

2.3.2 C# программалау тілін таңдауды негіздеу

Әлемдегі миллиондаған программистер C#-та жұмыс жасайды және олардың көбісі C#-та мәліметтерді өңдеуге және сақтауға арналған программалар құрады.

Күнбе-күн коммерциялық және жеке мәліметтерді санамағанда (адам

аттары, мекен-жайлар, есепшоттар) визуальды интерфейс көмегімен жұмыс істеу керек болатын көптеген ақпараттар бар: дыбыс файлдары, видео көріністер, WEB–парақшалары және т.б. Осындай көп ақпарат арасында өзімізге керекті мәліметтерді алу үшін, оларды өңдейтін программалар қажет.

Программа құруға ыңғайлы тамаша ортаның бірі – C# ортасы таңдап алынды. Оны оқып үйренуге, түсінуге және логикалық ойлауға өте қарапайым әрі жеңіл. C# программалаудың құрылымдық тәртібін және программалаудың басқа тілдерге қарағанда жалпы жақсы дамыған. Оның қызықты рекурсивті мүмкіндіктері, сонымен қатар қызықты есептер шығару мүмкіндіктері бар.

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

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

C# ортасы – бағдарламаушының жоғары әрі тиімді жұмысын қамтамасыз ететін күрделі механизм. Ол визуалды түрде бір уақытта бірнеше ашылған терезелермен жұмыс атқарады. Терезелер экранда толық немесе бөлшектелген түрде бір- бірін жаба отырып жылжи алады, бұл Word текстік 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 программа дайындау, программа 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 интерфейсті тез құру. Интерфейс Windows-тың барлық талаптарын қанағаттандырады және қолданушы компьютерінде орнатылған жүйеге автоматты түрде ыңғайланады. Өйткені Windows-тың көптеген функцияларын, кітапханаларын қолданады;

- программалаудың 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 Программалық қамтаманы құру

3.1 Программалық қамтаманың құрылымы

Программа сақтандыруға келген клиенттерге сақтандырудың әр түрі бойынша сақтандыру шарттарымен таныстыру, клиенттерді тіркеу, оларға жылдам қызмет көрсету, сақтандыру түрін қайта таңдағысы келген жағдайда орта және шағын бизнес-жобаларға қызмет көрсету аспектілерін теориялық және тәжірибелік түрде зерттеу болып табылады.

3.1-суретте Сақтандыру компаниясының ақпараттық қосымшасын құрудың программалық қамтамасы келтірілген.



3.1-сурет – Сақтандыру компаниясының ақпараттық қосымшасын құрудың программалық қамтамасы

3.2 Программаның баяндалуы

3.2.1 Жалпы мағлұматтар

Сақтандыру компаниясының ақпараттық қосымшасын құру C# программалау тілі және SQLiteStudio құрылымды сұраныстар тілі арқылы жазылған.

C# программалау тілі көп терезелік жүйе болып табылады. Қолданушылық интерфейс программалаушымен байланысты құруды қамтамасыз етеді және әртүрлі қолданбалы басқару элементтері бар бірнеше терезелерден тұрады. Интерфейстік жүйенің түрі әрбір қолданушының өзіндік өзгертулеріне байланысты әртүрлі болады. Құрылатын ақпараттық жүйе қызметкерлердің жұмысын жеңілдетеді, яғни:

- қажетті ақпаратты жылдам және тез алуға көмектеседі;
- мәліметтерді енгізуді жеңілдетеді;
- қолданушыға кез келген параметр бойынша ақпаратты табады.

3.2.2 Функционалдық тағайындалуы

Мәліметтер базасының көмегімен қолданушы мәліметтерге қатынаудың жолын қысқартады. Осы мүмкіндікті тиімді пайдалану үшін алдымен кестемізді дұрыс құруымыз керек. Мәліметтер базасы банктің кредиттік бөлімдерінде қолданылады.

3.2.3 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.

3.2.4 Қажетті техникалық жабдықтар

Сақтандыру компаниясының ақпараттық қосымшасын құру үшін Intel(R) Celeron(R) CPU 2.13 GHz / RAM 1.99 Gb / HDD 160 Gb/ VideoCard 256 Mb / Asus F80L ноутбугын қолданылды.

Сонымен қатар HP Laser-jet 1010 принтерін қолдандым.

3.2.5 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.

3.2.6 Кіріс мәліметтер

Кіріс мәліметтері ретінде қызметкерлер туралы мәліметтер қорының ақпараттарын енгізу, клиенттер жайлы, сақтандыру түрлері, келісім-шарттар түрлері, оларды таңдау, толтыру тәріздімәліметтерді базаға енгізу.

3.2.7 Шығыс мәліметтер

Шығыс мәліметтерге қолданушы туралы енгізілген мәліметтер бойынша сұраныстар қабылданған жағдайда мәліметтер бойынша баспаға шығару. Клиент өзі қалаған сақтандыру түрі бойынша келісім-шартқа отырғаннан кейін, оны баспаға жіберуге болады. Сонымен қатар қызметкерлер жайлы мәліметтерді де есеп беру арқылы шығарып алуға болады.

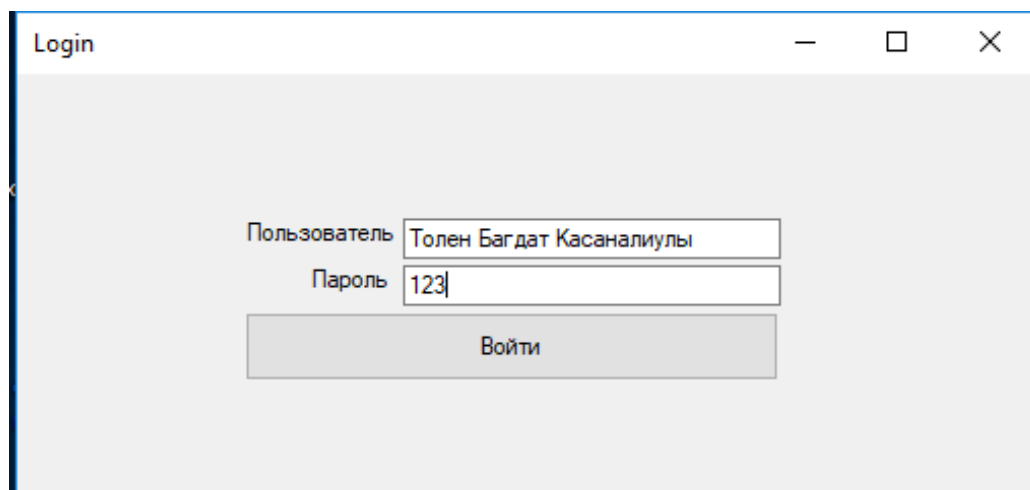
3.3 Жүйенің жұмыс жасау барысын сипаттау

Құрылып отырған жұмыс Visual Studio ортасында программалау ортасында құрастырылған. C# программасын Borland Softwar Corporation құрастырып жасаған.

Құрылған мәліметтер базасының дұрыстығын тексеру үшін F9 пернелері арқылы жүктейміз.

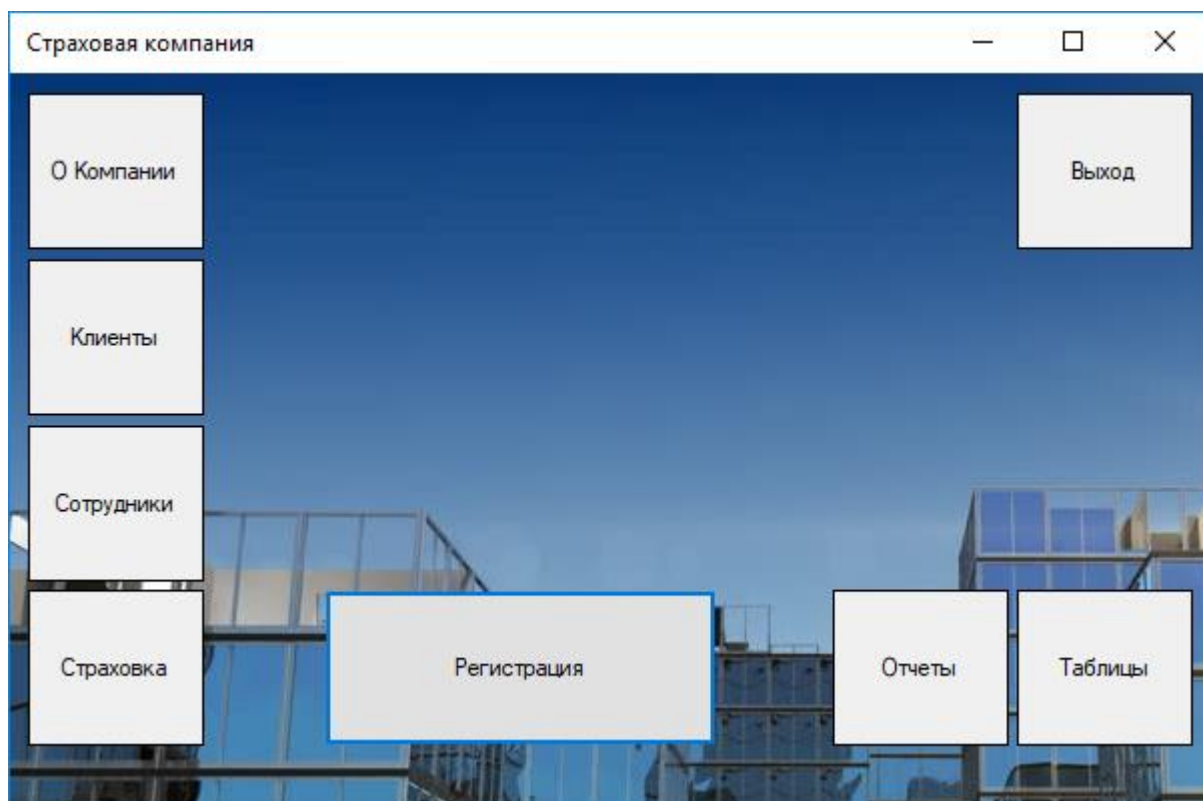
Бағдарламаны іске қосу – RUN батырмасы арқылы жүзеге асады. Бағдарламаны іске қосқанда «Негізгі меню» терезесі ашылады (3.2-сурет).

Негізгі меню терезесіненбазаға қызметкер өз фамилиясын таңдап, паролі арқылы кіреді. Егер пароль дұрыс емес терілген болса, томендегідей қате шығарады (3.2-сурет).



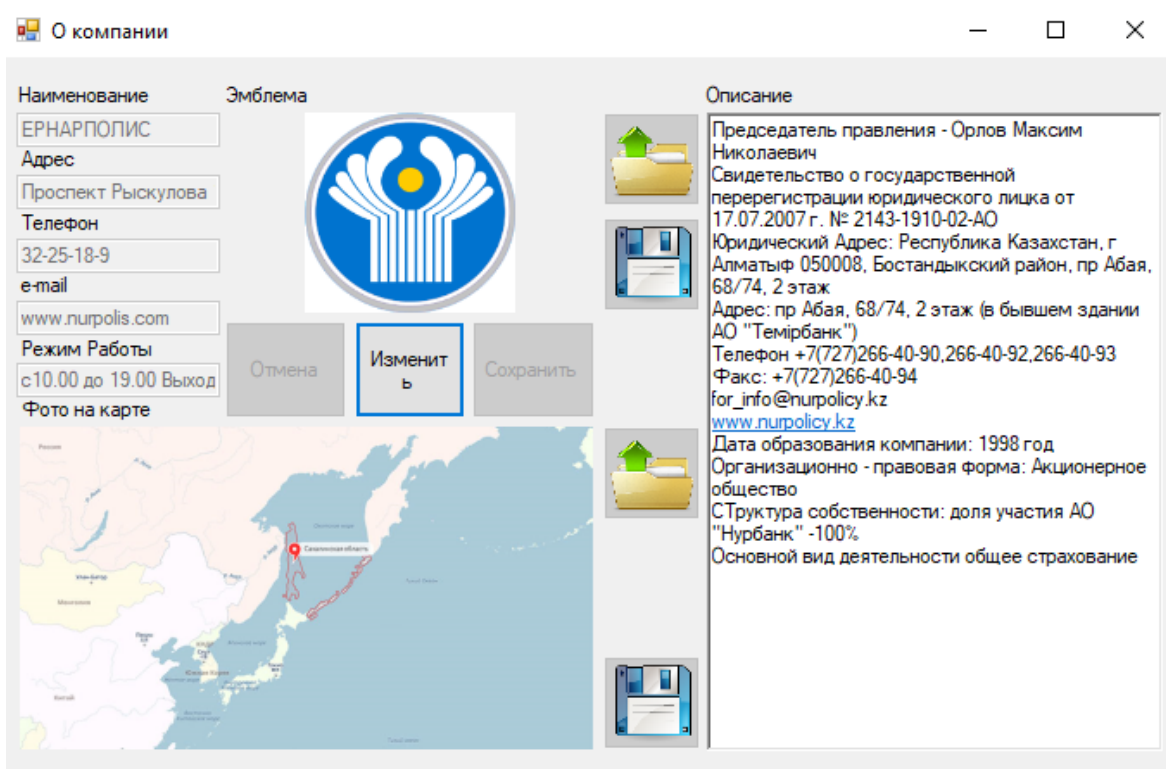
3.2-сурет - Жүйеге ену диалог терезесі

Парольды тергеннен кейін «Войти» батырмасын басамыз. Парольды дұрыс енгізгеннен кейін ғана базаға кіруге мүмкіндік аламыз(3.3-сурет).



3.3-сурет –Негізгі бет

Келесі 3.4-суретте сақтандыру компаниясы жайлы толық ақпарат алуға болады.



3.4-сурет – Компания жайлы мәліметтерді шығару терезесі

Келесі 3.5-суретте қызметкерлердің аты-жөні, мекен-жайы, білімі және тағы басқа толық ақпарат алуға болады.

The screenshot shows a window titled 'Сотрудник' (Employee) with a tabbed interface. The 'Личная информация' (Personal information) tab is active, displaying a photo of a man and a form with the following fields:

Фамилия	Толен
Имя	Багдат
Отчество	Касаналиулы
Пол	Мужской
Дата рождения	1994
Место рождения	Алматы
Паспорт	968523144
Телефон	77472110483
Адрес	Саина 45
Образование	Высшее
ИИН	960804125896

Below the form is a 'Замечания' (Remarks) field with the text 'Лучший работник января' (Best worker of January). At the bottom are buttons: 'Сформировать личную карточку' (Generate personal card), 'Список сотрудников' (List of employees), 'Добавить нового' (Add new), 'Добавить сотрудник' (Add employee), 'Изменить' (Edit), 'Сохранить изменения' (Save changes), 'Обновить список' (Update list), and 'Отмена' (Cancel).

On the right, there is a 'Навигация' (Navigation) table with columns: ID Сотрудника, Фамилия, Имя, Отчество, Пол, and Др. The table contains 5 rows of employee data.

ID Сотрудника	Фамилия	Имя	Отчество	Пол	Др.
1	Багдат	Толен	Касаналиулы	Мужской	19
2	Ерсин	Галымжанов	Ерболулы	Мужской	19
3	Саяжан	Сериктай	Айбеккызы	Женский	19
4	Тилеукабыл	Дидар	Даулетулы	Мужской	19
5	Кайсар	Касенов	Кайратулы	Мужской	19

3.5-сурет –Қызметкерлер терезесі

Келесі 3.6-суретте жаңа келген қызметкерлерге немесе парольдарын ұмытып қалған қызметкерлерге жаңа пароль беру терезесі келтірілген.

The screenshot shows the same 'Сотрудник' (Employee) window, but the 'Управление паролем' (Manage password) tab is active. It displays a 'Редактирование пароля' (Edit password) form with the following fields:

Старый пароль	123
Новый пароль	Axz12356

Below the form is an 'Ok' button. The 'Навигация' (Navigation) table on the right is the same as in the previous screenshot.

3.6-сурет –Қызметкерлерге жаңа пароль беру терезесі

Келесі 3.7-суретте жаңа келген қызметкерлердің қызметін бекіту терезесі келтірілген. Мұнда сонымен қатар қызметкерлер жайлы ақпаратты да көруге болады.

Сотрудник

Личная информация | Должность | Дата приема | Управление паролем

ID Сотрудника: 1

Навигация

ID Сотрудника	Фамилия	Имя	Отчество	Пол	Др
1	Багдат	Толен	Касаналиулы	Мужской	19
2	Ерсин	Галымжанов	Ерболулы	Мужской	19
3	Саяжан	Сериктай	Айбеккызы	Женский	19
4	123	Дидар	123	123	12
5	Кайсар	Касенов	Кайратулы	Мужской	19

Профессия
1:Первая ранг

Работает с вилп клиентами

Сформировать личную карточку

Список сотрудников

Добавить нового

Добавить сотрудник

Изменить

Сохранить изменения

Обновить список

Отмена

3.7-сурет – Қызметкерлердің қызметі жайлы мәліметтерді енгізу терезесі

Келесі 3.8-суретте жаңа клиентті енгізу, өзгерту және қалыптасқан ақпарат бойынша жалпы карточка шығару терезесі келтірілген.

Клиент

	ID_Клиента	Фамилия	Адрес	Паспорт
	1	Толен	Тынышбаева 59	543778
▶	2	Тилеукабыл	Саина 20	458596
	3	Уалиханов	Казыбек би	968574
	5	Абдулла	Абая 30	784565
	6	Тарыбаев	Жоракова 25	458535

Добавить Клиента Добавить Обновить таблицу Изменить Сохранить изменения Отменить

Фамилия: Тилеукабыл
 Имя: Дидар
 Отчество: Даулетулы
 Тип клиента: 2:Второй тип

Документы Дата регистраций
 Адрес: Саина 20
 Паспорт: 458596
 ИИН: 987452654789

Отчет по клиентам Распечатать карточку

3.8-сурет – Клиенттердің мәліметтерін көру терезесі

Келесі 3.9-суретте қызметкерлер жайлы ақпаратты Excel-де есеп беру ретінде шығарып алу терезесі келтірілген.

Excel-де есеп беру терезесі

	A1	ID_Сотрудника
1	ID_Сотру	2
2	Имя	Ерсин
3	Фамилия	Галымжанов
4	Отчество	Ерболулы
5	Пол	Мужской
6	Дата_рож	1997
7	Место_ро	Тараз
8	Паспорт	96524585
9	Телефон	77084568523
10	Адрес	Абая15
11	Образова	Высшее
12	ИИН	9,70105E+11
13	Замечани	

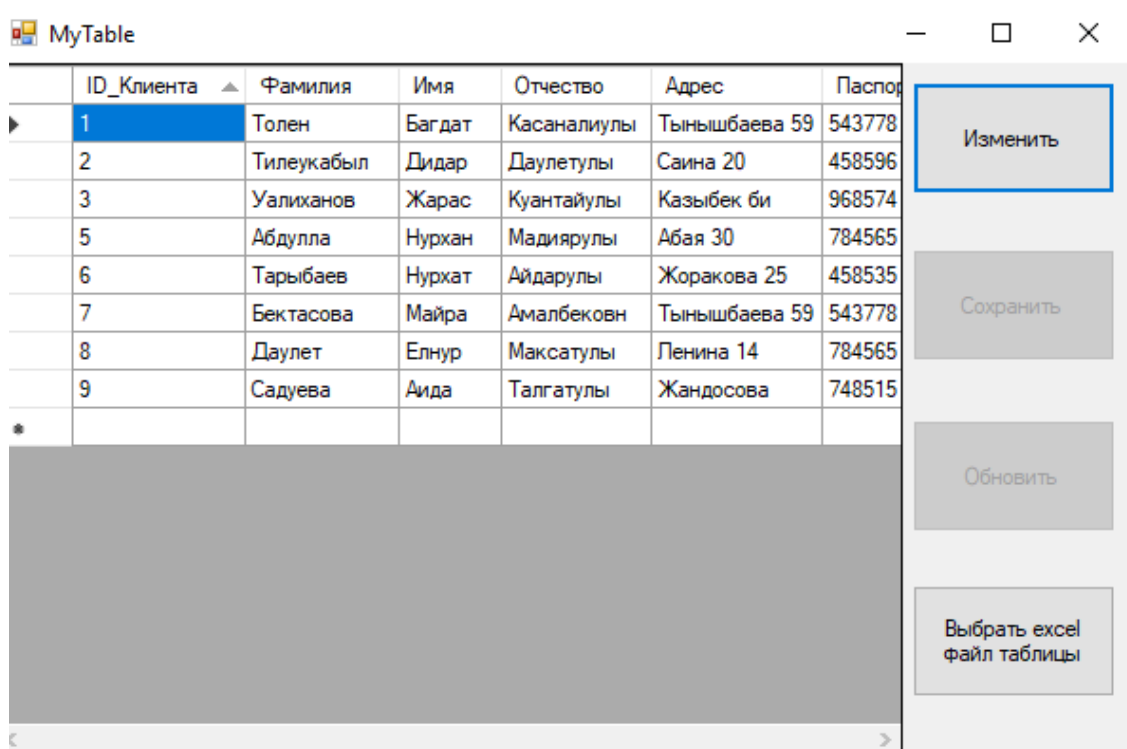
3.9-сурет –Қызметкерлер жайлы мәліметтерді шығару терезесі

Страховканың түрі, реттік нөмірі және бағасы бойынша қарау және оны шығарып алу терезесі 3.10-суретте көрсетілген.

ID_Товара	Наименование	Цена
1	Автострахование	15000
2	Медицинское	10000
3	Банковское	12000
4	Пенсионные	5000
5	Недвижимость	40000
6	Жизнь	30000
7	Финансовое	30000
8	Заболевание	20000
9	Грузовой	10000
10	Заем	30000
11	Ситуаций	30000
*		

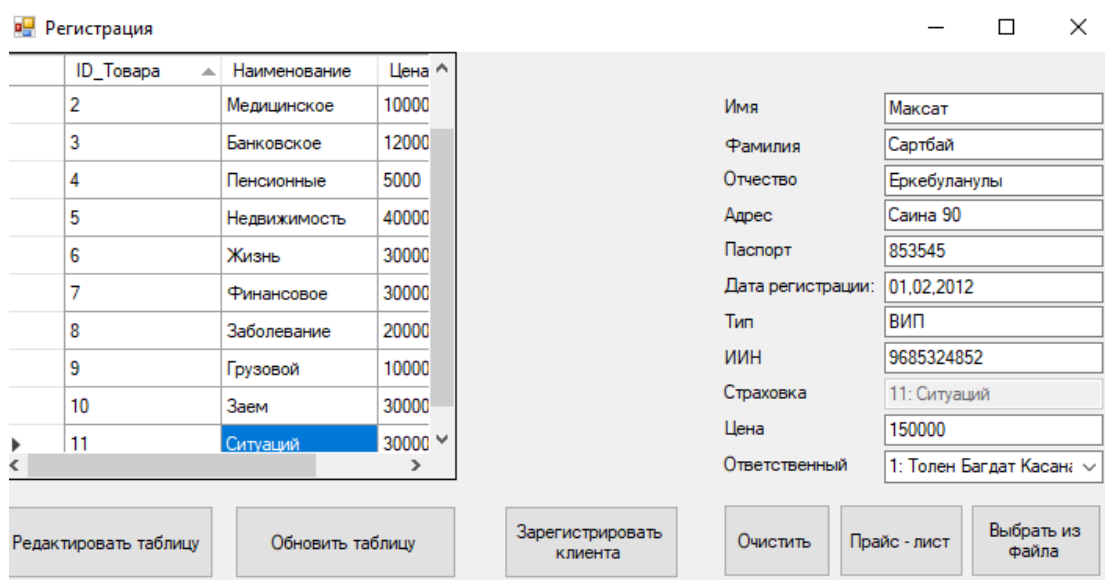
3.10-сурет – Страховка терезесі

Келесі 3.11-суретте клиенттерге берілетін келісім-шарт терезесі келтірілген. Мұнда клиенттің нөмірі, аты-жөні, адресі және қосымша мәліметтер көруге болады.



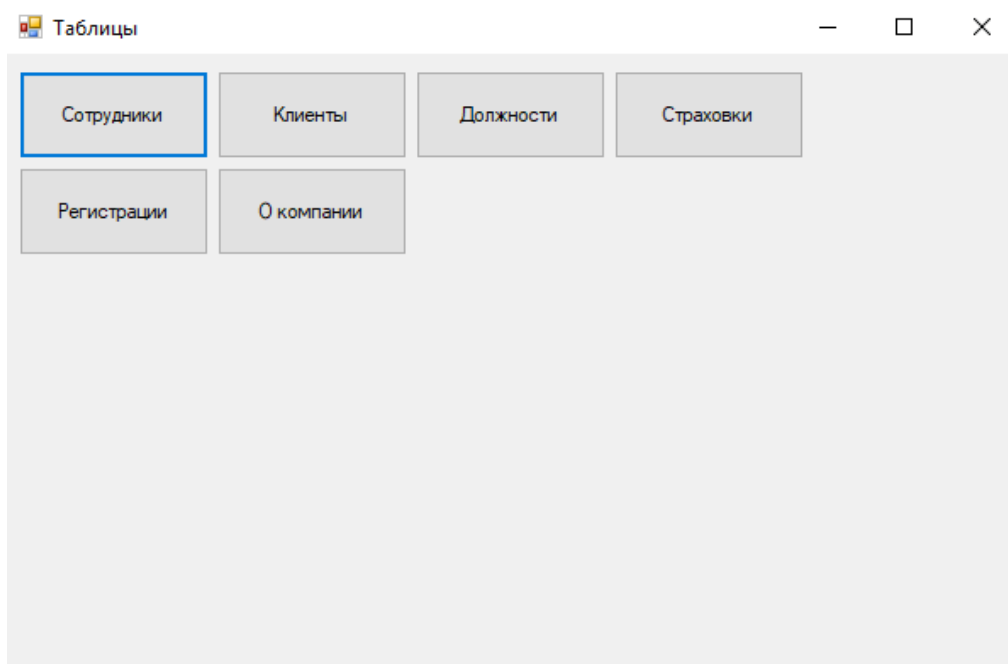
3.11-сурет– Клиенттерге берілетін келісім-шарт терезесі

Төменде келтірілген 3.12-суретте клиенттерді тіркеу, олардың таңдаған сақтандыру түрін шығару және прайс-листті шығару қарастырылған.



3.12-сурет – Клиенттерді тіркеу терезесі

Келесі 3.13-суретте мәліметтер базасы бойынша құрылған барлық модульдер келтірілген. Қызметкерлер, клиенттер, қызметкерлер қызметі, страховка түрі, клиентті тіркеу және компания жайлы ақпаратты қарауға болады.



3.13-сурет –Белгілі бір уақыт аралығына шығаруға болатын статистика

ҚОРЫТЫНДЫ

Заман талабына сай қазіргі кезде әрбір мекеме, жұмыс орындарының ақпараттық жүйесін құру кең тараған. Құрылып отырған дипломдық жұмыста сақтандыру компаниясының ақпараттық жүйесінің программасы құрылды. Бұл ақпараттық жүйенің көмегімен сақтандыру компаниясы қызметкерлерінің жұмысын жеңілдетуге болады. Базаға қызметкерлер өз парольдері арқылы кіріп, өзгертулер енгізе алады.

Дипломдық жұмыс тақырыбының маңыздылығы мен өзектілігі елімізде сақтандыру компанияларының қызмет көрсету сапасының деңгейін жоғарылатып, өзге де дамыған мемлекеттерде қолданылатын сақтандыру қызметтерінің санын арттыру және клиенттерге сапалы сақтандыру түрлерін ұсыну қызметінің ел экономикасында маңызды орын алуында болып тұр.

Осы дипломдық жұмысты жасау барысында құрылған программа сақтандыру қызметін пайдалануға тілек білдірген клиенттерді тіркеу, құжаттандыру жұмыстарын атқаратын бөлім қызметкерлерінің жұмысын автоматтандыруға бағытталған қосымша болып табылады. Бұл программа оператордың жұмысын жеңілдетіп қана қоймай, оның уақытын ұтымды пайдалануға да көмектеседі. Ол операторға қажетті мәліметтерді жүйелеп сақтап, оны керек кезінде іздеп табуды, соның ішінде клиенттерді тіркеу, қайсыбір қажетті мәліметтерді электронды іздеу жүйесі, құжаттар бойынша іздеу, қайсыбір құжаттар бойынша есеп берулерді қағазға шығаруды ұйымдастыруды жеңілдетеді.

Жүйені қарапайым қолданушы қолданғанда ешқандай қиындықтар мен кедергілерге тап келмеуі үшін қарапайым интерфейс жасалған.

Дипломдық жұмысты құру барысында келесі есептер қарастырылды:

- сұраныстарды қабылдау және сақтандыру қызметтерінің жұмысы зерттелді;
- клиенттерді қабылдау және тіркеу технологиялық үрдісінің сұлбасы құрылды;
- клиенттерге қосымша қызмет түрлерін таныстыру және оларды жұмыс аясына енгізу;
- көрсетілетін қосымша қызметтерге кез келген уақытта тапсырыс беру жүйесін құру;
- сақталып тұрған тізім ішінен керекті клиентті жеке мәліметтері бойынша іздеу жүйесі арқылы тез тауып алу;
- сақталатын ақпарат қауіпсіздігін қамтамасыз ету, жеке аккаунт құру және мәліметтерге рұқсат беруге белгілі бір құқықтарды тағайындау;
- клиент жайлы енгізілген мәліметтер негізінде екіжақты сақтандырудың келісім-шартын құру;
- міндетті сақтандыру мен ерікті сақтандыру түрлерінің күрделі нарядтарынан ағұрлым нақты, жылдам және сапалы өңдеу;
- бұрын қызмет көрсетілген және қызмет көрсетіліп жатқан клиенттер жайлы ақпаратты мұрағатқа сақтау.

Құрылып отырған қосымшаның басқа жұмыстардан артықшылығы – бұл қосымша қолданушыға қарапайым және қолдануға ыңғайлы интерфейс арқылы жасалды.

Программалық кешен Visual Studio ортасында жасалып, мәліметтер базасын басқару жүйесі ретінде SQLiteStudio таңдалды. Жасалған дипломдық жұмысты жобалау үрдісінде алға қойылған мақсаттардың бәріне қол жеткізілді және бұл жүйе сақтандыру компаниясы қызметкерлерінің жұмысын айтарлықтай жеңілдетеді деген үміттемін.

ПАЙДАЛАНЫЛҒАН ӘДЕБИЕТ

- 1 Назарбаев Н.А. Казахстан - 2030. Процветание, безопасность и улучшение благосостояния всех казахстанцев. Послание Президента страны народу Казахстана. - Казахстанская Правда, 11 октября 1997г
- 2 Қазақстан Республикасының Конституциясы. Алматы, 2003
- 3 К.Қ.Жүйріков “Қазақстандағы сақтандыру: күрделі мәселелер мен дамудың болашағы”. Алматы, 1994
- 4 Дэвид Б. “Страхование: принципы и практика”
- 5 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.
- 6 Васильков А.В. Информационные системы и их безопасность: Учебное пособие / А.В. Васильков, А.А. Васильков, И.А. Васильков. - М.: Форум, 2013. - 528 с.. Мамаев. Microsoft SQL Server 2000. Наиболее полное руководство. – СПб.: БХВ-Петербург, 2005.
- 7 Мезенцев К.Н. Автоматизированные информационные системы: Учебник для студентов учреждений среднего профессионального образования / К.Н. Мезенцев. - М.: ИЦ Академия, 2013. - 176 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 Кузин А.В. Базы данных: Учебное пособие для студ. высш. учеб. заведений / А.В. Кузин, С.В. Левонисова. - М.: ИЦ Академия, 2012. - 320 с.
- 11 Пирогов В.Ю. Информационные системы и базы данных: организация и проектирование: Учебное пособие / В.Ю. Пирогов. - СПб.: БХВ-Петербург, 2009. - 528 с.
- 12 Советов Б.Я. Базы данных: теория и практика: Учебник для бакалавров / Б.Я. Советов, В.В. Цехановский, В.Д. Чертовской. - М.: Юрайт, 2013. - 463 с.
- 13 <http://www.ubukulov.kz/sql/sql-degenimiz-ne.html>
- 14 В.В. Кириллов. Структурированный язык запросов (SQL) –СПб.: ИТМО, 1994.
- 15 Робинсон С.; Корнес, О.; Глинн, Д. и др. С# для профессионалов; М.: Лори, 2005. - 396 с.
- 16 Саттер Герб Новые сложные задачи на С#; Вильямс, 2005. -272 с.
- 17 Холзнер С. Программирование на С#. Учебный курс; Питер, 2001. - 576 с.
- 18 <https://tsgarant.kz/>
- 19 <https://www.nurpolicy.kz>

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 Страховая компания
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
        public static Dictionary<string, MyTable> tableForms = new Dictionary<string,
MyTable>();
        private Form aboutForm;
        private Form empForm;

        private string[] empColumnNames = new string[]
        {
            "ID_Сотрудника", "Имя", "Фамилия", "Отчество", "Пол", "Дата_рождения", "Место_рождения",
            "Паспорт", "Телефон", "Адрес", "Образование", "ИИН", "Замечания"
        };
        private void button2_Click(object sender, EventArgs e)
        {
            if (aboutForm != null)
            {
                if (aboutForm.IsDisposed)
                {
                    aboutForm = new Form2();
                    aboutForm.Show();
                }
            }
            else
            {
                aboutForm = new Form2();
                aboutForm.Show();
            }
        }
        private void button3_Click(object sender, EventArgs e)
        {
            if (empForm != null)
            {
                if (empForm.IsDisposed)
                {

```

Қосымша А жалғасы

```
empForm = new Form3();
empForm.Show();
}
}
else
{
empForm = new Form3();
empForm.Show();
}
}

private void Form1_Load(object sender, EventArgs e)
{
}
private void Form1_Closed(object sender, System.EventArgs e)
{
Application.Exit();
}
Form8 form8;

private void button6_Click(object sender, EventArgs e)
{
if (form8 != null)
{
if (form8.IsDisposed)
{
form8 = new Form8();
form8.Show();
}
}
else
{
form8 = new Form8();
form8.Show();
}
}
void OpenTableForm(string tableName,string[] ColumnNames)
{
if (tableForms.ContainsKey(tableName))
{
if (tableForms[tableName].IsDisposed)
{
tableForms[tableName] = new MyTable();
tableForms[tableName].Initiate(tableName,ColumnNames);
tableForms[tableName].Show();
}
}
else
{
tableForms.Add(tableName, new MyTable());
}
```


Қосымша А жалғасы

```
        tableForms[tableName].Initiate(tableName, ColumnNames);
        tableForms[tableName].Show();
    }
}
private Form cliForm;
private void button4_Click(object sender, EventArgs e)
{
    if (cliForm != null)
    {
        if (cliForm.IsDisposed)
        {
            cliForm = new Form4();
            cliForm.Show();
        }
    }
    else
    {
        cliForm = new Form4();
        cliForm.Show();
    }
}
Form regForm;
private void button1_Click(object sender, EventArgs e)
{
    if (regForm != null)
    {
        if (regForm.IsDisposed)
        {
            regForm = new Form7();
            regForm.Show();
        }
    }
    else
    {
        regForm = new Form7();
        regForm.Show();
    }
}

private void button7_Click(object sender, EventArgs e)
{
    Application.Exit();
}
Form6 strah;
private void button5_Click(object sender, EventArgs e)
{
    if (strah != null)
    {
        if (strah.IsDisposed)
        {
```

Қосымша А жалғасы

```
        strah = new Form6();
        strah.Show();
    }
}
else
{
    strah = new Form6();
    strah.Show();
}
}
Form9 otch;
private void button8_Click(object sender, EventArgs e)
{
    if (otch != null)
    {
        if (otch.IsDisposed)
        {
            otch = new Form9();
            otch.Show();
        }
    }
    else
    {
        otch = new Form9();
        otch.Show();
    }
}
```

Form2.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 System.IO;
using System.Data.SQLite;
using System.Drawing.Imaging;

namespace BЧК
{
    public partial class Form2 : Form
    {
        int count = 0;
        SQLiteConnection conn;
        string dataName = "Company";
        string tableName = "О_компаний";
    }
}
```

Қосымша А жалғасы

```
List<string> columnNames = new List<string>()
{
    "ID", "Наименование", "Адрес", "Телефон", "е_mail", "Режим_работы", "Описание"
};
public Form2()
{
    InitializeComponent();
}
private void Form2_Load(object sender, EventArgs e)
{
    UpdateData();
    HideElements(false);
}
private void button1_Click(object sender, EventArgs e)
{
    HideElements(true);
}
private void button2_Click(object sender, EventArgs e)
{
    using (SQLiteConnection connection = new SQLiteConnection("Data Source=" +
        dataName + ".db;"))
    {
        string sqlExpression = "UPDATE " + tableName + " SET
Наименование=@Наименование,Адрес=@Адрес,Телефон=@Телефон,е_mail=@е_mail,Ре
жим_работы=@Режим_работы,Описание=@Описание,Эмблема=@Эмблема,Фото_на_кар
те=@Фото_на_карте where ID=1";
        SQLiteCommand command = new SQLiteCommand(sqlExpression, connection);
        command.Parameters.AddWithValue("@Наименование", textBox1.Text);
        command.Parameters.AddWithValue("@Адрес", textBox2.Text);
        command.Parameters.AddWithValue("@Телефон", textBox3.Text);
        command.Parameters.AddWithValue("@е_mail", textBox4.Text);
        command.Parameters.AddWithValue("@Режим_работы", textBox5.Text);
        command.Parameters.AddWithValue("@Описание", richTextBox1.Text);
        command.Parameters.AddWithValue("@Эмблема",
ImageToByte(pictureBox2.Image, ImageFormat.Png));
        command.Parameters.AddWithValue("@Фото_на_карте",
ImageToByte(pictureBox1.Image, ImageFormat.Png));
        connection.Open();
        command.ExecuteNonQuery();
        connection.Close();
    }
    UpdateData();
    HideElements(false);
}
private void button3_Click(object sender, EventArgs e)
{
    HideElements(false);
}
private void Form2_Closing(object sender, System.ComponentModel.CancelEventArgs e)
```

Қосымша А жалғасы

```
{  
  
}  
private void button6_Click(object sender, EventArgs e)  
{  
    using (OpenFileDialog openFileDialog = new OpenFileDialog())  
    {  
        openFileDialog.InitialDirectory = "c:\\";  
        openFileDialog.Filter =  
        "*.bmp|*.bmp|*.ico|*.ico|*.gif|*.gif|*.jpg|*.jpg|*.jpeg|*.jpeg|*.png|*.png| All files (*.*)|*.*";  
        openFileDialog.FilterIndex = 2;  
        openFileDialog.RestoreDirectory = true;  
  
        if (openFileDialog.ShowDialog() == DialogResult.OK)  
        {  
            pictureBox2.Image = Image.FromFile(openFileDialog.FileName);  
            ImageFormat thisFormat = GetImageFormat(openFileDialog.FileName);  
            SaveImage(ImageToByte(Image.FromFile(openFileDialog.FileName), thisFormat),  
            "Эмблема");  
        }  
    }  
}  
private void button4_Click(object sender, EventArgs e)  
{  
    using (OpenFileDialog openFileDialog = new OpenFileDialog())  
    {  
        openFileDialog.InitialDirectory = "c:\\";  
        openFileDialog.Filter =  
        "*.bmp|*.bmp|*.icon|*.icon|*.gif|*.gif|*.jpeg|*.jpeg|*.png|*.png| All files (*.*)|*.*";  
        openFileDialog.FilterIndex = 2;  
        openFileDialog.RestoreDirectory = true;  
        if (openFileDialog.ShowDialog() == DialogResult.OK)  
        {  
            ImageFormat thisFormat = GetImageFormat(openFileDialog.FileName);  
            pictureBox1.Image = Image.FromFile(openFileDialog.FileName);  
            SaveImage(ImageToByte(Image.FromFile(openFileDialog.FileName), thisFormat),  
            "Фото_на_карте");  
        }  
    }  
}  
  
private void button7_Click(object sender, EventArgs e)  
{  
    // Displays a SaveFileDialog so the user can save the Image  
    // assigned to Button2.  
    SaveFileDialog saveFileDialog1 = new SaveFileDialog();  
    saveFileDialog1.Filter =  
    "*.bmp|*.bmp|*.icon|*.icon|*.gif|*.gif|*.jpeg|*.jpeg|*.png|*.png";  
    saveFileDialog1.Title = "Сохранить изображение на диск";  
    saveFileDialog1.ShowDialog();  
}
```

Қосымша А жалғасы

```
// If the file name is not an empty string open it for saving.
if (saveFileDialog1.FileName != "")
{
    // Saves the Image via a FileStream created by the OpenFile method.
    System.IO.FileStream fs =
        (System.IO.FileStream)saveFileDialog1.OpenFile();
    // Saves the Image in the appropriate ImageFormat based upon the
    // File type selected in the dialog box.
    // NOTE that the FilterIndex property is one-based.
    switch (saveFileDialog1.FilterIndex)
    {
        case 1:
            pictureBox2.Image.Save(fs,
                System.Drawing.Imaging.ImageFormat.Bmp);
            break;
        case 2:
            pictureBox2.Image.Save(fs,
                System.Drawing.Imaging.ImageFormat.Icon);
            break;

        case 3:
            pictureBox2.Image.Save(fs,
                System.Drawing.Imaging.ImageFormat.Gif);
            break;
        case 4:
            pictureBox2.Image.Save(fs,
                System.Drawing.Imaging.ImageFormat.Gif);
            break;
        case 5:
            pictureBox2.Image.Save(fs,
                System.Drawing.Imaging.ImageFormat.Jpeg);
            break;
        case 6:
            pictureBox2.Image.Save(fs,
                System.Drawing.Imaging.ImageFormat.Png);
            break;
    }
    fs.Close();
}
}

private void button5_Click(object sender, EventArgs e)
{
    // Displays a SaveFileDialog so the user can save the Image
    // assigned to Button2.
    SaveFileDialog saveFileDialog1 = new SaveFileDialog();
    saveFileDialog1.Filter =
        "*.bmp|*.bmp|*.icon|*.icon|*.gif|*.gif|*.jpeg|*.jpeg|*.png|*.png";
    saveFileDialog1.Title = "Сохранить изображение на диск";
    saveFileDialog1.ShowDialog();
    // If the file name is not an empty string open it for saving.
```

Қосымша А жалғасы

```
        if (saveFileDialog1.FileName != "")
        {
            // Saves the Image via a FileStream created by the OpenFile method.
            System.IO.FileStream fs =
                (System.IO.FileStream)saveFileDialog1.OpenFile();
            // Saves the Image in the appropriate ImageFormat based upon the
            // File type selected in the dialog box.
            // NOTE that the FilterIndex property is one-based.
            switch (saveFileDialog1.FilterIndex)
            {
                case 1:
                    pictureBox1.Image.Save(fs,
                        System.Drawing.Imaging.ImageFormat.Bmp);
                    break;
                case 2:
                    pictureBox1.Image.Save(fs,
                        System.Drawing.Imaging.ImageFormat.Icon);
                    break;

                case 3:
                    pictureBox1.Image.Save(fs,
                        System.Drawing.Imaging.ImageFormat.Gif);
                    break;
                case 4:
                    pictureBox1.Image.Save(fs,
                        System.Drawing.Imaging.ImageFormat.Gif);
                    break;
                case 5:
                    pictureBox1.Image.Save(fs,
                        System.Drawing.Imaging.ImageFormat.Jpeg);
                    break;
                case 6:
                    pictureBox1.Image.Save(fs,
                        System.Drawing.Imaging.ImageFormat.Png);
                    break;
            }
            fs.Close();
        }
    }
    void UpdateData()
    {
        conn = new SQLiteConnection("Data Source=" + dataName + ".db;");
        conn.Open();
        SQLiteCommand comm = new SQLiteCommand("Select * From " + tableName + ";",
conn);
        using (SQLiteDataReader read = comm.ExecuteReader())
        {
            while (read.Read())
            {
                count++;
            }
        }
    }
}
```


Қосымша А жалғасы

```
textBox1.Text = read["Наименование"].ToString();
textBox2.Text = read["Адрес"].ToString();
textBox3.Text = read["Телефон"].ToString();
textBox4.Text = read["e_mail"].ToString();
textBox5.Text = read["Режим_работы"].ToString();
richTextBox1.Text = read["Описание"].ToString();
try
{
    pictureBox1.Image = ByteToImage((System.Byte[])read["Фото_на_карте"]);
    pictureBox2.Image = ByteToImage((System.Byte[])read["Эмблема"]);
}
catch(InvalidCastException)
{
    MessageBox.Show("Изображение отсутствует");
}
}
}
conn.Close();
}
void HideElements(bool isChanging)
{
    textBox1.Enabled = isChanging;
    textBox2.Enabled = isChanging;
    textBox3.Enabled = isChanging;
    textBox4.Enabled = isChanging;
    textBox5.Enabled = isChanging;
    button1.Enabled = !isChanging;
    button2.Enabled = isChanging;
    button3.Enabled = isChanging;
    button4.Enabled = isChanging;
    button5.Enabled = isChanging;
    button6.Enabled = isChanging;
    button7.Enabled = isChanging;
    richTextBox1.Enabled = isChanging;
}
void SaveImage(byte[] imagen,string valueName)
{
    conn = new SQLiteConnection("Data Source=" + dataName + ".db;");
    SQLiteCommand cmd = conn.CreateCommand();
    cmd.CommandText = cmd.CommandText = String.Format("UPDATE " + tableName + "
SET " + valueName + " = @0 WHERE ID = 1");
    SQLiteParameter param = new SQLiteParameter("@0", System.Data.DbType.Binary);
    param.Value = imagen;
    cmd.Parameters.Add(param);
    conn.Open();
    try
    {
        cmd.ExecuteNonQuery();
    }
}
```

Қосымша А жалғасы

```
        catch (Exception exc1)
        {
            MessageBox.Show(exc1.Message);
        }
        conn.Close();
    }

    private static ImageFormat GetImageFormat(string fileName)
    {
        string extension = Path.GetExtension(fileName);
        if (string.IsNullOrEmpty(extension))
            throw new ArgumentException(
                string.Format("Unable to determine file extension for fileName: {0}", fileName));
        switch (extension.ToLower())
        {
            case @".bmp":
                return ImageFormat.Bmp;
            case @".gif":
                return ImageFormat.Gif;
            case @".icon":
                return ImageFormat.Icon;
            case @".jpg":
            case @".jpeg":
                return ImageFormat.Jpeg;
            case @".png":
                return ImageFormat.Png;
            default:
                throw new NotImplementedException();
        }
    }
}

public Image ByteToImage(byte[] imageBytes)
{
    // Convert byte[] to Image
    MemoryStream ms = new MemoryStream(imageBytes, 0, imageBytes.Length);
    ms.Write(imageBytes, 0, imageBytes.Length);
    Image image = new Bitmap(ms);
    return image;
}

public byte[] ImageToByte(Image image, System.Drawing.Imaging.ImageFormat format)
{
    using (MemoryStream ms = new MemoryStream())
    {
        // Convert Image to byte[]
        image.Save(ms, format);
        byte[] imageBytes = ms.ToArray();
        return imageBytes;
    } } }
```