

UMUMKASBIY VA IXTISOSLIK FANLARNI INTEGRATSIYALASH UCHUN VEB-ILOVALARDAN FOYDALANISH

N.T.Orinbetov – *pedagogika fanlari boʻyicha falsafa doktori, dotsent*

Ajiniyoz nomidagi Nukus davlat pedagogika instituti

Tayanch soʻzlar: veb-frameworka, veb-ilovalar, raqamli texnologiyalar, Flask Django va FastAPI veb-frameworkalar, Flask mikrofreymvorkom, saytlararo skriptlar, shifrlashdan va seansni boshqarish, HTML, CSS va JavaScript-dan foydalanish, modulli test, integratsiya testi, optimallashtirish usullari, adaptiv veb-sahifa.

Ключевые слова: веб-фреймворк, веб-приложения, цифровые технологии, Flask Django и fastapi веб-фреймворки, Flask microframework, межсайтовые скрипты, шифрование и управление сессиями, использование HTML, CSS и JavaScript, модульное тестирование, интеграционное тестирование, методы оптимизации, адаптивная веб-страница.

Key words: web framework, web applications, digital technologies, Flask Django and FastAPI web frameworks, Flask microframework, cross-site scripts, encryption and Session Management, use HTML, CSS and JavaScript, modular testing, integration testing, optimization methods, adaptive web page.

Zamonaviy axborot jamiyatida umumkasbiy va ixtisoslik fanlarni rivojlantirish va integratsiyalashuvida veb-ilovalardan foydalanish asosiy oʻrin tutadi. Bu jarayon jadallik bilan kengayib borayotgan bilim sohalar va doimo rivojlanib borayotgan texnologiyalar oʻrtasidagi sinergiya natijasidir. Veb-ilovalardan foydalanish asosida fanlar integratsiyasiga yondashuvlar keyingi oʻn yilliklar davomida sezilarli oʻzgarishlarga uchradi. Bunday tizimlarning rivojlanishiga axborot texnologiyalari sohasidagi ilmiy tadqiqotlar ham turli sohalaridagi murakkab muammolarni hal qilish uchun bilimlarni integratsiyalash zarurati ham sabab boʻladi. Ushbu tadqiqotda umumkasbiy va ixtisoslik fanlarni integratsiyalash uchun veb-ilovalardan foydalanish, bu jarayon oldida turgan hozirgi tendentsiyalar va muammolarni hisobga olgan holda tahlil qilinadi.

Texnologiya aql bovar qilmaydigan tezlikda rivojlanayotgan veb-ishlab chiqarishning bugungi dunyosida toʻgʻri vositalar va ramkalarini tanlash har qanday loyihaning muvaffaqiyatida hal qiluvchi rol oʻynaydi. Python oʻzining koʻp qirraliligi tufayli dunyodagi eng mashhur dasturlash tillaridan biriga aylandi. Uning soddaligi va kodning oʻqilishi uni yangi boshlanuvchi dasturchilar uchun qulay qiladi, kuchli kutubxonalar va freymvork esa tajribali ishlab chiquvchilarga murakkab tizimlarni yaratish uchun barcha zarur vositalarni taqdim etadi. Flask, oʻz navbatida, Python-da veb-ishlab chiqish uchun mikroframework boʻlib, u sizga loyiha tuzilmasida minimal aralashuv bilan veb-ilovalarni yaratishga imkon beradi, shu bilan birga ishlab chiquvchilarga asboblari va komponentlarni tanlashda moslashuvchanlikni beradi.

Flaskning asosiy afzalligi uning engilligi va soddaligidir. Kattaroq va murakkab veb-freymvorkdan farqli oʻlaroq, Flask qatʼiy dastur arxitekturasini yuklamaydi va loyihani keraksiz funktsionallik bilan ortiqcha yuklamaydi. Bu ishlab chiquvchilarga veb-ilovalarni bosqichma-bosqich yaratishga, loyiha oʻsishi va rivojlanishiga zarur komponentlarni qoʻshish imkonini beradi. Ushbu yondashuv nafaqat ishlab chiqish jarayonini soddalashtiradi, balki yuqori darajadagi modullik va kodni qayta ishlatish imkonini beradi.

Bizning veb-ilovani ishlab chiqish uchun Python va Flask-ni tanlashga ularning bir qator asosiy loyiha talablari javob bera olish qobiliyati sabab boʻldi:

- biz oʻzgaruvchan talablar va ish sharoitlariga moslasha oladigan moslashuvchan va kengaytiriladigan dastur yaratishga intilamiz;

- foydalanuvchilarga sifatli xizmat koʻrsatish uchun tizimning yuqori ishlashi va tezligini taʼminlash kerak;

- dastur xavfsizligi muhim jihat, jumladan maʼlumotlarni himoya qilish va zaifliklarning oldini olish.

Python va Flask ishlab chiquvchilarni ushbu talablarni amalga oshirish uchun kuchli vositalar va kutubxonalar bilan taʼminlaydi, masalan, maʼlumotlar bazalari bilan ishlash uchun SQLAlchemy, shablonlarni yaratish uchun Jinja2 va soʻrovlar va javoblarni qayta ishlash uchun Werkzeug.

Texnik jihatlardan tashqari, ushbu texnologiyalarni tanlash katta ishlab chiquvchilar hamjamiyati va keng qamrovli hujjatlar tomonidan qoʻllab-quvvatlanadi, bu esa

ishlab chiqish jarayonini osonlashtiradi va muammolar paydo boʻlganda yechimlarni topadi.

Veb-ilovalarni yaratish uchun Flask-dan foydalanishning afzalliklari va kamchiliklarini tahlil qilishdir. Flask Django va FastAPI kabi boshqa veb-frameworkalar bilan ishlab chiqish qulayligi, unumdorligi, kengaytirilishi va hamjamiyatni qoʻllab-quvvatlash nuqtai nazaridan taqqoslanadi. Bundan tashqari, mobil ilovani maʼlumotlar bazalari, keshlash tizimlari va bulutli platformalar kabi zamonaviy texnologiyalar bilan integratsiyalashuviga alohida eʼtibor qaratiladi.

Flask-da umumkasbiy va ixtisoslik fanlar integratsiyasining veb-ilovasini ishlab chiqish, vositalarni tanlash va dasturni loyihalash va amalga oshirish tafsilotlarini yanada chuqurroq oʻrganamiz.

Avvalo, asboblari va texnologiyalarning toʻgʻri toʻplamini tanlash har qanday dasturiy mahsulotni ishlab chiqishda muhim qadamdir. Flask-da veb-ilovalarni ishlab chiqish uchun ushbu tanlov nafaqat veb-frameworkaning oʻzini, balki dasturning funktsionalligini taʼminlaydigan va ish faoliyatini yaxshilaydigan qoʻshimcha kutubxonalar va xizmatlarni ham oʻz ichiga oladi. Ushbu bosqichda ishlab chiquvchilar maʼlumotlar bazasi, versiyalarni boshqarish tizimi, sinov va disk raskadrovka vositalari va dasturni joylashtirish platformasi haqida qaror qabul qilishlari kerak.

Rivojlanish muhitini oʻrnatish Python va Flaskni oʻrnatishdan boshlanadi. Loyiha bogʻliqliklarini global Python muhitidan ajratish uchun virtual muhitdan (virtualenv) foydalanish tavsiya etiladi, bu kutubxonalar oʻrtasidagi ziddiyatlarni oldini oladi va qaramlikni boshqarishni soddalashtiradi. Flask va kerakli kutubxonalarini oʻrnatish pip paket menejeri yordamida amalga oshiriladi. Maʼlumotlar bazasi bilan ishlash uchun Flask kengaytmalaridan foydalanish mumkin, masalan, SQLAlchemy bilan integratsiyalashuv uchun Flask-SQLAlchemy relyatsion maʼlumotlar bazalari bilan ishlash uchun qulay interfeysni taʼminlaydi yoki MongoDB kabi NoSQL maʼlumotlar bazalari bilan ishlash uchun Flask-MongoEngine. Texnologiya toʻplamini tanlagandan soʻng, keyingi qadam dastur arxitekturasini loyihalashdir. Rivojlanishga arxitektura yondashuvi asosan yakuniy mahsulotning moslashuvchanligi, kengaytirilishi va qoʻllab-quvvatlash qulayligini belgilaydi. Flask mikrofreymvorkom boʻlib, qatʼiy meʼmoriy cheklolvar qoʻymaydi, bu ishlab chiquvchilarga meʼmoriy uslubni tanlash erkinligini beradi. Biroq, bu aniq va samarali arxitekturani yaratish masʼuliyati ishlab chiquvchilarga tegishli ekanligini anglatadi. Ilovani mustaqil modullarga ajratishni oʻz ichiga olgan modulli yondashuv, ularning har biri funktsionallikning oʻziga xos qismi uchun javob beradi, ishlab chiqish va sinovdan oʻtkazishni soddalashtiradi, shuningdek, kelajakda dasturni kengaytirishni osonlashtiradi.

Xavfsizlik masalalari mobil ilovalarni ishlab chiqishda markaziy oʻrinlardan birini egallaydi. Flask dastur xavfsizligini taʼminlash uchun bir nechta mexanizmlarni taqdim etadi, jumladan saytlararo soʻrovlarni soxtalashtirishdan (CSRF) va saytlararo skriptlardan (XSS), maʼlumotlarni shifrlashdan va seansni boshqarishdan himoya qilish. Shaxsiy maʼlumotlarning himoyasini taʼminlash va dastur

funktsiyalariga ruqsatsiz kirishni oldini olish uchun foydalanuvchi autentifikatsiyasi va avtorizatsiya tizimini diqqat bilan ko'rib chiqish muhimdir.

Foydalanuvchi interfeysi (UI) ishlab chiqish mobil ilovalarni yaratishda asosiy rol o'ynaydi, chunki foydalanuvchi dastur bilan o'zaro aloqada bo'ladi. Flask kontekstida UINI ishlab chiqishda mobil qurilma ekranining turli o'lchamlariga moslashuvchi sezgir va intuitiv dizaynni yaratishga e'tibor qaratish lozim. HTML, CSS va JavaScript-dan foydalanish veb-interfeyslarni ishlab chiqishda standart yondashuv bo'lib, Flask Jinja2 shablon tizimi orqali ushbu texnologiyalarni server tomonidagi mantiq bilan integratsiyalashtirishni osonlashtiradi. Jinja2 shablonlari ishlab chiquvchilarga serverda olingan ma'lumotlardan foydalanishda dinamik HTML kontentini samarali yaratish imkonini beradi, bu esa sahifani qayta yuklamasdan interaktiv foydalanuvchi interfeyslarini yaratish imkonini beradi.

Mobil ilovalarni ishlab chiqishning muhim qismi ham ularning ishlashini ta'minlashdir. Bunga erishish uchun resurslarni minimallashtirish va siqish (CSS va JavaScript fayllari), resurslarni asinxron yuklash va keshlash kabi optimallashtirishning turli usullaridan foydalanish mumkin. Flask Flask-Assets kabi kengaytmalardan foydalanishni qo'llab-quvvatlaydi, ular statik fayllarni optimallashtirish jarayonini avtomatlashtiradi va shu bilan ilovalarni yuklash tezligini va umumiy foydalanuvchi tajribasini yaxshilaydi.

Server tomonidagi mantiqqa o'tadigan bo'lsak, Flask ishlab chiquvchilarga mijoz so'rovlarini bajarish uchun moslashuvchan marshrutizatsiya tizimini taklif qiladi. Marshrutlarni va ularga mos keladigan ishlov beruvchi funktsiyalarini aniqlash ishlab chiquvchilarga mijoz va server o'rtasidagi ma'lumotlar oqimini osongina boshqarish imkonini beradi, bu dasturda foydalanuvchining turli harakatlarini boshqarish mantiqini ta'minlaydi. Ma'lumotlar bazalari va boshqa tashqi xizmatlar bilan integratsiya Flask modullari va kengaytmalari orqali amalga oshiriladi, bu boy funktsionallikka ega murakkab veb-ilovalarni yaratishni osonlashtiradi.

Server mantig'ini amalga oshirish jarayonida dastur xavfsizligi muhim jihat bo'lib qolmoqda. Flask SQL in'ektsiyalari va CSRF hujumlari kabi keng tarqalgan tahdidlardan himoya qilish uchun o'rnatilgan mexanizmlarni taqdim etadi va Flask-Security kabi qo'shimcha kengaytmalardan foydalanish autentifikatsiya, avtorizatsiya, va foydalanuvchi sessiyasini boshqarish.

Flask yordamida mobil ilovani ishlab chiqishda test funktsionallikni rivojlantirishdan kam emas. Modulli testi, integratsiya testi va foydalanuvchi interfeysi testi ishlab chiquvchilarga xatolarni erta bosqichda aniqlash va tuzatish imkonini beradi, bu esa yuqori sifatli yakuniy mahsulotni ta'minlaydi. Flask PyTest va Flask-Testing kabi mashhur test tizimlari bilan integratsiyani qo'llab-quvvatlaydi, bu testlarni yaratish va ishga tushirish jarayonini soddalashtiradi [1].

Biz Flask-dagi veb-ilovani foydalanuvchi interfeysi va server mantig'ini ishlab chiqishning asosiy jihatlarini, jumladan, ishlashni optimallashtirish, xavfsizlikni ta'minlash va sinovdan o'tkazishning ahamiyatini ko'rib chiqdik.

Ilovaning jonli efirga chiqishidan oldin uning barqarorligi va ishonchligini ta'minlashda PyTest yoki Flask-Testing kabi avtomatlashtirilgan test vositalaridan foydalanish ushbu jarayonni sezilarli darajada soddalashtirishi mumkin, bu esa ishlab chiquvchilarga alohida dastur komponentlarining ishlashini, shuningdek, ular o'rtasidagi o'zaro ta'sirlarni samarali sinab ko'rish imkonini beradi.

Sinov bosqichida foydalanuvchi interfeysining barcha elementlarining funktsionalligini, shuningdek, server tomonida amalga oshirilgan funktsionallikni tekshirishga alohida e'tibor beriladi. Modulli testlari ilovani individual funktsiyalari va usullarining ishlashini ajratish va sinab

ko'rish imkonini beradi, integratsiya testlari esa dasturning turli modullari va qatlamlari o'rtasidagi o'zaro ta'sirni baholaydi, ularning to'g'ri integratsiyasini ta'minlaydi. Ko'pincha Selenium kabi veb-UI sinovini avtomatlashtirish vositalari bilan amalga oshiriladigan UI testi ilova turli brauzerlar va qurilmalarda to'g'ri ko'rsatilishi va ishlashini ta'minlashga yordam beradi.

Funktsional testdan tashqari, ishlash va xavfsizlik testlari muhim rol o'ynaydi. Ishlash testi ilovangizdagi to'siqlarni aniqlashga yordam beradi, bu esa uni sekinlashtirishi yoki serverlar yukini oshirishi mumkin. Load-Runner yoki JMeter kabi vositalardan foydalanish ko'p sonli foydalanuvchilar tomonidan bir vaqtda kirishni taqlid qilish imkonini beradi va shu bilan ilovani yuqori yuk sharoitida yuqori unumdorlik va barqarorlikni saqlab qolish qobiliyatini sinab ko'radi. Xavfsizlik testi tajovuzkorlar tomonidan ilova ma'lumotlari yoki funktsiyalariga ruqsatsiz kirish uchun foydalanishi mumkin bo'lgan potentsial dastur zaifliklarini aniqlashga qaratilgan. OWASP ZAP yoki Burp Suite kabi vositalar ishlab chiquvchilarga dastur zaifliklari va xavfsizlik kamchiliklarini har tomonlama tekshirish imkoniyatini beradi.

Mobil ilovani ishlab chiqishning yakuniy bosqichi uni joylashtirish va qo'llab-quvvatlashdir. Joylashtirish mos hosting platformasini tanlash, server infratuzilmasini o'rnatish va ilovani nashr etishni o'z ichiga oladi. Ilovaning unumdorligi va kengaytirilishi talablariga qarab, ishlab chiquvchilar an'anaviy veb-xosting, Amazon Web Services (AWS), Google Cloud Platform (GCP) yoki Microsoft Azure kabi bulutli xizmatlar yoki xizmat sifatidagi platformalar (PaaS) o'rtasida tanlov qilishlari mumkin. Heroku yoki DigitalOcean App Platform kabi ushbu variantlardan har biri o'zining afzalliklari va kamchiliklarini, jumladan, narxdagi farqlarni, boshqaruvning qulayligi va kengaytirilishini taklif qiladi.

Ilovani ishga tushirgandan so'ng, qo'llab-quvvatlash va yangilash bosqichi boshlanadi. Ilovani muntazam yangilash, aniqlangan xatolarni tuzatish, uning funktsionalligi va xavfsizligini yaxshilash muhimdir. Foydalanuvchilarning fikr-mulohazalarini olish ushbu jarayonning asosiy jihat hisoblanadi, chunki u ilovadagi kamchiliklarni aniqlash va keyingi rivojlanish yo'nalishlarini aniqlash imkonini beradi. Jira yoki GitHub Issues kabi loyihalarni boshqarish va muammolarni kuzatish tizimlaridan foydalanish loyiha ishingizni tashkil qilishni osonlashtiradi, bu sizga vazifalarni samarali rejalashtirish, ularning bajarilishini kuzatish va ishlab chiqish guruhingiz bilan muloqot qilish imkonini beradi.

Ushbu sahifada biz Flask mobil ilovasini sinovdan o'tkazish, joylashtirish va texnik xizmat ko'rsatishning asosiy jihatlarini ko'rib chiqdik. Ushbu bosqichlar ilovalarni ishlab chiqish hayotiy tsiklining ajralmas qismi bo'lib, uning sifati, ishonchligi va foydalanuvchi kutganlarini qondirishni ta'minlaydi.

Flaskda mobil ilovani ishlab chiqishning navbatdagi muhim jihatiga o'tsak, biz ilovani optimallashtirish strategiyalariga e'tibor qaratamiz.

Ilovani optimallashtirish uning ishlashi, tezligi va ilovani umumiy foydalanuvchi tajribasiga ta'sir qiluvchi asosiy omil hisoblanadi. Flask yordamida ishlab chiqilgan mobil ilova kontekstida optimallashtirish sahifani yuklash vaqtini yaxshilash, server samaradorligini oshirish va foydalanuvchining ilova bilan o'zaro aloqasidagi kechikishlarni minimallashtirishga qaratilgan bir qator harakatlarni o'z ichiga olishi mumkin [2].

Adaptiv veb-sahifa dizayni optimallashtirishda ham muhim rol o'ynaydi, bu dasturni turli qurilmalar va ekran o'lchamlarida to'g'ri ko'rsatishi va ishlashiga imkon beradi. Bootstrap yoki Foundation kabi CSS freymvorkdan foydalanish sezgir dizaynni ancha osonlashtirishi mumkin.

Ilovani optimallashtirishning muhim jihatlaridan biri ma'lumotlar bazasidan samarali foydalanishdir. Bunga

ma'lumotlar bazasi sxemasini to'g'ri loyihalash, ma'lumotlarni qidirishni tezlashtirish uchun indekslardan foydalanish va so'rovlarni bajarish vaqtini minimalashtirish uchun so'rovlarni optimallashtirish kiradi. Ishlab chiquvchilar qiyinchiliklarni aniqlash va hal qilish uchun profillash vositalaridan foydalangan holda ma'lumotlar bazasi ishlashini muntazam ravishda tahlil qilishlari kerak. Tez-tez foydalaniladigan ma'lumotlarni keshlash ma'lumotlar bazasi qo'ng'iroqlari sonini kamaytirish orqali dastur tezligini sezilarli darajada oshirishi mumkin.

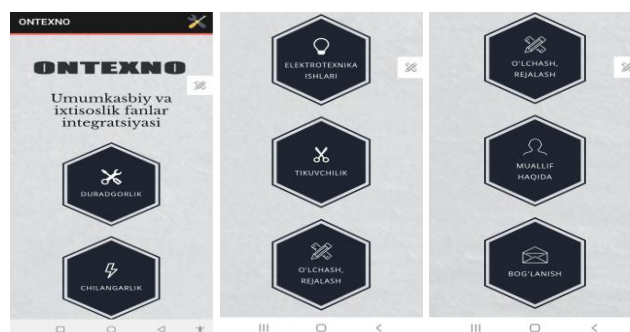
Sahifani yuklash vaqti foydalanuvchi tajribasi uchun muhim omil, ayniqsa mobil veb-ilovalarda. Uni kamaytirish uchun siz statik fayllarni (CSS va JavaScript) siqish va birlashtirish, resurslarni asinxron yuklash va tasvirlarni optimallashtirish usullaridan foydalanishingiz mumkin. WebP kabi zamonaviy tasvir formatlaridan foydalanish sifatini saqlab qolgan holda tasvir hajmini sezilarli darajada kamaytirishi va sahifalarning tezroq yuklanishini ta'minlaydi. Bundan tashqari, tasvirlar va videolar uchun Lazy Loading texnologiyasidan foydalanish ushbu elementlarning yuklanishini foydalanuvchiga ko'ringuncha kechiktirishga imkon beradi va shu bilan sahifaning dastlabki yuklanishini tezlashtiradi.

Foydalanuvchi tomonidan ishlashni yaxshilash uchun ishlab chiquvchilar DOMni samarali yangilashni va komponentga asoslangan yondashuvdan foydalangan holda yaxshilangan foydalanuvchi tajribasini ta'minlaydigan React yoki Vue.js kabi zamonaviy JavaScript freymvorki va kutubxonalaridan foydalanishlari mumkin. Qimmatbaho operatsiyalar va xotira oqishiga yo'l qo'ymaslik uchun JavaScript kodining ishlashiga ham e'tibor berish kerak.

Ilovani turli qurilmalarda to'g'ri ko'rsatish va ulardan foydalanish qulayligini ta'minlash uchun sezgir dizayn zarur. Adaptiv tartiblar, media so'rovlar va sezgir tasvirlardan foydalanish ilova interfeysini ekran o'lchamlari va qurilma yo'nalishiga moslashtirish orqali ushbu

maqsadga erishishga yordam beradi. Bu, ayniqsa, mobil ilovalar uchun juda muhimdir, bu erda foydalanuvchilar turli xil ekran o'lchamlariga ega bo'lgan keng turdagi qurilmalardan foydalanishlari mumkin.

Muntazam ishlash testlari optimallashtirish jarayonining ajralmas qismidir. Google PageSpeed Insights, Lighthouse va WebPageTest kabi vositalardan foydalanish sahifani yuklash tezligini, manba samaradorligini va ilovalar mavjudligini baholashga yordam beradi. Ushbu vositalar ishlab chiquvchilar o'z ilovalarini optimallashtirish uchun foydalanishi mumkin bo'lgan maxsus ishlash tavsiyalarini beradi [3].



1-rasm. ONTEXNO. Umumkasbiy va ixtisoslik fanlar integratsiyasining mobil ilovasi

Xulosa qilib aytganda, veb-ilovalarni o'quv jarayoniga integratsiyalash ta'lim sifatini oshirish, o'quvchilarning motivatsiyasi va faolligini oshirish, tanqidiy fikrlash va muammolarni hal qilish ko'nikmalarini rivojlantirish uchun kuchli vositadir. Veb-ilovalar turli ehtiyoj va imtiyozlarga ega bo'lgan talabalar uchun ochiq bo'lgan yanada moslashuvchan va inklyuziv ta'lim muhitini yaratishga yordam beradi.

Adabiyotlar

1. Прохоронок Н.А. Python.Самое необходимое. – Санкт-Петербург: БХВ-Петербург, 2011, –С.416.
2. <https://python.org>
3. <https://ontexno.uz>

REZYUME. Ushbu maqolada umumkasbiy va ixtisoslik fanlar o'rtasidagi chuqur o'zaro aloqani ta'minlash maqsadida veb-ilovalarni o'quv jarayoniga eng samarali integratsiyalashda qo'llanilishi mumkin bo'lgan usullar va algoritmlarni tahlil qilish bo'yicha tavsiyalar berilgan. Bundan tashqari, mobil ilovani ma'lumotlar bazalari, keshlash tizimlari va bulutli platformalar kabi zamonaviy texnologiyalar bilan integratsiyalashuviga alohida e'tibor qaratiladi.

РЕЗЮМЕ. В данном тексте приведены рекомендации по анализу методов и алгоритмов, которые можно использовать для наиболее эффективной интеграции веб-приложений в учебный процесс с целью обеспечения глубокого взаимодействия общих и профильных дисциплин. Кроме того, особое внимание уделяется интеграции мобильного приложения с современными технологиями, такими как базы данных, системы кэширования и облачные платформы.

SUMMARY. This text provides recommendations for the analysis of methods and algorithms that can be used for the most effective integration of web applications into the educational process in order to ensure a deep interaction between general and specialized disciplines. In addition, special attention is paid to the integration of the mobile application with modern technologies such as databases, caching systems and cloud platforms.

XALQ PEDAGOGIKASI VATANPARVARLIK TUYG'ULARINI SHAKLLANTIRISH VOSITASI

T.A.Pazilov – *pedagogika fanlari bo'yicha falsafa doktori*

Ajiniyoz nomidagi Nukus davlat pedagogika instituti

Tayanch so'zlar: folklor, xalq og'zaki ijodi, adabiyot, ijod, urf-odat, bolalar fol'klori, madaniy yodgorlik, etnopedagogika, tarbiya, milliylik.

Ключевые слова: фольклор, устное народное творчество, творчество, традиции, фольклорные произведения, памятник культуры, этнопедагогика, воспитание, национальность.

Key words: folklore, oral folk art, creativity, traditions, folklore works, cultural monument, ethnopedagogy, education, nationality.

Bolalar fol'klori bolalar uchun dastlabki turmush darsligi, ma'naviy boyligidir. Quyida ushbu bolalar fol'klori janrlarning ayrimlarining farqli xususiyatlari va ularning tarbiyaviy roli haqida qisqacha to'xtalib o'tishni ma'qul deb topdik.

Qoraqalpoq xalq pedagogikasidan o'rin olgan alla(besik jiri)lari ham juda erta zamonlardan vujudga kelganligi ma'lum. U asta, nozik, halqqa dastur bo'lgan kuy ohangida bajariladi. Allalarni aytishga maxsus ijrochi talab etilmay, beshik tebratuvchi ona, buvi yoki boshqa allalovchi odam bajaraveradi. Hammaga ma'lum nozik ohangli, tayyor satrlar xalq tomonidan asrlar davomida yaratilib kelingan

va ayni vaqtda ham uning yangi variantlari yaratilmoqda. Allaning bola tarbiyasida juda katta ahamiyatga ega ekanligi Orta Osiyoning mashhur olimi buyuk alloma Abu Ali Ibn Sino [1:368] tomonidan ham yuqori baholangan.

Ona allasi haqida fol'klorshunos olim A Tajimuratov "Qoraqalpoq halqining ta'lim tarbiyaviy dasturi alladan boshlanadi. Bu dastur butun O'rta Osiyo hattoki, dunyo xalqlarining bola tarbiyalash ishlarida asosiy o'rinni egallaydi..." [2:6] deb beshik kuylarini inson tarbiyasining asosi sifatida ko'rsatadi. O'zbekistonlik olim G.A.Mardonova o'zining ilmiy-tadqiqot ishini oilada bolalarni alla vositasida ma'naviy-axloqiy tarbiyalashning