

ҚАЗАҚСТАН РЕСПУБЛИКАСЫНЫҢ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

СӘТБАЕВ УНИВЕРСИТЕТІ

Ақпараттық және телекоммуникациялық технологиялар институты

Программалық инженерия кафедрасы

Байғалиева Гүлайым Ерланқызы

Андройд платформасында бонустық карталарды сақтауға арналған қосымшасы

Дипломдық жобаға  
**ТҮСІНІКТЕМЕЛІК ЖАЗБА**

5B060200 – «Ақпараттану» мамандығы

Алматы 2019

ҚАЗАҚСТАН РЕСПУБЛИКАСЫНЫҢ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

СӘТБАЕВ УНИВЕРСИТЕТІ

Ақпараттық және телекоммуникациялық технологиялар институты

Программалық инженерия кафедрасы

**ҚОРҒАУҒА ЖІБЕРІЛДІ**

ПИ кафедра меңгерушісі

тех. ғыл. кандидаты,

ассистент-профессор

 Р. Юнусов

"17"                      2019ж.

Дипломдық жобаға

**ТҮСІНІКТЕМЕЛІК ЖАЗБА**

Тақырыбы: Андроид платформасында бонустық карталарды сақтауға арналған қосымшасы


5B060200 – «Ақпараттану» мамандығы

Орындаған

Байғалиева Г. Е.

Ғылыми жетекші

Сениор-лектор

 Р. С. Алгожаева

"19"                      2019 ж.

Алматы 2019

ҚАЗАҚСТАН РЕСПУБЛИКАСЫНЫҢ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

СӨТБАЕВ УНИВЕРСИТЕТІ

Ақпараттық және телекоммуникациялық технологиялар институты

Программалық инженерия кафедрасы

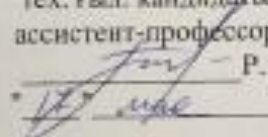
5B060200 – «Ақпараттану»

**БЕКТЕМІН**

ПИ кафедра меңгерушісі,

тех. ғыл. кандидаты,

ассистент-профессор

 - Р. Юнусов

2019ж.

**Дипломдық жоба орындауға  
ТАПСЫРМА**

Білім алушыға Байсазиева Гулайым Ералиқызы

Тақырыбы: « Android платформасында бөлшектік карталарды сақтау үшін арналған қосымша »

Университет ректоры бұйрығының № 1841-б "14" наурыз 2019 ж. шешімімен бекітілген.

Орындалған жобаның өткізу мерзімі

" 20 " мамыр 2019 ж.

Дипломдық жобаның бастапқы мәліметтері: Жобаның толқу жаты, технология бойынша техникалық құжаттама, техникалық тапсырма, жоба диаграммалары түрінде ақпаратты жасау, деректер қорына сақтау, тестілеу, тексеруге арналған программалық қамтамаларды жасау жүргізілген.

Есеп – түсініктеме жазбаның талқылауға берілген сұрақтардың тізімі:

a) тақырыбын талдау және соған орай мәліметтер көрсету;

b) жобаны құруға арналған қосымшаларды талдау;

в) жобаның жұмыс істеу принципін диаграмма түрінде талдау;

г) бағдарламаны құру, деректерді қосу және тестілеу;

Графикалық материалдар тізімі (міндетті суреттердің нақты көрсетілуімен):

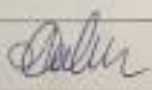
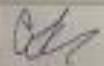
презентацияның 20 слайдпен берілген құжат түрінде ұсынылған.

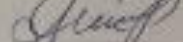
Ұсынылған негізгі әдебиеттер: 10 пайдаланылған әдебиеттер тізімінен


Дипломдық жобаны орындау  
КЕСТЕСІ

Бөлімдердің атаулары, зерттелген мәселелердің тізімі	Ғылыми жетекшіге және кеңесшілерге ұсыну мерзімі	Ескерту
1. Диплом жұмысының жоспар құрылымын құру.	14.01.2019	ноқ
2. Тапсырма қойылымы және бағдарламалау ортасын таңдау	18.01.2019	ноқ
3. Зерттеу тақырыбы бойынша ғылыми теориялық материалдарды жинау және негізгі бөлім беру бойынша есеп беру жазбасын дайындау	01.02.2019	ноқ
4. Дипломның екінші бөлімі – жобалау сызбаларын дайындау.	15.02.2019	ноқ
5. Жобаның веб-қосымшасын тестілеуден өткізу.	18.03.2019	ноқ
6. Дипломдық жобаға түсіндірме жазба жазуды аяқтау	26.04.2019	ноқ

Дипломдық жұмыс бөлімдерінің кеңесшілерінің аяқталған жұмысқа қойған қолтаңбалары

Бөлімдер атауы	Кеңес берушілер (аты-жөні, тегі, ғылыми дәрежесі, атағы)	Қолтаңба қойылған мерзімі	Қолы
Нормалақ бақылаушы	Алибиева Ж. А. Сенior - лектор	16.05.2019	
Бағдарламалық бөлім	Сман Н. О. Ассистент		

Ғылыми жетекші  Р.С.Алгожаева

Тапсырманы орындауға қабылдап алған студент  Г.Е.Байғалиева

Күні «01» 01 2019ж.

## АҢДАТПА

Бұл дипломдық жобада бонустық карталарды сақтауға арналған қосымша қарастырылады. Ол қосымша қолданушыларға, яғни көптеген дүкендердің сатып алушыларға арналған. Дүкендер өзінің бонустық жүйесін одан ары дамытып қоладнушыларға жеңілдіктер ұсынады және әр дүкен өз картасын береді. Қолданушыларда бар барлық карталарды ыңғайлы қолдану үшін жасалған қосымша.

Бұл жоба үш бөлімнен тұрады. Бірінші бөлімінде жалпы бонустық карталар жүйесі жөнінде түсіндіріледі және ол туралы мағлұматтар беріледі. Екінші бөлімінде қосымшаны жасауда қолданылаған бұлтты сервис жүйесі Firebase деректер базасы туралы және Android платформасы жайлы жазылған. Үшінші бөлімде жасалған қосымшаның жұмыс істеу принципі мен оның иллюстрациялары көрсетілген.

## АННОТАЦИЯ

В данном дипломном проекте рассматривается приложение для хранения бонусных карт. Оно предназначено для пользователей, то есть для покупателей магазинов, которые используют бонусные карты. Магазины развивают бонусную систему и каждый магазин выдает свою карту покупателям. Приложение создано для упрощения жизни клиентов использующих бонусные карты.

Этот проект состоит из трех частей. В первой части разъясняются общие сведения о системе бонусных карт и приводятся сведения о ней. Во втором разделе описана облачная система базы данных Firebase и платформа Android, используемая при создании приложения. В третьей части указаны принцип действия разработанного приложения и его иллюстрации.

## **ANNOTATION**

In this graduation project is considered an application for storing bonus cards. It is intended for users, that is, for buyers of stores that use bonus cards. Shops develop a bonus system and each store gives its card to customers. The application is designed to simplify the lives of customers using bonus cards.

This project consists of three parts. The first part explains the General information about the bonus card system and provides information about it. The second section describes the firebase cloud database system and the Android platform used to create the application. The third part shows the principle of operation of the developed application and its emulation.

## МАЗМҰНЫ

	Кіріспе	8
1	Дисконттау картасының тарихы.	9
1.1	Бонустық карталар туралы жалпы түсінік	9
1.2	Бонустық карталар жүйесі	10
1.3	Бонустық карталарды сақтауға арналған мобильді қосымшаға ұқсас қосымшаға шолу	12
2	Жобалау барысында қолданылатын бағдарламалық қамтамалар жайлы сипаттама	16
2.1	Firebase	15
2.2	Неге Android?	17
2.3	Android үшін даму ортасының негізгі ерекшеліктері	20
3	Жобалау және модельдеу барысында іске асырылатын амалдарды uml диаграммалар арқылы бейнелеу	22
3.1	Есептің қойлымы	26
	Қорытынды	32
	Пайдаланылған әдебиеттер тізімі	33
	А қосымшасы. Бағдарлама мәтіні	34
	Спецификация беті	45



## КІРІСПЕ

Компьютерлер пайда болғаннан кейін және олардың ақпаратты өңдеу үшін пайдалануы көптеген ақпараттық-анықтамалық және есеп айырысу тапсырмаларын автоматтандыруға мүмкіндік берді. Қоғамның дамуы адам қызметінің барлық салаларында басқару және ақпарат өңдеу принциптерін кеңінен қолдануды көздейді. Сонымен қатар, әртүрлі өндіріс аймақтарындағы автоматтандырылған басқару және ақпараттық өңдеу жүйелерін енгізу және уақытты және еңбек ресурстарын қысқартатын, әртүрлі қарапайым операцияларды орындау мен есептеулерді жүзеге асырудан босататын, еңбек нәтижесінің сапасын жақсарту, қателердің ықтималдығын азайтатын маңызды рөл атқарады, жұмысындағы қателіктер мен төтенше жағдайлар.

Қазіргі кезде мобильді қосымша смартфондарда, планшеттерде және басқа мобильді құрылғыларда жұмыс істеуге арналған бағдарламалық жасақтама болып табылады. Көптеген мобильді қосымшалар құрылғыда алдын ала орнатылған немесе оны App Store, BlackBerry App World, Google Play, 1Mobile Market, Windows Phone дүкені және басқалары сияқты онлайндық қолданбалар дүкендерінен тегін немесе ақысыз жүктеуге болады.

Мобильді қосымшалар жоғары деңгейлі бағдарламалау тілдерінде жазылады, одан кейін ең жоғары өнімділік үшін операциялық жүйенің машина кодексіне жазылады. Қолданбаның дамуы өз ерекшеліктеріне ие: мобильді құрылғылар батареяда жұмыс жасайды және дербес компьютерлерге қарағанда өнімділігі төмен процессорлармен жабдықталған. Сонымен қатар, заманауи смартфондар мен планшеттер гироскоптар, акселерометрлер және камералар секілді қосымша құрылғылармен әмбебап жабдықталған, олар қосымшаның функционалдығын кеңейтуге мүмкіндік береді. Сондықтан мобильді қосымша жасау қазіргі тандағы ең өзекті мәселе болып табылады.

Дипломдық жұмыстың мақсаты мен міндеттері

Бұл дипломдық жұмысты жазудың басты мақсаты бонустық карталармен жұмыс істейтін дүкендердің клиенттерінің өмірін жеңілдету. Барлық адамдар бір дүкенде ғана емес, әртүрлі дүкендерден керекті затын сатып алатындығын білеміз, және дүкендердің әрқайсысы өзінің жинақтау карталарын шығарады. Сондықтан ал сатып алушылар әмиянда көптеген карталар пайда болады, тіпті симай қалатындай жағдайға жетеді. Сонымен қатар үйде ұмытып кететін жағдайларда болады және сатып алған кезде бонустар берілмейді. Сондықтан, менің қосымшам клиенттерге осы мәселе бойынша көмектеседі, өйткені ол барлық карталарды смартфонға сақтайды және сіздің картаңызды үйде ұмытып кетпейсіз, әмияныңызда да жеткілікті орын босатады.

# 1 Дисконт картасының тарихы

## 1.1 Бонустық карталар жөнінде жалпы түсінік

Қазіргі заманғы дисконттық карталардың алғашқы үлгілері 20-шы жылдары АҚШ-та пайда болды. Ол карточкалар картоннан жасалған және ірі мәмілелер жасаған кезде бай азаматтарға арналған «лоялдык» бағдарламасына қатысуды растайды, сондай-ақ түрлі дүкендер мен салондарға барады. Мысал ретінде, 1949 жылы Diners Club дисконттық жүйесін құруды сипаттауға болады. Жүйе бүкіл ел бойынша сатып алуға, мейрамханаларда тамақтануға және жеңілдіктер алуға мүмкіндік берді. Карточкалардың таралуы оның сыртқы келбетін үнемі модернизациялауды талап етті. Ал өткен ғасырдың 50-ші жылдарының ортасында бірінші пластикалық карточка шығарылды.

1960 жылы IBM американдық үкіметтің қауіпсіздік жүйелері бойынша келісімшарт бойынша жұмыс істеп, пластикалық карталар үшін талап етілетін ақпаратты сақтау тығыздығын жеткілікті қамтамасыз етпейтіндіктен, штрих-код пен перфорацияны пайдалану нұсқаларын қарады бірақ ол ұсынысты қабылдамады. Магнитті медиа Екінші дүниежүзілік соғыстан бері және 1950 жылдары пайда болған компьютерлік ақпараттың репозитарийлерінен белгілі болды. Forrest Parry инженері ұзақ жұмыс барысында магниттік жолақты пластикалық картамен байланыстыру үрдісі ойлап тапты. Оның үйде болған әйелі күйеуінің киімін үтіктеп жатып оған магниттік картаны үтікпен жабыстырса болады деген ұсыныс айтты. Және ол таспаны пластмассадан темірмен ерітуге шақырды. Қызып тұрған үтік пластикалық картаның бетіне ериді және материалдың арасында өте жоғары адгезия дәрежесі алынды.

Дисконт карточкасы, дисконттық карта (купон, ваучер, купон және т.б.) - бұл тұтынушының сауда нүктелерінде (дискілерінде) немесе жеңілдіктер клубының мүшелеріне дисконттық карталарды пайдалану ережелерін сақтау кезінде жеңілдіктер алуға мүмкіндік беретін құрал болып табылады.

Дисконт карталары, әдетте, өз тұтынушыларының адалдық бағдарламаларына сервистік қызмет көрсету үшін тізбекті дүкендер, автожанармай құю станциялары, автосалондар, спорттық клубтар, кір жуатын орындар, дәріханалар, клиникалар, авиакомпаниялар, сұлулық салондары, мейрамханалар, клубтар, интернет-провайдерлер, вагон жөндеу цехтары, үй жануарлары дүкендерін және т.б. жерлерде қолданылады.

Дисконт карта түрлері:

– пластикалық карта штрих-кодпен, магниттік жолақпен немесе чиппен жабдықталған;

– карточка пластикалық, картоннан немесе қағаздан жасалған, онда жеңілдіктер мөлшері, жарамдылық мерзімі және / немесе тауарлар тобы көрсетіледі;

– электрондық дисконттық карта - жеңілдікті анықтаудың кез-келген тәсілін қамтитын электрондық тасымалдағышта;

– штрих-код– бұл бетіне қолданылатын графикалық ақпарат, техникалық құралдармен оқуға мүмкіндік беретін өнімдерді таңбалау немесе қарау, қара және ақ жолақтардың кезектілігі немесе басқа геометриялық пішіндер ретінде көрсетіледі.

## 1.2 Бонустық карта жүйесі

Жергілікті бонустық карта – бір дүкенде немесе бір тарату желісінде жұмыс істейтін магнит жолағы бар пластикалық карта. Сатып алушы сатып алуды жүзеге асырады және картаға бонустар алады, оны келесі сатып алуға тек дүкенде немесе осы бонустық бағдарламаны іске асыратын дүкендер желісінде жұмсауға болады. Егер бонустар жеткіліксіз болса, сатып алу ақшалай нысанда ішінара төленуі мүмкін.

Бонус – алдын-ала төленген карточка – магнит жолағы бар сатып алушы белгілі бір көлемде төлейтін пластикалық карточка, содан кейін оларды сауда-сервистік компания тауарына / қызметіне құның төлеуге жұмсайды.

Дүкенде кассалық функциясы бар банкомат орнатылған және ол болашақ сатып алулар үшін алдын-ала төлемдерді қабылдауға мүмкіндік береді. Алдын-ала төленген карточка бойынша төлем үшін, адамға стандартты нұсқаға қарағанда көбірек сыйақы беріледі. Сонымен қатар, дүкен клиенттерден алдын-ала төлеу ретінде тегін несие алады. Бонустық картаның артықшылығы - сіз өзіңіздің шотыңызда нақты ақшаны жинап, желідегі дүкендерде сатып алуға нақты ақшалай қаражат ретінде төлей алатындығыңыз.

Тұрақты пластикалық карта – дисконттық картаның ең әдеттегі түрі. Ол сатып алу кезінде пластикалық карточка кассирге ұсынылады, кассир өз кезегінде оның қол жетімділігі үшін жеңілдік береді. Әрине, мұндай карта сатып алуды жинақтау және жеңілдіктер беру үшін ішкі базаның сақталуын білдірмейді. Шағын бизнес үшін бұл дисконттық карточкалардың ең қолайлы жүйесі және арзан деп саналады. Мұнда қазіргі тұтынушы дүкендердің бір немесе бірнеше тізбегінде дисконттық жүйе сияқты нәрсе бар екеніне сенімді болу керек, клиенттің сенімділігі өсу үшін қолданылады.

Магниттік карта – компьютерді магниттік картаны оқу құрылғысы арқылы жүзеге асады. Сатып алуды жүзеге асырған кезде кассир карта оқырманына магниттік картаны енгізеді. Бұл жерде бәрі өздігінен жүреді: жеңілдік мөлшері компьютерден анықталады, ол сатып алынатын тауарлар санатына және сатып алу жүргізілген сомаға байланысты болады. Бонустық карта барлық жеңілдіктер мен жинақтарды бақылауға, бонустың құның өзгертуге мүмкіндік береді. Клиент магнит картасын сатып алуға келіссе, ол өзі туралы негізгі мәліметтерді ұсынады: паспортының деректері мен қызмет түрі үшін, сондай-ақ оған компанияға жеңілдік беретін компания туралы қай жерден білетінін көрсетіп кетуі керек. Бұл белгілі бір рейтингті ұстап тұру және өнімнің таралу аумағын ұлғайту үшін жасалатындығы анық.

Бұдан басқа, сұрақ қою дұрыс болады: тұтынушы мұндай карточкаларға пайда әкелетіндігінен нені жоғалтып жатыр? Мұнда маркетингтік және жарнамалық трюктердің күші енеді, бірақ психологиялық фактор маңызды рөл атқарады. Адам үнемі баратын дүкенге жеңілдік алған кезде, ол оған көп көлемде сатып алуға көмектеседі. Үлкен бөлшек сауда орындарында, әдетте, тауарларды жеңілдіктермен сатып ала алатын болсаңыз, онда мұндай тауарларға жеңілдік карточкасы түрінде қосымша жеңілдік қолданылмайды. Бұл науқанға қатысатын өнімдер көбінесе баға белгілеуін «қызыл бағамен» қояды, бірақ шын мәнінде, жеңілдік бойынша баға, ол болмаған кездегі бағадан ерекшеленбейді. 5 немесе 10% жеңілдік карточкасы компания «қызыл» өнімдерден алуға үміттенетін бюджетті қатты бұзады. Шын мәнісіне келгенде бұл сауатты қадам болып табылады, сондықтан ірі желілер ешқашан өз пайдасын жоғалтпайды.

Дисконттау жүйесі – бұл өте пайдалы нәрсе. Көп ұзамай қызметтердің барлық салаларында және саудадағы жеңілдіктер мен карталардағы жинақ ақылы болады. Клиент бір дүкенде мұндай артықшылығы бар екенін біліп ол басқа дүкенге бармайтындығын көрсетеді. Яғни карталар көмегімен клиенттердің ағыны мен компанияның айналымын айтарлықтай арттыруға болады, сондықтан жеңілдік жүйесі сатып алушылар үшін ғана емес көбінесе дүкендерге пайдалы.

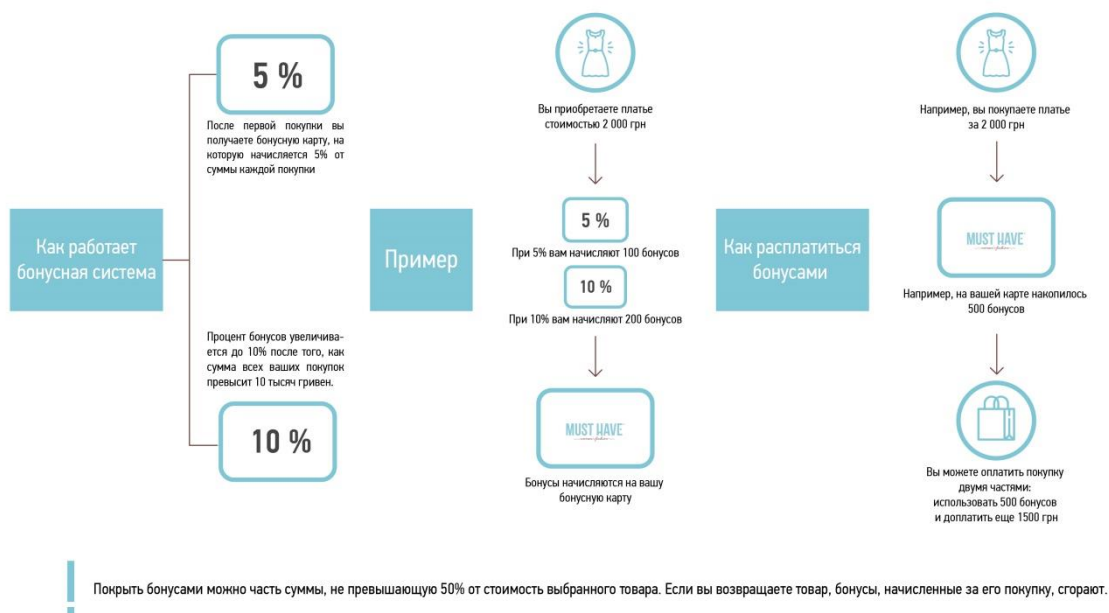
Ромирдің кезекті Бүкілресейлік сауалнамасы барысында зерттеуші холдингті респонденттерге бөлшек сауда желілерінің жеңілдік / жинақтау (бонустық) карталарын пайдалану туралы бірқатар сұрақтар қойды.

Сатып алушы 2012 және 2006 жылдары осындай зерттеулер жүргізілді және ресейліктердің адалдық бағдарламаларына деген көзқарасы уақыттың қалай өзгеретінін қадағалауға мүмкіндік береді.

Бірінші кезекте, дисконт немесе сыйлық карточкаларының бар екендігі туралы сұраққа жауап берген ресейліктердің 84% -ы жеңілдік карточкасы бар екенін мойындады. Респонденттердің жартысынан көбі (56%) бонустық карточка болуын атап өтті. Сыйлық карталары респонденттердің 18% -ына қол жетімді. Және әрбір сегізінші (12%) ешқандай картаға ие емес деп жауап берді.

Осылайша, алты жыл бұрынғы кезеңмен салыстырғанда жеңілдіктер мен қаржыландырылған ресейліктердің жалпы үлесі дерлік өзгерген жоқ. Бірақ, жинақ (бонустық) пен дисконттық карточкалардың ұстаушыларының арасында ғана бөлу болды. 2012 жылдан бастап дисконттық карточка ұстаушылардың саны біршама төмендеді, ал жинақтаушы карточкалар ұстаушылардың үлесі, керісінше, артты.

Бонустық карталар жүйесінің қалай жұмыс істейтінін төменде 1.1-суреттен көрсетілген .



## 1.1-сурет – Бонустық карта жұмысының жүйесі

### 1.3 Бонустық карталарды сақтауға арналған қосымшаға ұқсас қосымшаға шолу

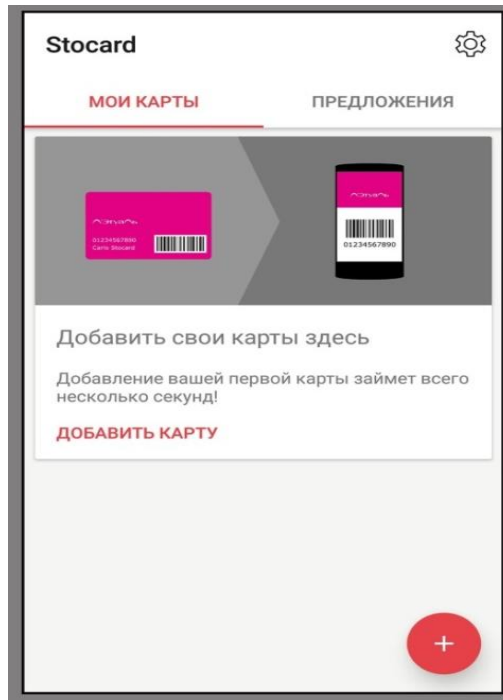
Қазіргі заманда кез-келген адам мобильдік телефонсыз жүрмейді және кез-келген адам мобильдік қосымша арқылы өз уақытын үнемдеуге тырысады. Сондықтан әрбір жүйе өзіне мобилді қосымша жасауды жөн көреді.

Виртуалды дисконттық карталар: мәні

Әмияныңызда немесе қапшығыңызда біреуден артық жеңілдік картасы бар және сауда желілері маркетингтік науқандардың бір бөлігі ретінде олардың пакеттерін береді. Бұрын, тегіс және лазерлік сканерлер кассаға жеңілдік карталарын оқуға арналған болса, қазір олар жаңа буынға өтіп- фото сканерлермен алмастырылады. Олардың көмегі арқылы жеңілдік туралы ақпарат телефонның экранындағы картаның фотосуретінен де оқуға болатын болды.

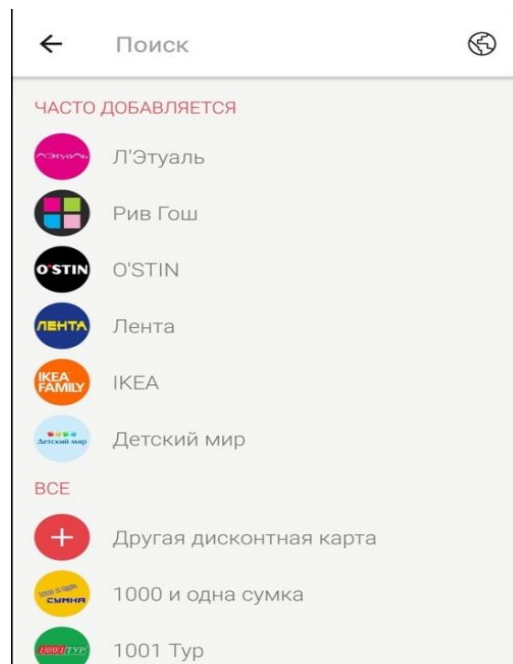
Осылайша, пластик қажет емес. Оның орнына сіз өздерінің суреттерін сандық түрінде сақтауға мүмкіндік беретін дисконттық карта агрегаттарының қолданысын қолдана аласыз.

Сіз өзіңіздің картаңызды «+» батырмасын басып енгізесіз және ол карталар жүйесінде сақталып қалады. Оны 1.2-суреттен көруге болады.



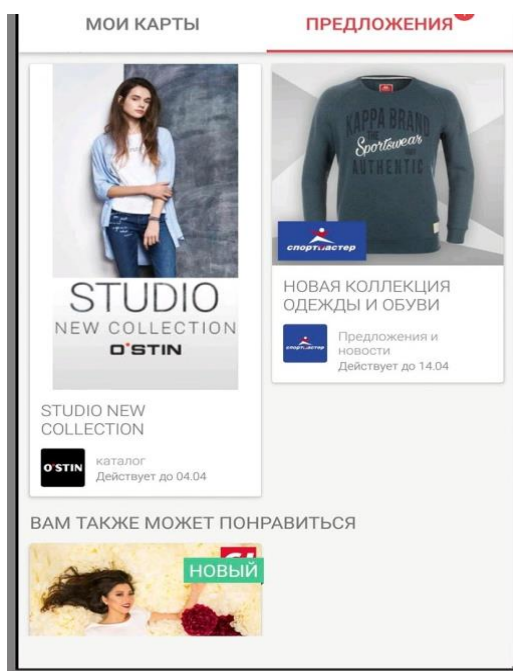
**1.2-сурет – Картаны енгізу**

PINbonus штрих-кодты картаңыздан оқиды және оны телефонда оқылатын түрде сақтайды. Енді дүкеннің кассасына штрих-кодты бағдарлама экранында көрсетуге болады. Кассир оны сканер деп санайды, ал сіз жеңілдік аласыз. Сіздің карталарыңыз карталар бөлімінде сақталады және бағдарламаның жадысында көптеген дүкендердің бонустық карталар тізімі бар. Карталар бөлімі 1.3-суретте көрсетілген.

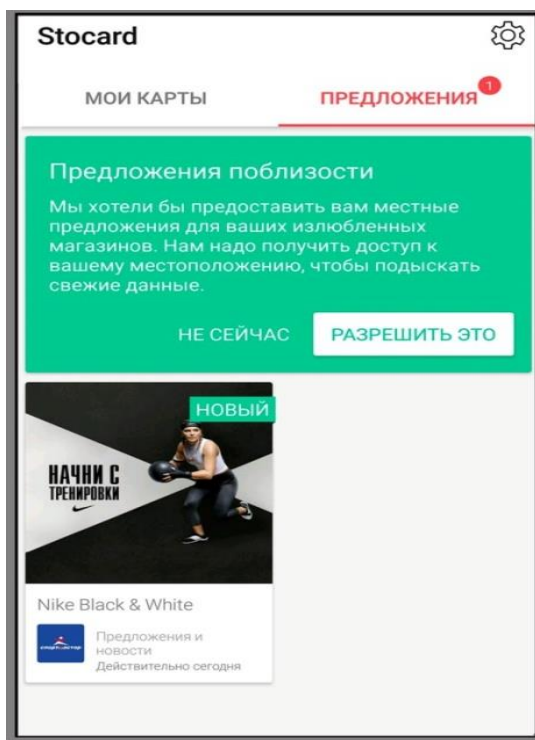


**1.3 сурет – Карталар бөлімі**

Қосымшадан дүкенде болып жатқан акциялар туралы біле аласыз. Дүкенге келген жаңа топтамалар туралы және жеңілдіктер бөліміне кеткен заттар туралы жарнама арқылы білсеңіз болады. Акциялар бөлімі 1.4-суретте және 1.5-суретте көрсетілген.



1.4 сурет – Акциялар бөлімі



1.5 сурет – Акциялар бөлімі (жалғасы)

Түрлі дүкендердің дайын штрих-коды базасында жұмыс істейтін дисконттық карталар үшін қарапайым сұраныс болып табылады. Бағдарламаны ашып, тізімнен қажетті дүкенді таңдаңыз (Sportmaster-ден Wild Orchid-қа 200+ түрлі нұсқалар бар) және дүкен картасын телефонға әкеліңіз.

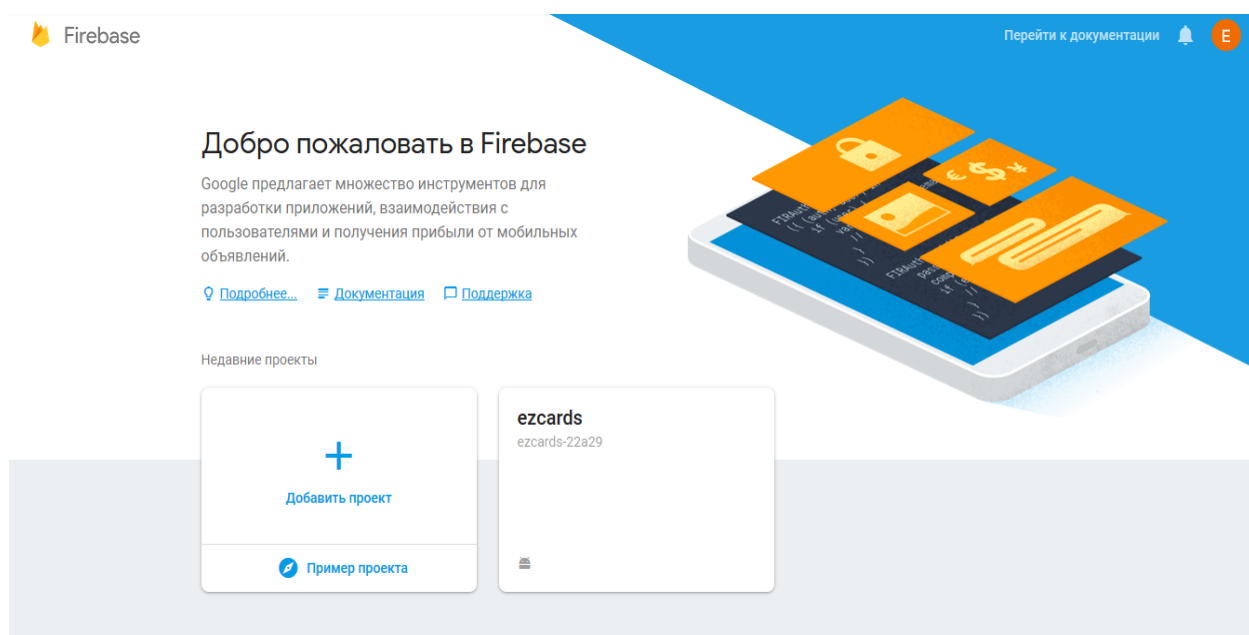
Әрбір дүкеннің жарнамасы, оның акциялары жөнінде «Предложения» бөлімінен білсеңіз болады.



## 2 Жобалау барысында қолданылатын бағдарламалық қамтамалар жайлы сипаттама

### 2.1 Firebase

Firebase – 2011 жылы Andrew Lee және James Templin негізін қалаған американдық компания, бұлтты сервистерді жеткізуші. Firebase деректер қорының бастапқы бетін 2.1-суреттен көре аласыз.



2.1 сурет – Firebase басқы беті

Негізгі қызмет – бұл бағдарлама әзірлеушілеріне бірнеше клиент арасында деректерді сақтауға және синхрондауға мүмкіндік беретін NoSQL бұлт дерекқоры. Android және iOS операциялық жүйелеріне арналған қосымшалармен интеграциялау ерекшеліктері қолданады, API JavaScript, Java, Objective-C және Node.js бағдарламаларында іске асырылады, сонымен қатар REST стилінде деректер базасымен тікелей JavaScript, JavaScript, AngularJS, React, Ember.js және Backbone.js. Деректерді шифрлау үшін API бар.

Firebase әзірлеуші пакеті интуитивті API интерфейсін біріктіреді, жеке пакеттерді басқару қажеттігін жоққа шығарады. Сондықтан, сізге қажетті нәрсені ғана таңдап, кешенді шешімді пайдаланыңыз. Google инфрақұрылымын пайдаланатын платформа даму мен дамудың әр кезеңіне қажетті мүмкіндіктерді ұсынады.

Негізгі артықшылықтары:

- жұмыс жылдамдығы;
- аяқталған инфрақұрылым;
- статистика;

- кросс-платформа;
- масштабтау;
- тегін электрондық пошта арқылы қолдау.

Жұмыс жылдамдығы. Firebase әзірлеу пакеті жоғары сапалы қосымшаларды жеңілдететін және жеделдететін интуитивті API-ді қамтиды. Соған қоса, сіздің иелігіңізде пайдаланушы базасын кеңейтуге, кірісті ұлғайтуға қажетті барлық құралдар бар - сіз өз мақсаттарыңыз үшін жарамдыларды таңдауыңыз керек.

Аяқталған инфрақұрылым. Күрделі инфрақұрылымды жасаудың немесе бірнеше басқару панелімен жұмыс істеудің қажеті жоқ. Оның орнына, пайдаланушылардың қажеттіліктеріне назар аударуға болады.

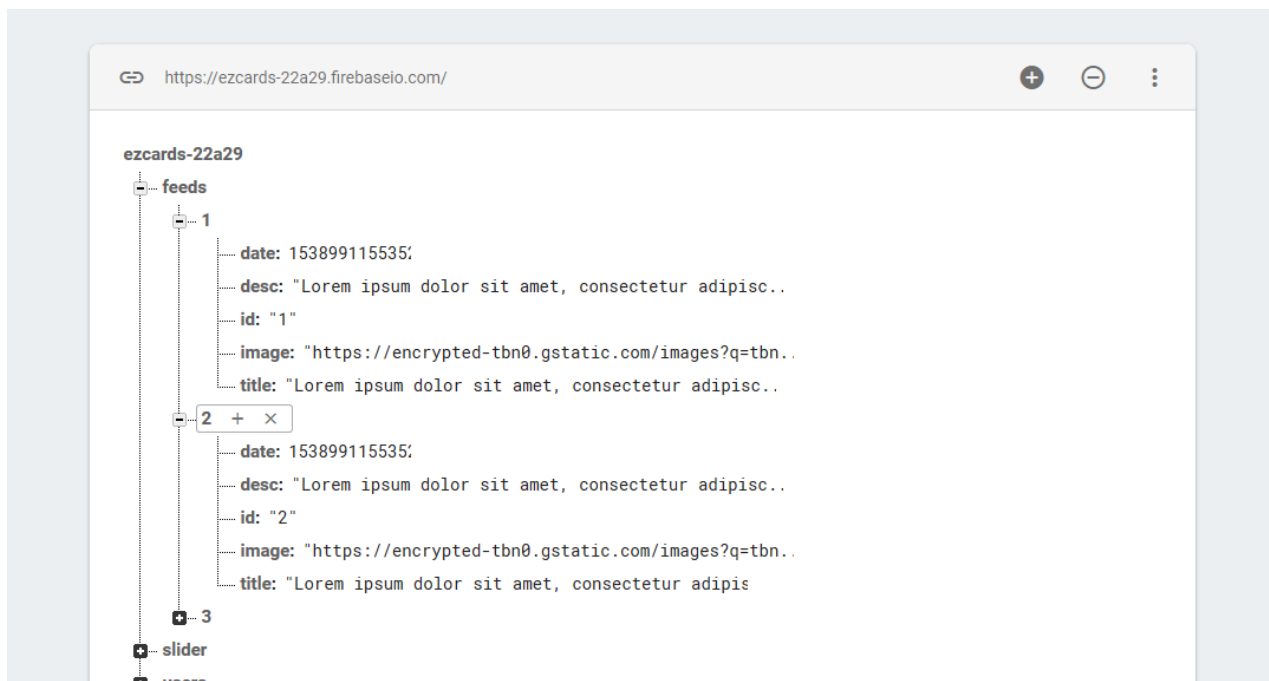
Статистика. Firebase-тің қақ ортасында мобильді құрылғыларға арналған арнайы аналитикалық құрал. Firebase үшін Google Analytics пайдаланушыларыңыздың әрекеттері туралы ақпаратты алуға, қосымша мүмкіндіктермен әрекетті дереу жасауға мүмкіндік береді.

Кросс-платформа. Firebase Android, iOS, JavaScript және C ++ үшін әзірлеуші пакеттері арқасында кез-келген платформада жұмыс істейді. Соған қоса, Firebase-ге сервер кітапханаларын немесе REST API-ні қолдана аласыз.

Масштабтау. Егер сіздің қосымшаңыз кеңінен танымал болса және оған жүктеме жоғарыласа, сіз сервер кодын өзгерте алмайсыз немесе қосымша ресурстар тартасыз - Firebase сіз үшін мұны жасайды.

Тегін электрондық пошта арқылы қолдау. Firebase тобы және Google даму сарапшылары Stack Overflow және GitHub ресурстарына қатысты сұрақтарыңызға жауап береді.

Проектіде қолданылған деректер базасы. Әрбір қоладнушының өзіне сай уникалды id номері бар және оның әрбір қосылған картасы деректер базасына автоматты түрде қосылып отырады. Онымен қоса администратор көмегімен база арқылы карта немесе басқа қолданушы және басқа да мәліметтер енгізсе болады. Қосымшада қолданылған Firebase деректер қорының қолдануын 2.2-суреттен көрсеніз болады.



**2.2-сурет – Деректер базасы**

## **2.2 Неге Android?**

Android қосымшаларының дамуы туралы айтатын болсақ, осы операциялық жүйенің танымалдығын атап өту мүмкін емес. Оның басты бәсекелесі Apple-мен салыстырғанда Android-тің нарықтағы үлесі өте үлкен - 70% -дан астам. Яғни, смарт құрылғылардың он иесінің жетеуі Android-ге арналған гаджетті пайдаланады.

Бұған қоса, Google бүгінгі күнге дейін Android-тің көптеген жаңылыстыратын кемшіліктерін анықтады: жүйенің қауіпсіздігі, төмен қуатты құрылғылардағы OS баяулауы және т.б. және оны түзеді.

Бағдарлама дизайны, өз кезегінде, ортақ номиналға келді - компания әзірлеушілерге арналған нұсқаулықтарды әзірледі, енді қосымшалар керемет көрінеді.

Google компаниясы Android әзірлеушілерінің көптеген топтарымен кездесіп, Android қосымшаларын дамытуды едәуір жеңілдетті және дамудың жылдамдығын едәуір арттыруға мүмкіндік беретін IntelliJ Idea негізіндегі Android Studio-ді ыңғайлы әзірлеу ортасын шығарды, себебі ол нарықтағы қолданыстағы шешімдермен салыстырғанда үлкен қадам болып табылады. Android Studio - бұл 2013 жылдың 16 мамырында Google I/O конференциясында анонсталған Android платформасымен жұмыс істеу үшін біріктірілген даму ортасы (IDE).

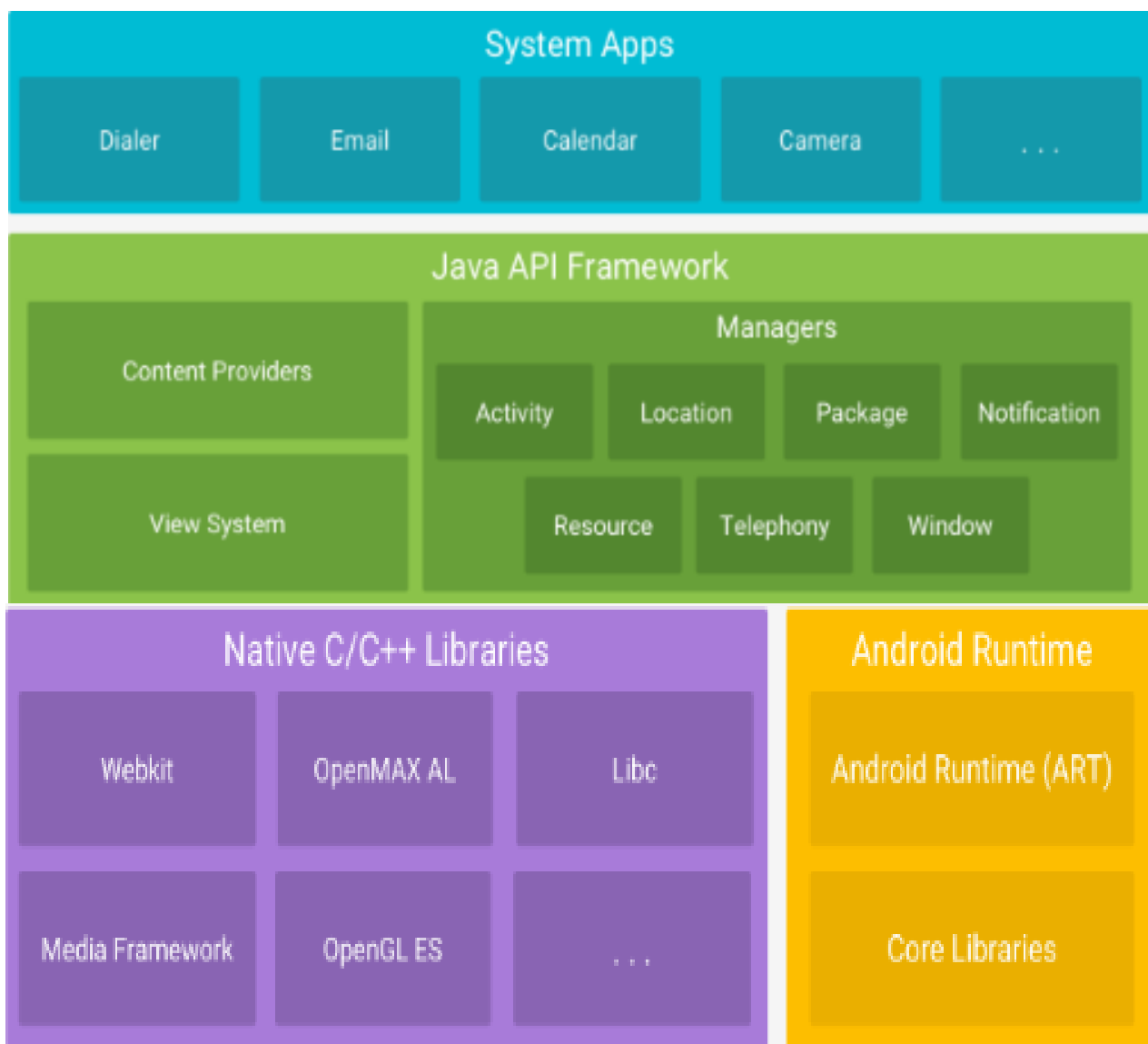
Android – бұл смартфондар, планшеттер, электронды кітаптар, сандық ойыншылар, сағаттар, фитнес білезіктер, ойын консолигі, ноутбуктар,

нетбуктар, смарт-кітаптар, Google Glass әйнектері , теледидарлар және басқа құрылғылар автокөлік ойын-сауық жүйелеріне қолдау көрсететін операциялық жүйе.

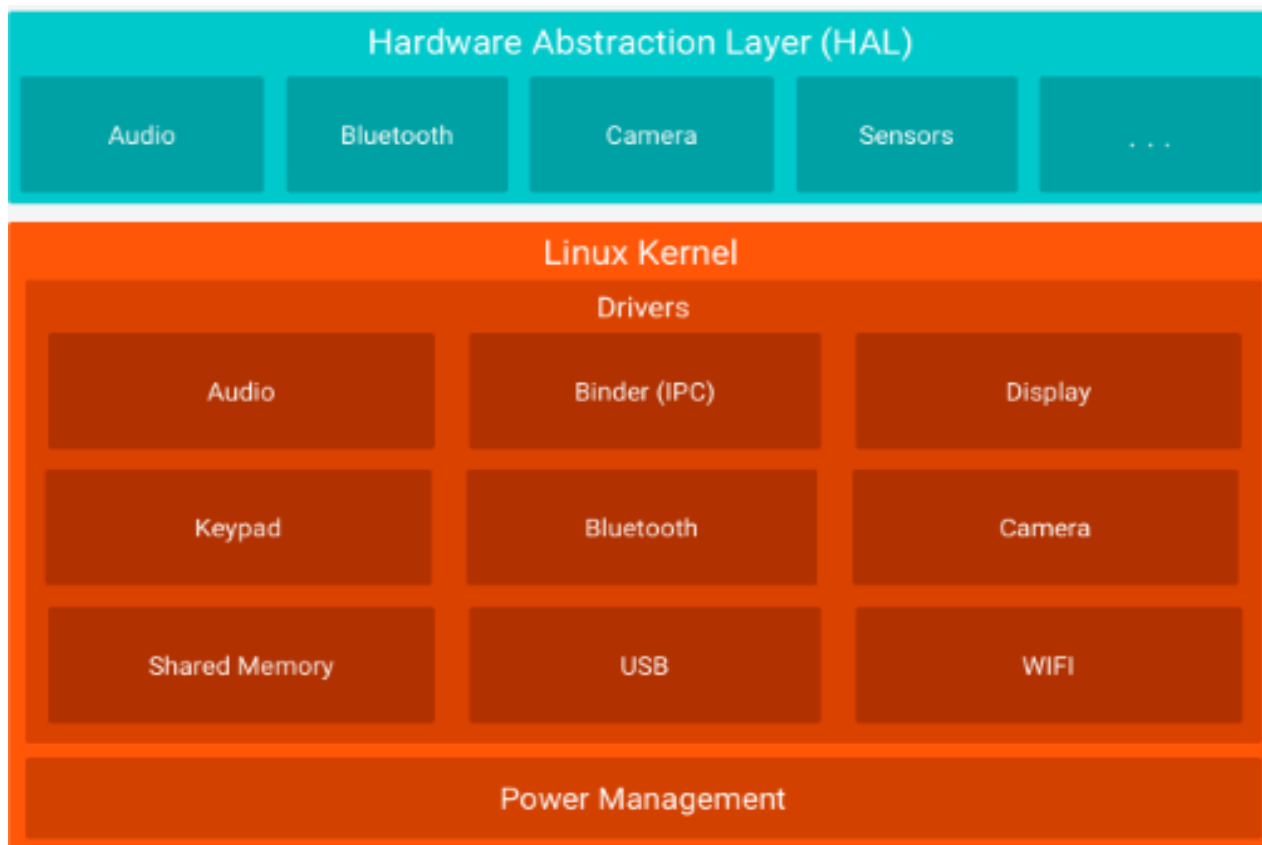
Қазірге кезде Android ен танымал және қолданысқа ие операциялық жүйелердің бірі болып табылады. Сол себепті қосымша Android платформасында жазылған.

Платформа сәулеті

Android – ашық көзі, Linux-ға негізделген бағдарламалар жиынтығы, кең ауқымды құрылғылар үшін және форматтағы факторлар. Төмендегі диаграмма Android платформасының негізгі компоненттерін 2.3-сурет, 2.4-сурет көруге болады.



**2.3-сурет – Android платформасының негізгі компоненттері**



**2.4-сурет – Android платформасының негізгі компоненттері(жалғасы)**

### **2.3 Android үшін даму ортасының негізгі ерекшеліктері**

Android әзірлеу ортасының ең қымбат бағасы API болатын. Android-ге арналған бағдарламалар – бейтарап платформа, сіздің телефоныңыздың ажырамас бөлігі, ол бағдарламаларды жасауға мүмкіндік береді. Келесі тізімде Android функциясы көрсетіледі:

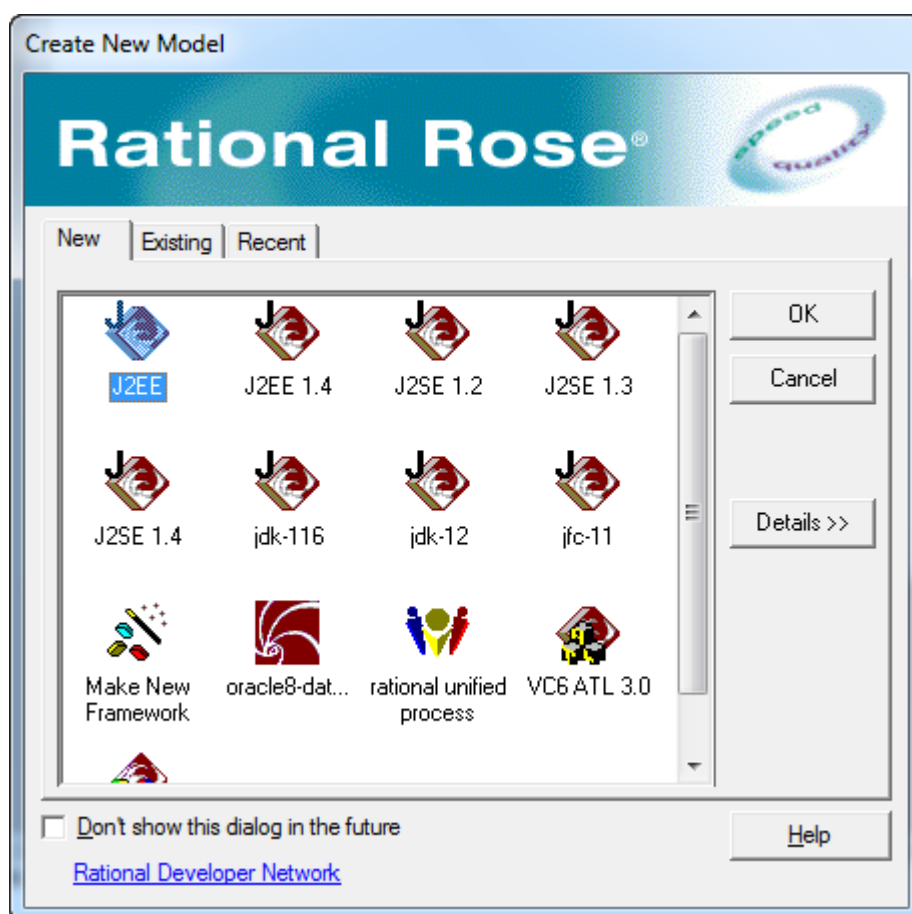
- пайдалануға, таратуға және өңдеуге лицензиялардың жоғалуы, сондай-ақ дайын бағдарламалық өнімді сертификаттау;
- Wi-Fi құрылғыларына кіру;
- GSM телефоны, EDGE және 3G желілерінде телефония мен деректерді беру үшін қоңыраулар шалу, қабылдау, SMS жіберу және қабылдау;
- GPS сияқты навигациялық қызметтері бар кешенді API;
- мультимедиялық құрылғыларды толық бақылау, соның ішінде камера мен микрофоннан ақпаратты жазу немесе жазу;
- сенсор сенсорлары үшін API;
- IPC хаттамаларын жеткізу;
- жалпы мәліметтер қоймасы;
- фондық қолданбалар мен процестер;

- тірі тұсқағаздар үшін виджеттер, Live wallpaper және Live wallpapers;
- WebKit браузері мен HTML5-нің қолдауымен кіріктірілген браузерде ашық көз.

### 3 Жобалау және модельдеу барысында іске асырылатын амалдарды uml диаграммалар арқылы бейнелеу

UML – бағдарламалық жасақтама әзірлеу, бизнес-үдерістерді модельдеу, жүйелік жобалау және ұйымдастыру құрылымдарын көрсету үшін және объектілі үлгілеу үшін графикалық сипаттама тілі болып табылады.

Rational Rose – UML әдістемесін тарату аспабы. Ұқсас есептерді шешу үшін қолданылатын басқа да визуальды аспаптар. Өңдеу функционалдығын сипаттау – қолдану варианттар диаграммасы, функция орындалу реттері– әрекеттердің тізбектер диаграммасы (талаптар және шектеулер), өңдеу элементтерінің бір-бірімен байланыс және бар болу сипаттамасы – кооперативті диаграммалар (талаптар және шектеулер). Интерфейсті жобалау. Объектілік жобаның өңделуі, ОМ 6 қосымшасына сәйкес, негізінен «прецеденттер диаграммасы» немесе екінші аталуы – «Қолдану варианттар диаграммасы» (ҚВД) басталады. Rational Rose диаграммалармен жұмыс жасау бағдарламасын 3.1-суреттен көре аласыз.



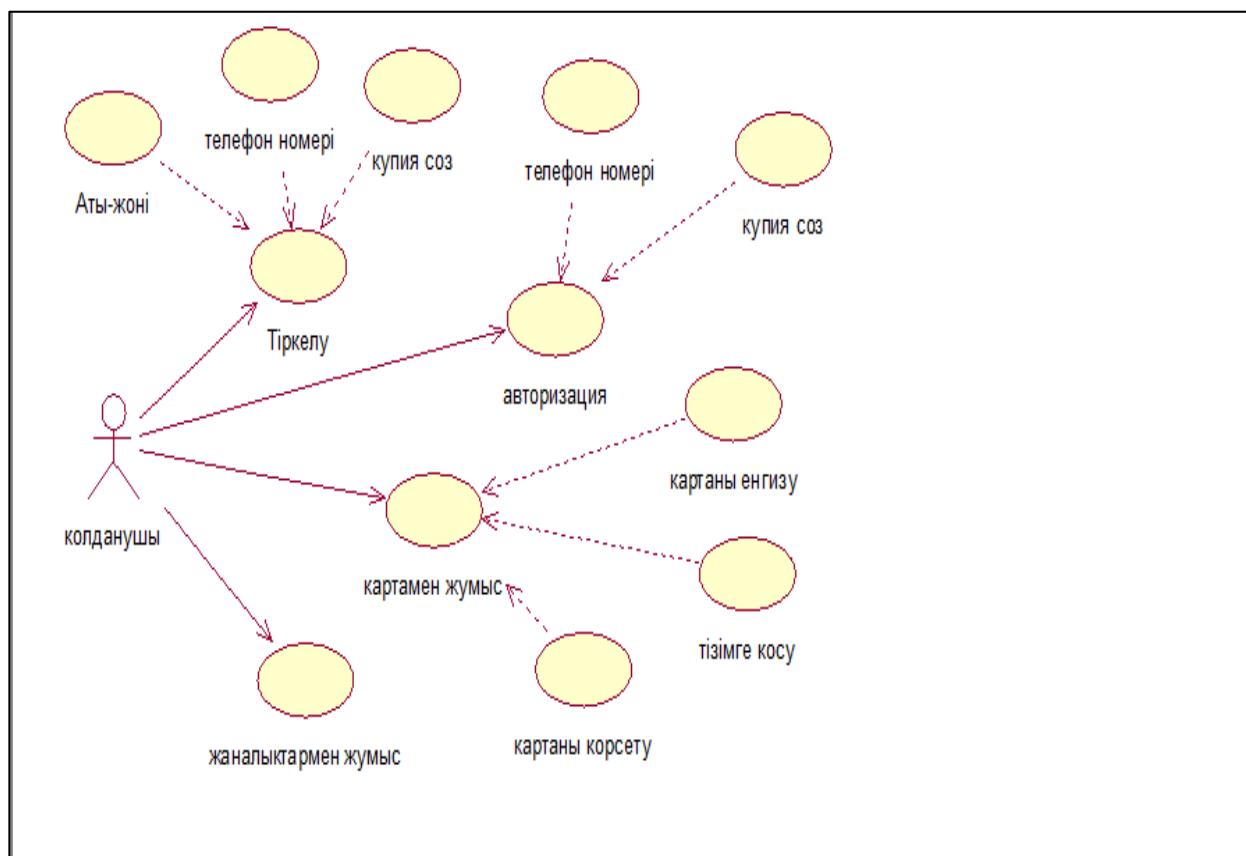
3.1-сурет – Rational Rose диаграммалармен жұмыс жасау бағдарламасы

Прецеденттер диаграммасы (UML диаграммасында қолданыңыз) - актерлер мен қолданушылар арасындағы қатынастарды бейнелейтін диаграмма

және жүйені тұжырымдамалық деңгейде сипаттауға мүмкіндік беретін қолдану үлгісінің ажырамас бөлігі.

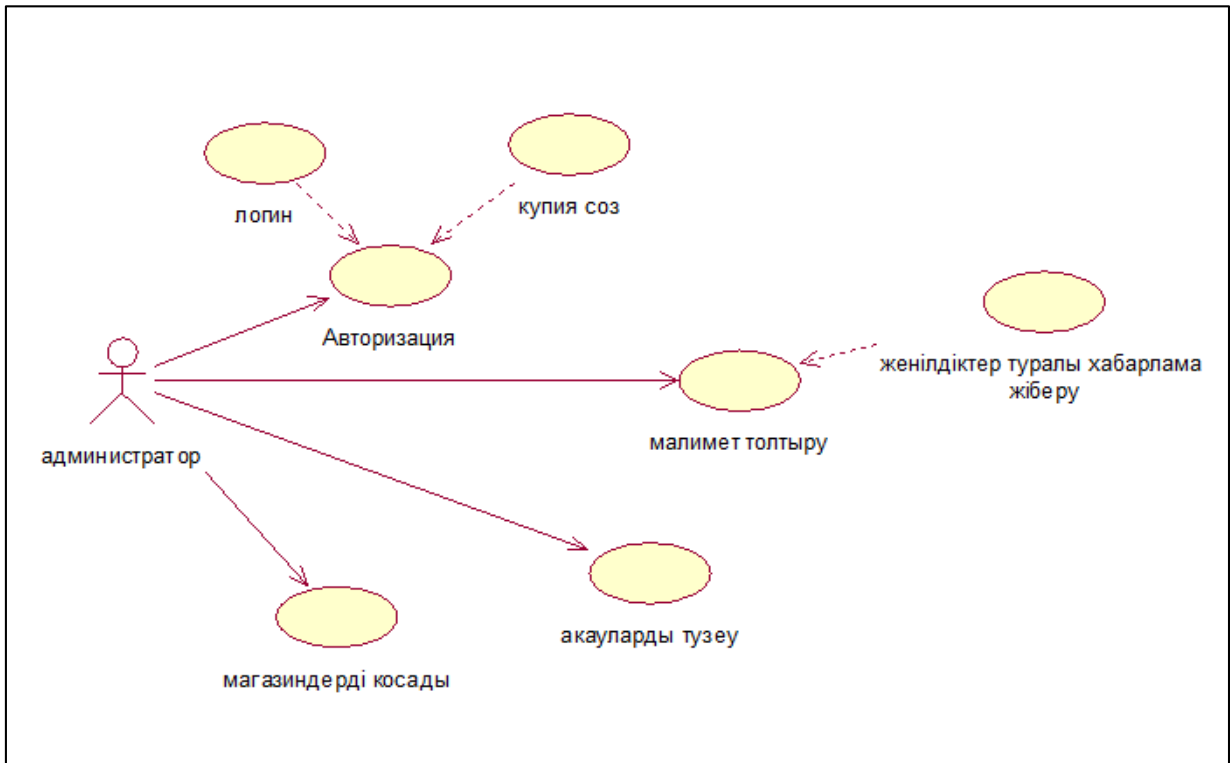
Прецедент – бұл пайдаланушыға нақты, өлшенетін және қалаған нәтиже ала алатын модельдік жүйенің (оның функционалдық бөлігінің) мүмкіндігі. Прецедент жекелеген сервистік жүйеге сәйкес келеді, оның қолданылу нұсқаларының бірін анықтайды және жүйемен өзара әрекеттесудің типтік әдісін сипаттайды. Пайдалану жағдайлар әдетте сыртқы жүйенің талаптарын анықтау үшін қолданылады. Бұл диаграммада қолданушының және администратордың қандай мүмкіндіктері бар екені көрсетілген.

Менің қосымшамның жобасы бойынша Rational Rose бағдарламасымен сызылған прецеденттер диаграммасы 3.2-суретте және 3.3-суретте көрсетілген.



**3.2-сурет – Прецеденттер диаграммасы**



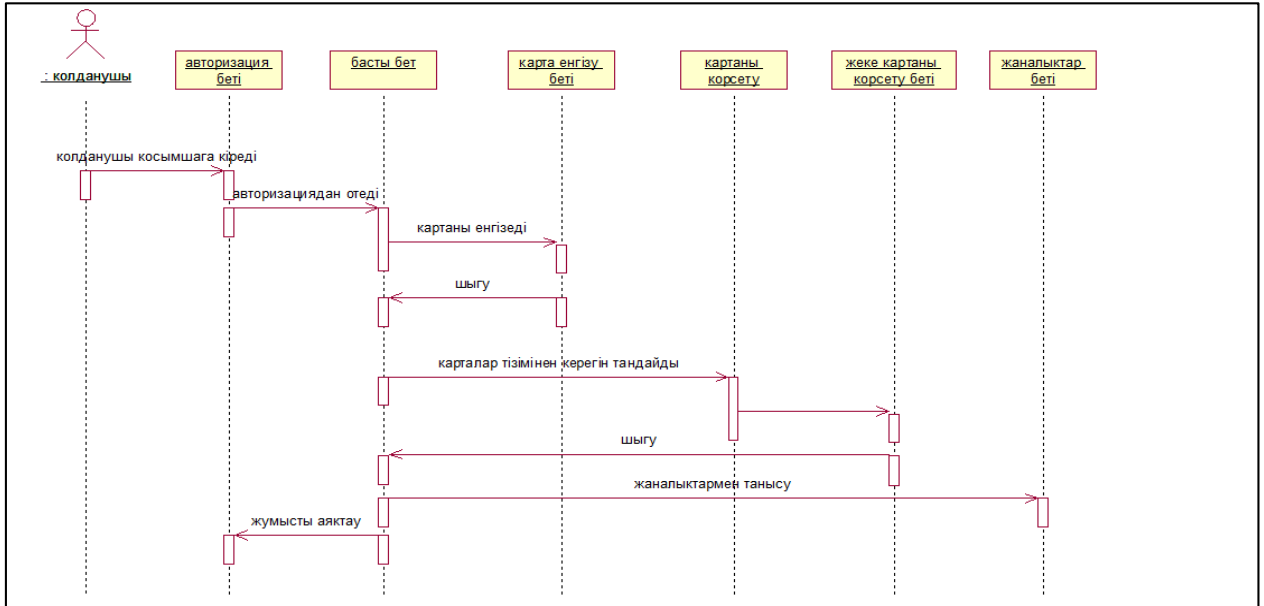


**3.3-сурет – Прецеденттер диаграммасы (жалғасы)**

Тізбек диаграммасы бұл белгілі бір уақыт кезіндегі объектілердің белгілі бір жиынтығы, белгілі бір объектінің өмірлік циклі (белгілі бір заңды тұлғаны жою-жасау қызметі) және АЖ субъектілерінің (таңбаларының) өзара іс-қимылы шеңберінде диаграмма. немесе белгілі бір прецедент (сұрау жіберу және жауап алу). UML-те қолданылған. Қолданушының өмірлік циклі көрсетілген(3.3-сурет).

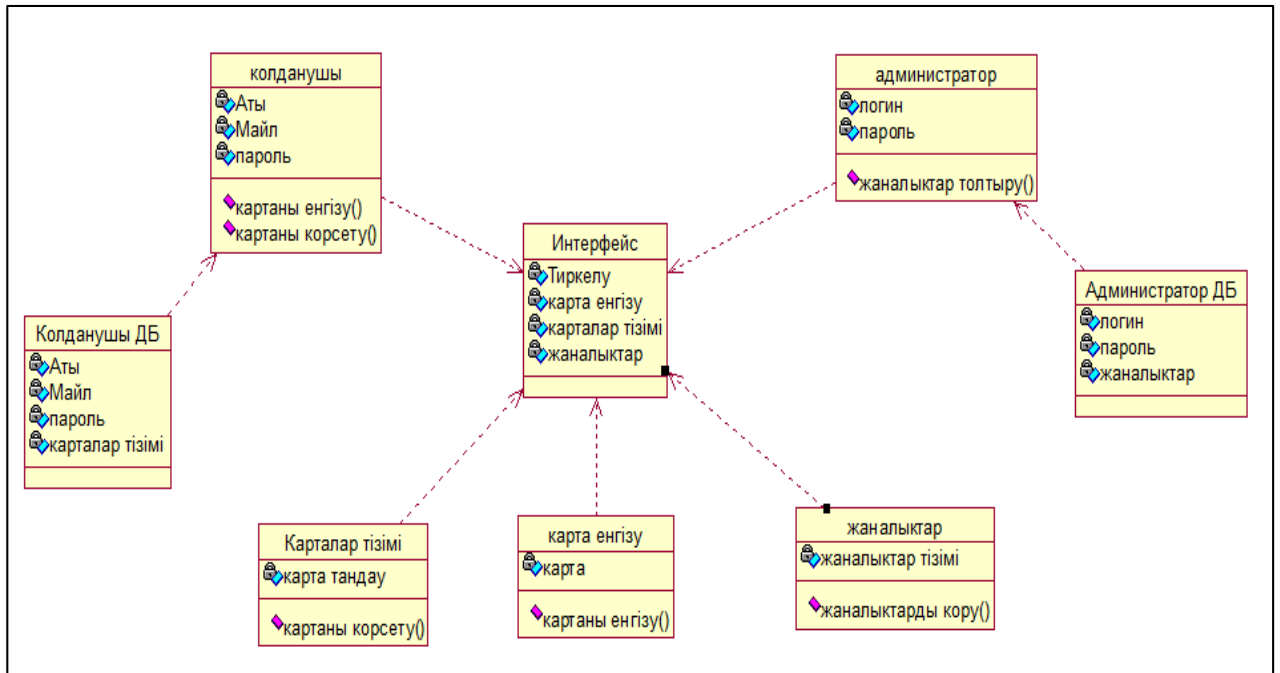
Тізбек диаграммасының негізгі элементтері – уақыттың өтуін көрсететін, объектінің белсенділігін көрсететін немесе белгілі бір функцияны орындаған тікбұрышты тіршілік сызықтары объектілер арасындағы сигналдар немесе хабар алмасуды көрсететін көрсеткілер.

Менің қосымшамның жобасы бойынша Rational Rose бағдарламасымен сызылған тізбек диаграммасы 3.4-суретте көрсетілген.



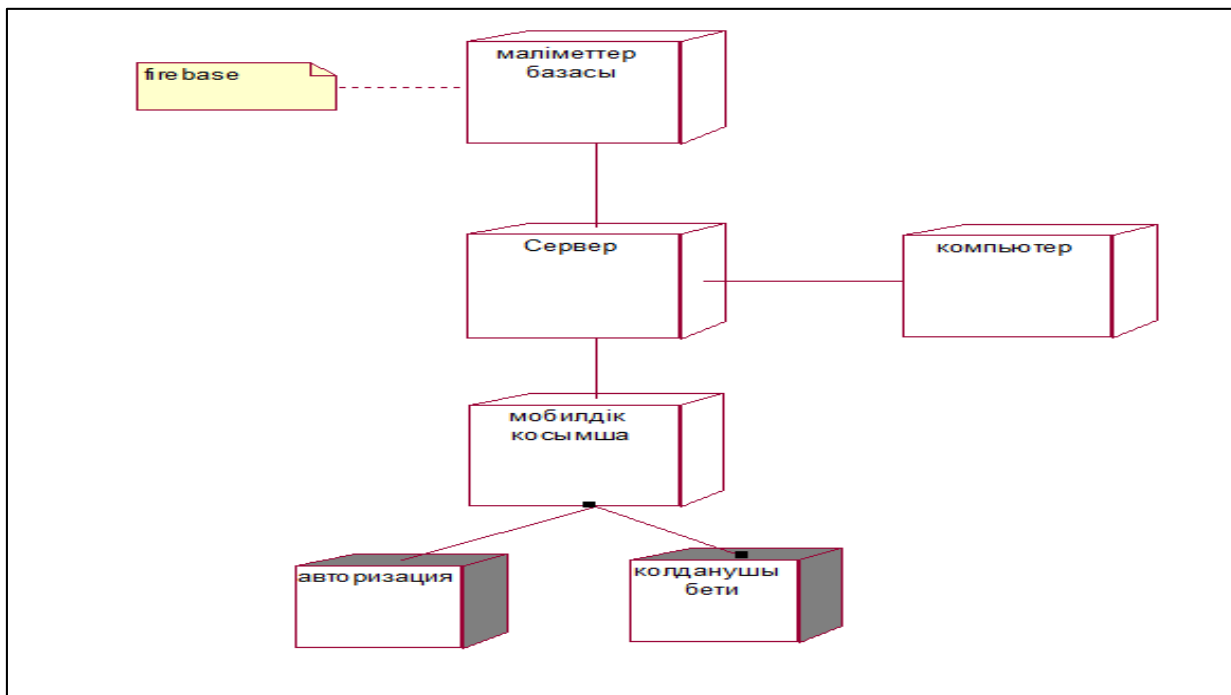
**3.4-сурет – Тізбек диаграммасы**

Класстік диаграмма – бұл UML модельдеу тілінің құрылымдық схемасы, ол жүйелік сыныптардың иерархиясының жалпы құрылымын, олардың кооперацияларын, атрибуттарын (өрістерін), әдістерін, интерфейстерін және олардың арасындағы қарым-қатынасты көрсетеді. Ол құжаттама мен көрнекілендіру үшін ғана емес, сонымен қатар жобалау үшін тікелей немесе кері технологиямен қолданылады. Менің қосымшамның жобасы бойынша Rational Rose бағдарламасымен сызылған класс диаграммасы 3.5-суретте көрсетілген



**3.5-сурет – Класс диаграммасы**

Орналастыру диаграммасы UML диаграммасының түрі болып табылады, ол жүйе архитектурасын көрсететін, оның ішінде аппараттық немесе бағдарламалық жасақтама жұмыс ортасы сияқты түйіндерді және оларды қосатын аралық бағдарламаны көрсетеді. Орналастыру диаграммаларын жүйенің қалай орын ауыстыратынын ұсынудың статикалық моделдеуі үшін қолданады. Бұл ұсыну таралуды, қоюды және физикалық жүйені тудыратын инсталляция бөліктерін қолдайды. Менің қосымшамның жобасы бойынша Rational Rose бағдарламасымен сызылған прецендеттер диаграммасы 3.6-суретте көрсетілген



**3.6-сурет – Орналастыру диаграммасы**

### **3.1 Есептің қойлымы**

Мобилді қосымша жасау. Негізінде мобильді құрылғылар әлдеқашан алдын-ала орнатылған бағдарламалармен сатылады. Ал қалғаның пайдаланушы арнайы қызметтер бойынша жүктеп алуға болады (ақылы және ақысыз): Apple AppStore, Google Play, Windows Phone дүкені және басқалар. Кейінірек Google Play болып табылатын Apple AppStore және Android Market сияқты алғашқы қосымшалар дүкендері 2008 жылы пайда болды.

Мобильдік қосымшалардың артықшылықтары:

- ақпаратқа қол жеткізу оңай;
- пайдаланушылардың өзара әрекеттесуі;
- пайдаланушы әдеттеріне негізделген ыңғайлы интерфейс;

- дербес режим пайдаланушы режимі;
- құрылғының мүмкіндіктерін пайдалану;
- Apple және Google Pay арқылы төлеңіз.

Бонустық карталарды сақтауға арналған Android платформасында құрылған мобилді бағдарлама. Ол сатып алушыларға дүкеннен берілген бонустық карталардың бәрін бір бағдарламада сақтап, сіздің әмияныңызда орын босатуға және әрдайым карточкаларды өзіңізбен бірге алып жүруге мүмкіндік береді.

Бағдарламаға сіз кірген сәтте өзіңіздің жеке парақшанызды ашасыз, келесі қадамда бонустық картаны алып, телефон камерасы арқылы сканерден өткізесіз, сонда камера бар коды автоматты түрде бағдарламаға аударады. Ол қаратңыз «Карталар» бөлімінде сақталып тұрады және одан кейін енгізілген карталарды бәрі сол бөлімде сақталып қалады. Дүкенге келген кезде кассада бар коды сіздің ұялы телефоныңыздан сканерден өткізеді және сіздің картанызға бонустар түседі. Ол бонустар туралы сіз бағдарламадан білсеңіз болады. Қолдану аймағы – дүкенде бонустық карталарды қолданатын қолданушыларға арналған.

Бастапқы бетте сізге жұмыстың басталуы туралы және егер сізде қолданушы жазбасы болса соны қолданып кіруіңізді немесе жаңадан тіркелуіңізді сұрайды. Қосымшаның басқы бетін 3.7-суреттен көре аласыз.



Все Ваши данные находятся  
под защитой самых передовых  
алгоритмов

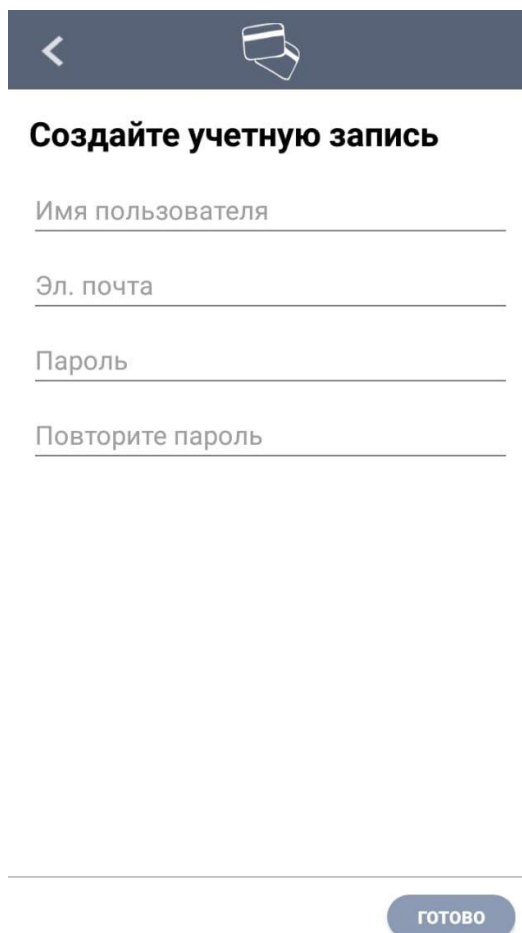


НАЧАТЬ

Уже есть учетная запись? **Войти**

### 3.7-сурет – Басқы бет

Кирген кезде тіркелуіңізді сурайды. Оған атыңызды, почтаңызды және кем дегенде 8 символдан тұратын құпия сөз енгізуіңізді сурайды. Құпия сөз дұрыс енгізілгенің және сіздің есіңізде қалу үшін құпия сөзді екі рет енгізуіңізді сұрайды. Сол кезде мәліметтер базасында сіздің қолданушыңыз пайда болады. Сіздің почтаңыздың дұрыс жазылғандығын тексереді. Яғни, @ белгісінен басталып, арасында нүктесі болғанын қадағалайды. Қосымшаның тіркелу бетін 3.8-суреттен көре аласыз.



**Создайте учетную запись**

Имя пользователя

Эл. почта

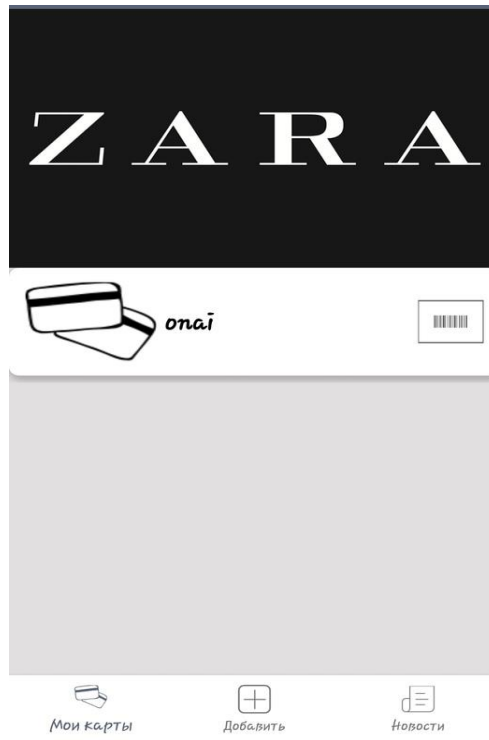
Пароль

Повторите пароль

ГОТОВО

### 3.8-сурет – Тіркелу беті

«Мои карты» бөлімінде біз өзімізде бар бонустық карталарды қосамыз, яғни карталармыздың тізімін көреміз. Ол карталардың тізімінен керегін тандап, дүкен желісінде bar коды көрсетіп, бізге берілу керек бонусты аламыз. Ол тізімге карталардың бәрін өзіңіз қолдану аясында қосып, толықтырып отырасыз. Картаның біреуін тандап, оны ашып оның сипаттамасын оқысаңыз болады. Карталар тізімі енгізу ретімен сақталады. Керек картаңызды тізімді қарап отырып табасыз. Қосымшаның карталар бөлімі бетін 3.9-суреттен көре аласыз.



**3.9-сурет – Карталар бөлімі**

«Новости» бөлімінде біз қосылған интернет желісі арқылы, әртүрлі дүкендердің жарнамасын, сол дүкендердегі акциялар туралы біле аламыз. Жарнамалар беті әрқашан жаңарып отырады. Ол Қазақстан бойынша бар үлкен және белгілі дүкендердің жарнамасын, онда бар акцияларды көрсетеді. Қосымшаның жаңалықтар бетін 3.9-суреттен көруге болады.



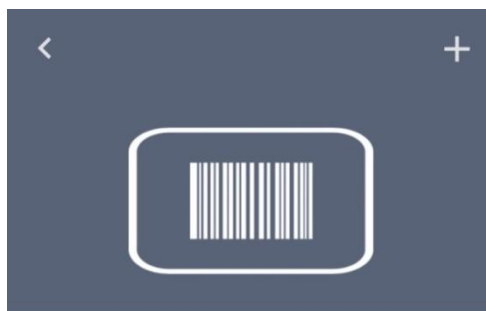
**3.9-сурет – Жаңалықтар беті**

«Добавить» батырмасын басып, біз картаны қоса аламыз. Яғни, сол кезде смартфондағы камера ашылады, сіз камераны өз картанызға дәлдесеңіз, ол штрих-коды сканерлеп, картаны «Мои карты» бөліміне және сіздің тіркелген логиніңіз бойынша деректер базасына сақтайды. Егер камера сіздің картанызды сканерден өткізе алмаған жағдай болса, оны өз қолыңызбен енгізеңіз болады. Қосымшаның карталарды енгізу бетін 3.10-суреттен көруге болады.



**3.10-сурет – Картаны енгізу беті**

. Картаны енгізгеннен кейін сіз ол картаның атауын және оған сипаттама беріп кете аласыз. Сипаттама беру міндетті емес болып табылады. Сипаттаманы сіз тізімнің ішінен бір картаны тандаған кезде және оны ашып караған кезде көре аласыз. Жоба қосымшасының картаны енгізу бетін 3.11-суреттен көрсеңіз болады.



**Добавить**

Название карты

9643908503307203087

Описание карты

  
Мои карты

  
Добавить

  
Новости

### 3.11-сурет – Картаны қосу беті



## ҚОРЫТЫНДЫ

Қосымшаның қаншалықты актуалдылығын, рейтингісін анықтап, қосымша пайдалы функциялар қызметін қосылды. Бұл мобильді қосымша барлық пайдаланушыларға арналған қарапайым қолжетімді болып табылады. Оны жасаудағы мақсатым пайдаланушылардың мүмкіншіліктерін арттыру, уақытын үнемдеу, ақпараттарды қолдетімді ету болып табылады.

Жоба заманауи бағдарламалық платформалар негізінде жасалды. Артықшылықтарын арттырып, кемшіліктерін жоюға тырыстым. Жобаны жүзеге асыру барысында көптеген жаңа технологиялар қолданылды.

Деректер қорымен байланыс жасау үшін бұлттық Firebase сервисін пайдаландылды. Ол бағдарламалаушыларға арналған деректер қорының шоғырланған ортасы. Ол да жаңа технология болып табылады. Деректерді енгізу, жою, іздеу үшін өте ыңғайлы және түсінікті. Деректер базасына үлкен деректермен жұмыс жасауға арналған NoSQL тілін пайдаланды.

Қосымшаны жасаудағы негізгі мақсат пайдаланушылар мүмкіндіктерін арттыру, бұл қосымшаның қаншалықты қажеттілігін, артықшылықтарын көрсету және түсіндіру болып табылады.

## ПАЙДАЛАНЫЛҒАН ӘДЕБИЕТТЕР ТІЗІМІ

- 1 Исакова С., Жемеров Д. Kotlin в действии / пер. с англ. Киселев А.Н. — М.: ДМК-Пресс, октябрь 2017 г., 402 стр.
- 2 Материал из Национальной библиотеки им. Н. Э. Баумана [Электронды ресурс] – Қатынау режимі: <https://ru.bmstu.wiki/Firebase>, тегін.
- 3 Makan K., Bown S.A. Firebase Essentials – Android Edition. – С.: CreateSpace, 2017. – 9 с.
- 4 Коматинени С., Маклин Д. Android 4 для профессионалов. Создание приложений для планшетных компьютеров и смартфонов. – М.: Вильямс, 2012. – 98 с.
- 5 Колисниченко Денис Программирование для Android. Самоучитель; БХВ-Петербург - Москва, 2012. - 272 с.
- 6 Бонни А.Н., Доуг С.Б. Flash 8. Создание и публикация интерактивных проектов. – Санкт-Петербург: НТ Пресс, 2008. – 592 с.
- 7 Ибраева Л.К., Сатимова Е.Г. Проектирование баз данных: Методическое указание к выполнению лабораторных работ. – Алматы: АИЭС, 2016.
- 8 Майер Рето Android 2. Программирование приложений для планшетных компьютеров и смартфонов; Эксмо - Москва, 2011. - 672 с.
- 9 Вигерс К.И. Разработка требований к программному обеспечению. – М.: Русская Редакция, 2004. – 9с.
- 10 Дейтел П., Дейтел Х., Дейтел Э., Моргано М. Android для программистов: создаем приложения. – СПб.: Питер, 2013. – С. 349 – 350.

**А қосымшасы**  
(міндетті)

**Бағдарламаның мәтіні**

1. *MainActivity.java* Бастапқы бағдарлама беті

```
package kz.app.excard.ezcards.Pages;
import android.support.annotation.NonNull;
import android.support.design.widget.BottomNavigationView;
import android.support.v4.app.FragmentTransaction;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.Toolbar;
import android.view.MenuItem;
import kz.app.excard.ezcards.Fragments.AddCardFragment;
import kz.app.excard.ezcards.Fragments.FeedsFragment;
import kz.app.excard.ezcards.Fragments.MyCardsFragment;
import kz.app.excard.ezcards.R;
public class MainActivity extends AppCompatActivity {
    private BottomNavigationView navigationView;
    private FragmentTransaction fragmentTransaction;
    private Toolbar toolbar;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        initView();
    }
    private void initView() {
        navigationView = findViewById(R.id.bottom_navigation);
        toolbar = findViewById(R.id.toolbar);
        navigationView.setOnNavigationItemSelectedListener(new
BottomNavigationView.OnNavigationItemSelectedListener() {
        @Override
        public boolean onNavigationItemSelected(@NonNull MenuItem menuItem)
{
            fragmentTransaction = getSupportFragmentManager().beginTransaction();
            switch (menuItem.getItemId()) {
                case R.id.menu_cards:
                    MyCardsFragment myCardsFragment = new MyCardsFragment();
                    fragmentTransaction.replace(R.id.main_frame, myCardsFragment,
MyCardsFragment.TAG);
```

## А Қосымшасының жалғасы

```
        fragmentTransaction.addToBackStack(MyCardsFragment.TAG);
        break;
    case R.id.menu_add:
        AddCardFragment addCardFragment = new AddCardFragment();
        fragmentTransaction.replace(R.id.main_frame, addCardFragment,
AddCardFragment.TAG);
        fragmentTransaction.addToBackStack(AddCardFragment.TAG);
        break;
    case R.id.menu_news:
        FeedsFragment feedsFragment = new FeedsFragment();
        fragmentTransaction.replace(R.id.main_frame, feedsFragment,
FeedsFragment.TAG);
        fragmentTransaction.addToBackStack(FeedsFragment.TAG);
        break;
    }
    fragmentTransaction.commit();
    return true;
}
});
navigationView.setSelectedItemId(R.id.menu_cards);
}
public void setPage(int page){
    navigationView.setSelectedItemId(page);
}
}
```

### 2 AuthorizationActivity.java Авторизация беті

```
package kz.app.excard.ezcards.Pages;
import android.content.Intent;
import android.support.v4.app.Fragment;
import android.support.v4.app.FragmentTransaction;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
import kz.app.excard.ezcards.Fragments.StartFragment;
import kz.app.excard.ezcards.R;
public class AuthorizationActivity extends AppCompatActivity {
```

## А Қосымшасының жалғасы

```
private FragmentTransaction fragmentTransaction;
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_authorization);
    openLogin();
}
private void openLogin() {
    fragmentTransaction = getSupportFragmentManager().beginTransaction();

    StartFragment startFragment = new StartFragment();
    fragmentTransaction.add(R.id.auth_frame, startFragment, StartFragment.TAG);
    fragmentTransaction.addToBackStack(StartFragment.TAG);
    fragmentTransaction.commit();
}
@Override
public void onBackPressed() {
    int count = getSupportFragmentManager().getBackStackEntryCount();
    if(count!=1){
        super.onBackPressed();
    }
}
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    try {
        for (Fragment fragment : getSupportFragmentManager().getFragments()) {
            fragment.onActivityResult(requestCode, resultCode, data);
            Log.d("Activity", "ON RESULT CALLED");
        }
    } catch (Exception e) {
        Log.d("ERROR", e.toString());
    }
}
}
```

3 *Card.java* Карта енгізу беті

```
package kz.app.excard.ezcards.Entities;
import android.os.Parcel;
```

## А Қосымшасының жалғасы

```
import android.os.Parcelable;
public class Card implements Parcelable{
    private String name;
    private String number;
    private String desc;
    public Card() {
    }
    public Card(String name, String number, String desc) {
        this.name = name;
        this.number = number;
        this.desc = desc;
    }
    protected Card(Parcel in) {
        name = in.readString();
        number = in.readString();
        desc = in.readString();
    }
    public static final Creator<Card> CREATOR = new Creator<Card>() {
        @Override
        public Card createFromParcel(Parcel in) {
            return new Card(in);
        }
        @Override
        public Card[] newArray(int size) {
            return new Card[size];
        }
    };
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
    public String getNumber() {
        return number;
    }
    public void setNumber(String number) {
        this.number = number;
    }
    public String getDesc() {
        return desc;
    }
}
```

## А Қосымшасының жалғасы

```
}  
public void setDesc(String desc) {  
    this.desc = desc;  
}  
@Override  
public int describeContents() {  
    return 0;  
}  
@Override  
public void writeToParcel(Parcel dest, int flags) {  
    dest.writeString(name);  
    dest.writeString(number);  
    dest.writeString(desc);  
}  
}
```

4 *User.java* Қолданушы беті

```
package kz.app.excard.ezcards.Entities;  
public class User {  
    private String name;  
    private String email;  
    private String userId;  
    private String password;  
    public User(String name, String email, String userId) {  
        this.name = name;  
        this.email = email;  
        this.userId = userId;  
    }  
    public User() {  
    }  
    public String getName() {  
        return name;  
    }  
    public void setName(String name) {  
        this.name = name;  
    }  
    public String getEmail() {  
        return email;  
    }  
}
```

## А Қосымшасының жалғасы

```
}  
public void setEmail(String email) {  
  
    this.email = email;  
}  
public String getPassword() {  
    return password;  
}  
public void setPassword(String password) {  
  
    this.password = password;  
}  
public String getUserId() {  
    return userId;  
}  
public void setUserId(String userId) {  
    this.userId = userId;  
}  
}
```

### 5 *AddCardFrgment.java* Картаны қосу фрагменті

```
package kz.app.excard.ezcards.Fragments;  
import android.Manifest;  
import android.content.pm.PackageManager;  
import android.os.Bundle;  
import android.support.annotation.NonNull;  
import android.support.v4.app.ActivityCompat;  
import android.support.v4.app.Fragment;  
import android.support.v4.app.FragmentTransaction;  
import android.support.v7.widget.Toolbar;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.view.ViewGroup;  
import android.view.animation.Animation;  
import android.view.animation.AnimationUtils;  
import android.widget.Button;  
import android.widget.ImageView;
```

## А Қосымшасының жалғасы



```

import android.widget.TextView;
import com.google.zxing.Result;
import java.util.Objects;
import kz.app.excard.ezcards.Pages.MainActivity;
import kz.app.excard.ezcards.R;
import kz.app.excard.ezcards.Utills.Constants;
import kz.app.excard.ezcards.Utills.Utills;
import me.dm7.barcodescanner.zxing.ZXingScannerView;
public class AddCardFragment extends Fragment implements
ZXingScannerView.ResultHandler{
    public static final String TAG = Constants.ADDCARDFRAGMENT;

    private static final int REQUEST_CAMERA_PERMISSION = 12;
private ImageView cardView;
private Button addByArmButton;
private ZXingScannerView mScannerView;
private Toolbar toolbar;
private TextView toolbarTitle;
private FragmentTransaction fragmentTransaction;
    @Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
        Bundle savedInstanceState) {
        View view = inflater.inflate(R.layout.fragment_add_card, container, false);
        setHasOptionsMenu(true);
        initView(view);
        return view;
    }
    @Override
public void onResume() {
        super.onResume();
        if(hasPermission()){
            mScannerView.setResultHandler(this);
            mScannerView.startCamera();
        }else {
            ActivityCompat.requestPermissions(getActivity(), new String[]
{Manifest.permission.CAMERA}, REQUEST_CAMERA_PERMISSION);
        }
    }
    @Override
public void onPause() {
        super.onPause();

```

**А Қосымшасының жалғасы**

```

        mScannerView.stopCamera();
    }
    private void initView(View view) {
        toolbar = getActivity().findViewById(R.id.toolbar);
        toolbarTitle = toolbar.findViewById(R.id.toolbar_title);
        Utils.initToolbar((MainActivity) Objects.requireNonNull(getActivity()),
R.drawable.ic_back,
            toolbar, false, toolbarTitle, "");
        toolbar = getActivity().findViewById(R.id.toolbar);
        addByArmButton= (Button) view.findViewById(R.id.fsc_add_by_arm_button);
        cardView= (ImageView) view.findViewById(R.id.fsc_card_image);
        mScannerView= (ZXingScannerView) view.findViewById(R.id.zxscan);
        toolbar.getMenu().clear();
        animateView();
        addByArmButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Bundle b = new Bundle();
                b.putBoolean(Constants.ADDRESS, false);
                fragmentTransaction = getFragmentManager().beginTransaction();
                AddByArmFragment addCardByArmFragment = new
AddByArmFragment();
                addCardByArmFragment.setArguments(b);
                fragmentTransaction.replace(R.id.main_frame, addCardByArmFragment,
AddByArmFragment.TAG);
                fragmentTransaction.addToBackStack(AddByArmFragment.TAG);
                fragmentTransaction.commit();
            }
        });
    }
    private void animateView() {
        Animation shake = AnimationUtils.loadAnimation(getContext(), R.anim.shake);
        cardView.startAnimation(shake);
    }
    @Override
    public void handleResult(Result result) {
        Bundle b = new Bundle();
        b.putBoolean(Constants.ADDRESS, true);
        b.putString(Constants.CODE, result.getText());
        fragmentTransaction = getFragmentManager().beginTransaction();
        AddByArmFragment addCardByArmFragment = new AddByArmFragment();

```

**А Қосымшасының жалғасы**

```

        addCardByArmFragment.setArguments(b);
        fragmentTransaction.replace(R.id.main_frame, addCardByArmFragment,
AddByArmFragment.TAG);
        fragmentTransaction.addToBackStack(AddByArmFragment.TAG);
        fragmentTransaction.commit();
    }

    @Override
    public void onRequestPermissionsResult(int requestCode, @NonNull String[]
permissions, @NonNull int[] grantResults) {
        super.onRequestPermissionsResult(requestCode, permissions, grantResults);
        switch (requestCode){
            case REQUEST_CAMERA_PERMISSION:
                if (grantResults.length > 0 && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {

                    mScannerView.setResultHandler(this);
                    mScannerView.startCamera();
                } else {
                    requestPermissions(new String[]{Manifest.permission.CAMERA},
REQUEST_CAMERA_PERMISSION);
                }
                break;
            default:
                requestPermissions(new
String[]{Manifest.permission.CAMERA},REQUEST_CAMERA_PERMISSION);
                break;
        }
    }
    private boolean hasPermission(){
        String permission = Manifest.permission.CAMERA;
        int res = getContext().checkCallingOrSelfPermission(permission);
        return (res == PackageManager.PERMISSION_GRANTED);
    }
}

```

6 *BarcodeFragment.java* Бар кодқа айналдыру беті

```

package kz.app.excard.ezcards.Fragments;
import android.graphics.Bitmap;

```

**А Қосымшасының жалғасы**

```

import android.graphics.Color;
import android.os.Bundle;
import android.support.v4.app.Fragment;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;
import com.google.zxing.BarcodeFormat;
import com.google.zxing.WriterException;
import com.google.zxing.common.BitMatrix;
import com.google.zxing.oned.Code128Writer;
import kz.app.excard.ezcards.Entities.Card;
import kz.app.excard.ezcards.R;
import kz.app.excard.ezcards.Utils.Constants;
public class BarcodeFragment extends Fragment {
    public static final String TAG = Constants.BARCODERAGMENT;
    private static final String CARD = "card";
    private Card card;
    private ImageView barcode;
    private TextView nameText, numberText;
    private Bitmap mBitmap;
    public static BarcodeFragment newInstance(Card card) {
        BarcodeFragment fragment = new BarcodeFragment();
        Bundle args = new Bundle();
        args.putParcelable(CARD, card);
        fragment.setArguments(args);
        return fragment;
    }
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        if (getArguments() != null) {
            card = getArguments().getParcelable(CARD);
        }
    }
    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
        Bundle savedInstanceState) {
        View view = inflater.inflate(R.layout.fragment_barcode, container, false);

```

**А Қосымшасының жалғасы**

```

    initView(view);
    return view;
}
private void initView(View view) {
    barcode = view.findViewById(R.id.fbc_imageview);
    nameText = view.findViewById(R.id.fbc_title);
    numberText = view.findViewById(R.id.fbc_number);
    nameText.setText(card.getName());

    numberText.setText(card.getNumber());
    generateBarCode(card.getNumber(), barcode);
}
public void generateBarCode(String data, ImageView imageView){
    com.google.zxing.Writer c9 = new Code128Writer();
    try {
        BitMatrix bm = c9.encode(data, BarcodeFormat.CODE_128, 800, 350);

        mBitmap = Bitmap.createBitmap(800, 350, Bitmap.Config.ARGB_8888);
        for (int i = 0; i < 800; i++) {
            for (int j = 0; j < 350; j++) {
                mBitmap.setPixel(i, j, bm.get(i, j) ? Color.BLACK : Color.WHITE);
            }
        }
    } catch (WriterException e) {
        e.printStackTrace();
    }
    if (mBitmap != null) {
        imageView.setImageBitmap(mBitmap);
    } else {
        Toast.makeText(getActivity(), "Null", Toast.LENGTH_LONG).show();
    }
}
}
}

```

Қ.И. Сәтбаев атындағы Қазақ ұлттық техникалық зерттеу университеті

Мамандығы 5В060200 – Ақпараттану

Студент Байғалиева Гүлайым

**Тақырыбы:** «Андройд платформасында бонустық карталарды сақтауға арналған қосымшасы»

### ҒЫЛЫМИ ЖЕТЕКШІНІҢ СЫН-ШІКІРІ

Дипломдық жобала күнделікті кездесетін және адам өміріне тереңінен еніп жатқан компьютерлік техника, ақпараттық жүйе мен желінің дамуы ақпарат көлемінің жылдам өңделуіне жақсы әсер етуде. Қазіргі таңда электронды басқару және бақылау жүйелерінің көптеген түрлері қолдануда.

Дипломдық жоба бонустық карталарды қолданылатын адамдар үшін өте тиімді және қолайлы қосымша болып табылады.

Дипломдық жұмыс кіріспеден, үш бөлімнен, қолданылған әдебиеттер тізімінен, қорытындыдан тұрады. Бірінші бөлімде мәселенің қазіргі замандағы жағдайына шолу және оны талдау жасалынды. Екінші бөлімде программалау тілдерін пайдалану ерекшеліктері жобаланған және қосымшаға қатысты диаграммалар көрсетілген. Үшінші бөлімде бонустық карталарды сақтауға арналған қосымшаны жобалау мен жасау дайындалған.

Дипломдық жобаны қарастыра келгенде жақсы жақтарын айта кетуге болады. Жұмыс материалы жоспарлы түрде, қойылған талаптарға сай, тақырып төңірегінде жазылған. Студенттің жұмысты рәсімдеудегі біліктілігін, талдаушылық қабілетін көруге болады. Осы жұмыста негізгі өзекті мәселелерді шеше отырып, зерттеу жұмысында оңтайлы нәтижелер алып, жұмысты қойылған талаптарға сай орындап шықты.

Қортындылай келгенде, осы зерттеу жұмысын орындау барысында Байғалиева Гүлайым Ерланқызының айтарлықтай жақсы еңбектенгені көрінеді.

Жоба жетекшісі ретінде бұл дипломдық жобаны өз деңгейіне сәйкес деп есептей отырып Байғалиева Гүлайым «Андройд платформасында бонустық карталарды сақтауға арналған қосымшасы» атты тақырыптағы дипломдық жұмысын қорғауға жіберуге және 5В060200– «Ақпараттану» мамандығы бойынша «Жаратылыстану бакалавры» академиялық дәрежесін тағайындауға болады деп есептеймін.

ҒЫЛЫМИ ЖЕТЕКШІ  
«Программалық инженерия»

Р. С. Алғожаева

Кафедрасының сениор-лекторы «17» 05 2019 жыл