

«М.А. Гендельманның 110 жылдығына арналған «Сейфуллин оқулары – 19» халықаралық ғылыми-практикалық конференциясының материалдары = Материалы международной научно-практической конференции «Сейфуллинские чтения – 19», посвященной 110 - летию М.А. Гендельмана» - 2023.- Т. II, Ч.1.- Б. 320-323.

ОӘК 004.056.2

«ЭЛЕКТРОНДЫ КІТАПТАРДЫ ОҚУҒА ЖӘНЕ ЖҮКТЕП АЛУҒА АРНАЛҒАН КРОСС-ПЛАТФОРМАЛЫҚ МОБИЛЬДІ ҚОСЫМШАНЫ ӘЗІРЛЕУ»

Амырканова Камила Кадыркановна

4 курс студенті

Таныкпаева Балауса Ерденовна.,

*Ақпараттық жүйелер кафедрасының аға оқытушысы
«С.Сейфуллин атындағы Қазақ агротехникалық зерттеу
университеті», КеАҚ Астана қ.*

Қазіргі қоғамда ақпараттық және мобильді технологиялар жылдам қарқынмен дамып, мобильді технология мен интернеттің біздің өмірімізге әсері артып келеді. Қазір сауатсыз адам оқуды, жазуды білмейтін адам емес, ұялы телефонды, компьютерді, интернетті пайдалана алмайтын адам саналады, бірақ мұндай адамдар жылдан жылға азайып барады.

Бүгінгі таңда ақпаратқа мобильді және жылдам қол жеткізуді көбірек адамдар қалайды. Жыл сайын сайттардың мобильді нұсқаларын пайдаланушылардың пайызы өсуде, сондықтан компьютерлік нұсқалар аз танымал бола бастады [1].

Сандық дәуірдегі көптеген бағдарламалық өнімдер, ұялы телефондарға арналған. Электрондық құрылғылардың соңғы жетілдірілген модельдерінің көпшілігі функционалдығы мен сипаттамалары бойынша көлемді ұялы телефондар мен компьютерлерден кем түспейді. Біз телефондарды кез-келген күнделікті жұмыс үшін қолданамыз: біз интернеттен жылдам және қол жетімді ақпаратты іздейміз, қажетті мекен-жайды іздейміз, жаттығу кестесін жүргіземіз, отбасылық бюджетті жоспарлаймыз. Мобильді қосымшалар біздің өміріміздің ажырамас бөлігі болып табылады. Nielsen мәліметтері бойынша, пайдаланушылар оларда айына орта есеппен 30 сағат жұмсайды.

Тұтынушылар мобильді қолданбалар мен веб-сайттарды көбірек пайдаланады және маркетингтер мұны есте ұстауы керек. Бүгінгі күні ұялы телефондардың негізгі классификациясы: BlackBerry, SymbianOS, Windows Mobile, Android, IOS және т.б. Android және IOS ОЖ ең танымал болып саналады, өйткені олар мобильді құрылғылар нарығының 98% алады. Мобильді қосымшаның пайдаланушы интерфейсі жүйенің маңызды бөлігі болып табылады. Қолданбаларды әзірлеу ландшафты пайдаланушы интерфейстерін жасауға арналған технологиялар мен шешімдерге толы. Ішкі

әзірлемелермен қатар қолданба интерфейстерін жасау үшін көптеген кросс-платформалық, веб, гибридті және тіпті кодталмаған құралдар бар. Бұл үшін арнайы құралды Flutter фреймворкті қарастырылады [2].

Flutter-бұл iOS және Android үшін мобильді қосымшаларды оңай және жылдам жасауға мүмкіндік беретін Google әзірлеген ашық бағдарламалық жасақтама. Фреймворк - бұл бір жол арқылы Dart жобасына қосуға болатын дайын абстракциялардың, код бөлімдерінің, функциялардың және әдістердің жиынтығы. Сонымен қатар, Flutter жұмысында жергілікті компоненттерді мүлдем пайдаланбайды. Оның орнына жақтаудағы барлық UI элементтері өздерінің графикалық қозғалтқышымен жасалады. Flutter қолданбаның барлық интерфейс элементтерін дайын виджеттерден жасауға мүмкіндік береді. Бұл жағдайда Flutter басқа құрылымдарға ұқсас-React және Vue, сонымен бірге олардан бірқатар айырмашылықтарға ие. Сонымен, ол Javascript бағдарламалау тілін қолданбайды, оның орнына Flutter Dart тілін қолданады.

Flutter қосымшалары Dart бағдарламалау тілінде жасалады. Ол осы тілдің көптеген жетілдірілген функцияларын қолданады. Flutter жұмыс уақытында, MacOS, Linux, Windows жүйелерінде және Flutter Desktop Embedding жобасы аясында құрастыру механизмі бар Dart виртуалды машинасында жұмыс істейді. Flutter қосымшаны жазу және күйін келтіру кезінде jit салыстыруын қолданады. Бұл пайдаланушыларға жұмыс істеп тұрған қолданбаға бастапқы файлға өзгертулер енгізуге мүмкіндік беретін ыстық қайта жүктеуге мүмкіндік береді. Flutter ыстық қайта жүктеу кезінде күйді бақылау виджеттерін қолдауды кеңейтеді. Бұл бастапқы кодтың өзгеруін күйді жоғалтпай немесе қайта қосусыз жұмыс істеп тұрған қосымшаларда көрсетуге мүмкіндік береді. Flutter қосымшасының шығарылым нұсқалары iOS және Android жүйелерінде ерте құрастыру (AOT) көмегімен құрастырылған. Бұл мобильді құрылғыларда платформаның жоғары өнімділігін қамтамасыз етеді. Flutter Github-та 121 мың жұлдыз жинады және Stackoverflow сауалнамасы бойынша 2021 жылы ең танымал фреймворктардың, кітапханалардың және құралдардың үштігіне енді.

Flutter-де жазылған қосымшалардың мысалы ретінде келесі қосымшаларды қарастыруға болады: Google Ads, Alibaba, Square, Ebay, Hamilton Musical, Reflectly, Groupon, Cryptomaniac, SpaceX Go, Realtor.com.

Flutter екі маңызды бөлікке бағытталған:

SDK (Software Development Kit): қолданбаларды әзірлеуге көмектесетін құралдар жинағы. Ол жергілікті машиналық кодта, кодты құрастыру құралдарын қамтиды (iOS және Android үшін код).

Framework (виджеттерге негізделген пайдаланушы интерфейсіннің кітапханасы): жеке қалау бойынша жекелендіруге болатын функционалды интерфейс элементтерінің жиынтығы (түймелер, мәтіндік кірістер, жүгірткілер және т.б.) [3].

Негізінен c++ - та жазылған Flutter қозғалтқышы Google-дің skia графикалық кітапханасының төмен деңгейлі рендерингіне қолдау көрсетеді. Сонымен қатар, ол белгілі бір платформалар үшін SDK-мен, мысалы, iOS

және Android ұсынған платформалармен өзара әрекеттесе алады. Flutter Engine-бұл Flutter қосымшаларын орналастыруға арналған портативті жұмыс уақыты. Қозғалтқыш Flutter кітапханаларын жүзеге асырады, мысалы, файл және желілік енгізу-шығару, анимация және графика, архитектура, қол жетімділікті қолдау, сонымен қатар бірнеше Dart жұмыс уақыты мен құрастыру компоненттері. Көптеген әзірлеушілер Flutter-мен өзара әрекеттесу үшін Flutter Framework пайдаланады. Flutter фреймворк икемді құрылымды, сонымен қатар орналасу виджеттерін, негіздерді және платформаларды ұсынады. Dart - та жазылған Foundation кітапханасы Flutter қосымшаларын құру үшін қолданылатын көптеген іргелі функциялар мен сыныптарды ұсынады, соның ішінде API қозғалтқышпен өзара әрекеттесуді жеңілдететін интерфейстер. Кітапханада сонымен қатар Android және iOS енгізулері бар арнайы дизайн виджеттері бар.

Flutter негізінен Dart бағдарламалау тілінде жазылған. Dart-қатаң терілген объектіге бағытталған тіл. Flutter React Native-ге ұқсас және реактивті және декларативті бағдарламалау стиліне ие. Flutter қолданбаның іске қосылу уақыты мен өнімділігін арттыру үшін JavaScript көпірін қажет етпейді. Dart AoT компиляциясын Just-in-time компиляциясымен бірге ұсынады. Flutter жұмыс процесін жақсарту үшін JIT компиляциясын қолданады. Бұл толық жаңа құрастыруды жасамай-ақ, әзірлеу кезінде ыстық қайта жүктеуді және пайдаланушы интерфейсін жаңартуды жеңілдету арқылы қол жеткізіледі. Екі құрылым да мобильді қосымшаларды әзірлеу үшін қолданылады. Тұтастай алғанда, React Native-те Flutter виджет кітапханасына қарағанда UI элементтерінің кең кітапханасы бар [4].

Дегенмен, бұл жағдайда соңғысының артықшылығы-бұл React Native сияқты үшінші тарап элементтер кітапханаларына тәуелді емес. Олардағы кейбір элементтер белгілі бір платформалармен үйлеспейтіндігін көрсетеді. Бұл жағдайда Flutter әмбебап және кеңінен қолданылады деп айтуға болады. Сонымен қатар, Flutter React Native-тен асып түседі және өнімділігі жағынан өте жақсы көрсету тәсілін қолданады. Сонымен, Flutter өзінің виджеттерін жасайды және графикалық процессорды басқа платформалардан жергілікті компоненттерді алудың орнына, рендеринг үшін пайдаланады.

Dart тілінде жазылған Flutter коды процессормен тікелей өңделетін ARM кодын құрастырады. Осының арқасында Flutter-де жасалған қолданбалар айтарлықтай жылдамырақ жұмыс істейді. Ал React Native-де UI элементтерін түсіндіру үшін қолданылатын және сәйкесінше iOS және Android компоненттерін көрсету үшін Java API немесе Objective-C шақыратын Javascript көпірі қосымшаларды баяулатуы мүмкін.

Жергілікті рендеринг қозғалтқышы — Flutter әзірлеушілерге басқа платформаларға қарағанда көбірек қол жеткізуге мүмкіндік береді. Ол қуатты платформалық көрсету механизмі сияқты қуатты құрылымдармен жақсы нәтиже береді. Flutter платформаның инфрақұрылымына өзін-өзі көрсету үшін Skia пайдаланады. Оның қозғалтқышы кез-келген платформада Flutter көмегімен жасалған пайдаланушы интерфейсін іске қосуға мүмкіндік береді.

Flutter пайдаланушылары дамуды жеңілдету үшін платформаны тасымалдау үшін пайдаланушы интерфейсін конфигурациялаудың қажеті жоқ [5].

Мобильді қосымшаны құру технологиялары, сондай-ақ әзірлеу кезінде ескеру қажет кейбір қадамдар талданды. Қолданыстағы платформалар мен бағдарламалау тілдеріне егжей-тегжейлі талдау жасалды. Dart бағдарламалау тілі мен Flutter фреймворкі қарастырылады.

Пайдаланған әдебиеттер тізімі:

- 1 Скотт А. Разработка на JavaScript. Построение кроссплатформенных приложений с помощью GraphQL, React, React N. – 2021. – С. 320
<https://revall.info/razrabotka-na-javascript-postroenie-krossplatformennyh-prilozhenij-s-pomoshhyu-graphql-react-react-native-i-electron.html>
- 2 Фрэнкс Б. Укрощение больших данных: как извлекать знания из массивов информации с помощью глубокой аналитики [Текст] / Билл Фрэнкс; пер. с англ. Андрея Баранова. М. : Манн, Иванов и Фербер. - 2014. – С. 352
https://f.ua/statik/files/products/515942/ukrosheenie-bolshih-danniyh-9785000571460_534.pdf
- 3 Вайгенд А. BIG DATA. Вся технология в одной книге. «Издательство «Эксмо». Москва. - 2018. – С. 220
https://docs.google.com/gview?url=https://www.phantastike.com/other/big_data/doc/
- 4 Марц Н, Уоррен Д. Большие данные. Принципы и практика построения масштабируемых систем обработки данных в реальном времени [Текст] / Издательский дом "Вильямс" Москва. Санкт Петербург. - 2016. – С. 268
<http://i.uran.ru/webcab/system/files/bookspdf/bolshie-dannye/bolshiedannye.pdf>
- 5 Topka, V.V. The Cross-Impact Analysis of Innovative Projects in a Portfolio. J. Comput. Syst. Sci. Int. 58, 736–746 (2019).
(<https://doi.org/10.1134/S1064230719050149>)