

O'ZBEKISTON RESPUBLIKASI OLIY VA O'RTA MAXSUS TA'LIM
VAZIRLIGI

MIRZO ULUG'BEK NOMIDAGI O'ZBEKISTON MILLIY UNIVERSITETI

MEXANIKA-MATEMATIKA FAKULTETI
PROGRAMMALASH VA TARMOQ TEXNOLOGIYALARI
KAFEDRASI

“Web texnologiyalari yordamida kafedra faoliyatini avtomatlashtirish tizimini
yaratish” mavzusidagi

BITIRUV MALAKAVIY ISH

Bajardi: Amaliy matematika va informatika ta'lim
yo'nalishi bitiruvchisi 4-kurs talabasi Hayitov Z.B.

Ilmiy rahbar: o'qituvchi Mirzayev A.I.

Bitiruv malakaviy ishi kafedradan daslabki himoyadan o'tdi.

17-sonli bayonnomasi 22-may 2014-yil

Toshkent-2014

Mundarija:

Kirish	3
I BOB. Masalaning qo'yilishi va yechilish mexanizmi.....	8
1.1. Masalaning qo'yilishi	8
1.2. Masalani yechish mexanizmi.....	9
II BOB. Berilganlar bazasini tashkil qilish va uni boshqarish tizimi	21
2.1. Berilganlar bazasi tashkil qilish.....	21
2.2. Berilganlar bazasi tashkil qilishdagi asosiy talablar	21
2.3. Tarmoqda ma'lumotlar bazalari arxitekturasini	23
III BOB. Avtomatlashgan kafedra tizimining ishlash tamoili	27
3.1. Berilganlar bazasidagi jadvallar sxemasi	27
3.2. Tizimning tavsifi	32
3.3. Tizimning qulayligi va avzalligi.....	33
Xulosa.....	39
Fodalanilgan adabiyotlar	41
Ilova.....	42

Kirish

Yaqinda O‘zbekiston Respublikasi Oliy Majlisi Qonunchilik palatasining Axborot va kommunikatsiya texnologiyalari masalalari qo‘mitasi O‘zbekiston Respublikasi Aloqa, axborotlashtirish va kommunikatsiya texnologiyalari davlat qo‘mitasi bilan hamkorlikda «Dasturiy mahsulotlar bozorini rivojlantirishga doir qonunchilik bazasini takomillashtirishning ustuvor yo‘nalishlari» mavzusida davra suhbatini tashkil etdi [1].

Unda O‘zbekiston Respublikasi Oliy Majlisi Qonunchilik palatasining deputatlari, Vazirlar Mahkamasi, manfaatdor vazirlik va idoralar mas'ul xodimlari, dasturiy mahsulotlar ishlab chiqaruvchi kompaniyalar vakillari, axborot texnologilari sohasining mutaxassislari, dasturchilar, ommaviy axborot vositalari va fuqarolik jamiyati institutlarining vakillari ishtirok etdilar.

Davra suhbatidan ko‘zlangan maqsad-“Axborotlashtirish to‘g‘risida”gi O‘zbekiston Respublikasi Qonunida belgilangan dasturiy mahsulotlar ishlab chiqarishni rag‘batlantirish va rivojlantirish bo‘yicha normalar ijrosini tahlil qilish, axborot texnologiyalari bozorining muhim qismi bo‘lgan dasturiy ta'minot industriyasini davlat tomonidan ko‘llab-kuvvatlashning optimal yo‘llarini muhokama qilish va mamlakatimiz dasturiy mahsulotlar industriyasini rivojlantirishning qonunchilik asoslarini takomillashtirish yuzasidan aniq taklif va tavsiyalar ishlab chiqishdan iborat edi.

O‘zbekiston Respublikasi Prezidentining mamlakatimizda yaqin istiqbolda axborotlashtirish jarayonlarini rivojlantirish borasida aniq maqsadli yo‘nalishlarni belgilab bergan «Zamonaviy axborot-kommunikatsiya texnologiyalarini yanada keng joriy etish va rivojlantirish chora-tadbirlari to‘g‘risida»gi 2012 yil 21 martda qabul qilgan qaroridan kelib chiqsak, ushbu tadbirning dolzarbligi yanada ayon bo‘ladi.

Darhaqiqat, ayni kunda dasturiy mahsulotlar ishlab chiqarish butun jahonda alohida ahamiyat kasb etayotir. Zotan, dasturiy mahsulotlar axborot-kommunikatsiya sohasini rivojlantirishning asosi hisoblanadi, axborot xavfsizligini

ta'minlashga ko'maklashadi, milliy iqtisodiyotlar innovatsion rivojlanishining juda muhim sharti bo'lib, globallashtirish sharoitlarida mamlakatning raqobatbardoshligini ta'minlaydi.

Davra suhbatida ta'kidlanganidek, mustaqillik yillarida mamlakatimizda dasturiy ta'minot industriyasini rivojlantirishda sifat jihatdan yuksalishni amalga oshirish uchun qulay shart-sharoitlar yaratildi. Sohani rivojlantirishning mustahkam qonunchilik bazasi yaratildi. Xususan, shu davr ichida sohaga tegishli 10 dan ortiq Qonunlar, mamlakat Prezidenti Farmoni va Vazirlar Mahkamasining qator qarorlari qabul qilindi.

Keyingi yillarda Parlament tomonidan amaldagi bir qator qonunlarga zamonaviy axborot va kommunikatsiya texnologiyalarini va dasturiy mahsulotlar bozorini yanada rivojlantirishga qaratilgan qo'shimcha va o'zgartishlar kiritildi. "Axborotlashtirish to'g'risida"gi, "Telekommunikatsiyalar to'g'risida"gi, "EHM dasturlari va ma'lumotlar bazasini himoya qilish to'g'risida"gi Qonunlar shular jumlasidandir.

Ta'kidlash kerakki, qabul qilingan normativ-xukuriy hujjatlarga muvofiq, dasturiy mahsulot ishlab chiqaruvchi korxonalarga soliq va bojxona sohasida katta imtiyozlar berildi. Chunonchi, bir necha yillar davomida kompyuter va axborot texnologiyalari bo'yicha o'qitish xizmatlarini, shuningdek, dasturiy vositalarni realizatsiya qilishni qo'shimcha qiymat solig'idan ozod etish bo'yicha normalari amal qildi. Bundan tashqari, dasturiy mahsulotlar ishlab chiqarishga ixtisoslashgan korxonalar mikrofirmalar va kichik korxonalar uchun umumiy belgilangan yagona soliq to'lovi stavkasidan kamaytirilgan soliq to'laydilar.

Muhokamada ta'kidlanganidek, hozirgi kunda davlat hokimiyati va boshqaruvi organlari dasturiy mahsulotlarning faol iste'molchisi hisoblanadi. Keyingi vaqtda mamlakatimizda vazirliklar, idoralar, xo'jalik birlashmalari va yirik korxonalarning xarajatlar smetasi va biznes rejalarida ishlab chiqarish jarayonlarini avtomatlashtirish, axborot tizimlari va resurslarini yaratish, dasturiy mahsulotlar hamda interfaol davlat xizmatlarini joriy etishga qaratilgan xarajatlarni nazarda tutish bo'yicha tegishli ishlar olib borilmoqda. Joriy yilda mamlakatimizda

davlat hokimiyati va boshqaruvi organlarining idoralararo integratsiyalashgan axborot tizimlarini joriy etish, ma'lumotlar uzatishning idoralararo tarmoqlarini yaratish, «Elektron hukumat» tizimini hamda Internet tarmog'ining milliy segmentini shakllantirish borasidagi loyihalarni ro'yobga chiqarish uchun davlat buyurtmasini shakllantirish va moliyalashtirish ishlari boshlab yuborildi.

Tabiiyki, ushbu chora-tadbirlar dasturiy mahsulotlar industriyasining milliy bozori ko'lamini kengaytirishga ijobiy ta'sir ko'rsatib, uning jadal rivojlanishini rag'batlantirmoqda. Natijada mahalliy dasturiy ta'minot ishlab chiquvchilari mahsulotlari hajmi yil sayin kengayib, xorijiy dasturiy mahsulotlar bilan raqobatlashishga qodir yangi dasturiy mahsulotlar ishlanmalari yaratilmoqda.

Davra suhbatida axborot texnologiyalari va dasturiy ta'minot bozorini rivojlantirishga qaratilgan qonun ijodkorligi ishlarini ustuvor yo'nalishlari bilan bir qatorda amaldagi qonun hujjatlari ijrosining borishi, dasturiy mahsulotlar ishlab chiqarishning bugungi holati va ularni rivojlantirish istiqbollari ham muhokama qilindi. O'zbekiston Respublikasi Qonunchilik palatasidagi siyosiy partiyalar fraksiyalarining deputatlari o'z partiyalarining ustav qoidalari va saylovoldi dasturlardan kelib chiqqan holda, davra suhbatida muhokama qilingan masalalar yuzasidan o'z fikr-mulohazalarini bildirdilar.

Suhbat yakunlariga ko'ra, dasturiy mahsulotlarni yaratish, tarqatish va joriy etish sohasidagi munosabatlarni tartibga solish, milliy dasturiy mahsulotlar industriyasini rivojlantirish va uning eksport salohiyatini oshirish maqsadida qulay iqtisodiy muhitni yaratish uchun qonunchilik bazasini takomillashtirishga qaratilgan bir qator amaliy taklif va tavsiyalar ishlab chiqildi.

Bugungi kunda tashkilot qanday bo'lishidan qat'iy nazar ular moliyaviy bo'ladimi, savdo tashkilotlari, sanoat bo'ladimi barchasining faoliyatini berilganlar bazasisiz tasavvur qilib bo'lmaydi. Agar berilganlar bazasi bo'lmaganida ular axborot oqimi ichida qolib ketar edilar.

Hozirgi davrda sohani kompyuter, aloqa, texnika-texnologiya, internet tarmog'isiz tasavvur qilib bo'lmaydi. Shuning uchun bu sohalar jahon bozorida o'z o'rnini topishi egallashi uchun texnika olami bo'lmish bu davrda internet

tarmo'gida o'z sahifalariga ega bo'lishlari kerak. Ayniqsa bugungi kunda rivojlangan mamlakatlar oliy ta'limida internet saytining o'zni beqiyos. Bugungi kunda O'zbekistonda ham ta'lim muassasalari o'zlarining internet saytlarini yaratishga harakat qilmoqdalar va yaratmoqdalar.

Mazkur o'rganilayotgan tadqiqot ishning dolzarbligi Respublikadagi barcha chakana savdo tizimlarini avtomatlashtirish va uni bevosita global tarmoqda ya'ni internet tarmog'iga ulash alohida ahamiyatga egadir. Buning ahamiyatli tomoni shundaki global tarmoq orqali mavjud bo'lgan chakana savdo bilan shug'ullanadigan do'konlar faoliyati yangiliklari va omborxonadagi bor mahsulotlar bilan bevosita tanishish mumkin.

O'zbekistonda axborotlashtirishni rivojlantirish, jamiyat taraqqiyotining barcha sohalarida zamonaviy axborot texnologiyalari, texnika va telekommunikatsiya vositalarini ommaviy ravishda joriy etish hamda ulardan foydalanish, fuqarolarning axborotga bo'lgan talab-ehtiyojlarini yanada to'liqroq qondirish, jahon axborot resurslaridan bahramand bo'lishni kengaytirish uchun qulay shart sharoitlarni yaratish aloqa va axborot kommunikatsiyasi sohasini rivojlantirish asosiy vazifalari hisoblanadi.

Bulardan tashqari fuqarolarning erkin tarzda o'zlariga qulay vaqtda internet tarmog'i orqali axborot olishlari va ma'lumotlarni erkin tarzda almashishlarini ta'minlash ham rivojlanishning asosiy vaziflari deyish mumkin.

Shuni aytish zarurki bugungi kunda axborot, axborot resurslari, kompyuter tarmoqlari va turli korporativ tizimlarga nisbatan tahdidlarni oldini olish asosi dolzarb masalalardan biri hisoblanib kelmoqda. Chunki, har qanday korxonaning samarali faoliyat ko'rsatishida faqatgina moddiy, tabiiy, moliyaviy resurslarning o'zi yetarli emas. Ulardan qanday qilib samarali, oqilona foydalanish uchun o'sha sohadagi texnologiyalar haqida yetarlicha axborot bo'lishi talab etiladi. Mavjud bo'lgan axborot esa qolgan barcha resurslardan samarali foydalanish uchun keng imkoniyatlar yaratib beradigan yagona resurs hisoblanadi. Axborotlar saqlanayotgan berilganlar bazasidagi ma'lumotlar maxfiy bo'lishi talab etiladi. Maxfiylik deganda alohida shaxslar yoki tashkilotlarning boshqa shaxslar yoki

tashkilotlarga qachon, qancha miqdorda ma'lumotlar berilgani umumiy qilib aytganda qachon va qancha miqdorda ma'lumotlar almashinganligini aniqlash huquqi tushiniladi. Saqlanayotgan axborot undan foydalanilayotgan tashkilot uchun muhim hisoblanadi. U ya'ni axborot o'g'irlanmasligi va yo'qotilmasligi kerak. Shuning uchun ham deyarli barcha xorij mamlakatlarida, balki respublikamizda ham axborot tizimlarning muhofazasiga jiddiy e'tibor berilmoqda.

I BOB. Masalaning qo'yilishi va yechilish mexanizmi

1.1. Masalaning qo'yilishi

Web-texnologiyalari jadal suratda rivojlanishi, hamda tarmoq texnologiyalari rivojlanishi axborot tizimlarini shu texnologiyalardan foydalanib yaratish ommaviylashib ketgan. Qulaylik tomonlaridan biri tizimni biror serverga o'rnatib shu serverni tarmoqqa ulansa, foydalanuvchi ixtiyoriy kompyuter yoki mobil apparatlardan tizimga kirishi mumkin.

Ushbu bitiruv malakaviy ishining asosiy maqsadi: Web texnologiyalari yordamida kafedra faoliyatini avtomatlashtirish axborot tizimini yaratish.

Bitiruv malakaviy ishining asosiy vazifasi kafedra faoliyatini avtomatlashtirish va yengillashtirishdan iborat .

Kafedra tizimini yaratishdagi asosiy vazifalar quyidagilardan iborat:

1. Berilganlar bazasini yaratish;
2. Admin moduli yaratish;
3. Foydalanuvchi moduli yaratish;
4. Tizim dizaynini yaratish.

Web-texnologiya asosida yaratilgan kafedra tizimi xavsizligini ta'minlash ham asosiy vazifalardan biri hisoblanadi. Xavfsizlikni taminlashda quyidagi masalalar yechiladi:

1. Foydalanuvchilar parolini saqlashda kriptografiyadan foydalaniladi;
2. Talaba va domlalar rollarini yaratish. Shu rollarni tizimdan o'rnini aniqlash;
3. Mahfiy foydalanuvchilarning tizimdagi o'rnini hisobga olish.

Qo'yilgan masalani yechish uchun quyidagilarni o'rganib chiqish talab qilinadi:

1. HTML, CSS, JavaScript, AJAX, jQuery, PHP;
2. MySQL;
3. Yii framework, Bootstrap.

1.2. Masalani yechish mexanizmi

Kafedra faoliyatini avtomatlashtirish tizimini yaratishda XAMPP va Aptana Studio 3 dasturiy ta'minoti Windows operatsion tizimida o'rnatildi. Bu kafedra axborot tizimi php tilida yaratildi. Chunki bu tilning tushunarligi va soddaligidir. Yana shu tilda yaratilgan yiiframeworkdan foydalanildi, bunga sabab bu frameworkning imkoniyatlari biz yaratgan tizimga mos. Berilganlar bazasini saqlash MySql tizimida amalga oshirildi va server sifatida Apache server tanlandi.

AJAX

Ajax texnologiyasi terminini Jesse James Garrent 2005-yil 18-fevraldagi "Web sahifalarga yangi yondashuv" nomli maqolasida birinchi bo'lib ishlatgan. Keyinchalik uni amalda qo'llagan. AJAX - Asynchronous Javascript and XML ya'ni asinxron [JavaScript](#) va [XML](#) [9].

Web texnologiyalari 2 turga bo'linadi.

- Server
- Klient

Klient qismdagi ayrim qulayliklarning ta'minlanishi ajax texnologiyasi bilan bog'liq ya'ni internet tezlik uncha yuqori bo'lmasa ham ajax texnologiyasi yordamida serverdan ma'lumotlarni olish mumkin. Agar php yordamida serverga murojaat qiladigan bo'lsak sahifalar qayta yuklanishi kerak bo'ladi. Bunda internet tezligi yuqori bo'lishi kerak va har bir harakat uchun sahifalar yangilanishini kutish kerak. Ajax texnologiyasi serverga qismga yordamchi qism hisoblanadi, ya'ni foydalanuvchining imkoniyatlarini oshiradi.

Ajax texnologiyasining eng muhim jihatlaridan biri bu sahifalarni qayta yuklamasdan serverdan ma'lumotlarni yuklab oladi.

Ajax ning 2 ta xususiyati bor:

1. Sahifalar qayta yuklanmasidan serverga borib amal bajarib qaytib kela oladi. Bunda ma'lumotlar JSON yoki XML formatida jo'natiladi va qabul qilinadi.

2. DHTML ya'ni o'zgaruvchan sahiflar yaratish, ya'ni javascript bilan ishlash.

Yiiframework

Yiiframework dunyo bo'ylab ancha keng tarqalgan. Bu framework bepul va qulay uni <http://yiiframework.ru> saytidan yuklab olish mumkin. Shuning uchun ham shu frameworkdan foydalanildi. Bu framework faqat Web serverlar (Denwer, XAMPP) kabilarda ishlaydi [5].

Framework bu— web sayt qilish jarayonida qiyinchilik tug'diradigan yoki ko'p takrorlanadigan funksiyalar to'plamidir. Umuman olganda dasturchiga yordamchi funksiyalar.

CMS — Content Management System bu deyrli tayyor sayt.

Umuman olganda PHP dan framework, frameworkdan CMS, CMS dan Website yaratiladi.

MVC (Model View Controller, Model Ko'rinish Nazorat) Bu umumiy tushuncha bo'lib, loyihalash shablonidir.

Model bu- ma'lumotlar bazasidagi biror bir jadval bilan ishlaydigan obyekt (class)dir.

View bu- HTML kodlarni yozasidan fayl yoki foydalanuvchiga (browserda) ko'rinadigan kod.

Controller bu— foydalanuvchi so'roviga mos holda kerakli model bilan ishlab, kerakli viewni foydalanuvchi jo'natadi. Yiiframeworkdan foydalanish uchun yii ning classlaridan voris class olib, shu classlardan foydalanib sayt quriladi.

PHP

1994 yili php tilinig yaratuvchisi Rasmus Lerdorf o'zinig saytiga mehmonlar kirishini hisoblash uchun Perl dasturlash tilada maxsus qobiq yozib amalda qo'llagan. Ko'p o'tmay qobiqni ishlash unumdorligi juda past va sekinligi aniqlanganidan so'ng, dasturlarni yangidan "C" tilida yozib chiqishga to'g'ri keladi. Keyin, dastlabki dastur kodlari muallif tarafidan barchaga ko'rish uchun

serverga nashr qilingan. Server foydalanuvchilari kodlar bilan qiziqib, uni ishlatish muxlislari ham paydo bo'lgan. Hademay, bu dasturlar alohida loyihaga aylanib, 1995 yilning iyun oyida dasturiy mahsulot PHP(Personal Home Page) nomi bilan birinchi nashri chiqarildi. Imkoniyatlari oddiy va sodda bo'lib: bir necha buyruqlarni tushunadigan kod analizatori(tekshiruvchisi), mehmonlar hisoblagichini, kitobini, chatini yaratish uchun foydali dasturlar to'plamidan iborat bo'lgan [3].

1996 yil aprel oyida dasturlar jiddiy qayta ishlanganidan so'ng, PHP/FI (Personal Home Page / Forms Interpreter) nomi bilan mahsulotning ikkinchi nashri paydo bo'ldi. Bu mahsulot html-kod ichiga yozilib, html-formalarni qayta ishlab, hozirgi PHP dasturlash tilining tayanch imkoniytlarini ichiga olgan. PHP/FI kod yozilishi Perl tiliga juda oxshagan, lekin soddaroq bo'lgan.rasmuslerdorf ZivSuraski EndiGutmans Rasmus Lerdorf, Endi Gutmans va Ziv Suraski 1997 yili PHP/FI 2.0 nashri chiqdi. O'sha paytda bu mahsulot bilan dunyo bo'yicha bir necha ming odam foydalanib, taxminan 50 ming domen bo'lib, Internetning 1%-ni tashkil qildi.1997 yilda Endi Gutmans va Ziv Suraski PHP/FI kodini boshqatdan yozib chiqishdi, chunki eski kod ular ishlatayotgan elektron tijorat tizimlari uchun yaroqsiz edi. Eski kodning mualliflaridan yordam olish uchun ular birlashishni taklif etib, PHP3 nomli loyihani PHP/FI -ni rasmiy vorisi deb e'lon qilishdi. Yangi loyiha uyushgandan keyin PHP/FI loyihasi ishlab chiqarilishi to'xtatilgan.PHP 3.0 -ning eng kuchli taraflaridan biri uning kengaytirala olinadigan yadrosi(tizimning bosharuv qismi) bo'lib, bundan tashqari, ma'lumot jamg'armalar bilan, turli protokollar va interfeyslar bilan birgalikda ishlash keng imkoniyatlari yaratildi. Muvaffaqiyatga erishishga ancha ahamiyatli fakt bu yangi tilni boyligi va ob'ektlarga mo'ljallangan dasturlashni ko'lla olishi. Yangi loyiha bilan birga nafaqat tilni tashqi, ichki tuzulishi o'zgardi, balki o'zini nomi ham. Endi PHP qisqartmasi "PHP: Hypertext Preprocessor" ma'nosini anglatishi bildirildi.1998 yilning oxirida PHP foydalanuvchilarning soni o'n minglardan oshdi. Yuz mingdan oshiq veb-saytlar bu tilni qo'llashini e'lon qilishdi. Taxminan Internetning 10% serverlarida PHP 3.0 o'rnatilgan

edi.1998 yilning iyun oyda PHP3 to'qqiz oy ommaviy tekshiruvidan keyin rasman e'lon qilindi. Shu yilning qishida Endi Gutmans va Ziv Suraski PHP yadrosini qaytadan ishlab chiqarishni boshlashgan. Ularning asosiy vazifasi PHP tiziminig unumdorligini ko'tarish va kodning modullarini yaxshilash edi.1999 yilning o'rtalarida birinchi marta taqdim qilingan yangi yadro "Zend Engine" deb nomlangan ("Zend": mualliflar "Zeev" va "Andi" ismlardan tashkil topgan). Uni asosida tuzilgan yangi til PHP4 2000 yilning may oyida rasman chiqarilgan. Unumdorlik yaxshilangandan tashqari, PHP 4.0 muhim yangiliklarga ega bo'lib, sessiyalarni ko'llash, buferli chiqarish, kiritilgan ma'lumotlarni havfsiz qayta ishlash va yana bir necha yangi til tuzuvchilarini paydo bo'lishidan iborat.Hozirgi kunlarda "Zend Engine" qayta yaxshilanib PHP5 tili ishlab chiqarildi. Asosiy o'zgarishlar ob'ektlarga mo'ljallangan dasturlash modelida bo'lib, tilning imkoniyatlari yanada kengaytirdi.Hozirda PHP tilini yuz minglab dasturchilar ishatib, bir necha million saytlar yaratilib, Internetning 20% domenlarini tashkil etmoqda.

MySQL

MySQL Shvetsiyadagi MySQL AB firmasi tomonida yaratilgan bo'lib, 2008 yilning fevralida Sun Microsystems tomonida sotib olingan. Endlikda dastur kodini yaratishgda ushbu firma javobgar. MySQL Server ochiq, ya'ni tekin dasturiy ta'minot bo'lib, General Public License himoyasidadir ya'ni turli himoyalar qo'yilmagan(kalitlar,litsenziya). MySQL AB/Sun firmasi dastur kodiga mualliflik huquqiga egaligi sababli, ushbu dasturiy ta'minotning kommerzial turi ham bor.MySQL nomining kelib chiqishi aniq emas. 1996 yildan beri ko'plab biblioteka va tool (vosita)lar nomi oldida My prefiksi qo'yila boshlangan [2].

Bu dastur o'zi bilan quyidagi amallarni bajaradi, yani kompyuterlar ichki tizimi bilan aloqa o'rnatish va ma'lumotlarni takrorlash va ma'lumotlarning saqlanish joyini aniqlash.

Berilganlar bazasining alohida bo'limlaridan turli foydalanuvchilar qismlarga ajratib foydalanishlari mumkin, berilganlar bazasining asosiy komponent tizimi bu ma'lumotlar hisoblanadi.

Berilganlar bazasining boshqa bir komponenti, foydalanuvchi, texnik hodim (hardware) va dastur ta'minotchisi (software) xisoblanadi, bu yerda Hardware-ichki qo'shimcha xotiraga ega bo'lgan disk va magnit lentalaridir. Berilganlar bazasining dasturiy ta'minoti foydalanuvchi tizimi va berilganlar bazalari bilan o'zaro aloqani o'rnatish vazifasini bajaradi, shunday dasturiy ta'minot bo'limiga, berilganlar bazalarini boshqarish tizimlari (BBBT) deb yuritiladi.

Foydalanuvchilarning berilganlar bazalari bo'yicha barcha talabnomalarini berilganlar bazalarini boshqarish tizimlari boshqarib turadi, berilganlar bazalaridan foydalanuvchilar ham bir necha turga bo'linadi:

- *Dasturchi foydalanuvchi*
- *Tizimli dasturchi*
- *Administratorlar*

Berilganlar bazalaridagi dasturlarning to'g'ri yozilishiga dasturchi javob beradi, administrator esa berilganlar bazalaridagi barcha ma'lumotlarning saqlanishini va kengayishini qo'llab quvvatlaydi.

Ko'p tarmoqli foydalanuvchilar uchun administrator berilganlar bazasida asosiy jarayon hisoblanadi.

Berilganlar bazasining tizimi, quyidagi jarayonlarga ega bo'lishi kerak:

- a) Bajarish- tizimning ishlashi vaqtida, foydalanuvchi so'roviga foydalanuvchi terminali mos tushishi kerak.
- b) Minimal takrorlanish-ichki tashuvchilardagi axborotlar hajmi, berilganlar bazasidagi saqlanayotgan axborotlarning takrorlanish darajasi bilan minimal holda bo'lishi talab etiladi.

Bundan tashqari har qanday jarayon, berilganlar bazasidan, hech bo'lmaganda bitta amal bajarilishini va undagi axborotning takrorlanishini talab etadi, bu esa boshqa jarayonlarning bajarilishi darajasini oshiradi. Shuning

uchun berilganlar bazasini tashkil etishda, axborotlarning ko'p marotaba takrorlanmas-ligiga erishish kerak.

v) To'liqlik- berilganlar bazasi bilan ishlovchilar uchun, yangi axborotlarni kiritish uchun qonunlar mavjud bo'lib shulardan foydalanish talab etiladi. Shuni e'tiborga olib tizimda saqlanadi-gan axborotlar ichki ma'lumotlar bilan o'zaro aloqaga kirishadi.

g) Xavfsizlik- Berilganlar bazalari ishonchli ximoya qilingan bo'lishi kerak, berilganlar bazalaridan noqonuniy foydalanish, va undagi ma'lumotlarni o'zgartirishga ruxsat etilmaydi, foyda-lanuvchi yoki tashkilot axborot olish yoki kiritish uchun maxsus ruhsatnomaga ega bo'lishi kerak.

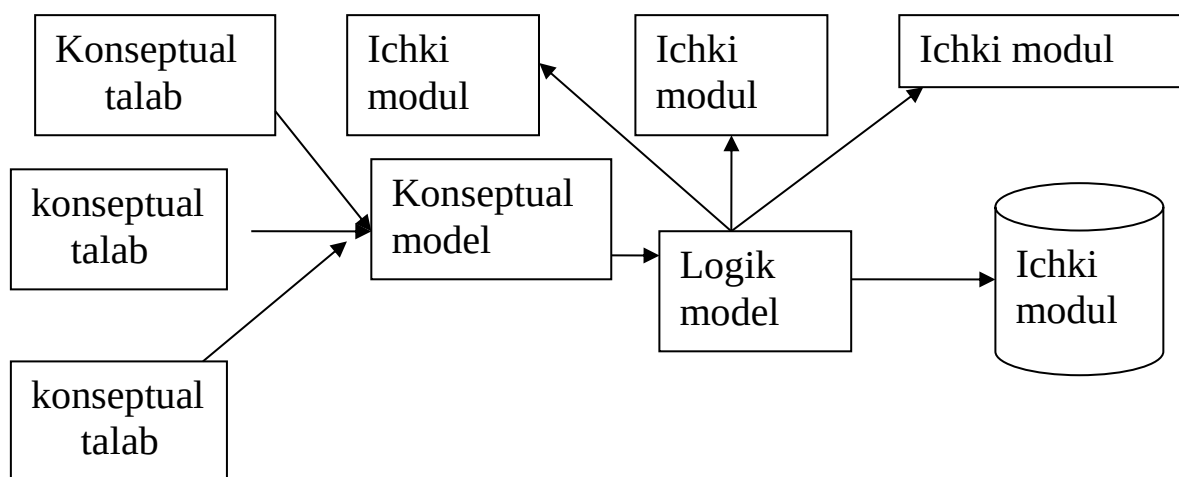
d) Migrasiya- Ba'zi bir ma'lumotlarga foydalanuvchilar tamoni-dan ba'zi-ba'zida murojaat amalga oshiriladi. Shuni e'tiborga olib, ma'lumotlarni ichki ma'lumot tashuvchilarga joylashtirish qulay bo'ladi, bu esa doimo ishlatiladigan axborotlarni tez qidi-rib topishga va uni ishlatishga imkon beradi.

Berilganlar bazalarini boshqarish tizimlarida mustaqil ma'lumot-lar mavjud bo'lib, ulardan faqat dasturchilar foydalanishlari mumkin. Dasturchi berilganlar bazasidan quyidagilarga javob olishi kerak bo'ladi:

- a) Berilganlar bazasida qanday qilib ma'lumotni format qilish kerak ?
- b) Berilganlar bazalarida ma'lumotlar qayerda joylashtiriladi?
- c) Berilganlar bazalarida ma'lumotlar qanday etib boradi?

So'ralayotgan savollarga tegishli javob berilgandan so'ng foydala-nuvchilar dasturining o'zgarishiga olib keladi, barcha ma'lumotlar, yuqoridagi uchta savol korinishida yozilib foydalanuvchining dasturida kod ko'rinishida ishtirok etadi. Bundan tashqari komp-yuterlarning ishlash darajasining o'sishi sababli undagi axborotga kirish usullari va bazida saqlab qo'yilgan axborotlarning o'zgari-shiga olib keladi. Berilganlar bazasidagi dasturiy ta'minot qo'yidagicha chizma asosida ish olib boradi:

1-foydalanuvchi



1-rasm . Berilganlar bazalarida mustaqil ma'lumotlarning joylashuvi

Berilganlar bazalarini ishlatish imkoniyatiga ega foydalanuvchilar uchun barcha imkoniyatlar namoyon etilishi bilan bir vaqtda , mustaqil ma'lumotlardan ham foydalanish imkoniyatlari yaratiladi. Boshqacha qilib aytganda (BBBT) berilganlar bazalarini boshqa-rish tizimlari ham shu tarzda quriladi.

Mustaqil ma'lumotlarga erishishning qulay usuli, yuqoridagi rasm-da keltirib o'tilgan. Foydalanuvchilarning har qanday talabi, bitta konseptual modelda jamlangan bo'ladi, shu konseptual model, so'ngra ma'lumotlarning namoyishini amalga oshiradi, qachonki berilganlar bazasini boshqarish tizimlari birgalikda o'rnatilgan bo'lsa.

Bootstrap

Hozirgi kundagi har bir internet foydalanuvchisi o'zlarining kundalik web-saytlarining mobil versiyasi ham bo'lishini hohlashadi. Buni amaliy jihatdan olib qaraganda bir dizayn har bir qurilma ekraniga mos tushishi muhimdir [8].



2-rasm. Moslashuvchan arxitektura

Responsive web design(2-rasm) – shunday yondoshuvki, bunda dizayn va “razrabotka” foydalanuvchi hatti harakati va muhiti(qurilma) ga ekran o’lchami, platformasi va orientatsiyasi asosida javob berishidir. Bu amaliyotda moslashuvchan to’r(grid) va maketlar aralashmasi, rasmlar va CSS media-so’rovlarning intellektual foydalanishidan iboratdir. Foydalanuvchi laptopdan iPadga o’tgan zahoti, web-sayt avtomatik moslashuvchanlikka(“razrishiya”, rasm va skriptlar-javascript) o’tishi kerak. Boshqacha qilib aytganda web-saytda shunday texnologiya bo’lishi kerakki, sayt avtomatik foydalanuvchi sharoitiga moslanishi kerak. Bu esa bozordagi har bir qurilma uchun har hil dizayn va “razrabotka” jarayonidan qutulish yo’lini yaratadi. **“Responsive web design” konsepsiyasi** Hozirgi kunda rivojlanib borayotgan “Moslashuvchan arxitektura” sohasida fizik sath qanday qilib soha orqali o’tayotgan odamlarga moslashishi mumkin degan savol ko’p berilmoqda. Arxitektorlar devorga o’rnatilgan robototexnika va cho’ziluvchan materiallar orqali, odamlar yaqinlashganda egiladigan va kengayadigan devor strukturalarini tajribadan o’tkazmoqdalar. Yoki harakatni ilg’ovchi datchiklar xonada odamlar soni ko’payishi davomida xona temperaturasi va yoritishini nazorat qilish uchun iqlim-boshqarish tizimlari bilan birlashtirilishi mumkin. Ba’zi kompaniyalar allaqachon odam oynaga qanchadir masofada yaqinlashganda avtomatik qorong’ulashishi yoki yorug’lashishi mumkin bo’lgan “aqlli oyna texnologiyasi”ni ishlab chiqishgan. Agar ushbu nazariyalarga web-dizayn

tomonidan qaraganda, webda ham shunga o'xshash bo'ladi. Moslashuvchan arxitektura singari, web-dizaynlar ham avtomatik moslashishi zarur. Bu har bir foydalanuvchi kategoriyasiga sanoqsiz individual yechimlar talab qilmasligi kerak.

Moslashuvchan web-dizaynlar ko'proq fikrlashning abstrakt yo'lini talab qiladi. Shuningdek, ba'zi g'oyalar allaqachon tajribadan o'tkazilgan: "cho'ziluvchan layout(maket)" lar, media so'rovlar va scriptlar(jquery, javascript) yordamida sahifalarni reformat qilish. Lekin "Responsive web-design" faqat ekran o'lchamiga moslashuvchan va avtomatik o'lchami o'zgaruvchan rasmlardan iborat emas. Tizimni yaratishda huddi shu texnologiyani qo'llagan ya'ni bootstrapdan foydalanilgan. Bootstrapning barcha komponentlari bepul. Bootstrapni o'rnatish juda oson bunda bootstrap-theme.css, bootstrap-theme.min.css, bootstrap.css, bootstrap.min.js, bootstrap.js larni bog'lab qo'ysa kifoya va bu bootstrapda foydalanish uchun bootstrapning har bir classi va div lari nima ish qilishi bilan tanishib chiqish kerak. Shundagina ko'zlangan maqsadga erishish mumkin

Html, CSS, JavaScript

HTML hujjati mantiqiy tarkibga ega. HTML hujjati sarlavhasida, inson boshidagi kabi, imkoniyatlar taqdim etilgan; ular "sahnadan tashqarida" murakkab amallarni boshqarish uchun ishlaydi. HTML tanasi fizik atributlar sohasi hisoblanadi. HTML tanasi – bu brauzer orqali sahifada ko'rinadigan barcha ma'lumotlar hisoblanadi. Standart deteglar yordamida tanani bo'lishning formal usullari mavjud emas. Ammo, BODY sohasida va mantiqan HTML hujjatining hammasida, buni amalga oshirishning sodda usuli mavjud.

Bu masala maxsus izoh detegi yordamida yechiladi. U hech qanday qoidaga bo'yinsinmasligi bilan HTML ning boshqa deteglaridan ajralib turadi. Lekin, sahifa haddan tashqari uzun va murakkab bo'lganda, sahifa ichida yo'nalish olishni osonlashtiruvchi kuchli deteg hisoblanadi.

Izohlarni hujjatning ixtiyoriy joyida ishlatish va bir necha satrlarga joylashishi mumkin. Izohlar sizni va boshqa xodimlarni chalgʻitmasligi, yaʼni yordam berish lozim. Ulardan faqat zarurat tugʻilgandagina foydalaning.

Hujjat sarlavhasiga quyidagi axborotlar kiradi.

1. *Sahifa nomi.* <TITLE> va unga mos boʻlgan </TITLE> detegi sahifaning nomlanishini aniqlaydi. U HTML-sahifaning tanasida kelmaydi, yaʼni brauzerda koʻrilganda bu maʼlumot asosiy oynada koʻrinmaydi. Bu nom brauzerning sarlavha satrida chiqariladi. <TITLE> detegida sahifaning nomini koʻrsatish lozim. U sodda boʻlishi, ammo sahifaning mazmunini tushunarli bayon etishi lozim. TITLE elementiga HTML ning boshqa deteglarini qoʻshish mumkin emas, ammo maxsus belgilardan foydalanish mumkin.

2. *Ssenariy.* Sahifada bajariluvchi har qanday ssenariy, masalan, JavaScript tilida yozilgan, hujjat sarlavhasiga biriktiriladi. Bosh (HEAD) vizual emas, balki mantiqiy jarayonlar uchun moʻljallangan. Ssenariyning oʻzi koʻrinmaganligi sababli, uning bajarilish jarayoni ham koʻrinmaydi. Jarayonnig natijalari sahifaning holatini aniqlaydi.

3. *Stil.* Oʻz HTML sahifalariga boshqaruv elementlari va stilarni qoʻshuvchilar uchun sarlavhaga kaskadli sahifalar stillarini kiritish yoki ularni Web-sahifalar bilan bogʻlash imkoniyati mavjud. Bu maʼlumotlar ham HTML hujjati sarlavhasida joylashtiriladi.

4. *Metaaxborot.* META detegi oʻzining koʻplab “mantiqiy” jarayonlarni bajarish imkonini beruvchi kuchi bilan ajralib turadi. Masalan, hujjat muallifini, kalit soʻzlarni yoki maxsus harakatlarni koʻrsatish.

Koʻplab sahifalar ssenariylarni, varaq stillarini yoki metaaxborotlarni talab qilmaydi, ammo barcha sahifalar uchun nom zarur.

HTML hujjati tanasida (BODY deteglar juftligi) koʻrish uchun taqdim etiladigan barcha axborot joylashadi. Qobiq yaratilgandan soʻng birinchi masalalardan biri sahifa tanasiga axborot kiritish boʻladi. U quyidagilardan iborat boʻlishi mumkin.

5.*Matn.* Bo‘g‘inning matnli mazmuni tanada joylashadi. Axborotni taqdim etish uchun formatlashdan (matn bo‘g‘inga masofaviy kiruvchilarga o‘qishda qulay bo‘lishi uchun) foydalaniladi.

6.*Tasvirlar.* Bo‘g‘in tematikasini ko‘rsatish uchun grafikadan, matnni to‘ldiruvchi fototasvirlardan yoki navigatsion tugmalardan foydalaniash mumkin. Har qanday holda ham tasvirlar hujjat tanasida joylashgan tasvirlar axborotning muhim qismi hisoblanadi.

7.*Muloqotlar.* Web ning yuragi va qalbi bo‘lgan muloqotlar foydalanuvchilarga bo‘g‘inda harakatlanish imkonini beradi. Muloqotlar har doim sahifa tanasida joylashadi.

8.*Multimedia va maxsus dasrurlar.* Shockwave, Flash, Java apletlari va hattoki biriktirilgan videoni boshqarish uchun HTML-hujjat tanasida joylashuvchi kod qo‘llaniladi.

Birinchi bor JavaScript 1995 chi yil o‘rtaga chiqdi. JavaScript 1995 yili ishlatilishi boshladi, ammo 1998 yilga kelib keng qo‘llanishga kirdi. Uning chiqishini asosiy sabablaridan biri bu Client tomonidan bo‘ladigan kiritishlarni tekshirish uchun, yani server tomonida qilinadigan ishlarni bir qismini olib tashlash uchun. Lekin shuncha vaqt o‘tgandan so‘ng javascript web development'ning eng asoslaridan biri bo‘lib qoldi. Microsoft kompanyasi ham o‘zining scripting language yani Vbscriptni chiqardi ammo lekin uni keng qo‘llab bo‘lmas edi, sababi Internet Explorer'dan boshqa hech qaysi Web Browser uni ishlata olmasdi. Keyin esa Microsoft ham Internet Explorer ichiga JavaScript'ni qo‘shdi. Hozirgi kunga kelib hamma Web

Browserlar JavaScriptni ishlata oladi. JavaScript bu “Client side programming” yani kimki sahifangizni ochsa shu odam tomonida ishlaydi. PHP, Perl, CGI, JSP lar kabi serverda ishlaymaydi. JavaScriptda DHTML yani Dynamic HTML sahifalarini yozish mumkin. Bu degani foydalanuvchining qandaydir harakatiga qarab ish qilishi, va boshqalardir. JavaScript'ning asosi uch qismdir bular:

1. ECMAScript

2. The Document Object Model (DOM)

3. The Browser Object Model (BOM)

ECMAScript: Bu JavaScriptning asosi, yani programmalshidir, bu hech qanday browser'ga bog'liq bo'lmagan, hech qanday server ga bog'liq bo'lmagan qismidir. Buning ichida buyruqlar, script yozish usullari, o'zgaruvchan/o'zgarmaslar, operatorlar, kalit so'zlar dir (syntax, types, keywords, operators, statements)

DOM(The Document Object Model): API (Application programming interface) for HTML and XML,yani programmalar yaratish ko'rinishidir. Agar boshqa tillarda misol uchun Java yoki Delphi, Visual Basic'da programmalarida ular OOP Object Oriented Programming,ya'ni ularda asos va uning ichida elementlari bor,bu yerda ham huddi shunday.

BOM (Browser Object Model) Bu web browser bilan aloqali ishlardir. Yani bularga statusbar'ning tekstini o'zgartirish, toolbarni o'zgartirish va hkz, faqatgina browser bilan aloqali bo'lib web sahifaning o'ziga aloqali emasdir.

Bular:

1. Yangi browser ochish
2. Browserni yurgizish, katta kichik qilish, yopish
3. Cookie yuklanishi
4. Ekran kattaligi
5. Toolbar, Navigator lar

II BOB. Berilganlar bazasini tashkil qilish va uni boshqarish tizimi

2.1. Berilganlar bazasi tashkil qilish

Berilganlar bazasi-bu o'zaro bog'langan va tartiblangan ma'lumotlar majmui bo'lib, u ko'rilayotgan ob'yektlarning xususiyatini, holatini va ob'yektlar o'rtasidagi munosabatni ma'lum sohada tavsiflaydi. Darhaqiqat hozirgi kunda inson hayotida berilganlar bazasida kerakli axborotlarni saqlash va undan foydalanish juda muhim rol o'ynaydi. Sababi: jamiyat taraqqiyotining qaysi jabhasiga nazar solmaylik o'zimizga kerakli ma'lumotlarni olish uchun, albatta, berilganlar bazasiga murojaat qilishga majbur bo'lamiz. Demak, berilganlar bazasini tashkil qilish axborot almashuv texnologiyasining eng dolzarb hal qilinadigan muammolaridan biriga aylanib borayotgani davr taqazosi. Shuni qayd qilish lozimki, berilganlar bazasini yaratishda ikkita muhim shartni hisobga olish zarur:

Birinchidan, ma'lumotlar turi, ko'rinishi, ularni qo'llaydigan dasturlarga bog'liq bo'lmasligi lozim, ya'ni MBga yangi ma'lumotlarni kiritganda yoki ma'lumotlar turini o'zgartirganda, dasturlarni o'zgartirish talab etilmasligi lozim.

Ikkinchidan, berilganlar bazasidagi kerakli ma'lumotni bilish yoki izlash uchun biror dastur tuzishga hojat qolmasin.

Shuning uchun ham MBni tashkil etishda ma'lum qonun va qoidalarga amal qilish lozim.

2.2. Berilganlar bazasi tashkil qilishdagi asosiy talablar

Oddiyliigi: Foydalanuvchilar o'z ixtiyorida qanday ma'lumotlar borligini juda oson tanib va tushinishi mumkin.

Umumdorligi: Qaysi ma'lumotlardan foydalanish talab qilinsa, bu ma'lumotlarga qo'yiladigan talablar shunday tezliklar bilan ta'minlanadi. Tayyorligi: Foydalanuvchi har doim tez ma'lumot oladi, qachonki bu narsa unga zarur bo'lsa. BBBT quyidagicha tavsiflanadi:

- Исполнимость-Bajarilishlik,foydalanuvchi so'roviga hozirjavoblik bilan muloqotga kiritish
- Минималная повторяемость-Minimal takrorlanishlik
Ma'lumotlar bazasidagi ma'lumot iloji boricha kam takrorlanishi lozim,aks holda ma'lumotlarni izlash susayadi.
- Yaxlitlik-axborotni MBda saqlash iloji boricha ma'lumotlar orasidagi bog'liqlikni asragan holda bo'lgani,ayni muddao.
- Безопасность - Xavfsizlik berilganlar bazasi ruxsat berilmagan kirishdan ishonchli himoya qilingan bo'lishi lozim.Faqat foydalanuvchi va tegishli tashkilotgina ma'lumotlarga kira olish va foydalanish huquqiga egalik qilishi mumkin.
- Миграция-ba'zi bir ma'lumotlar foydalanuvchilar tomonidan tez ishlatilib turiladi,boshqalari esa faqat talab asosida ishlatiladi.Shuning uchun ma'lumotlarni tashqi xotiralarda joylashtiriladi va uni shunday tashkil qilish kerakki,eng ko'p ishlatiladigan ma'lumotlarga murojaat qilish qulay bo'lsin.

Berilganlar bazasi boshqarish tizimida har bir MB modeli quyidagi xususiyatlari bo'yicha tavsiflanadi:

1. Ma'lumotlar tuzilmalarining turi.
2. Ma'lumotlar ustida bajariladigan amallar.
3. Butunlikning cheklanganligi.

Bu xususiyatlarni e'tiborga olgan holda ma'lumotlar bazasi modellari quyidagi turlarga bo'linadi:

- Daraxtsimon (ierarxik) modellar.
- Tarmoqli (to'rli) modellar.
- Relyatsion modellar.

Yana shu narsani ta'kidlash lozimki, ma'lumotlar bazasi modellarining faqatgina yuqorida qayd qilingan modeli mavjud deyish noto'g'ri. Chunki, bulardan tashqari yana Berilganlar bazasining binar munosabatlar modeli, ER-

modellari, semantic model kabi boshqa turlari ham mavjud. Lekin amalda asosan dastlabki ta'kidlangan 3 turli modellar ko'proq qo'llanilib kelinmoqda. Shuning uchun ham ushbu modellar haqida qisqacha to'xtalib o'tish joiz.

Daraxtsimon (ierarxik) modelda ob'yektlar yozuvlar ko'rinishida ifodalanadi.

Ierarxik modelda ikki yarusdagi elementlar bog'langan bo'lsa, unday ma'lumotlar tarmoqli (to'rli) modelda ifodalangan deyiladi. Tarmoqli modellarda ham ob'yektlar daraxtsimon modellardagi kabi yozuvlar ko'rinishida tasvirlanadi. Ob'yektlarning o'zaro aloqalari yozuvlar o'rtasidagi aloqalar sifatida tavsiflanadi.

Relyatsion modellarda esa ob'yektlar va ularning o'zaro aloqalari ikki o'lchovli jadval ko'rinishida tasvirlanadi. Ma'lumotlarni bunday ko'rinishda tasvirlanishi ob'yektlarning o'zaro aloqalarni yaqqol tasvirlanishiga asos bo'ldi.

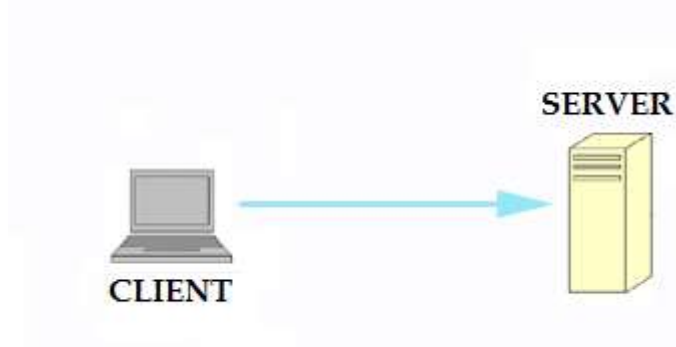
2.3. Tarmoqda ma'lumotlar bazalari arxitekturasini

Ma'lumotlar bazasi bilan ishlash uchun har xil tillarda amaliy dasturlar yaratish xususiyatlarni ko'rishdan oldin, bu amaliy dasturlarni loyihalashni ko'rib chiqish kerak. Biz kontseptual tushuncha ya'ni ma'lumotlar bazasini dasturlash asosida yotadigan klient/ server arxitekturasini ko'rib chiqamiz. Bu masalalar MySQL va MSSQL da dasturlash uchun muxim, lekin faqat ularga xos emas. Aksincha ular ma'lumotlar bazalarini dasturlash ixtiyoriy muhiti uchun muximdir. Agar arxitektura printsiplarini hisobga olmasa, sizni amaliy dasturlaringiz talablaringizga javob berolmaydi va o'zgaruvchi muxitga moslasha olmaydi. Biz quyida murakkab mavzularga ya'ni oddiy ikki bo'g'inli arxitektura, ob'ektlar va relyatsion ma'lumotlar orasidagi munosabat, hamda yangi uch bo'g'inli klient/server arxitekturasiga to'xtalib o'tamiz.

Klient/server arxitekturasini

Sodda xolda klient/server arxitektura amaliy dasturdagi qayta ishlashni ikki yoki undan ko'p mantiqiy qismlarga ajratishga asoslangan. Ma'lumotlar bazasi qandaydir amaliy dastur tomonidan foydalanish uchun yaratilgandir.

Soddalashtirib aytish mumkinki ma'lumotlar bazasi klient/server arxitekturasining bir qismini tashkil qiladi. Ma'lumotlar bazasi «server», undan foydalanuvchi har qanday amaliy dastur «klient». Odatda klient va server har xil mashinalarda joylashgan; ko'p xollarda klient amaliy dasturi ma'lumotlar bazasiga do'stona interfeysdir. Quyidagi grafik shaklda klient/server soddta tizimi tasviri berilgan.



3-rasm Klient/server arxitekturasini

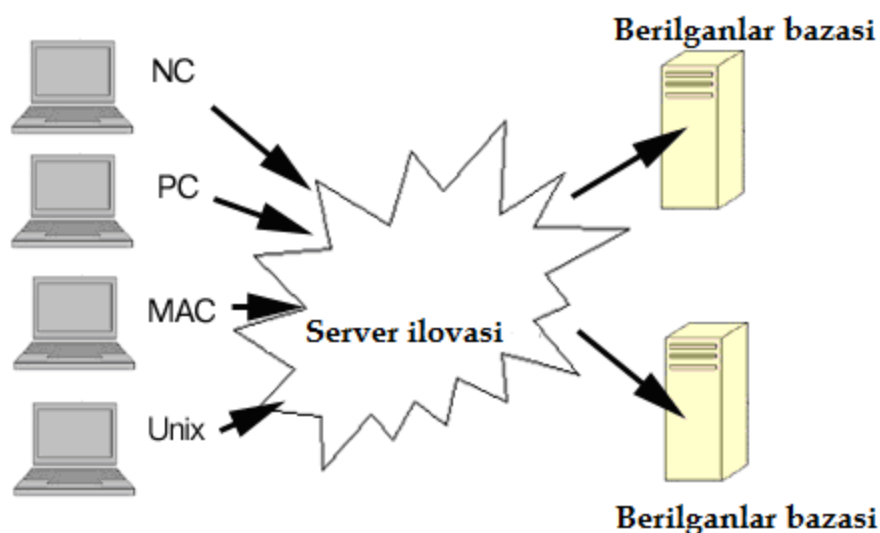
Berilganlar bazasi bilan ishlaydigan amaliy dasturlar yaratilganda avvalambor klientni ma'lumotlar bazasi bilan bog'lash imkoniyatiga ega bo'lish kerak. Berilganlar bazalari yaratuvchila dasturchilardan konkret tilga mo'ljallangan, API yordamida bog'lanish asosiy mexanizmlarni berkitadilar. Ma'lumotlar bazasi bilan ishlovchi dastur yaratganingizda sizni so'rovlaringizni tarmoq orqali ma'lumotlar bazasi serveriga uzatiluvchi TCP/IP paketlariga translyatsiya qiladi. Berilganlar bazasiga murojaat API larining tashqi ko'rinishi har xil va dasturlash tillariga, ko'p xollarda ma'lumotlar bazasining o'ziga bog'liq. MySQL uchun API lar MSSQL bilan o'xshash qilib yaratilgani uchun, biz ko'radigan API lar orasidagi farq minimaldir. Uch bo'g'inli arxitektura Arxitekturani Amaliy dasturlar rivojlanishi bilan takomillashtirish ancha murakkabdir. Bu arxitekturada ob'ektga-yo'naltirilgan dasturlash imkoniyatlaridan foydalanish ham qiyin. Birinchi muammo «nozik klientlar» haqidagi baxslarda o'z aksini topdi. Nozik klientlarga bo'lgan talab, klientga uzatilayotgan ma'lumotlar o'sib boish tendetsiyasidan kelib chiqdi. Bu muammo PowerBuilder va VisualBasic larda ko'rindi. Ular bazadan ma'lumotlarni GUI ga oladi va bu ma'lumotlar ustidagi hamma amallarni GUI da bajaradi. Foydalanuvchi interfeysini baza yadrosiga bog'lab qo'yish

foydalanuvchilar soni va ma'lumotlar xajmi oshishi bilan o'zgartirish va masshtablash qiyin bo'lgan dasturlar yaratilishiga olib keladi. Agar sizda foydalanuvchi interfeysi yaratish tajribasi bo'lsa, foydalanuvchi xoxishiga qarab interfeysni qayta ishlab chiqish muammosiga duch kelgansiz. Bunday qayta ishlashni kamaytirish yo'li GUI uchun faqat bitta vazifa – foydalanuvchi interfeysi vazifasini qoldirish kerak. Foydalanuvchi bunday interfeysi chindan ham nozik klientdir.

Masshtablanishga ta'sir o'tkazish boshqa tomondan ham ko'rinadi. Agar foydalanuvchilar soni va ma'lumotlar xajmi oshgani munosabati bilan amaliy dasturni qayta ishlab chiqish kerak bo'lsa, modifikatsiya ma'lumotlar bazasiga o'zgartirish kiritish yo'li bilan amalga oshirilishi mumkin. Masalan ma'lumotlar bazasini bir necha serverlarga taqsimlash yo'li bilan. Interfeysni ma'lumotlar bazasiga bog'lab qo'yish masshtablash muammosini xal qilish uchun GUI ni o'zgartirishga majbur qiladi. Aslida esa bu server bilan bog'liq muammolardir.

Nozik klientlar – bugunda yagona yo'nalish emas. Boshqa yo'nalish-koddan qayta foydalanish. Xar xil amaliy dasturlar uchun kod biznes logika deb atalgan qayta ishlashga yo'naltiriladi. Agar biznes lgogika foydalanuvchi interfeysida joylashgan bo'lsa, koddan qayta foydalanishni ta'minlash qiyin bo'ladi. Bu muammoni xal qilish yo'li Amaliy dasturni ikki qismga emas uch qismga ajratishdir. Bunday arxitektura uch bo'g'inli deyiladi.Klientdagi foydalanuvchi interfeys haqida gapirganimizda,mantiqiy farqni nazarda tutamiz. Nozik klient bir turi «O'ta nozik klient», bo'lib ko'pchilik Web-sahifa deb qabul qiladi. Web-sahifa dinamik tarzda Web-serverda yaratilishi mumkin. Bu holda klient ishining ko'p qismi serverda HTML-sahifalarni dinamik generatsiya qilish shaklida bajariladi.1- rasmda ko'rsatilgan ikki bo'g'inli arxitekturani 2- rasmda ko'rsatilgan uch bo'g'inli arxitektura bilan solishtiring. Foydalanuvchi interfeysi va berilganlar bazasi orasida qo'shimcha qatlam joylashtiriladi. Bu yangi qatlam amaliy dasturlar serveri o'zida biror soha uchun umumiy bo'ldgan Amaliy dastur ish mantig'i – biznes mantiqni oladi. Klient o'rta yarus ob'ektlarini ko'rish vositasi, ma'lumotlar bazasi bo'lsa shu ob'ektlar omboriga aylanadi.

Buning eng katta yutug'i-foydalanuvchi interfeysi bilan berilganlar bazasini ajratilishidir. Endi berilganlar bazasi haqidagi ma'lumotni GUI ga kiritishga to'g'ri kelmaydi. Balkim, ma'lumotlar bazasi bilan qanday ishlash haqidagi hamma ma'lumotlar, o'rta yarusda joylashishi mumkin. Amaliy dasturlar serverining ikki asosiy vazifasi-berilganlar bazasiga ulanishlarni izolyatsiya qilish va biznes mantiq uchun markazlashgan omborni ta'minlash. Foydalanuvchi interfeysi faqat ma'lumotlarni kiritish va akslantirish bilan shug'ullanadi, ma'lumotlar bazasi yadrosi bo'lsa faqat ma'lumotlar bazasi muammolari bilan shug'ullanadi. Ma'lumotlarni qayta ishlashni markazlashtirish Amaliy dasturlar serverining bitta dasturini har xil foydalanuvchi interfeyslari ishlatishi mumkin va har gal yangi amaliy dastur yaratilganda ma'lumotlarni qayta ishlash qoidalarini yozish kerak bo'lmay qoladi.

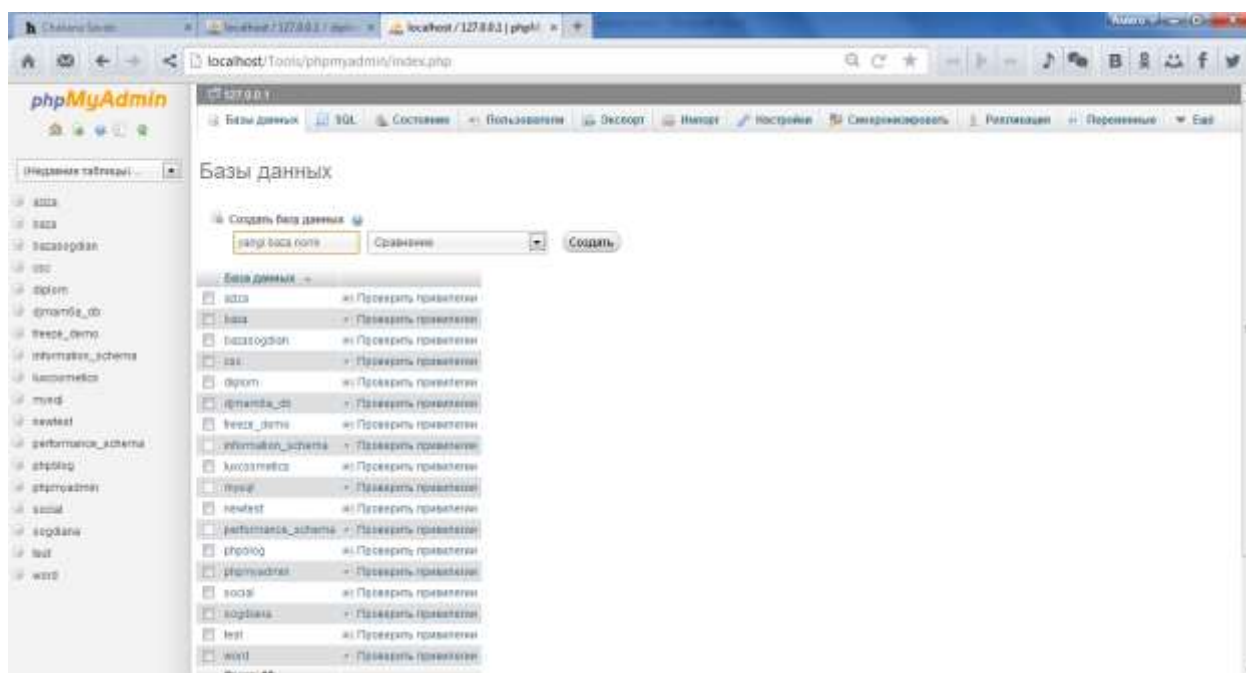


4-rasm Uch bo'g'inli arxitektura

III BOB. Avtomatlashgan kafedra tizimining ishlash tamoiili

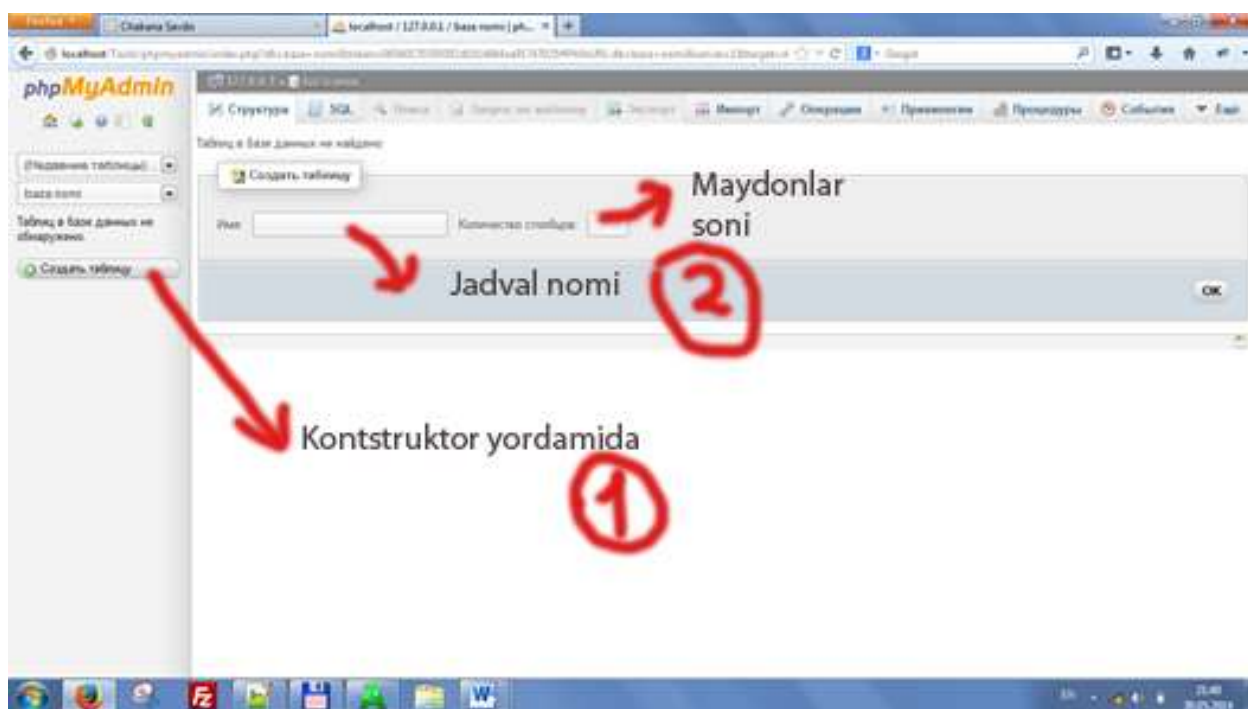
3.1. Berilganlar bazasidagi jadvallar sxemasi

Berilganlar bazasini yaratishda MySQL bazasida foydalanildi. MySQLda baza yaratish uchun denwerni o'rnatib ixtioryity brovserni ochib adresga localhost/tools/phpmyadmin yoziladi va "Базы данных" tugmasi bosiladi." Создать базу данных" degan joyiga bazani nomi yoziladi va "Создать" tugmasi bosiladi (5 - rasm).



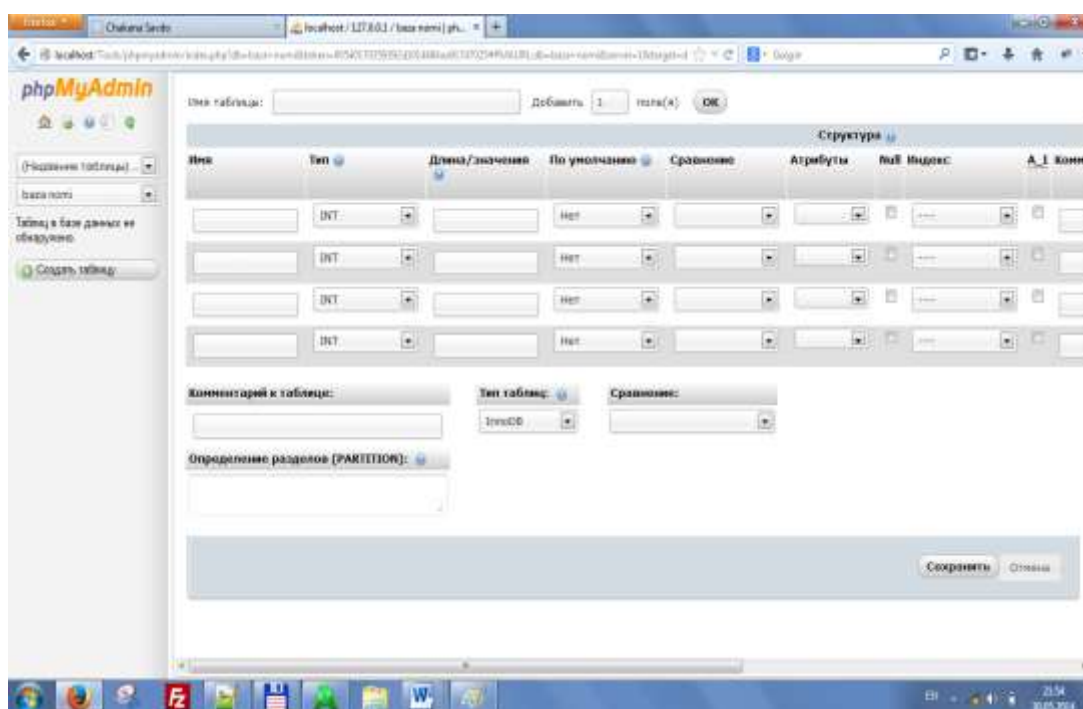
5 - rasm. Yangi baza yasash sahifasi.

Baza yaratilgandan so'ng jadvallar yaratilindi. Jadval 2-xil(6-rasm) usul bilan yaratiladi:



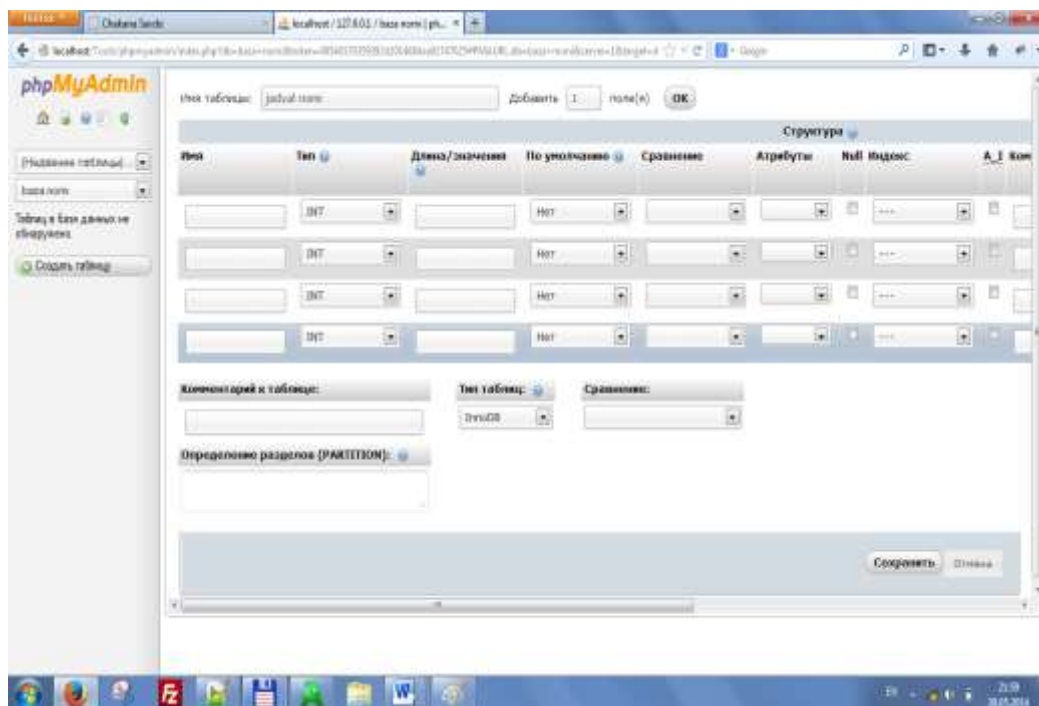
6- rasm. Yangi jadval yaratish sahifasi

1. Konstruktor yodamidan ya'ni bu yerda konstruktor 4 ta maydonli jadvalni avtomatik ravishda yaratib beradi.



7-rasm.Konstruktor yordamida yaratilgan jadval.

2. Bunda jadval nomi va maydonlari soni kiritiladi va shunga mos maydonlar hosil bo'ladi(8-rasm).



8-rasm.Konstruktorsiz yaratilgan jadval.

Yuqorida ko'rsatilgan qadamlarni qilgan holda quyidagi jadvallarni yaratamiz(9-rasm):

yKnownLanguages - jadvalda foylanuvchi qaysi tillarni qay darajada bilishi haqidagi ma'lumotlar saqlanadi.

xwbLanguage – foylanuvchi qaysi tillarni bilishi haqidagi ma'lumotlar saqlanadi.

yScientificWorks – foydalanuvchi ishlayotgan yoki ishlagan davri va muassasi haqidagi ma'lumotlar saqlanadi.

yScientificTitles – foydalanuvchining fsdjfs saqlanadi.

yRelatives – foydalanuvchining qarindoshlari haqida ma'lumotlar saqlanadi.

yScientificDegrees – ilmiy ishlari,diplom nomerlari haqida ma'lumotlar saqlanadi.

yDeputation – foydalanuvchining deputatlik haqidagi ma'lumotlar saqlanadi.

yPersonnal - foydalanuvchi haqida umumiy ma'lumotlar saqlanadi.

yBonus – foydalanuvchining davlat mukofatlari haqida ma'lumotlar saqlanadi.

xwbBonusType – foydalanuvchining qaysi davlat mukofatlarini olgan haqidagi ma'lumotlar saqlanadi.

yFormations – foydalanuvchining haqidagi ma'lumotlar saqlanadi.

yPostHistory – hozirda iwlalayotgan joyi haqida ma'lumotlar saqlanadi.

xwbCountry - tug'ilgan joyi adresi haqida ma'lumotlar saqlanadi.

xwbDeputyType – deputatning qaysi turi haqidagi ma'lumotlar saqlanadi.

xwbDocumentType – foydalanuvchining hujjatlari haqidagi ma'lumotlar saqlanadi.

xwbEducationType - ta'lim turi haqidagi ma'lumotlar saqlanadi.

(kunduzgi , kechki)

xwbEducationBase – ta'lim turi haqidagi ma'lumotlar saqlanadi.

xwbFormationType – foydalanuvchining ma'lumoti haqidagi hujjatlar saqlanadi.

xwbGenre – foydalanuvchining jinsi haqidagi ma'lumotlar saqlanadi.

xwbNationality – foydalanuvchining millati haqidagi ma'lumotlar saqlanadi.

xwbPersonalType –

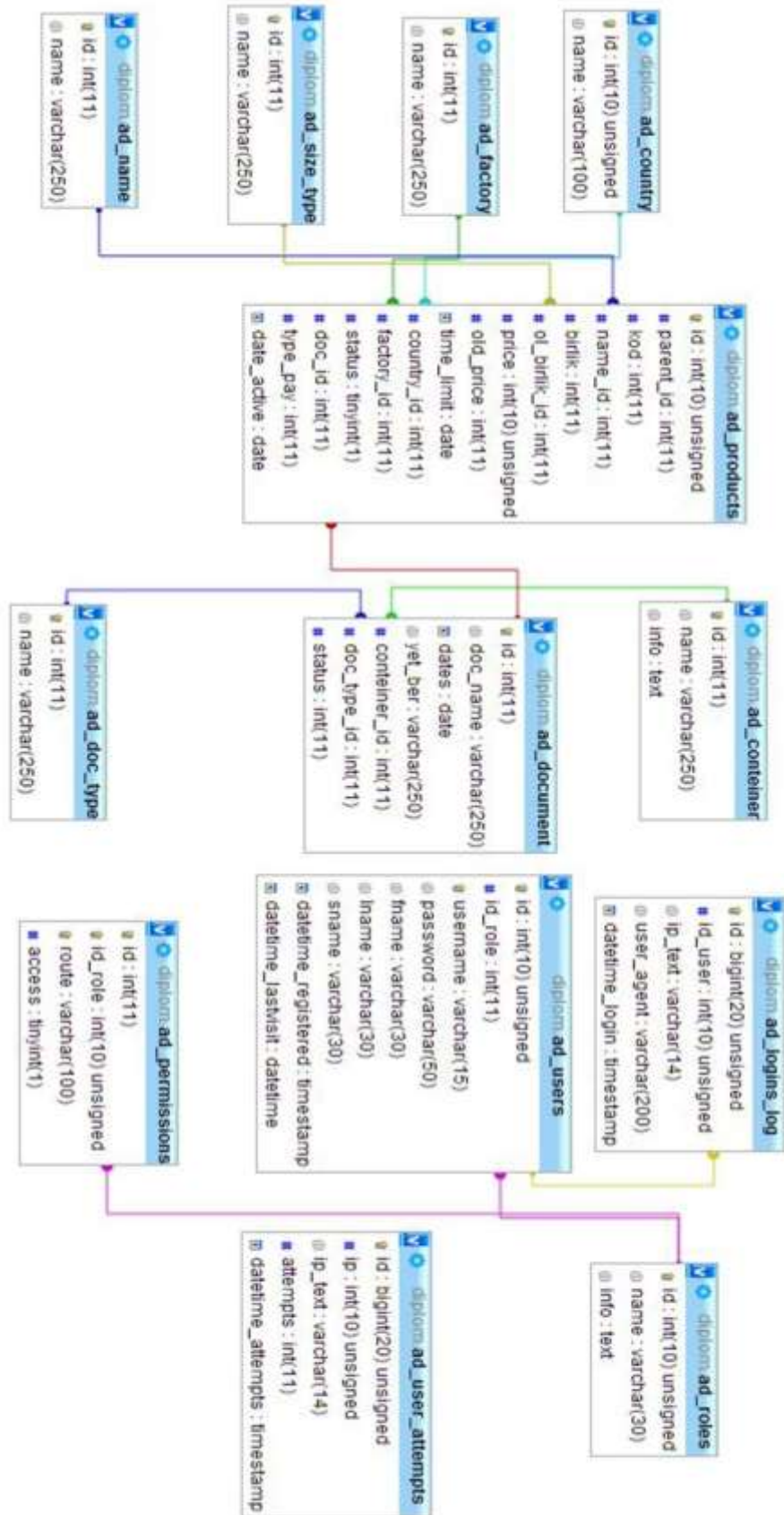
xwbRegion –

xwbRelationship – turmush qurgan yoki qurmaganligi haqidagi ma'lumotlar saqlanadi.

xwbRelativeType – qarindoshlari turi haqidagi ma'lumotlar saqlanadi.

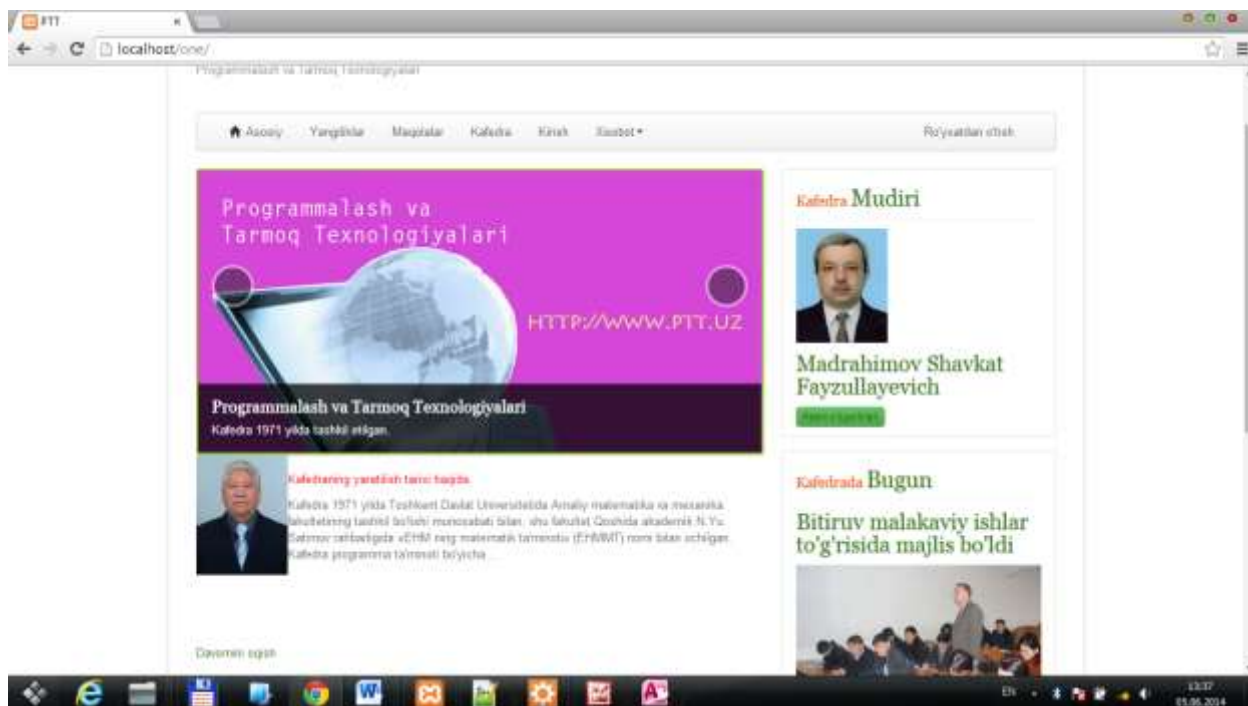
yPostHistory – foydalanuvchining oldin ishlagan joyi haqidagi ma'lumotlar saqlanadi.

yRelatives – qarindoshlari haqidagi ma'lumotlar saqlanadi.



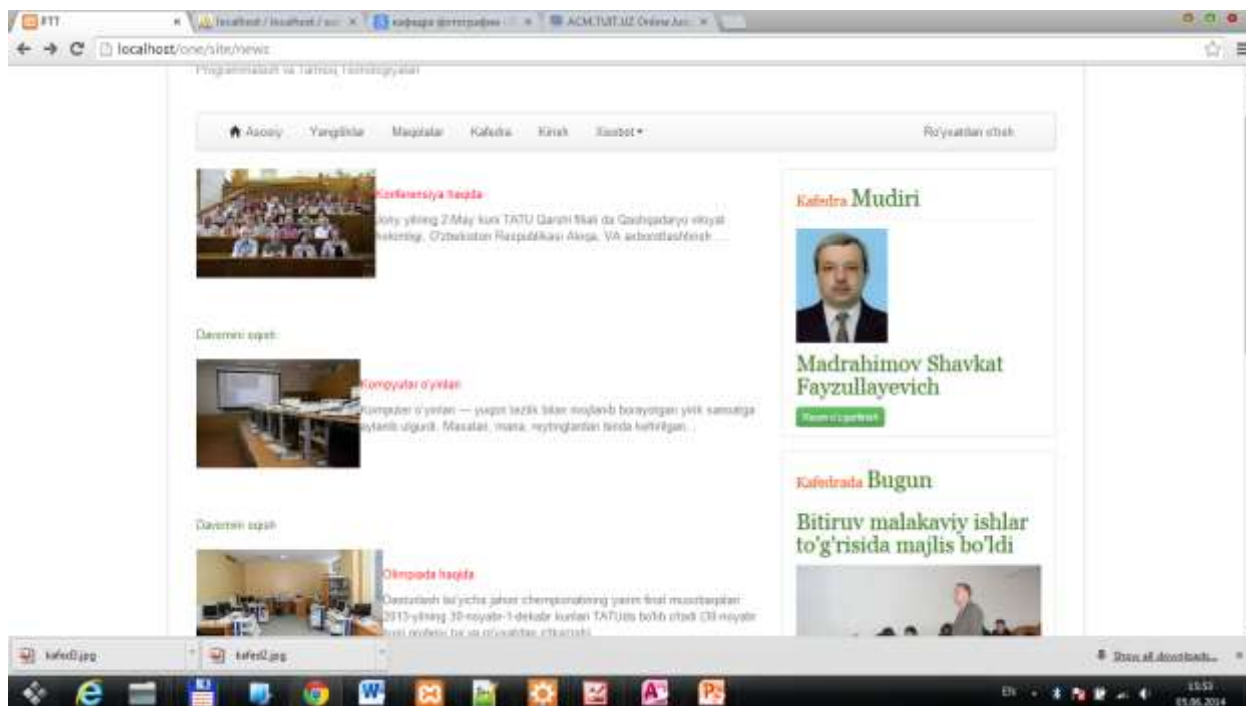
3.2. Tizimning tavsifi

3.2.1. Tizimga kirish. Tizimni (localhostda) ishga tushirish uchun denwer (apache, mysql) dasturi o'rnatiladi va denwerga start berib, host nomi orqali browserda ishga tushuriladi. Host nomi yozilganda quyidagi ko'rinishda sahifa ochiladi, bunda saxifaning asosiy qismi ko'rsatilgan (10-rasm).



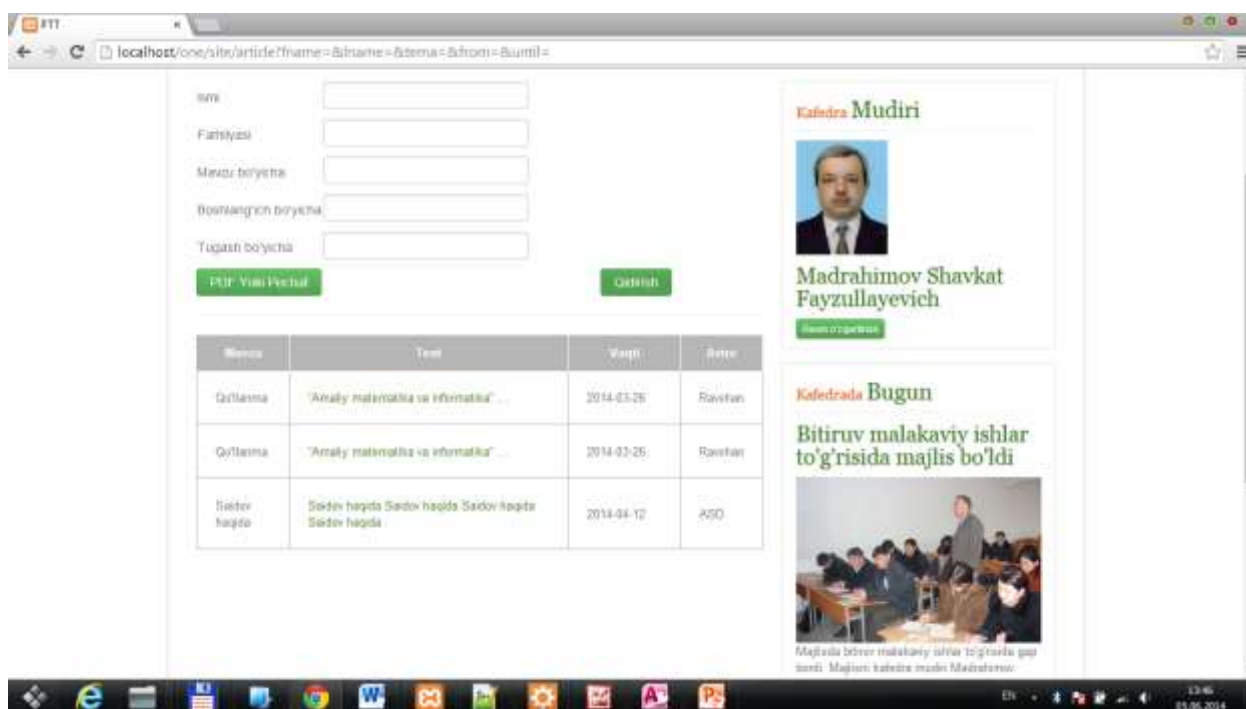
10-rasm.Tizimga kirish sahifasi.

Ikkinchi saxifada esa kafedra faoliyatiga tegishli yangiliklar joy olgan (11-rasm).



11-rasm. Tizimning yangiliklari saxifasi.

Quyidagi saxifada kafedrada chop qilingan maqolalar ro'yxati (12-rasm).



12-rasm. Tizimning maqolalar ro'yxati.

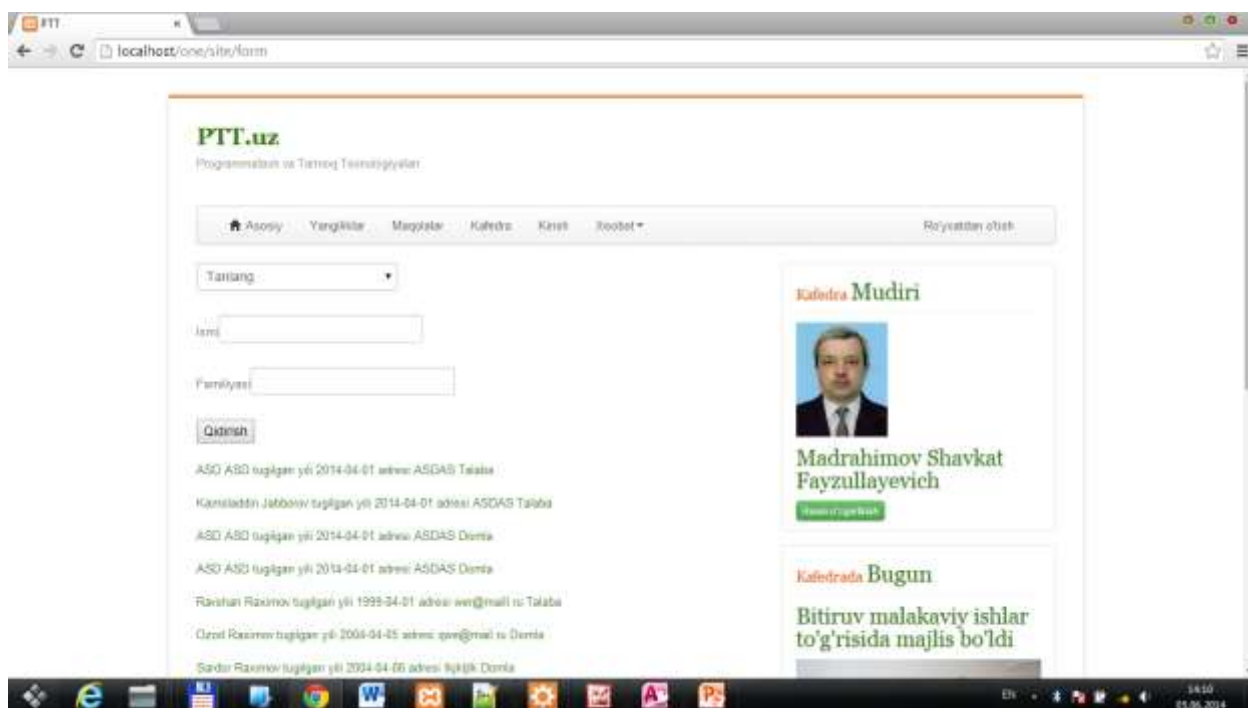
Yuqoridagi saxifadagi maydonlarning tavsifi quyidagicha:

1. Ismi – bu maydonga foydalanuvchining ismi yoziladi;
2. Familiyasi – bu maydonga foydalanuvchining familiyasi yoziladi;

3. Mavzu bo'yicha - bu maydonga foydalanuvchining chop etgan maqolasining mavzusi yoziladi;
4. Boshlang'ich bo'yicha – bu chop etilgan maqolaning sanasining boshlang'ich chegarasi;
5. Tugash bo'yicha – bu chop etilgan maqolaning sanasining tugash chegarasi.

Yuqoridagi maydonlar maqolalar ro'yxatini filtirlash uchun ishlatiladi. Barcha maydonlarni bir vaqtning o'zida to'ldirish shart emas.

Quyidagi saxifada kafedra o'qituvchilar va talabalar ro'yxati keltirilgan (13 - rasm)



13–rasm.Hodimlar ro'yxati.

Yuqoridagi saxifadagi maydonlarning tavsifi quyidagicha:

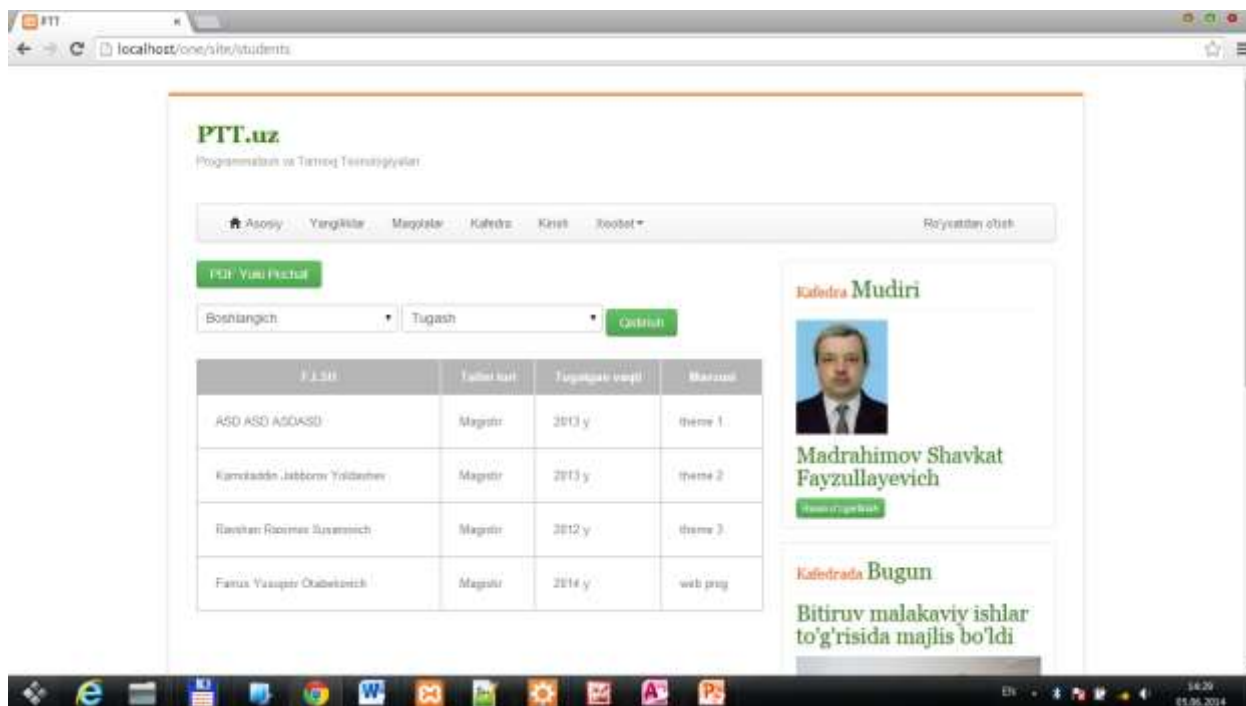
1. Tanlang – bunda talaba, o'qituvchi va hodimlar tanlanadi;
2. Ismi – bu maydonga foydalanuvchining ismi yoziladi;
3. Familiya – bu maydonga foydalanuvchining familiyasi yoziladi;
4. Qidirish – bu shartlar bajarilgandan keyin bosiladi.

Yuqoridagi maydonlar maqolalar ro'yxatini filtirlash uchun ishlatiladi.

Barcha maydonlarni bir vaqtning o'zida to'ldirish shart emas.

Tanlash amali bajarilgandan keyin bizga ro'yxat chiqadi. Keyin o'zimizga kerakli ma'lumotlar chiqadi. Ulardan shu shaxs haqida ma'lumotlar olamiz, o'zimizga kerakli ma'lumotlarni chop qilib olishimiz mumkin.

Quyidagi saxifada kafedraning yillik xisobotlari ro'yxati keltirilgan (14 - rasm)

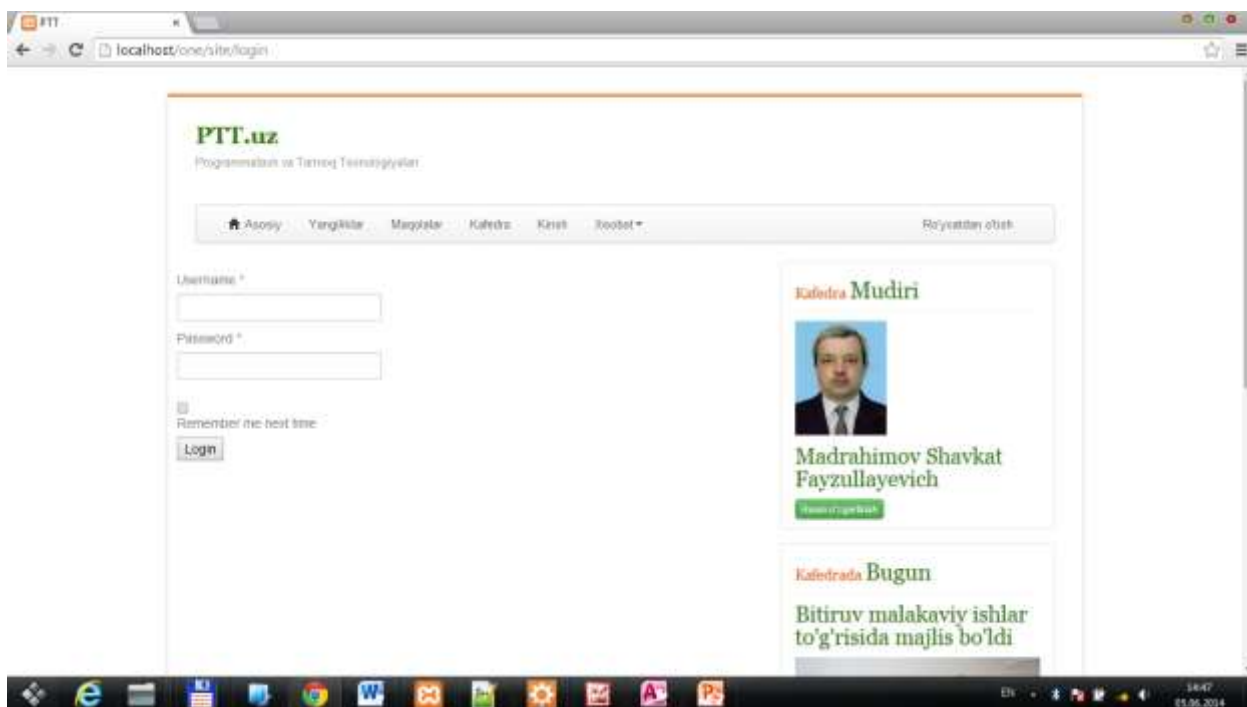


14-rasm. Yillik xisobotlar.

Yuqoridagi maydonlar xisobotlar ro'yxatini filtirlash uchun ishlatiladi.

1. Boshlang'ich – bu chop etilgan maqolaning sanasining boshlang'ich chegarasi;
2. Tugash - bu chop etilgan maqolaning sanasining tugash chegarasi;
3. Qidirish – yuqoridagilar tanlangandan keyin tugma bosiladi.

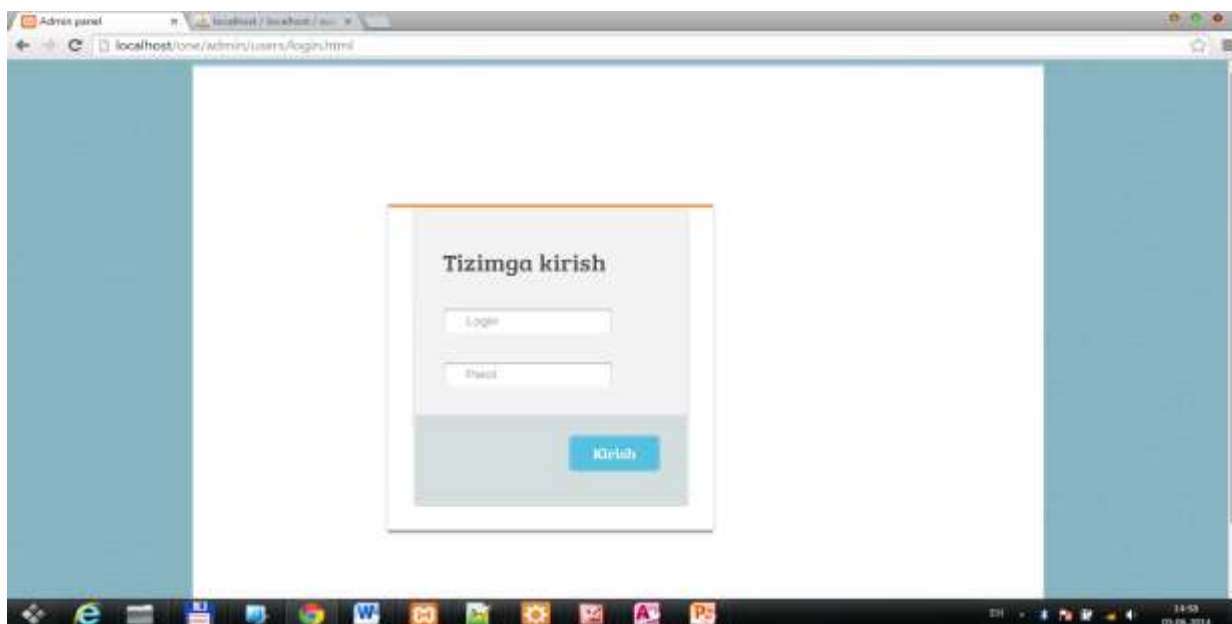
Quyidagi saxifada foydalanuvchining kirish paneli keltirilgan. (15-rasm)



15 – rasm. Kirish paneli

Yuqoridagi maydonlarda foydalanuvchi “Username” va “Password” kiritadi va “Login” tugmasini bosadi va ro’yxatdan o’tgandan keyin foydalanuvchi o’zining ma’lumotlarini o’zgartirish imkoniyatiga ega bo’ladilar.

Quyidagi saxifada admin moduli saxifasi keltirilgan. (16-rasm)



16-rasm. Admin panel

Yuqoridagi maydonlarda admin o'zining "Login" va "Parol" ini kiritib tizimning ma'lumotlarini o'zgartirib turadi.

Quyidagi saxifada esa ro'yxatdan o'tish paneli keltirilgan. (17-rasm)

The screenshot shows a web browser window with the address bar displaying 'localhost/one/site/registration'. The page features a registration form on the left and a sidebar on the right. The form includes the following fields: 'Tug'ilgan kuni *', 'Jinsi *', 'Millati *', 'Davlati *', 'Viloyati *', 'Tumani *', 'Qarindoshlari *', 'Tillari *', 'Citmonahp *', 'Passport nomeri *', and 'Passport serisi *'. The sidebar on the right contains a profile section for 'Kafedra Mudiri' with a photo of Madrahimov Shavkat Fayzullayevich and a 'Saqlash' button. Below this is a news section titled 'Kafedra Bugun' with the headline 'Bitiruv malakaviy ishlar to'g'risida majlis bo'ldi' and a photo of a meeting.

17-rasm. Ro'yxatdan o'tish

Yuqoridagi maydonlarning tavsifi quyidagicha:

1. Tug'ilgan kuni – bu maydonga foydalanuvchining tug'ilgan kuni yoziladi;
2. Jinsi - bu maydonga foydalanuvchining jinsi yoziladi;
3. Millati - bu maydonga foydalanuvchining millati yoziladi;
4. Davlati - bu maydonga foydalanuvchining davlati yoziladi;
5. Viloyati - bu maydonga foydalanuvchining viloyati yoziladi;
6. Tuman - bu maydonga foydalanuvchining tumani yoziladi;
7. Qarindoshlari - bu maydonga foydalanuvchining qarindoshlari yoziladi;
8. Tillar - bu maydonga foydalanuvchining biladigan tillari yoziladi;
9. Passport nomeri - bu maydonga foydalanuvchining passport nomeri yoziladi;

10.Passport seriasi – bu maydonga foydalanuvchining passport seriasi yoziladi;

11.Email - bu maydonga foydalanuvchining emaili yoziladi;

3.3. Tizimning qulayligi va avzalligi.

Yaratilayotgan loyhalar, dasturlar yoki dasturlar paketining qay darajada qulayligi va avzalligi alohida kasb etadi. Ushbu web-texnologiyalar asosida yaratilgan kafedra tizimi ham bir qancha qulayliklarga ega:

- Ixtiyoriy sistema foydalanuvchisi internet tarmog‘i orqali sistemada ixtiyoriy joyda ishlay olishi;

- Dastur o‘zbek tilda ekanligi;

- Foydalanuvchi ushbu sistemada ma‘lumotlarni qidirish jarayonida informatsion sohada taaluqli bilimlarni talab qilmaydi;

- Sahifaning sodda strukturaga ega ekanligi;

Xulosa

Ushbu bitiruv malakaviy ishini yozish davomida ko'pgina ma'lumotlar bilan tanishdim. Shulardan web-texnologiyalardan foydalanishni, ularni hayotdagi ahamiyatini yoki hayotga taqbiq qilishdagi rolini, berilganlar bazasini boshqarishni, unda turli murakkab so'rovlar yozishni o'rgandim.

Zamonaviy texnologiyalar jadallik bilan rivojlanyotgan kunda, dasturlash tillarining ham yangidan yangi turlari ishlab chiqilmoqda. Shu bilan birga ularning ishlash prinsiplari tobora mukammallashib foydalanuvchi ishlashi uchun qulay imkoniyatlar tug'ilmoqda.

Web texnologiyalari yordamida har bir yaratilgan tizim ayrim sohalarning avtomatlashishiga sabab bo'lmoqda. Avtomatlashgan tizimlar esa inson mehnati kamaytiradi va u bir qancha qulayliklar yaratadi. Har bir sohani avtomatlashtirish esa ish unumdorligini, ishning sifatini oshiradi va eng muhimi vaqtni tejaydi.

Yangi texnologiyalarni o'rganish, ularni ishlatish va ular uchun dasturlar tuzish va hayotga tadbiq qilish ham zamonamizning dolzarb masalalaridan biriga aylandi. Ushbu malakaviy ishini bajara turib web-texnologiyalar yordamida dastur yozish usullarini, yangi web texnologiyalarni (Ajax, bootstrap), ularning yozilish turlarini, rivojlanish tarixini va bir qancha shu dastur bilan bog'liq bo'ladigan hodisalarni o'rganib chiqdim. Dasturlarning xilma xilligi meni hayratga soldi, o'zim uchun yangi bo'lgan dasturlarning turlari bilan tanishib chiqdim.

Ishlash davomida men uchun tanish bo'lgan dastur PHP tilidan va berilganlar bazasini (MySQL) boshqarish tizimlaridan foydalangan bo'lsakda undagi malum tushunchalar haqida yanada ko'roq malumot oldik va ular ustida bitiruv malakaviy ishini yaratish davomida ishladik. Menga malum bo'lgan tushunchalar bir qatorda noma'lum ya'ni texnologiyalarni o'rganishda, ularning ma'nosini tushunishda, ularni axbotor tizimga tadbiq qilishda bir qancha muammolarga duch keldim va ular bartaraf qilishda oldingi bilimlarimni yanada mukammallashtirishga to'g'ri keldi. Bu orqali tajribam va dasturlash sohasidagi bilimlarimni oshirdim. va o'zim uchun

yangi bo'lgan ma'lumotlarni,yangi texnologiyalarni,dasturlashdagi bilimlarimni o'zlashtirdim.

Barchasini xulosa qilib aytadigan bo'lsak bitiruv malakaviy ishini bajarib shunday tushunchaga ega bo'ldimki: dasturiy vositalar bilan birgalikda ularning tuzilishi,ularning dasturiy boshqarish usullarini o'rganish,shu o'rganilgan tushunchalarni hayotta keng joriy etishdadur deb o'ylayman.

Fodalanilgan adabiyotlar

1. http://uzhurriyat.uz/foydali_saytlar/akt_taraqqiyotga_xizmat_qiladi.mgr
2. Веллинг Л., Томсон Л. Разработка Web-приложений с помощью PHP и MySQL, 3-е издание, 2008
3. Мориссон М, Бейли Л, Изучаем PHP и MySQL ,2010,Москва
4. А. Мазуркевич Д, Еловой ,PHP настольная книга программиста,2010, Санкт-Петербург
5. <http://yiiframework.com>
6. <http://yiiframework.ru>
7. <http://getbootstrap.com>
8. <http://wrapbootstrap.com>
9. <http://ru.wikipedia.org/wiki/AJAX>

Ilova

- Hujjatlar bilan ishlaydigan Controllar

```
<?php
class SiteController extends Controller
{
    public function actions()
    {
        return array(
            'captcha'=>array(
                'class'=>'CCaptchaAction',
                'backColor'=>0xFFFFFF,
            ),
            'page'=>array(
                'class'=>'CViewAction',
            ),
        );
    }
    ///yangiliklar boshlandi
    public function actionNews()
    {
        $model = YnewsModel::model()->findAll();
        $this->render("news",array("model" => $model));
    }

    public function actionMore($id = false)
    {
        $model = YnewsModel::model()->findAll(array("condition" => " id =
$id"));
        $this->render("more" , array("model" => $model));
    }

    public function actionMorearticle($id = false)
    {
        $model = YarticleModel::model()->findAll(array("condition" => " id =
$id"));
        $this->render("more" , array("model" => $model));
    }

    ///yangiliklar tugadi
    public function actionIndex()
    {
        $model = Yboshsaxifa::model()->findAll();
        $this->render('index',array("model" => $model));
    }
}
```

```

    }

    public function actionStudents( )
    {
        if(empty($_POST['from']) and empty($_POST['to']))
            $model = Ypersonal::model()->findAll('type_user = 2');
        else {
            $from = $_POST['from'];
            $to = $_POST['to'];
            $model = Ypersonal::model()->findAll
            (
                array(
                    'join'=>"inner join yinfostudent as i ON t.id=i.personal_id
",
                    'condition'=>"Year(i.enddate) between ".$from." and
".$to." and t.type_user = 2",
                )
            );
        }

        $this->render('student',array("model" => $model,'type'=>1));
    }

    public function actionStudentsWithT( )
    {
        if(empty($_POST['from']) and empty($_POST['to']))
            $model = Ypersonal::model()->findAll('type_user = 2');
        else {
            $from = $_POST['from'];
            $to = $_POST['to'];
            $model = Ypersonal::model()->findAll(array(
                'join'=>"inner join yinfostudent as i ON t.id=i.personal_id ",
                'condition'=>"Year(i.enddate) between ".$from." and ".$to." and
t.type_user = 2",
            ));
        }

        $this->render('student',array("model" => $model,'type'=>2));
    }

    public function actionArticle($fname = false, $lname = false, $tema = false,
    $from = false, $until = false)
    {
        if($fname)

```

```

        {
            $w = Ypersonal::model()->find(array("condition" => "name =
'$fname' and surname = '$lname' "));
            if($w)
            {
                $model = YarticleModel::model()-
>findAll(array("condition" => "author_id = '$w->id'"));
                $this->render("article" , array("model" => $model));
            }
            else
            {
                $this->render("article" , array("model" => ""));
            }
        }

        if($tema)
        {
            $model = YarticleModel::model()->findAll(array("condition"
=> "title = '$tema'"));
            $this->render("article" , array("model" => $model));
        }

        if($from and $until)
        {
            $model = YarticleModel::model()->findAll(array("condition"
=> " date_time between '$from' and '$until'"));
            $this->render("article" , array("model" => $model));
        }

        if(!$fname and !$tema and !$from and !$lname and !$until)
        {
            $model = YarticleModel::model()->findAll();
            $this->render("article" , array("model" => $model));
        }
    }

    public function actionMor($id = false)
    {
        $model = Ybosh saxifa::model()->findAll(array("condition" => " id =
'$id'"));
        $this->render("mor" , array("model" => $model));
    }

```

```

public function actionList()
{
    $model = Ypersonal::model()->findAll();
    $this->render("list",array("model" => $model));
}

public function actionForm()
{
    $model = Ypersonal::model()->findAll();
    $this->render("form" , array("model" => $model));
}

public function actionInfo($id = false)
{
    $model = Ypersonal::model()->find(array("condition" => "id =
'Sid'"));
    $this->render("info" , array("model" => $model));
}

public function actionSearch()
{
    $fn = $_GET['fname'];
    $ln = $_GET['lname'];
    $ty = $_GET['type'];
    echo $fn." ".$ln." ".$ty;

    if(!empty($ty) and !empty($fn) AND !empty($ln))
    {
        $model = Ypersonal::model()->findAll
        (
            array("condition" => "name = '$fn' and surname = '$ln' and
type_user = '$ty'")
        );
    }
    if(!empty($ty) and empty($fn) and empty($ln))
    {
        echo "asas".$ty;
        $model = Ypersonal::model()->findAll(array("condition"=>"type_user =
'$ty'"));
    }
    if(empty($ty) and !empty($fn) and empty($ln))
    {

```

```

        $model = Ypersonal::model()->findAll(array("condition"=>"name =
'$fn'"));
    }
    if(empty($ty) and empty($fn) and !empty($ln))
    {
        $model = Ypersonal::model()-
>findAll(array("condition"=>"surname = '$ln'"));
    }
    if(empty($ty) and empty($fn) and empty($ln))
    {
        $model = Ypersonal::model()->findAll();
    }
    if(!empty($ty) and empty($fn) and !empty($ln))
    {
        $model = Ypersonal::model()-
>findAll(array("condition"=>"type_user='$ty' and surname='$ln' "));
    }
    if(!empty($ty) and !empty($fn) and empty($ln))
    {
        $model = Ypersonal::model()-
>findAll(array("condition"=>"type_user='$ty' and name='$fn' "));
    }
    if(empty($ty) and !empty($fn) and !empty($ln))
    {
        $model = Ypersonal::model()->findAll(array("condition"=>"name='$fn'
and surname='$ln' "));
    }
    $this->render("form",array("model" => $model));
}

public function actionFull_info( $id = false)
{
    $model = Ypersonal::model()->find(array("condition" => "id =
'$id'"));
    $this->render("full_info" , array("model" => $model));
}
public function actionFull( $id = false)
{
    $model = Ypersonal::model()->find(array("condition" => "id =
'$id'"));
    $this->render("info" , array("model" => $model));
}

public function actionSendMsg()

```

```

{
    $from = $_POST['email'];
    $msg = $_POST['msg'];

    if(mail("zero_tmi@mail.ru","Salom",$msg,$from))
        echo json_encode(array('success'=>'true'));
    else
        echo json_encode(array('success'=>'false'));
}

public function actionError()
{
    if($error=Yii::app()->errorHandler->error)
    {
        if(Yii::app()->request->isAjaxRequest)
            echo $error['message'];
        else
            $this->render('error', $error);
    }
}

public function actionRegistration () {

    $model = new Ypersonal;
    if(isset($_POST['Ypersonal']))
    {
        $model->attributes = $_POST['Ypersonal'];
        if($model->save())
            $this->redirect(array('site/index'));
    }

    $this->render('registration',array(
        'model'=>$model
    ));

}

public function actionLogin()
{
    $model=new LoginForm;

    if(isset($_POST['ajax']) && $_POST['ajax']==='login-form')
    {

```

```

        echo CActiveForm::validate($model);
        Yii::app()->end();
    }

    if(isset($_POST['LoginForm']))
    {
        $model->attributes=$_POST['LoginForm'];
        if($model->validate() && $model->login())
            $this->redirect(Yii::app()->user->returnUrl);
    }
    $this->render('login',array('model'=>$model));
}

public function actionRegion(){
    $country = $_POST['country'];
    $tip = $_POST['tip'];
    if($tip == 1){
        $model = Xwbregion::model()->findAll('countryid='.$country);
        echo "<option value=\">region tanlang</option>";
    }
    else {
        $model = Xwbcity::model()->findAll('regionid='.$country);
        echo "<option value=\">shahar tanlang</option>";
    }

    foreach($model as $data) {
        echo "<option value='{ $data->id }'>$data->name</option>";
    }
}

public function actionLogout()
{
    Yii::app()->user->logout();
    $this->redirect(Yii::app()->homeUrl);
}
}

```

To'liq ilova malakaviy ishning so'nggi varog'idagi CD diskda keltirilgan.