**MUNDАRIJА**

[KIRISH 2](#_Toc138446770)

[I.Tizimli tahlil 4](#_Toc138446771)

[**1.1 Fermer xo'jaliklariga xizmat ko'rsatuvchi korxona uchun web ilova tahlili.** 4](#_Toc138446772)

[**1.2 Web sahifalarni yaratish usul va texnologiyalari** 6](#_Toc138446773)

[**1.3 Masalaning qo’yilishi** 8](#_Toc138446774)

[II.Asosiy qism. 9](#_Toc138446775)

[**2.1 Web ilova yaratish uchun dasturlash muhitini sozlash** 9](#_Toc138446776)

[**2.2 Fermer xo'jaliklariga xizmat ko'rsatuvchi korxona uchun web** 13](#_Toc138446777)

[**ilovani yaratish. PHP haqida ma’lumot.** 13](#_Toc138446778)

[**2.3 Foydalanuvchi uchun yo’riqnoma** 17](#_Toc138446779)

[Xulosa 24](#_Toc138446780)

[Foydalanilgan internet resurslari 25](#_Toc138446781)

[Ilova 26](#_Toc138446782)

**KIRISH**

Hozirda Kompyuter va axborot texnologiyalari jadal sur’atlar bilan yangilanib, rivojlanishi bilan birga kundalik turmushimizning asosiga aylanib bormoqda.

Hozirgi kunda zamоnaviy talablarni hisоbga оlgan hоlda ma`lumоtlar tarmоg’ini tashkil qilishni takоmillashtirish, o`sib kelayotgan yosh avlоdning intellektual ehtiyojlarini qоndirishga, madaniy, ma`naviy-axlоqiy qadriyatlarni saqlab qоlishga yo`naltirilgan printsipial yangi axbоrоt markazlarini barpо etish, shuningdek, ahоlini yanada kengrоq va tizimli axbоrоt bilam ta`minlash uchun zarur shart-sharоitlar yaratish maqsadida O`zbekistоn Respublikasida ta`lim sоhasida quyidagilarni amalga оshirish ko`zda tutilgan:

* ta`lim muassasalari o`quvchilari hamda axоlining zamоnaviy axbоrоt texnоlоgiyalaridan fоydalangan hоlda muntazam ta`lim оlishi va mustaqil ravishda ta`lim оlishiga ko`maklashish;
* yangi axbоrоt texnоlоgiyalari
* asоsida ahоliga axbоrоt xizmati ko`rsatish;
* madaniy, ta`lim, axbоrоt hamda bоshqa dastur va lоyihalarni birgalikda amalga оshirish uchun ta`lim muassasalari, mahalliy o`zini o`zi bоshqarish о`rganlari, milliy madaniyat markazlari bilan xamkоrlikni rivоjlantirish.
* Respublikamizda ta`lim tizimida tub islоhatlar оlib bоrilmоqda. Shulardan yana biri masоfadan turib o`qitish sistemasining jоriy qilinishi. Bunda Respublikamiz оliygоhlarida malakali prоfessоrlarning ma`ruzalarini tinglash, bevоsita muloqat o`rnatish mumkin.
* Masоfaviy ta`lim sistemasining ishlab chiqilishi keyingi yillarda ta`lim xizmatiga bo`lgan talabning keskin o`sishi tufayli hamda uning ijtimоiy ahamiyatiga asоslangan hоlda sоdir bo`ladi. O`zbekistоnda yagоna masоfaviy ta`lim sistemasining tashkil etilishi va faоliyat ko`rsatishi Respublikamizda zarur bo`lgan kadrlar, pedagоgik, ilmiy texnik va ilmiy metоdik pоtentsiallarning mavjudligi tufayli bo`lsa kerak. Bundan tashqari masоfaviy ta`limning yangi texnоlоgiyalari haqida ko`plab tajriba yig`ilgan va ko`plab ishlar amalga оshirilgan.
* Ta`lim sоhasida elektrоn darslik va qo`llanmalarni yaratish uchun quyidagi muxim masalalarni hal qilish kerak bo`ladi.
* Internet texnоlоgiyasi bilan tanishib chiqish ;
* prоgramma vоsitalarini o`rganish, masalan Web -saytlarni yaratish va qayta ishlash bo`yicha;
* Web - saxifalarda har-xil infоrmatsiyalarni tasvirlash uchun usullar va imkоniyatlarni o`rganish va qo`llash;
* Web - saytlarni yaratish va qayta ishlashga dоir tavsiyalar va asоsiy qоidalar bilan tanishib chiqish;
* Web - saxifani strukturasini aniqlash;
* Web - saytni qayta ishlash va yaratish uchun maqsadni tanlash;
* WEB saxifalarini o`qish vоsitalari .

HTML va bоshqa prоgramma vоsitalari yordamida tayyorlangan Web saxifalarida fоydalanuvchiga tushunarli ko`rinishda tasvirlash uchun maxsus prоgrammalar ishlab chiqilgan bo`lib, bunday prоgrammalar brauzer prоgrammalar deb ataladi. Hоzirda bir necha shunday prоgrammalar ishlab chiqilgan bo`lib, ular tabiiy ravishda hujjatlarni ko`rishni turlicha tahrir qiladilar. Bular оrasida keng tarqalgan Microsoft Internet Explorer va Netscape Navigator prоgrammalaridir. Shunday qilib, brauzerning asоsiy vazifasi URL adreslarda jоylashgan Web saxifalarini kоmpyuterga yuklash va uni fоydalanuvchiga tushunarli ko`rinishda mоnitоr ekranida ko`rsatib berishdir.

# I.Tizimli tahlil

## **1.1 Fermer xo'jaliklariga xizmat ko'rsatuvchi korxona uchun web ilova tahlili.**

Bugungi kunda aksariyat kompaniyalar soliq va buxgalteriya yozuvlarini 1C: Buxgalteriya dasturi yordamida yuritadilar. Bu har qanday murakkablikdagi jarayonlarni avtomatlashtirishga imkon beradigan tayyor echim. "1C: Buxgalteriya 8" dasturidan foydalanish uchun maxsus texnik bilimlar talab etilmaydi: aniq interfeys, batafsil ko'rsatmalar va yaxshi tashkil etilgan texnik yordam rivojlanishni ancha soddalashtiradi.

Ish yuritish uchun "1C: Buxgalteriya" tizimidan foydalanish korxona faoliyatining turidan qat'i nazar samarali bo'ladi. Bu sizga qonunchilikdagi o'zgarishlar to'g'risida ma'lumotni o'z vaqtida olish, hisobotlarni o'z vaqtida va aniq tarzda tuzish imkonini beradi. Dasturning ishlashi uchun buxgalteriya hisobi uchun umumiy konfiguratsiya parametrlariga ega bo'lgan "1C: Enterprise" platformasi ishlatiladi.

**Ijobiy va salbiy tomonlari**

Dasturning afzalliklari, agar siz 1C: Buxgalteriya dasturini yuklab olib, kompyuterga o'rnatib qo'ysangiz aniq bo'ladi.

* + buxgalteriya hisobining barcha turlarini amalga oshirish imkoniyati;
  + ma'lum bir turdagi biznes sharoitlari uchun modifikatsiya qilish;
  + dasturni soliq kodeksidagi o'zgarishlar bilan yangilab, Rossiya qonunchiligiga to'liq mos keladi.

Kamchiliklari:

* + ba'zi bir aniq korxonalar samaradorligini oshirish uchun takomillashtirish zarurligi (to'liq universallik emas);
  + hisobotlarni qayta ishlashda noaniqliklarni topishning murakkab jarayoni;
  + shunga o'xshash dasturdan o'tishda ma'lumotlarning bir qismini uzatish mustaqil ravishda amalga oshirilishi kerak;
  + o'zlashtirish maxsus bilimlarni talab qiladi.

Siz 1C: Buxgalteriya 8.3-ni yuklab olishingiz mumkin. Dasturning ushbu takomillashtirilgan versiyasi bir nechta xususiyatlarga ega:

* + deklarativ foydalanuvchi interfeysi;
  + bir vaqtning o'zida bir nechta korxonalar bilan bitta ma'lumotlar bazasida ishlash;
  + standart echimlar doirasini kengaytirish;
  + yakka tartibdagi korxonalar bilan ishlash:
  + xarajatlarni hisobga olish funktsiyasini modernizatsiya qilish;
  + eng ko'p ishlatiladigan buxgalteriya yozuvlarini guruhga yozib qo'yish;
  + bir nechta soliqqa tortish sxemalarini hisobga olish.

1C: Accounting 8.3 dasturini bepul yuklab olish va kompyuterga o'rnatish uchun sizda 1C: Enterprise platformasi bo'lishi kerak.

## **1.2 Web sahifalarni yaratish usul va texnologiyalari**

Hozirgi axborot texnologiyalar rivojlanib borayotgan davrda butun jahon tarmog'idagi veb-sahifalar kundan kunga ko'payib bormoqda. Kimdir o'z shaxsiy sahifasini yaratsa, kompaniya va firmalar korporativ saytlarini yaratmoqda. Yana kimdir o'z kundaliklarini, ya'ni bloglarini olib boradilar.

Shu bilan birga, yirik portallar faoliyati ham rivojlanmoqda. Bularning hammasi esa biror dasturiy vosita yordamida yaratiladi. Kimdir HTML va CSS tilida, kimdir PHP da, yana kimlardir CMS deb ataluvchi tizimlar yordamida turli xil veb-saytlarni yaratmoqda.

1989-yilda Tim Berners-Li tomonidan World Wide Web, ya'ni Bununjahon o'rgimchak to'ri ixtiro qilinganidan boshlab uning har bir foydalanuvchisi oddiy bo'lsa-da, o'z veb-sahifasini yarata olish imkoniyatiga ega bo'ldi. Lekin, buning uchun HTML (Hyper Text Markup Language) tilini bilish shart edi. Turli xil IT-kompaniyalari tomonidan bu ishni osonlashtiruvchi bir qancha dasturiy paketlar, jumladan Adobe Dreamweaver va Microsoft FrontPage kabilar yaratildi. Bunday dasturlardan ba'zilari sayt yaratishning barcha bosqichlarini boshqarishni o'z zimmasiga oladi. Shu bilan birga, ko'pgina hollarda foydalanuvchi bu kodni bilishi shart bo'lmagan. Sababi bu dasturlarda veb-sahifalarni yaratishning vizual usullaridan foydalanilgan.

Sayt yangilanishi muammosini, ya'ni yangilanishning avtomatlashtirishini hal qilish maqsadida yirik axborot kompaniyalari, jumladan, Newsweek va Time yangi dasturiy ta'minot yaratish ishlarini boshlashga majbur bo'ladi. Bu dasturiy ta'minotlar kontentni (ma'lumotlarni) boshqarish tizimlari (CMS – Content Management System) deb ataldi.

**1. Dasturlash tillari va malumotlar bazasi.**

Hozirda dasturlash tillari ham ko'p, Eng mashxurlari, PHP, ASP, Python, Lekin juda ko'p saytlar PHP tilida qilingan. php haqida juda ko'p darsliklar bor. Dasturlash tillari bilan birgalikda ma’lumotlar bazasi bilan ishlanadi,asosan ularga MySQL va PostgreSQLlar kiradi.Ularda saytrning asosiy matnlar bazasi saqlanadi

**2. CMS –Tayyor boshqaruv tizimlari.**

CMS (Content Management System) bu tayyor qilingan sayt, Dasturlash va dizaynni bilmaydigan odam ham CMS yordamida chiroyli saytlar qilishi mumkin. Kod va dizaynlari ham tayyor qilingan bo'ladi. Siz uni faqat internetga joylash va ichini to'ldirishingiz kerak xolos. Hozirda mashxur CMS dasturlari bu Joomla, Wordpress, Drupal, hisoblanadi. Bundan tashqari PHP Framework dasturlari ham bor. Framework bu tayyor yozilgan kodlar. Masalan malumotlarni bazadan olish va uni ko'rsatish uchun PHP dasturinida bir necha qator kod yozish kerak, bu kod yozishda ham ancha bosh qotirasiz. Framework esa hamma kodlarni tayyor yozib qo'ygan, siz ularni bir biriga ulaysiz xolos.

**3. Hosting va Domen**

Hosting bu saytingizni internetda saqlaydigan server, Domen esa sayt nomi (masalan: www.saytnomi.uz) . Saytni internetga joylash, va hostingni moslash (htaccess fayllari yoki cron bilan ishlash) kabi bilimlar ham dastur ishlashimiz uchun nancha kerakdir.

**1.3 Masalaning qo’yilishi**

Hozir internet rivojlanib bormoqda. Dunyodagi hamma insonlar internet orqali ma’lumotlar almashishishadi, internet tarmog’idan juda ko’p maqsadlardan foydalaniladi. Hatto internet saytlari orqali xohlagan narsasiga buyurtma berib savdo sotiq ishlarini ham amalga oshirmoqdalar. Hozirda shunday turli saytlar ham mavjud. Intrnet tarmog’idagi veb-sahifalar kundan kunga ko’payib bormoqda. Bu holat O’zbekistonda ham jadallik bilan amalga oshmoqda. Kimdir o’z shaxsiy sahifasini yaratsa, o’quv muassasalari, kompaniya va firmalar korporativ saytlarini yaratmoqda. Yana kimdir o’z kundaliklarini ya’ni bloglarini olib boradilar.

Shu bilan birga, yirik portallar faoliyati ham rivojlanmoqda. Bularning hammasi esa biror dasturiy vosita yordamida yaratiladi. Kimdir HTML va CSS tilida, kimdir PHP da, yana kimlardir CMS deb ataluvchi tizimlar yordamida veb-saytlarni yaratmoqda.

**Ushbu kurs ishi esa “Fermer xo'jaliklariga xizmat ko'rsatuvchi korxona uchun” web ilovasini yaratishga mo'ljallangan bo'lib, saytni tuzish jarayonida quyidagi ishlar amalga oshirilishi lozim:**

1. Laravel da pdf export qilishni o’rganish;

2.Laravel da Fermer xo'jaliklariga xizmat ko'rsatuvchi korxona uchun web ilova arxitekturasi;

3. Laravel da Fermer xo'jaliklariga xizmat ko'rsatuvchi korxona uchun web ilovasini ishlab chiqish;

**Dasturda quydagi ishlar amalga oshirilishi lozim:**

- Web ilovani tushunarli qilish;

- Web ilovada pdf exportni qo‘shish;

- Web ilovaning tuzilishi sayt ko’rinishini ishlab chiqish;

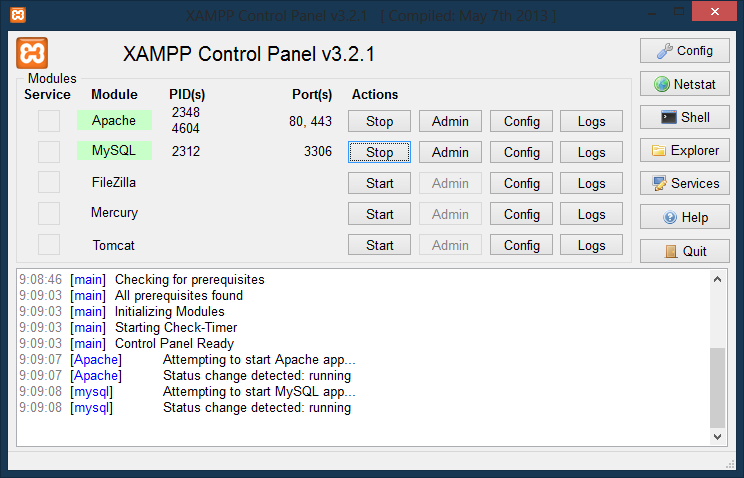
# II.Asosiy qism.

## **2.1 Web ilova yaratish uchun dasturlash muhitini sozlash**

Web-ilova - mijoz (jumladan foydalanuvchi) veb-brauzerida ishlaydigan mijoz-server dasturidir. Umumiy veb-ilovalar veb-pochtani, onlayn chakana savdoni, onlayn-auksionlarni, vikislarni, tezkor xabar xizmatlarini va boshqa ko'pgina funktsiyalarni o'z ichiga oladi.

Har qanday dinamik veb-sahifa va "veb-ilovalar" o'rtasidagi umumiy farq noaniq. Ko'proq "veb-ilovalar" deb ataladigan veb-saytlar ish stoli dasturiga yoki mobil ilovaga o'xshash funksiyalarga ega bo'lganlardir. HTML5 veb-sahifalar sifatida yuklangan ilovalarni yaratish uchun aniq tilni qo'llab-quvvatladi, lekin ma'lumotlarni mahalliy sifatida saqlashi va oflayn rejimda ishlashga davom etishi mumkin.

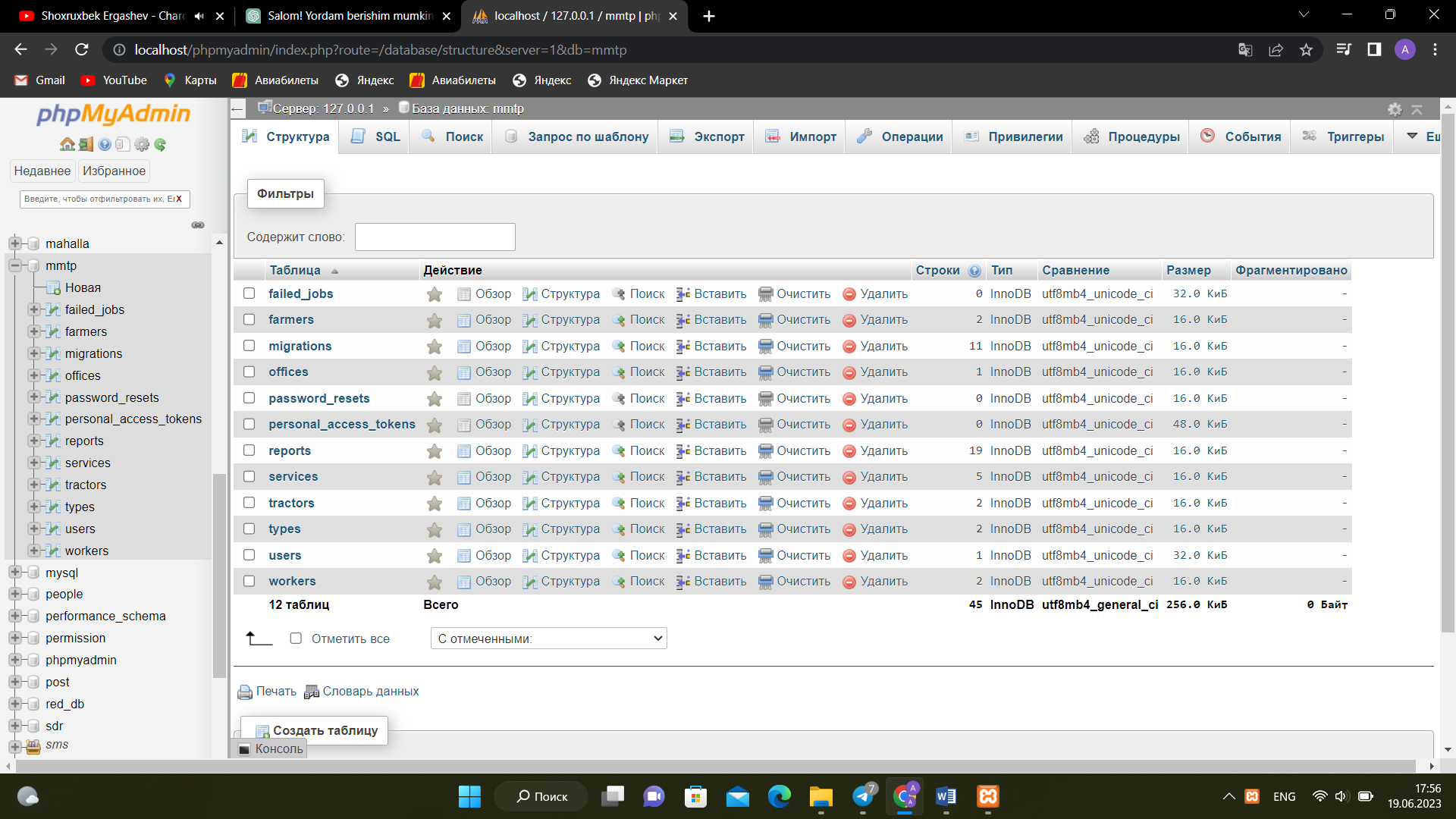
**Web sahifa** va **web saytlarga** xizmat ko`rsatish uchun mezbon kompyuterlar va serverlarda ishlatish uchun server dasturiy ta'minoti kerak bo`ladi. Web saytlar va bu saytlarni yaratish, boshqarish uchun mo`ljallangan dasturiy ta'minot mezbon kompyuterda joylashgan bo`ladi, mezbon kompyuterlar bog`lanish protokollarini boshqaradi. Mezbon kompyuterlar Unix, Windows yoki Macintosh kabi turli operatsion tizim ostida ishlaydilar. Bu operatsion tizimlarning ammasi tarkibiga TCP/IP protokolini qo`llab-quvvatlovchi dasturiy vositalar kiradi. Server dasturiy ta'minoti serverlarda joylashgan bo`lib, klient kompyuteridagi web brauzer talablariga ko`ra so`ralgan web sahifalarga xizmat ko`rsatadi va boshqa amallarni bajaradi. Server dasturiy ta'minoti TCP/IP bog`lanishlar uchun javob bermaydi. Buni serverdagi operatsion tizim bajaradi. Lekin server dasturiy ta'minoti operatsion tizim yordamida HTTP bog`lanishlar va talablarini qo`lga kiritadi va ularni qayta ishlaydi. Server dasturiy ta'minotining har xil turlari, masalan ma'lumotlar ombori serveri, FTP serveri, tarmoq serverlari mavjud. Xususan, web server HTTP serverining bir turi bo`lib, uning vazifasiga klient kompyuterlariga HTTP protokoli yordamida turli ma'lumotlarni jo`natish kiradi. Odatda, klient kompyuteridagi web brauzer serverga HTML xujjatni jo`natish kabi talablarni qo`yadi. Server bunday talablarni olgach, unga javob beradi. Bu javobning boshida uzatish haqida ma'lumot, qolgan qismi HTML fayldan iborat bo`ladi.  
Albatta, web serverlar sahifalarni web brauzerga uzatishdan boshqa ishlarni ham qiladi. U web sahifadagi CGI ssenariylarini mos CGI ilovalariga uzatadi. Bu ssenariylar tashqi mini-dasturlarni ishga tushiradi. Zarurat bo`lganda, bu dasturlar ishining natijalarini web server orqaga web brauzerga qaytaradi. Bulardan tashqari, web serverlar tarkibiga konfiguratsiya fayllari va utilitlar kiradi, ular web saytlarni turli usullarda himoya qilish va boshqarish imkonini beradilar.



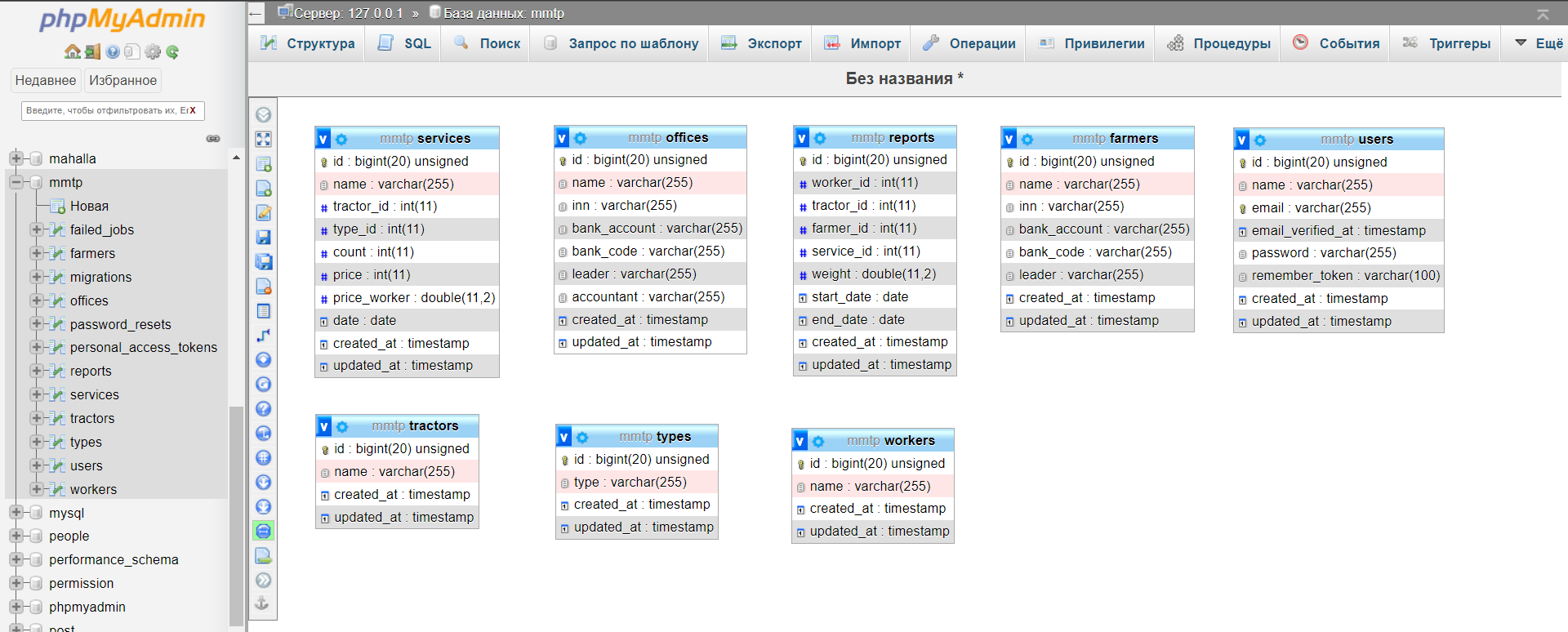
1.1 –rasm. ''Xampp Control Panel''

Bizning ma`lumotlar bazamizni jadvallarini PhpMyadmin dasturi orqali “two\_factor\_auth” deb nomlangan ma’lumotlar bazasini yaratib olamiz.

Ma’lumotlar bazamizni quyidagicha yaratiladi:

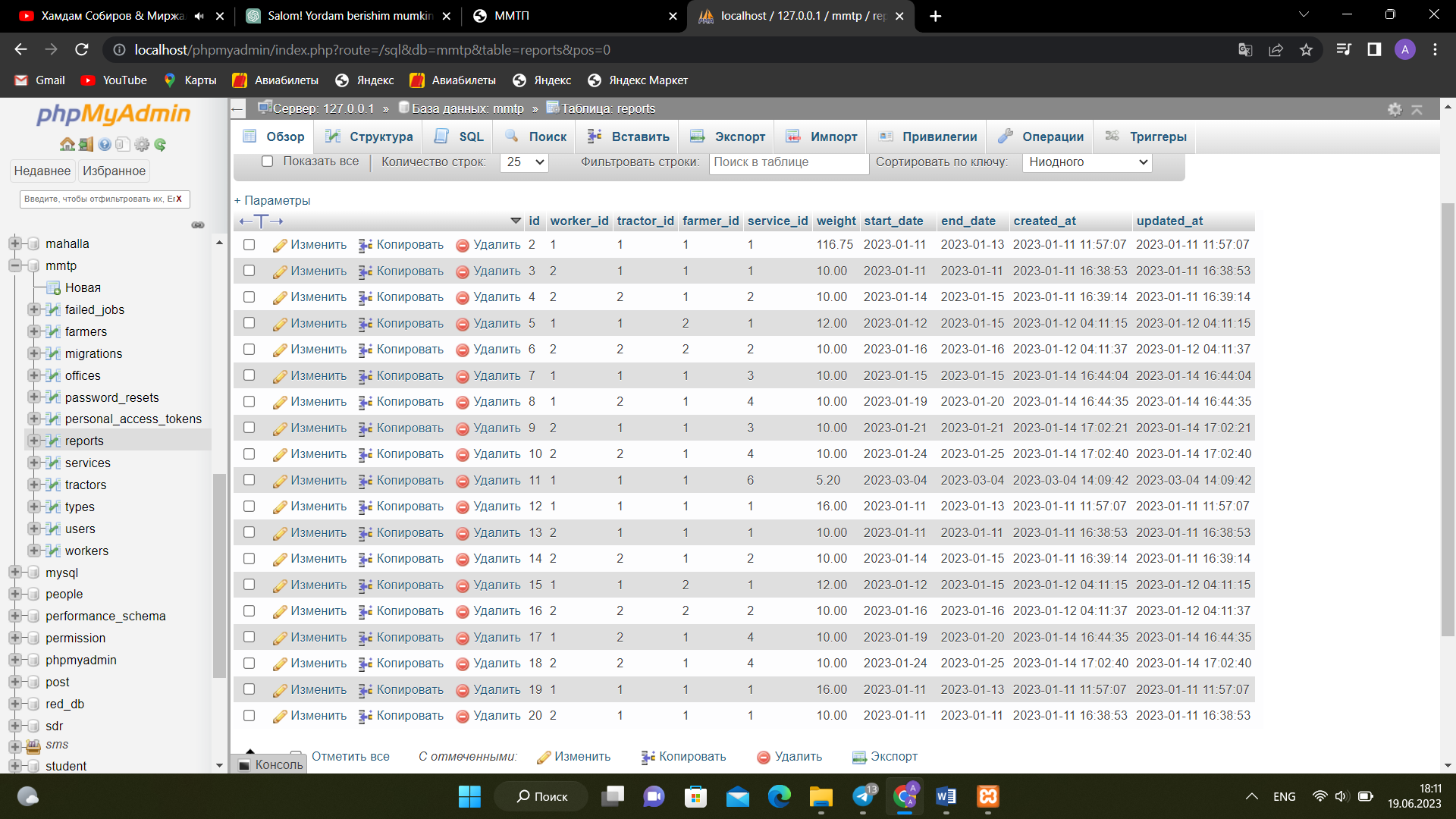


1.2 - rasm Ma’lumotlar bazasi



1.3 – rasm Ma’lumotlar bazasi Strukturasi

Brauzerga kirib phpmyadminni chaqirib ma’lumotlar bazasini ko’rishimiz mumkin.



1.4 – rasm reports jadvalining ma’lumotlar bazasi

**2.2 Fermer xo'jaliklariga xizmat ko'rsatuvchi korxona uchun web**

**ilovani yaratish. PHP haqida ma’lumot.**

PHP (Hypertext Preprocessor) — eng ko‘p tarqalgan dasturlash tillaridan biri bo‘lib, Web — dasturlar yaratishda ishlatiladi. Hozirda PHP juda ko‘p hosting- provayderlar tomonidan foydalaniladi, bu esa uni harqanday internet- proyektlarni (oddiygina saytdan tortib, to yirik portallargacha) yaratishda ishlatilatiladigan, sal kam, eng asosiy tilga aylantirdi. PHP da juda ko‘p miqdorda alohida skriptlar, shuningdek, forum, kontent tizimi boshqaruvi kabi tugallangan proyektlar yozilgan. PHP tarixi ... 1994 yili php tilinig yaratuvchisi Rasmus Lerdorf o ’zinig saytiga mehmonlar kirishini hisoblash uchun Perl/ CGI (skriptlar to‘plami) da maxsus qobiq yozib, amalda qo’lladi va uni «Personal Home Page»(PHP nomi shundan kelib chiqqan) deya nomladi.

**PHP tilida dasturlash asoslari** PHP dasturlash tili tilida tuzilgan har qanday dastur **php** kengaytmali fayllarda saqlanadi.

Misol: uchun **functions.php**, **index.php**, **admin.php** kabi. Birnchi misolni ko'ramaiz. **test.php** nomli fayl tashkil etamiz va unga quyidagi kodlarni yozamiz:

|  |
| --- |
| <html>  <head>  <title>HTML hujjat</title>  <head>  <body>  Hujjat tanasi  </body>  </html> |

Bu matn HTML asosida yozilgan. 1-misoldan ko'rinib turibdiki php hujjat HTML hujjat kabi shakllantirilmoqda. Lekin php hujjatda biz qo'shimcha imkoniyatlarga ega bo'lamiz.**test.php** faylga quyidagi o'zgarishlarni kiritamiz:

|  |
| --- |
| <?php  Echo "<html>";  Echo "<head>";  Echo "<title>HTML hujjat</title>";  Echo "<head>";  Echo "<body>";  Echo "Bu matn HTML asosida yozilgan";  Echo "</body>";  Echo "</html>";  ?> |

 2-misol ham 1-si kabi hujjat hosil qiladi. Farqi shundaki ikkinchi holda HTML teglarni PHP dasturi yordamida hosil qilinmoqda. Lekin ko'p hollarda HTML teglari butunicha PHP yordamida hosil qilinmay ora-oralarda PHP kodlari yoziladi:

|  |
| --- |
| <?php  $title = “HTML hujjat”;  $text = “Bu matn PHP asosida yaratilgan”;  ? >  <html>  <head>  <title><?php echo $title; ?></title>  </head>  <body>  <?php echo $text; ?>  </body>  </html> |

Bu yerda PHP kod HTML teglarni generatsiya qilyapti. Demak PHP dasturining vazifalaridan biri HTML teglarini generatsiya qilish ekanini bilib oldik. PHP ning bundan tashqari yana ko'plab imkoniyatlari mavjud bo'lib unga ma'lumotlar bazasiga murojaat qilish, turli hisob-kitoblarni amalga oshirish kabilar ham kiradi. Lekin PHPdan ko'pchilik aynan klent kompyuterida emas, balki serverda bajarilishi uchun foydalanadi. PHP dasuri tuzish haqida shu vaqtgacha bilib olgnlarimizni umumlashtiramiz:

1) \*.php kengaytmali fayl yaratiladi;

2) faylda istalgancha HTML teglarini ishlatish mumkin;

3) PHP kodlarini ochish va yopish belgilari orasida yoziladi:

a) <?php –ochish belgizi va ?> -yopish belgisi;

b) <? –ochish belgisi va ?> - yopish belgisi;

c) <script language = "php"> -ochish belgisi va </script> - yopish belgisi;

d) <?= -ochish belgisi ba ?> yopish belgisi.

PHP kodlarini yozish bo'yicha yana bitta misol ko'ramiz:

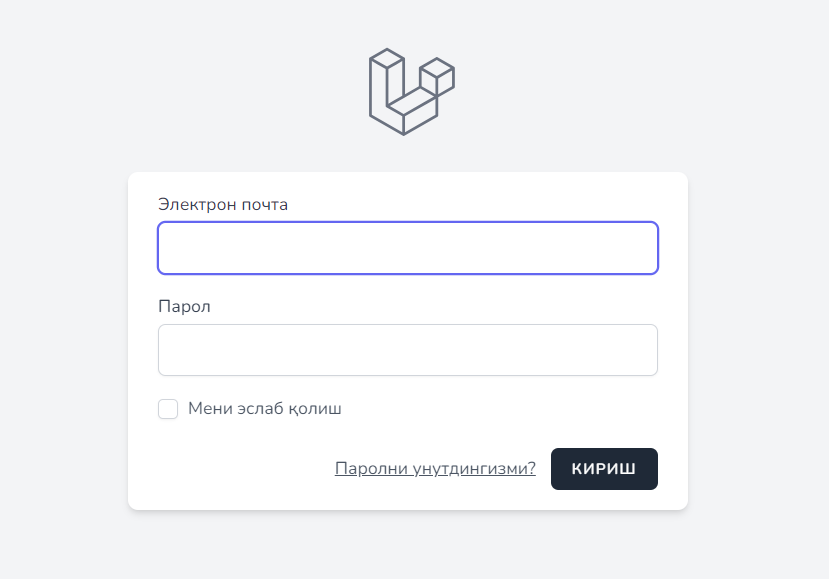
|  |
| --- |
| <?php // birinchi usuldan foydalanilmoqda  $title = “HTML hujjat”;  $ text = “Bu matn PHP asosida yaratilgan”;  ? >  <html>  <head>  < title>  <? // ikkinchi usuldan foydalanilmoqda  eecho $title;  ?>  </title>  </head>  <body>  <script language = “php”> // uchunchi usuldan foydalanilmoqda  Echo $text;  </script>  </body>  </html> |

Universal usul sifatida birinchi usulni aytish mumkin. Serverdagi PHP ning sozlamasiga ko'ra ikkinchi va to'rtinchi usullarni ham qo'llash PHP kodlarini yozishda yengilliklar yaratadi. Uchunchi usulni dasturchilar kamdan-kam foydalanishdi. Bu kurs ishi ya’ni “Fermer xo'jaliklariga xizmat ko'rsatuvchi korxona” uchun web ilovasini yaratishda Laravel framework foydalangan holda yaratiladi.

Laravel - universal framework va uni barcha turdagi web ilovalar uchun qo’llash mumkin. Uning kompanentalar strukturasi va keshlanishi zo’r qo’llab quvvatlanishi evaziga framework asosan portlar, forumlar, CMS, magazinlar yoki REST ful ilovalar kabi proektlar uchun katta qo’l keladi. Bu kurs ishini bazasini yaratganimizdan keyin, biz bazani PHP ga bog’lashda Laravel frameworkidagi CRUD dan foydalanamiz.

## **2.3 Foydalanuvchi uchun yo’riqnoma**

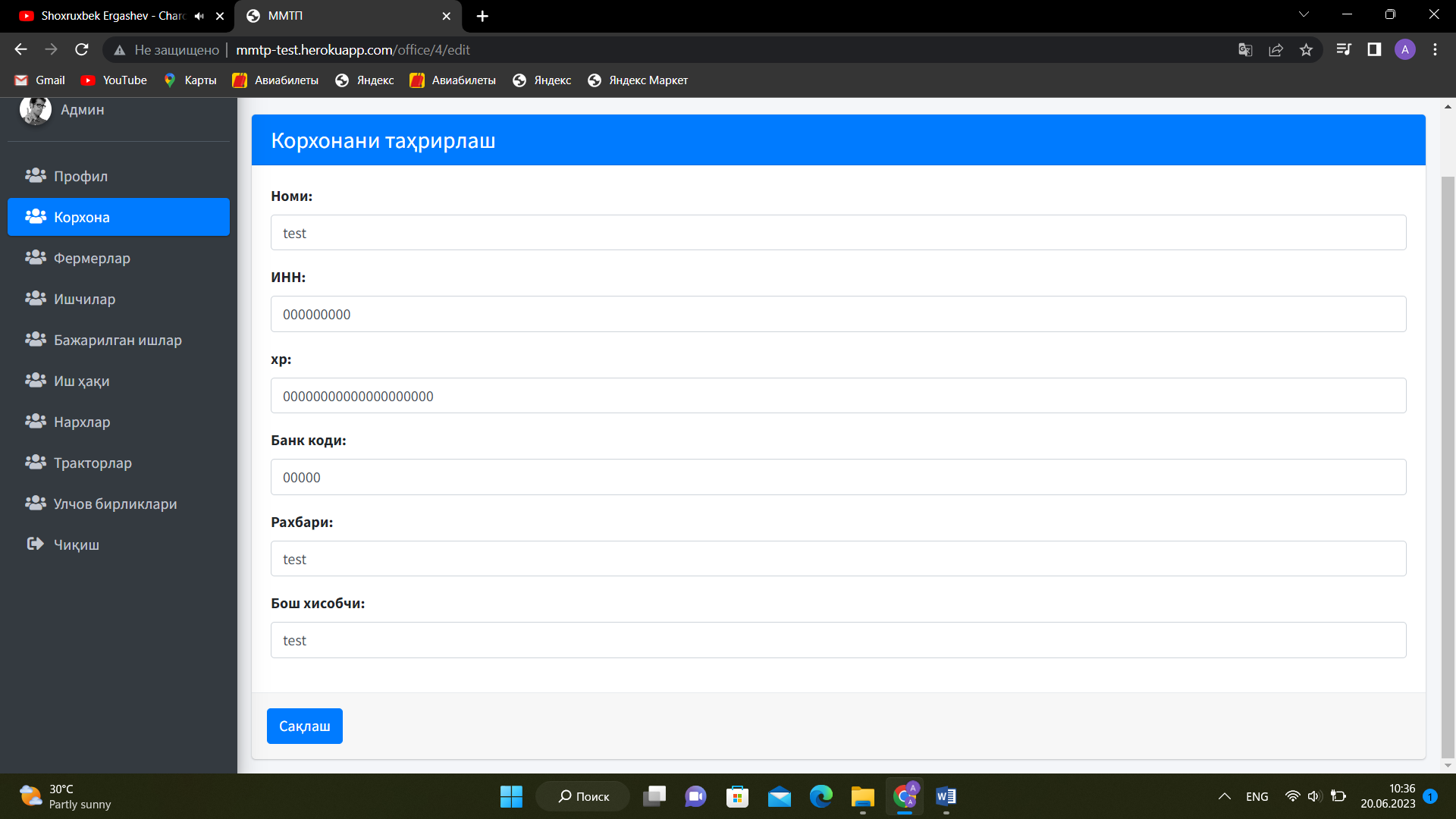
Web ilovamizni kirish oynasi ko’rinishi quyida ko`rsatilgan:



1.5 - rasm “Web ilovamizni kirish oynasi ko’rinishi”

Dasturning kirish qismida foydalanuvchi o‘zining email va parolidan foydalangan holda tizimga kirishi mumkin.

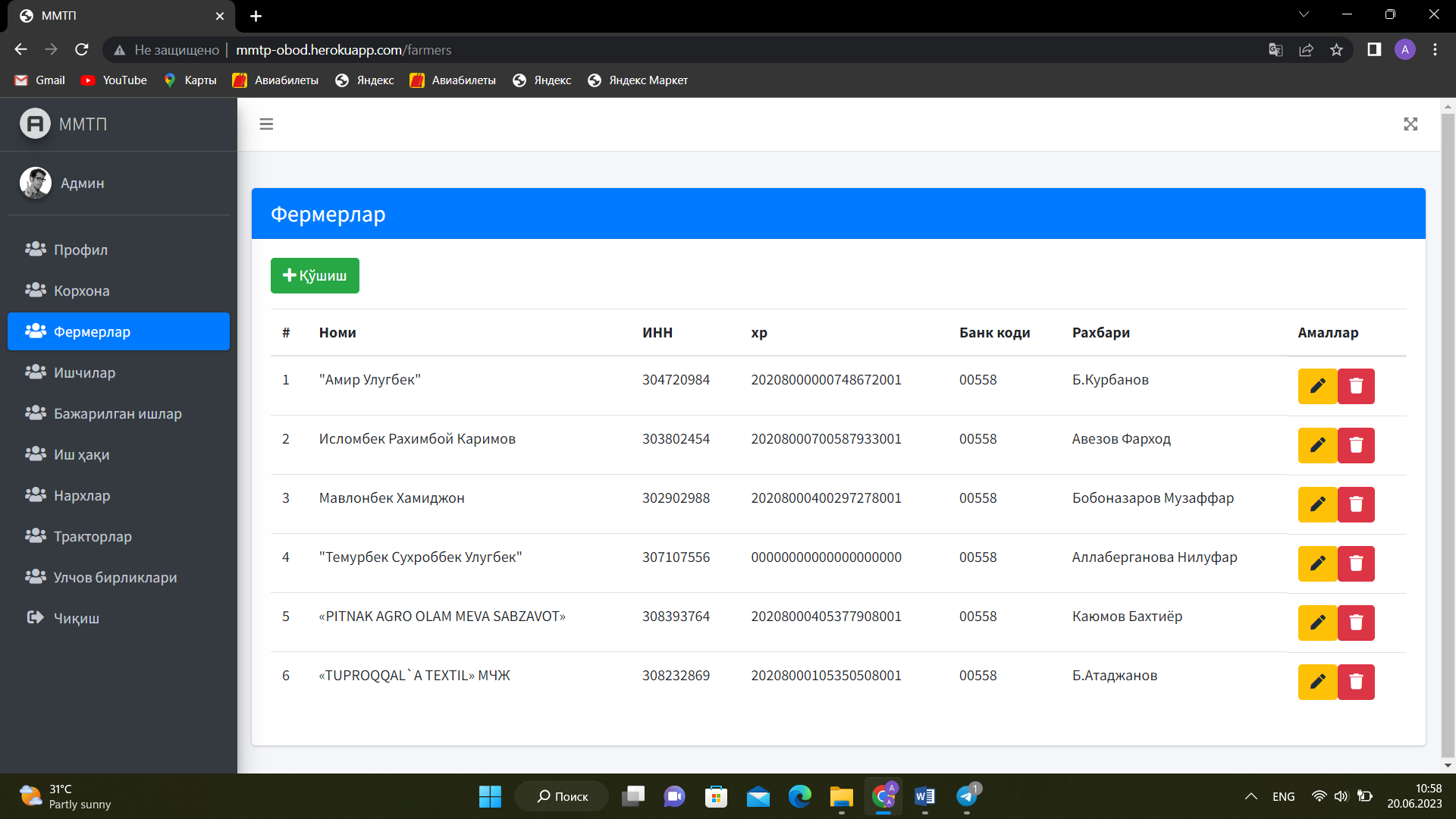
Web ilovamizni korxona ma’lumotlari oynasi ko’rinishi quyida ko`rsatilgan:



1.6 - rasm “Web ilovamizni korxona ma’lumotlarini oynasi ko’rinishi”

Dasturning korxona qismida korxona ma’lumotlarini yangilash mumkin. Korxona ma’lumotlari hujjatlarni yaratish paytida foydalaniladi.

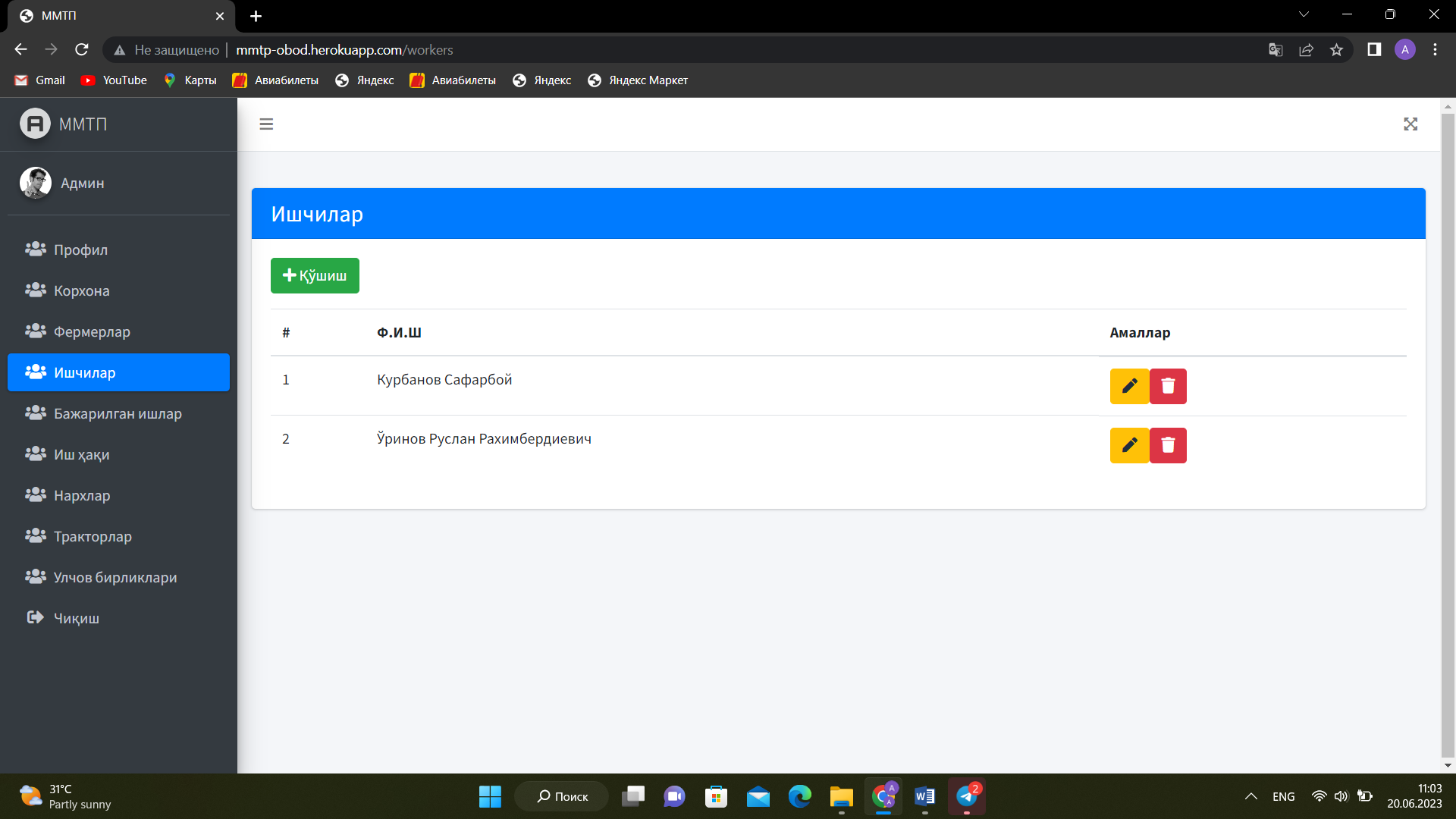
Web ilovamizni fermerlar oynasi ko’rinishi quyida ko`rsatilgan:



1.7 - rasm “Web ilovamizni fermerlar oynasi ko’rinishi”

Dasturning fermerlar qismida fermerlar ro‘yxati ko‘rishimiz, yangi fermerni yaratish, mavjud fermer ma’lumotlarni yangilash, fermer ma’lumotlarini o‘chirish mumkin.

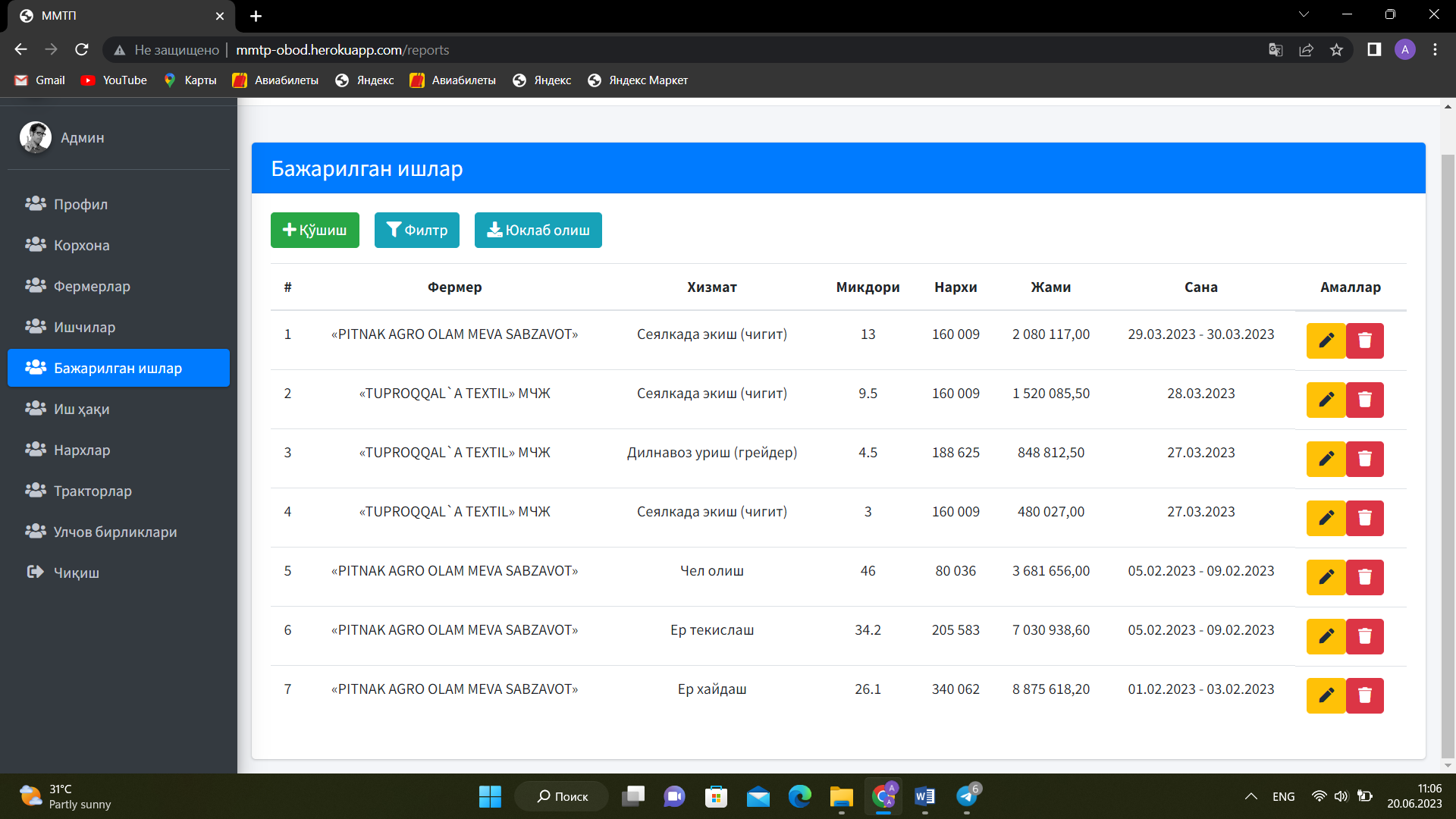
Web ilovamizni ishchilar oynasi ko’rinishi quyida ko`rsatilgan



1.8 - rasm “Web ilovamizni ishchilar oynasi ko’rinishi”

Dasturning ishchilar qismida ishchilar ro‘yxati ko‘rishimiz, yangi ishchini yaratish, mavjud ishchi ma’lumotlarni yangilash, ishchi ma’lumotlarini o‘chirish mumkin.

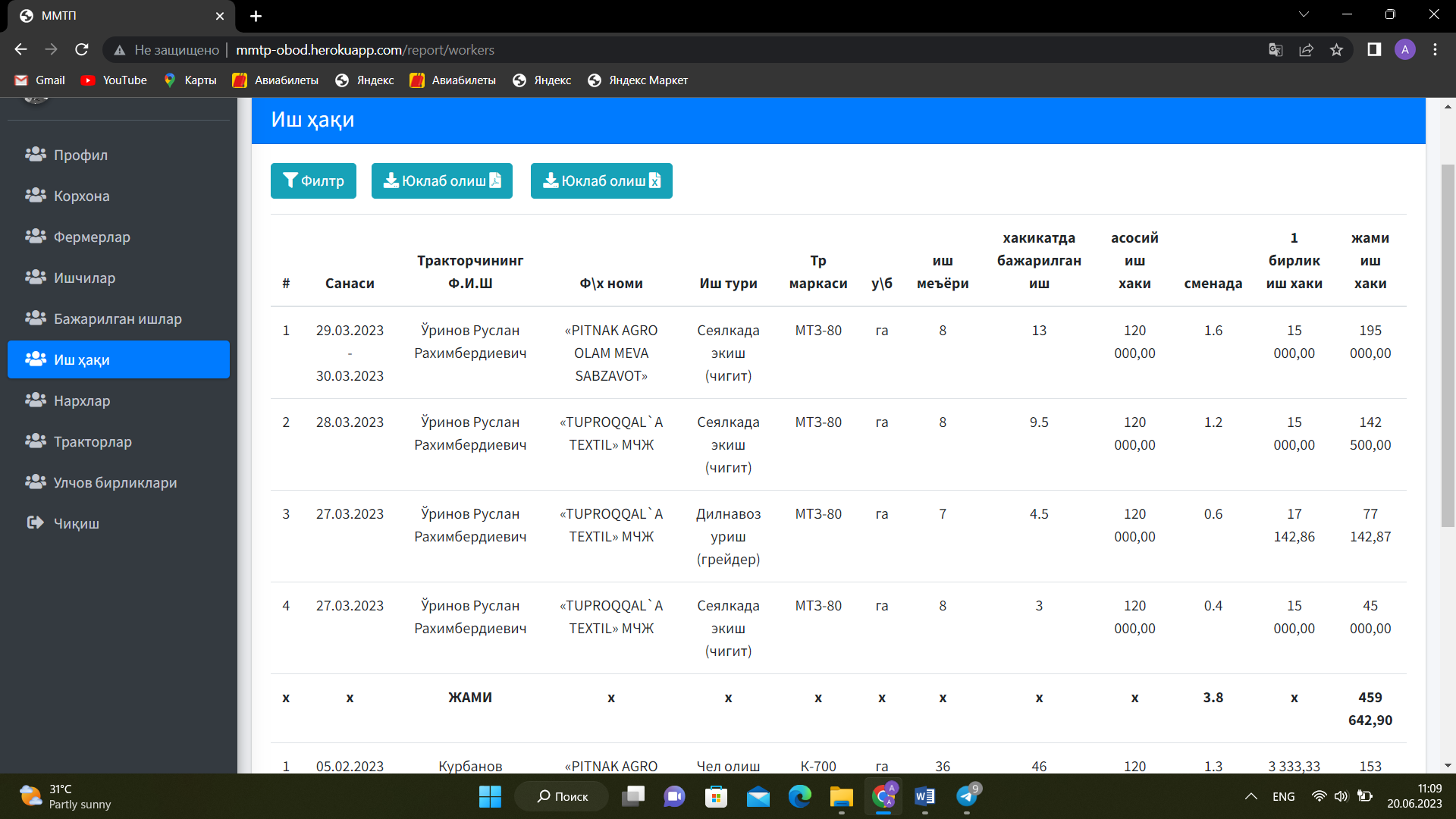
Web ilovamizni bajarilgan ishlar oynasi ko’rinishi quyida ko`rsatilgan



1.9 - rasm “Web ilovamizni bajarilgan ishlar oynasi ko’rinishi”

Dasturning bajarilgan ishlar qismida ishlar ro‘yxati ko‘rishimiz, yangi ishni yaratish, mavjud ishni yangilash, ishni o‘chirish va fermerga ko‘rsatilgan xizmatlar bo‘yicha shartnomani yuklab olishimiz mumkin.

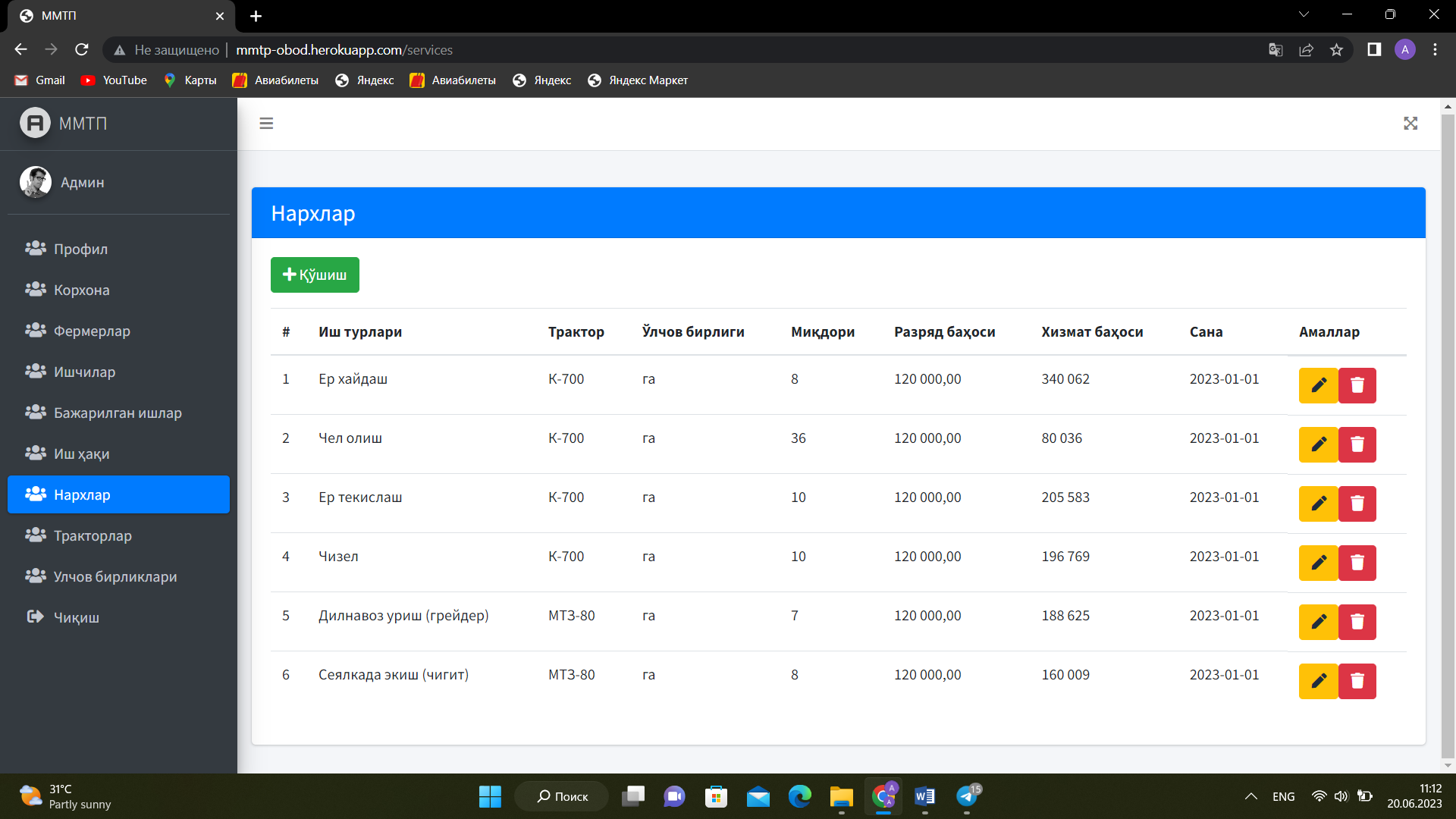
Web ilovamizni ish haqi oynasi ko’rinishi quyida ko`rsatilgan



2.0 - rasm “Web ilovamizni ish haqi oynasi ko’rinishi”

Dasturning ish haqi qismida ichilar qilgan ishlari uchun ish haqini hisoblangan jadvalni yuklab olishimiz mumkin.

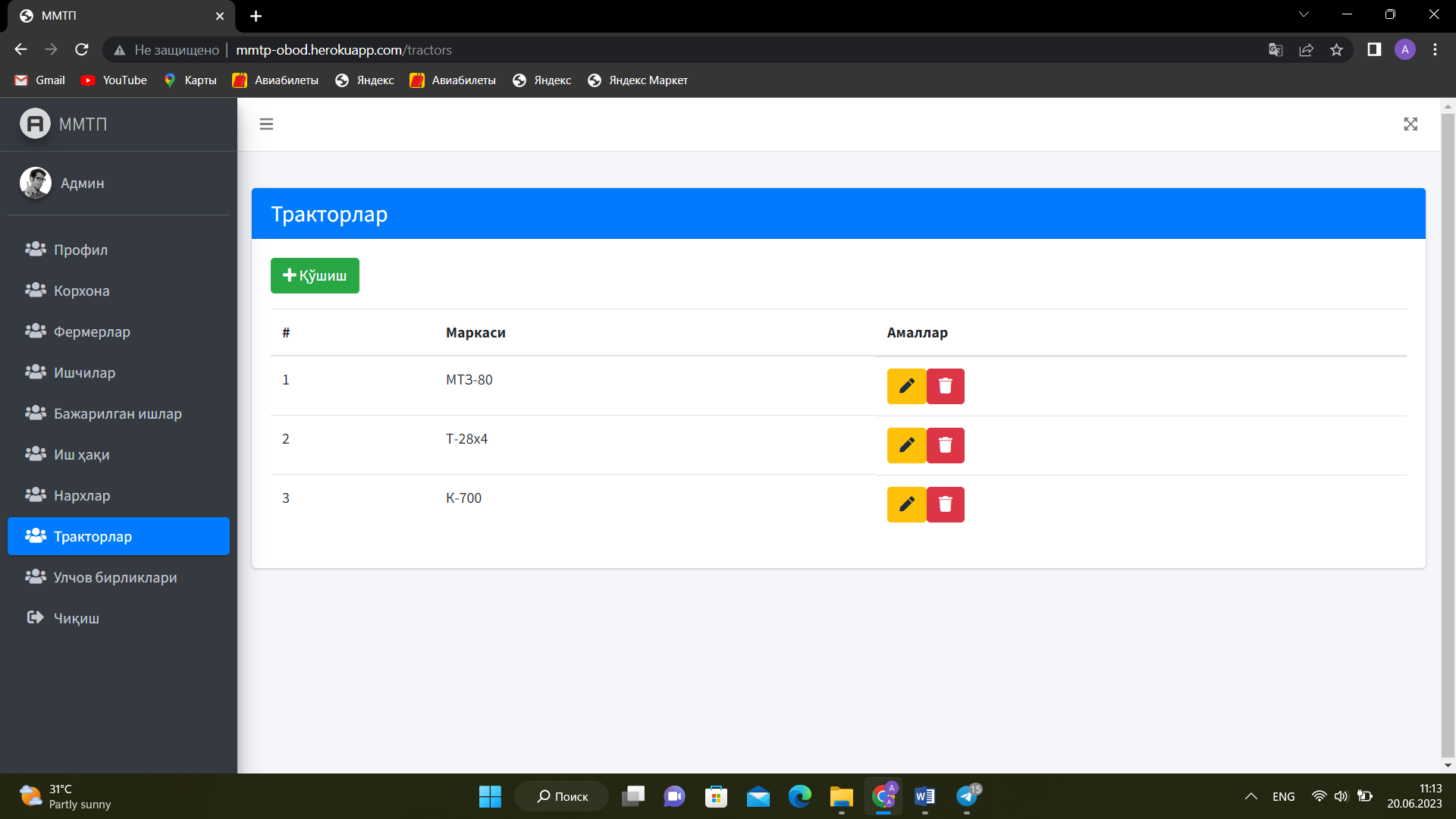
Web ilovamizni narxlar oynasi ko’rinishi quyida ko`rsatilgan



2.0 - rasm “Web ilovamizni narxlar oynasi ko’rinishi”

Dasturning narxlar qismida xizmat ko‘rsatish narxlarini yaratish, tahrirlash va o‘chirish mumkin.

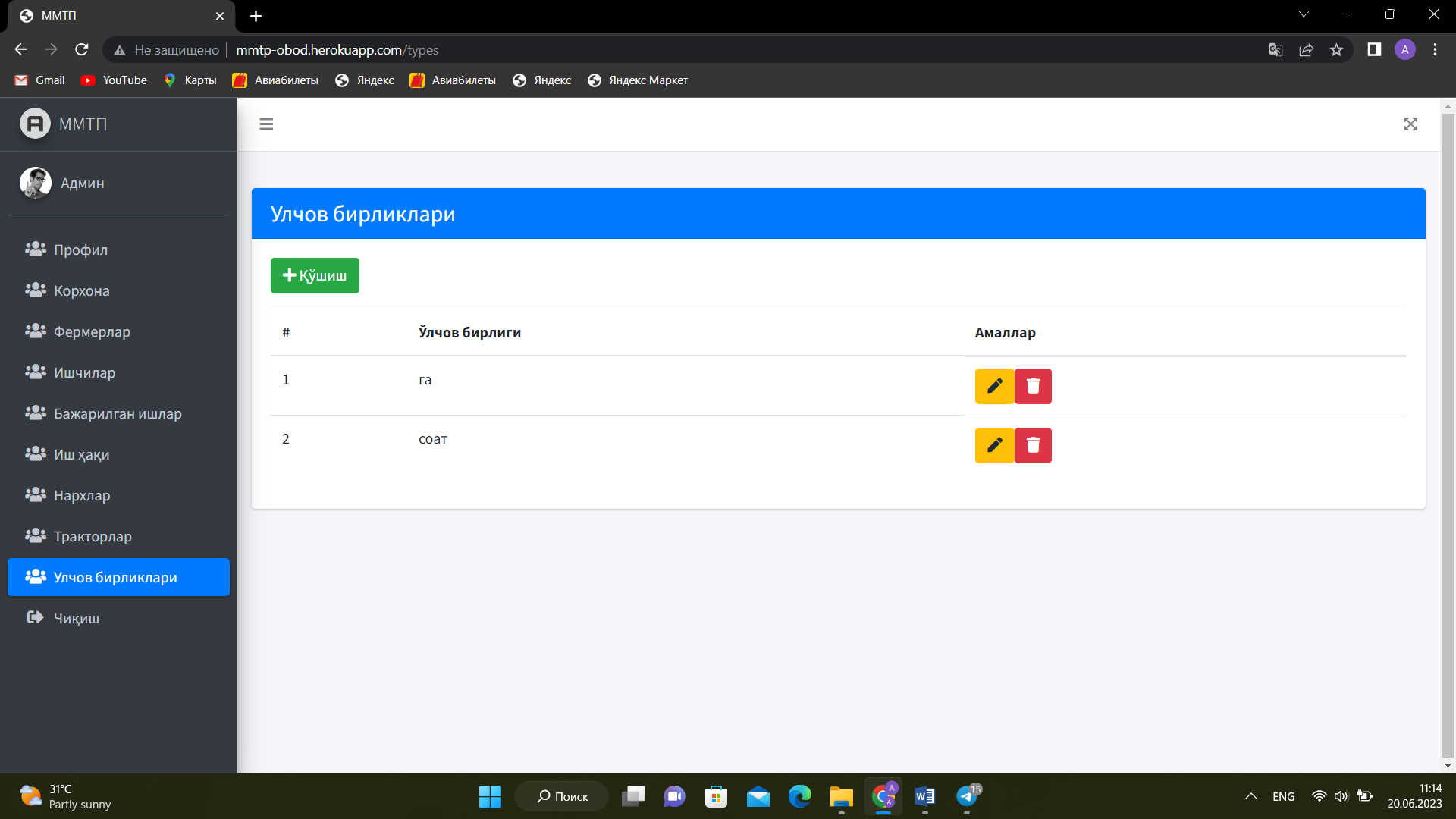
Web ilovamizni traktorlar oynasi ko’rinishi quyida ko`rsatilgan



2.1 - rasm “Web ilovamizni traktorlar oynasi ko’rinishi”

Dasturning traktorlar qismida traktorlarni yaratish, tahrirlash va o‘chirish mumkin.

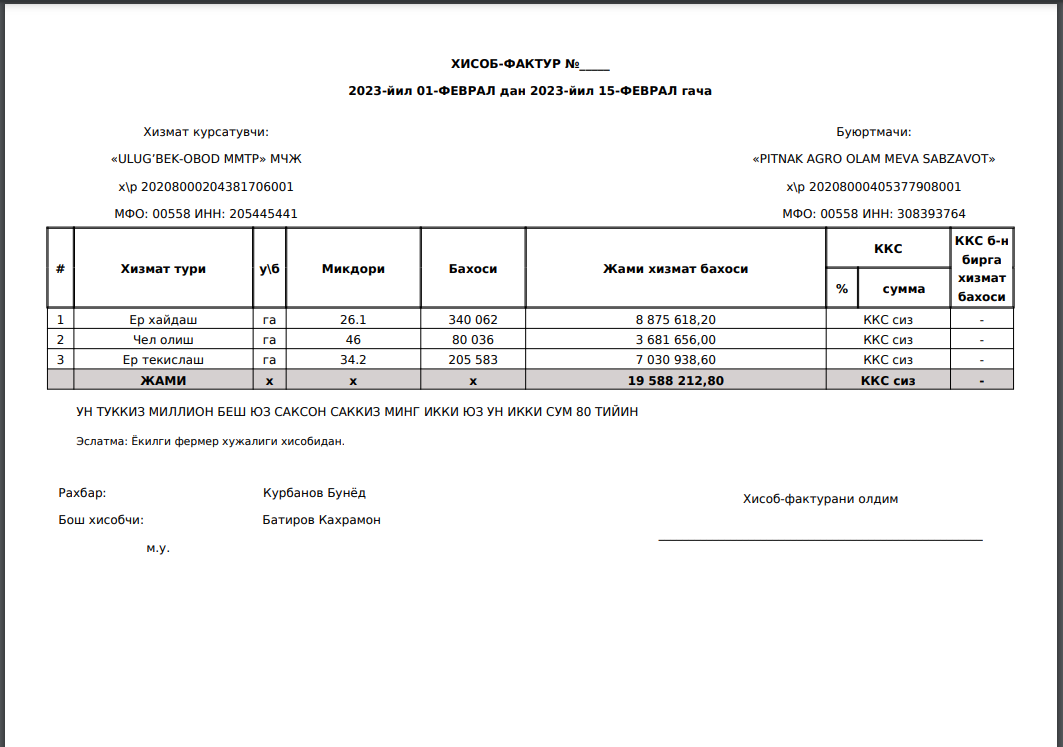
Web ilovamizni o‘lchov birliklari oynasi ko’rinishi quyida ko`rsatilgan



2.1 - rasm “Web ilovamizni o‘lchov birliklari oynasi ko’rinishi”

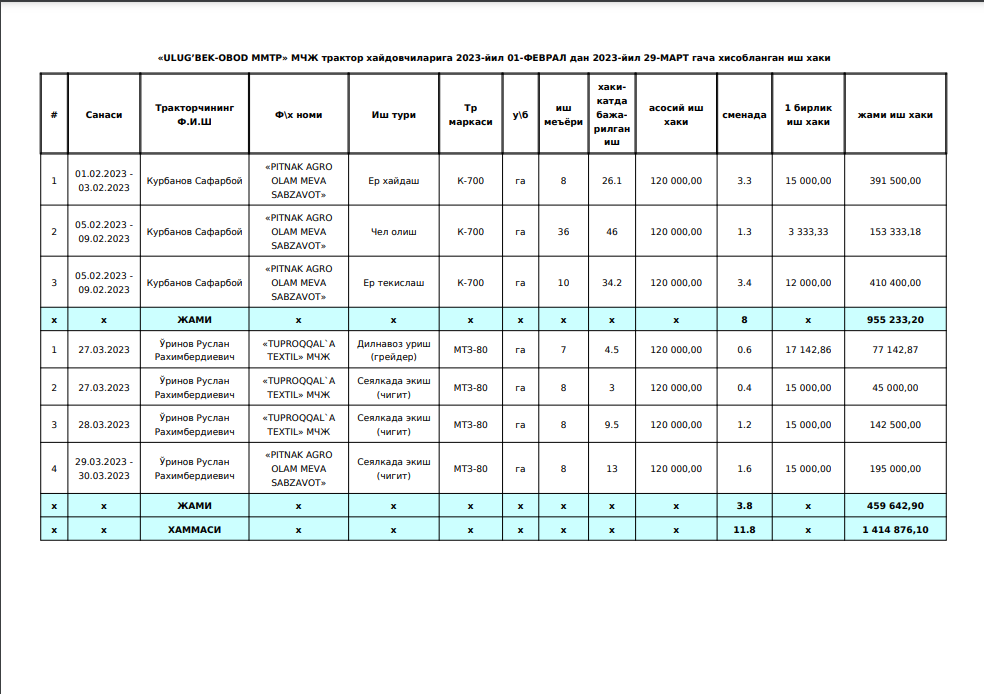
Dasturning o‘lchov birliklari qismida o‘lchov birliklarini yaratish, tahrirlash va o‘chirish mumkin.

Fermer bilan tuzilgan shartnoma quyida keltirilgan



2.1 - rasm “Fermer bilan tuzilgan shartnoma ko’rinishi”

Ishchilarni hisoblangan ish haqi quyida keltirilgan



2.1 - rasm “Ishchilarni hisoblangan ish haqi ko’rinishi”

**Yuklab olish**

1. **https://github.com/AsadbekPulatov/MMTP** githubdan ushbu fayllarni yuklab oling.
2. **Phpmyadmin** ga kirib **MMTP** nomli ma’lumotlar omborini yarating.
3. Ilovangiz ichiga kirib terminalga ushbu buyruqlarni yozing:
   1. composer update
   2. npm install
   3. npm run dev
   4. copy .env.example .env
   5. php artisan key:generate
   6. php artisan migrate --seed
   7. php artisan serve

# Xulosa

Hozirgi kunga kelib hisob kitob ishlari avtomatlashtirila boshladi. Bu ilova ham fermer xo‘jaliklariga xizmat ko‘rsatuvchi korxonalar ishini avtomatlashtirish uchun xizmat qiladi. Ilovada ishchilarning oylik maoshlarini hisoblash chiqarish, fermerlar bilan shartnomalar tuzish, fermerlarga xizmat ko‘rsatish va fayllarni yuklab olishimiz mumkin. Bundan tashqari foydalanuvchi uchun qulay interfeys qilishimiz kerak. Web ilovani yaratish davomida PHP dasturlash tilida PDF export qilish o‘rganildi.

# Foydalanilgan internet resurslari

**1. https:://www.php.net/  
2. https:://www.laravel.com/**

**3. https://www.mysql.com/**

# Ilova

**Report.php**

<?php

namespace App\Http\Service;

use App\Models\Report as ReportModel;

class Report

{

public function report($farmer\_id, $worker\_id, $from\_date=Null, $to\_date=Null, $page="none", $order\_by="DESC"){

if(!isset($from\_date)){

$from\_date = ReportModel::orderby('start\_date', 'ASC')->first();

if(isset($from\_date)){

$from\_date = $from\_date->start\_date;

}

}

if(!isset($to\_date)){

$to\_date = ReportModel::orderby('start\_date', 'DESC')->first();

if(isset($to\_date)){

$to\_date = $to\_date->start\_date;

}

}

if ($page == "none") {

if (isset($worker\_id))

$reports = ReportModel::orderBy('start\_date',

$order\_by)->whereBetween('start\_date', [$from\_date, $to\_date])

->whereIn('worker\_id', $worker\_id)->get();

else $reports = ReportModel::orderBy('start\_date', $order\_by)->whereBetween('start\_date', [$from\_date, $to\_date])->get();

$arr = [];

if (isset($reports)){

foreach ($reports as $worker) {

$arr[$worker->worker\_id]['data'] = [];

$arr[$worker->worker\_id]['sum\_staj'] = 0;

$arr[$worker->worker\_id]['sum\_price'] = 0;

}

foreach ($reports as $worker) {

if ($worker->start\_date == $worker->end\_date) {

$date = date('d.m.Y', strtotime($worker->start\_date));

} else {

$date = date('d.m.Y', strtotime($worker->start\_date)).' - '.date('d.m.Y', strtotime($worker->end\_date));

}

if ($worker->service->count){

$price = round($worker->service->price\_worker / $worker->service->count, 2);

$staj = round($worker->weight / $worker->service->count, 1);

}else{

$price = 0;

$staj = 0;

}

$arr[$worker->worker\_id]['data'][] = [

'id' => $worker->id,

'worker' => $worker->worker->name,

'farmer' => $worker->farmer->name,

'service' => $worker->service->name,

'tractor' => $worker->tractor->name,

'type' => $worker->service->type->type,

'count' => $worker->service->count,

'weight' => $worker->weight,

'price\_worker' => $worker->service->price\_worker,

'staj' => $staj,

'price\_worker\_oneday' => $price,

'price\_worker\_all' => round($price \* $worker->weight, 2),

'date' => $date,

];

$arr[$worker->worker\_id]['sum\_staj'] += $staj;

$arr[$worker->worker\_id]['sum\_price'] += round($price \* $worker->weight, 1);

}

$sum['staj'] = array\_sum(array\_column($arr, 'sum\_staj'));

$sum['price'] = array\_sum(array\_column($arr, 'sum\_price'));

}

return ['data' => $arr, 'sum' => $sum, 'from\_date' => $from\_date, 'to\_date' => $to\_date];

}

//reports.index

else {

$sum['staj'] = 0;

$sum['price'] = 0;

if (!isset($from\_date)) {

$reports = ReportModel::orderBy('start\_date', $order\_by)->get();

$page = 'farmer';

} else {

if (isset($worker\_id)) {

if (isset($farmer\_id)) {

$reports = ReportModel::orderBy('start\_date', $order\_by)->whereBetween('start\_date', [$from\_date, $to\_date])->where('worker\_id', $worker\_id)->where('farmer\_id', $farmer\_id)->get();

} else {

$reports = ReportModel::orderBy('start\_date', $order\_by)->whereBetween('start\_date', [$from\_date, $to\_date])->where('worker\_id', $worker\_id)->get();

}

foreach ($reports as $report) {

$sum['staj'] += $report->weight;

$sum['price'] += $report->service->price \* $report->weight;

}

$page = 'worker';

} else {

if (isset($farmer\_id)) {

$reports = ReportModel::orderBy('start\_date', $order\_by)->whereBetween('start\_date', [$from\_date, $to\_date])->where('farmer\_id', $farmer\_id)->get();

} else

$reports = ReportModel::orderBy('start\_date', $order\_by)->whereBetween('start\_date', [$from\_date, $to\_date])->get();

$page = 'farmer';

}

}

return ['data' => $reports, 'sum' => $sum, 'page' => $page, 'from\_date' => $from\_date, 'to\_date' => $to\_date];

}

}

}

**number\_to\_word.php**

<?php

namespace App\Http\Service;

class number\_to\_word{

public function number\_to\_word($number)

{

$length = strlen($number);

$new\_word = '';

$slice = $length % 3;

if ($slice == 0) {

$slice = 3;

}

for ($i = $slice; $i < 3; $i++)

$number = '0' . $number;

$length = strlen($number);

$index = 0;

while ($length > 0) {

$slice\_number = substr($number, $index, 3);

$n\_100 = $slice\_number[0];

$n\_10 = $slice\_number[1];

$n\_1 = $slice\_number[2];

switch ($n\_100) {

case 1:

$new\_word .= 'Бир Юз ';

break;

case 2:

$new\_word .= 'Икки Юз ';

break;

case 3:

$new\_word .= 'Уч Юз ';

break;

case 4:

$new\_word .= 'Турт Юз ';

break;

case 5:

$new\_word .= 'Беш Юз ';

break;

case 6:

$new\_word .= 'Олти Юз ';

break;

case 7:

$new\_word .= 'Етти Юз ';

break;

case 8:

$new\_word .= 'Саккиз Юз ';

break;

case 9:

$new\_word .= 'Туккиз Юз ';

break;

}

switch ($n\_10) {

case 1:

$new\_word .= 'Ун ';

break;

case 2:

$new\_word .= 'Йигирма ';

break;

case 3:

$new\_word .= 'Уттиз ';

break;

case 4:

$new\_word .= 'Кирк ';

break;

case 5:

$new\_word .= 'Эллик ';

break;

case 6:

$new\_word .= 'Олтмиш ';

break;

case 7:

$new\_word .= 'Етмиш ';

break;

case 8:

$new\_word .= 'Саксон ';

break;

case 9:

$new\_word .= 'Туксон ';

break;

}

switch ($n\_1) {

case 1:

$new\_word .= 'Бир';

break;

case 2:

$new\_word .= 'Икки';

break;

case 3:

$new\_word .= 'Уч';

break;

case 4:

$new\_word .= 'Турт';

break;

case 5:

$new\_word .= 'Беш';

break;

case 6:

$new\_word .= 'Олти';

break;

case 7:

$new\_word .= 'Етти';

break;

case 8:

$new\_word .= 'Саккиз';

break;

case 9:

$new\_word .= 'Туккиз';

break;

}

switch ($length) {

case 4:

$new\_word .= ' Минг ';

break;

case 5:

$new\_word .= ' Минг ';

break;

case 6:

$new\_word .= ' Минг ';

break;

case 7:

$new\_word .= ' Миллион ';

break;

case 8:

$new\_word .= ' Миллион ';

break;

case 9:

$new\_word .= ' Миллион ';

break;

case 10:

$new\_word .= ' Миллиард ';

break;

case 11:

$new\_word .= ' Миллиард ';

break;

case 12:

$new\_word .= ' Миллиард ';

break;

}

$length -= 3;

$index += 3;

}

$new\_word = mb\_strtoupper($new\_word, "utf-8");

return $new\_word;

}

}

**DateFormat.php**

<?php

namespace App\Http\Service;

class DateFormat

{

public function date($from\_date, $to\_date){

$year = date('Y', strtotime($from\_date));

$month = date('m', strtotime($from\_date));

$day = date('d', strtotime($from\_date));

$date = $year."-йил";

switch ($month){

case "01": $month = \_\_("messages.january"); break;

case "02": $month = \_\_("messages.february"); break;

case "03": $month = \_\_("messages.march"); break;

case "04": $month = \_\_("messages.april"); break;

case "05": $month = \_\_("messages.may"); break;

case "06": $month = \_\_("messages.june"); break;

case "07": $month = \_\_("messages.july"); break;

case "08": $month = \_\_("messages.august"); break;

case "09": $month = \_\_("messages.september"); break;

case "10": $month = \_\_("messages.october"); break;

case "11": $month = \_\_("messages.november"); break;

case "12": $month = \_\_("messages.december"); break;

}

$date = $date." ".$day."-".mb\_strtoupper($month, 'UTF-8');

$year = date('Y', strtotime($to\_date));

$month = date('m', strtotime($to\_date));

$day = date('d', strtotime($to\_date));

$date .= " дан ".$year."-йил";

switch ($month){

case "01": $month = \_\_("messages.january"); break;

case "02": $month = \_\_("messages.february"); break;

case "03": $month = \_\_("messages.march"); break;

case "04": $month = \_\_("messages.april"); break;

case "05": $month = \_\_("messages.may"); break;

case "06": $month = \_\_("messages.june"); break;

case "07": $month = \_\_("messages.july"); break;

case "08": $month = \_\_("messages.august"); break;

case "09": $month = \_\_("messages.september"); break;

case "10": $month = \_\_("messages.october"); break;

case "11": $month = \_\_("messages.november"); break;

case "12": $month = \_\_("messages.december"); break;

}

$date = $date." ".$day."-".mb\_strtoupper($month, 'UTF-8')." гача";

return $date;

}

}

**web.php**

<?php

use App\Http\Controllers\ProfileController;

use App\Http\Controllers\TractorController;

use App\Http\Controllers\FarmerController;

use App\Http\Controllers\ReportController;

use App\Http\Controllers\TypeController;

use App\Http\Controllers\OfficeController;

use App\Http\Controllers\ServiceController;

use App\Http\Controllers\WorkerController;

use Illuminate\Support\Facades\Route;

Route::get('/', function () {

return view('welcome');

});

Route::get('/dashboard', function () {

return view('admin.master');

})->middleware(['auth', 'verified'])->name('dashboard');

Route::middleware('auth')->group(function () {

Route::get('/download/workers/',[\App\Http\Controllers\DownloadController::class, 'workers'])->name('download.workers');

Route::get('/download/farmers/',[\App\Http\Controllers\DownloadController::class, 'farmers'])->name('download.farmers');

Route::get('/report/workers/', [ReportController::class, 'worker'])->name('reports.workers');

Route::resource('reports', ReportController::class);

Route::resource('office', OfficeController::class);

Route::resource('farmers', FarmerController::class);

Route::resource('services', ServiceController::class);

Route::resource('tractors', TractorController::class);

Route::resource('types', TypeController::class);

Route::resource('workers', WorkerController::class);

Route::get('/profile', [ProfileController::class, 'edit'])->name('profile.edit');

Route::patch('/profile', [ProfileController::class, 'update'])->name('profile.update');

Route::delete('/profile', [ProfileController::class, 'destroy'])->name('profile.destroy');

});

require \_\_DIR\_\_.'/auth.php';

**DownloadController.php**

<?php

namespace App\Http\Controllers;

use App\Exports\ExportReport;

use App\Imports\ImportTest;

use App\Models\Office;

use App\Models\Report;

use Barryvdh\DomPDF\Facade\Pdf;

use Illuminate\Http\Request;

use App\Http\Service\Report as ReportService;

use App\Http\Service\DateFormat;

use Maatwebsite\Excel\Facades\Excel;

class DownloadController extends Controller

{

public function workers(Request $request){

$type = $request->type;

$report = new ReportService();

$from\_date = $request->from\_date;

$to\_date = $request->to\_date;

$worker\_id = $request->worker\_id;

$report = $report->report(NULL,$worker\_id,$from\_date,$to\_date, "none", "ASC");

$workers = $report['data'];

$sum = $report['sum'];

$from\_date = $report['from\_date'];

$to\_date = $report['to\_date'];

$date = new DateFormat();

$date = $date->date($from\_date, $to\_date);

$office = Office::all()->first();

if ($type == "xls"){

$excel = new ExportReport($workers, $sum, $from\_date, $to\_date, $worker\_id, $date, $office);

return Excel::download($excel, "Иш хаки ({$from\_date} {$to\_date}).xlsx");

}

$pdf = Pdf::loadView('admin.download.worker',[

'workers' => $workers,

'sum' => $sum,

'from\_date' => $from\_date,

'to\_date' => $to\_date,

'worker\_id' => $worker\_id,

'date' => $date,

'office' => $office

])->setPaper('a4', 'landscape');

$pdf->setOption(['dpi' => 150, 'defaultFont' => 'sans-serif']);

return $pdf->download("Иш хаки ({$from\_date} {$to\_date}).pdf");

}

public function farmers(Request $request){

$from\_date = $request->from\_date;

$to\_date = $request->to\_date;

$worker\_id = $request->worker\_id;

$farmer\_id = $request->farmer\_id;

$report = new ReportService();

$report = $report->report($farmer\_id,$worker\_id,$from\_date,$to\_date,"1", "ASC");

$reports = $report['data'];

$sum = $report['sum'];

$page = $report['page'];

$from\_date = $report['from\_date'];

$to\_date = $report['to\_date'];

$date = new DateFormat();

$date = $date->date($from\_date, $to\_date);

$pdf = Pdf::loadView('admin.download.farmer',

compact('reports', 'page','sum',

'from\_date', 'to\_date','worker\_id','farmer\_id','date'))->setPaper('a4', 'landscape');

$pdf->setOption(['dpi' => 150, 'defaultFont' => 'sans-serif']);

if ($page == "farmer"){

$farmer = $reports[0]->farmer->name;

return $pdf->download("{$farmer}({$from\_date} {$to\_date}).pdf");

}

if ($page == "worker"){

$worker = $reports[0]->worker->name;

return $pdf->download("{$worker}({$from\_date} {$to\_date}).pdf");

}

}

}

**ReportController.php**

<?php

namespace App\Http\Controllers;

use App\Http\Service\number\_to\_word;

use App\Models\Farmer;

use App\Models\Report;

use App\Models\Service;

use App\Models\Tractor;

use App\Models\Type;

use App\Models\Worker;

use Illuminate\Http\Request;

use App\Http\Service\Report as ReportService;

use Illuminate\Support\Facades\Log;

use Stevebauman\Location\Facades\Location;

class ReportController extends Controller

{

public function worker(Request $request)

{

$report = new ReportService();

$from\_date = $request->from\_date;

$to\_date = $request->to\_date;

$worker\_id = $request->worker\_id;

$report = $report->report(NULL,$worker\_id,$from\_date,$to\_date);

$workers = $report['data'];

$sum = $report['sum'];

return view('admin.reports.worker', compact('workers', 'sum','from\_date','to\_date','worker\_id'));

}

public function index(Request $request)

{

$from\_date = $request->from\_date;

$to\_date = $request->to\_date;

$worker\_id = $request->worker\_id;

$farmer\_id = $request->farmer\_id;

$report = new ReportService();

$report = $report->report($farmer\_id,$worker\_id,$from\_date,$to\_date,"1");

$reports = $report['data'];

$sum = $report['sum'];

$page = $report['page'];

return view('admin.reports.index', compact('reports', 'page','sum', 'from\_date', 'to\_date','worker\_id','farmer\_id'));

}

public function create()

{

$workers = Worker::all();

$tractors = Tractor::all();

$farmers = Farmer::all();

$services = Service::orderby('date', 'desc')->get();

$types = Type::all()->pluck('type', 'id');

return view('admin.reports.create', compact('workers', 'tractors', 'farmers', 'services', 'types'));

}

public function store(Request $request)

{

$request['weight'] = str\_replace(',','.', $request['weight']);

Report::create($request->all());

return redirect()->route('reports.index')->with('success', \_\_("messages.report\_created"));

}

public function edit(Report $report)

{

$workers = Worker::all();

$tractors = Tractor::all();

$farmers = Farmer::all();

$services = Service::orderby('date', 'desc')->get();

$types = Type::all()->pluck('type', 'id');

return view('admin.reports.edit', compact('report', 'workers', 'tractors', 'farmers', 'services', 'types'));

}

public function update(Request $request, Report $report)

{

$report->update($request->all());

return redirect()->route('reports.index')->with('success', \_\_("messages.report\_updated"));

}

public function destroy(Report $report)

{

$report->delete();

return redirect()->route('reports.index')->with('success', \_\_("messages.report\_deleted"));

}

}

**ServiceController.php**

<?php

namespace App\Http\Controllers;

use App\Models\Service;

use App\Models\Tractor;

use App\Models\Type;

use Illuminate\Http\Request;

class ServiceController extends Controller

{

public function index()

{

$services = Service::orderBy('date', 'desc')->get();

return view('admin.services.index', compact('services'));

}

public function create()

{

$tractors = Tractor::all();

$types = Type::all();

return view('admin.services.create', compact('tractors', 'types'));

}

public function store(Request $request)

{

Service::create($request->all());

return redirect()->route('services.index')->with('success', \_\_("messages.service\_created"));

}

public function edit(Service $service)

{

$types = Type::all();

return view('admin.services.edit', compact('service', 'types'));

}

public function update(Request $request, Service $service)

{

$service->update($request->all());

return redirect()->route('services.index')->with('success', \_\_("messages.service\_updated"));

}

public function destroy(Service $service)

{

$service->delete();

return redirect()->route('services.index')->with('success', \_\_("messages.service\_deleted"));

}

}

**FarmerController.php**

<?php

namespace App\Http\Controllers;

use App\Models\Farmer;

use Illuminate\Http\Request;

class FarmerController extends Controller

{

public function index()

{

$farmers = Farmer::all();

return view('admin.farmers.index', compact('farmers'));

}

public function create()

{

return view('admin.farmers.create');

}

public function store(Request $request)

{

Farmer::create($request->all());

return redirect()->route('farmers.index')->with('success', \_\_('messages.farmer\_created'));

}

public function edit(Farmer $farmer)

{

return view('admin.farmers.edit', compact('farmer'));

}

public function update(Request $request, Farmer $farmer)

{

$farmer->update($request->all());

return redirect()->route('farmers.index')->with('success', \_\_('messages.farmer\_updated'));

}

public function destroy(Farmer $farmer)

{

$farmer->delete();

return redirect()->route('farmers.index')->with('success', \_\_('messages.farmer\_deleted'));

}

}