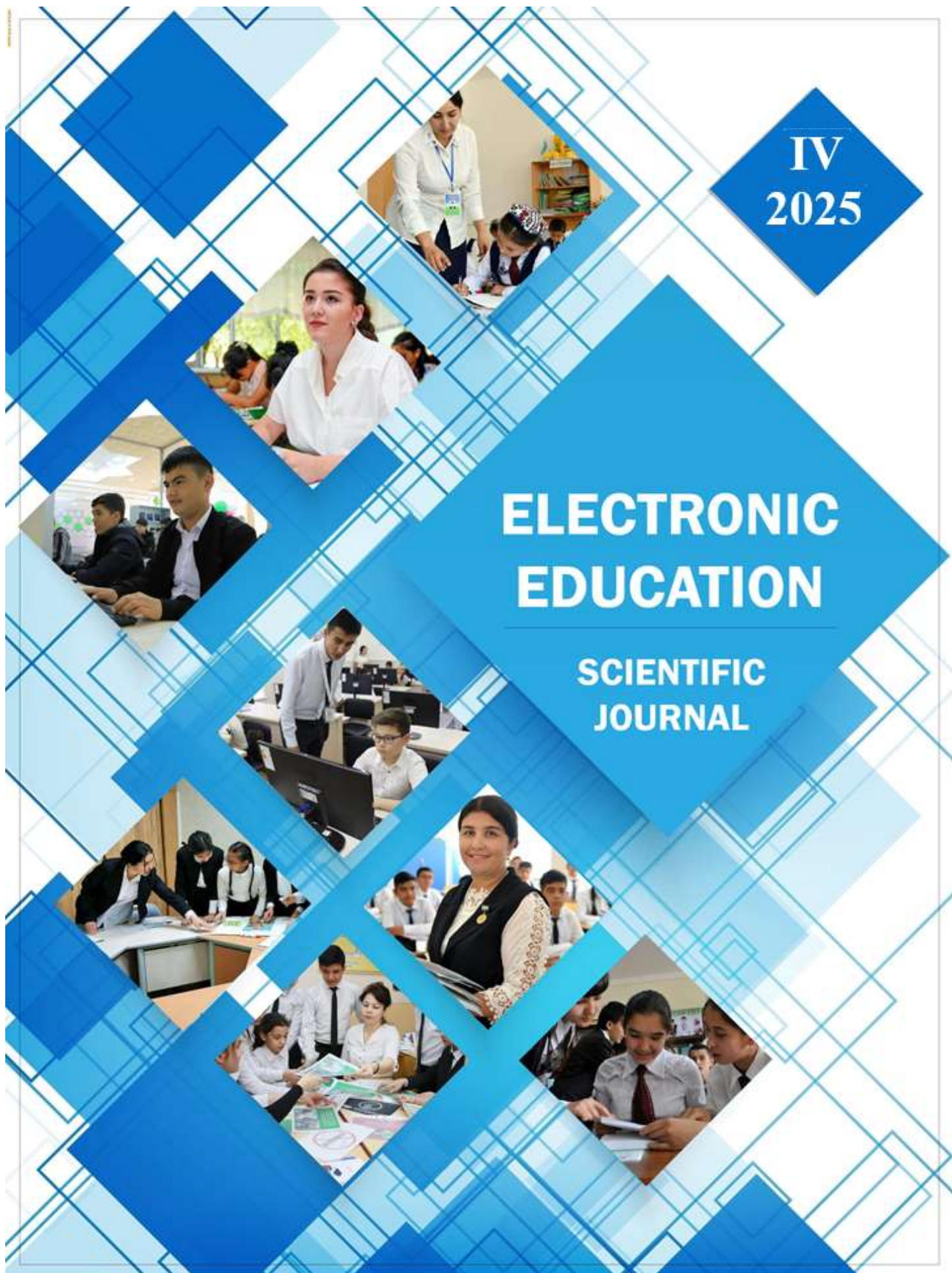


IV  
2025

# ELECTRONIC EDUCATION

SCIENTIFIC  
JOURNAL



## TAHRIRIYAT

### **Bosh muharrir**

**Laqayev Saidaxmad Norjigitovich**  
fizika-matematika fanlari doktori, akademik

### **Bosh muharrir o‘rinbosari**

**Ro‘ziyev Rauf Axmadovich**  
fizika-matematika fanlari nomzodi, dotsent

### **Mas’ul muharrir**

**Mirsanov Uralboy Mukhammadiyevich**  
pedagogika fanlari doktori DSc, professor

### **Editor-in-Chief**

**Saidakhmad Norjigitovich Lakayev**  
doctor of physical and mathematical sciences,  
academician

### **Deputy Editor-in-Chief**

**Ruziyev Raup Akhmadovich**  
Candidate of Physical and Mathematical Sciences,  
Associate Professor

### **Responsible editor**

**Mirsanov Uralboy Mukhammadiyevich**  
doctor of Pedagogical Sciences DSc, Professor

## TAHRIRIYAT A’ZOLARI

**Kalonov Muxiddin Baxriddinovich** - iqtisodiyot fanlari doktori, professor. (O‘zbekiston)

**Xujjiyev Sodiq Oltiyevich**- biologiya fanlari nomzodi, dotsent. (O‘zbekiston)

**Ibragimov Alimjon Artikbayevich**-fizika-matematika fanlari nomzodi, dotsent. (O‘zbekiston)

**Suvonov Olim Omonovich**- texnika fanlari nomzodi, dotsent. (O‘zbekiston)

**Yodgorov G‘ayrat Ro‘ziyevich**-fizika- matematika fanlari nomzodi, dotsent. (O‘zbekiston)

**Nasirova Shaira Narmuradovna**-texnika fanlari doktori, professor. (O‘zbekiston)

**O‘tapov Toyir Usmonovich**-pedagogika fanlari nomzodi, dotsent. (O‘zbekiston)

**Xudoyorov Shuxrat Jumaqulovich**- fizika-matematika fanlari nomzodi, dotsent. (O‘zbekiston)

**Djurayev Risbay Xaydarovich**- akademik (O‘zbekiston)

**Negmatov Sayibjon Sodiqovich**- akademik (O‘zbekiston)

**Aripov Mersaid Mirsiddikovich**- fizika-matematika fanlari doktori, professor. (O‘zbekiston)

**Turabdjano Sadritdin Maxamatdinovich** - texnika fanlari doktori, akademik. (O‘zbekiston)

**Raximov Isomiddin Sattarovich**- fizika-matematika fanlari doktori, professor. (Malayziya)

**Shariy Sergey Petrovich**- fizika-matematika fanlari doktori, professor. (Rossiya)

**Ibraimov Xolboy**- pedagogika fanlari doktori, akademik. (O‘zbekiston)

**Yunusova Dilfuza Isroilovna**- pedagogika fanlari doktori, professor. (O‘zbekiston)

**Aloyev Raxmatillo Djurayevich**- fizika-matematika fanlari doktori, professor. (O‘zbekiston)

**Abdullayeva Shaxzoda Abdullayevna**- pedagogika fanlari doktori, professor. (O‘zbekiston)

**Norov Abdusaid Murodovich** – texnika fanlari bo‘yicha falsafa doktori, dotsent (O‘zbekiston).

**Yuldoshev Ismoil Abriyevich** – pedagogika fanlari bo‘yicha falsafa doktori, dotsent (O‘zbekiston)

**Mo‘minov Bahodir Boltayevich**- texnika fanlari doktori, professor. (O‘zbekiston)

**Rosmayati Mohemad** - professor. (Malayziya)

**Zainidin K. Eshkuvatov** – fizika-matematikafanlari doktori (DSc). (Malayziya)

**Muhammad Suzuri bin Hitam** - professor. Malayziya)

**Amiza binti Mat Amin**- professor. (Malayziya)

**Korshunov Igor Lvovich**- texnika fanlari nomzodi, dotsent. (Rossiya)

**Kolbanyov Mixail Olegovich**- texnika fanlari doktori, professor. (Rossiya)

**Verzun Natalya Arkadyevna**- texnika fanlari nomzodi, dotsent. (Rossiya)

**Stelmashonok Yelena Viktorovna**- iqtisod fanlari doktori, professor. (Rossiya)

**Tatarnikova Tatyana Mixaylovna** - texnika fanlari doktori, professor. (Rossiya)

**Alekseyev Vladimir Vasilyevich** - texnika fanlari doktori, professor. (Rossiya)

**Satikov Igor Abuzarovich** – fizika-matematika fanlari nomzodi, dotsent. (Belarus)

**Boyarshinova Oksana Aleksandrovna** – fizika-matematika fanlari nomzodi, dotsent. (Belarus)

**Makarenko Sergey Nikolayevich** – texnika fanlari nomzodi, dotsent. (Belarus)

**Sednina Marina Aleksandrovna** – texnika fanlari nomzodi, dotsent. (Belarus)

**Xolmurodov Abdulhamid Erkinovich**- fizika-matematika fanlari doktori, professor. (O‘zbekiston)

**Lutfillayev Maxmud Xasanovich**- pedagogika fanlari doktori, professor (O‘zbekiston)

**Ergasheva Gulruxsor Surxonidinovna** - pedagogika fanlari doktori (DSc), dotsent. (O‘zbekiston)

**Maxmudova Dilfuza Mileyevna** – pedagogika fanlari doktori, professor (O‘zbekiston)

**Xudjayev Muxiddin Kushshayevich** – texnika fanlari doktori, dotsent (O‘zbekiston).

**Ibragimov Abdusattar Turgunovich** – texnika fanlari doktori, dotsent (O‘zbekiston).



---

**Karaxonova Oysara Yuldoshevna** – pedagogika fanlari  
bo'yicha falsafa doktori (O'zbekiston).

**Kurbaniyazova Zamira Kalbaevna**- pedagogika fanlari  
doktori, dotsent. (O'zbekiston)

**Jabbarov Oybek Rakhmanovich**- fizika-matematika  
fanlari bo'yicha falsafa doktori, dotsent (O'zbekiston).

**Kabiljanova Firuza Azimovna**-fizika-matematika  
fanlari nomzodi, dotsent. (O'zbekiston)

**Kalonova Mohigul Baxriddinovna**-iqtisod fanlari  
bo'yicha falsafa doktori. (O'zbekiston)

**Baxodirova Umida Baxodirovna**-pedagogika fanlari  
bo'yicha falsafa doktori, dotsent. (O'zbekiston)

**Sharipov Ergash Oripovich**-pedagogika fanlari  
bo'yicha falsafa doktori, dotsent. (O'zbekiston)

**Xamroyeva Dilafro'z Namozovna** – fizika-matematika  
fanlari bo'yicha falsafa doktori, dotsent. (O'zbekiston).

**Toxirov Feruz Jamoliddinovich** – pedagogika fanlari  
bo'yicha falsafa doktori (O'zbekiston)

**Isroilova Lola Sunnatovna** – pedagogika fanlari  
bo'yicha falsafa doktori, dotsent. (O'zbekiston)

**Otaqulova Durdona Raxmonovna** – pedagogika  
fanlari bo'yicha falsafa doktori (O'zbekiston)

**Ruziyeva Dilafruz Raupovna** – pedagogika fanlari  
bo'yicha falsafa doktori (O'zbekiston)

**Jo'rakulov Tolib Toxirovich**- texnik muharrir

© Mazkur jurnal O'zbekiston Respublikasi Vazirlar Mahkamasi huzuridagi Oliy Attestatsiya komissiyasi rayosatining 2022-yil 28-fevraldagi 312/6 qaroriga asosan Pedagogika fanlari bo'yicha falsafa doktori (PhD) va fan doktori (DSc) ilmiy darajasiga talabgorlarning dissertatsiya ishlari yuzasidan dissertatsiyalari asosiy ilmiy natijalarini chop etish uchun tavsiya etilgan ilmiy nashrlar ro'yxatiga kiritilgan

Adress: Navoiy sh., Janubiy ko'chasi, 1-A uy. (1-A, South Street, Navoi city) URL:  
<http://www.el-nspi.uz>

## MUNDARIJA

### *Aniq fanlarda axborot texnologiyalari*

|                                                                                                                                                                      |     |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| <b>Mirsanov U. M., Nurullayeva I. Z.</b><br><i>INSON YUZINI ANIQLASH BO‘YICHA DASTURIY TA‘MINOTNI (FACE ID) ISHLAB CHIQISH USULI</i>                                 | 7   |
| <b>Ruziyev R. A., Farmanova M.Sh.</b><br><i>TA‘LIMNI RAQAMLASHTIRISH JARAYONIDA VIZUAL TA‘LIM TEXNOLOGIYALARI DASTURIY VOSITALARNING IMKONIYATLARI</i>               | 17  |
| <b>Yarashov Sh. T.</b><br><i>INSON YUZINI ANIQLOVCHI (FACE ID) WEB-ILOVA ISHLAB CHIQISH USU</i>                                                                      | 27  |
| <b>Ruziyev R. A., Ismatova Sh. U.</b><br><i>TA‘LIMNI RAQAMLASHTIRISH SHAROITIDA ELEKTRON TA‘LIM RESURSLARINI YARATISH MOHIYATI VA TURLARI</i>                        | 52  |
| <b>Mirsanov U. M., Ramazonov I. R., Majidov Sh. A.</b><br><i>GOOGLE FORMS ASOSIDA ONLAYN TESTLARNI O‘TKAZISHDA FACE ID TIZIMIGA INTEGRATSIYA QILISH</i>              | 63  |
| <b>Xamroyeva D. N., Kamolova M. F.</b><br><i>TURLI MURAKKABLIK DARAJADAGI MANTIQUIY MASALALARNI O‘QITISHDA KO‘RGAZMALI VA ALGORITMIK YONDASHUVLAR</i>                | 72  |
| <b>Otaqulova D. R.</b><br><i>TALABALARDA GRAFIK MODELLAR LOYIHALASHDA SUN‘IY INTELLEKT TEXNOLOGIYASIDAN FOYDALANISH USULI</i>                                        | 82  |
| <b>Xamroyev U. N.</b><br><i>MUAMMOLI TA‘LIM TEXNOLOGIYASI ORQALI TALABALARNING ALGORITMIK KOMPETENTLIGINI BOSQICHMA-BOSQICH RIVOJLANTIRISH USULI</i>                 | 91  |
| <b><i>Tabiiy fanlarda axborot texnologiyalari</i></b>                                                                                                                |     |
| <b>Baxodirova U. B.</b><br><i>OLIY TA‘LIM MUASSASALARIDA BIOLOGIYA FANLARIDAN VIRTUAL LABORATORIYALARNING AMALIY SAMARADORLIG</i>                                    | 105 |
| <b><i>Ijtimoiy-gumanitar fanlarda axborot texnologiyalari</i></b>                                                                                                    |     |
| <b>Sharifjanova G.</b><br><i>AXBOROT-KOMMUNIKATSIYA TEXNOLOGIYALARI ASOSIDA TALABALAR IQTISODIY KOMPETENTLIGINI RIVOJLANTIRISH USULI</i>                             | 112 |
| <b>Ergasheva F. T.</b><br><i>TA‘LIMNI RAQAMLASHTIRISH SHAROITIDA TALABALARNING METAPROFESSIONAL KOMPETENTLIGINI RIVOJLANTIRISHNING INNOVATSION METODLARI TAHLILI</i> | 124 |

## СОДЕРЖАНИЕ

| <b>Информационные технологии в точных науках</b>                                                                                                           |     |
|------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| <b>Мирсанов У. М., Нуруллаева Л. З.</b><br>СПОСОБ РАЗРАБОТКИ ПРОГРАММНОГО ОПРЕДЕЛЕНИЯ ЛИЦА ЧЕЛОВЕКА (FACE ID)                                              | 7   |
| <b>Рузиев Р. А., Фарманова М.</b><br>ВОЗМОЖНОСТИ ВИЗУАЛЬНЫХ ОБРАЗОВАТЕЛЬНЫХ ТЕХНОЛОГИЙ И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ В ПРОЦЕССЕ ЦИФРОВИЗАЦИИ ОБРАЗОВАНИЯ      | 17  |
| <b>Ярашов Ш.</b><br>МЕТОД РАЗРАБОТКИ ВЕБ-ПРИЛОЖЕНИЯ ДЛЯ РАСПОЗНАВАНИЯ ЛИЦ (FACE ID)                                                                        | 27  |
| <b>Рузиев Р. А., Исматова Ш.</b><br>СУЩНОСТЬ И ВИДЫ СОЗДАНИЯ ЭЛЕКТРОННЫХ ОБРАЗОВАТЕЛЬНЫХ РЕСУРСОВ В КОНТЕКСТЕ ЦИФРОВИЗАЦИИ ОБРАЗОВАНИЯ                     | 52  |
| <b>Мирсанов У. М., Рамазонов И., Мажидов Ш. А.</b><br>ИНТЕГРАЦИЯ СИСТЕМЫ FACE ID В ПРОВЕДЕНИЕ ОНЛАЙН-ТЕСТОВ НА ОСНОВЕ GOOGLE FORM                          | 63  |
| <b>Хамроева Д. Н., Камолова М.</b><br>ОБРАЗЦОВЫЕ И АЛГОРИТМИЧЕСКИЕ ПОДХОДЫ К ОБУЧЕНИЮ ЛОГИЧЕСКИМ ЗАДАЧАМ РАЗЛИЧНОЙ СЛОЖНОСТИ                               | 72  |
| <b>Отакулова Д. Р.</b><br>МЕТОД ИСПОЛЬЗОВАНИЯ ТЕХНОЛОГИЙ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА ПРИ РАЗРАБОТКЕ ГРАФИЧЕСКИХ МОДЕЛЕЙ СТУДЕНТАМИ                           | 82  |
| <b>Хамроев У. Н.</b><br>МЕТОД ПОЭТАПНОГО РАЗВИТИЯ АЛГОРИТМИЧЕСКОЙ КОМПЕТЕНТНОСТИ УЧАЩИХСЯ ПОСРЕДСТВОМ ПРОБЛЕМНО-ОРИЕНТИРОВАННЫХ ОБРАЗОВАТЕЛЬНЫХ ТЕХНОЛОГИЙ | 91  |
| <b>Информационные технологии в естественных науках</b>                                                                                                     |     |
| <b>Баходирова У. Б.</b><br>ПРАКТИЧЕСКАЯ ЭФФЕКТИВНОСТЬ ВИРТУАЛЬНЫХ ЛАБОРАТОРИЙ В БИОЛОГИЧЕСКОМ ОБРАЗОВАНИИ В ВЫСШИХ УЧЕБНЫХ ЗАВЕДЕНИЯХ                      | 105 |
| <b>Информационные технологии в социально-гуманитарных науках</b>                                                                                           |     |
| <b>Шарифжанова Г.</b><br>МЕТОД РАЗВИТИЯ ЭКОНОМИЧЕСКОЙ КОМПЕТЕНТНОСТИ СТУДЕНТОВ НА ОСНОВЕ ИНФОРМАЦИОННО-КОММУНИКАЦИОННЫХ ТЕХНОЛОГИЙ                         | 112 |
| <b>Эргашева Ф. Т.</b><br>АНАЛИЗ ИННОВАЦИОННЫХ МЕТОДОВ РАЗВИТИЯ МЕТАПРОФЕССИОНАЛЬНЫХ КОМПЕТЕНЦИЙ СТУДЕНТОВ В УСЛОВИЯХ ЦИФРОВИЗАЦИИ ОБРАЗОВАНИЯ              | 124 |

## CONTENT

| <i>Information technologies in exact sciences</i>                                                                                                                                                              |     |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| <b>Mirsanov Uralboy , Nurullaeva Laylo</b><br>DEVELOPMENT METHOD FOR PROGRAMM DETERMINATION OF HUMAN FACE (FACE ID)                                                                                            | 7   |
| <b>Ruziev Raup, Farmanova Malika</b><br>CAPABILITIES OF VISUAL EDUCATIONAL TECHNOLOGIES AND SOFTWARE IN THE PROCESS OF DIGITALIZATION OF EDUCATION                                                             | 17  |
| <b>Yarashov Shokhruxh</b><br>A METHOD FOR DEVELOPING A WEB APPLICATION FOR FACE RECOGNITION (FACE ID)                                                                                                          | 27  |
| <b>Ruziev Raup, Ismatova Shakhnoza</b><br>THE ESSENCE AND TYPES OF CREATION OF ELECTRONIC EDUCATIONAL RESOURCES IN THE CONTEXT OF DIGITALIZATION OF EDUCATION                                                  | 52  |
| <b>Mirsanov Uralboy, Ramazonov Ismoiljon, Majidov Sherzod</b><br>INTEGRATION OF THE FACE ID SYSTEM INTO ONLINE TESTS BASED ON GOOGLE FORMS                                                                     | 63  |
| <b>Khamroeva Dilafruz, Kamolova Mokhiniso</b><br>EXEMPLARY AND ALGORITHICAL APPROACHES IN TEACHING LOGICAL PROBLEMS OF VARIOUS COMPLEXITY LEVELS                                                               | 72  |
| <b>Otakulova Durdona</b><br>A METHOD OF USING ARTIFICIAL INTELLIGENCE TECHNOLOGIES IN THE DEVELOPMENT OF GRAPHIC MODELS BY STUDENTS                                                                            | 82  |
| <b>Khamroev Utkir</b><br>METHOD FOR STAGE-BY-STEP DEVELOPMENT OF STUDENTS' ALGORITHICAL COMPETENCE THROUGH PROBLEM-BASED EDUCATIONAL TECHNOLOGY                                                                | 91  |
| <i>Information technologies in natural sciences</i>                                                                                                                                                            |     |
| <b>Bakhodirova Umida</b><br>THE PRACTICAL EFFECTIVENESS OF VIRTUAL LABORATORIES IN BIOLOGY EDUCATION AT HIGHER EDUCATIONAL INSTITUTION ORGANIZING EXTRACURRICULAR LEARNING ACTIVITIES OF STUDENTS IN GEOGRAPHY | 105 |
| <i>Information Technologies in Social Sciences and Humanities</i>                                                                                                                                              |     |
| <b>Sharifjanova Gulchehra</b><br>A METHOD OF DEVELOPING STUDENTS' ECONOMIC COMPETENCE BASED ON INFORMATION AND COMMUNICATION TECHNOLOGIES                                                                      | 112 |
| <b>Ergasheva Fatima</b><br>ANALYSIS OF INNOVATIVE METHODS OF DEVELOPING STUDENTS' META-PROFESSIONAL COMPETENCIES IN THE CONTEXT OF DIGITALIZATION OF EDUCATION                                                 | 124 |

---

## *Aniq fanlarda axborot texnologiyalari*

---

### INSON YUZINI ANIQLOVCHI (FACE ID) WEB-ILOVA ISHLAB CHIQISH USULI

**Yarashov Shoxrux Tolmas ug‘li**

Navoiy davlat universiteti, O‘zbekiston

**Annotatsiya:** Ushbu maqolada inson yuzini aniqlovchi (FACE ID) dasturiy ta’minot yaratish muammolari va unga oid olimlarning tadqiqotlari tahlil etilgan. Shuningdek, mazkur maqolada inson yuzini aniqlovchi (FACE ID) web-ilova ishlab chiqish usuli keltirilgan.

**Tayanch so‘zlar:** FACE ID, web-ilova, dasturiy ta’minot, Visual Studio, malumotlar bazasi, loyiha, dastur kodi.

---

### МЕТОД РАЗРАБОТКИ ВЕБ-ПРИЛОЖЕНИЯ ДЛЯ РАСПОЗНАВАНИЯ ЛИЦ (FACE ID)

**Ярашов Шохрух**

Навоийский государственный университет, Узбекистан

**Аннотация:** В данной статье анализируются проблемы создания программного обеспечения для распознавания лиц (FACE ID) и исследования ученых в этой области. Также в статье представлен метод разработки веб-приложения для распознавания лиц (FACE ID).

**Ключевые слова:** FACE ID, веб-приложение, программное обеспечение, Visual Studio, база данных, проект, программный код.

---

### A METHOD FOR DEVELOPING A WEB APPLICATION FOR FACE RECOGNITION (FACE ID)

**Yarashov Shokhrukh**

Navoi State University, Uzbekistan

**Abstract:** This article analyzes the challenges of creating facial recognition software (FACE ID) and research in this field. It also presents a method for developing a web application for facial recognition (FACE ID).

**Key words:** FACE ID, web application, software, Visual Studio, database, project, program code.

**Kirish.** So‘nggi yillarda raqamli texnologiyalar va sun‘iy intellekt texnologiyalari jadallik bilan rivojlanib foydalanuvchilarni ishonchli autentifikatsiya qilish borasida dolzarb muammlolar paydo bo‘lmoqda. Bundan kelib chiqadiki odatiy login va parol bilan foydalanuvchilar o‘z akkountlariga kirishda bir qator kamchiliklar vujudga keltirib ulani o‘g‘rilab olishi, unutib qo‘yishi yoki ularning ruxsatisiz boshqa shaxslar foydalanishi, foydalanish xavfini yuzaga keltirib chiqaradi Xusainov N.O[8].

Biometrik inson yuzini aniqlash kamera orqali olingan rasmdan yuzni aniqlab olish tanib olish texnologiyalar ishonchliligi bilan bir qatorda tezkorligi hamda aniqligi juda yuqoriligi bilan ajralib turadi. Face id texnologiyasi inson yuzidan ajratib olingan 128 o‘lchamli vektor asosida yuzni aniqlash imkoniyatini beradi va buni real vaqtda amalga oshirish imkoniyatiga ega bo‘lib jarayonni ishonchliligni oshirishga xizmat qiladi. Ushbu texnologiyani oliy ta’lim muassasalarida masofaviy ta’lim platformalarida va tashkilotlarda axborot tizimlarida qo‘llashimiz ko‘rsatilayotgan odamlarning aynan o‘zi ekanligini tasdiqlash imkoniyatini beradi va xavfsizligini ta’minlashga keng miqyosida xizmat qiladi A. B. Sallow[9].

Mazkur tadqiqot ishida Face id texnologiyasiga asoslangan autentifikatsiya qilish tizimini loyihalash masalasi ko‘rib chiqiladi. Tizimni frontend qismi Angular platformasi asosida ishlab chiqilib, backend qismi ASP.NET Core muhitida ishlab chiqish taklif etilgan va amalga oshirilgan. Yuzni tasvini ajratib olish va uni 128 o‘lchamli vektorga aylantirish hamda uni ma’lumotlar bazasiga saqlash hamda uni solishtirish orqali foydalanuvchini aniqlash amalga oshirish ko‘zda tutilgan.

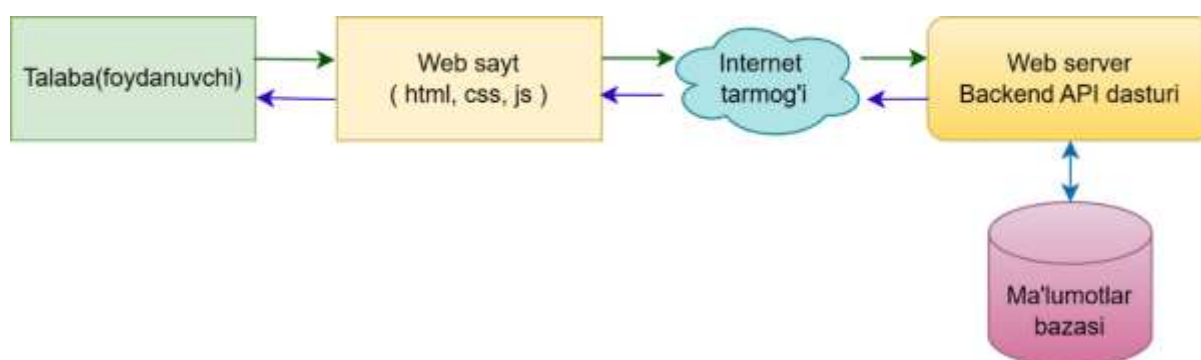
**Adabiyotlar tahlili.** Inson yuzini aniqlash va uni ma’lumotlar omboriga saqlab aniqlash va tanib olish texnologiyalariga oid tadqiqotlar J.X. Xolmuratov[1], F.A. Umirzoqova[2], B.W. Mugalu [3] R.C. Wamala[3], J. Serugunda[3], A. Katumba[3], S.S. Radjabov[4], I.M. Rabbimov[4], N.O. Xusainov[4], B.M. Abdullayeva[4], kabi olimlar tomonidan izlanishlar olib borilgan. Shuningdek, yuzni tanishda neyro tarmoqlarni o‘qitish bilan bog‘liq muammolariga oid izlanishlarni Sh.Nurmuradova[5], Sh.Yarashov[5] va O.Goziev[5] kabi olimlarning ishlarida keltirilgan.

Yoqorida keltirilgan olimlarning ishlarini tahliliga ko‘ra, inson yuzini aniqlash bo‘yicha dasturiy ta‘minotni (face id) ishlab chiqish muhim va aniqlik jihatdan yuqori aniqlikda ishlaydigan texnologiyalardan biri sifatida e’tirof etish lozim degan xulosaga kelindi va buni yanada rivolantirib ta’limda qo‘llash muhim sanaladi.



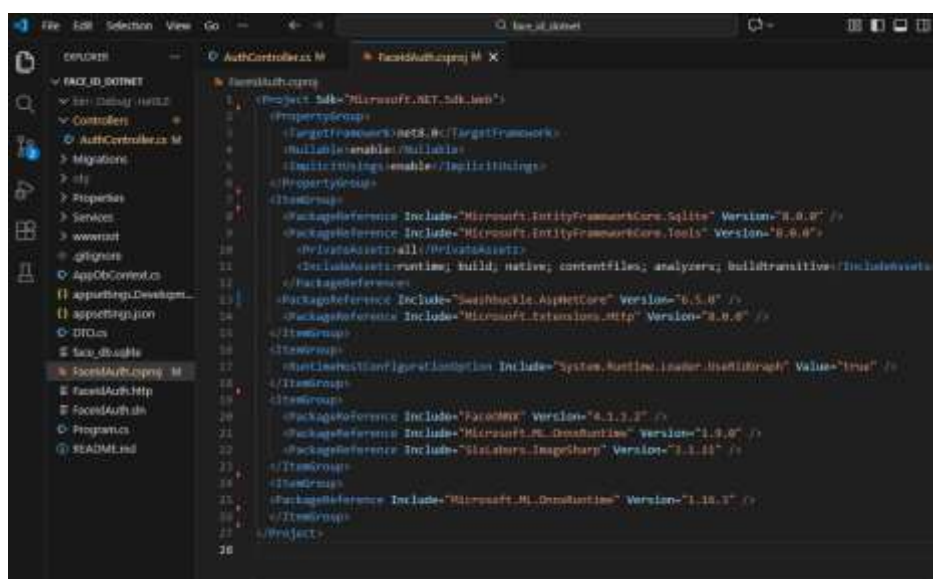
**Tadqiqot metodologiyasi.** Inson yuzini aniqlash va undagi ma’lumotlari ma’lumotlar bazasiga saqlash bo’yicha texnologiyalar va dasturiy ta’minotni (face id) ishlab chiqish uchun C# dasturlash tilini ONNX(Open Neural Network Exchange) neyo tarmog’i kutubxonasidan, SixLabors.ImageSharp kutubxonasidan rasmlarga ishlov va boshqa yordamchi kutubxonalar va texnologiyalardan dasturni ishlab chiqishda o’rnatish kerak bo’ladi.

Tizim arxetikturasini ishlab chiqish va texnik vazifalarini to’g’ri qo’yib olish. Biz yaratayotgan tizim arxitekturasini quyidagicha (1-rasm) ko’rinishga ega bo’ladi.



### 1-rasm. Qurilayotgan FaceId autentifikatsiya tizimini arxetikturasini

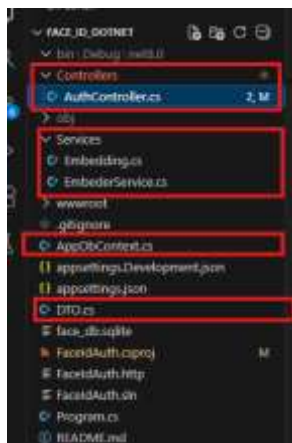
Taklif tuzilma asosida C#, Asp.NET Core 8, Angular(NodeJS bilan) va VS Code ga **extension** o’rnatilish talab qilinadi (2-rasmga qarang).



### 2-rasm. C# bilan ishlash uchun va yaratilgan Asp.net core 8 va unga o’rnatilgan kutubxonalar ko’rsatilgan VS Code dasturi

Bundan so‘ng, kompyuterni CMD (command line) terminalini ochamiz va unga aspnet loyiha yaratish uchun *dotnet new webapi -o face\_id\_dotnet* nom yoziladi.

Loyihani Visual Studio Code dasturi yordamida ochib, undga quyidagi (3-rasmda) belgilangan fayllarni rasmda ko‘rsatilgan tartibda yaratib olamiz.



**3-rasm. Loyihadagi yaratish uchun belgilangan fayllar va papkalar.**

Endilikda biz ushbu yaratgan fayllarni vazifasi va undagi kerakli kodlarni joylashtiramiz.

1. Program.cs faylini ichiga quyidagi kodlarni yozamiz:

```
var builder = WebApplication.CreateBuilder(args);
builder.Services.AddControllers();
builder.Logging.AddConsole();
builder.Logging.AddDebug();
builder.Services.AddEndpointsApiExplorer();
builder.Services.AddSwaggerGen();
builder.Services.AddDbContext<AppDbContext>(options =>
options.UseSqlite(builder.Configuration.GetConnectionString("DefaultConnection"))
);
builder.Services.AddCors(options =>
{
options.AddDefaultPolicy(policy =>
```

*policy*

*.AllowAnyOrigin()*

*.AllowAnyHeader()*

*.AllowAnyMethod());*

*});*

*var app = builder.Build();*

*app.UseDeveloperExceptionPage();*

*app.UseHttpsRedirection();*

*app.UseAuthorization();*

*app.UseStaticFiles();*

*app.MapControllers();*

*app.Run();*

2. Frontendan kelgan rasm kontrollerlarga keladi yani web api loyihlarda frontemddan kelgan ma’lumotlar kontrollerda tutib olinadi va keyingi navbatda boshqa joyga yo’naltirishi mumkin bo’ladi. Ushbu metodni quyidagicha yozamiz AuthController.cs deb nomlangan va ushbu classni ichiga quyidagi dastur kodi yoziladi:

*[ApiController]*

*[Route("v1/[controller]")]*

*public class AuthController : ControllerBase*

*{*

*private readonly IConfiguration \_config;*

*private readonly HttpClient \_httpClient;*

*private readonly AppDbContext \_context;*

*private readonly IHttpClientFactory \_httpClientFactory;*

*private readonly IWebHostEnvironment \_env;*

*public AuthController(IConfiguration config, AppDbContext context,  
IHttpClientFactory httpClientFactory, IWebHostEnvironment env){*

---

```

        _config = config;
        _context = context;
        _httpClient = new HttpClient();
        _httpClientFactory = httpClientFactory;
        _env = env;}
[HttpPost("add")]
public async Task<IActionResult> AnalyzeImage([FromForm]
AuthenticationRequestDto requestDto)
{
    if (requestDto.image == null || requestDto.image.Length == 0)
        return BadRequest("Rasm yuborilmadi.");
    var tempPath = Path.Combine("wwwroot/uploads",
requestDto.image.FileName);
    using (var stream = new FileStream(tempPath, FileMode.Create))
    {await requestDto.image.CopyToAsync(stream);}
    EmbedderService embedder = new EmbedderService();
    var embedding = embedder.GenerateEmbedding(tempPath);
    var results = new List<FaceData>
    {new FaceData
        { Encoding = embedding.ToList(),
          Location = new List<int> {0,0,0,0}}
    };
    if (results == null || results.Count == 0)
        return NotFound("Yuz aniqlanmadi.");
    FaceRecord? data = null;
    foreach (var face in results)
    { var record = new FaceRecord
        { Name = requestDto.Name,

```



```
        FileName = requestDto.image.FileName,
        Location = string.Join(",", face.Location),
        Encoding = string.Join(",", face.Encoding),
        CreatedAt = DateTime.UtcNow };
    _context.FaceRecords.Add(record);
    data = record;
}await _context.SaveChangesAsync();
return Ok(new { success = true, count = results.Count, data =
data?.Name ?? "Name is not found" });
}
[HttpGet("records")]
public async Task<IActionResult> GetAllRecords()
{
    var data = await _context.FaceRecords.ToListAsync();
    return Ok(data); }
[HttpGet("record/{id}")]
public async Task<IActionResult> GetRecord(int id)
{
    var record = await _context.FaceRecords.FindAsync(id);
    if (record == null)
        return NotFound("Ma'lumot topilmadi.");
    return Ok(record); }
[HttpPost("RecognizeFace")]
public async Task<IActionResult> RecognizeFace([FromForm]
CheckDto checkDto)
{
    var image = checkDto.image;
    if (image == null || image.Length == 0)
```

```
        return BadRequest("Rasm yuborilmadi.");
    var    tempPath    =    Path.Combine(Path.GetTempPath(),
image.FileName);
    using (var stream = new FileStream(tempPath, FileMode.Create))
    {await image.CopyToAsync(stream);}
    EmbedderService embedder = new EmbedderService();
    var allRecords = _context.FaceRecords.ToList();
    FaceRecord? matchedRecord;
    double bestDistance = double.MaxValue;
    matchedRecord = embedder.getFaceEmbeddingScore(allRecords,
tempPath);
    if (matchedRecord != null)
        return Ok(new
        {
            success = true,
            message = "Yuz tanildi",
            person = new
            {
                matchedRecord?.Id,
                matchedRecord?.Name,
                matchedRecord?.FileName,
                matchedRecord?.CreatedAt,
                Similarity = (bestDistance)
            }
        });
    return Ok(new { success = false, message = "Bu yuz bazada topilmadi."
});
}}
```

3. Endlikda biz rasmdan yuzni aniqlab olib uni tahrirlaymiz. Tahrirlangan rasmni Emdedding 128 o'lchamli vektorga aylantirib olishimiz kerak. Buning uchun

bizga **EmbedderServiceni** yozishimiz kerak. Buning chun biz EmbedderServicefayl.cs ning ichiga **EmbedderService** quyidagi kodni yozamiz.

```
public class EmbedderService
{
    static FaceDetector faceDetector;
    static Face68LandmarksExtractor faceLandmarksExtractor;
    private static FaceEmbedder _faceEmbedder;
    public EmbedderService()
    {
        faceDetector = new FaceDetector();
        faceLandmarksExtractor = new Face68LandmarksExtractor();
        _faceEmbedder = new FaceEmbedder();
    }
    public float[] GenerateEmbedding(string imagePath)
    {
        using var theImage =
SixLabors.ImageSharp.Image.Load<Rgb24>(imagePath);
        var embedding = GetEmbedding(theImage);
        return embedding; }
    static float[] GetEmbedding(Image<Rgb24> image)
    {
        var array = GetImageFloatArray(image);
        var rectangles = faceDetector.Forward(array);
        var rectangle = rectangles.FirstOrDefault().Box;
        if (!rectangle.IsEmpty)
        {
            var points = faceLandmarksExtractor.Forward(array, rectangle);
            var angle = points.RotationAngle;
            var aligned = FaceProcessingExtensions.Align(array, rectangle, angle);
```

---

```

        return _faceEmbedder.Forward(aligned); }
    return new float[512];
}

public FaceRecord? GetFaceEmbeddingScore(List<FaceRecord> faces,string
score)
{
    var embeddings = new Embeddings(faces);
    using var theImage = SixLabors.ImageSharp.Image.Load<Rgb24>(score);
    var embedding = GetEmbedding(theImage);
    var proto = embeddings.FromSimilarity(embedding);
    var filename = Path.GetFileName(score);
    return proto;    }

static float[,] GetImageFloatArray(Image<Rgb24> image)
{
    var array = new[] {
        new float [image.Height,image.Width],
        new float [image.Height,image.Width],
        new float [image.Height,image.Width] };
    image.ProcessPixelRows(pixelAccessor =>
    {
        for (var y = 0; y < pixelAccessor.Height; y++) {
            var row = pixelAccessor.GetRowSpan(y);
            for (var x = 0; x < pixelAccessor.Width; x++)
            {
                array[2][y, x] = row[x].R / 255.0F;
                array[1][y, x] = row[x].G / 255.0F;
                array[0][y, x] = row[x].B / 255.0F;
            }
        }
    });
    return array; } }

```

4. **Endi biz Embedding** qilingan yani 128 o‘lchamli vektorlarni solishitib o‘xshashligini topishimiz uchun **Emdeddings.cs** fayliga Emdeddings classini yaratib olishimiz kerak. Bu classda 2 xil metod bilan tekshirimiz mumkin yani yuzdan olingan verktorlarni bir biriga o‘xshashligini hisoblashimiz uchun kerak bo‘ladi.

1) Euclidean(512D vector uchun)

2) Cosine(128D vector uchun)

bulardan Educlidean va Cosine orqali o‘xshashligni topshimiz mumkin bo‘ladi.

```
public class Embeddings
{
    public Embeddings(List<FaceRecord> vectors)
    {Vectors = vectors;}
    public int Count{ get{return Vectors.Count;}}
    public List<FaceRecord> Vectors { set;get;}
    public FaceRecord FromDistance(float[] vector)
    {
        var euclidean = new Euclidean();
        var length = Count;
        var min = float.MaxValue;
        var index = -1;
        for (var i = 0; i < length; i++) {
            var storedEncoding = Vectors[i]?.Encoding?
                .Split(',')
                .Select(x => float.TryParse(x, out var val) ? val : 0)
                .ToArray();
            var d = euclidean.Compute(storedEncoding, vector);
            if (d < min)
```



```

        {
            index = i;
            min = d;
        }
    }
    return Vectors[index];}

public FaceRecord? FromSimilarity(float[] vector)
{
    var cosine = new Cosine(true);
    var length = Vectors.Count;
    var max = float.MinValue;
    var index = -1;
    for (var i = 0; i < length; i++)
    {
        var storedEncoding = Vectors[i]?.Encoding?
            .Split(',')
            .Select(x => float.TryParse(x, out var val) ? val : 0)
            .ToArray();
        var d = cosine.Compute(storedEncoding, vector);
        if (d > 0.6)
        {
            index = i;
            max = d;
        }
    }
    if (index == -1)
        return null;
    return Vectors[index]; }

```

Ushbu kodda yozilgan jarayonni quyidagi rasm orqali tushunib olishingiz mumkin. Bunda inson rasmdan qanday qilib 128 o'lchamli vektor hosil qilinishi ko'rsatilgan bunda albatta neyro tarmoq yordamida amlaga oshiriladi.



**4-rasm. Inson yuzidan 128 o‘lchamli vektorga aylantirish.**

5. Ma’lumotlar bazasi bilan bo‘g‘lanib unga ma’lumotlar saqlash va olish uchun xizmat qiladigan fayl hisoblanadi. Malumotlar bazasidagi jadvalimizni shakllantishimiz kerak bu jadval (5-rasm)da foydalanuvchining ma’lumotlari qanday strukturada saqlanagnini ko‘rishimiz mumkin.

| FaceRecord(foydalanuvchi ma'lumotlari) |  |
|----------------------------------------|--|
| Id(int)                                |  |
| Name(Text)                             |  |
| Location(Text)                         |  |
| Encoding(Text)                         |  |
| CreatedAt(DateTime)                    |  |

**5-rasm. Bu ma’lumotlar bazasidagi foydalanuvchi jadvalining tuzilish strukturasi**

Id-foydalanuvchini identifikatsiya raqami har bir foydalanuvchi uchun yagona bo‘ladi va butun solardan iborat bo‘ladi. Name- foydalanuvchining ismi familiyasi yani foydalanuvchining ismi desak ham bo‘ladi. Location-tizimga birinchi marta foydalanuvchining ro‘yhatdan o‘tkaziklayotganda chiqadigan rasmi yuklanadi uni tizim saqlab qoladi va o‘sha saqlangan rasm joylashgan joyi xotiradagi addressi bo‘ladi.

Encoding – bu yuzdan ajratib olingan 128 o'lchamli embedding vektorini saqlash uchun matn. Bu foydalanuvchinig asosiy autentifikasiya ma'lumoti hisoblanadi. Ushbu ma'lumot orqali foydalanuvchi tizimga kiradi(embedding vektorga qiymatiga misol: [0.023, -0.118, 0.982, ... , 0.114] ).

CreatedAt – bu yaratilgan sanasi yani foydalanuvchini tizimga qo'shilgan sanasi hisoblanadi.

AppDbContext.cs faylini ichiga quyidagi faylni yozamiz:

```
public class AppDbContext : DbContext
{
    public AppDbContext(DbContextOptions<AppDbContext> options)
        : base(options){}
    public DbSet<FaceRecord> FaceRecords { get; set; }
    protected override void OnModelCreating(ModelBuilder modelBuilder)
    {base.OnModelCreating(modelBuilder);
    modelBuilder.Entity<FaceRecord>(entity =>
        { entity.Property(e => e.Location).HasColumnType("TEXT");
          entity.Property(e => e.Encoding).HasColumnType("TEXT");
          entity.Property(e => e.CreatedAt).HasColumnType("TEXT");
        });}
}

public class FaceRecord
{
    [Key]
    public int Id { get; set; }

    public string? FileName { get; set; }
    public string? Name { get; set; }
    public string? Location { get; set; }
    public string? Encoding { get; set; }
    public DateTime CreatedAt { get; set; } = DateTime.UtcNow;}
public class FaceData
```

```
{  
    public List<float> Encoding { get; set; }  
    public List<int> Location { get; set; }  
}
```

Frontend qismi, bu foydanuvchi ko‘radigan qismi hisoblanib unda rasmga olish va uni backend qismiga yuborish uchun xizmat qiladigan dastur tuzish kerak bo‘ladi. Buning uchun biz birinchi o‘rinda kamera bilan ishlaydigan service yaratishimiz kerak bo‘ladi. Bunda service kamerani ishga tushiradi va tasvirni boshqa bir komponentga uzatadi, component esa uni foydalanuvchiga ko‘rsatadi hamda uni rasm tushirish uchun `captureImage` metodi mavjud bo‘lib ushbu metod rasmga oladi.

Bu dastur uchun biz JavaScript ning Angular frameworkini tanlashimiz kerak. Chunki unda kod yozish juda oson va qulay hisoblanadi. Angularda loyiha yaratib olamiz. Angular frameworkda loyiha yaratib uni Visual Studio Code dasturida ochib unda 2 ta sahifa yaratamiz. Face id yordamida kirish uchun login sahifasi va muvaffaqiyatli kirishimiz uchun success sahifasi va kamera bilan rasmga olish uchun `CameraService` fayllarini shakllantirishimiz kerak.

`camera.service.ts` fayliga quyidagi kodni yozamiz:

```
import { Injectable } from "@angular/core";  
@Injectable({ providedIn: "root" })  
export class CameraService {  
    async startCamera(videoElement: HTMLVideoElement) {  
        const stream = await navigator.mediaDevices.getUserMedia({ video: true });  
        videoElement.srcObject = stream;  
        await videoElement.play();  
    }  
    captureImage(videoElement: HTMLVideoElement): string {  
        const canvas = document.createElement("canvas");  
        canvas.width = videoElement.videoWidth;  
        canvas.height = videoElement.videoHeight;  
        const ctx = canvas.getContext("2d")!;  
        ctx.drawImage(videoElement, 0, 0);  
    }  
}
```

```
return canvas.toDataURL("image/png");  
} }
```

Endilikda foydalanuvchiga ko‘rinadigan qismi yaratishimiz kerak. Buning uchun biz login sahifasini yaratib olamiz va bu sahifada camera service yordamida kamerdan olinayotgan videoni ko‘rsatamiz pastiga tugma qo‘shib qo‘yib tugma bosilganda rasmga oladi va keyin uni tekshirish uchun serverda ishlab turga dasturga https so‘rovi orqali jo‘natiladi.

Login componentni 3 ta fayldan tashlik topadi bunda login.ts login.css va login.html fayllari, login.ts fayliga mantiq(logika) yoziladi, login.css fayliga style lar yoziladi va login.html fayliga esa dizayn yoziladi va ularga quyidagilarni yozamiz:

```
login.css  
  
.face-container {  
  display: flex;  
  flex-direction: column;  
  align-items: center;  
  margin-top: 40px;  
}  
  
.circle-frame {  
  width: 260px;  
  height: 260px;  
  border-radius: 50%;  
  border: 6px solid #00ff66;  
  display: flex;  
  justify-content: center;  
  align-items: center;  
  position: relative;  
  overflow: hidden;  
  background: #000;
```

```
}  
video, .captured-img {  
    width: 100%;  
    height: 100%;  
    object-fit: cover;  
}  
.loading-text {  
    font-size: 20px;  
    color: #00ff88;  
    font-weight: bold;  
}  
.capture-btn {  
    margin-top: 20px;  
    padding: 12px 22px;  
    border-radius: 50px;  
    background: #111;  
    color: #fff;  
    border: none;  
    font-size: 18px;  
    cursor: pointer;  
}  
.spinner {  
    width: 22px;  
    height: 22px;  
    border: 3px solid #fff;  
    border-top-color: transparent;  
    border-radius: 50%;  
    display: inline-block;
```



```
        animation: spin 0.8s linear infinite;
    }
    @keyframes spin {
        100% { transform: rotate(360deg); }}
    login.ts fayli
import { AfterViewInit, Component, ElementRef, ViewChild,NgZone } from
'@angular/core';
import { CommonModule, NgIf } from '@angular/common';
import { Router } from '@angular/router';
import { HttpClient } from '@angular/common/http';
import { AlertService } from '../services/alert_service';
@Component({
    selector: 'app-faceid',
    templateUrl: './login.html',
    styleUrls: ['./login.css'],
    imports: [CommonModule,NgIf]
})
export class Login implements AfterViewInit {
    @ViewChild('video ')
    videoRef!: ElementRef<HTMLVideoElement>;
    loading = false;
    capturedImage: string | null = null;
    errorMessage = "";
    apiUrl = '/api/v1/Auth/recognize';
    constructor(private http: HttpClient, private router: Router,private alert:
AlertService,private zone: NgZone) { }
    async ngAfterViewInit() {
        await this.startCamera();
```

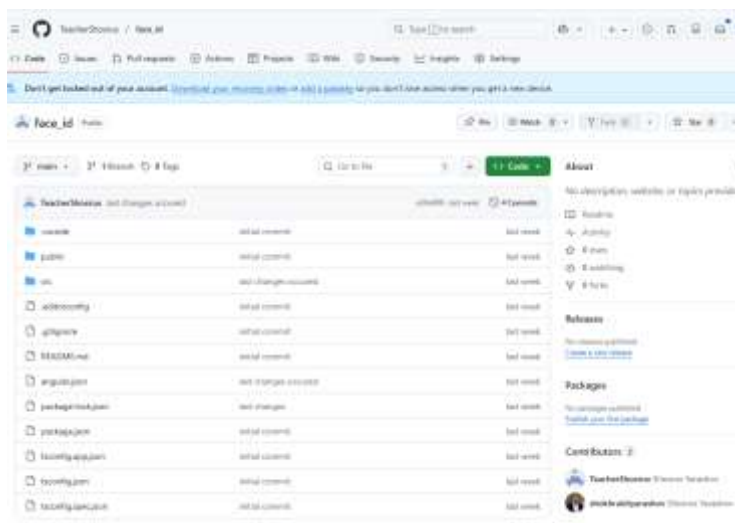
```
}  
async startCamera() {  
  try {  
    const stream = await navigator.mediaDevices.getUserMedia({ video: true });  
    const video = this.videoRef.nativeElement;  
    video.srcObject = stream;  
    await video.play();  
  } catch (e) {  
    console.error("Camera error:", e);  
  } }  
async capture() {  
  if (this.loading) return;  
  this.loading = true;  
  this.errorMessage = "";  
  const video = this.videoRef.nativeElement;  
  const canvas = document.createElement('canvas');  
  canvas.width = video.videoWidth || 640;  
  canvas.height = video.videoHeight || 480;  
  const ctx = canvas.getContext('2d');  
  let imageBlob: Blob;  
  if (ctx) {  
    ctx.drawImage(video, 0, 0, canvas.width, canvas.height);  
    imageBlob = await new Promise(resolve => canvas.toBlob(resolve as any,  
'image/jpeg'));  
  } else {  
    this.loading = false;  
    this.errorMessage = 'Cannot capture image!';  
    return;  
  }  
}
```

```
}  
  
const formData = new FormData();  
formData.append('image', imageBlob, 'face.jpg ');  
this.http.post<any>(this.apiUrl, formData, { observe: 'response' })  
  .subscribe({  
    next: (res) => {  
      this.loading = false;  
      console.log('API  
if (res.status === 200 && res.body?.success) {  
      const person = res.body.person;  
      localStorage.setItem('person', JSON.stringify(person));  
      const capturedImageUrl = URL.createObjectURL(imageBlob);  
      localStorage.setItem('capturedImage', capturedImageUrl);  
      this.router.navigate(['/success'], {  
        });  
    } else  
      if (res.status === 200 && res.body?.success === false) {  
        this.zone.run(() => {  
          this.loading = false;  
          this.showAlert("Yuz topilmadi! Qayta urinib ko 'ring.");  
        });  
      } else if (res.status === 400) {  
        this.errorMessage = 'Face not recognized! Please try again. 400';  
        this.loading = false;  
      }    },  
    error: (err) => {  
      this.loading = false; if (err.status === 404) {  
        this.errorMessage = 'Face not recognized!';
```

```
    } else {  
        this.errorMessage = 'Server error!';  
    }  
}); }  
showAlert(message: string) {  
    alert(message);  
    window.location.reload(); }  
login.html
```

```
<div class="face-container">  
    <div class="circle-frame">  
        <div *ngIf="loading" class="loading-text">Tekshirilmoqda..</div>  
        <video  
            #video  
            *ngIf="!loading"  
            autoplay  
            playsinline>  
        </video>  
    </div>  
    <button class="capture-btn" (click)="capture()">  
        @if (!loading) { <span *ngIf="!loading">Yuzni aniqlash</span> }  
        @if (loading) { <span *ngIf="loading" class="spinner"></span> }  
    </button>  
    @if (errorMessage) { <p class="error">{{ errorMessage }}</p> }  
</div>
```

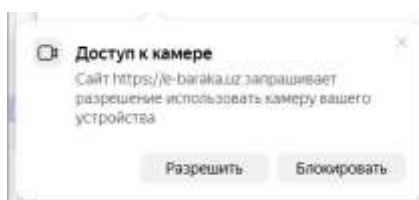
Bunda Success sahifasi yaratilishi kerak, chunki foydalanuvchi ma'lumotlarini ko'rsatish kerak bo'ladi (6-rasm).



**6-rasm.** [https://github.com/TeacherShoxrux/face\\_id](https://github.com/TeacherShoxrux/face_id)

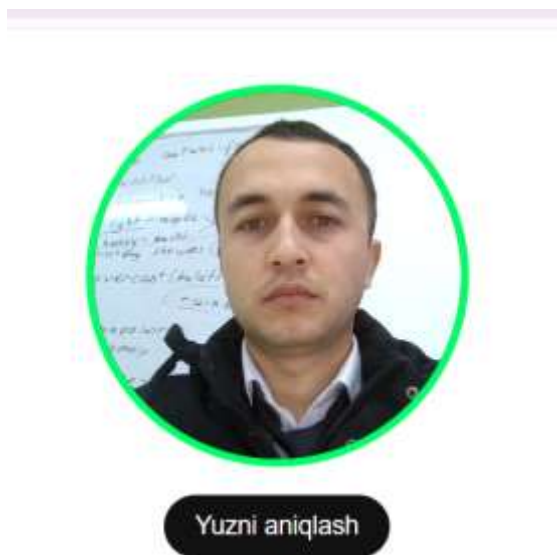
Dastur kodini to'liq variantda github saytidan ko'chirib olishimiz mumkin.

**Tahlil va natija.** Ishlab chiqilgan web-saytimizni ishga tushiramiz va rasmimizni kamera orqali olib, uni tekshirishimiz uchun oldindan rasm analiz qilingan bo'lishi kerak. Loyihdadagi swagger orqali ma'lumotlarni qo'shib olamiz. Keyingi o'rinda brauzer dasturi orqali unga kirishimiz kerak. Loyihani ishga tushirganimizdan keyin web-sayt foydalanuvchidan kameraga ruxsat(6-rasm) so'raydi ruxsat berishni(razreshit) tanlanadi.



**6-rasm. Brauzer foydanuvchi kameradan foydalanish ruxsat so‘rashi.**

Ruxsat berganimizda ekranda quyidagicha ko‘rinish hosil bo‘ladi (face id sahifasi ochiladi) (7-rasmga qarang):



**7-rasm. Login sahifasini ko‘rinishi.**

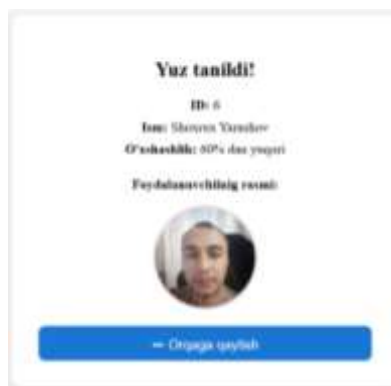
Ushu rasmda ko‘ringan yuzni aniqlash tugmani bosganimizda biz yozgan mantiq(logika) ishga tushib rasmga oladi va rasm tekshiriladi (8-rasm).



**8-rasm. Ma’lumotlarni tekshirish jarayoni.**

Yuz ma’lumotlari avvaldan tizimga kiritilganligi uchun, quyidagicha ma’lumotlarni olishimiz mumkin va ularni success sahifasi bizga ko‘rsatib beradi.





### 9-rasm. Natijaning ekrandagi holati

**Xulosa va takliflar.** Taklif etilayotga yuz orqali autentifikatsiya g'oyasi asosida uning dasturi ishlab chiqildi. Uning masofaviy ta'lim yo'nalishida o'qiydigan talabalar yuzi(face id) orqali masofaviy ta'lim platformasiga kirishi mumkin bo'ladi va bu kirish jarayonini osonlashtirishga hamda yangi zamonaviy texnologiya asosida xavfsizlikni oshirishga xizmat qiladi. Yuzni to'g'ri olishi ya'ni yuzni aniqlab olishi yorug'likka bog'liq ravishta o'rtacha holatda 85-95% bo'ladi va rasm bilan ishlashda ONNX kutubxonasining tezligi 0.5-1.5 soniyada amalaga oshirildi. Umumiy tizimning aniqlik darajasi ya'ni texnik samaradorlik o'rtacha hisobda 85-90% deb baholandi.

### Adabiyotlar

1. Xolmuratov J. X. “Biometrik autentifikatsiya tizimlarining zamonaviy qo'llanilishi – barmoq izi, yuzni aniqlash va ovoz tanish texnologiyalari” // Journal of Multidisciplinary in Science and Education, [Vol. 1 No. 2 \(2025\).](#) – B. 56-67.
2. Umirzoqova Farida Abdulla qizi “Biometrik autentifikatsiya usullarini o'rganish” // Ilmiy tadqiqotlar va ularning yechimlari jurnali, [Vol. 2 No. 2 \(2024\).](#) – B. 33-35.
3. Ben Wycliff Mugalu, Rodrick Calvin Wamala, Jonathan Serugunda, Andrew Katumba (2021). Face Recognition as a Method of Authentication in a Web-Based System // Computer Science, Computer Vision and Pattern Recognition, 2021-03.
4. Radjabov Sobirjon Sattorovich, Rabbimov Ilyos Mehriddinovich, Xusainov Narimon Odilovich, Abdullayeva Barno Mo'ydinjon qizi, “Biometrik

texnologiyalarni rivojlantirishning istiqbolli yo‘nalishlari” // “Al-Farg‘oniy avlodlari” elektron ilmiy jurnali (Vol: 1 | Iss: 3 | 2024). – [B. 68-72.](#)

5. V. Pavlevski “Image Face Detection with C# NET”// “DevInDeep” January 31, 2021.

6. Shakhnoza Nurmuradova, Shokhrukh Yarashov, Olmos Goziev, “Efficient training of deep neural networks on resource-constrained devices for industrial applications” // International Conference on Optical and Computational Technologies for Measurements and Industrial Applications (OptiComp 2025), Volume 13803.

7. Xusenov M. Z. “Biometrik autentifikatsiya va identifikatsiya tizimlarini ishlab chiqish hamda biometrik identifikatsiya mezonlari” // Ta'lim innovatsiyasi va integratsiyasi 7-son\_4-to‘plam\_Mart -2024. -B.161-167.

8. Xusainov N.O, Jo‘rayev Sh.U. Abdullayeva B.M. “Shaxsni yuz tasviri bo‘yicha biometrik identifikatsiya qilishga asoslangan kirishni nazorat qilish va boshqarish tizimlar” // Development of science ilmiy jurnali NOYABR/11(2024) Volume/ 1 .-B. 353-362.

9. Sh. T. Yarashov, O. S. G‘oziyev “Flutter va python asosida interaktiv o‘quv ilovalarini yaratish va ishlab chiqarish jarayonlariga moslashtirish”, Ustozlar uchun jurnali. Vol. 74 No. 1 (2025). -B 186-189.

10. Ramadan TH. Hasan, A. B. Sallow, Face Detection and Recognition Using OpenCV // Journal of soft computing and data mining Vol.2 N.1(2021). -B 86-97.