

дамытудың мемлекеттік бағдарламасы болып табылады.

Кедейшілікпен күрестегі ең басты мәселе – бұл жұмыссыздық деңгейін төмендету. Егер басымдықтарды белгілейтін болсақ, мынадай: жұмысқа орналастыруға жәрдемдесу, кәсіптік дайындау кезінде оқудың, қайтадан оқудың күнын төлеу, кәсіби біліктілігін арттыру және қайтадан дайындау, қоғамдық жұмыстарды ұйымдастыру және еңбек ақысын төлеу, шағын несие беру үшін жағдай туғызуға, кедейшілікке душар болғанда материалдық көмек көрсету, шағын және орта бизнес саласын жұмыспен қамтуға қолбаныс беру.

УДК 021.63

Айсу Бисеновна Касекеева¹, Надежда Олеговна Мишунина²

1) старший преподаватель кафедры «Информационных систем» Казахского Агротехнического университета им.С.Сейфуллина, магистр математики.

2) старший преподаватель кафедры «Информационных систем» Казахского Агротехнического университета им.С.Сейфуллина.

МОБИЛЬДІ ҚОСЫМШАЛАРДЫ ӨЗІРЛЕУ КЕЗІНДЕГІ КРОССПЛАТФОРМАЛЫ ҚҰРАЛДАРҒА ТАЛДАУ ЖАСАУ

Аңдатпа: Мақалада кроссплатформалы қосымшалардың ерекшеліктері сипатталған. Жүйелік бағдарламаны дайындау кезінде бейімдейтін талаптар көрсетілген. Кітапханалардың және фреймворктардың салыстырмалы сипаттамасы берілген. Qt C++ арналған бағдарламалар жасау үшін Eclipse кроссплатформалы фреймворк java - қосымша өзірлеу ортасының (IDE) интеграцияланған қысқаша сипаттамасы жазылған.

Abstract: In article differences of cross-platform appendices from the native are briefly described. Requirements with which emergence it is necessary to adapt the developed program system are allocated. The comparative characteristic of libraries and frameworks is given. The short characteristic of the integrated environment of development (IDE) of Java-applications of Eclipse is

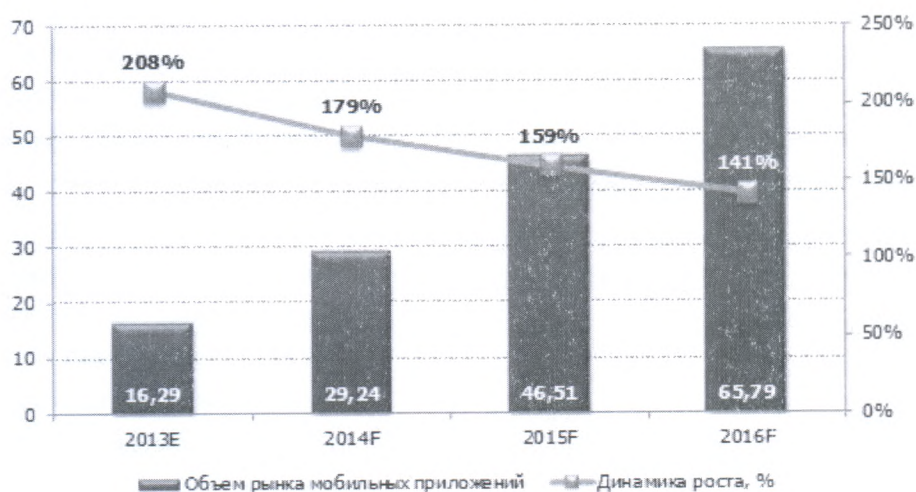
ҚОЛДАНЫЛҒАН ӘДЕБИЕТТЕР:

1. Қазақстан Республикасының Конституциясы. Алматы: Қазақстан. 2007ж. 21 мамыр. 55б.
2. "Халықты жұмыспен қамту туралы" Қазақстан Республикасының 2001 жылғы 23 қаңтардағы N 1493аңы
3. «Қазақстандағы кедейшіліктің себептері мен олардың шешу жолдары» Қазақстандаағы БҰҰДБ басылымдар сериясы. Алматы: 2014ж. 89б.
4. Қазақстанның электрондық мәліметтердің жиынтығы <http://student.zoomru.ru>

құрылғылардың мұншалықты пайдасы болмас еді.

J'son & Partners Consulting деректеріне сәйкес 2016 жылға қарай мобильді қосымшалар әлемдік нарықта \$65,79 млрдты құрайды [3]. Сарапшы J'son & Partners Consulting мобильді

қосымшалардың өсуін болжап отыр. Оның нарықта өсу қарқыны құндық бағасының төмендеуіне әсер етпейді. Сарапшы J'son & Partners Consulting болжамдары бойынша нарықтық тез өім болмаса да, өсу қарқыны тоқтамайды (1-сурет) [3]



1-сурет – Мобильді қосымшаларға деген сұраныстардың арту динамикасы, млрд USD, 2013-2016 жж.

Бұл мақалада кроссплатформалы бағдарламалық қамтама ретінде бірден артық платформалы аппаратта жұмыс істейтін бағдарлама екендігі айтылады. Тәжірибеде бағдарламалық қамтаманы жасау кезінде жаңа бағдарламаны дайындаушыларға жаңа талаптар қойылатындығы айтылады. Мұндай жағдайды біз ұялы телефондарды қамтамасыз ету кезінде байқай аламыз. Мобильді құрылғыларды стандарты компьютерлерден ерекшелеп көрсету үшін бағдарламалық қамтама құратын жаңа әдіс, модель, алгоритм және құрал-саймандар қажет етіледі. Мобильдік жүйеге жаңа кроссплатформалы бағдарламалық қамтама жасау үшін жаңа көзқарастар мен әдістер қажет болады [1].

Кроссплатформалы қосымша өңделмеген қосымшадан несімен ерекшеленеді?

Өңделмеген қосымша арнайы белгілі бір платформаға жазылған, мысалы, iOS, Android, Windows Phone, Symbian және т.б. Әрбір

қолданбаның өз ережелері мен өз тілдері бар, осылайша әрқайсысы бөлек әзірленеді. Мұндай қосымшаның үлкен мүмкіндіктері мен артықшылығы олардың жоғарғы жылдамдықпен жұмыс істеуінде. Ал олардың айтарлықтай кемшілігі – қымбат тұратындығында. Әрбір қосымшаны және API интерфейсі әртүрлі болғандықтан, бағдарламалау интерфейсін жеке жазу керек және әрқайсысын бөлек тестілеуден өткізу қажет, ал бағдарлама жасаушылар қымбат тұрады. iOS және Android-ты бірдей жақсы бағарламашыларды бір командаға жинау мүмкін емес, ол үшін әртүрлі бағытта iOS және Android үшін Objective-C және SDK-дан тәжірибелер қажет.

Кроссплатформалы қосымша тәуелсіз тілде жазылады, одан кейін әрбір құрылғыға жеке компиляцияланады. Кроссплатформалы қосымшаның жетістігі: бағдарлама коды бір рет жазылады да, ол жұмыс барысына қарай

JavaScript кітапханасы мен фреймворкты HTML коды ауыстырылып отырады. Ал кемшілігі - жылдамдығы өте баяу.

Мұндай бағдарламаны тіпті енді үйрене бастаған бағдарламашының өзі де жаза алады. Көбіне платформа таңдау жөнінде сұрақтар туындайды. Әрине бағдарламаның бірден iOS және Android, WP7 және MeeGo, десктоп және браузерде болғаны жақсы болар еді. Сонымен қатар бұлардың барлығын ақысыз құралдар арқылы тарату мүмкін болса тіптен жақсы.

Қазіргі заманғы жоғары деңгейлі бағдарламалау тілдерінің көпшілігін кроссплатформалы деп атауға болады. Мысалы, C, C++, Free Pascal, PureBasic компиляция деңгейіндегі кроссплатформалы тілдер, яғни осы тілдерге әртүрлі платформалы компиляторлар бар. Операциялық жүйеде осы тілдердің бірінде жазылған бағдарлама басқа бағдарламада іске қосылуы үшін кішігірім өзгертулерді ғана талап етуі мүмкін. Сапалы кодтаудың нәтижесінде оған әсер ететін жүйеден тәуелді бөлігі ғана өзгереді. Бірақ жоғарыда айтылғандар тек қана консолды бағдарламаларға байланысты. Әртүрлі операциялық жүйелерде графикалық қосымшаларды жасау үшін бағдарланған бағдарламалардың интерфейс функциялары (application programming interface, API) әртүрлі болады. Сонымен қатар пайдаланушының графикалық интерфейсі (graphic user interface, GUI) кем дегенде қолданбалы бағдарламалар үшін көптен бері де-факто болды.

Кроссплатформалы қосымшалар үшін стандартталған кітапханалардың орындалу уақыты маңызды. Атап айтқанда, стандартты кітапхана тілі болып Си есептелінді. Күрделі кроссплатформалы кітапханалар болып - Qt, GTK+, FLTK, STL, Boost, OpenGL, SDL, OpenAL, OpenCL.

PHP, ActionScript, Perl, Python, Tcl және Ruby есептелінді, олардың интерпретаторлары көптеген платформалар үшін қолайлы. Бұл кроссплатформалы кітапхана әмбебап болып келеді, яғни графикалық интерфейсін әзірлеу үшін ғана емес, бірақ бұл пайдаланушының (GUI toolkit) сыныптарын береді және жинау үшін түрлі мұқтажына жарайды (дыбыс желісін кескіндеумен жұмыс істеу, XML және т.б.). Осының есебінен олардың көмегімен түрлі бағыттағы бағдарламаларды әзірлеушілер кітапханаларды пайдаланбай қосымша құра алады.

Сондай-ақ, Java Virtual Machine және .NET кроссплатформалы, алайда олардың бастапқы мәтіні емес, аралық коды беріледі. Сондықтан Java және C#-та жазылған бағдарламаны әртүрлі операциялық жүйелерімен алдын ала перекомпиляциясыз іске қосуға болады.

Негізгі мәселелердің бірі - бағдарламалық жүйелерді әзірлеушілерге жасалатын өнімге қойылатын талаптардың әзірлеу барысында тез өзгеріп отыратындығы. Үнемі осы мәселені шешу үшін теориялық-әдістемелік деңгейде (жаңа әдіснаманы әзірлеу және бейімдеу бойынша қолданыстағы әдіснаманы құрудың жаңа міндеттеріне қарай) және тәжірибелік-құралдық деңгейде (қойылған мәселелені шешуге арналған құралдардың жаңа түрлерін әзірлеу мүмкіндік беретін, қолданылатын компоненттердің абстракциялық деңгейін көтеруге) жұмыс атқарылып отырады.

Жасалынған бағдарламалық жүйені бейімдеу үшін келесі талаптар қойылады:

1) пәндік аймақтың және өзгеріс алгоритміне әсер ететін жаңа функционалдық талаптардың пайда болуына байланысты ("Бизнес логика" жүйелеріндегі алгоритмдерде) формальды модельдер мен өнімділік өзгеріссіз қалуы мүмкін;

2) Бағдарламалық жүйенің өнімділігіне қойылатын талаптарға байланысты (өткізу мүмкіндігі, қауіпсіздік мәселелері, энергия тиімділігі);

3) платформаға қойылатын талаптар мен жасалу технологиясының өзгеруіне әкеліп соқтырады. Бұл өзгерістер жасалып отырған жаңа жағдайға бейімдеу мақсатында бағдарламалау тілдеріне операциялық жүйелер, бағдарламалық жүйенің жаңа аппаратты қамтамасыз етуіне толық немесе ішінара өзгерістерді талап етеді [1].

Ұзақ уақыт бойы қосымша бағдарламаларды дайындау сырттай компоненттердің көмегісіз жүзеге асырылды. Нәтижесінде кодтарды қайталап қолдану конфепциясы пайда болды. Статикалық және бөлікті кітапханаларды құру бағдарламасын пайдалану үшін құрылған, бірақ эволюциялық процесс тоқтап қалмай, бағдарламашылар қосымшалардың бағдарламалық фреймвортарымен жұмыс істеуге кірісті. Олардың арасында елеулі функциялық ұқсастықтар бар болса да, олардың арасында бірқатар айырмашылықтар да бар екендігін түсіну керек:

1. Кітапханалар функциялардың қайта пайдалануға мүмкіндігін берген кезде, Фреймворктарды пайдалану мүмкіндіктері өзгерістер тұжырымдамасын қайта пайдалануға мүмкіндік беретін болады. Мысалы, фреймворк абстрактілі сокетке кітапханамен жұмыс істеу үшін TCP және UDP-сокетты класстарды бере алады.

2. Фреймворк тікелей бағдарламалық жүйе мен бағдарламалық басқа компоненттік сигналдар көмегімен іс-қимыл жасауға мүмкіндік берген кезде, арнайы функциялардың көмегімен кітапхана оқиғалар мен сигналдарды өңдеуге мүмкіндік береді.

3. Кітапхана функциялары қосымша кодтар

арқылы шақырылады, ал фреймворк тығыз қосымшаға немесе бағдарлама кодынан толықтырылады.

Тәуелсіз платформалы қосымшаны жасау үшін барлық платформаларға сәйкес келетін бір кітапхана жасау жеткілікті болып есептелінеді, тек қана оның кодтарын қайта компиляция жасаса болғаны. Бұл әдіс Sun Microsystems компаниясы Java бағдарламалау тілідегі әлемдік виртуалды машиналардың тұжырымдамасын ұсынғанға дейін бағдарлама дайындаушылардың жұмысын жеңілдетті.

Java қосымшасы Java виртуалды машинаның көмегімен орындалады (Java Virtual Machine). Бір рет жазылған код кез келген Java виртуалды машинасының мүмкіндігі бар платформада ашыла алады. Бірақ бәрі ақылы болып келеді. Java тілі бойынша төлем өнімділікпен есептелінеді. C/C++ бағдарламасына қарағанда машиналық кодқа тәуелді Java бағдарламасының өнімділігі соншалықты жоғары емес. Осы қосымшалардың үлкен кемшілігі табылатын – ол жадыны көп талап етуі. Java тілі көшбасшылық позицияда, бірақ бүгінгі күні үлкен көлемді деректерді өңдейтін қосымшалар бар (мысалы, биотехнология саласында) олар аса қатты өнімділікті талап етеді.

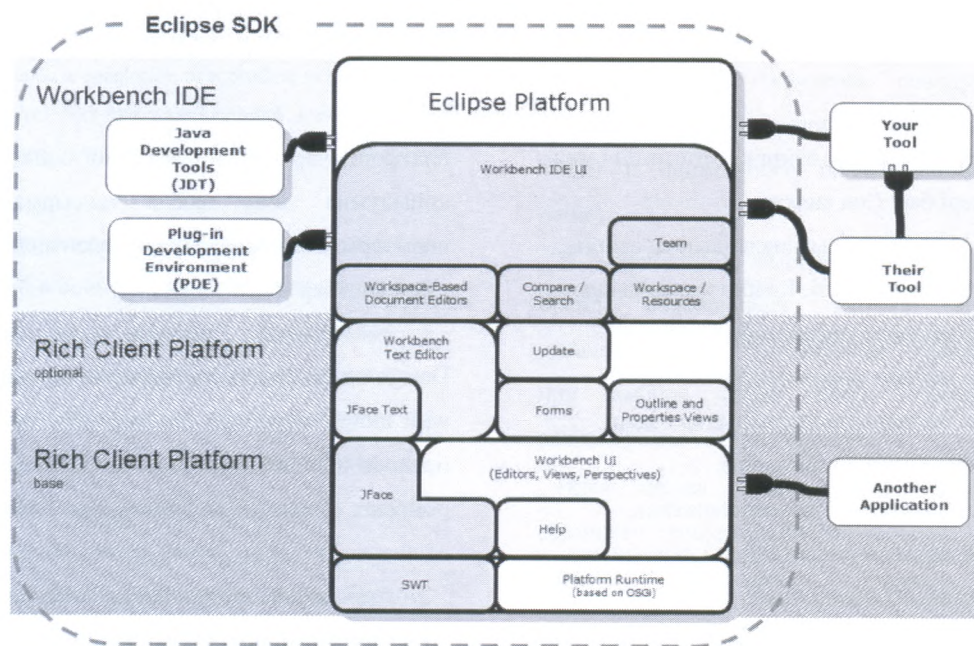
Интеграцияланған Java-қосымшалардың (IDE) ішінде Eclipse ортасы танымал. Eclipse Foundation-ды қолдайтын және дамып отыратын Java қосымшасында жазылған орта.

Eclipse тәуелсіз-платформасы, swt кітапханасын қоспағанда, таралған платформаларды өңдейтін өнім болып табылады. SWT кітапханасы өдеттегі Java үшін Swing кітапханасының орнына пайдаланылады. Жылдамдық пен қолданушы интерфейсінің табиғи сыртқы құрылысын сақтай отырып, өзінен төменгі платформалардың

(операциялық жүйелердің) көмегіне сүйенеді, бірақ кейде үйлесімділігі мен тұрақтылығы жағынан қайшылықтар туғызады. "Eclipse архитектурасы 2-суретте берілген.

Eclipse-ті www.eclipse.org сайтынан жүктеуге болады. Оны ашып, іске қосу керек. Eclipse ашылғаннан кейін жұмысқа дайын болады. Ешқандай қосымша қондырғылар талап етілмейді.

Осы платформаның модельділігі, кроссплатформалылығы, мульттілін қолдайтындығы, ақысыз болуы, фреймворк, кітапхана, плагиндердің көптігі оның жетістігі болып табылады. Осының бәрі коммерциялық деңгейдегі бағдарлама жасауға мүмкіндік береді.



2-сурет – Eclipse архитектурасы

Егер бағдарламашы машиналық кодты компиляциялау бағдарламасын әзірлеуі керек болса, онда ең үздік шешім болып C++ болып табылады. C++ арналған бағдарлама әзірлеу кезінде пайдалануға арналған бірқатар фреймворктар бар, солардың бірі - Qt.

Қосымшаны дайындау үшін Qt кроссплатформалы фреймворк болып табылады. Фреймворк C++ тілінде API-ді береді. Бағдарлама әзірлеушілер кроссплатформалы қосымшаларды дайындау үшін Qt жиі пайдаланады.

Qt көмегімен графикалық интерфейстермен, желімен жұмыс істейтін, қосымшалар деректер қорымен қорлармен және мультимедиялық

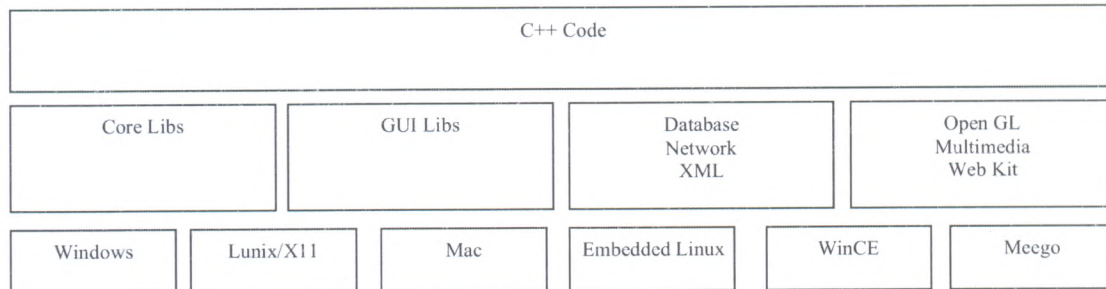
қосымшамен, 3D-графика мен XML – құрылымымен жұмыс істейтін алатын және сурет салып, желілік ресурстарға шыға алатын қосымша жасауға болады. Себебі, Qt - Windows, Linux, Mac OS, Meego-ларда және Linux-жүйесі мен Symbian қосымшаларында жұмыс істей алады. 3-суретте Qt архитектурасын бейнелейтін қарапайым сызба келтірілген.

Жоғарғы дейгейде C++ тіліндегі бағдарламаның коды орналасқан. Одан кейінгі деңгейде Qt класы орналасқан, олар графикалық интерфейстерді, WebKit-тің өзара іс-қимылын, деректер қорымен

жұмысты қамтамасыз етеді, ал одан да төмен әртүрлі операциялық жүйелердің қолдау деңгейі

орналасқан. Бұрын Qt Java бағдарламалау тілін қолдаған, ол Jambie нұсқасы деп аталған. Qt-дің даму қарқынына байланысты, бұрынғы C++ пен Java-ны қатар қолдау қиындықтар туғыза

бастағаннан кейін, тек қана C++ бағдарламасына ғана қолдау көрсетуге шешім қабылданды [2].



3-сурет - Qt архитектурасы

Әрбір технологияның өзіндік жетістіктері мен кемшіліктері бар. Сол сияқты кроссплатформалы қосымшалардың да өзіндік ерекшеліктері жеткілікті. Кейбір жағдайларда оларды қолдану өте ыңғайлы болып келсе, кейде олар мүлдем жарамсыз боп жатады. Сондықтан бағдарлама жасаушылар өздері қолайлы қосымшаны таба алады.

Жаңа бағдарламалық және аппараттық платформалардың бүгінгі таңдағы дамуын екі анық үрдістер айқындап отыр:

- бағдарламалық және аппараттық платформалар барған сайын бір-бірімен үйлесімділік табуда, сондықтан олардың арасындағы шекараны оңай жеңуге болады;

- бір есепті шешудің әртүрлі тәсілдерін ұсынатын жаңа технологиялар кейінгі кезде көбейіп келеді.

Осылайша, бір бағдарламалық продукт дайындау үшін бағдарлама құрушы қосымша

платформалар мен технологияны есептің қойылуына және нақты тапсырмалар мен шарттарына байланысты таңдауы тиіс.

Әдебиеттер

1. Lebedev A.S., Bolshakov O. S., Petrov A.V. Design of the distributed system of relaying of data with mobile clients on the basis of cross-platform methods of development of the software//Modern problems of science and education. – No. 1, 2013 of.
2. Applications programming with use of Qt. URL:<http://rus-linux.net/MyLDP/algol/Qt/developing-applications-qt-part-1.html>.
3. The market of mobile applications in Russia and in the world. URL:http://www.json.ru/poleznye_materialy/free_market_watches/analytics/rynok_mobilnyh_prilozhenij_v_rossii_i_mire/.