

O'ZBEKISTON RESPUBLIKASI AXBOROT TEXNOLOGIYALARI
VA KOMMUNIKATSIYALARNI RIVOJLANTIRISH VAZIRLIGI

MUHAMMAD AL – XORAZMIY NOMIDAGI TOSHKENT AXBOROT
TEXNOLOGIYALARI UNIVERSITETI SAMARQAND FILIALI

“Kompyuter injiniringi” fakulteti
“Kompyuter tizimlari” kafedrası

5330500-“Kompyuter injiniringi”(Kompyuter injiniringi, AT servisi) yo'nalishi

“Himoyaga ruxsat etildi”

Kafedra mudiri

_____ Bekmurodov Q.A.

“ _____ ” “ _____ ” 2020 yil

Python dasturlash tilini o'qitish uchun mobil trenajer-ilovasini ishlab chiqish.

BITIRUV MALAKAVIY ISHI

Bitiruvchi	_____	402- guruh talabasi Mamatqulov M.
Ilmiy rahbar	_____	Sattarov M.
Taqrizchi	_____	_____.
HFX bo'yicha maslahatchi	_____	dots.Asrarov Sh.A.

SAMARQAND 2020

MUNDARIJA

KIRISH	3
I BOB. MOBIL QURILMALARNING DASTURIY TA'MINOTINI VA ARXITEKTURASI TAHLILI.	6
1.1. Masalaning qo'yilishi	6
1.2. Mobil ilovani ishlab chiqish uchun OTni tanlash.....	6
1.3. Android platformasi arxitekturasini.....	10
II BOB. PYTHON DASTURLASH TILINI O'QITISH UCHUN MOBIL TRENAJER-ILOVASINI LOYIHALASH VA ILOVASINI.....	13
2.1. Mobil o'quv uslubiy majmuani loyihalash.....	13
2.2. Android Studio dasturini o'rnatish va undan foydalanish	16
2.3. PhoneGap texnologiyasi	22
III BOB. DASTURIY VOSITA TAFSIFI VA UNDAN FOYDALANISH UCHUN TAVSIYALAR	27
3.1. Python dasturlash tilini o'qitish uchun mobil trenajer-ilovasining tafsifi...	27
3.2. Dasturdan foydalanish bo'yicha tavsiyalar	27
3.3. Kompyuter xonalarida ishlashda hayot havfsizligi qoidalari	33
XULOSA	38
FOYDALANILGAN ADABIYOTLAR RO'YXATI	39
ILOVA	41

KIRISH

O'zbekiston Respublikasi Prezidentining 2018 yil 5 iyunda "Oliy ta'lim muassasalarida ta'lim sifatini oshirish va ularning mamlakatda amalga oshirilayotgan keng qamrovli islohotlarda faol ishtirokini ta'minlash bo'yicha qo'shimcha chora-tadbirlar to'g'risida" PQ-3775-son qarori qabul qilindi.

Qarorda O'zbekiston Respublikasi Oliy va o'rta maxsus ta'lim vazirligi manfaatdor vazirlik va idoralar bilan birgalikda oliy ta'lim muassasalari talabalarini ilmiy faoliyatga yo'naltirishga va ularda innovasion fikrlashni shakllantirishga qaratilgan chora-tadbirlar dasturi Vazirlar Mahkamasiga kiritilgan bo'lib, unda quyidagilar:

o'quv mashg'ulotlarini talabalarni innovasion fikrlashga yo'naltiradigan o'qitish texnologiyalari va interfaol uslublarni joriy etish asosida tashkil etish, asosiy e'tiborni talabalarning mustaqil ta'lim olishi bilan bog'liq mexanizmlarni amalga oshirishga qaratish;

oliy ta'lim tizimida boshqaruv jarayonlarini axborot-kommunikasiya texnologiyalari asosida optimallashtirish, mavjud axborot tizimlarini integrasiyalash, tahliliy ma'lumotlar tayyorlashning samarali tizimlarini joriy etish nazarda tutilgan.

Bu esa zamon talabiga mos bo'lgan elektron, mobil – yangi avlod texnologiyalari talablariga javob beruvchi o'quv uslubiy qo'llanmalarni ishlab chiqish va dars jarayonida qo'llash kabi vazifalarni keltirib chiqaradi.

Bitiruv dolzarbligi: So'nggi bir necha yillardan beri, mobil devayslar sanoati shunchalik ulkanlashib kettiki, internetdan umumiy foydalanish va faollik bo'yicha hattoki kompyuterlar bozorini ham egallab bormoqda. Bu albatta mobil vositalardagi ustunlik, hamda ma'lumotlar tarmog'i bo'yicha muntazam aloqa tezligi sababidandir. Zamonaviy planshet yoki smartfonda ishlaydigan dasturlarni Android platformasida yaratilishini hammamiz bilamiz. Unda yaratilgan milliy va mahalliy dasturlar esa kam. Yurtimizda ham mobil qurilmalar uchun ilovalar yaratuvchi dasturchilarni tayyorlash borasida

ko‘plab ishlar amalga oshirilmoqda. Bunga misol sifatida yurtimiz oliy o‘quv yurtlarida mobil qurilmalarda ilovalar yaratish fanining o‘qilayotgani, o‘tkazilayotgan tanlovlarni olishimiz mumkin.

Mazkur bitiruv malakaviy ishida — Python dasturlash tilini o‘qitish uchun mobil trenajer-ilovasini ishlab chiqish bo‘yicha izlanishlar olib borildi va amaliy ish sifatida — Python dasturlash fanidan mobil elektron qo‘llanmasi yaratildi.

Bitiruv malakaviy ishi natijasi sifatida yaratilgan dastur O‘zbekiston Respublikasi Prezidentining 2018 yil 5 iyunda “Oliy ta’lim muassasalarida ta’lim sifatini oshirish va ularning mamlakatda amalga oshirilayotgan keng qamrovli islohotlarda faol ishtirokini ta’minlash bo‘yicha qo‘shimcha chora-tadbirlar to‘g‘risida” PQ-3775-son qarorigi belgilan ba’zi bazifalarni bajarishda ya’ni talabalarning mustaqil ta’lim olishi bilan bog‘liq mexanizmlarni amalga oshirishga xizmat ko‘rsatishligi bilan muximdir.

Bitiruv malakaviy ishining maqsadi: Mazkur bitiruv malakaviy ishining maqsadi “Python dasturlash tili” fanini mazmun mohiyatini yorutuvchi mobil ilova yaratish, ilovani yaratishga mo‘ljallangan zarur dasturiy vosutalar va qurilmalar platformalarini tahlil qilishdan, shuningdek mobil qurilmalar foydalanuvchilariga ya’ni talabalar, mustaqil tadqiqotchilar hamda professor o‘qituvchilar uchun Python dasturlash tilini o‘qitish uchun mobil trenajer-ilovasidan foydalanish bo‘yicha tavsiyalar ishlab chiqishdan iborat.

Mavzuning amaliy ahamiyati. Bitiruv malakaviy ishi natijasi sifatida yaratilgan mobil ilovadan talabalar, mustaqil tadqiqotchilar hamda professor o‘qituvchilar Python dasturlash tilini o‘qitishda, shuningdek ixtiyoriy foydalanuvchilarga kerakli ma’lumiotlarni taqdim etishda foydalanish mumkin.

Bitiruv ishining qismlari bo‘yicha qisqacha tafsilotlar: Ushbu bitiruv malakaviy ishi kirish qismi, uchta bob, xulosa, adabiyotlar va ilovadan iborat.

Birinchi bobda Mobil qurilmalarning dasturiy ta’minotini va arxitekturasini tahlili, Android va iOS operatsion tizimlarining qiyosiy tahlili, Android platformasi arxitekturasini haqidagi ma’lumotlar keltirilgan.

Ikkinchi bobda mobil o'quv uslubiy majmuani loyihalash, Android App Book maker dasturini o'rnatish va undan foydalanish, PhoneGap texnologiyasini qo'llash tavsiflari keltirilgan.

Uchinchi bobda mobil ikovaning tafsifi va undan foydalanish uchun ko'rsatmalar keltirilgan.

Xulosa qismida bitiruv ishining asosiy natijalari va uning amaliy tadbiqlari bayon qilingan

Ilova qismida dasturning oynalari ko'rinishlari keltirilgan.

I BOB. MOBIL QURILMALARNING DASTURIY TA'MINOTINI VA ARXITEKTURASI TAHLILI.

1.1.Masalaning qo'yilishi

Bugungi kunda ta'lim tizimi ning barcha bosqichlarida axborot texnologiyalari ni qo'llash, axborot texnologiyalaridan foydalangan xolda dars mashg'ulotlarini olib borish, dasturiy vositalardan foydalangan xolda o'rgatuvchi dasturlarni yaratish va ulardan foydalanish dolzarb vazifalardan biridir. Umumta'lim maktablari va akademik litseylarda, oliy ta'lim muassasalarida fanlardan bilimlarni o'zlashtirishda zamonaviy texnologiyalardan foydalanib dars o'tish ommalashib bormoqda.

Ushbu bitiruv malakaviy ishida qo'yilgan masala oliy ta'lim muassasalari talabalari uchun "Python dasturlash tilini" fanini o'rgatishda, nazariy va amaliy bilimlarni berishda qo'llanma vazifasini o'tovchi trenajor-mobil ilova yaratishdan iborat. Ushbu bitiruv malakaviy ishi tayyorlash jarayonida quyidagi vazifalar amalga oshirilishi lozim:

- "Python dasturlash tili" bo'yicha mavjud mobil trenajer-ilovalarni tahlil qilish.
- Fanni o'qitish usullarini tahlil qilish.
- Mobil ilovani loyihalash va arxitekturasini ishlab chiqish.
- Mobil ilovani ishlab chiqish uchun algoritmni va dasturlash vositalarni tanlash.
- Foydalanuvchi uchun yo'riqnomani shakllantirish.

Ishni avvalo Android va iOS operatsion tizimlarining xususiyatlarini o'rganish va qiyosiy tahlil qilishdan boshlaymiz.

1.2. Mobil ilovani ishlab chiqish uchun OTni tanlash

So'nggi bir necha yillardan beri, mobil qurilmalar sanoati shunchalik ulkanlashib ketdiki, internetdan umumiy foydalanish va faollik bo'yicha hattoki kompyuterlar bozorini ham egallab bormoqda. Bu albatta mobil vositalardagi

ustunlik, hamda ma'lumotlar tarmog'i bo'yicha muntazam aloqa tezligi sababidandir.

Mobil qurilmalar – planshet, smartfonlar uchun maxsus ishlab chiqilgan operatsion tizimlar ham mavjud. Xuddi kompyuterlar o'tmishidagi singari, mobil operatsion tizimlari (O.T.) ishlab chiqaruvchilari orasida ham yakka liderlik uchun kurash boshlanib ketdi. Eng talabgorlar ro'yxatidan Apple iOS, Windows Phone 7 va Google Android misol bo'ladi. Ular turli sanoat ishtirokchilari tomonidan kiritilgan. Har bir O.T ba'zi foydalanuvchilarga mos keluvchi va ba'zilarini yuzini burishtiruvchi o'z xususiyatlariga ega [4].

Android tizimining eng o'ziga xos xususiyatlaridan biri – unda o'zlashtirish (customization) imkoniyatlari miqdoridir. Bu qulaylik telefonning butun telefonning barcha qismi, o'rnatilgan fon, vidjetlar va ikonkalarni o'z ichiga oladi. Android Google Play yoki nashr qiluvchining o'zidan bevosita yuklab olinadigan, bir milliondan oshiq turli ilovalarga ega. **Android telefonlar o'zgartirishga** sezilarli darajada ochiq hisoblanadi. Misol uchun, Google Play ilova do'konidagi eng ommabop dasturlardan SwiftKey bo'lib, u Android smartfoni klaviaturasini o'zgartirib, Google va Facebook akkauntlari bilan hamohang holda keyingi yozadigan so'zingizni tahmin qiladi. Android ekotizimining imkoniyatlari deyarli cheksizdir.

iOS mobil O.T. Apple kompaniyasining iPhone smartfonida 2007 yilda ishga tushirilgan bo'lib, mavjud eng birinchi mobil O.Tlaridan biridir. Ushbu tizimdan foydalanish ortidagi asosiy g'oya -boshqa barcha Apple mobil mahsulotlaridan foydalana olish imkoni bo'ladi. Ma'lumki Apple mavjud eng zo'r telefonlarni ishlab chiqadi, shu bilan o'z operatsion tizimini ahamiyatini oshiradi. Undan tashqari iOS uchun mobil ilovalar soni ulkandir, hayolingizga kelishi mumkin bo'lgan barcha dastur va hizmatlar muhayyo.

Windows Phone. Kompyuterlar orasida Windows eng hurmatga sazovor O.T. bo'lishiga qaramay, Window mobil muqobillari ham yaratildi. Ijro bobida Windows Phone o'z raqiblari iOS va Androidga yaqin darajadagi ko'rsatkichga erishdi; biroq uning ilovalari miqdori sezilarli darajada kamchilikni tashkil etadi.

Shuning uchun Windows O.T.dan ko'ra, Android va iOS yaxshi tanlov bo'lib qolmoqda mumkin.

iOS va Android O.T.larning ijobiy/salbiy jihatlari

iPhone'ning Android telefonlaridan eng katta ustunligi uning **ulkan app ekotizimidir**. Divayslar ichida birinchi haqiqiy App Store egasi iPhone bo'lib, natijada developerlar birinchi aynan unga moslab dastur tuzishadi. Misol uchun Instagram iPhone'da anchadan beri mavjud bo'lib kelgan bo'lsa, Facebook o'z dasturini Androidda yangilash(upgrade)dan avval iPhone'da qayta dizayn yaratdi. Apple'ning rasmiy bayonotida, so'nggi bor App Store'da ilovalar soni 1500000 dan oshgan[20].

Bu ham yetmagandek iPhone Twitter va Facebook bilan chuqur integrasiya (birlashuv)ga ega, oqibatda platformaning o'zidan turib status yangiliklari kiritish mumkin. Appni chuqur kastomizatsiya(moslashtirish) qilish shart emas.

Bundan tashqari Apple o'ziga xos iPhone'ga ixtisoslashgan dasturlar, masalan **Passbook** kabilarga ega. Hozirda cheklangan xizmatga qaramay ushbu dastur xar hil qo'shimcha bilet, savdo chegirmalari (gift certificate)ni tartiblash imkonini beradi.

Eng muhimi esa, foydalanuvchi tajribasi (user experience) qulflangan. Bu degani – telefoningizning ichki ishlash va ko'rinishini ko'nglingiz tusagancha o'zgartirolmaysiz degani, masalan klaviatura. Lekin texnologiyaga uncha "mukkasidan ketmaganlar"ga yoki kastomizasiya shaydosi bo'lmaganlarga bu ayni muddao bo'lishi ham mumkin.

iOS afzalliklari

- **Uulkan miqdordagi dasturlar:** planshetlar uchun alohida afzalligi, chunki smartfonlardagi ommabop dasturlar ko'pincha ikkala platformada ham mavjud bo'ladi;

- **Facebook hamda Twitter bilan chuqurroq birlashuv:** yangiliklarni kiritish va ijtimoiy sahifalarda bo'lishish iOSda Androiddan ko'ra osonroq;

- **Faqatgina iOSda uchrovchi dasturlar** masalan Passbook, FaceTime, Square kabi mobil to'lovlar dasturi (iOS 3GS, 4, 4S, 5 va yuqori versiyalarda mavjud, Androidlarda esa chegaralangan miqdordagi telefonlarda);

- **Interfeys qulflangan:** Uy ekrani uchun chegaralangan moslashtirish variantlari; dasturlarni faqatgina qatorlashgan ko'rinishi. Faqatgina App Store dan dastur yuklay olish imkoniyati;

- **Dastur yangilanishi:** Apple barcha "jismonan ko'tara oladigan" divayslari uchun yangilangan ilova versiyasini chiqaradi (software upgrades). Bu degani divays 2 yo 3 yilcha joriy "soft"da ishlay oladi;

- **Nisbatan yaxshiroq shaxsiyat muxofazasi:** iOS foydalanuvchining kontakt va joylashuvini talab etuvchi dasturlar ustida yaxshiroq nazorat o'rnatgan.

Android afzalliklari

- **Ulkan miqdordagi divayslar:** Ko'plab Android gadjetlarini istalgan afzalliklar, ekran kattaligi, xususiyatlari va narxlari bilan topish mumkin;

- **Yuqori moslashuvchan UX:** Uy ekrani nafaqat dastur tugmasi bilan balki foydalanuvchini doimo ogohlantirib va habardor etib turuvchi – turli vidjetlar bilan ham boshqariladi. Yana misol qilib SwiftKey'ni olamiz, u Android smartingiz klaviaturasini o'zgartirib beradi. Google o'z do'konida joylashtirish uchun dasturlar ustidan Applega qaraganda yengilroq talablar qo'ygan. Undan tashqari, istasangiz Google Playdan boshqa joydan ilova o'rnatish mumkin [6].

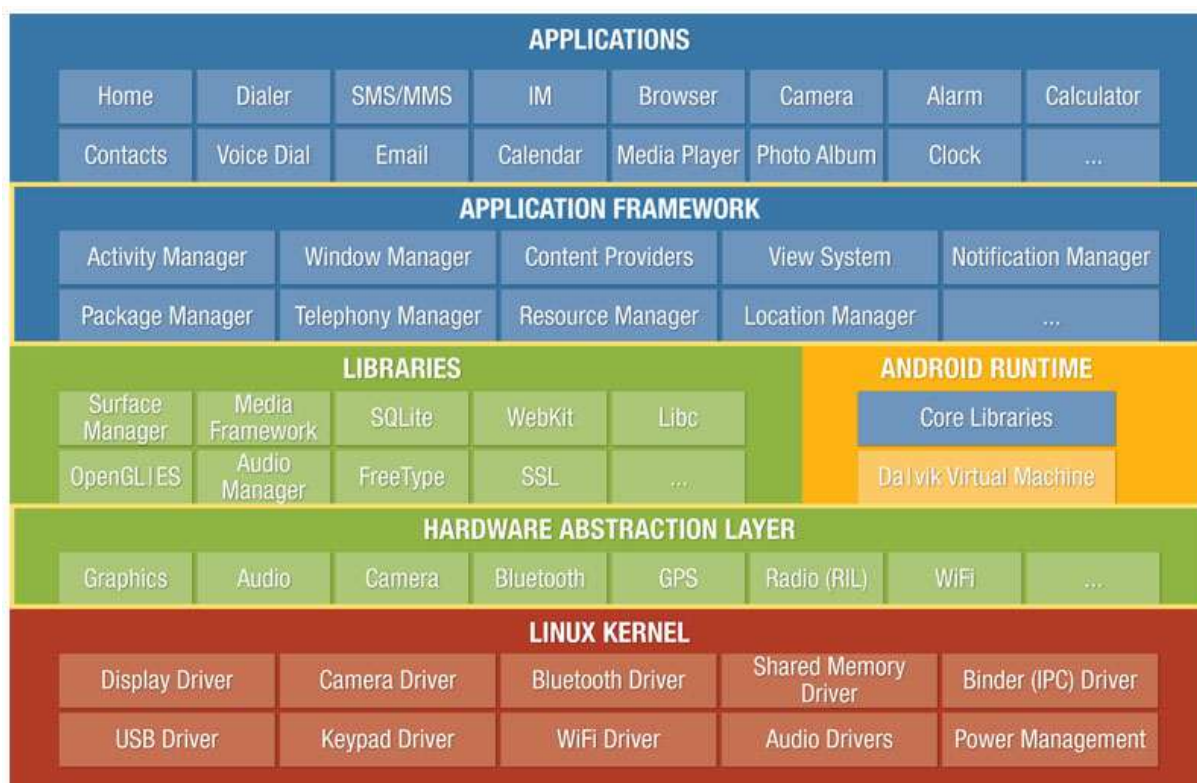
Qolgan ikkita ulkan platformalar – **Windows Phone** va **BlackBerry** – o'z alohida afzalliklariga ega bo'lsada, Android va iPhonedagi ilovalar ko'lamiga yetib olishlari vaqt talab qiladi. Kompyuterlar orasida Windows eng hurmatga

sazovor operatsion tizim bo‘lishiga qaramay, ijro bobida Windows Phone o‘z raqiblari iOS va Androidga yaqin darajadagi ko‘rsatkichga erishdi, biroq uning ilovalari miqdori sezilarli darajada kamchilikni tashkil etadi.

1.3. Android platformasi arxitekturası

Android platformasida dasturchilash uchun instrumentlar to‘olami va API kutubxonasi – Android SDK talab etiladi.

Quyida keltirilgan 1.3.1-rasmda Android AT arxitekturası ko‘rsatilgan.



1.3.1-rasm. Android OT arxitekturası

Ilovalar sathi (Applications). Android tarkibiga kiruvchi asosiy ilovalar: elektron pochta va SMS klienti, kalendar, turli xil kartalar, brauzer, kontaktlarni boshqaruvchi dasturlar va ko‘plab boshqa ilovlar. Android platformasida ishlovchi deyarli barcha dasturlar Java tilida yoziladi.

Freymvork sathi (Application Framework). Android dastur yadroida foydalanilayotgan API dan to‘liq quvvatda foydalanishga imkon beradi. Arxitektura shunday qurilganki, istalgan ilova imkoniyatlaridan boshqa ilova amalga oshirilganda ham foydalanish imkonini beradi. Shunday qilib, arxitektura OT va ilovalardan ko‘p qirrali foydalanish prinsipini amalga oshiradi.

Tizim va xizmatlar to'plami barcha ilovalarning asosi hisoblanadi:

1. Ko'rinish tizimi (View System) – bu ilovaning tashqi ko'rinishini qurish uchun xizmat qiluvchi boy ko'rinishlar to'plamidir. Uning tarkibiga ro'yxatlar, jadvallar, kiritish maydonlari, tugmalar va shu kabilar kiradi.
2. Kontent provayderlari (Content Providers) – bu ilovaga boshqa ilova ma'lumotlariga ruxsat olishni ta'minlovchi, shuningdek o'zining ma'lumotlariga ruxsatni taqdim etuvchi xizmatlar.
3. Resurslar menejeri (Resource Manager) satrli, grafikli va boshqa turdagi resurslarga ruxsat berish uchun mo'ljallangan.
4. Xabarlar menejeri (Notification Manager) - istalgan ilovaga holatlar satrida foydalanuvchi xabarlarini ko'rsatish imkonini beradi.
5. Harakatlar menejeri (Activity Manager) – ilovaning hayotiy siklini boshqaradi.

Kutubxona sathi (Libraries). Android platformasi OTning turli xil komponentalarida ishlatiladigan C/C++ kutubxonalar to'plamini o'z ichiga oladi. Dasturchilar uchun bu kutubxonalar vazifalariga ruxsat Application Framework dan foydalanish orqali amalga oshiriladi. Quyida ulardan bir nechta taqdim etilgan:

1. System C Library – Linuxga asoslangan o'rnatilgan qurilmalar uchun C (libs) standart kutubxonalar tizimini amalga oshirish.
2. Media Libraries – PacketVideo's OpenCORE ga asoslangan MPEG4, H.264, MP3, AAC, AMR, JPG, PNG va boshqa formatlardagi audio va videolarni o'qish va yozish uchun mo'ljallangan.
3. LibWebCore – o'rnatilgan android brauzerining bor quvvatini amalga oshiruvchi kutubxona.
4. SGL – 2D- tasvirlar bilan ishlovchi kutubxona.
5. 3D libraries – OpenGL ES 1.0 API ga asoslangan 3D-tasvirlar bilan ishlovchi kutubxona.
6. FreeType – yozuv shriftlari bilan ishlash uchun mo'ljallangan kutubxona.

7. SQLite – relatsion MBlari bilan ishlovchi kutubxona.

Amalga oshirish muhiti sathi (Android Runtime.). Android tarkibiga Java tilining asosiy kutubxonalarining ko'p funksiyasini ta'minlovchi yadro kutubxonalari to'plami kiradi.

Linux yadrosi sathi (Linux Kernel). Android Linux OTning 2.6 versiyasiga asoslangan. Yadro apparat va dasturiy ta'minot o'rtasida ajralmaslikning bir qatlami sifatida xizmat qiladi.

II BOB. PYTHON DASTURLASH TILINI O'QITISH UCHUN MOBIL TRENAJER-ILOVASINI LOYIHALASH VA ILOVASINI YARATISH VOSITALARI.

2.1. Mobil o'quv uslubiy majmuani loyihalash

Mobil ilovani loyihalashdan oldin loyihalash loyihalash tillaridan biri bo'lgan UML tili haqida biroz suhbatlashsak chunki mobil ilovani shu til yordamida loyihaladim.

UML - Unifikatsiyalangan modellash tili. UML (Uniform Modeling Language) bu - vizual modellashtirish tili bo'lib, tizimlash arxitektorlariga tizim qay darajada standart va tushunish uchun oson shaklga ega ekani haqidagi o'z tasavvurlarini namoyon etishga imkon beradi. UML loyihaviy yechimlar va ishlab chiquvchilarning o'zaro aloqalarini qo'shgan holda birgalikda foydalanishning samarali mexanizmini taqdim etadi.

UML ning unumdorligi. Analitik (tahlilchi) muayyan muammo tavsifi berilgan hujjatlarni ishlab chiqadi hamda ularni ishlab chiquvchilar - dasturchilarga beradi.

Dasturchilar talab qilinayotgan masala yechimi uchun dasturiy ta'minot tayyorlaydi hamda uning apparat vositalarida yoyilishini kafolatlaydi.

Tasavvurdagi tizimni ifodalab berish tizim ustida ishlash jarayonining g'oyat muhim bosqichidir. Avval tahlil "barmoqda sanab", ya'ni tavakkalchilik asosida o'tkazilar edi. Hozirgi paytda ishlab chiqishning hal qiluvchi bosqichi bu puxta ishlangan rejadir. O'z navbatida, reja ham mijoz talablari puxta tahlil qilib chiqilgach, tuzilishi kerak.

Loyihalash jarayonining muhim jihatida bu jarayonni to'g'ri tashkil qilishdir.

Bunda tizimni ishlab chiqishda ishtirok etayotgan analitiklar, mijozlar, dasturchilar va boshqa mutaxassislar bir-birini tushunib, umumiy fikrga kela olishlari talab qilinadi.

Zamonaviy tizimlarni ishlab chiqish jarayonining yana bir o'ziga xos jihat shundaki, ishlarni bajarish uchun vaqt yetishmaydi. Agar tarmoq tizimlarning chegaraviy topshirish muddatlari bir-biriga yaqin bo'lsa, ishlab chiqish jarayonining uzluksizligini ta'minlash hayotiy muhim zaruratga aylanadi.

Hozirgi zamonning yana bir o'ziga xos jhati — korporatsiyalarning birlashishi — ham ishlab chiqish jarayoniga o'z talablarini qo'yadi.

UML asoslari. Boshqa har qanday til kabi, UML ham lug'at va qoidalar iborat bo'lib, ular UML tiliga kiritilayotgan so'zlarni kombinatsiyalab, mazmunli konstruksiyalar olish imkonini beradi.

Modellashtirish tilida lug'at va qoidalar tizimni konseptual va jismoniy jihatdan taqdim etishga yo'naltirilgan. UML ga o'xshash modellashtirish tili dasturiy ta'minotning «chizmalari»ni tuzish uchun standart vosita bo'lib xizmat qiladi. UML lug'ati asosiy konstruksiyalarning uchta turini o'z ichiga oladi:

- mohiyatlar— modelning asosiy elementlari bo'lgan abstraksiyalar;
- munosabatlar— mohiyatlar o'rtasidagi aloqalar;
- ko'plab mohiyatlar va munosabatlarning manfaatlarini ko'zlovchi, guruhlashtiruvchi diagrammalar.

Diagrammalar tizimning turli tasvirlarini aks ettirish uchun qo'llanadi. Turli tasvirlarning ushbu to'plami model deb ataladi. UML tizimi modelni binoning badiiy bezatilgan modeli bilan qiyoslash mumkin. Shuni ta'kidlab o'tish muhimki, UML modeli tizim nima qilishi lozimligini tavsiflaydi. Shu vaqtning o'zida bu qanday joriy qilinishi ma'lum qilinmaydi.

UML modellashtirish tili loyiha yaratishda eng ko'p qo'llanayotgan dasturlardan biri bu StartUML dasturidir. Bu dastur yuqori darajadagi

moellashtirish uchun mo'ljallangan bo'lib, unda 14 xil model va loyihalar yaratish imkoniyati mavjud.

Dastur modellashtirish texnologiyalarining barcha standartlariga to'la javob beradi. Unda quyidagicha turdagi modellar yaratish mumkin: Class Diagram, Object Diagram, Package Diagram, Composite Structure Diagram, Deployment Diagram, Use Case Diagram, Sequence Diagram, Communication Diagram, Statechart Diagram, Activity Diagram, Profile Diagram va ER Diagram. Mobil ilovani Star UML dastur yordamida loyihalash juda oson bo'lganligi sababli bu ilovani loyihasini shu dasturda ya'ni Star UML dasturi yordamida loyihalash kerak.

Use Case Diagram diagrammasi orqali tuzilgan diagramma quyidagicha: Bu diagrammada foydalanuvchi (user) dasturga kirishi bilan asosiy oyna ochilishi kerak.

Tilni esa shu qurilmani tili bilan bir xil qilib qo'yish kerak chunki foydalanuvchi shu tilni yaxshi biladi.

Har bir dasturda sozlashlar bo'lgani kabi bu ilovada ham sozlashlar bo'lishi kerak. Ho'sh bu sozlashlarda nimalar bo'lishi kerak. Foydalanuvchiga ilova ogohlantirish ya'ni dasturda chiqib ketilgandan so'ng ma'lum bir vaqtda budilnik sifatida yuqorida уведомление bo'lishi kerak. Tilni o'zgartirish imkoniyati va interfeys fonini o'zgartirish imkoniyatlari bo'lishi kerak.

Bog'lanishda dastur tuzuvchisi bilan bog'lanish imkoniyat bo'lishi. Bu bog'lanishda dastur kamchiliklari va takliflarni yo'llashlari kerak.

Sequence Diagram diagrammasi bu ketma ketlik diagrammasi hisoblanadi. Bunda asosan dasturni ishlatishdagi ketma ketlik loyihalanadi. Sequence Diagram diagrammasi orqali tuzilgan diagramma quyidagicha: Bu diagrammada foydalanuvchi (user) dasturga kirishi bilan asosiy oyna ochilishi kerak. Tilni esa shu qurilmani tili bilan bir xil qilib qo'yish kerak chunki foydalanuvchi shu tilni yaxshi biladi. Asosiy oynada o'yin turi, online bellashuv, o'rganish, sozlash bo'g'lanish bo'limlari boladi.

2.2. Android Studio dasturini o'rnatish va undan foydalanish

Ko'p yillar davomida, Eclipse IDE Android dasturlash uchun ishlatilib kelindi. Lekin bugun Android Studio o'zining imkoniyatlari bilan yetakchilikni qo'lga oldi.

Android Studio Google ning Android ilovalarini qurish uchun ishlab chiqqan rasmiy IDE hisoblanadi. Bu IDE IntelliJ IDEA asosida qurilgan, bu IDE dasturchilarga kerakli bo'lgan ko'plab foydali imkoniyatlarni taqdim etadi.

Google Android Studioni Windows, Mac OS X va albatta Linux platformalari uchun ishlab chiqqan. Android Studioni uning rasmiy sahifasidan ko'chirib olish mumkin [21], u yerda Android Studioda ishlash uchun kerak bo'ladigan SDK ni ham yuklab olish mumkin bo'ladi. Android Studioni ko'chirishdan oldin uning ishlashi uchun kompyuterdan talab etiladigan texnik xususiyatlarini qarab chiqamiz:

Windows OT uchun talablar

- Microsoft Windows 7/8/10 (32-bit or 64-bit)
- 3 GB RAM minimum, 8 GB RAM (*tavsiya qilingan*) (yana 1 GB Android Emulator uchun)
- 2 GB hotiradan joy minimum, 4 GB (*tavsiya qilingan*) (500 MB IDE yana 1.5 GB Android SDK uchun va emulator system image uchun)
- 1280 x 800 minimum ekran o'lchami

Mac OS uchun talablar

- Mac OS X 10.10 (Yosemite) yoki yuqori, 10.13 gacha (High Sierra)
- 3 GB RAM minimum, 8 GB RAM (*tavsiya qilingan*) (yana 1 GB Android Emulator uchun)
- 2 GB hotiradan joy minimum, 4 GB (*tavsiya qilingan*) (500 MB IDE yana 1.5 GB Android SDK uchun va emulator system image uchun)
- 1280 x 800 minimum ekran o'lchami

Linux OS uchun talablar

- GNOME or KDE desktop. Ubuntu 14.04 LTS, Trusty Tahr
- GNU C Library (glibc) 2.19 yoki yuqori

- 3 GB RAM minimum, 8 GB RAM (*tavsiya qilingan*) (plus 1 GB Android Emulator uchun)
- 2 GB hotiradan joy minimum, 4 GB (*tavsiya qilingan*) (500 MB IDE plus 1.5 GB Android SDK uchun va emulator system image uchun)
- 1280 x 800 minimum ekran o'lchami

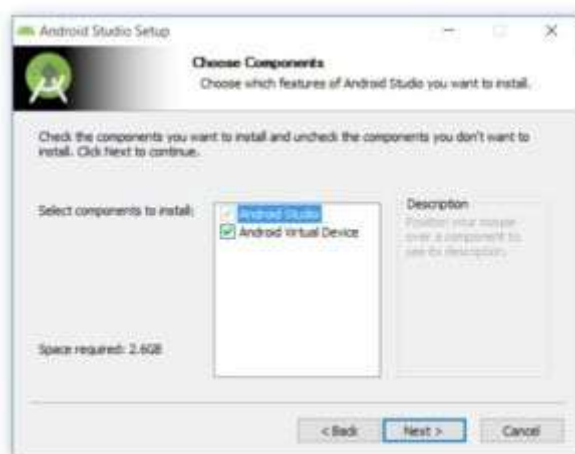
Kompyuter bu talablarga javob berishini bilganingizdan keyin, Android Studio o'zingizni operatsion tizimizga ko'chirib oling, Misol(*android-studio-ide-182.5314842-linux.zip*)

Android Studionni Windows 10 x64-bit ga o'rnatish
Ko'chirib bo'lganingizdan so'ng sizda xuddi shu ekran ochilishi kerak. Android Studio Setup (2.2.1-rasm)



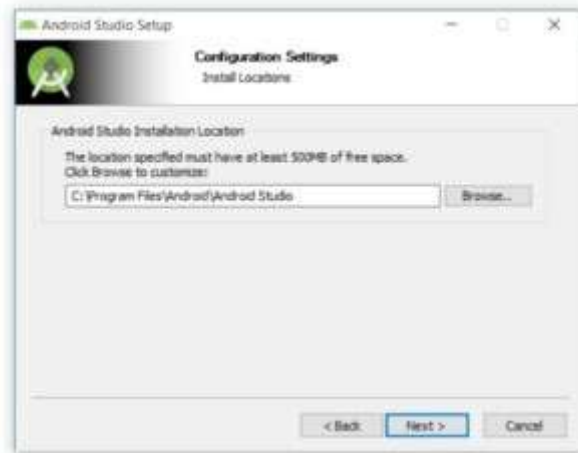
2.2.1-rasm. Android Studio ni Windows 10 ga o'rnatish

Next tugmasini bosgan holda keyingi oynaga o'tamiz, ya'ni Android Virtual Device paneliga (AVD).



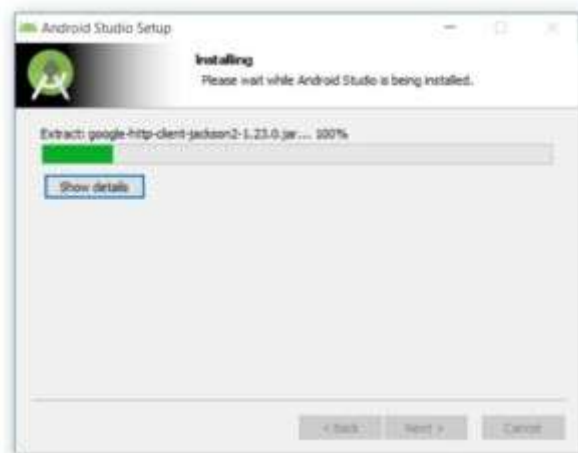
2.2.2-rasm. Android AVD ni o'rnatish

Yana **Next** tugmasini bosamiz va **Configuration Settings** bo'limiga o'tamiz. Bu yerda siz Android Studioni qayerga o'rnatish kerakligini belgilaysiz.



2.2.3-rasm. Configuration Settings bo'limi

Next tugmasini bosamiz, va Choose Start Menu Folder paneliga o'tamiz. Bu yerda ham hech narsaga tegmagan holda Install tugmasini bosamiz.



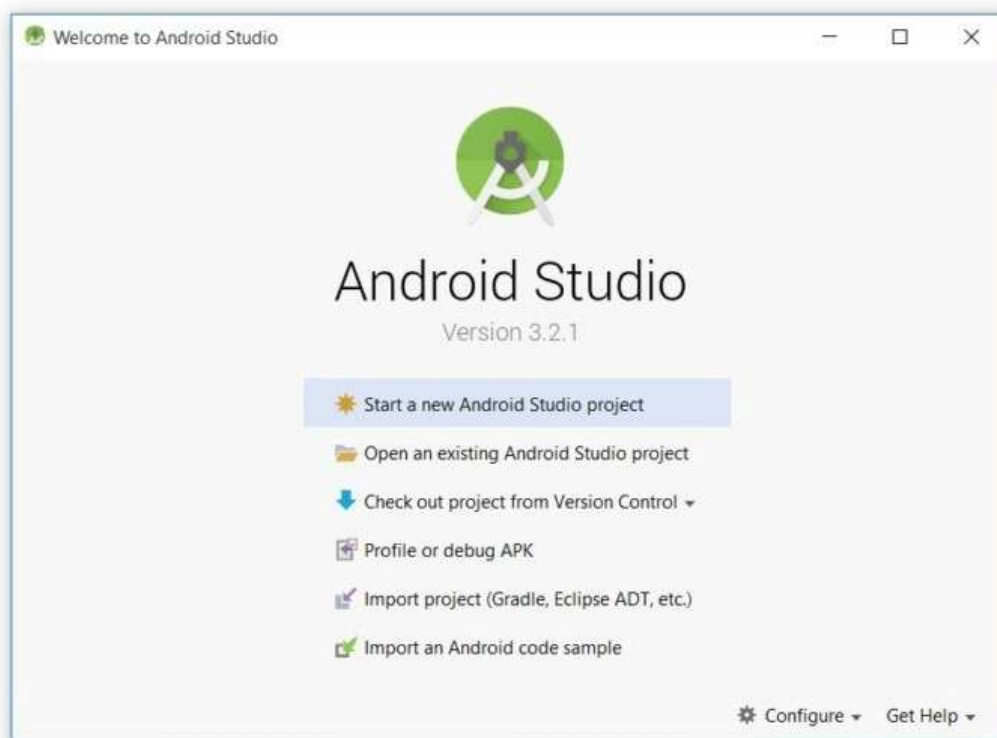
2.2.4-rasm. Android Studioni o'rnatish jarayoni boshlandi.

Finish tugmasi bosilishi bilan Android Studio avtomat ishga tushadi va quyidagi oyna hsoil bo'ladi.



2.2.5-rasm. Android Studio yuklanish jarayoni

Bundan so'ng **Welcome Android Studio** oynasiga o'tamiz.

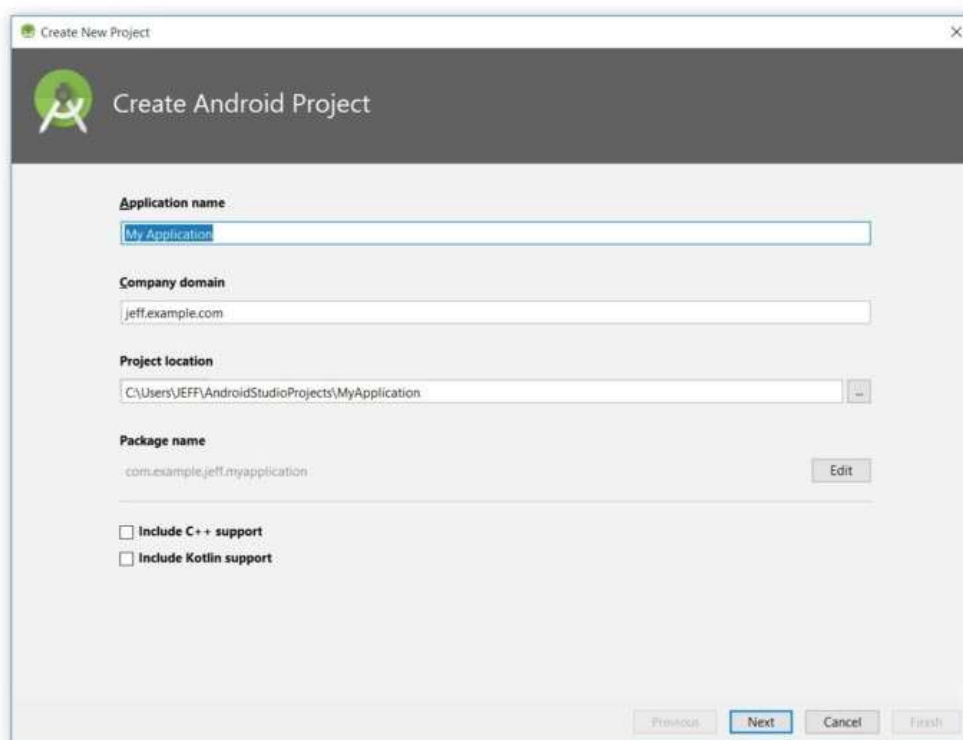


2.2.6-rasm. Welcome Android Studio oynasi

Shu bilan bizni Android Studioni o'rnatish jarayoni yakunlanadi va biz o'zimizni birinchi Android dagi loyihamizni **Start a new Android Studio project** tugmasi orqali ishga tushiramiz.

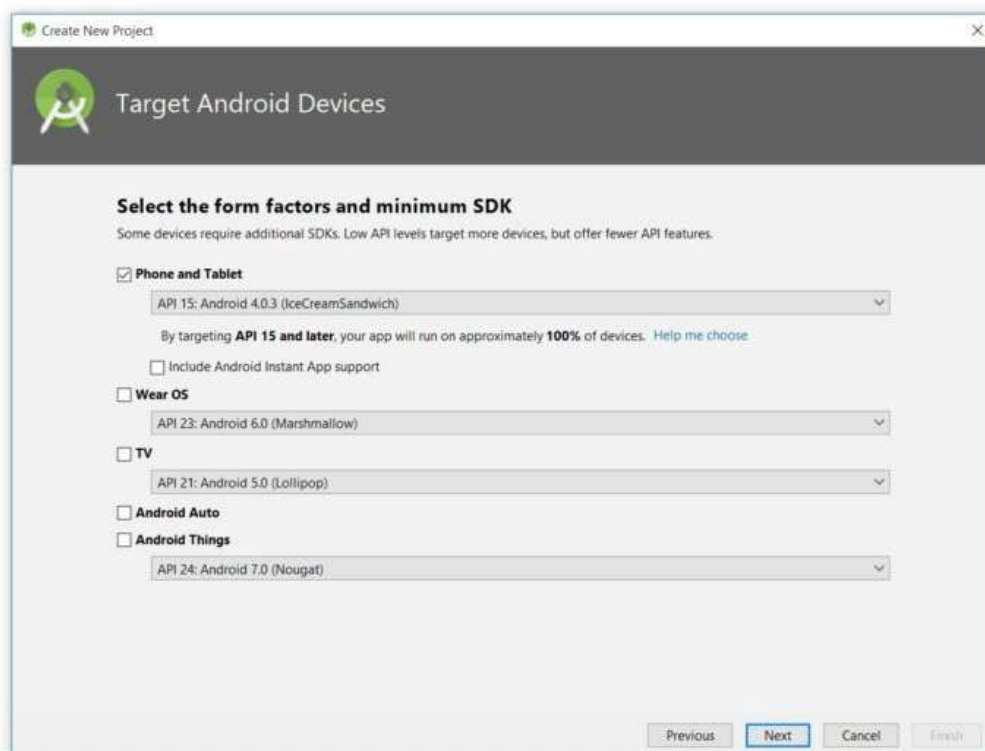
Android Studioni bilan tanishib chiqishni eng yaxshi yo'llaridan biri bu albatta ilova yaratish ya'ni **Hello World**.

Start a new Android Studio project tugmasini bosgan so'ng u bizni shu oynaga olib keladi.



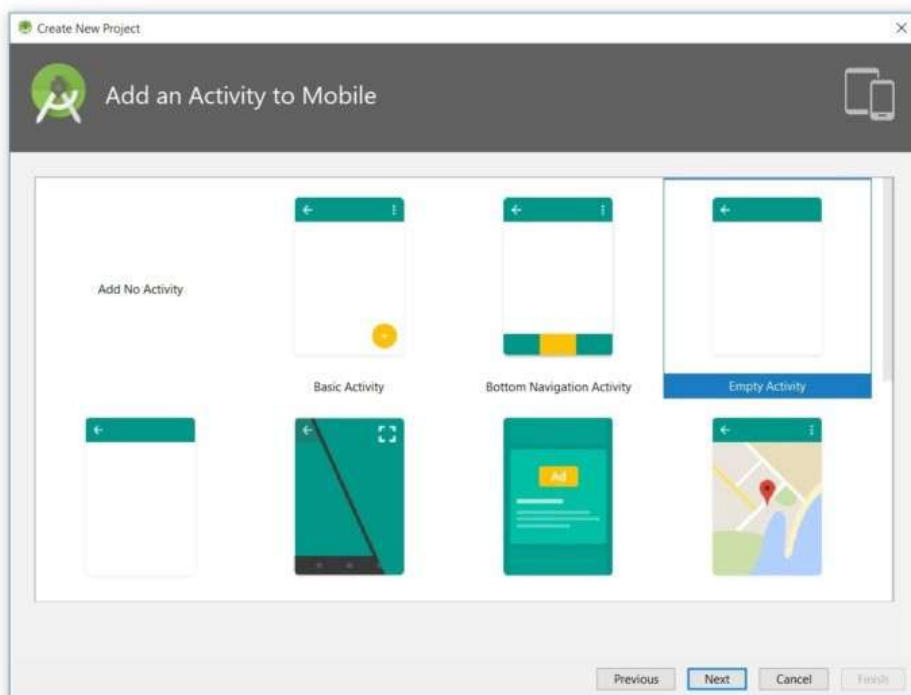
2.2.7-rasm. Create Android Project oynasi

Company domain o'zingiz hohlagan so'zlarni qo'ysangiz bo'ladi bu asosan siz ishlayotgan proektdan kelib chiqib yoki siz ishlayotgan kompaniyadan kelib chiqiladi. Bir papka oching Android loyihalaringizni o'sha yerda saqlash uchun va uni Project Location da ko'rsating va Next tugmasi bosiladi.



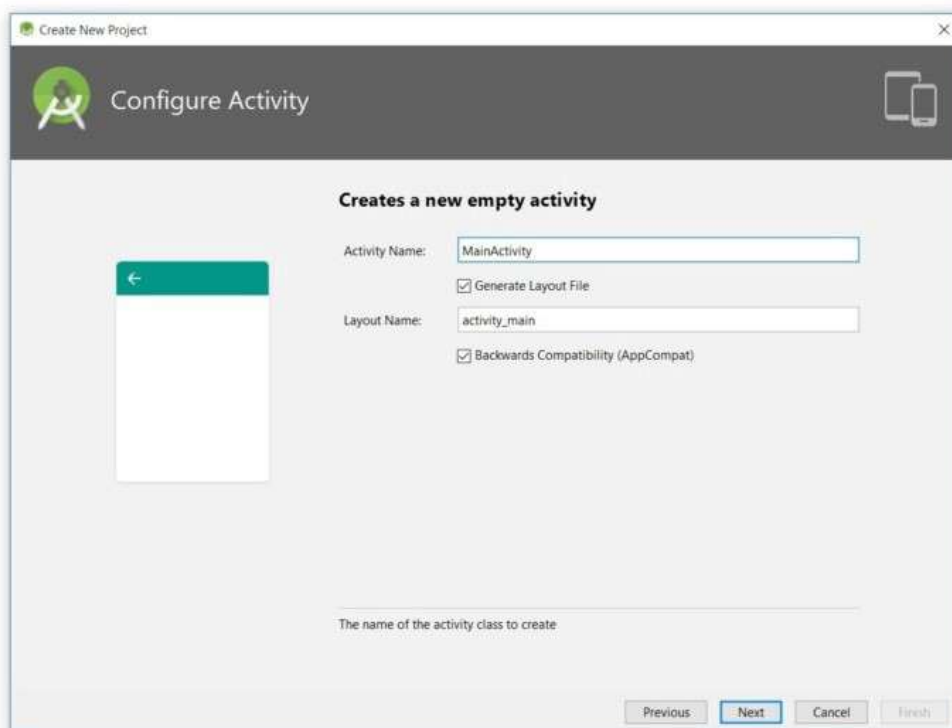
2.2.8-rasm. Android SDK va tabletni tanlash oynasi

Android Studio sizdan loyiha uchun kerakli faktorlarni tanlash imkoniyatni beradi ya'ni target devices... Buni boshida shunday qoldirsa ham bo'ladi va Next.

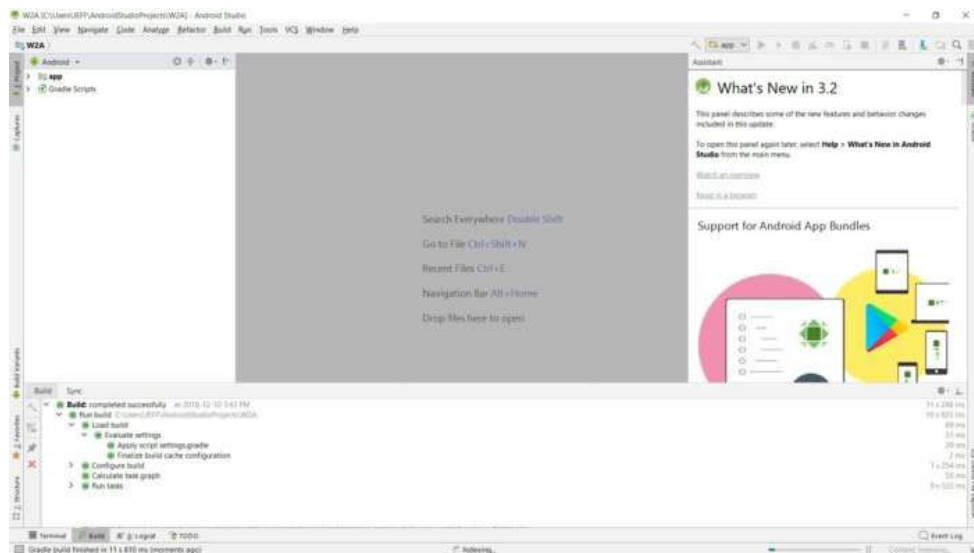


2.2.9-rasm. Activity tanlash oynasi

Endi activityni sozlashga ham keldik, o'zimiz uchun kerak bo'ladigan activity turini tanlaymiz va davom etamiz **Next**.



2.2.10-rasm. Activity konfiguratsiyalari oynasi



2.2.11-rasm. Android Studio dasturlash muhitining bosh oynasi

2.3. PhoneGap texnologiyasi

Mobil ilovalarni ishlab chiquvchilar har bir operatsion tizim uchun kod ishlab chiqishda turli xil SDKlar, kompilyatorlar va emulyatorlar bilan ishlashda qiyinchiliklarga duch kelishadi.

PhoneGap bu standart web-texnologiyalardan foydalangan holda bir nechta mobil platformalar uchun ilovalarni ishlab chiqishga imkon beruvchi Nitobi (hozirda Adobe tomonidan sotib olingan) kompaniyasining ochiq chiquvchi kodli platformasidir.

PhoneGap krossplatformali freymvork yaratuvchilari vazifani soddalashtirishga harakat qilishdi: ular oltita mobil platformani qo'llab-quvvatlovchi yagona open source mobil freymvork sifatida joylashtirilgan [8].

Hozirda Android, iOS, Symbian, Windows Phone, Bada, WebOS kabi operatsion tizimlari qo'llab-quvvatlanadi. Ko'rsatilgan operatsion tizimlarning har biri to'liq yoki cheklangan funksionallikka ega [9].

Ilova yozishda formatlash uchun HTML va CSSni qo'llab JavaScript-da yozilgan. Mobil ilovalar muntazam veb-sayt yoki web-xizmat sifatida yoziladi.

PhoneGap platformasi brauzer API-ni kengaytiradi va quyidagilarni qo'shib qo'yadi: akselerometrga kirish, kameraga kirish (hozircha faqat rasm), kompasga kirish, kontaktlar ro'yxatiga kirish, audio fayllarni yozib olish va

tinglash, fayl tizimiga kirish imkonini beradi, turli xil HTML5 omborlari localStorage, Web-SQL va sh.o'. bilan ishlash imkoniyatini beradi. Shuningdek, ishonchli tarzda har qanday kross-domenli manzilga murojaat qilish imkoniyati beradi [12].

PhoneGap-dan foydalanish uchun telefoningizning so'nggi versiyasini rasmiy saytdan yuklab olishingiz kerak va kutubxonaning rivojlanish muhitida joylashgan manzilini ko'rsatishingiz kerak. Bundan tashqari siz kerakli platformada kutubxona bilan birga jildda joylashgan JavaScript faylini ulashingiz kerak.

PhoneGap Build-dan foydalanish onlayn HTML5-konvertor bo'lib, ilovani html / js / css-ga yuklab olishingiz kerak bo'ladi va oxirida Apple App Store, Android Market, Palm, Symbian yoki BlackBerry uchun tayyor binar fayl hosil bo'ladi.

PhoneGap uchun ishlab chiqilgan ilovalar bir nechta afzalliklarga va kamchiliklarga ega.

Imkoniyatdan boshlaylik:

1. Ilovani birgalikda ishlab chiqish juda oson.
2. Kodni HTML, Java Script, CSS -da boshqa kutubxonalarini ishlatish orqali amalga oshiriladi.
3. Kross-platformani qo'llab-quvvatlash (hozirda 6 ta platforma);
4. Brauzer yordamida dasturlarni otladka qilish.
5. PhoneGap yordamida foydalanib, ilovalarni barcha zaruriy platformalarga konvertasiya qilish qobiliyatining mavjudligi.

PhoneGap uchun ishlab chiqilgan ishlanmalarning kamchiliklari:

1. Ilovalar ko'poqimlilikni qo'llab-quvvatlamaydi.
2. Uzun ro'yxatlarning (1000 dan ortiq) muammoli hal qilinishi.
3. Ishlab chiqilgan ilovalar hammasi ham web-ilovalar sifatida qabul qilinmasligi mumkin.
4. Mobil qurilmaning apparat qismiga kirish turli usullar bilan amalga oshiriladi, bu kutilmagan natijaga olib kelishi mumkin.

5. PhoneGap-da dasturlarni yozish uchun dasturiy ta'minot muhitini sozlashning murakkabligi.

PhoneGap platformasi yordamida yaratilgan ko'p foydalanuvchilarni qondiradigan kuchli krossplatformali ilova hisoblanadi. Sintaksisi va otladkaning soddaligi ushbu texnologiyani Qt yoki Java kabi tillarga nisbatan o'rganish vaqtini qisqartiradi. Natijada, agar dastur hisoblash va resurslardan unumli foydalanmasa, PhoneGap platformasi yordamida ishlab chiqish maqbuldir va turli xil operatsion tizimlar uchun o'zaro faoliyat platformalar ilovasini ishlab chiqish bilan solishtirganda jami kamroq vaqt talab etadi.

Ishlanmaning asosiy vositalari. PhoneGap bu mashhur instrumental vositalardan iborat gibril ilovalarni yaratish uchun platforma hisoblanadi.

Bu akselerometr va kamera kabi turli xil qurilmalar vazifalariga kirish uchun JavaScript-ni interfeyslarini o'z ichiga olgan ochiq kodli mobil infratuzilma.

Dastlabki talablar

Rivojlanish muhitini o'rnatish uchun quyidagi amallarni bajarishingiz kerak:

1. JDK va Eclipse yoki Rational Application Developer dasturini o'rnatish.
2. Android SDK-ni yuklab oling.
3. Eclipse uchun ADT pluginini yuklab oling va o'rnatish.
4. Android uchun Eclipse ni sozlang.
5. Android platformasi uchun kerakli bo'lgan dasturlarni o'rnatish.
6. Yangi Android Virtual Device (AVD) qurilmasini yarating.
7. PhoneGap SDK-ni yuklab oling.
8. Dojo Toolkit-ni yuklab oling.

JDK va Eclipse yoki Rational Application Developer ni o'rnatish

Avvalo, JDK 5 yoki undan yangi versiyasi o'rnatilganligini tekshiring. Aks holda Java SE-ni yuklab oling.

RAD IBM web-versiyasini qo'llab-quvvatlaydigan qo'shimcha IBM dasturlari bilan birga Eclipsening IBM versiyasini o'z ichiga oladi.

RADdan foydalanish uchun 8 yoki undan yuqori versiya talab qilinadi. RAD standart IBM JDK ni o'z ichiga oladi. Biroq, ushbu JDK Android ilovalarini yaratish va imzolash uchun zarur bo'lgan Java paketlarini o'z ichiga olmaydi.

Buning o'rniga Java SE JDK dasturidan foydalanish uchun C: / Program Files / IBM / SDP / eclipse.ini (yoki sizning RAD o'rnatmaning o'rnini) mazmunini Listing 1da ko'rsatilgan ma'lumotlarga almashtirishingiz kerak.

Ro'yxat 1. Windowsda RAD uchun eclipse.ini tarkibi

```
-startup
plugins/org.eclipse.equinox.launcher_1.1.1.R36x_v20101122_1400.jar
--launcher.library
plugins/org.eclipse.equinox.launcher.win32.win32.x86_1.1.2.R36x_v20101222
--launcher.defaultAction
openFile
--launcher.XXMaxPermSize
256M
-showsplash
org.eclipse.platform
--launcher.XXMaxPermSize
256m
--launcher.defaultAction
openFile
-install
C:/Program Files/IBM/SDP
-vmargs
-Dosgi.requiredJavaVersion=1.5
-Xms40m
-Xmx512m
```

Agar siz Eclipse dan foydalanishni xohlasangiz, sizga versiya 3.6 yoki undan keyin kerak bo'ladi. Ushbu seriyadagi web-xizmatlarni yaratamiz, Java EE Developer uchun Eclipse ID-ni yuklab oling. Agar siz Android dasturini yaratmoqchi bo'lsangiz, Java dasturchilari uchun Eclipse ID dan foydalanishingiz mumkin.

Android SDK'ni yuklab olish

Android ilovalarini yaratish uchun siz Android SDK-dan foydalanishingiz kerak. Bu Android dasturlarini kompilyatsiya qilish, yaratish, taqlid qilish va disk raskadrovka uchun ishlatiladigan buyruq qatorlari dasturlarining to'plamidir.

SDK Windows-ga o'rnatilgan bo'lsa, zip-to'plamini yuklab olish va uni C:\ diskning ildiz katalogiga chiqarishning eng oson yo'lidir.

R12 instalyatoriga Java-ni aniqlay olmaydigan va bo'sh joylarni o'z ichiga olgan katalogga o'rnatilishi bilan bog'liq bo'lgan bir nechta izohlar mavjud.

III BOB. DASTURIY VOSITA TAFSIFI VA UNDAN FOYDALANISH UCHUN TAVSIYALAR

Ushbu bobda biz yaratilgan dasturiy ilovaning asosiy komponentalari bilan tanishib o'tamiz va undan foydalanish bo'yicha tavsiyalar beramiz.

3.1. Python dasturlash tilini o'qitish uchun mobil trenajer-ilovasining tafsifi

Yaratilgan mobil ilova Python dasturini o'rganish uchun mo'ljallangan.

Mobil ilova quyidagi mavzularni o'z ichiga olgan:

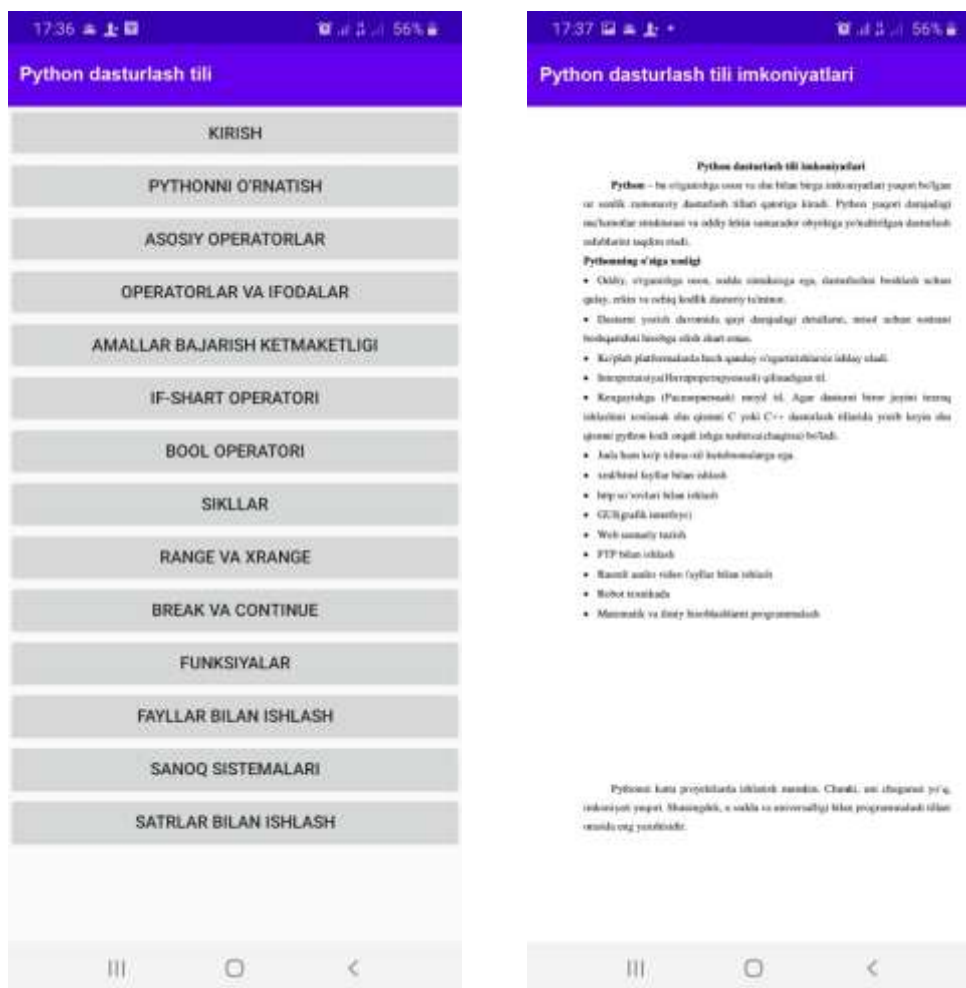
- Kirish;
- Pythonni o'rnatish;
- Asosiy operatorlar;
- Operatorlar va ifodalar;
- Amallar bajarish ketma-ketligi;
- Shartli operatorlar;
- Mantiqiy operatorlar;
- Sikllar;
- Range, xrange, break, continue;
- Funksiyalar bilan ishlash;
- Klasslar bilan ishlash;
- Sanoq sistemalari bilan ishlash va h.k.

Mobil ilovadan foydalanish oson va qulay. Navbatdagi bo'limda undan foydalanish buyicha tavsiyalarni berib o'tishni lozim deb topdik.

3.2. Dasturdan foydalanish bo'yicha tavsiyalar

Dasturiy vositamizdan foydalanish uchun uni mobil qurilmaga yuklab olish lozim. Dasturning bosh oynasi 3.1-rasmda keltirilgan.

Asosiy oynada yuqorida sanab o'tilgan bo'limlar mavjud bo'ib, ma'ruzalar bo'limiga barmoq tegishi bilan ma'ruzalar matni ochiladi, unda 10 mavzu yoritib berilgan (3.2-rasm).



3.2-rasm. Dasturning bosh oynasi va ma'ruzalar bo'limi oynasi ko'rinishi.

Ma'ruza materiallarini alohida ochib o'qib didaktik ma'lumotlar bilan tanishish mumkin (3.3-rasm).



3.3-rasm. Ma'ruza materiallari

Mustaqil ta'lim mavzulari bandida talabalar mustaqil ishlari mavzulari keltirilgan. Ushbu mavzular bo'yicha referativ shakldagi topshiriqlar bajariladi (3.5,3.5-rasm).

**O'ZBEKISTON RESPUBLIKASI AXBOROT TEXNOLOGIYALARI VA
KOMMUNIKATSIYALARI RIVOJLANTIRISH YAZIRLIGI**

**MUHAMMAD AL-XORAZMIY NIMIDAGI TOSHKENT AXBOROT
TEXNOLOGIYALARI UNIVERSITETI SAMARQAND FILIALI**

"KOMPYUTER INJINIRINGI" FAKULTETI

- 5350500 - "Kompyuter injiniringi (Kompyuter injiniringi,
AT-servis)"
5350600 - "Dasturiy injiniring"
5330300 - "Axborot xavfsizligi"

yo'nalishi talabalari uchun

"MA'LUMOTLAR KOMMUNIKATSIYASI"

**FANIDAN MUSTAQIL ISHLARNI BAJARISH UCHUN USLUBIY
KO'RSATMA**

*Uslubiy kechasiga uslubiy ko'rsatma
sifatida tavsiya etilgan*

SAMARQAND - 2019

1

Sattarov M.A., De'sanov X.T. "Ma'lumotlar kommunikatsiyasi" fanidan
mustaqil ishlarni bajarish uchun uslubiy ko'rsatma – TATU Samarqand filiali,
2019, 28 bet

Talabalar mustaqil ishlarni quyidagi maqolalarda bajaradilar:

3.4-rasm

Mustaqil ishlar mavzulari

№	Mustaqil ta'lim mavzulari	Berilgan topshiriqlar	Xajmi (soatda)
1	Kompyuter tarmoqlarida asbomt almashtirish jarayoni. Ma'lumotlarni uzatish usullari	Texnik adabiyotlar, Internet materiallarini yig'ish. Individual topshiriqlarni bajarish	6
2	Ma'lumotlarni uzatish va kommutsiyalash usullari, xanda vositalari. Ma'lumotlarni uzatish qurilmalari va ularning xillari.	Texnik adabiyotlar, Internet materiallarini yig'ish. Individual topshiriqlarni bajarish	8
3	Zamonaviy aloqa va kommutsiyalash tizimlarining texnik vositalarini va ko'rsatgichlarini o'rganish	Texnik adabiyotlar, Internet materiallarini yig'ish. Individual topshiriqlarni bajarish	8
4	Ma'lumotlarni uzatishda qo'llaniladigan kodlash Kodlarning xillar va ularning qo'llanilishi	Texnik adabiyotlar, Internet materiallarini yig'ish. Individual topshiriqlarni bajarish	8
5	OSI modeli asosida kompyuter tarmoqlarni qurish va unda ma'lumotlar kommunikatsiyasini tashkil qilish xususiyatlarini	Texnik adabiyotlar, Internet materiallarini yig'ish. Individual topshiriqlarni bajarish	8
6	Xar-xil protokollar asosida lokal tarmoqlarda ulanishlarni tashkil qilish xususiyatlarini chuqurroq o'rganish	Texnik adabiyotlar, Internet materiallarini yig'ish. Individual topshiriqlarni bajarish	8
7	Ma'lumotlar kommunikatsiyasida kommutatorlardan foydalanish	Texnik adabiyotlar, Internet materiallarini yig'ish. Individual topshiriqlarni bajarish	6
8	Ma'lumotlar kommunikatsiyasida marshrutizatorlardan foydalanish.	Texnik adabiyotlar, Internet materiallarini yig'ish. Individual topshiriqlarni bajarish	6
9	Lokal va global kompyuter tarmoqlari ma'lumotlarni uzatish	Texnik adabiyotlar, Internet materiallarini yig'ish. Individual topshiriqlarni bajarish	6

5

MUSTAQIL ISHNI BAJARISHGA MISOL

**MAVZU: MA'LUMOTLARNI UZATISH VA KOMMUTATSIYALASH USULLARI, XANDA
VOSITALARI. MA'LUMOTLARNI UZATISH QURILMAPARE VA ULARNING XILLARI.**

REJA

Kirish

1. Kommunikatsiya tizimining umumiy tushunchasi

3.5-rasm

Amadras bujarilich kotima-keeligi

2 + 3 * 4 (Belada qayy amal birinchi bo'ladik: qo'shishni sotki ko'paytirish)

[illegible]

Operator	Isde
Lambda	lambda (Isde)
De	Mustang 'yok'
And	Mustang 've'
Not &	Mustang 'yeme'
is, not is	Tegridadiles isohabirih
is, is not	Be isidiles isohabirih
is, is, is, is, not, is, not	Tegridadiles
	'yok' but operator
&	'shantik 'yok' but operator
!!	've' but operator

α_1, α_2	Soetliklar
α_1, α_2	Qorilik va qatilik
$\alpha_1, \alpha_2, \alpha_3$	Ko'paytirish, bo'lish, qoldiqqa bo'lish va qoldiqlik bo'lish
α_1, α_2	Mislat va misli
α_1	Imoni (bi operator)
α_1	Darajaga ko'tarish

E-shaft operator.

Исправки статьи вкратце таковы: исправил. Pythona chart operationa на место и в кавычки ввел.

5. If (multiplicity holds) then operating on the variable multiplicity holds and the type holds regardless of how many values are calculated.

2. **if** *ismatigiz* (Hafiz)... *afso-khat* *aynamatigiz* *bu* *ku* *istidhola* *namatigiz* *Yoda* *ma* *ku* *bu*, *harachi* *shidalar* *hiki* *hajarad* *bu* *hik* **"if-hik"** *zib* *namatigiz*) *aka* *hikla* *harachi* *shidalar* *hiki* *hajarad* *bu* *hik* **"afso-hik"** *zib* *namatigiz*)

[illegible]

If operator is *mixed*:

```

int
of_natrevr
  (uint32_t data, *p2, *p1)

```

Natijs

1. $\frac{1}{2}$ of the 1000 is 500.

Kürzungen werden je nach Bedarf angeschlossen.

```

if (age < 18)
    print("too young")
else
    print("too old")

```

Modique

[illegible]

Challenging current medical management strategies

```

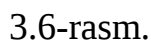
    def __init__(self, name, value):
        self.name = name
        self.value = value
        self.parent = None
        self.children = []

    def __str__(self):
        return f"{self.name}: {self.value}"

    def __repr__(self):
        return f"Node({self.name}, {self.value})"

```

Read time



32

3.3. Kompyuter xonalarida ishlashda hayot havfsizligi qoidolari

Zamonaviy axborot kommunikatsiya tizimlaridan, jumladan axborot va kompyuter texnologiyalari, raqamli va keng formatli telekommunikatsiyalar, Internetni nafaqat oliy o'quv yurtlari, kollej, litsey, maktablarda balki har bir oilada joriy etish kundan-kunga rivojlanib bormoqda. Aynan zamonaviy aloqa va axborot texnologiyalarini keng ko'lamda rivojlantirish mamlakatimiz va jamiyatimizning taraqqiyot darajasini ko'rsatadigan mezonlardan biri bo'lib qolmoqda.

Axborot - kommunikatsiya texnologiyalari asosida axborotni uzatish, qabul qilish, saqlash, qayta ishlash, axborot matnini yozish, qog'ozga chiqarish, nusxalarini ko'paytirish kabi jarayonlar bir qancha texnik vositalar (zamonaviy kompyuterlar, printerlar, kserokopiyalar, mobil aloqa vositalari, televideniya, radio va b.) yordamida amalga oshiriladi. Albatta axborot texnologiyalarining rivojlanishi bir qancha qulayliklar va imkoniyatlar yaratsada, shu bilan bir qatorda inson hayoti va sog'ligi uchun zararli va xavfli bo'lgan turli xil omillarni keltirib chiqaradi.

Xalqaro axborot resurslaridan foydalanish bo'yicha mutlaqo yangi imkoniyatlar yaratilayotgan bir vaqtda, ushbu tizimda ishlatiladigan texnik vositalardan xavfsiz foydalanishni ham bilish muhim ahamiyat kasb etadi.

- Stol imkoni boricha katta bo'lishi lozim. Bu asosiy shartdir, chunki agar barcha moslamalarni o'rnatish uchun joy kamlik qilsa, ergonomika to'g'risida eslamasa ham bo'ladi. Stolning balandligi qorin o'rtasi bilan bir sathda bo'lishi, oyoqlar polda tekis turishi, bo'ksa pol bilan parallel bo'lishi, gavda esa tik holatda bo'lishi kerak.

- Shuningdek, stol qancha og'ir bo'lsa, shuncha yaxshi. Stol qimirlamay, mahkam turishi kerak, chunki vibratsiya – texnika dushmandir. 2ta stolni to'g'ri burchak ostida bir-biriga yonma-yon qilib qo'ysangiz undan ham yaxshi, bunda ikkinchi stolni ishlayotgan qo'lingiz sichqonchadan bemalol foydalanishi uchun o'ng tomonga qo'ying. Stol va devor o'rtasi ochiq tursin.

Yaxshisi komputer uchun maxsus stol olganingiz ma'qul.

▪ Gipodinamiya – inson organizmi (qon aylanish, nafas olish, taomni hazm qilish, tayanch –harakat apparati) funksiyalari harakat qilish faolligining cheklanganligi, mushaklar qisqarishi kuchining pasayishi natijasida buzilishidir. Mehnatni avtomatlashtirish va mexanizasiyalashtirish, kommunikasiya vositalari rolining o‘ssishi tufayli gipodinamiya kundankunga ommalashib bormoqda (accoopa.ru.dagi ensiklopediya). Bu muammo bevosita komputer bilan bog‘liq emas. O‘tirgan holda mehnat qiladigan odamlarning barchasi uchun gipodinamiyaga chalinish xavfi bor. Uzoq vaqt o‘tirganda gavdaning har qanday holati tayanch-harakat apparati uchun zararli bo‘lib, bundan tashqari ichki a‘zolarida va kapillarlarda qonning to‘xtab qolishiga olib keladi. Ko‘proq revmatik og‘riqlar va varikoz paydo bo‘ladi.

- Bunday hollarning oldini olish uchun:
 - - komputer bilan ishlash vaqtida tez-tez gavda holatini o‘zgartirib turing;
 - - miriqib kerishing;
 - - stol tagida oyoqlar holatini o‘zgartirib turishni unutmang;
 - - telefonni o‘zingizdan uzoqroq joyga qo‘ying;
 - - mabodo kofe, choy yoki boshqa ichimlik ichmoqchi bo‘lsangiz, also komputer oldida ichmang, buning uchun qulay bo‘lgan boshqa joyga o‘ting;
 - - vaqt-vaqti bilan turib, shunchaki sayr qilishga erinmang;
 - - ish vaqtida deraza oynasi yoniga kelib, ko‘chani tomosha qilib turishni odat qiling, bu har tomonlama foydalidir.
- - komputer bilan ishlashdan avval va undan keyin vaqtingizni faollikda o‘tkazing – sayr qiling, sport o‘yinlarini o‘ynang, raqsga tushing, xullas, istaganingizni qiling, lekin harakat qiling!
- Komputer bilan ishlash davridagi harakatsizlikni serharakatlik bilan almashtiring.
- Bir xil harakatlarni bajaruvchi mushaklar guruhi uchun charchash zararlidir.

Charchash bo'g'in va paylarning shikastlanishiga olib kelishi mumkin.

Komputer sichqonchasi va klaviatura bilan uzoq vaqt bir xil holatda ishlash natijasida qo'l paylari tendovaginiti ayniqsa ko'p uchraydi. Tendovaginit – (yangi lotinchadan tendo – pay) paylarning yallig'lanishidir (ko'proq qo'l panjalari, bilak, tizza). Zo'riqish, shikastlanish, panarisiya va komputer bilan uzoq vaqt muntazam ishlash natijasida bu kasallik rivojlanib boradi.

Belgilari: paylar shishadi, og'riq paydo bo'ladi.

- Ichidagi, tarkibida og'ir metall bo'lgan kukun ham g'oyat xavflidir.

Shuningdek, uzoq vaqt komputer oldida ishlash ko'z uchun nihoyatda zararlidir.

- stresslarga berilish darajasi oshadi. Yurak- tomir sistemasi va yuqori nafas olish yo'llari kasalliklari vujudga keladi, immunitet pasayib ketadi. *Lekin quyidagi maslahatlarga amal qilsangiz, bu dardlarni chetlab o'tishingiz mumkin:*

- Monitordan 40-50 sm masofada bo'ling;
- Sifatli himoyasi bo'lgan yaxshi monitor sotib oling;
- Monitoringdagi tasvir yetarli darajada aniq bo'lsin;
- Tez-tez nam latta bilan komputerni artib turing, chunki chang, ayniqsa monitordagi chang nurni o'zida to'plash qobiliyatiga ega; - Havo ionizatorlaridan foydalaning.

- —Ish joyingizni tashkil etishga alohida ahamiyat bering: monitorgacha bo'lgan masofa 40-50 smdan kam bo'lmasligi; 2ta ishlayotgan komputer orasidagi masofa 1 metrdan kam bo'lmasligi shart, aks holda ulardan chiqayotgan magnit maydonlari birlashib, nur taratishni kuchaytiradi. Nur kamroq taralishi va ko'zlarga yuk kamroq tushishi uchun monitordagi tasvir ravshanligi minimal darajada bo'lishi kerak. 45 daqiqa davomida komputerda ishlang, 15 daqiqa dam oling, xonani tez-tez shamollatib turing. Ish joyingizni to'g'ri yoritish yodingizdan chiqmasin. Ekranga nurlar tushmasligi uchun

yorug'lik chap tomondan tushishi kerak. Texnika xavfsizligi qoidalariga qat'iy amal qiling.

Umuman, agar o'z xavfsizligingiz va sog'lingiz haqida qayg'ursangiz, o'zingizga g'amxo'rlikni sifatli texnika sotib olishdan boshlang. Yaxshi monitorlarda turli yo'nalishda nur taralishini kamaytirish maqsadida ekran usti va korpus ichi himoya qoplamasi bilan qoplanadi.

- Odatda bir daqiqada 20 marta kiprik qoqamiz. Bunda ko'z yoshlarini ishlab chiqaruvchi bezlar shohpardani namlaydi. Ekranga tikilib o'tirganimizda esa 3 marta kamroq kiprik qoqamiz. Ko'zlar —quriydi, achishadi. Mutaxassislar ko'z shohpardasini sun'iy ravishda namlab, ko'zlarga yordam berishini maslahat beradilar;

- Ko'z va ob'ekt orasidagi optimal masofa 33 santimetrni tashkil etadi. Biroq komputer ekrani odatda 50 santimetrdan ortiq masofada bo'ladi, bu esa qo'shimcha zo'riqishga olib keladi, natijada ko'zlar uzoqni yaxshi ko'rolmaydigan bo'lib qoladi;

- Vaqt faktori: kamdan-kam inson 8-9 soat davomida kitob mutolaa qiladi.

Komputer bilan ishlaganda esa bu odatiy hol hisoblanadi. Natijada ko'zlar zo'riqadi.

Monitoringiz ekranidagi tasvirlarning alohida belgilarini aytib o'taylik:

- u o'zidan nur taratadi, tasvir aks etmaydi;
- kontrastlik past, yoritilgan xona uni yanada pasaytiradi;
- tasvir lipillaydi, ya'ni nuqtalar ma'lum bir chastotada yonadi va o'chadi;

- aniq chegarasi yo'q, chunki piksel asta-sekinlik bilan fonga o'tib ketadi. Komputerlarning inson ko'ziga zararli ta'siri ular paydo bo'lganidayoq ma'lum bo'lgan. Monitor oldida bir necha soat o'tirib ishlagandayoq ko'zlar charchar, qizarib ketar va yoshlanar edi. So'ngra boshda og'riq paydo bo'lar edi. Ko'zlari uzoqni ko'ra olmaydigan, astigmatizmi bor, yaqinni ko'ra olmaydigan xodimlarning ko'rish qobiliyati keskin pasayib ketdi, ular tez-tez

ko‘zoynaklarini almashtirar, so‘ngra asta-sekin nogironlarga aylanib borar edilar. Ilgari ko‘rish qobiliyatining pastligidan nolimaydiganlar esa endi ko‘zoynak taqa boshlar va yuqoridagilar qatoridan joy olar edilar.

Nafas olayotganingizda boshingiz bilan asta-sekin aylana bo‘ylab chap tomonga harakat qiling. Yelkalarinigizni bo‘sh qo‘ying.

Nafas chiqarayotganingizda boshingiz bilan o‘ng tomonga va pastga harakat qiling, so‘ngra boshlang‘ich holatga qayting.

Bu mashqni 2 marta takrorlang, so‘ng harakat yo‘nalishini o‘zgartiring va mashqni 3 marta takrorlang. Maqsad—ensa qismlarining akupressur nuqtalarini yaxshilash va bo‘yin mushaklarining taranglashuvini yo‘qotish.

Bu mashq o‘tirgan holda 40 soniya davomida (har bir tomonga) bajariladi. Akupressur nuqtalar – kuchli sezuvchanlikka ega qismlardir. Ensa akupressur nuqtalari umurtqa pog‘onasining ikkala tomoni va bosh suyakning pastki qismida joylashgan bo‘yin orqa tomoni mushaklari kesishgan joyda joylashgan(rasm) Nafas ola turib ikki barmog‘ingiz uchlari bilan ohista akupressur nuqtalarni bosing. Bosimni kamaytirmay, 10 soniya davomida o‘ng va chap tomonga aylana bo‘ylab asta-sekin harakat qiling. Bosimni kamaytiring, nafas chiqaring. Maqsadensa qismlarining akupressur nuqtalarini yaxshilash va bo‘yin mushaklarining taranglashuvini yo‘qotish. Bunday mashqalar imson organizmiga foydalai bo‘lib biroz bo‘lsa han toliqishning oldini oladi.

Bunday maslahatlarga amal qilingan holda ish jarayonimizni yo‘lga qo‘ysak maqsadga muvofiq bo‘ladi.

XULOSA

Hozirgi paytda ta'lim jarayonida turli xil elektron darsliklar, o'rgatuvchi va nazorat qiluvchi dasturlardan keng foydalanilmoqda. Bu vositalar yordamida ta'lim jarayonini avtomatlashtirish va ta'lim sifatini oshirishga erishilmoqda. Bundan tashqari mobil qurilmalar uchun mo'ljallangan o'quv resurslar so'nggi paytlarda rivojlanib borayotgan masofaviy o'qitish texnologiyasining asosiy elementlaridan biri hisoblanadi.

Bitiruv malakaviy ishini bajarish jarayonida Oliy ta'lim muassasalarida o'qitiladigan Python dasturlash tili fanini mazmun mohiyatini yorituvchi mobil ilova yaratish masalasi qaraldi.

Ushbu maqsad yo'lida quyidagi masalalar yechildi:

- Mobil qurilmalarning dasturiy ta'minoti bo'yich nazariy ma'lumotlar o'rganildi va arxitekturasini tahlil qildik;
- mobil ilovani ishlab chiqishdagi talablar o'rganildi, mobil ilovaning sifati va asosiy ko'rsatkichlarini tahlil qilqik;
- Android va iOS operatsion tizimlarining qiyosiy tahlili ishlab chiqildi;
- Android platformasi arxitekturasini haqidagi ma'lumotlar o'rganildi;
- “Python dasturlash tili” fanidan mobil ilovasinining tuzilishi loyihasini ishlab chiqdik;
- Python dasturlash tilini o'qitish uchun mobil trenajer-ilovasini yaratishda PhoneGap texnologiyasini qo'llash tavsiflari keltirildi;
- mobil ilovadan foydalanish uchun ko'rsatmalar ishlab chiqildi;

Yaratilgan ilovadan Oliy ta'lim muassasalari talabalari, mustaqil tadqiqotchilar hamda professor o'qituvchilar “Python dasturlash tili” fanini o'qitishda, shuningdek ixtiyoriy foydalanuvchilarga kerakli ma'lumiotlarni taqdim etishda foydalanish mumkin

FOYDALANILGAN ADABIYOTLAR RO'YXATI

1. Тереза Нейл. Мобильная разработка. Галерея шаблонов.
2. Дейтел П., Дейтел Х., Уолд А., Android для разработчиков. 3-е издание, 2016.
3. Майер Р. Android 4. Программирование приложений., 2013г.
4. Сильвен Ретабоуил - Android NDK. Руководство для начинающих".2-е издание.2017
5. Брайн Харди, Билл Филлипс. Программирование под Android, 2014г.
6. Угрюмов э.П. Сифровая схемотехника. - СПб: БХВ, Санкт-Петербург, 2000г.
7. Герберт Шилдт. Java 8. Руководство для начинающих. 2015г.
8. Филлипс Б., Стюарт К., Марсикано К. Android. Программирование для профессионалов. 2017.
9. Яценков В. Java за неделю. Вводный курс. 2018.
10. Дарвин Ян. Android. Сборник рецептов. Задачи и решения для разработчиков приложений. 2018.
11. Jerome F. Dimarzio, Wei-Meng Lee. Beginning Android Programming with Android Studio, 4th Edition (2017).
12. Васильев А. Н. Python на примерах. Практический курс по программированию. - СПб.: Наука и Техника, 2016. - 432 с.
13. Irv Kalb. Learn to Program with Python. Mountain View, California, USA. 2016. ISBN-13 (pbk): 978-1-4842-1868-6, DOI 10.1007/ 978-1-4842-2172-3.
14. Златопольский Д. М. Основы программирования на языке Python. – М.: ДМК Пресс, 2017. – 284 с. ISBN 978-5-97060-552-3.
15. Raximov O.D. Hayot faoliyat xavfsizligi. O'quv - uslubiy majmua. - Qarshi, 2012, 857b.

16. <https://www.macdigger.ru/news/post/kolichestvo-prilozhenij-v-app-store-prevysilo-15-milliona>
17. <https://developer.android.com/studio/index.html>

Internet manbaalari

- a [http://: www.gov.uz.](http://www.gov.uz)
- b [http://: www.mitc.uz](http://www.mitc.uz)
- c [http://: www.ziyonet.uz.](http://www.ziyonet.uz)
- d [http://: www.tuit.uz.](http://www.tuit.uz)
- e [http://: www.tuitkf.uz.](http://www.tuitkf.uz)
- f <http://developer.android.com>
- g [http://wrox.com.](http://wrox.com)
- h <http://www.it-ebooks.info>
- i <http://www.appmk.com>
- j [www.wiley.com.](http://www.wiley.com)
- k www.ziyonet.uz

ILOVA

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.newbook">
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".ResultTest"></activity>
        <activity android:name=".Test" />
        <activity android:name=".MeyoriyHujjatlar" />
        <activity android:name=".Amaliyotlar" />
        <activity android:name=".Maruzalar" />
        <activity android:name=".Muallif" />
        <activity android:name=".Meyoriy" />
        <activity android:name=".Laboratoriya" />
        <activity android:name=".Amaliyot" />
        <activity android:name=".Maruza" />
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

AndroidManifest.java

```
package com.example.newbook;
```

```
import android.content.DialogInterface;
```

```
import android.content.Intent;
```

```
import android.support.v7.app.AlertDialog;
```

```
import android.support.v7.app.AppCompatActivity;
```

```
import android.os.Bundle;
```

```
import android.view.View;
```

```
import android.widget.TextView;
```

```
public class MainActivity extends AppCompatActivity {
```

```
    private TextView m,a,l,c,textView,me,mu,t;
```

```
    private AlertDialog.Builder builder;
```

```
    @Override
```

```
    protected void onCreate(Bundle savedInstanceState) {
```

```
        super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.activity_main);
```

```
        m=findViewById(R.id.btn1);
```

```
        a=findViewById(R.id.btn2);
```

```
        l=findViewById(R.id.btn3);
```

```
        c=findViewById(R.id.btn4);
```

```
        me=findViewById(R.id.btn5);
```

```
        mu=findViewById(R.id.btn6);
```

```
        t=findViewById(R.id.btn7);
```

```
        t.setOnClickListener(new View.OnClickListener() {
```

```

@Override
public void onClick(View v) {
    Intent intent=new Intent(MainActivity.this, Test.class);
    startActivity(intent);
}
});

m.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
    Intent intent=new Intent(MainActivity.this, Maruza.class);
    startActivity(intent);
}
});

a.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
    Intent intent=new Intent(MainActivity.this, Amaliyot.class);
    startActivity(intent);
}
});

l.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
    Intent intent=new Intent(MainActivity.this, Laboratoriya.class);
    startActivity(intent);
}
});

me.setOnClickListener(new View.OnClickListener() {
@Override

```

```

public void onClick(View v) {
    Intent intent=new Intent(MainActivity.this, Meyoriy.class);
    startActivity(intent);
}
});
mu.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intent=new Intent(MainActivity.this, Muallif.class);
        startActivity(intent);
    }
});

```

```

c.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        builder = new
android.support.v7.app.AlertDialog.Builder(MainActivity.this);
        builder.setTitle(getResources().getString(R.string.title));
        builder.setMessage(getResources().getString(R.string.message));
        builder.setCancelable(false);

        builder.setPositiveButton(getResources().getString(R.string.yes), new
DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                MainActivity.this.finish();
            }
        });
    }
});

```

```

        builder.setNegativeButton(getResources().getString(R.string.no), new
DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                dialog.dismiss();
            }
        });

        android.support.v7.app.AlertDialog dialog = builder.create();
        dialog.show();
    }
});
}
}

package com.example.newbook;

import android.content.Context;
import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.LinearLayoutManager;
import android.support.v7.widget.RecyclerView;
import android.support.v7.widget.Toolbar;
import android.view.MenuItem;
import android.view.View;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

```

```

import com.example.newbook.Adapter.TitleAdapter;
import com.example.newbook.forClick.CustimItemClickistener;
import com.example.newbook.items.Book;
import com.example.newbook.nomlar.MaruzaNomlari;

import java.util.ArrayList;
import java.util.List;

public class Maruza extends AppCompatActivity {

    ArrayList<String> titleArrayList;
    ArrayList<String> titleArrayList2;
    private RecyclerView mRecyclerView;
    private ImageView img;
    List<Book> lstBook;
    public      int      first=0,      a=0,b=0,c=0,d=0,e=0,f=0,g=0,h=0,
i=0,j=0,k=0,l=0,m=0,n=0,o=0;
    public      String
ism,ism2,ism3,ism4,ism5,ism6,ism7,ism8,ism9,ism10,ism11,ism12,ism13,ism14
;
    TextView textView;
    private Context mContext;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_maruza);

        img = (ImageView)findViewById(R.id.title_img2);
        textView=findViewById(R.id.dk);

```

```

Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
setSupportActionBar(toolbar);

if (getSupportActionBar() != null){
    getSupportActionBar().setDisplayHomeAsUpEnabled(true);
    getSupportActionBar().setDisplayShowHomeEnabled(true);
}

```

```

mContext = Maruza.this;
titleArrayList = new ArrayList<String>();
titleArrayList2 = new ArrayList<String>();
titleArrayList.add(MaruzaNomlari.Maruza_1);
titleArrayList.add(MaruzaNomlari.Maruza_2);
titleArrayList.add(MaruzaNomlari.Maruza_3);
titleArrayList.add(MaruzaNomlari.Maruza_4);
titleArrayList.add(MaruzaNomlari.Maruza_5);
titleArrayList.add(MaruzaNomlari.Maruza_6);
titleArrayList.add(MaruzaNomlari.Maruza_7);
titleArrayList.add(MaruzaNomlari.Maruza_8);
titleArrayList.add(MaruzaNomlari.Maruza_9);
titleArrayList.add(MaruzaNomlari.Maruza_10);

```

```

lstBook=new ArrayList<>();

```

```

lstBook.add(0,new Book("1-mavzu", R.drawable.ic_action_block1 ));
lstBook.add(1,new Book("2-mavzu", R.drawable.ic_action_block1));
lstBook.add(2,new Book("3-mavzu", R.drawable.ic_action_block1));
lstBook.add(3,new Book("4-mavzu", R.drawable.ic_action_block1));

```

```

lstBook.add(4,new Book("5-mavzu", R.drawable.ic_action_block1));
lstBook.add(5,new Book("6-mavzu", R.drawable.ic_action_block1));
lstBook.add(6,new Book("7-mavzu", R.drawable.ic_action_block1));
lstBook.add(7,new Book("8-mavzu", R.drawable.ic_action_block1));
lstBook.add(8,new Book("9-mavzu", R.drawable.ic_action_block1));
lstBook.add(9,new Book("10-mavzu", R.drawable.ic_action_block1));

```

```

mRecyclerView=findViewById(R.id.title_recyclerview);
mRecyclerView.hasFixedSize();
mRecyclerView.setLayoutManager(new
LinearLayoutManager(getApplicationContext()));

```

```

TitleAdapter adapter = new TitleAdapter(mContext, lstBook, new
CustimItemClickistener() {
    @Override
    public void onItemClick(View v, int position) {
        Intent intent=new Intent(Maruzza.this, Maruzalar.class);
        intent.putExtra("titles", (position));
        intent.putExtra("title", titleArrayList.get(position));
        startActivity(intent);
    }
});

mRecyclerView.setAdapter(adapter);
}

```

```

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    if (item.getItemId()== android.R.id.home){
        Intent intent=new Intent(Maruza.this, MainActivity.class);
        startActivity(intent);
    }

    return true;
}
}

package com.example.newbook;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.Toolbar;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.widget.TextView;

import com.github.barteksc.pdfviewer.PDFView;

public class Maruzalar extends AppCompatActivity {

    private Bundle extras, name;
    private TextView view;
    public int first; int a=12;

    @Override

```

```

protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_maruzalar);
    PDFView pdfView;

    Toolbar toolbar = (Toolbar)findViewById(R.id.toolbar_test);
    setSupportActionBar(toolbar);

    if (getSupportActionBar() != null){
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);
        getSupportActionBar().setDisplayShowHomeEnabled(true);
    }

    pdfView=findViewById(R.id.pdfView);
    name = getIntent().getExtras();
    if (!name.equals(null)){
        first = name.getInt("titles");
        String data = name.getString("title");
        toolbar.setTitle(data);
    }

    switch (first)
    {
        case 0: pdfView.fromAsset("maruza1.pdf").load(); break;
        case 1: pdfView.fromAsset("maruza2.pdf").load(); break;
        case 2: pdfView.fromAsset("maruza3.pdf").load(); break;
        case 3: pdfView.fromAsset("maruza4.pdf").load(); break;
        case 4: pdfView.fromAsset("maruza5.pdf").load(); break;
        case 5: pdfView.fromAsset("maruza6.pdf").load(); break;
    }
}

```

```

        case 6: pdfView.fromAsset("maruza7.pdf").load(); break;
        case 7: pdfView.fromAsset("maruza8.pdf").load(); break;
        case 8: pdfView.fromAsset("maruza9.pdf").load(); break;
        case 9: pdfView.fromAsset("maruza10.pdf").load(); break;
    }
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    MenuInflater inflater = getMenuInflater();
    inflater.inflate(R.menu.men_maruzalar, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    if (item.getItemId() == android.R.id.home) {
        Intent intent = new Intent(Maruzalar.this, Maruza.class);
        startActivity(intent);
    }

    int res_id = item.getItemId();
    if (res_id == R.id.action_settings) {
        switch (first) {
            case 0:
                Intent intent = new Intent(Maruzalar.this, MainActivity.class);
                intent.putExtra("titles", a);
                startActivity(intent);
                break;
            case 1:
                Intent intent1 = new Intent(Maruzalar.this, MainActivity.class);
                startActivity(intent1);
                break;
        }
    }
}

```

case 2:

```
Intent intent2 = new Intent(Maruzalar.this, MainActivity.class);  
startActivity(intent2);  
break;
```

case 3:

```
Intent intent3 = new Intent(Maruzalar.this, MainActivity.class);  
startActivity(intent3);  
break;
```

case 4:

```
Intent intent4 = new Intent(Maruzalar.this, MainActivity.class);  
startActivity(intent4);  
break;
```

case 5:

```
Intent intent5 = new Intent(Maruzalar.this, MainActivity.class);  
startActivity(intent5);  
break;
```

case 6:

```
Intent intent6 = new Intent(Maruzalar.this, MainActivity.class);  
startActivity(intent6);  
break;
```

case 7:

```
Intent intent7 = new Intent(Maruzalar.this, MainActivity.class);  
startActivity(intent7);  
break;
```

case 8:

```
Intent intent8 = new Intent(Maruzalar.this, MainActivity.class);  
startActivity(intent8);  
break;
```

case 9:

```

        Intent intent9 = new Intent(Maruzalar.this, MainActivity.class);
        startActivity(intent9);
        break;
    case 10:
        Intent intent10 = new Intent(Maruzalar.this, MainActivity.class);
        startActivity(intent10);
        break;
    case 11:
        Intent intent11 = new Intent(Maruzalar.this, MainActivity.class);
        startActivity(intent11);
        break;
    case 12:
        Intent intent12 = new Intent(Maruzalar.this, MainActivity.class);
        startActivity(intent12);
        break;
    case 13:
        Intent intent13 = new Intent(Maruzalar.this, MainActivity.class);
        startActivity(intent13);
        break;

    }

}

return true;
}
}

package com.example.newbook;

import android.support.v7.app.AppCompatActivity;

```

```

import android.os.Bundle;

import com.github.barteksc.pdfviewer.PDFView;

public class MeyoriyHujjatlar extends AppCompatActivity {
    private int first;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_meyoriy_hujjatlar);
        PDFView pdfView=findViewById(R.id.pdfVlew);
        Bundle extras = getIntent().getExtras();
        if (extras!=null){
            first = extras.getInt("titles");
        }

        switch (first)
        {
            case 0: pdfView.fromAsset("fandastur.pdf").load(); break;
            case 1: pdfView.fromAsset("iod.pdf").load(); break;
            case 2: pdfView.fromAsset("kalendar.pdf").load(); break;
        }
    }
}

package com.example.newbook;

import android.content.Context;
import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;

```

```

import android.support.v7.widget.LinearLayoutManager;
import android.support.v7.widget.RecyclerView;
import android.support.v7.widget.Toolbar;
import android.view.MenuItem;
import android.view.View;
import android.widget.ImageView;

import com.example.newbook.Adapter.TitleAdapterAmaliyot;
import com.example.newbook.forClick.CustimItemClickistener;
import com.example.newbook.items.Book;
import com.example.newbook.nomlar.MaruzaNomlari;

import java.util.ArrayList;
import java.util.List;

public class Amaliyot extends AppCompatActivity {
    private Context mContext;

    ArrayList<String> titleArrayList;
    private RecyclerView mRecyclerView;
    private ImageView img;
    List<Book> lstBook;
    public      int      first=0,      a=0,b=0,c=0,d=0,e=0,f=0,g=0,h=0,
i=0,j=0,k=0,l=0,m=0,n=0,o=0;
    public      String
ism,ism2,ism3,ism4,ism5,ism6,ism7,ism8,ism9,ism10,ism11,ism12,ism13,ism14
;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

```

```

setContentView(R.layout.activity_amaliyot);

img = (ImageView)findViewById(R.id.title_img2);

Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
setSupportActionBar(toolbar);

if (getSupportActionBar() != null){
    getSupportActionBar().setDisplayHomeAsUpEnabled(true);
    getSupportActionBar().setDisplayShowHomeEnabled(true);
    toolbar.setTitle("Amaliyotlar");
}

Intent intent=getIntent();
ism=intent.getStringExtra("num");
ism2=intent.getStringExtra("num1");
ism3=intent.getStringExtra("num2");
ism4=intent.getStringExtra("num3");
ism5=intent.getStringExtra("num4");
ism6=intent.getStringExtra("num5");
ism7=intent.getStringExtra("num6");
ism8=intent.getStringExtra("num7");
ism9=intent.getStringExtra("num8");

mContext = Amaliyot.this;
titleArrayList = new ArrayList<String>();
titleArrayList.add(MaruzaNomlari.Amaliyot_1);
titleArrayList.add(MaruzaNomlari.Amaliyot_2);
titleArrayList.add(MaruzaNomlari.Amaliyot_3);
titleArrayList.add(MaruzaNomlari.Amaliyot_4);
titleArrayList.add(MaruzaNomlari.Amaliyot_5);

```

```

titleArrayList.add(MaruzaNomlari.Amaliyot_6);
titleArrayList.add(MaruzaNomlari.Amaliyot_7);
titleArrayList.add(MaruzaNomlari.Amaliyot_8);
titleArrayList.add(MaruzaNomlari.Amaliyot_9);
titleArrayList.add(MaruzaNomlari.Amaliyot_10);

```

```

lstBook=new ArrayList<>();

```

```

        lstBook.add(0,new Book("1-amaliyot",
R.drawable.ic_action_lobaratoriya));
        lstBook.add(1,new Book("2-amaliyot",
R.drawable.ic_action_lobaratoriya));
        lstBook.add(2,new Book("3-amaliyot",
R.drawable.ic_action_lobaratoriya));
        lstBook.add(3,new Book("4-amaliyot",
R.drawable.ic_action_lobaratoriya));
        lstBook.add(4,new Book("5-amaliyot",
R.drawable.ic_action_lobaratoriya));
        lstBook.add(5,new Book("6-amaliyot",
R.drawable.ic_action_lobaratoriya));
        lstBook.add(6,new Book("7-amaliyot",
R.drawable.ic_action_lobaratoriya));
        lstBook.add(7,new Book("8-amaliyot",
R.drawable.ic_action_lobaratoriya));
        lstBook.add(8,new Book("9-amaliyot",
R.drawable.ic_action_lobaratoriya));
        lstBook.add(9,new Book("10-amaliyot",
R.drawable.ic_action_lobaratoriya));
        mRecyclerView=findViewById(R.id.title_recyclerviewAM);
        mRecyclerView.hasFixedSize();

```

```

        mRecyclerView.setLayoutManager(new
LinearLayoutManager(getApplicationContext()));
        TitleAdapterAmaliyot adapter = new TitleAdapterAmaliyot(mContext,
lstBook, new CustimItemClickListener() {
            @Override
            public void onItemClick(View v, int position) {

                Intent intent=new Intent(Amaliyot.this, Amaliyotlar.class);
                intent.putExtra("titles", (position));
                intent.putExtra("title", titleArrayList.get(position));
                startActivity(intent);
            }
        });
        mRecyclerView.setAdapter(adapter);
    }
    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        if (item.getItemId()== android.R.id.home){
            Intent intent=new Intent(Amaliyot.this, MainActivity.class);
            startActivity(intent);
        }
        return true;
    }
}

```