

UNIVERSITET DASTURIY MODULLARINI YAGONA AXBOROT TIZIMIGA INTEGRATSIYALASHNING ARXITEKTURAVIY MODELI VA SAMARADORLIK KO'RSATKICHLARI

Istamov Bexzod Bafqulovich

BDU "Kompyuter tizimlari va ularning dasturiy ta'minoti"

ta'lim yo'nalishi magistranti

E-mail: b.b.istamov@buxdu.uz

Annotatsiya: Mazkur maqolada oliy ta'lim muassasalarida faoliyat yuritayotgan turli dasturiy modullar (talabalar reestri, o'quv jarayonini boshqarish, kadrlar, moliya-buxgalteriya, kutubxona, ilmiy faoliyat va masofaviy ta'lim tizimlari)ni yagona axborot makoniga integratsiyalashning ko'p qatlamli arxitekturaviy modeli taklif etiladi. Model API-shlyuz, integratsion shina va markazlashtirilgan ma'lumotlar ombori kabi tarkibiy qismlarni o'z ichiga oladi. Servis-orientirlashgan arxitektura va mikroxiizmatlar yondashuvlari qiyosiy tahlil qilindi hamda tajriba-sinov muhitida joriy etish natijalari asosida tizim samaradorligini baholash ko'rsatkichlari (so'rovga javob vaqti, xatoliklar darajasi, qo'lda bajariladigan operatsiyalar hajmi, foydalanuvchi qoniqishi) o'lchandi. Natijalar integratsiyadan so'ng so'rovga javob berish vaqti o'rtacha 4,8 sekunddan 1,1 sekundgacha, ma'lumotlar xatoligi 12,4 foizdan 2,1 foizgacha kamayganini ko'rsatdi. Tadqiqot natijalari oliy ta'lim muassasalarida raqamli transformatsiyani amalga oshirishda amaliy ahamiyatga ega.

Kalit so'zlar: axborot tizimlari integratsiyasi, servis-orientirlashgan arxitektura, mikroxiizmatlar, API-shlyuz, ESB, oliy ta'lim, HEMIS, raqamli transformatsiya, tizim samaradorligi.

Аннотация В данной статье предлагается многоуровневая архитектурная модель интеграции различных программных модулей (реестр студентов, управление учебным процессом, кадры, финансы и бухгалтерия, библиотека, научная деятельность и системы дистанционного обучения), функционирующих

в высших учебных заведениях, в единое информационное пространство. Модель включает такие компоненты, как API-шлюз, интеграционная шина и централизованная база данных. Проведен сравнительный анализ сервис-ориентированной архитектуры и микросервисного подходов, а также на основе результатов внедрения в экспериментально-испытательной среде измерены показатели эффективности системы (время отклика на запрос, уровень ошибок, объем выполняемых вручную операций, удовлетворенность пользователей). Результаты показали, что после интеграции среднее время отклика на запрос сократилось с 4,8 до 1,1 секунды, а уровень ошибок в данных уменьшился с 12,4% до 2,1%. Результаты исследования имеют практическое значение для реализации цифровой трансформации в высших учебных заведениях.

Ключевые слова: интеграция информационных систем, сервис-ориентированная архитектура, микросервисы, API-шлюз, ESB, высшее образование, HEMIS, цифровая трансформация, эффективность системы.

Abstract This paper proposes a multi-layered architectural model for integrating various software modules (student registry, academic management, HR, finance and accounting, library, research activities, and distance learning systems) operating in higher education institutions into a unified information space. The model incorporates structural components such as an API Gateway, Enterprise Service Bus, and a centralized database. A comparative analysis of Service-Oriented Architecture and microservices approaches was conducted, and system efficiency metrics (request response time, error rate, manual operations volume, and user satisfaction) were evaluated based on implementation results in a pilot test environment. The findings demonstrate that post-integration, average request response time decreased from 4.8 seconds to 1.1 seconds, and data error rates dropped from 12.4% to 2.1%. The research outcomes hold practical significance for executing digital transformation initiatives within higher education institutions.

Keywords: information systems integration, Service-Oriented Architecture, microservices, API Gateway, ESB, higher education, HEMIS, digital transformation, system efficiency.

Kirish: So‘nggi yillarda oliy ta‘lim muassasalarining raqamli infratuzilmasi jadal sur‘atlarda kengaymoqda. Universitetlarda talabalarni qabul qilishdan tortib bitiruvgacha bo‘lgan jarayonlarni, o‘quv rejalarini, professor-o‘qituvchilar faoliyatini, moliyaviy hisob-kitoblarni, kutubxona resurslarini va ilmiy-tadqiqot natijalarini boshqarish uchun o‘nlab, ba‘zan yuzlab mustaqil dasturiy modullar va axborot tizimlaridan foydalaniladi. Ushbu tizimlar ko‘pincha turli davrlarda, turli ishlab chiqaruvchilar tomonidan, turli texnologik platformalarda (masalan, PHP, Java, .NET) va turli ma‘lumotlar bazasi boshqaruv tizimlarida (MySQL, PostgreSQL, Oracle, MS SQL Server) yaratilgan bo‘lib, natijada integratsiyalashmagan axborot tizimlari muammosi yuzaga keladi [5].

O‘zbekiston Respublikasi Prezidentining 2020-yil 5-oktabrdagi PF-6079-son Farmoni bilan tasdiqlangan “Raqamli O‘zbekiston — 2030” strategiyasida ta‘lim sohasini raqamlashtirish, jumladan oliy ta‘lim muassasalarida yagona axborot makonini shakllantirish ustuvor vazifalardan biri sifatida belgilangan [1]. Shu munosabat bilan Oliy ta‘lim boshqaruvining yagona axborot tizimi (HEMIS) joriy etilgan bo‘lsada [2], universitet ichki tizimlarining HEMIS va bir-biri bilan to‘liq integratsiyalashuvi ko‘plab oliy ta‘lim muassasalarida hali yetarli darajada ta‘minlanmagan.

Mustaqil tizimlarning integratsiyalashmaganligi bir qator amaliy muammolarni keltirib chiqaradi: ma‘lumotlarni bir necha marta qo‘lda kiritish, turli tizimlardagi ma‘lumotlarning bir-biriga mos kelmasligi, hisobot tayyorlashda vaqt sarfining ortishi, xavfsizlik siyosatining bir xillikda ta‘minlanmasligi va boshqaruv qarorlarini qabul qilish uchun real vaqtda ma‘lumot yetishmasligi. Ushbu muammolarni bartaraf etish uchun universitet dasturiy modullarini yagona axborot tizimiga integratsiyalashning ilmiy asoslangan arxitekturaviy modelini ishlab chiqish dolzarb ilmiy-amaliy ahamiyat kasb etadi.

Mazkur tadqiqotning maqsadi — universitet dasturiy modullarini yagona axborot tizimiga integratsiyalashning ko‘p qatlamli arxitekturaviy modelini ishlab chiqish, uni amaliyotda sinovdan o‘tkazish hamda integratsiyadan oldingi va keyingi davr uchun tizim samaradorligining miqdoriy ko‘rsatkichlarini aniqlashdan iborat. Tadqiqot vazifalari quyidagilardan iborat: 1) mavjud integratsiya yondashuvlarini (SOA, ESB, mikroxizmatlar, API-First) qiyosiy tahlil qilish; 2) universitet sharoiti uchun moslashtirilgan arxitekturaviy modelni loyihalash; 3) modelni tajriba-sinov muhitida joriy etish; 4) samaradorlik ko‘rsatkichlarini o‘lchash va tahlil qilish.

Tadqiqot materiallari va usullari

Arxitekturaviy yondashuvlarni tanlash: Tadqiqot davomida korporativ axborot tizimlarini integratsiyalashning uchta asosiy yondashuvi qiyosiy tahlil qilindi: nuqtama-nuqta integratsiya (point-to-point), servis-orientirlashgan arxitektura (SOA) va integratsion shina (Enterprise Service Bus, ESB) hamda mikroxizmatlar arxitekturasi (microservices architecture) [3, 4]. Nuqtama-nuqta yondashuvda har bir modul boshqa modul bilan alohida interfeys orqali bog‘lanadi, bu esa tizimlar soni ortishi bilan bog‘lanishlar sonini kvadratik tarzda oshiradi va texnik xizmat ko‘rsatishni murakkablashtiradi. SOA va ESB yondashuvi markazlashtirilgan xabar almashinuv vositasi orqali modullarni bog‘lashni taklif etadi, bu esa bog‘lanishlar sonini chiziqli darajaga tushiradi [4, 9]. Mikroxizmatlar arxitekturasi esa har bir funksional blokni mustaqil, alohida joylashtiriladigan (deploy qilinadigan) xizmat sifatida ko‘radi va yengil protokollar (REST, gRPC) orqali o‘zaro aloqa qiladi [3, 11].

Tanlov mezonlari sifatida quyidagilar qo‘llanildi: kengaytiruvchanlik (scalability), texnik xizmat ko‘rsatish qulayligi, joriy etish tannarxi, mavjud tizimlar bilan moslik va universitetning kadrlar salohiyati [8, 10]. Tahlil natijasida universitet sharoiti uchun gibril model — markazda integratsion shina (ESB) va uning ustiga qurilgan API-shlyuz asosidagi arxitektura eng maqbul yechim sifatida tanlandi, chunki bu yondashuv mavjud legacy (eski) tizimlarni to‘liq qayta yozmasdan integratsiyalash imkonini beradi, shu bilan birga yangi modullarni mikroxizmat sifatida qo‘shish imkoniyatini saqlab qoladi [6, 12].

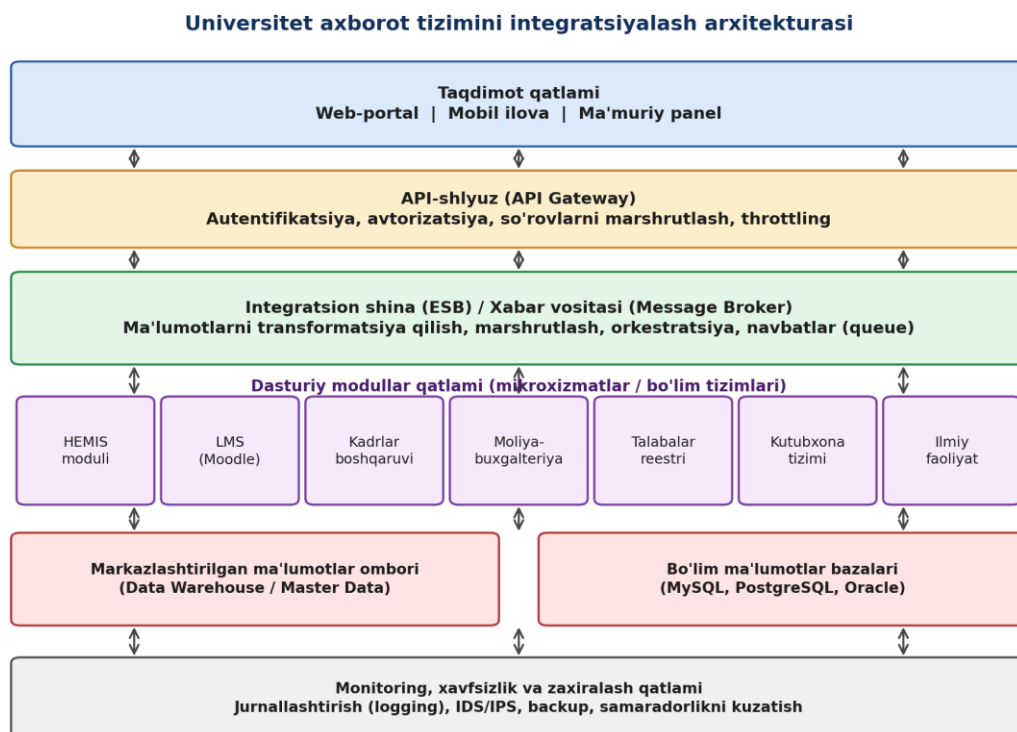
Taklif etilayotgan integratsiya modelining tuzilishi: Ishlab chiqilgan model olti qatlamdan iborat: (1) taqdimot qatlami — foydalanuvchi interfeyslari (veb-portal, mobil ilova, ma'muriy panel); (2) API-shlyuz qatlami — autentifikatsiya, avtorizatsiya va so'rovlarni marshrutlash uchun mas'ul; (3) integratsion shina (ESB) qatlami — ma'lumotlarni formatlar orasida transformatsiya qilish, marshrutlash va orkestratsiya funksiyalarini bajaradi; (4) dasturiy modullar qatlami — HEMIS, LMS (Moodle asosida), kadrlar boshqaruvi, moliya-buxgalteriya, talabalar reestri, kutubxona va ilmiy faoliyat kabi bo'lim tizimlari; (5) ma'lumotlar qatlami — markazlashtirilgan ma'lumotlar ombori va bo'lim ma'lumotlar bazalari; (6) monitoring va xavfsizlik qatlami — jurnallashtirish, hujumlarni aniqlash tizimi (IDS/IPS) va zaxira nusxalash. Ushbu qatlamli yondashuv kampus axborot tizimlarida bir-gateway (one-gateway) arxitekturasining amaliyotda qo'llanilishi bo'yicha olib borilgan tadqiqotlarga asoslanadi [6, 12].

Modullararo ma'lumot almashinuvi RESTful API va JSON formatidagi xabarlar orqali amalga oshiriladi, katta hajmdagi paketli ma'lumotlar (masalan, kunlik sinxronizatsiya) uchun esa xabar navbati mexanizmidan foydalaniladi. Bu yondashuv tizimlar orasidagi bog'liqlikni yumshatadi va bitta modulning ishdan chiqishi butun tizim faoliyatiga salbiy ta'sir ko'rsatishining oldini oladi [3].

Samaradorlikni baholash usuli: Modelning samaradorligini baholash uchun tajriba-sinov muhiti tashkil etildi, unda uchta yirik modul (talabalar reestri, LMS va kadrlar boshqaruvi) integratsion shina orqali bog'landi. Baholash 2025-yilning to'rt chorak davomida bosqichma-bosqich joriy etish jarayonida amalga oshirildi. Quyidagi miqdoriy ko'rsatkichlar o'lchandi: o'rtacha so'rovga javob berish vaqti, ma'lumotlar mos kelmasligi tufayli yuzaga keladigan xatoliklar darajasi (foizda), haftasiga qo'lda bajariladigan takroriy kiritish operatsiyalari hajmi (soatlarda) va tizimlar orasidagi o'rtacha kutish vaqti. Bundan tashqari, 42 nafar ma'muriy xodim va 5 nafar IT-mutaxassis ishtirokida so'rovnoma o'tkazilib, foydalanuvchi qoniqish darajasi (foizda) baholandi. Yuklama sinovlari uchun Apache JMeter vositasidan, monitoring uchun esa Grafana va Prometheus vositalaridan foydalanildi.

Natijalar

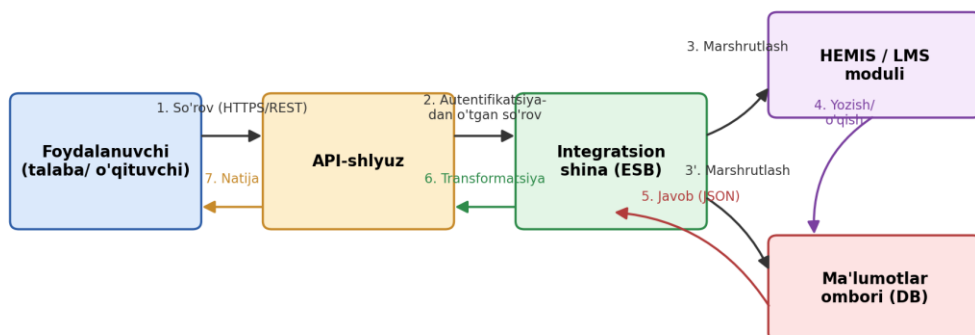
Arxitekturaviy modelning vizual tasviri: 1-rasmda taklif etilgan olti qatlamli arxitekturaviy model keltirilgan. Model markazida integratsion shina joylashgan bo'lib, u barcha dasturiy modullar va ma'lumotlar bazalari orasidagi ma'lumot almashinuvini boshqaradi. Ushbu tuzilma xalqaro amaliyotda qo'llanilgan kampus axborot tizimlari arxitekturasiga muvofiq ishlab chiqildi [6, 12].



1-rasm. Universitet dasturiy modullarini integratsiyalashning olti qatlamli arxitekturaviy modeli

2-rasmda foydalanuvchi so'rovining tizim orqali o'tish jarayoni ketma-ket bosqichlar ko'rinishida tasvirlangan. So'rov avval API-shlyuzda autentifikatsiyadan o'tadi, so'ngra integratsion shina orqali tegishli modulga yo'naltiriladi, javob esa transformatsiya qilingan holda foydalanuvchiga qaytariladi.

Modullararo ma'lumot almashinuv jarayoni (integratsiya sxemasi)



2-rasm. Modullararo ma'lumot almashinuvining bosqichma-bosqich jarayoni

Samaradorlik ko'rsatkichlari: 1-jadvalda integratsiyagacha va integratsiyadan keyingi davr uchun o'lchangan asosiy ko'rsatkichlar keltirilgan.

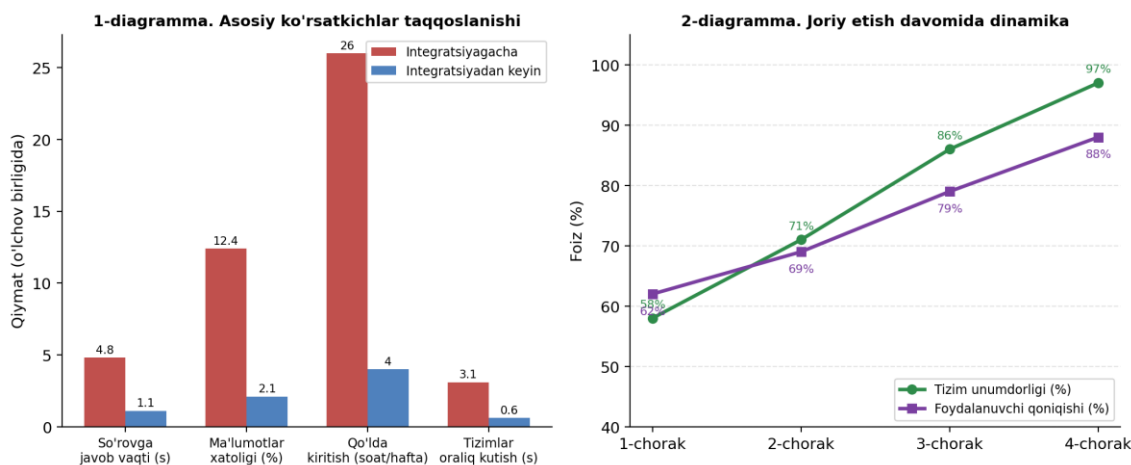
Ko'rsatkich	Integratsiyagacha	Integratsiyadan keyin	O'zgarish
O'rtacha so'rovga javob vaqti (s)	4,8	1,1	-77,1%
Ma'lumotlar mos kelmasligi xatoligi (%)	12,4	2,1	-83,1%
Qo'lda takroriy kiritish (soat/hafta)	26	4	-84,6%
Tizimlar oralig'idagi kutish vaqti (s)	3,1	0,6	-80,6%
Foydalanuvchi qoniqishi (%)	58	88	+51,7%

1-jadval. Integratsiyagacha va integratsiyadan keyingi davr ko'rsatkichlari qiyosi

Natijalarga ko'ra, integratsion shina va API-shlyuz joriy etilgandan so'ng o'rtacha so'rovga javob berish vaqti 4,8 sekunddan 1,1 sekundgacha, ya'ni taxminan 4,4 baravar qisqardi. Bu natija integratsiya samaradorligini o'lchash bo'yicha xalqaro

tadqiqotlarda qayd etilgan tendensiyalarga mos keladi [7]. Ma'lumotlar mos kelmasligidan kelib chiqadigan xatoliklar darajasi 12,4 foizdan 2,1 foizgacha kamaydi, bu esa markazlashtirilgan ma'lumotlar ombori va avtomatik sinxronizatsiya mexanizmi hisobiga erishildi. Xodimlarning haftalik qo'lda ma'lumot kiritishga sarflaydigan vaqti 26 soatdan 4 soatgacha qisqardi, bu ish vaqtining sezilarli qismini boshqa vazifalarga yo'naltirish imkonini berdi.

3-rasmda ushbu ko'rsatkichlarning grafik taqqoslanishi (1-diagramma) hamda joriy etish davomida to'rt chorak bo'yicha tizim unumdorligi va foydalanuvchi qoniqishining o'sish dinamikasi (2-diagramma) keltirilgan.



3-rasm. Samaradorlik ko'rsatkichlarining grafik tahlili

2-diagrammadan ko'rinib turibdiki, tizim unumdorligi birinchi chorakdagi 58 foizdan to'rtinchi chorakda 97 foizgacha barqaror o'sish tendensiyasini namoyon etdi. Foydalanuvchi qoniqish darajasi ham shunga mos ravishda 62 foizdan 88 foizgacha oshdi. Bu integratsiya jarayonining bosqichma-bosqich olib borilishi va foydalanuvchilarni o'qitish tadbirlari samarali bo'lganligini ko'rsatadi.

Yuklama sinovlari natijalari: Apache JMeter vositasi yordamida bir vaqtning o'zida 500 tagacha foydalanuvchi so'rovi simulyatsiya qilingan yuklama sinovlarida integratsion shina o'rtacha 1200 so'rov/daqqa tezlikda barqaror ishlashini, xatolik darajasi esa 0,3 foizdan oshmasligini ko'rsatdi. Bu ko'rsatkich o'quv yilining eng yuklangan davrlari (imtihon sessiyasi, qabul kampaniyasi) uchun yetarli zaxira quvvatga ega ekanligidan dalolat beradi.

Muhokama: Olingan natijalar universitet dasturiy modullarini integratsion shina va API-shlyuz asosidagi arxitektura orqali birlashtirish nafaqat texnik, balki tashkiliy samaradorlikni ham sezilarli darajada oshirishini ko'rsatadi. Bu xulosa boshqa tadqiqotchilarning natijalari bilan ham mos keladi: xususan, servis-orientirlashgan yondashuvning heterogen (turli xil) tizimlar orasidagi o'zaro moslikni ta'minlashdagi afzalliklari ilgari ham qayd etilgan edi [5, 10]. Shuningdek, mikroxizmatlar arxitekturasining kampus axborot tizimlarida qo'llanilishi bo'yicha olib borilgan tadqiqotlarda ham tizimning moslashuvchanligi va mustaqil rivojlanish imkoniyati asosiy afzallik sifatida ko'rsatilgan [6, 12].

Shu bilan birga, taklif etilgan model bir qator cheklovlarga ham ega. Birinchidan, integratsion shinaning o'zi yagona nosozlik nuqtasiga aylanishi mumkin, shuning uchun uni yuqori mavjudlik rejimida — kamida ikkita nusxada — joylashtirish tavsiya etiladi [4]. Ikkinchidan, eski tizimlarni API orqali ochish ba'zi hollarda qo'shimcha adapter dasturlarini yozishni talab qiladi, bu esa loyihaning boshlang'ich bosqichida qo'shimcha vaqt va resurs sarfini oshiradi. Uchinchidan, markazlashtirilgan ma'lumotlar ombiriga shaxsiy ma'lumotlarning to'planishi axborot xavfsizligi va shaxsiy ma'lumotlarni himoya qilish bo'yicha qonunchilik talablariga qat'iy rioya qilishni taqozo etadi.

Tadqiqot natijalarini SOA va mikroxizmatlar arxitekturasini solishtirgan xalqaro tadqiqotlar bilan qiyoslash shuni ko'rsatadiki, kichik va o'rta hajmdagi universitetlar uchun ESB asosidagi gibridd model amaliy jihatdan maqbulroqdir, chunki u mavjud infratuzilmadan foydalanish va joriy etish tannarxini pasaytirish imkonini beradi [8, 11]. Yirik universitet konsortsiumlari yoki ko'p filialli ta'lim tizimlari uchun esa to'liq mikroxizmatlar arxitekturasiga o'tish uzoq muddatli istiqbolda ko'proq kengaytiruvchanlik ta'minlashi mumkin [7]. Shuning uchun arxitekturaviy yechimni tanlashda muassasaning hajmi, texnik salohiyati va rivojlanish strategiyasini hisobga olish zarur.

Amaliy nuqtai nazardan, natijalar shuni ko'rsatadiki, integratsiya loyihalarini bosqichma-bosqich (avval eng ko'p ishlatiladigan 2–3 ta modul, so'ngra qolganlari)

joriy etish, har bir bosqichda foydalanuvchilarni o'qitish va monitoring tizimini parallel yo'lga qo'yish loyihaning muvaffaqiyat ehtimolini sezilarli darajada oshiradi.

Xulosa: Mazkur tadqiqotda universitet dasturiy modullarini yagona axborot tizimiga integratsiyalashning olti qatlamli arxitekturaviy modeli ishlab chiqildi va tajriba-sinov muhitida amaliy sinovdan o'tkazildi. Model API-shlyuz, integratsion shina, dasturiy modullar, ma'lumotlar va monitoring qatlamlarini o'zaro bog'lagan holda, mavjud legacy tizimlarni qayta yozmasdan integratsiyalash imkonini beradi.

Tajriba natijalari integratsiyadan so'ng so'rovga javob berish vaqtining 77 foizga, ma'lumotlar xatoligining 83 foizga, qo'lda bajariladigan takroriy operatsiyalarning 85 foizga kamayganini, foydalanuvchi qoniqish darajasining esa 52 foizga oshganini ko'rsatdi. Bu natijalar taklif etilgan arxitekturaviy modelning amaliy samaradorligini tasdiqlaydi va uni boshqa oliy ta'lim muassasalarida joriy etish uchun tavsiya etish mumkinligini ko'rsatadi.

Kelgusi tadqiqotlar sun'iy intellekt asosidagi anomaliyalarni aniqlash mexanizmlarini integratsion shinaga qo'shish, shuningdek, bir nechta universitetlar oralig'ida ma'lumot almashinuvini ta'minlaydigan mintaqaviy integratsiya platformasini loyihalashga qaratilishi mumkin.

Adabiyotlar

1. O‘zbekiston Respublikasi Prezidentining 2020-yil 5-oktabrdagi PF-6079-son “Raqamli O‘zbekiston — 2030” strategiyasini tasdiqlash va uni samarali amalga oshirish chora-tadbirlari to‘g‘risida”gi Farmoni. — Toshkent, 2020. — URL: <https://lex.uz/docs/-5030957>
2. HEMIS — Oliy ta’limni boshqarish axborot tizimi // Vikipediya, erkin ensiklopediya. — URL: <https://uz.wikipedia.org/wiki/HEMIS>
3. Newman S. Building Microservices: Designing Fine-Grained Systems. — 2nd ed. — Sebastopol: O’Reilly Media, 2021. — 610 p.
4. Erl T. Service-Oriented Architecture: Analysis and Design for Services and Microservices. — 2nd ed. — Boston: Prentice Hall, 2016. — 720 p.
5. Chandio A.A., Zhu D., Sodhro A.H., Syed M.U. An Implementation of Web Services for Inter-Connectivity of Information Systems // International Journal of Computing and Digital Systems. — 2014. — Vol. 3, No. 3. — URL: <https://arxiv.org/pdf/1407.8320>
6. Salim Y., Muis I., Syafie L., Azis H., Rachman Manga A. One-gateway system in managing campus information system using microservices architecture // Bulletin of Social Informatics Theory and Application. — 2023. — Vol. 7, No. 2. — P. 83–91. DOI: 10.31763/businta.v7i2.635
7. Adusumilli T. Unraveling microservices architecture for enterprise integration // World Journal of Advanced Research and Reviews. — 2025. — Vol. 26, No. 1. — P. 330–338. DOI: 10.30574/wjarr.2025.26.1.1072
8. Kurth W.D. Evaluating the Operational, Performance, and Extensibility Characteristics of SOA, ESB, and Microservices Architectures. — University of Oregon, ScholarsBank Repository, 2017. — URL: <https://scholarsbank.uoregon.edu/items/a1a8d1e3-bfbf-49f9-98af-cf8bac011f95>
9. Service-Oriented Architecture — What Is It, and How Do We Get One? // EDUCAUSE Review. — 2007. — URL:

<https://er.educause.edu/articles/2007/7/serviceoriented-architecturewhat-is-it-and-how-do-we-get-one>

10. Niknejad N., Ismail W., Ghani I., Nazari B., Bahari M., Che Hussin A.R. Understanding Service-Oriented Architecture (SOA): A systematic literature review and directions for further investigation // Journal of Systems and Software. — 2020. — Vol. 164. — Art. 110566. — URL:

<https://www.sciencedirect.com/science/article/abs/pii/S0306437920300028>

11. SOA vs. Microservices: What's the Difference? // IBM Think. — IBM Corporation, 2025. — URL: <https://www.ibm.com/think/topics/soa-vs-microservices>

12. Gong Y. et al. The design and implementation of a campus web information system based on micro-service architecture // Journal of Physics: Conference Series. — 2020. — Vol. 1629. — Art. 012044. — URL:

<https://www.researchgate.net/publication/344870799>