

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Сәтбаев университеті

Ақпараттық және телекоммуникациялық технологиялар институты

Киберқауіпсіздік, ақпаратты өңдеу және сақтау кафедрасы

Темірлан М.Т.

«Өнім тапсырысы мобильді қосымшасын жасау »

ДИПЛОМДЫҚ ЖҰМЫС

5B070300 – «Ақпараттық жүйелер» мамандығы

Алматы 2019

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Сәтбаев университеті

Ақпараттық және телекоммуникациялық технологиялар институты

Киберқауіпсіздік, ақпаратты өңдеу және сақтау кафедрасы

ҚОРҒАУҒА ЖІБЕРІЛДІ
КҚФЖС кафедра меңгерушісі,
канд.тех.наук, ассистент
профессор

 Н.А Сейлова
« 15 » мамыр 2019 ж.

ДИПЛОМДЫҚ ЖҰМЫС

Тақырыбы «Өнім тапсырысы»мобильді қосымшасын жасау

5B070300 – «Ақпараттық жүйелер» мамандығы бойынша

Орындаған

Темірлан М.Т

Пікір беруші:

Front-end разработчик

Ж.Ш Тілеубаев



« 06 » мамыр 2019ж.

Ғылыми жетекші :

Техн.ғыл.канд, ассоц проф

 Б. И. Жумағалиев

« 13 » мамыр 2019ж.

Алматы 2019


Ақпараттық және телекоммуникациялық технологиялар институты

Киберқауіпсіздік, ақпаратты өңдеу және сақтау кафедрасы

5B070300 – «Ақпараттық жүйелер» мамандығы

БЕКІТЕМІН

КҚФЖС кафедра меңгерушісі,
канд.тех.наук, ассистент
профессор

 Н.А.Сейлова
« 15 » мамыр 2019 ж.

**Дипломдық жұмысты орындауға
ТАПСЫРМА**

Білім алушы: Темірлан Мадитемірұлы

Тақырыбы: «Өнім тапсырысы мобильді қосымшасын жасау»

Университет Ректоры: 2017 жылғы «20» қазан № 1571-б бұйрығымен
бекітілген

Аяқталған жұмысты тапсыру мерзімі: 2019 жылғы «16» мамыр

Дипломдық жұмыстың бастапқы берілістері: диплом алдындағы
практикалық жұмыс қорытындысы, тақырып бойынша әдебиеттерге шолу
нәтижелері, теориялық мәліметтердің жиыны

Дипломдық жұмыста қарастырылатын мәселелер тізімі:

а) қойылған мәселенің қазіргі жағдайын пайымдау

ә) ақпараттық қамтаманы құру

б) программалық қамтаманы құру

Сызбалық материалдар тізімі: Power Point бағдарламасындағы слайдтар

Сызба материалдар: 15 слайдпен көрсетілген



Ұсынылатын негізгі әдебиет: 24 атау





Дипломдық жұмысты дайындау
КЕСТЕСІ

Бөлімдер атауы, қарастырылатын мәселелер тізімі	Ғылыми жетекші мен кеңесшілерге көрсету мерзімдері	Ескерту
Мәселенің қазіргі жағдайына шолу және оны талдау	11.01.2019 - 08.03.2019	
Ақпараттық қамтаманы құру	06.02.2019 - 10.03.2019	
Программалық қамтаманы құру	12.03.2019 - 28.04.2019	

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, ассистент	15.05.19	
Программалық қамтама	М.Б. Бауыржан тьютор, тех. ғыл магистрі	8.05.19ж	

Ғылыми жетекші  Б.И.Жұмағалиев

Тапсырманы орындауға алған білім алушы  М.Т.Темірлан

Күні

« 8 » қаңтар 2019ж

Ғылыми жетекшінің пікірі

Дипломдық жоба

Темірлан Мади Темірланұлы

5B070300 – Ақпараттық жүйелер

Тақырыбы: «Өнім тапсырысы» мобильді қосымшасын жасау

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

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

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

Кемшілік ретінде кейбір шағын стилистикалық қателерді атап кетуге болады.

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

Ғылыми жетекші

Ассоц.проф, т.ғ.к.



Жумағалиев Б.И.

«15» 05 2019 ж.

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Сәтбаев университеті

Дипломдық жұмысқа
СЫН-ПІКІР

Білім алушы : Темірлан Мәди Темірланұлы

Дипломдық жұмыстың тақырыбы: «Өнім тапсырысы»мобильді қосымшасын жасау

Бұл дипломдық жұмыс – Кәсіпорын қызметін талдау міндеттер класына жатады және кәсіпорынның мақсаттарына қол жеткізу.

Дипломдық жұмыста түсіндірме хат кіріспеден, қорытынды, әдебиеттер тізімінен тұрады. Дипломдық жұмыс барлығы 35 беттен тұрады.

Кіріспеде мәселенің маңыздылығы, дипломдық жұмыстың мақсаты және қолданылған құралдар қысқаша келтірілген. Бірінші бөлімде қазіргі кездегі кәсіптік бәсекелестілік қарастырылды. Екінші бөлімде Android платформасының архитектірасы құрастырылды. Үшінші бөлімде программаның сипатамасы ,жалпы мәліметтер,функционалдық құрылымы,логикалық құрылымның баяндалуы, қолданылған техникалық құрал-жабдықтар, кіріс және шығыс мәліметтер жайында толық қарастырылып жазылған. Дипломдық жұмыста елеулі кемшіліктер анықталмады. Бірақ, әдебиетке деген сілтемелер аз. Алайда, бұл кемшіліктер орындалатын жұмыстардың жалпы деңгейін төмендетпейді.

Темірлан Мәдидің дипломдық жұмысты орындау барысында өз беттілік көрсетіп, әдебиеттермен жұмыс істеген. Дипломдық жұмыс Ақпараттық жүйелер мамандығының бітіру жұмыстарына қойылатын талаптарға сәйкес.

Қорытындылай келе, Темірлан Мәдидің дипломдық жұмысына

« 43 балл » деген баға беруге және оның орындаушысына 5B070300 – «Ақпараттық жүйелер» мамандығы бойынша бакалавр біліктілігін беруге болады деп есептеймін.

Сын-пікір беруші:

Front-end разработчик

Тілеубаев

Тілеубаев Ж.ІІІ



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: Темірлан Мади

Название: Өнім тапсырысы» мобильді қосымшасын жасау

Координатор: Биржан Жумағалиев

Коэффициент подобия 1: 3,1

Коэффициент подобия 2: 0,7

Тревога: 9

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

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



Обоснование:

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

Дата 16.05.19

Подпись заведующего кафедрой /

начальника структурного подразделения

Окончательное решение в отношении допуска к защите, включая обоснование:

Воржужа нейберисск

Дата 16.05.1992

Подпись заведующего кафедрой

начальника структурного подразделения

И.И.И.И.И.

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: Темірлан Мади

Название: Өнім тапсырысы» мобильді қосымшасын жасау

Координатор: Биржан Жумағалиев

Коэффициент подобия 1:3,1

Коэффициент подобия 2:0,7

Тревога:9

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:

.....

.....

.....


.....

.....

.....

15.05.19

Дата

Рук. ДР 

Подпись Научного руководителя

Краткий отчет



Университет:	Satbayev University
Название:	Өнім тапсырысы» мобильді қосымшасын жасау
Автор:	Темірлан Мади
Координатор:	Биржан Жумағалиев
Дата отчета:	2019-05-13 10:37:44
Коэффициент подоби́я № 1:	3,1%
Коэффициент подоби́я № 2:	0,7%
Длина фразы для коэффициента подоби́я № 2:	25
Количество слов:	4 747
Число знаков:	40 602
Адреса пропущенные при проверке:	
Количество завершенных проверок:	55



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 9

>>

Самые длинные фрагменты, определенные, как подобные

№	Название, имя автора или адрес гиперссылки (Название базы данных)	Автор	Количество одинаковых слов
1	Жамписова Гулзада.docx Satbayev University (ИИИТТ)	Жамписова Гулзада	31
2	Лигай Э.docx Kazakh University of Economics, Finance and International Trade (KUEFIT) (ФАКУЛЬТЕТ ПОСЛЕВУЗОВСКОГО ОБРАЗОВАНИЯ)	Лигай Э.	12
3	URL_ http://www.zakazdiplom.ru/9808.html		12
4	URL_ http://www.zakazdiplom.ru/9808.html		11
5	URL_ http://www.zakazdiplom.ru/9808.html		11
6	Шамшиден М.Ж.-дипломдық жұмыс.doc Kazakh University of Economics, Finance and International Trade (KUEFIT) (ФАКУЛЬТЕТ "ЭКОНОМИКА")	Шамшиден М.Ж.	10
7	Документ из базы НЭУ 51d4f368-9480-4bfc-b237-0731c0a80d1d.docx NARXOZ (NEU) (Information Technology Center)	na	10
8	URL_ https://studwork.org/shop/18250-uchet-raspredelenie-i-analiz-rashodov-po-obslujivaniyu-proizvodstva-i-upravleniyu-organizaciy		9
9	Документ из базы НЭУ 51d4f368-9480-4bfc-b237-0731c0a80d1d.docx NARXOZ (NEU) (Information Technology Center)	na	9

>>

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	Жамписова Гулзада.docx Satbayev University (ИИИТТ)	Жамписова Гулзада	39 (2)

>>

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 51d4f368-9480-4bfc-b237-0731c0a80d1d.docx <i>NARXOZ (NEU) (Information Technology Center)</i>	на	19 (2)
2	Лигай Э.docx <i>Kazakh University of Economics, Finance and International Trade (KUEFIT) (ФАКУЛЬТЕТ ПОСЛЕВУЗОВСКОГО ОБРАЗОВАНИЯ)</i>	Лигай Э.	12 (1)
3	Шамшиден М.Ж.-дипломдық жұмыс.doc <i>Kazakh University of Economics, Finance and International Trade (KUEFIT) (ФАКУЛЬТЕТ "ЭКОНОМИКА")</i>	Шамшиден М.Ж.	10 (1)

>>

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://www.zakazdiplom.ru/9808.html	39 (4)
2	URL_ https://studwork.org/shop/18250-uchet-raspredelenie-i-analiz-rashodov-po-obslujivaniyu-proizvodstva-i-upravleniyu-organizaciy	29 (4)

АНДАТПА

Дипломдық жұмыстың тақырыбы: "Өнім тапсырысы мобильді қосымшасын құру".

Адам қызметінің барлық салаларында ақпараттық технологияларды енгізу тиімділіктің айқын өсуімен ғана емес, сондай-ақ кәсіпорындарда кәсіпкерлік және өндірістік процестерді ұйымдастыру үшін ескірген технологиялармен байланысты ұзақ уақытқа созылатын мәселелерді да атап өтті.

Бұл мәселе барлық қызмет түрлеріне, меншік нысанына және саласына тәуелсіз барлық компанияларға тән. Егер жобалау, техникалық және технологиялық құжаттамалардың көп көлемі өнеркәсіптік кәсіпорындар мен ғылыми-зерттеу институттарына тән болса, ұйымдық-әкімшілік құжаттаманың, бухгалтерлік есептердің және персоналдың жазбаларының болуы кез келген кәсіпорын үшін тән. Осы құжаттаманың бір бөлігі «компьютер алдындағы дәуірден» сақталған, басқа бөлігі белгілі бір уақыт кезеңінде құжаттарды мұрағаттарда сақтауды көздейтін бухгалтерлік есеп пен есептілік жүйесінің нәтижесінде пайда болды.

Дипломдық жұмыстың мақсаты: «Өнім тапсырысы мобильді қосымшасын» құру.

Қойылған мақсатқа жету үшін келесі міндеттер орындалуы тиіс:

- «өнім тапсырысы» саласын қарастыру;
- «өнім тапсырысы» негізін зерттеу;
- «өнім тапсырысы» мобильдік қосымшасын құру.

Осы мақсатқа сәйкес техникалық құралдарды басқару мобильдік қосымшасы жасалуда.

Зерттеу объектісі. Android тілінде жазылған өнім тапсырысы мобильді қосымшасы.

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

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

АННОТАЦИЯ

Тема дипломной работы: «Разработка мобильного приложения заказ продуктов».

Внедрение информационных технологий во все сферы человеческой деятельности не только явный рост эффективности, но и долгосрочные проблемы, связанные с устаревшими технологиями организации бизнеса и производственных процессов на предприятиях.

Эта проблема характерна для всех предприятий, независимо от формы собственности и отрасли. Наличие организационно-распорядительной документации, бухгалтерского учета и кадрового учета характерно для любого предприятия, если большая часть проектной, технической и технологической документации характерна для промышленных предприятий и научно-исследовательских институтов. Часть этой документации была сохранена с «компьютерной эры», а другая часть представляет собой систему учета и отчетности, которая предполагает хранение документов в архивах в течение определенного периода времени.

Целью дипломной работы является «Разработка мобильного приложения заказ продуктов».

Для достижения этой цели необходимо решить следующие задачи:

- рассмотрение области «заказ продукции»;
- изучение основ «заказа продукции»;
- разработка мобильного приложения «заказ продуктов».

С этой целью разрабатывается мобильное приложение заказ продуктов.

Объект исследования. Бизнес операции заказа продуктов с возможностью реализации в мобильном приложении для платформы android.

Степень анализа проблемы. При изучении темы дипломного проекта, отечественных и зарубежных научных работ и учебников использовались научные статьи в электронных изданиях.

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

ANNOTATION

The theme of the thesis: "the development of a mobile application order products."

The introduction of information technology in all spheres of human activity is not only a clear increase in efficiency, but also long-term problems associated with obsolete technologies of business organization and production processes in enterprises.

This problem is typical for all enterprises, regardless of the form of ownership and industry. The presence of organizational and administrative documentation, accounting and personnel records is typical for any enterprise, if most of the design, technical and technological documentation is typical for industrial enterprises and research institutes. A part of this documentation was saved from the "computer era", and another part is an accounting system that preserves documents in archives for a certain period of time.

The aim of the thesis is to "Develop a mobile application ordering products."

To achieve this goal it is necessary to solve the following tasks:

- consideration of the field "product order";
- learning the basics of "product order";
- development of the mobile application "product order".

For this purpose, a mobile application is being developed to order products.

Object of study. Mobile application order products written on android.

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

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

МАЗМҰНЫ

КІРІСПЕ	9
1 Пәндік саланы талдау	10
1.1 Қазіргі кездегі кәсіптік бәсекелеске шолу	11
1.2 Мәселелерді жою үшін техникалық қолдану техникасын қолдану мақсаты мен міндеті	12
1.3 Есептің қойылымы	14
2 ANDROID ПЛАТФОРМАСЫНЫҢ АРХИТЕКТУРАСЫ	16
2.1 Android платформасының құрылымы	16
2.2 Негізгі бағдарламалар	22
3 БАҒДАРЛАМАЛЫҚ ҚАМТАМАНЫ ҚҰРУ	24
3.1 Бағдарламалау тілін таңдауды негіздеу	24
3.2 Жалпы мағлұматтар	24
3.3 Функционалдық тағайындалуы	25
3.4 Қажетті техникалық жабдықтар	33
3.5 Шақыру және жүктеу	33
3.6 Кіріс мәліметтер	33
3.7 Шығыс мәліметтер	33
ҚОРЫТЫНДЫ	34
ПАЙДАЛАНЫЛҒАН ӘДЕБИЕТТЕР ТІЗІМІ	35
А Қосымшасы	36

КІРІСПЕ

Қазіргі уақытта интернет тұтынушылармен қарым-қатынас жасау үшін дамып келе жатқан ортаға айналып келеді. Сонымен қатар, интернеттің ыңғайлы және арзан арзан «сауда платформасы» болуының маңызы зор. Көптеген компаниялар өз өнімдерін онлайн ортада таныстыруға тырысады. Интернет-орта дамуы арқылы көптеген әртүрлі ұсыныстар дамып келеді. Енді адамдар қызығушылық танытқан ақпаратты ғана емес, сонымен қатар сатып алуды да жүзеге асыра алады. Бұл жағдайда интернет-дүкендер арқылы негізгі тұтынушы ретінде де, жоғары технологиялармен де толықтай әр түрлі санаттағы өнімдерді сатып алуға болады. Бұл жағдай, біріншіден, ғаламдық желіні пайдаланушылардың аудиториясының тұрақты өсуіне байланысты болады. Екіншіден, адамдар арасында уақыт болмауы, олардың жұмыс орны мен жұмыс жүктемесі өз әсерін тигізеді. Интернет-аудиторияның негізгі бөлігі кеңседе жұмыс істейтін және өз уақытының көп бөлігін компьютерде өткізетін адамдардан тұрады. Әдеттегідей, олардың өмір салты тұрақты дүкендерге бару үшін көп уақыт жұмсауға мүмкіндік бермейді.

Интернет әрдайым байланыста болуға және барлық жаңалықтардан хабардар болуға үздіксіз мүмкіндік береді. Бұл әсіресе бизнес үшін өте қолайлы, себебі хабардарлық болашақтағы бизнесті дамытуда үлкен рөл атқарады. Интернетте жыл сайын дамып келеді, көптеген адамдарға үлкен әсер тигізуде. Ғаламдық желінің барынша дамыған ортаға айналуына байланысты қарапайым адамдар интернетте жұмыс істеу дағдыларын, электронды поштаны, іздеу жүйелерін, жаңалық топтарын және интернеттің басқа да көптеген мүмкіндіктерін пайдалану мүмкіндігін білуі керек.

Есептеу техникасын пайдалану жұмыс үрдісімен байланысты жұмыстарды жеделдетуге, қызметкерлерді азайтуға және өнімділікті арттыруға мүмкіндік береді.

Тұтынушыларға қажетті бәсекеге қабілетті өнімдерді сатумен айналысатын фирмалар клиенттердің санын көбейту мәселесін шешеді. Ал қолмен жұмыс құжаттары көп уақыт, еңбек және материалдарды талап етеді, құжаттарды толтыру, жаңарту, сақтау және өңдеу процестерін автоматтандыру қажет. Бұл технологиялар құжаттардың жұмысын болдырмауға, ақпарат өңдеуге және жинауға жұмсалатын уақытты қысқартуға және техникалық қызмет көрсетудің қаржылық құнын төмендетуге көмектеседі.

Тапсырыстарды алуды, автоматтандыру тапсырыс беруші туралы және оның сатып алған өнімдері туралы ақпаратты нақты қызметкерге қызмет көрсететін, елиенттердің саны туралы жылдам алуға және өңдеуге мүмкіндік беретін мобильдік қосымша жасалуда. Жеделдетілетін өнім жеткізілетін мекен-жайлардың жеткізілімі бойынша қызмет көрсетіледі. Уақытты үнемдеуге байланысты, қызметкер клиенттердің көп санына қызмет көрсетуге мүмкіндік алады.

1 Пәндің саланы талдау

Компания жұмысын оңтайландыру және клиенттерге қызмет көрсету сапасын арттыру мақсатында тапсырысты және есепке алу жүйесін жетілдіру міндеті тұр. Қарастыру нысаны сатылатын тауарлардың регламенті болып табылады. Бұл қызметті автоматтандыру арқылы кәсіпорынның негізгі мақсаттары: өнім қорларын уақытылы және негізді толықтыру, ұсынылатын өнімдер қатарын қамтамасыз ету, көтерме және бөлшек сатып алушыларға өнімнің кең таңдауы қамтамасыз етіледі.

Осылайша, осы тезис жобасының міндеті «Сатуға арналған тауарларды жеткізудің автоматтандырылған есепке алу жүйесін ұйымдастыру» ретінде қалыптастыруға болады. Бұл міндет «Кәсіпорын қызметін талдау» міндеттер класына жатады және кәсіпорынның мақсаттарына қол жеткізу үшін қажет.

Осы міндеттің нәтижесі стратегиялық басқару шешімдерін қабылдау үшін негіз болып табылады. Сондықтан «жеткізуді есепке алу» міндеті кәсіпорынның шаруашылық қызметін басқаруды талдаудың маңызды және ажырамас бөлігі болып табылады. Мәселені шешуге арналған ақпарат сату бөлімінде бастапқы есепке алу және жеткізілім бөлімінде шоғырландырылған тапсырыстарды қалыптастыру кезінде алынады. Мәселені шешудің нәтижесі жеткізудің сәтсіздіктерін болдырмауға немесе дұрыс емес өнімдердің жеткізіліміне жол бермеуге мүмкіндік береді, сондай-ақ басқарушылық есепке алудың бастапқы деректері болуы мүмкін. Қолмен есепке алу кезінде, сатушының әрбір бөлімі өздеріне жүктелген функцияларды орындай отырып, есептік және операциялық және статистикалық есепке алудың басқа да құжаттарын дайындауға негізделген бастапқы құжаттардан дәл сол деректерді қайта жазады. Қолмен автоматтандырылмаған ақпаратты өңдеуді пайдалану заманауи бухгалтерлік талаптарды қанағаттандырудан бас тартты. Бұл ақпараттың бірнеше рет қайталануына, уақытты көбейтіп, өңдеудің өзіндік құнына әкеледі.

Автоматтандырылған жүйенің негізгі мақсаты:

- жеткізілім бөлімінің және жоспарлау-экономикалық бөлімнің негізгі функцияларын жүзеге асырудың тиімділігін арттыру, себебі олардың жұмыс істеуі өте үлкен құжаттық және ақпараттық ағындармен байланысты болады;
- шешім қабылдау жылдамдығын арттыру;
- еңбек өнімділігін арттыру;
- ақпаратты өңдеуді автоматтандыру арқылы есептеу қателерінің санын көбейту;
- ақпараттың тиімді және қауіпсіз сақталуын және қолжетімділігін алға жылжыту.

Көтерме кәсіпорындардың қызметкерлері жұмыс уақытының жартысынан астамын еңбек ақысын есепке алу және техникалық ақпаратты өңдеу операцияларын орындау үшін жұмсайды, олар түсімдерді, сатуды және тауар қорларын операциялық есепке алуды жүргізеді. Деректерді өңдеудің қарапайым рәсімдерін орындау арнайы білімді талап етпейді. Ақпараттың

көлемі өскен сайын, мұндай жұмыстардың үлесі артады. Бұл сауда ортасын зерделеу, тауарларға деген қажеттілікті анықтау, тауардың жеткізілуін бақылау және талдау, реттеу және т.б. сияқты маңызды шығармашылық жұмыстарды орындау уақытын азайтуға алып келеді.

Көтерме тауарларды тазалау, операциялық жазбаларды сақтау бойынша жаппай, қайталанатын операциялар, форматталуы мүмкін және, демек, автоматтандыру.

Сауда процестерін жедел басқаруды автоматтандыру айнымалы және тұрақты ақпараттың құрамын мұқият зерделеуді талап етеді. Тауарларды, көтерме сатып алушыларды сипаттайтын деректер тапсырыс беруді автоматтандыруды, тауарлардың түсуін және сатуды жедел есепке алуды қамтамасыз етуі тиіс.

Тапсырма бухгалтерлік есеп және өнім сатылымы туралы ақпаратты тиімді сақтауға, өңдеуге, талдауға және пайдалануға мүмкіндік беретін бірыңғай ақпараттық жүйені құру.

Өз қызметін жүзеге асыратын кез келген компания жеткізушілерден өнімдер алу үшін өніммен қамтамасыз ету үшін соңғы келісім жасасуы тиіс. Әдетте, кәсіпорын-тапсырыс беруші бірдей өнімге арналған өнім жеткізушілермен бірнеше келісім жасасады. Кейін, тапсырыс беруші белгілі бір өнімге деген қажеттілік ретінде жеткізушіге өнім жеткізуге сұрау жібереді және оның тауардың атауын және оның сату бағасын көрсететін соңғы шот-фактураны алады. Осы шот-фактуралардың негізінде клиенттік компания оңтайлы тәртіпті анықтайды және жеткізушілерге өнім жеткізуге тапсырыс жібереді. Тапсырыс берілген өнімдерді алғаннан кейін тапсырыс беруші шот-фактураны бухгалтерияға жібереді, ол шарт бойынша көзделген мерзімде банкте төлейді.

1.1 Қазіргі кездегі кәсіптік бәсекелеске шолу

Бүгінде бизнестің бәсекеге қабілеттілігі мен пайдалылығы менеджменттің шешімдерін қабылдайтын менеджерлердің қаншалықты жылдам және тиімді бизнес-процестер туралы деректеріне байланысты. Қазіргі заманғы ақпараттық технологияларды пайдаланатын және ақпараттық арналар арқылы деректерді таратудың тұйық циклін ұйымдастырған фирмалар тек жоғары басқару тиімділігіне қол жеткізе алады. Мұндай компаниялар бәсекелестер арасында жоғары сапалы менеджмент пен кез келген уақытта қол жетімді ақпарат негізінде тез және тиімді шешім қабылдау мүмкіндігіне ие.

Ақпараттық технологияларды енгізу тек қана компьютерлік басқару жүйесінің болуын ғана емес, сондай-ақ ақпаратты енгізудің жеңілдетілуіне, қолмен жұмыс істеу санын азайтуға және деректерді енгізу кезінде қателер санын барынша азайтуға бағытталған ақпараттық жинаудың бастапқы нүктелерінде сандық құрылғылардың болуы дегенді білдіреді.

Орындалған жұмыстардың нәтижесі ретінде келесідей әсерлері бар:

- компанияға жасалған тапсырыстарды есепке алу үшін қажетті уақытты шығару;

Жабдықтауды басқаруды автоматтандыру;

- кәсіпорын қызметінің тиімділігін барынша толық есептеу мүмкіндігін қамтамасыз ету үшін кәсіпорынға ұзақ мерзімді кезеңде жеткізілімдер туралы ақпаратты ұзақ уақыт сақтау мүмкіндігі;

- жеткізілімдер төлеу мерзімдерін үнемі білу;

-тапсырыстың алдын-ала есептелуі және бұйрықтың жасалуы.

Тапсырыс мәртебесін анықтау.

Төменде біз компанияның жүйесінде айналымға түсетін мәліметтерді сипаттаймыз:

- Алдыңғы тапсырыстар туралы мәліметтер.

Аяқталған тапсырыстар туралы компьютерлік ақпаратқа нақты үстеме ақпараттар негізінде енгізіледі. Шот-фактурада: Құжаттың нөмірі, күні, алушысы, жеткізушісі, атауы, саны, бағасы және жалпы сомасы көрсетіледі. Бұл деректер жазбаларды жүргізу және түрлі есептерді жасау үшін қажет.

-План тапсырыстары.

Тауар тапсырыс берушіге жіберілген кезде шығыс құжаттары шығарылып, қоймадағы теңгерімді анықтау қажет. Осы деректерге сүйене отырып, құжаттың нөмірі, күні, атауы, саны, бағасы және жалпы сомасы көрсетіледі.

- Қоймадағы тауардың болуы.

Өнімдерді қабылдаған кезде және сатқан кезде тауардың немесе артықтың қайта бағалануы болады. Қоймадағы деректер шындыққа сәйкес келуі үшін осы процесті реттейтін құжаттарды енгізу қажет. Құжат түгендеу негізінде шығарылуы мүмкін. Құжаттың сипаттамасы стандартты түрде толтырылады - жолды енгізу немесе каталогты таңдау арқылы. Құжатта келесі салалар болуы тиіс: тауарлардың түрлері, олардың саны, алу күні, құны.

- Тауарлардың құны.

Бұл құжат әрбір тауардың құны мен оның мөлшерін белгілейді.

- Бюджеттің көлемі.

Жаңа өнімдерді сатып алу үшін бюджет қажет. Бұл бюджет сатылған тауарлардан түскен пайда негізінде қалыптасады. Осы бюджет негізінде жоспарлау жүзеге асырылады.

1.2 Мәселелерді жою үшін техникалық қолдану техникасын қолдану мақсаты мен міндеті

Мәселенің шешілетін технологиялық үдерісі үш кезеңнен тұрады. Бірінші кезеңнің мақсаты - өндеу үшін деректерді жинау, тіркеу және беру. Нәтиже - құжат (қосымша). Екінші кезеңнің мақсаты - деректерді компьютерлік медиаға беру және ақпараттық базаны қалыптастыру. Үшінші

кезеңге жинақтау, сұрыптау, түзету, деректерді өңдеу және нәтижелер беру кіреді.

Мынадай талаптарды қарау қажет:

- өңделген ақпараттың дұрыстығын қамтамасыз ету;
- міндеттерді уақытында шешу;
- деректерді өңдеудің минималды еңбегін және шығындарын қамтамасыз ету;
- компьютерде деректерді өңдеудің болуы;
- мәселені түрлі режимдерде шешу қабілеті.

Бұл талаптарды бірнеше факторлармен қанағаттандыруға болады:

- операциялар санының азаюы, әсіресе қолмен жасалған;
- кіріс туралы ақпаратты қатаң бақылау жүйесін жетілдіру;
- пайдаланушылардың біліктілігін арттыру, еңбек жағдайларын жақсарту және, тиісінше, өнімділікті арттыру.

Деректерді жинау, жазу және беру әдісінің таңдауына келесі факторлар әсер етеді:

- деректерді өңдеу орталығынан ақпарат көздерін қамтамасыз ету;
- арнаулы байланыс арналары арқылы ақпарат көздерімен байланыс жасау мүмкіндігі.

Ақпаратты өңдеу кезінде деректер базасын пайдалану қажет. Бұл іздестіру, таңдау, сұрыптау және т.б. жылдамдықтарында артықшылық береді. Шығарылған ақпаратты өңдеу және жіберу алдында нәтижелерді көру керек.

Осы мақалада сипатталған мәселелер кешенін шешуде есептеулерді пайдалану бірқатар факторларға байланысты. Шығару туралы ақпараттың көлемі мен сапасы компьютерлік технологияны тез және ең маңыздысы дұрыс пайдаланбай проблемаларды шешуге мүмкіндік бермейді.

Қазіргі уақытта сауда-сервистік компания қызметіндегі компьютерлік технологияның рөлі жоғары бағаланбайды. Үлкен және ықшам дерекқорлар үлкен сатылым кітапшаларын ауыстырады. Бірнеше жүз данадан тұратын шот-фактураны қолмен берудің орнына, компьютер бірнеше секундтан кейін құжат жасайды.

Компьютерлік бағдарламалық қамтамасыз етудің жұмыс істеуі қажет. Егер жүйелік бағдарламалық жасақтама бүгінгі күні соңғы пайдаланушылары үшін кең ассортиментпен болмаса, онда қолданбалы бағдарламалар нарығында қатаң бәсекелестік бар. Соңғы пайдаланушыға арналған ірі бағдарламалық жасақтама корпорацияларының күресі аясында бір бағдарламалық жасақтама өнімдері жай көрінбейді.

Жүйемен ыңғайлы пайдаланушылары үшін жүйелік интерфейсті мұқият қарастыру қажет - бұл қарапайым және бір уақытта функционалды. Дерекқордың оңтайлы жұмыс істеуі үшін кестелер арасындағы логикалық қатынастарды дұрыс анықтау қажет болады. Осылайша, дерекқордың өзі

және клиенттік қосымшаның негізін дамыту айтарлықтай уақытты жұмсауға болады.

Ақпараттық жүйені құру нәтижесінде экономикалық тиімділіктің негізгі көздері:

- уақытты үнемдеу және өнімділікті арттыру;
- тапсырысты орындау жылдамдығын арттыру;
- қажетті құжаттарды неғұрлым нақты ұсыну арқылы кәсіпорын сапасын жақсарту және экономикалық тиімділікті арттыру.

1.3 Есептің қойылымы

Тапсырма жылдам іздестіру үшін қажетті тағамдар мен өнімдер туралы ақпаратты өңдеу үшін қажетті тағамдарды тапсырыс беру үрдісін android қосымшасында автоматтандырады. Процесті автоматтандыру арқылы мейрамхана ең көп сатып алынатын және танымал өнімдерді анықтауға мүмкіндік береді, бұл кейінірек сұраныстың өзгеруін қадағалап, тиісінше шикізатты сатып алуға тиімді жоспар жасайды. Тұтынушылардың дәмі бойынша ұйымның тұтынушылары арасында танымалдығы артып, түсімнің ұлғаюы.

Бұл тапсырмада атқарылған қызметтерге мыналар жатады:

- қордағы өнімдердің санын есепке алу және тіркеу;
- тағамдарды атау және баға бойынша тандау.
- белгіленген параметрлер бойынша есептер шығару;
- ыдыстарды түрі бойынша сұрыптау.

Процесті автоматтандыру кезінде енгізілген есептер:

- дүкеннің мәзірін қалыптастыру;
- ас үйге тапсырылған тағамдарды көрсету.

Жоғарыда келтірілген бірнеше домендік шектеулер:

- әрбір жазба бірегей болуы керек және олардың көшірмелері болмауы керек;
- қоймадағы шектеулі өнімдер;
- өнімдердің бағасы рубльмен өлшенеді.

Автоматтандырылған жүйесін дамытудың мақсаты әмбебап жүйені дамыту болып табылады.

Осы мақсатқа жету үшін келесі міндеттерді шешу қажет:

- анализ және пәннің сипаттамаларын ескеру;
- бағдарламалық өнімдерді жасау және енгізу;
- электрондық есептеу техникасын пайдалану кезінде қауіпсіздік шараларын зерделеу;
- дамыған жүйе үшін бизнес-тапсырма беру.

Ақпараттық жүйенің мақсаты:

- Кез келген уақытта басқару объектісінің жай-күйі туралы ақпарат беру;
- Ақпаратты жеделдету және автоматтандыру;

- Қателермен және ақпаратты өңдеу жылдамдығымен байланысты шығындарды алып тастау.

Басқару шешімдерін қабылдау уақытын қысқарту және деректерді жинау уақыты.

- Ақпараттың қауіпсіздігі мен тұтастығын жақсарту.

2 ANDROID ПЛАТФОРМАСЫНЫҢ АРХИТЕКТУРАСЫ

2.1 Android платформасының құрылымы

Android платформасы - құрылғыны, жады мен процестерді басқаруға арналған Linux негізіндегі амалдық жүйенің бағдарламалық құралы. Бұл жинақтың әртүрлі құрамдас бөліктері иерархияның бірнеше деңгейлерін және тұтастай алғанда мобильді құрылғының жұмысын қамтамасыз етеді. Android компоненттері құрылғы жабдығымен тікелей жұмыс істеп, телефонияға, мультимедиа және т.б. байланысты функцияларды қамтамасыз етеді, ал мобильді құрылғының көптеген мүмкіндіктерін іске асыру үшін негіздемеге ұсынады. Басқаша айтқанда, Android платформасында тікелей операциялық жүйе, аралық бағдарлама және қосымшалар жинағы бар [1].

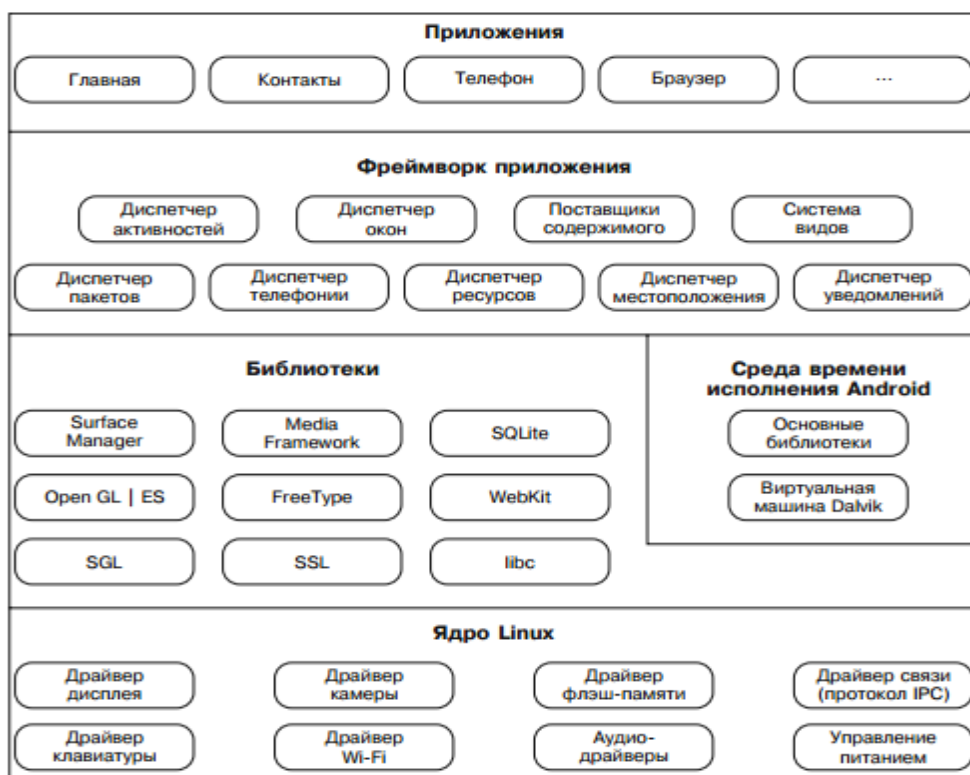
Android - бағдарламалық жасақтама жинағы, яғни бағдарламалық жасақтама компоненттерінің жиынтығы. Екінші жағынан, Android платформасындағы құрылғы экран, кілттер, батарея, процессор, жады, түрлі сенсорлар және датчиктер сияқты жабдықтардың жиынтығы болып табылады. Android платформасында келесі құрылымы бар бірыңғай аппараттық-бағдарламалық кешені ретінде айтуға мүмкіндік береді:

- жабдық - құрылғы жұмысын қамтамасыз ететін элементтер жиынтығы;
- linux kernel - бұл linux операциялық жүйенің ядросы және
- аппараттық құралдарды басқару және жадты бөлу, тапсырмаларды басқару, қауіпсіздік және т.б. үшін негізгі механизмдерді қамтамасыз ететін драйверлер жиынтығы;
- операциялық жүйенің әр түрлі компоненттері пайдаланатын
- кітапханалар (c/ c++);
- dalvik virtual machine (DVM) - android қосымшалары мен
- операциялық жүйенің компоненттері (OS) үшін орындау ортасын қамтамасыз ететін, dalvik виртуалды машинасы;
- негізгі кітапхана - негізгі android кітапханалары, java тіл
- кітапханасының негізгі қол жетімді функциялардың көпшілігін, сондай-ақ android операциялық жүйесінің арнайы функцияларын қамтиды;
- application framework - қосымшаларды операциялық жүйенің
- функциялары мен кітапханаларына интерфейсімен қамтамасыз ететін java сыныптарының жиынтығы (API).

Android операциялық жүйесіне арналған бағдарламалар көбіне java бағдарламалау тілі арқылы жасалады. Құрастырылған бағдарлама коды арнаулы мұрағаттық файлдарға, android package-ға қапталған. Бұл файл *.apk кеңейтіміне ие және арнайы утилита aapt құралымен жабдықталып, бағдарлама ретінде таратылып, мобильді құрылғыларға орнатылады. Осындай файлдардың біреуі бір бағдарламаның кодымен байланысты болады. Әрбір Android қосымшасы әдепкі бойынша, әрбір бағдарлама өз процесінде іске қосылады, ол Linux ядросында басқарылып, сондан кейін

жадты басқаруды орындайды. Осылайша, көбінесе қолданба коды басқа барлық қосымшалардан бөлек орындалады. Android-тің кейбір бағдарламалары кодын орындау қажет болғанда, процесті бастайды және қажет болмаған кезде жүйе ресурстары әртүрлі қолданбаларды тоқтатады. Әдепкі бойынша, әрбір бағдарламаға бірегей Linux пайдаланушы идентификаторы тағайындалады. Қол жеткізу рұқсаттары, бағдарлама файлдары, тек осы пайдаланушыға және осы бағдарламаға көрінетін етіп орнатылады. Жүйе ресурстарын үнемдеу үшін сол идентификатормен қосымшаларды бірдей виртуалды машинаны ортақ пайдалану арқылы сол Linux процесінде іске қосуға болады.

Бағдарламашы тұрғысынан Android - әзірлеушіге ядродан тез мәлімет беретін және оған Java кодын жасауға мүмкіндік беретін платформа. Android бірнеше пайдалы мүмкіндіктері бар. Біріншіден, бұл әртүрлі қосымшаларды жасауға API-дің үлкен жиынтығын ұсынады. Сонымен қатар платформа қосымшаларын қайта пайдалануға болатын және ауыстырылатын компоненттерден тұрады. Екіншіден, Android қосымшаларын іске асыруға жауап беретін Dalvik виртуалды машинасының болуы. Бұдан басқа, әзірлеуші қызметтеріне 2D және 3D-қосымшалар үшін графикалық кітапханалар жиынтығы, мультимедиялық пішімдерді қолдау, камераға қол жеткізу үшін API, GPS, компас, акселерометр, сенсорлық экран, джойстик және пернетақта кіреді. Android архитектурасы компоненттер жиынтығынан жасалады. Әр компонент төменгі деңгейлі элементтер негізінде құрылады. 2.1-суретте – Android негізгі компоненттеріне қысқаша шолу жасайды.

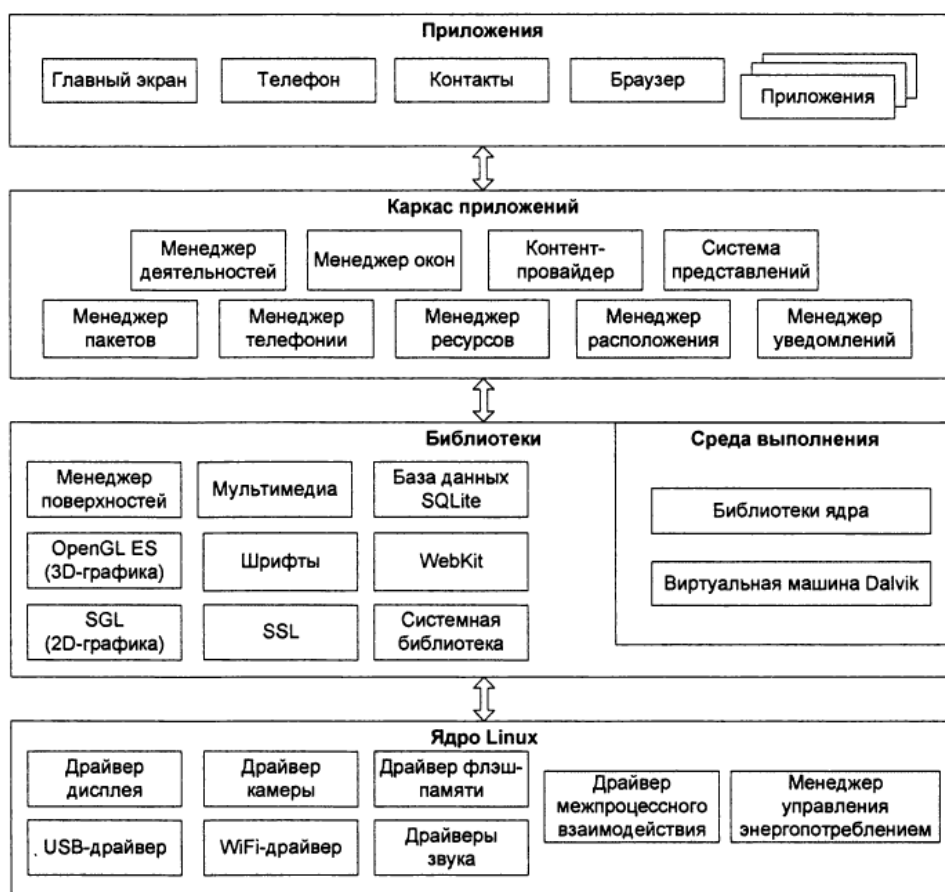


2.1-сурет – Android платформасының негізгі компоненттері

Әрбір бағдарлама өзінің Dalvik виртуалды машинасымен жұмыс істейді. Dalvik бағдарламасы DEX байт-код форматында жұмыс істейді. SDK-да қол жетімді арнайы dx утилитасын пайдаланып, CLASS кеңейтімі бар Java файлдары DEX пішіміне түрлендіріледі. DEX пішімі кесте бөлу және бірнеше CLASS файлдарын біріктіру арқылы қол жеткізіледі. Dalvik виртуалды машинасы Java бағдарламалары үшін негізгі функцияларды ұсынатын негізгі кітапханалармен өзара әрекеттеседі. Бұл кітапханаларда Java SE арқылы қол жетімді үлкен, бірақ толық емес жиынтықтар бар.

Android архитектурасын алты деңгейге бөлуге болады:

- аппараттық деңгей;
- Linux ядросының деңгейі;
- отандық кітапханалардың деңгейі;
- Android жұмыс уақыты деңгейі;
- қолданба аясы;
- қолданба деңгейі.



2.2-сурет – Android платформасының сәулеті

Linux ядросы және драйвер жиынтығы. Android платформасының орталығы - операциялық жүйенің негізгі міндеттерін шешуге және жабдықпен өзара әрекеттесуге жауап беретін Linux операциялық жүйенің

ядрасы болып табылады. Бұл деңгейде процестерді басқару және жады бөлу, файлдық жүйені басқару, қауіпсіздік, сондай-ақ желілік функцияларды басқару қызметтері - негізгі қызметтер болып табылады. Тек қана ядро деңгейінде, құрылғының аппараттық құралдарымен тікелей өзара әрекеттесу драйверлер жиынтығы арқылы жүзеге асады. Ядролық деңгейдің негізгі компоненттері:

- ілесіме байланыс драйвері (IPC драйвері);
- драйвердің қуатын басқару (android power management);
- мобильді құрылғыға қосылған жабдыққа арналған драйверлер
- жиынтығы [2].

Android операциялық жүйесі Linux ядросына негізделгенімен, олар арасында кейбір айырмашылықтар бар (Android және Linux таза түрде). Мәселен, Android, мысалы, өзінің жады бөлу тетіктері, процестердің өзара әрекеттесуі және т.б.

IPC драйвері. Android-тегі қолданбалар мен қызметтер бір-бірімен өзара әрекеттесе алатын және ортақ жадқа қол жеткізе алатын бөлек (қорғалған) үрдістерде жұмыс істейді. Android платформасында IPC (Inter-Process Communication) механизмі қамтамасыз етіледі, ол жүйедегі процестердің өзара әрекеттесуінің негізгі механизмі болып табылады. IPC драйвері процестердің өзара әрекеттесуін қамтамасыз етіп, процестердегі пулдарды құру және өңдеу, процестердегі нысандарға сілтемелерді санау және көрсету, процестердің синхронды сұрауларын орындайды.

Энергияны басқару. Қуатты басқару жүйесі (Android Power Management) мобильді құрылғыларға оңтайландырылған Linux басқаруындағы стандартты драйвер. Бұл жүйенің негізгі функциясы, төмен қуат режимдерін қолдау есебінен мобильді құрылғының аккумуляторлық зарядын, экономикалық тұрғыда пайдалану болып табылады.

Аппараттық драйверлер. Android бағдарламалық жасақтама жинағы аппараттық құрал конфигурациясын таңдау кезінде талап етілетін икемділікпен жасалады. Басқаша айтқанда, Android бағдарламалық жасақтамасының стилінің архитектурасы ұтқыр құрылғыларға қосымша функционалдылықты қамтамасыз ететін түрлі аппараттық құралдар мен құралдарды пайдалануға мүмкіндік береді [3].

Жоғарыда айтылғандай, құрылғыдағы аппараттық құралдармен тікелей өзара әрекеттесу тек Linux операциялық жүйесінің ядросының деңгейінде орын алады, бұл Android бағдарламалық жасақтамасының басқа деңгейлері үшін аппараттық құралдағы барлық айырмашылықтар ашық болуы керек. Дәл осы себепті ядро деңгейі құрылғыда барлық қажетті аппараттық драйверлерді қамтиды. Драйверлер жиынтығы өндірушінің және құрылғы үлгісіне байланысты әр түрлі болуы мүмкін.

Жергілікті кітапханалар. Linux ядросының үстінен келесі деңгей ОЖ-ның әр түрлі компоненттері пайдаланатын кітапханалардың жиынтығын қамтиды. Осы деңгейдегі кітапханалар функционалдық мақсатына сәйкес келесі топтарға бөлуге болады:

- биондық жүйелік кітапхана (libc), с-да жазылған;
- басқарушы беттер;
- C/C++ функционалды кітапханалары.

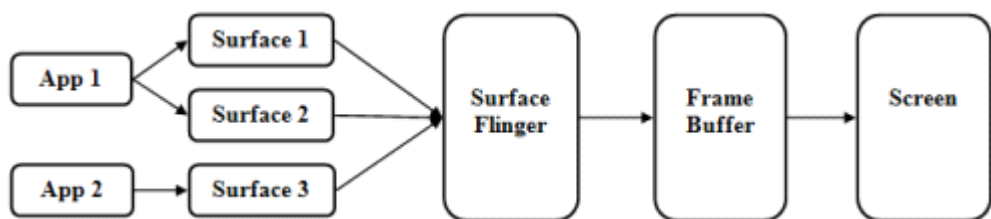
Bionic жүйелік кітапханасы (libc). Жүйелік кітапхана Berkeley Software Distribution (BSD) лицензиясына негізделген және Linux негізіндегі мобильді құрылғыларға арналған.

Google осы кітапхананың жеке нұсқасын (liion Bionic) келесі себептер бойынша жасады:

- жадты сақтау. Бұл кітапхана әр процесте жүктелгендіктен, оның
- өлшемі мүмкіндігінше аз болуы керек. bionic кітапханасы көлемі шамамен 200 кбайт болады;
- ең үлкен жылдамдық. Bionic кітапханасы ұялы құрылғыда жұмыс
- істеу үшін оңтайландырылған.

Bionic кітапханасы Android үшін маңызды жүйелік қызметтерді қолдайды, атап айтқанда жүйелік оқиғаларды тіркеу қызметін қолдау. Bionic кітапханасы GNU libc және POSIX стандарттарына сәйкес келмейді.

Жоғарғы менеджер (Surface Manager). Android жүйесі Compoz сияқты жоғарғы басқаруды пайдаланады. Жоғарғы басқарушы барлық беткі рендерингті кадр буферіне өңдейді [4]. Басқарушы бірнеше қосымшалардан 2D және 3D беттерін біріктіре алады. Графика бейнелеу буферіне тікелей салудың орнына, жүйе кіріс сурет командаларын дауыспен буферге жібереді, сонда олар басқалармен бірге жиналып, белгілі бір композиция құрып, содан кейін экранда көрсетіледі. Бұл процесті 2.3-суретте көруге болады.



2.3-сурет – Беттік менеджердің графикалық ақпаратын өңдеу

Функционалды кітапханалар. Android операциялық жүйесінің түрлі компоненттері пайдаланатын кітапханалар болып табылады. Олардың кейбіреулері төменде келтірілген:

- Media framework - мультимедиа функцияларын (аудио және бейне
- ойнату) және түрлі медиа пішімдеріне қолдауды қамтамасыз етеді.
- SQLITE - реляциялық деректер базасына қолдау көрсету
- кітапханасы.
- OpenGL - 3D графикалық қозғалтқыш.
- Freetype - бұл растр және векторлық қаріптермен жұмыс істеуге

- арналған қаріптер кітапханасы.
- Libwebcore - веб-браузерді ұсынылуы.
- SGL - 2d графикалық қолдау.
- SSL (secure sockets layer) - кітапхана сокеттермен қауіпсіз режимде

- жұмыс істеуге арналған механизм.

Жұмыс уақытының ортасы (Android Runtime). Жұмыс уақытының ортасы негізгі Android кітапханаларының жиынтығынан және Dalvik виртуалды машинасынан тұрады. Бұл деңгей Android қолданбалары үшін жұмыс уақытының ортасын қамтамасыз етеді.

Негізгі кітапханалар Java ядросы кітапханаларына (Java API) қол жетімді төменгі деңгейдегі функционалдылықты қамтамасыз етеді. Бұл кітапханалар Android үшін Java қосымшалары негізгі даму платформасы болып табылады. Тіркелген Java виртуалды машинасы, Dan Bornstein Android мобильді платформасының бөлігі ретінде әзірленген және жазылған [5].

Dalvik виртуалды машинасы - коммуникаторлар мен планшетті компьютерлер сияқты кіріктірілген құрылғыларда жиі пайдаланылатын RISC-негізделген процессорлық сәулет жүйелерінде жұмыс істеуге ыңғайлы, стандартты емес тіркелімге бағытталған виртуалды машина. Жұмыс үстелі виртуалды машиналарының көбісі Oracle стандартты Java виртуалды машинасын қоса алғанда стектерге бағытталған.

Жалпы айтқанда, Dalvik - бұл Android операциялық жүйесінің компоненттерін және қолданушы қосымшаларын орындау үшін ортасы. Әрбір процесс жеке, оқшауланған мекенжай кеңістігінде орындалады. Пайдаланушы бағдарламаны бастаған кезде, Dalvik Virtual Machine ядросы қолданбаны іске қосу ортасы ретінде VM тікелей таратылатын, қорғалған процесті жасайды. Басқаша айтқанда, Android Dalvik виртуалды машиналарының жиынтығына ұқсас болып, олардың әрқайсысы қолданбалы тапсырманы орындайды.

Қолданба аясы. Қолданба аясы - жергілікті кітапхана деңгейі мен Android жұмыс уақыты мен бағдарлама деңгейінің арасындағы байланыс. Бұл қабат амалдық жүйенің функцияларына интерфейстермен қамтамасыз етеді. Бұл деңгейде қосымшалардың, пакеттердің, ресурстарды және т.б. өмірлік циклін басқару үшін негізгі Android қызметтері бар болады. Тұтынушы бағдарламалары жүйе қолданбалары сияқты бірдей API-ке толық қатынас алады.

Осы деңгейдегі кітапханалар мен қызметтер Android операциялық жүйесіндегі барлық қосымшалар үшін негіз болып табылады және мыналарды қамтиды:

- пайдаланушы интерфейсінің кеңейтілген жиынтығы. бұл
- мәтіндік өрістер, түймелер, тізімдер, кестелер, ендірілген веб-шолғыш және т.б., барлық бағдарламаларда қолданылатын элементтер.
- activity manager (басқару қызметі) - бағдарламаның өмірлік циклі мен

- activity stack қызметін басқарады. әрекет арасында навигациялық жүйемен қамтамасыз етеді;

- package manager - пакет менеджері құрылғыға қосымшалар
- бумаларын орнатуды және орналастыруды басқарады;
- ресурс реттеушісі - бағдарлама ресурстарына қатынасу үшін
- функцияларды қамтиды;
- контент-провайдерлер - қолданбаларға басқа қолданбалардан
- деректерге қол жеткізуге немесе олардың деректеріне кіруге рұқсат беретін мазмұн жеткізушілері;
- телефония менеджері - телефония менеджері, телефонияның негізгі
- функцияларына қатынауды (арі) ұсынады;
- location manager - орналасқан жердің менеджері, қолданыстағы орын
- туралы және оның өзгеруі туралы ақпаратпен жұмыс істеу функцияларын ұсынады;

- Notification Manager - хабарландыру менеджері, Android
- хабарландыру жүйесімен жұмыс істеу үшін API ұсынады.

Қолданбаның негіздері компоненттері оларды қолдануды жеңілдететін және қосымшаларды жасаған кезде операциялық жүйенің функцияларына және жергілікті кітапханаларға мүмкіндігінше қол жеткізуге мүмкіндік береді.

Қолданба деңгейі. Android платформасында мобильді құрылғы негізгі немесе басқа сөзбен жүйелік қосымшалармен бірге келеді. Сондай-ақ, жүйеге үшінші тарап қолданбаларын да қосуға болады. Барлық осы қосымшалар Java-ге жазылған. Маңыздысы, Android-де жүйе мен үшінші тарап қосымшалары арасында айырмашылық жоқ. Бұл дегеніміз, кез-келген жүйелік қосымшаны балама бағдарламамен ауыстыруға болады.

Кез келген қолданба амалдық жүйе қамтамасыз ететін функционалдылыққа толық қол жеткізе алады. Барлық қосымшалардың архитектурасы басқа қосымшалардың бөлшектерін немесе функцияларын оңай пайдалану мүмкіндігі бар.

Java SE-ның кейбір функционалдық мүмкіндіктерін ұсынатын ядро кітапханаларына қосымша, бағдарлама негізі үшін негіз жасайтын C/C++ кітапханаларының жиынтығы бар. Бұл жүйелік кітапханалар негізінен күрделі, Dalvik виртуалды машина үшін өте қолайлы емес. Оларда API-лар ойын жазу кезінде қолданылатын бағдарлама шеңберіндегі Java сыныптарын пайдаланып оралады.

2.2 Негізгі бағдарламалар

Бағдарлама жүйесі жүйелік кітапханаларды және іске қосу уақытын біріктіреді, осылайша Android пайдаланушысын жасайды. Негіздеме қосымшаларды басқарып, жұмыс істейтін және ойланатын ортаны ұсынады. Өзірлеушілер Java Framework интерфейстерінің жиынтығын пайдаланып, интерфейсті дамыту, өндік қызметтер, ескертулер, ресурстарды басқару, шеткерушілерге қол жеткізу және т.б. сияқты салаларды қамтиды. Android

OS-мен бірге келген барлық негізгі бағдарламалар осы API-лердің көмегімен жазылған. Қолданбалар, интерфейс немесе фондық байланыса алады. Бұл байланыс бір қосымшаның басқа компоненттерін пайдалануға мүмкіндік береді. Қолданба осы әрекетті қамтамасыз ететін басқа бағдарламаның жүйелік құрамдас бөлігін сұрайды. Сонымен қатар, бірінші қолданба осы құрамдасты қайта пайдалануға болады. Бұл алгоритм бағдарламашыдан жүктеменің маңызды бөлігін алып тастап, сонымен қатар Android-тің мінез-құлқының әртүрлі аспектілерін реттеуге мүмкіндік береді.

Android SDK (Бағдарламалық жасақтама әзірлеу жинағы) Android қолданбаларын әзірлеу үшін пайдаланылады. Ол құралдардың кең ауқымы, құжаттама және әртүрлі қызметтерден тұрады. Сондай-ақ, Android қосымшаларын жасау үшін және қосымшалар құрылымы үшін API-ді қамтитын Java кітапханаларын қамтиды. Даму құралы ретінде барлық негізгі операциялық жүйелерге қолдау көрсетіледі.

SDK-ның негізгі ерекшеліктері:

- нақты құрылғыда немесе эмуляторда жұмыс істейтін қолданбаларды өңдеуге мүмкіндігі;
- тиімсіз кодты табуға көмектесетін жады және өнімділік профильдері;
- qemu-ге негізделген құрылғының эмуляторы әрдайым жылдам болмаса да, өте дәл болып табылады;
- құрылғымен байланысу үшін пәрмен жолының утилиталары;

SDK Java үшін ашық, кең танымал және функционалдық даму ортасы болып табылатын Netbeans интеграциялануы мүмкін. Бұл интеграция Android жобаларын құруға, оларды эмуляторда немесе құрылғыда жөндеуге және оларды Android Market-ке орналастыру үшін пакеттер жасау үшін Netbeans-дің жаңа мүмкіндіктерін қосатын Android Development Tools плагині арқылы қол жеткізіледі. Eclipse үшін SDK және ADT плагині үнемі жаңартылып, жаңа мүмкіндіктерді қосады. Сондықтан оларды үнемі жаңартып отыру дұрыс. Кез-келген жақсы SDK толық құжаттамамен бірге болуы керек. Android SDK - көптеген мысалдармен бірге құжаттамаға қосымша. Сонымен қатар, әзірлеуші нұсқаулығын және ресми веб-сайтқа қосымшаның барлық модульдері үшін API толық сипаттамасын табуға болады.

3 БАҒДАРЛАМАЛЫҚ ҚАМТАМАНЫ ҚҰРУ

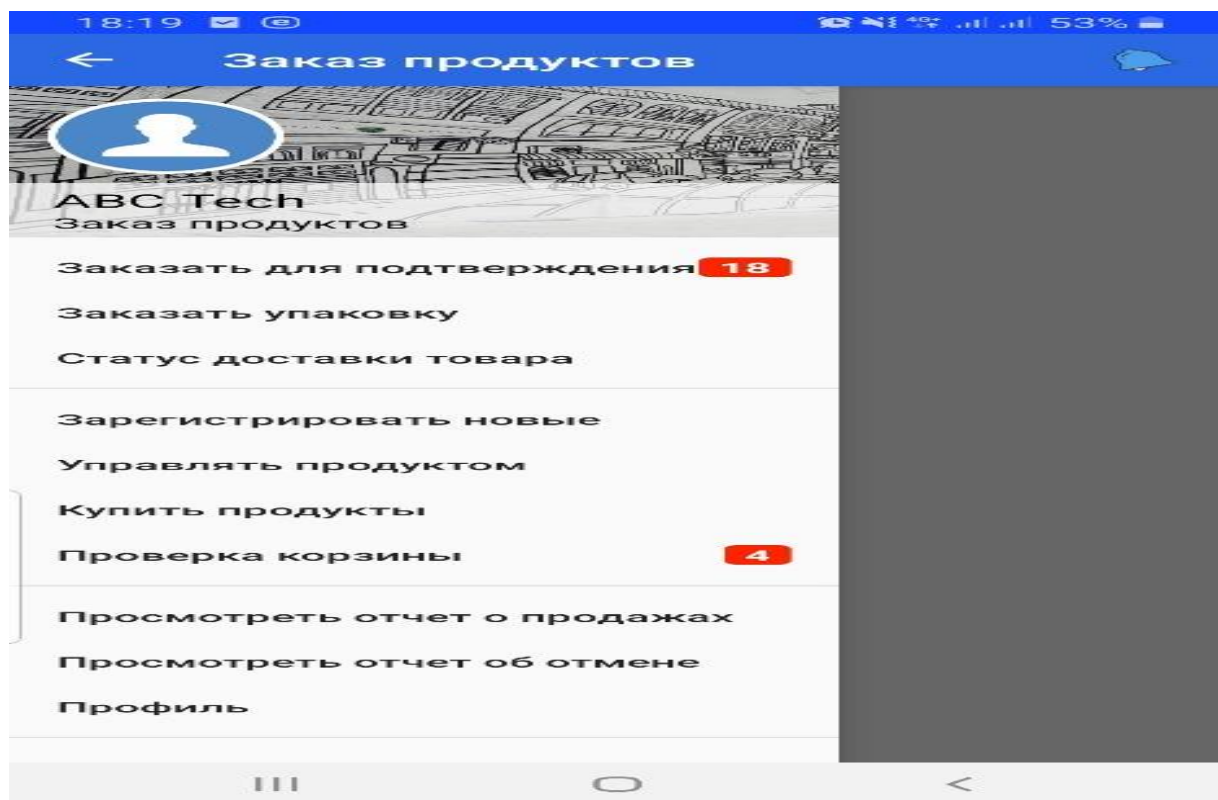
3.1 Бағдарламалау тілін таңдауды негіздеу

Java бағдарламалық платформасы - Sun Microsystems компаниясының бұрынғы тәуелсіз компаниясы бағдарламалық өнімдер мен спецификациялар жиынтығы. Қазіргі уақытта Oracle компаниясының еншілес компаниясы, ол бірлесіп қолданбалы бағдарламалық жасақтама әзірлеу және кез-келген кросс-платформалық бағдарламалық жасақтамаға интеграциялау үшін жүйе ұсынады. Java әр түрлі компьютерлік платформаларда, төменгі бағалық сегменттегі ендірілген құрылғылар мен ұялы телефондардан, жоғары баға сегментінде корпоративтік серверлерге және суперкомпьютерлерге қолданылады. Java қосымшалары үстел компьютерлерінде сирек қолданылғанымен, олар кейде кең ауқымды веб-торапты қарап шығу кезінде функционалдылықты жақсарту және қауіпсіздікті арттыру үшін пайдаланылады [5].

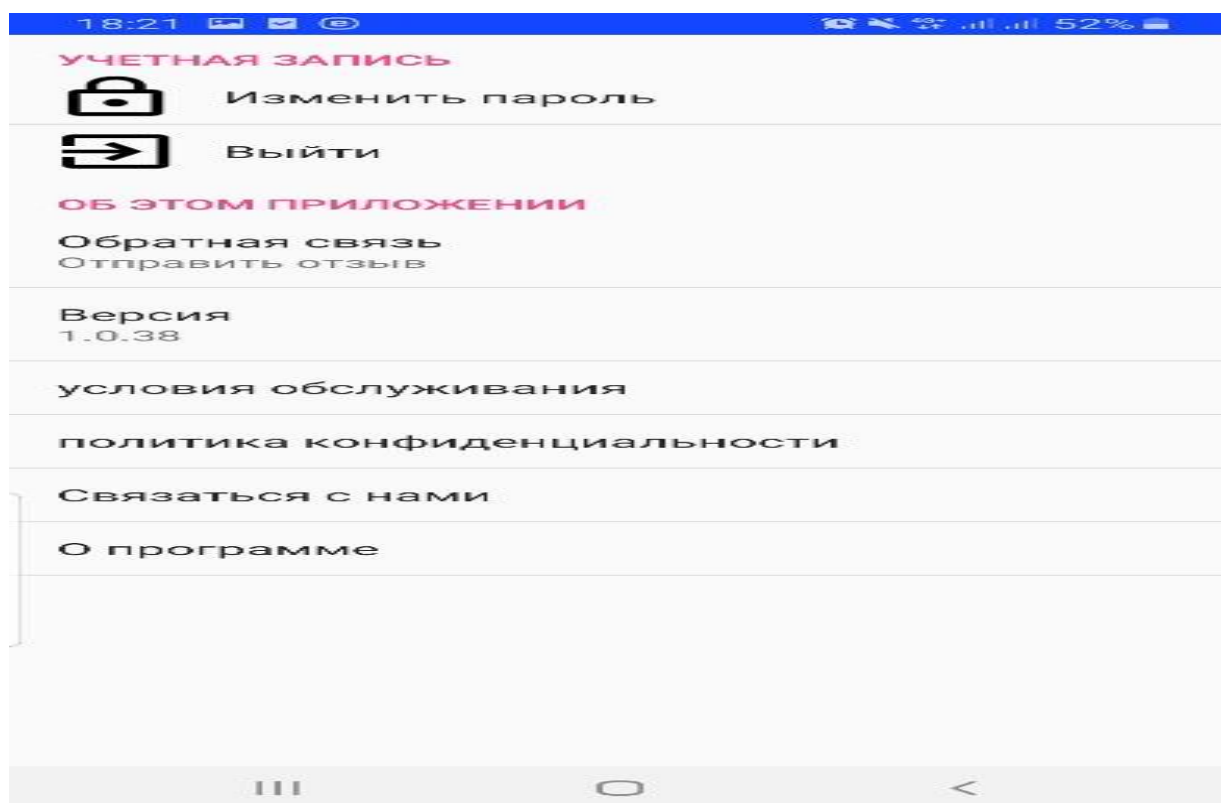
3.2 Жалпы мағлұматтар

Android OS-мен жұмыс істейтін телефондарға арналған көптеген бағдарламалар қолданушылардың мүмкіндіктерін шексіз кеңейтеді. Олардың көмегімен телефонды ұялы банкке, музыкаға немесе бейне ойнатқышқа, навигаторға және т.б. жасауға болады. Арнайы қосымшалардың арқасында телефон жеке фитнес-тренерге немесе сөйлейтін ойыншыққа айналуы мүмкін.

3.3 Функционалдык тағайындалуы



3.1-сурет – Мобильдік қосымшаның негізгі мәзірі



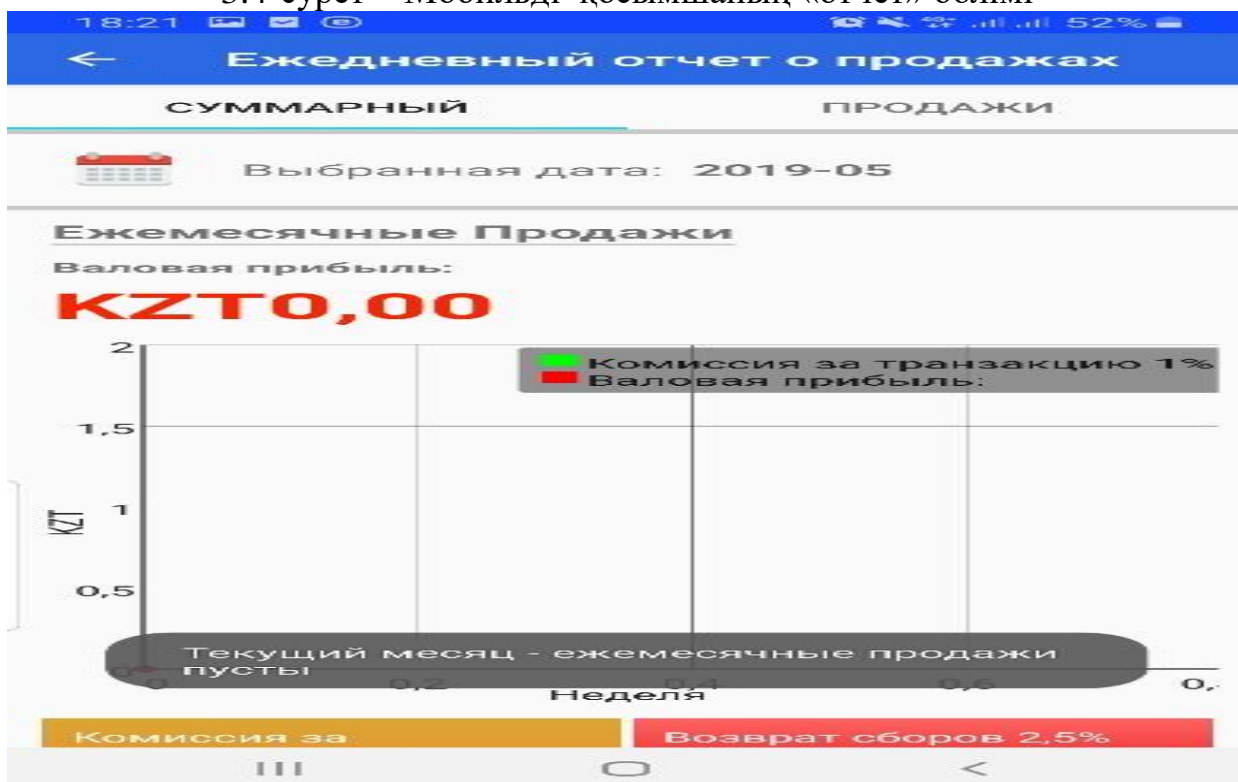
3.2-сурет – Мобильді қосымшаның «учетная запись» бөлімі



3.3-сурет – Мобильді қосымшаның «профиль» бөлімі



3.4-сурет – Мобильді қосымшаның «отчет» бөлімі



3.5 – сурет. Мобильді қосымшаның «отчет о продажах» бөлімі

18:21 53%

Summary Of Transaction

Net Charges MYR 1.00

Payment Option

Credit Card

Credit Card Details

Name

Credit Card No.

CVC/CCV2

Expiry Date

Issuing Country Malaysia

Issuing Bank Please Select

I authorize Mobile88.com Sdn Bhd to debit the above net charges from my credit card.

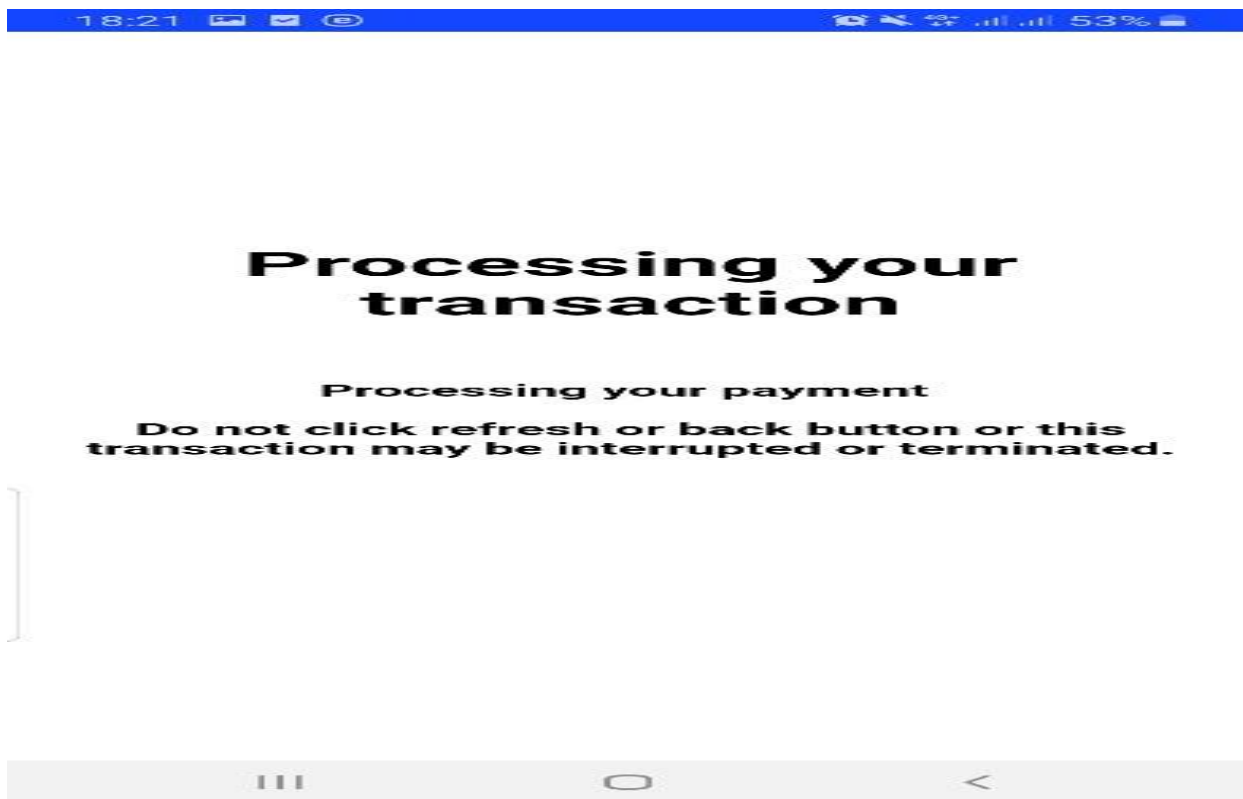
I have read & agreed to iPay88 Privacy Statement.

Important Note:

» Proceed Cancel

MasterCard SecureCode VERIFIED by VISA

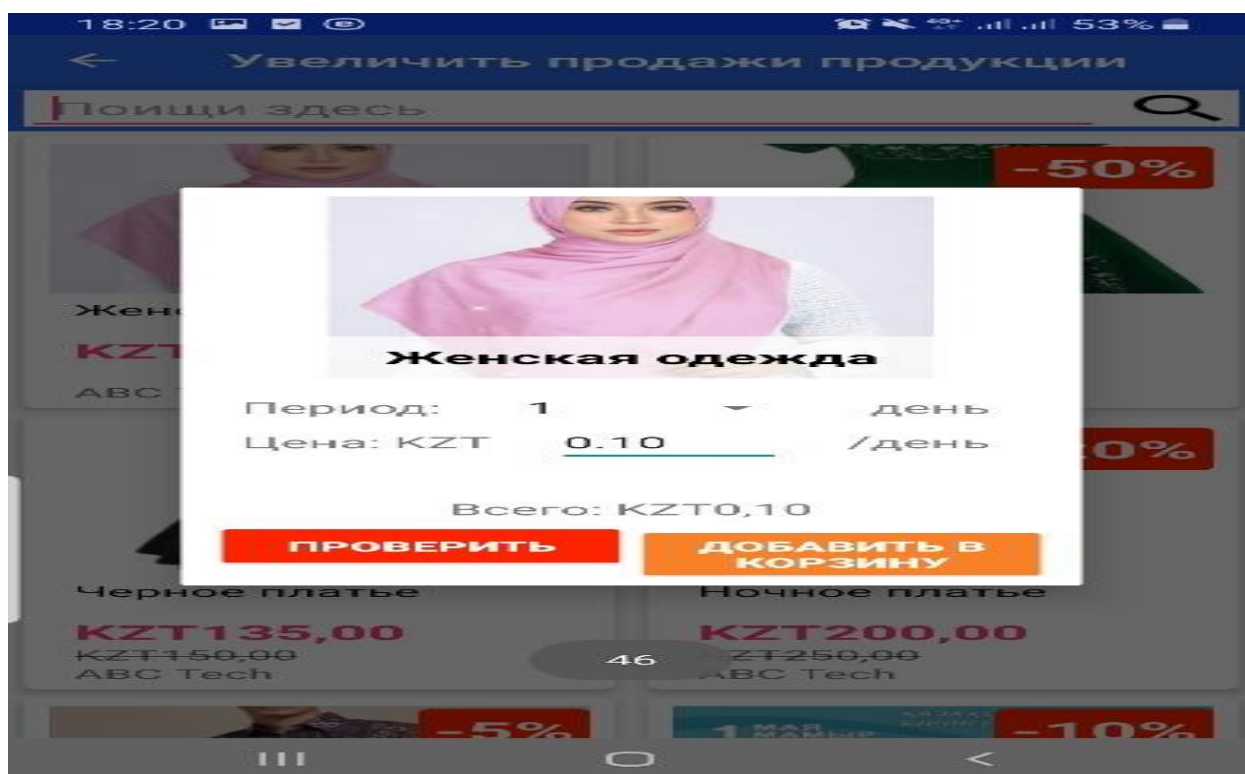
3.6 – сурет. Мобильді қосымшаның «төлем жүйесі» бөлімі



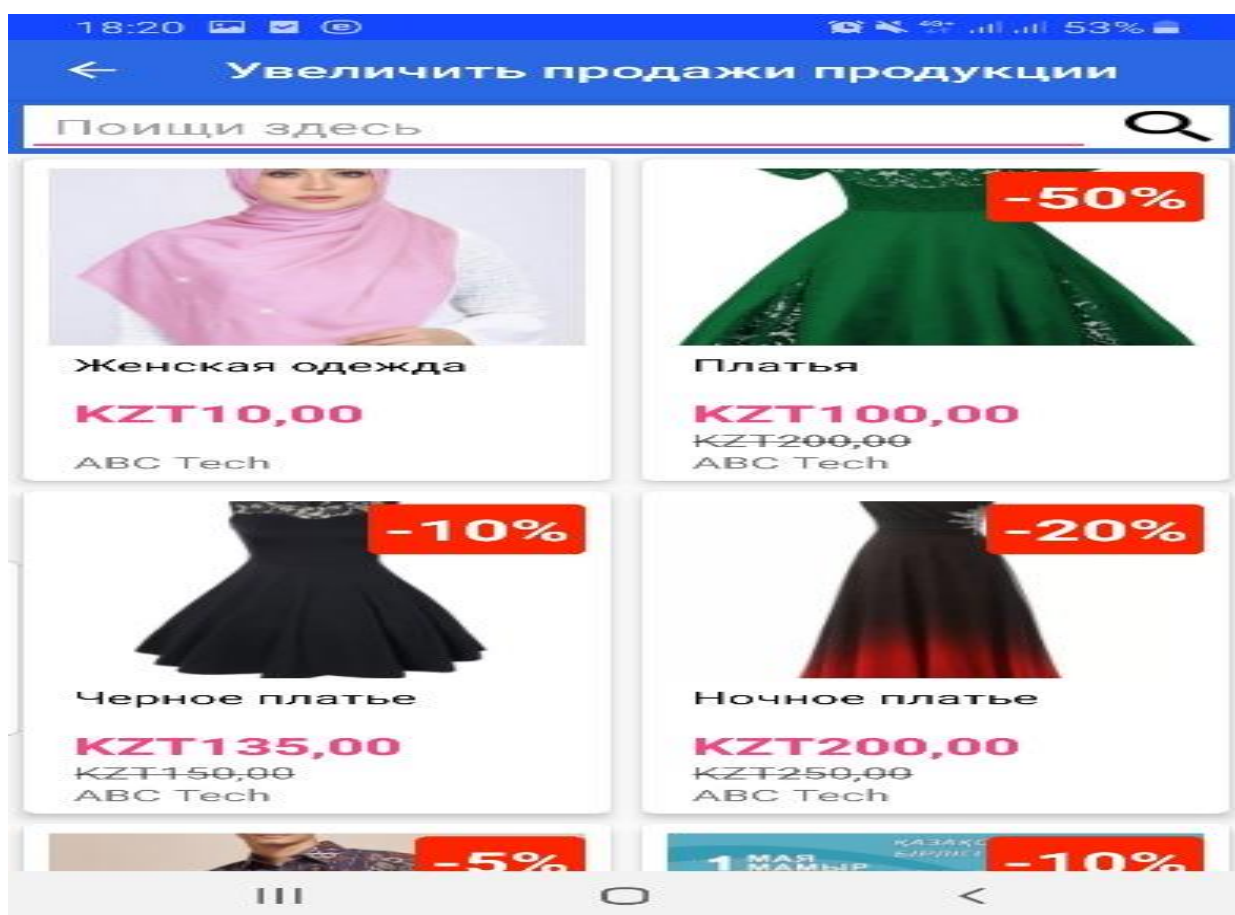
3.7-сурет – Мобильді қосымшаның «төлем жүйесі кезіндегі үрдіс» бөлімі



3.8-сурет – Мобильді қосымшаның «өнімдер тізімінің» бөлімі



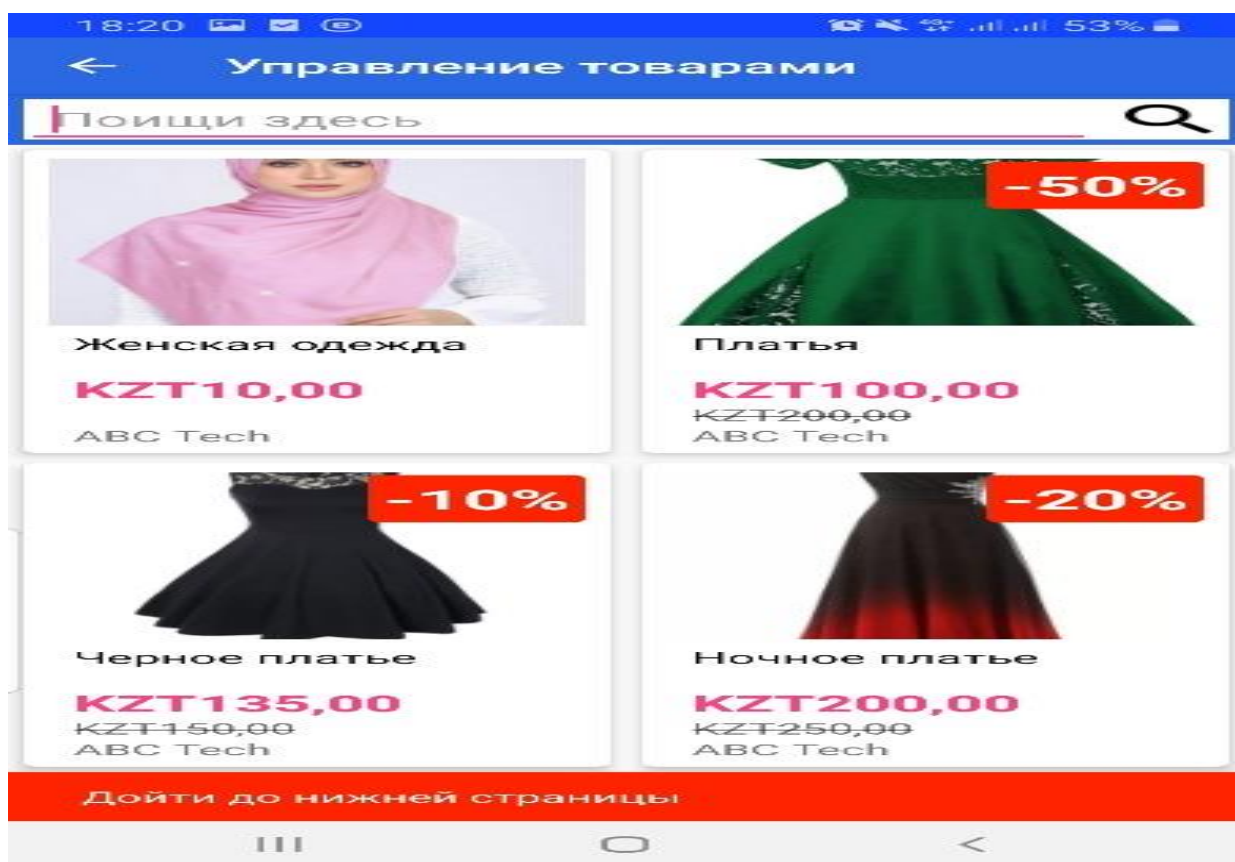
3.9-сурет – Мобильді қосымшаның «өнімді таңдау» бөлімі



3.10-сурет – Мобильді қосымшаның «өнімдер» бөлімі






3.11-сурет – Мобильді қосымшаның «өнімді өңдеу» бөлімі



3.12-сурет – Мобильді қосымшаның «управление товарами» бөлімі

18:19 53%

← Зарегистрировать детали продугу...

Категория
Электронные а..

Размер товара
small

Название продукта

Детали варианта продукта

Тип	Вариант	Количество	Количество
KZT	Цена	Скидка	%

ДОБАВИТЬ ВАРИАНТ

Описание товара

ОТПРАВИТЬ

3.13-сурет – Мобильді қосымшаның «жаңа өнімді енгізу» бөлімі

18:19 53%

← Состояние доставки товара

Поиски здесь

Выбранная дата: 2019-5-3

номер заказа : 32 ready to collect

 Вещь: Adidas Yeezy Boost 350 V2
Вариант : black
Количество : 1
Клиент: Eric Tam
Дата заказа: 2018-08-03 15:45:37

номер заказа : 33 delivered

 Вещь: MSI GTX 1080 ti Gaming X
Вариант : 12GB
Количество : 1
Клиент: Eric Tam
Дата заказа: 2018-08-03 15:55:07

номер заказа : 34 delivered

 Вещь: Adidas Yeezy Boost 350 V2
Вариант : black
Количество : 1

3.14-сурет – Мобильді қосымшаның «жеткізу мәртебесі» бөлімі



3.15-сурет – Мобильді қосымшаның «өнімдердің қойма» бөлімі



3.16-сурет – Жұмыс үстелінде жұмыс істейтін сервер

3.4 Қажетті техникалық жабдықтар

«Өнім тапсырысы android мобильдік қосымшасын» құру үшін Intel Core i3 Duo 3.4 GHz / RAM 2048 Mb / HDD 2048 Gb/ VideoCard 1024 Mb / HP Compaq ноутбугын қолдандым.

3.5 Шақыру және жүктеу

Өнім тапсырысы android мобильдік қосымшасын Пуск → Программы → Adnroid studio → FYR жобасын ашамыз.

3.6 Кіріс мәліметтер

Мобильді қосымшаның кіріс мәліметтері арқылы бағдарламаға пайдалушының негізгі функционал батырмаларын басуын келтіреміз.

3.7 Шығыс мәліметтер

Мобильді қосымшаның шығыс мәліметтері арқылы мәліметтерді беру, нәтиже ретінде алуға болады.

ҚОРЫТЫНДЫ

«Өнім тапсырысы» бағдарламалық өнімін әзірлеу кезінде тапсырма толығымен орындалды. Бағдарлама арнайы бағдарламамен жұмыс істеуге мүмкіндік беретін бағдарламамен жұмыс істеуді жеңілдететін ыңғайлы интерфейске ие.

«Өнім тапсырысы» бағдарламасы - жылдам тамақтану мейрамханасының мен өнімдер автоматтандырылған жүйесінің бөлек дамыған бөлігі. Бағдарлама компьютерлік технологиялар бөлімінде толықтырылып отырады, соның арқасында бағдарламалық жасақтама тапсырыс беруші бөлімшелерінің ғана емес, бүкіл мейрамхананың жұмысын, өнім сатып алу жұмысын жеңілдетеді.

Бүгінгі күні адамға ұялы телефон, планшет, смартфон немесе басқа портативті мультимедиялық құрылғысыз елестету қиын. Мобильді құрылғылардың сатылымы бүкіл әлемде өсіп келе жатқандықтан, олар үшін әртүрлі қосымшаларға сұраныс бар. Әрбір өзін-өзі құрметтейтін компания кемінде бір ұялы қосымшаны өз клиентімен бірге ұстауға тырысады. Сондай-ақ, кейбір компаниялардың болуы мобильді құрылғылар мен мамандандырылған бағдарламаларсыз, мысалы, дерекқорларды басқаруға немесе кез-келген уақытта өнім жағдайын бақылап отыруға мүмкіндік бермейді.

Бүгінгі күні мобильдік қосымшаларды әзірлеу құралдары үшін нақты стандарт жоқ. Әрбір өндіруші операциялық жүйені өз құрылғысында пайдаланушыға ерекше және есте сақтауға тырысады, нәтижесінде әртүрлі операциялық жүйелердегі әртүрлі қосымшалардың сыйымдылығы бар.

ПАЙДАЛАНЫЛҒАН ӘДЕБИЕТТЕР ТІЗІМІ

1. Автоматизированные информационные технологии в экономике: Учебник/ Под ред. проф. Г.А. Титоренко. М.: ЮНИТИ, 2015. - 399 с.
2. Бекаревич Ю.Б., Пушкина Н.В. Microsoft Access 2015. - СПб.: "БХВ – Санкт-Петербург", 2015.
3. Белобородова В.А. Бухгалтерский учет с основами экономического анализа: Учебник, М: Финансы и статистика, 2015. - 352 с.
4. Булгакова С.В. Бухгалтерский учет на предприятиях. М., Приор, 2005.
5. Валебникова Н.В., Василевич И.П. Современные тенденции управленческого учета. Бухгалтерский учет. 2015. №18. С.53-59.
6. Вахрушина М.А. Бухгалтерский управленческий учет: Учебник для вузов. - М.: ЗАО "Финстатинформ", 2016. - 533с.;
7. Врублевский Н.Д. Построение системы счетов управленческого учета. Бухгалтерский учет. 2015. №17. С.63-68.
8. Джон Вейкас. Эффективная работа с Microsoft Access 2000. СПб: Питер, 2016. - 1040 с.
9. Друри К. Введение в управленческий и производственный учет: Учеб. пособие/ Пер. с англ. Эрнашвили Н.Д. - 3-е изд. перераб. и доп. - М.: Аудит, 2016. - 783 с.
10. Ивашкевич В.Б. Управленческий учет в информационной системе предприятия / Бух. учет. - 2015. - №4. - с.99-102.
11. Каратыгин С.А. Access 2016. Руководство пользователя с примерами. - М.: Лаборатория Базовых Знаний, 2016.
12. Карпов Б. Microsoft Access 2017: справочник. - СПб: Изд-во "Питер", 2000.
13. Карпова Т.П. Управленческий учет: Учебник для вузов. - М.: Аудит, ЮНИТИ, 2016. - 237с.
14. Керимов В.Э. Управленческий учет: учебник. - М.: Дашков и К°, 2015.
15. Козлова Е. П, Парашутин Н. В., Бабченко Т.Н. Галанина Бухгалтерский учет - 2-е изд., доп. - М.: Финансы и статистика, 2015. - 522с.
16. Корчагина Л.М. Управленческие аспекты бухгалтерского учета на предприятии / Бух. учет. - 2016. - №7. - с.73.
17. Кукукина И.Г. Автоматизация управленческого учета на предприятиях.М., 2015.

```
package com.ecommerce.merchant.fypproject;
import android.app.Activity;
import android.app.Dialog;
import android.app.ProgressDialog;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.SharedPreferences;
import android.net.ConnectivityManager;
import android.net.NetworkInfo;
import android.os.AsyncTask;
import android.provider.Settings;
import android.support.v7.app.AlertDialog;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.GridLayoutManager;
import android.support.v7.widget.LinearLayoutManager;
import android.support.v7.widget.RecyclerView;
import android.support.v7.widget.SimpleItemAnimator;
import android.support.v7.widget.Toolbar;
import android.util.Log;
import android.view.Gravity;
import android.view.MenuItem;
import android.view.MotionEvent;
import android.view.View;
import android.view.inputmethod.InputMethodManager;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.ImageView;
import android.widget.LinearLayout;
import android.widget.TextView;
import android.widget.Toast;
import com.ecommerce.merchant.fypproject.adapter.BoostCartAdapter;
import com.ecommerce.merchant.fypproject.adapter.BoostItem;
import com.ecommerce.merchant.fypproject.adapter.OnItemClick;
import com.ecommerce.merchant.fypproject.adapter.UploadProcess;
import com.crashlytics.android.Crashlytics;
```



```

import com.crashlytics.android.answers.Answers;
import com.crashlytics.android.answers.PurchaseEvent;
import com.crashlytics.android.answers.StartCheckoutEvent;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.iid.FirebaseInstanceId;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import java.io.IOException;
import java.io.Serializable;
import java.math.BigDecimal;
import java.text.DecimalFormat;
import java.util.ArrayList;
import java.util.Currency;
import java.util.HashMap;
import java.util.List;
import java.util.Objects;
import com.ipay.Ipay;
import com.ipay.IpayPayment;
import com.ipay.IpayResultDelegate;
public class BoostCartActivity extends AppCompatActivity implements
OnItemClickListener {
    private RecyclerView recyclerView;
    private RecyclerView.LayoutManager layoutManager;
    private ProgressDialog progressDialog,deleteDialog;
    private CheckBox selectAllChkBox;
    private Button finalChkOutBtn;
    private TextView subtotal,emptyText;
    private List<BoostItem> cartItemList;
    private BoostCartAdapter cartAdapter;
    private boolean doneLoading=false;
    private boolean isEmpty=false;
    private ImageView emptyImg;
    private String uid,gets subtotal;
    private String
PHPURL="http://ecommercefyp.000webhostapp.com/retailer/manage_retailer_product.php";
    private final DecimalFormat df2 = new DecimalFormat("0.00");
    private final String

```

```

merchantkey="JTKBto70Xe",merchantcode="M03216";
    private ResultDelegate resultDeligate;
    private final int
OUT_SUCCESS_IPAY=1,OUT_FAIL_IPAY=2,OUT_CANCEL_IPAY=3;
    public static int ipay_response=0;
    public static String ipay_error_msg="";
    private final String
ipayURL="http://ecommercefyp.000webhostapp.com/retailer/ipay88response.
php";
    public static String ipay_transid="";
    private FirebaseAuth firebaseAuth;
    private ProgressDialog placeOrderDialog;
    private Dialog cancelDialog;
    private AlertDialog alertDialog;
    private String orderid;
    private SharedPreferences pref;
    @Override
    public void onClick(String value, String prodvar, String prodcode, String rid) {
        if(value.equalsIgnoreCase("uncheck")){
            selectAllChkBox.setChecked(false);
        }else if(value.equalsIgnoreCase("check")) {
            selectAllChkBox.setChecked(true);
        }else if(value.equalsIgnoreCase("refresh")){
            deleteDialog.show();
        }else if(value.equalsIgnoreCase("done refresh")){
            JSON_HTTP_CALL();
            callToast("продукт удален");
            Log.d("delete but:", "Продукт удален из корзины");
        }else{
            subtotal.setText("KZT"+df2.format(Double.parseDouble(value)));
            getsubtotal=df2.format(Double.parseDouble(value));
            if(Double.parseDouble(value)==0){
                finalChkOutBtn.setEnabled(false);
            }else{
                finalChkOutBtn.setEnabled(true);
            }
        }
    }
}

```

@Override

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_boost_cart);  
    Toolbar myposttoolbar = findViewById(R.id.boostCartToolbar);  
    setSupportActionBar(myposttoolbar);  
    Objects.requireNonNull(getSupportActionBar()).setTitle(getResources().getString(  
R.string.cartChkout));  
    if(getSupportActionBar() != null) {  
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);  
        getSupportActionBar().setDisplayShowHomeEnabled(true);  
    }  
    FirebaseAuth();  
    progressDialog=new ProgressDialog(BoostCartActivity.this);  
    progressDialog.setCancelable(true);  
    progressDialog.setTitle(R.string.loading);  
    progressDialog.setMessage(getResources().getString(R.string.preparingCart));  
    progressDialog.setProgressStyle(ProgressDialog.STYLE_SPINNER);  
    deleteDialog=new ProgressDialog(this);  
    deleteDialog.setCancelable(false);  
    deleteDialog.setTitle(R.string.deleteItem);  
    deleteDialog.setMessage(getResources().getString(R.string.pleaseWait));  
    deleteDialog.setProgressStyle(ProgressDialog.STYLE_SPINNER);  
    placeOrderDialog=new ProgressDialog(BoostCartActivity.this);  
    placeOrderDialog.setCancelable(false);  
    placeOrderDialog.setTitle(R.string.processingUrOrder);  
    placeOrderDialog.setMessage(getResources().getString(R.string.pleaseWait));  
    placeOrderDialog.setProgressStyle(ProgressDialog.STYLE_SPINNER);  
    selectAllChkBox=findViewById(R.id.checkAll);  
    finalChkOutBtn=findViewById(R.id.payAll);  
    subtotal=findViewById(R.id.subtotalText);  
    emptyImg=findViewById(R.id.shopCartEmptyImg);  
    emptyText=findViewById(R.id.shopCartEmptyText);  
    viewPhotoDialog();  
    recyclerView=findViewById(R.id.cartRecycler);  
    recyclerView.setHasFixedSize(true);  
    RecyclerView.LayoutManager layoutManagerOfrecyclerView = new  
LinearLayoutManager(this);  
    recyclerView.setLayoutManager(layoutManagerOfrecyclerView);
```

```

selectAllChkBox.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if(cartAdapter==null||!doneLoading){
            JSON_HTTP_CALL();
        }else {
            cartAdapter.chkAll(selectAllChkBox.isChecked());
            subtotal.setText("KZT" + df2.format(cartAdapter.calculateTotal()));
        }
    }
});

if (selectAllChkBox.isChecked() && isEmpty) {
    finalChkOutBtn.setEnabled(true);
} else {
    finalChkOutBtn.setEnabled(false);
}
}

});
finalChkOutBtn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if(cartAdapter==null||!doneLoading){
            JSON_HTTP_CALL();
        }else {
            boolean[] checkList=cartAdapter.getCheckedList();
            JSONArray JSONUploadArray=new JSONArray();
            // cartItemList;
            for(int a=0;a<checkList.length;a++){
                if(checkList[a]){ //if checked the item
                    Log.d("cartID", cartItemList.get(a).getRetailerCartID());
                    Log.d("price", cartItemList.get(a).getPrice());
                    Log.d("period",cartItemList.get(a).getPeriod());
                    //get quantity
                    //cartID
                    //product variant
                    JSONObject JSONPlaceOrderProdcut = new JSONObject();
                    try {
                        JSONPlaceOrderProdcut.put("cartID",
cartItemList.get(a).getRetailerCartID());
                        JSONPlaceOrderProdcut.put("price",

```

```

cartItemList.get(a).getPrice());
JSONPlaceOrderProdcut.put("period",cartItemList.get(a).getPeriod());
JSONPlaceOrderProdcut.put("prodcode",cartItemList.get(a).getProdcode());
JSONUploadArray.put(JSONPlaceOrderProdcut);
        Answers.getInstance().logPurchase(new PurchaseEvent()
.putItemName(cartItemList.get(a).getProdname()+":"cartItemList.get(a).getPr
odname())

        .putItemPrice(new
BigDecimal(cartItemList.get(a).getPrice()))
        .putCurrency(Currency.getInstance("MYR")));
    } catch (JSONException e){
        Crashlytics.logException(e);
        // handle your exception here!
        e.printStackTrace();
    }
}
String orderList=JSONUploadArray.toString();
Log.d("pay item list",orderList);
SharedPreferences pref =
BoostCartActivity.this.getSharedPreferences("MyPref", 0); // 0 - for private
mode

        SharedPreferences.Editor editor = pref.edit();
        editor.clear();
        editor.putString("shop_cart_list", orderList); //save the string array
in share preference and will use it when user done ipay88 payment
        editor.putString("total_amount",Double.toString(cartAdapter.calculateTotal()));
        editor.apply();

//        Intent addressDelivery = new Intent(BoostCartActivity.this,
AddressDeliveryActivity.class);
//        addressDelivery.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
//        startActivity(addressDelivery);
if(isOnline()){
    placeOrderDialog.show();
    JSONObject objectDetail=new JSONObject();
    JSONArray array=new JSONArray();
    try{
        String addressID;
        addressID="1";

```

```

objectDetail.put("addressid",addressID);
                objectDetail.put("delivery","0");
                objectDetail.put("paymentmethod","ipay88");
                objectDetail.put("rid",uid);
                array.put(objectDetail);
            }catch (Exception e){
                Crashlytics.logException(e);
                // handle your exception here!
                e.printStackTrace();
            }
            CREATE_ORDER_JSON_HTTP_CALL(array.toString());
        }else{
            Toast.makeText(BoostCartActivity.this,R.string.pleaseEnableInternet,Toast.LENGTH_SHORT).show();
        }
    }
});
((SimpleItemAnimator)
recyclerView.getItemAnimator()).setSupportsChangeAnimations(false);
JSON_HTTP_CALL();
}
public void JSON_HTTP_CALL() {
    progressDialog.show();
    selectAllChkBox.setChecked(false);
    finalChkOutBtn.setEnabled(false);
    subtotal.setText("KZT0.00");
    cartItemList=new ArrayList<>();
    cartAdapter=null;
    doneLoading=false;
    class AsyncTaskUploadClass extends AsyncTask<Void,Void,String> {
        @Override
        protected void onPreExecute() {
            super.onPreExecute();
        }
        @Override
        protected void onPostExecute(String response) {
            super.onPostExecute(response);
            try {

```

```

        if(response.equalsIgnoreCase("[]")){
            isEmpty=false;
            emptyImg.setVisibility(View.VISIBLE);
            emptyText.setVisibility(View.VISIBLE);
            recyclerView.setVisibility(View.INVISIBLE);
        }else{
            isEmpty=true;
            emptyImg.setVisibility(View.INVISIBLE);
            emptyText.setVisibility(View.INVISIBLE);
            recyclerView.setVisibility(View.VISIBLE);
        }
        ParseJsonResponse(response);
    } catch (JSONException e) {
        Crashlytics.logException(e);
        // handle your exception here!
        e.printStackTrace();
    }
    Log.d("Response", response);
}

@Override
protected String doInBackground(Void... params) {
    UploadProcess ProcessClass = new UploadProcess();
    HashMap<String,String> HashMapParams = new HashMap<>();
    HashMapParams.put("fetchCart", uid);
    return ProcessClass.HttpRequest(PHPURL, HashMapParams);
}

}

AsyncTaskUploadClass AsyncTaskUploadClass = new
AsyncTaskUploadClass();
AsyncTaskUploadClass.execute();
}

//get response string and set into recyclerView
private void ParseJsonResponse(String array) throws JSONException {
    cartItemList=new ArrayList<>();
    cartAdapter=null;
    JSONArray jarr = new JSONArray(array);//lv 1 array
    int totalQty=0;
    for(int a=0;a<jarr.length();a++){
        BoostItem x=new BoostItem();

```



```

JSONObject json;
json=jarr.getJSONObject(a);
x.setPeriod(json.getString("period"));
x.setPrice(json.getString("price"));
x.setProcode(json.getString("procode"));
x.setProname(json.getString("proname"));
x.setRetailerCartID(json.getString("cartid"));
x.setUrl(json.getString("url"));
cartItemList.add(x);
}
cartAdapter= new BoostCartAdapter(cartItemList, this,this);
RecyclerView.Adapter recyclerViewadapter = cartAdapter;
recyclerView.setAdapter(recyclerViewadapter);
recyclerView.addOnItemTouchListener(new
RecyclerView.OnItemTouchListener() {
    @Override
    public boolean onInterceptTouchEvent(RecyclerView rv, MotionEvent e)
{
    return false;
}
    @Override
    public void onTouchEvent(RecyclerView rv, MotionEvent e) {
    }
    @Override
    public void onRequestDisallowInterceptTouchEvent(boolean
disallowIntercept) {
    }
});
progressDialog.dismiss();
doneLoading=true;
if(deleteDialog.isShowing()){
    //subtotal.setText(df2.format(cartAdapter.calculateTotal()));
    selectAllChkBox.setChecked(false);
    cartAdapter.notifyItemRangeChanged(0,cartItemList.size());
    deleteDialog.dismiss();
}
}
public boolean onOptionsItemSelected(MenuItem item){
    if (item.getItemId()==android.R.id.home)

```

```

        finish();
        return super.onOptionsItemSelected(item);
    }
    private void GetFirebaseAuth(){
        firebaseAuth = FirebaseAuth.getInstance();
        if(firebaseAuth.getCurrentUser()==null){
            Intent intent = new Intent(getApplicationContext(), SplashActivity.class);
            try {
                FirebaseInstanceId.getInstance().deleteInstanceId();
            } catch (IOException e) {
                e.printStackTrace();
            }
            intent.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK |
Intent.FLAG_ACTIVITY_CLEAR_TASK);
            startActivity(intent);
            Toast.makeText(this,getString(R.string.sessionexp),Toast.LENGTH
H_LONG).show();
        }else uid = firebaseAuth.getCurrentUser().getUid();
    }
    private void callToast(String x){
        Toast.makeText(BoostCartActivity.this,x,Toast.LENGTH_LONG).show();
    }
    static public class ResultDelegate implements IpayResultDelegate,Serializable
    {
        private int STATUS=0;
        private final int SUCCESS_IPAY=1;
        private final int FAIL_IPAY=2;
        private final int CANCEL_IPAY=3;
        public void onPaymentSucceeded (String transId, String refNo, String
amount, String remarks, String auth)
        {
            STATUS=SUCCESS_IPAY;
            ipay_response=STATUS;
            ipay_transid=transId;
            Log.e("transaction ID",transId);
            Log.e("transaction ID",transId);
            Log.e("transaction ID",transId);
            Log.e("transaction ID",transId);
            Log.e("transaction ID",transId);

```

```

        Log.e("transaction ID",transId);
        Log.e("transaction ID",transId);
        Log.e("transaction ID",transId);
    }
    public void onPaymentFailed (String transId, String refNo, String amount,
String remarks, String err)
    {
        STATUS=FAIL_IPAY;
        // STATUS=FAIL_IPAY;
        ipay_response=STATUS;
        // mCallback.onClick("fail ipay");
        Log.d("ipay fail response:",err);
        Log.d("ipay fail response:",err);
        Log.d("ipay fail response:",err);
        Log.d("ipay fail response:",err);
        Log.d("ipay fail response:",err);
        Log.d("ipay fail response:",err);
        Log.d("ipay fail response:",err);
        Log.d("ipay fail response:",err);
        Log.d("ipay fail response:",err);
        Log.d("ipay fail response:",err);
        Log.d("ipay fail response:",err);
        Log.d("ipay fail response:",err);
        Log.d("ipay fail response:",err);
        ipay_error_msg=err;
    }
    public void onPaymentCanceled (String transId, String refNo, String amount,
String remarks, String errDesc)
    {
        //
        STATUS=CANCEL_IPAY;
        // STATUS=CANCEL_IPAY;
        // mCallback.onClick("success ipay");
        // mCallback.onClick("fail ipay");
        ipay_response=STATUS;
        Log.d("ipay cancel response:",errDesc);
        Log.d("ipay cancel response:",errDesc);
        Log.d("ipay cancel response:",errDesc);
        Log.d("ipay cancel response:",errDesc);

```

```

        Log.d("ipay cancel response:",errDesc);
        Log.d("ipay cancel response:",errDesc);
        Log.d("ipay cancel response:",errDesc);
        Log.d("ipay cancel response:",errDesc);
        Log.d("ipay cancel response:",errDesc);
        Log.d("ipay cancel response:",errDesc);
        Log.d("ipay cancel response:",errDesc);
        Log.d("ipay cancel response:",errDesc);
        Log.d("ipay cancel response:",errDesc);
        ipay_error_msg=errDesc;
    }
    public void onRequeryResult (String merchantCode, String refNo, String
amount, String result)
    {
        Log.e("ipay requery:",result);
        Log.e("ipay requery:",result);
        Log.e("ipay requery:",result);
        Log.e("ipay requery:",result);
        Log.e("ipay requery:",result);
        Log.e("ipay requery:",result);
        Log.e("ipay requery:",result);
        Log.e("ipay requery:",result);
    }
}
//create order here, to get order id
    public void CREATE_ORDER_JSON_HTTP_CALL(final String orderDetail)
    {
        class AsyncTaskUploadClass extends AsyncTask<Void,Void,String> {
            @Override
            protected void onPreExecute() {
                super.onPreExecute();
            }
            @Override
            protected void onPostExecute(String response) {
                super.onPostExecute(response);
                try {
                    CREATE_ORDER_ParseJSonResponse(response);
                    //orderid=response;
                } catch (Exception e) {
                    Crashlytics.logException(e);

```

```

        // handle your exception here!
        e.printStackTrace();
    }
    Log.e("ResponseCreateOrder", response);
    Log.e("ResponseCreateOrder", response);
    Log.e("ResponseCreateOrder", response);
}
@Override
protected String doInBackground(Void... params) {
    UploadProcess ProcessClass = new UploadProcess();
    HashMap<String,String> HashMapParams = new HashMap<>();
    HashMapParams.put("createOrder",orderDetail);
    return ProcessClass.HttpRequest(PHPURL, HashMapParams);
}
}
AsyncTaskUploadClass AsyncTaskUploadClass = new
AsyncTaskUploadClass();
AsyncTaskUploadClass.execute();
}
public void CREATE_ORDER_ParseJsonResponse(String array) throws
JSONException {
    JSONArray jarr = new JSONArray(array);//lv 1 array
    JSONObject json;
    json=jarr.getJSONObject(0);
    String orderdate=json.getString("orderdate");
    orderid=json.getString("orderid");
    pref = BoostCartActivity.this.getSharedPreferences("MyPref", 0); // 0 - for
private mode
    SharedPreferences.Editor editor = pref.edit();
    //editor.clear();
    editor.putString("currentOrderid", orderid); //save the string array in
share preference and will use it when user done ipay88 payment
    editor.apply();
    resultDeligate=new ResultDelegate();
    //ipay88 setting
    final IpayPayment payment = new IpayPayment();
    payment.setMerchantKey (merchantkey);
    payment.setMerchantCode (merchantcode);
    payment.setPaymentId ("2");

```

```

payment.setCurrency ("MYR");
//total amount here, unlock this when it launch,TQ
String ipayTotalAmount=pref.getString("total_amount", null);
Log.e("total amount here:",ipayTotalAmount);
Log.e("total amount here:",ipayTotalAmount);
Log.e("total amount here:",ipayTotalAmount);
Log.e("total amount here:",ipayTotalAmount);
payment.setAmount ("1");
//show all item string here
pref = BoostCartActivity.this.getSharedPreferences("MyPref", 0); // 0 - for
private mode
orderid=pref.getString("currentOrderid",null);
payment.setProdDesc ("Order: "+orderid);
payment.setUsername
(Objects.requireNonNull(firebaseAuth.getCurrentUser()).getEmail());
payment.setUserEmail (firebaseAuth.getCurrentUser().getEmail());
payment.setUserContact ("test contact");
payment.setRemark ("retailer pay boost");
payment.setCountry ("MY");
payment.setBackendPostURL (ipayURL);
payment.setRefNo (orderid+orderid);
Intent checkoutIntent = Ipay.getInstance().checkout(payment,
BoostCartActivity.this, resultDeligate);
startActivityForResult(checkoutIntent, 223);
}
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
super.onActivityResult(requestCode, resultCode, data);
if (resultCode == RESULT_OK) {
if(requestCode==223){
switch (ipay_response){
case OUT_SUCCESS_IPAY:
pref = BoostCartActivity.this.getSharedPreferences("MyPref", 0);
// 0 - for private mode
orderid=pref.getString("currentOrderid",null);
String orderList=pref.getString("shop_cart_list", null);
Log.e("success ipay",ipay_transid);
JSON_HTTP_SEND("cartList",orderList,orderid,ipay_transid);
break;

```

```

default:
    CANCEL_ORDER_JSON_HTTP_CALL(orderid);
    placeOrderDialog.dismiss();
    Answers.getInstance().logPurchase(new PurchaseEvent()
        .putSuccess(false)
        .putCustomAttribute("Failed Reason",ipay_error_msg));
    Toast.makeText(BoostCartActivity.this,"Transaction
fail"+ipay_error_msg,Toast.LENGTH_SHORT).show();
    finish();
}

//      if(ipay_response==OUT_SUCCESS_IPAY){ //perform server update
//      only when success payment is made
//
//
//
//      }else{
//          placeOrderDialog.dismiss();
//
//      Toast.makeText(AddressDeliveryActivity.this,"TransactionFailed",Toast.LENGTH
//      _SHORT).show();
//
//
//
//      }
    }else{
        CANCEL_ORDER_JSON_HTTP_CALL(orderid);
        placeOrderDialog.dismiss();
        Toast.makeText(BoostCartActivity.this,"Unexpected
error"+ipay_error_msg,Toast.LENGTH_SHORT).show();
        Answers.getInstance().logPurchase(new PurchaseEvent()
            .putSuccess(false)
            .putCustomAttribute("Failed Reason",ipay_error_msg));
        finish();
    }
} else if(requestCode==124){
    //Toast.makeText(this,"recreate activity",Toast.LENGTH_SHORT).show();
    Intent intent = getIntent();
    finish();
    startActivity(intent);
} else{

```



```

        CANCEL_ORDER_JSON_HTTP_CALL(orderid);
        placeOrderDialog.dismiss();
        Toast.makeText(BoostCartActivity.this, "Transaction
fail", Toast.LENGTH_SHORT).show();
        Answers.getInstance().logPurchase(new PurchaseEvent()
            .putSuccess(false)
            .putCustomAttribute("Failed Reason", "Cancel by user"));
        cancelDialog.show();
    }
}

public void CANCEL_ORDER_JSON_HTTP_CALL(final String orderid) {
    class AsyncTaskUploadClass extends AsyncTask<Void, Void, String> {
        @Override
        protected void onPreExecute() {
            super.onPreExecute();
        }
        @Override
        protected void onPostExecute(String response) {
            super.onPostExecute(response);
            Log.d("Response", response);
        }
        @Override
        protected String doInBackground(Void... params) {
            UploadProcess ProcessClass = new UploadProcess();
            HashMap<String, String> HashMapParams = new HashMap<>();
            HashMapParams.put("cancelOrder", orderid);
            return ProcessClass.HttpRequest(PHPURL, HashMapParams);
        }
    }
    AsyncTaskUploadClass AsyncTaskUploadClass = new
AsyncTaskUploadClass();
    AsyncTaskUploadClass.execute();
}

//send order details to own server
public void JSON_HTTP_SEND(final String name, final String orderList, final
String ORDERID, final String ipayid) {
    class AsyncTaskUploadClass extends AsyncTask<Void, Void, String> {
        @Override
        protected void onPreExecute() {

```

```

        super.onPreExecute();
    }
    @Override
    protected void onPostExecute(String response) {
        super.onPostExecute(response);
        Log.e("Response.....",response);
        Answers.getInstance().logPurchase(new PurchaseEvent()
            .putSuccess(true));
        placeOrderDialog.dismiss();
        finish();
    }
    @Override
    protected String doInBackground(Void... params) {
        UploadProcess ProcessClass = new UploadProcess();
        HashMap<String,String> HashMapParams = new HashMap<>();
        HashMapParams.put(name, orderList);// name=cartList
        // HashMapParams.put("orderobj",orderobj);
        HashMapParams.put("rid",uid);
        HashMapParams.put("orderid",ORDERID);
        HashMapParams.put("ipay_transid",ipayid);
        return ProcessClass.HttpRequest(PHPURL, HashMapParams);
    }
}
AsyncTaskUploadClass AsyncTaskUploadClass = new
AsyncTaskUploadClass();
AsyncTaskUploadClass.execute();
}
//cancel order dialog prompt
private void viewPhotoDialog(){
    cancelDialog = new Dialog(this, R.style.MaterialDialogSheet);
    cancelDialog.setContentView(R.layout.cancel_dialog); // your custom view.
    cancelDialog.setCancelable(true);
    Objects.requireNonNull(cancelDialog.getWindow()).setLayout(LinearLayout.Lay
outParams.MATCH_PARENT, LinearLayout.LayoutParams.WRAP_CONTENT);
    cancelDialog.getWindow().setGravity(Gravity.CENTER);
    Button helpBut=cancelDialog.findViewById(R.id.cancelHelpBut);
    Button closeHelpBut = cancelDialog.findViewById(R.id.cancelCloseBut);
    closeHelpBut.setOnClickListener(new View.OnClickListener() {
        @Override

```

```

        public void onClick(View v) {
            cancelDialog.dismiss();
        }
    });
    helpBut.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            alertDialog=new AlertDialog.Builder(BoostCartActivity.this)
                .setMessage(R.string.internetErrorTip)
                .setCancelable(false)
                .setPositiveButton(R.string.close,null)
                .setNeutralButton(R.string.setting, new
DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialog, int which) {
                        cancelDialog.dismiss();
                        startActivityForResult(new
Intent(Settings.ACTION_WIRELESS_SETTINGS),124);
                        alertDialog.dismiss();
                    }
                }).show();
        }
    });
}

protected boolean isOnline() {
    ConnectivityManager cm =
(ConnectivityManager)getSystemService(Context.CONNECTIVITY_SERVICE);
    NetworkInfo netInfo = Objects.requireNonNull(cm).getActiveNetworkInfo();
    return netInfo != null && netInfo.isConnectedOrConnecting();
}
}

```