Fan olimpiadalarini onlayn o‘tkazish platformasi: Toshkent xalqaro matematika olimpiadasi misolida arxitektura, yuklama sinovi va saboqlar

Akbar Mansurov
Xalqaro Nordik universiteti

Annotatsiya. Tadqiqot Toshkent xalqaro matematika olimpiadasi (TASIMO) uchun yaratilgan onlayn platformani tahlil qilib, fan olimpiadalarini o‘tkazuvchi tizimlar uchun muhandislik xulosalarini chiqarishni maqsad qiladi. Yagona keys dizaynida to‘rtta repozitoriyning manba kodi (230 022 ta bo‘sh bo‘lmagan qator), 435 ta kommitdan iborat versiyalar tarixi va loyiha hujjatlari o‘rganildi, ekspluatatsiya davridagi 10 ta muhim nosozlik va loyihaviy kamchilik tizimlashtirildi, ulardan to‘rttasining mexanizmi qayta hosil qilindi va mahalliy test muhitida yuklama sinovi o‘tkazildi. Platforma — NestJS asosidagi, uchta nusxada ishlaydigan modulli monolit: umumiy holat Redis’da, ma’lumotlar PostgreSQL’da saqlanadi, imtihon sessiyasi WebSocket orqali boshqariladi va idempotent yakunlanadi. Yuklama sinovida 5 000 nafargacha virtual ishtirokchi imtihonni bir vaqtda boshladi: o‘lchangan 1 850 990 ta amalning birortasi ham xato bilan yakunlanmadi, ma’lumotlar yaxlitligi har bir sinovda saqlandi. Uch nusxada 5 000 ishtirokchida boshlash kechikishining 95-persentili 1,54 s, javob tasdig‘iniki 16 ms bo‘ldi; bitta nusxada esa 3 000 ishtirokchida boshlash kechikishining 95-persentili 3,42 s ga yetdi. Nosozliklarning asosiy manbai uchinchi tomon xizmatlari cheklovlari, komponentlararo taymautlar, ko‘p nusxali joylashtirish va mijoz–server shartnomalari bo‘ldi. Bunday platformalar uchun idempotent yakunlash, tashqi chaqiruvlarni foydalanuvchi so‘rovi yo‘lidan chiqarish va to‘liq foydalanuvchi yo‘lini qamrovchi yuklama sinovlari hal qiluvchi ahamiyatga ega.

Kalit so‘zlar: onlayn olimpiada; elektron baholash; WebSocket; yuklama sinovi; idempotentlik; modulli monolit; ekspluatatsiya nosozliklari.

An online olympiad platform: architecture, load testing and lessons from the Tashkent International Mathematics Olympiad

Abstract. This study analyses the online platform of the Tashkent International Mathematics Olympiad (TASIMO) to derive engineering lessons for systems that run academic olympiads. Using a single-case design, we examined the source code of four repositories

CC BY 4.0 https://doi.org/10.5281/zenodo.23947589

(230,022 non-blank lines), the 435-commit version history and the project documentation, catalogued ten significant production incidents and design flaws, reproduced the mechanisms of four of them and ran a load test in a local test environment. The platform is a NestJS modular monolith running as three instances: shared state lives in Redis, data in PostgreSQL, and each exam session runs over WebSocket and is finalised idempotently. In the load test, up to 5,000 virtual participants started an exam simultaneously; none of the 1,850,990 measured operations failed and data integrity held in every run. With three instances and 5,000 participants, the 95th-percentile latency was 1.54 s for starting the exam and 16 ms for acknowledging an answer; with a single instance, the 95th-percentile start latency reached 3.42 s at 3,000 participants. Most incidents stemmed from third-party limits, timeouts between components, multi-instance deployment and client–server contracts. Idempotent finalisation, keeping third-party calls off the user’s request path and load tests covering the whole user journey are essential for such platforms.

Keywords: online olympiad; e-assessment; WebSocket; load testing; idempotency; modular monolith; production incidents.

Kirish

Fan olimpiadalari iqtidorli o‘quvchilarni erta aniqlash va ilmiy faoliyatga yo‘naltirishning sinalgan vositasidir. AQShda matematika, kimyo va fizika olimpiadalarida qatnashgan 345 nafar sobiq olimpiadachining 52 foizi doktorlik darajasini olgani va texnik sohalarda ishlayotgani aniqlangan (Campbell & Walberg, 2011). Xalqaro matematika olimpiadasidagi natija keyingi ilmiy unumdorlikni bashorat qiladi, biroq daromadi past mamlakatlardan chiqqan olimpiadachilar boy mamlakatlardagi xuddi shunday iqtidorli tengdoshlariga qaraganda 34 foiz kam maqola chop etadi va 56 foiz kam iqtibos oladi (Agarwal & Gaule, 2020). Demak, iqtidorli o‘quvchilarning imkoniyatlardan keng va teng foydalana olishi muhim, olimpiadada ishtirok etishni osonlashtiruvchi onlayn infratuzilma esa bu zanjirning birinchi bo‘g‘inidir.

COVID-19 pandemiyasi davrida aksariyat milliy olimpiadalar onlayn shaklga o‘tkazildi. Masalan, Italiya informatika olimpiadasining odatda 500 ga yaqin maktabda qog‘ozda o‘tkaziladigan va 10–15 ming o‘quvchini qamraydigan birinchi bosqichi 2021-yilda onlayn o‘tkazildi; onlayn bosqichlardan birining boshidagi cho‘qqi yuklamada nginx serveridagi so‘rovlar chegarasi noto‘g‘ri sozlangani sababli ulanishlar uzilgan va mualliflar yuklama sinovini zarur deb hisoblagan (Audrito et al., 2021). Oliy ta’limdagi onlayn imtihonlarga oid 36 ta maqola tahlilida ko‘chirmachilik, shaxsni tasdiqlash va xavfsizlik hamda

CC BY 4.0 https://doi.org/10.5281/zenodo.23947589

texnik muammolar asosiy mavzular qatorida ajratilgan (Butler-Henderson & Crawford, 2020); ko‘chirmachilik turlari alohida tizimlashtirilgan (Noorbehbahani et al., 2022), masofaviy kuzatuv (proktoring) esa maxfiylik va adolat nuqtai nazaridan axloqiy savollar tug‘diradi (Coghlan et al., 2021).

O‘zbekistonda «Raqamli O‘zbekiston — 2030» strategiyasi doirasida 2020–2022-yillarda ta’lim va boshqa ijtimoiy sohalarni raqamli transformatsiya qilish belgilangan (O‘zbekiston Respublikasi Prezidentining PF-6079-son Farmoni, 2020), oliy ta’lim misolidagi tadqiqot esa masofaviy ta’lim mamlakat uchun yangi soha bo‘lganini va talabalarga qo‘shimcha ko‘mak zarurligini ko‘rsatgan (Akhmedjanova & Kerimova, 2024). Mahalliy sharoitda foydalanuvchilarga kamida uch tilda (o‘zbek, rus, ingliz) xizmat ko‘rsatish talab etiladi, o‘zbek tili esa kirill va lotin yozuvlarida parallel qo‘llanadi va yozuvlar o‘rtasida o‘girish alohida qoidalarni talab qiladi (Salaev et al., 2022); bu ism-familiyalarni qidirishni ham murakkablashtiradi.

Onlayn imtihonlarning pedagogik va axloqiy jihatlari, shuningdek, yirik tizimlarni yuklama ostida sinash usullari (Jiang & Hassan, 2015) yaxshi o‘rganilgan. Biroq olimpiadaning ro‘yxatdan o‘tishdan sertifikatgacha bo‘lgan to‘liq siklini qamrab oluvchi platformalarning muhandislik yechimlari, ularning minglab ishtirokchi imtihonni bir vaqtda boshlagandagi xatti-harakati va real ekspluatatsiyadagi nosozliklari kamdan-kam hujjatlashtiriladi, Markaziy Osiyo sharoiti uchun esa bunday hisobotlar deyarli uchramaydi.

Ushbu maqolaning maqsadi — Toshkent xalqaro matematika olimpiadasi (TASIMO) platformasining arxitekturasi, yuklama ostidagi xatti-harakati va ekspluatatsiya nosozliklarini empirik o‘rganib, shunga o‘xshash tizimlar uchun umumiyroq muhandislik xulosalarini chiqarishdir. Tadqiqot uchta savolga javob izlaydi:

1) qaysi arxitektura qarorlari olimpiadaning to‘liq siklini bitta platforma doirasida qamrab olish imkonini berdi;
2) minglab ishtirokchi imtihonni bir vaqtda boshlaganda imtihon sessiyasi mexanizmi qanday ishlaydi;
3) ekspluatatsiya jarayonida qanday nosozliklar yuzaga keldi va ulardan qanday umumiy saboqlar chiqarish mumkin.

Materiallar va usullar

Tadqiqot dizayni va keys

Tadqiqot yagona holatni real sharoitida, bir nechta ma’lumot manbai asosida o‘rganuvchi keys-tadqiqot dizaynida olib borildi (Runeson & Höst, 2009; Yin, 2018); platforma

CC BY 4.0 https://doi.org/10.5281/zenodo.23947589

muayyan muammoni hal qilish uchun yaratilgan artefakt sifatida baholandi (Hevner et al., 2004). Muallif tahlil qilinayotgan loyihaga bevosita aloqador bo‘lgani sababli tadqiqot ishtirokchi-kuzatuvchi pozitsiyasidan olib borildi; bu talqinlarning xolisligiga ta’sir qilishi mumkinligi uchun xulosalar asosan qayta tekshirish mumkin bo‘lgan artefaktlarga — manba kodi, versiyalar tarixi va takrorlanadigan o‘lchovlarga — tayandi.

TASIMO — O‘zbekiston va xorijdagi 3–11-sinf o‘quvchilari hamda akademik litsey, kollej va texnikumlarning 1–2-kurs o‘quvchilari uchun xalqaro matematika olimpiadasi; olimpiada saytiga ko‘ra (2026-yil 10-oktabr holati), 2026-yilgi mavsum onlayn bosqich (29-mart), oflayn bosqich (19-aprel) va finaldan (18–21-may) iborat bo‘lgan. Platforma ommaviy sayt, ishtirokchi kabineti, boshqaruv paneli va server qismidan — dasturlash interfeysidan (API, Application Programming Interface) iborat; Telegram-bot server qismining moduli sifatida ishlaydi. O‘zbekistondagi ishtirokchilar telefon raqami va Telegram-bot orqali parolsiz, xorijiy ishtirokchilar esa elektron pochta va parol bilan kiradi. Tahlil qilingan tizim platformaning ikkinchi versiyasi bo‘lib, ishchi muhitga (production) 2025-yil dekabr oxirida joylashtirilgan; tahlil davri birinchi kommitdan (2025-yil 26-sentabr) oxirgi tahlil qilingan kommitgacha (2026-yil 9-oktabr) bo‘lgan vaqtni qamraydi.

Ma’lumot manbalari va tahlil

To‘rt turdagi manba bir-biri bilan solishtirildi (triangulyatsiya; Runeson & Höst, 2009). **Manba kodi:** to‘rtta git-repozitoriy (server qismi, sayt, kabinet va panel) mos ravishda 9c55ff8, c6ee968, 942605b va a699de2 kommitlari holatida tahlil qilindi. Hajm ko‘rsatkichlari Python skripti (Python 3.14.5, git 2.54.0) bilan faqat git kuzatuvidagi fayllar bo‘yicha hisoblandi: kod qatori sifatida barcha bo‘sh bo‘lmagan qatorlar, jumladan izoh qatorlari (230 022 tadan 13 854 tasi) olindi, HTTP (HyperText Transfer Protocol) marshrutlari NestJS marshrut dekoratorlari bo‘yicha, WebSocket nomlar fazolari esa @WebSocketGateway e’lonlari bo‘yicha sanaldi va natijalar git, grep va awk buyruqlari bilan qayta tekshirildi. **Versiyalar tarixi:** 435 ta kommitning xabarlari, o‘zgarishlar farqlari va kod izohlari asosida xronologiya tiklandi va foydalanuvchilarga ta’sir qilgan yoki qilishi mumkin bo‘lgan, alohida tuzatish kommitiga ega holatlardan nosozliklar katalogi tuzildi. Har bir holat «alomat — asosiy sabab — yechim — saboq» sxemasi bo‘yicha tizimlashtirildi, imkon bo‘lganda sabab uni qayta hosil qiluvchi kichik skript bilan tekshirildi. **Loyiha hujjatlari:** talqinlarni aniqlashtirish uchun ishlatildi. **Yuklama sinovi:** imtihon mexanizmining bir vaqtdagi yuklama ostidagi ishlashi o‘lchandi (keyingi bo‘lim).

CC BY 4.0 https://doi.org/10.5281/zenodo.23947589

Yuklama sinovi

Sinov bitta ish stansiyasida (Intel Core i9-13900K, 32 GB xotira, Windows 11, Node.js 24.16.0), faqat mahalliy test infratuzilmasida o‘tkazildi: PostgreSQL 17.11 va Redis 7.4.11 Docker konteynerlarida standart sozlamalar bilan ishladi, server qismining 9c55ff8 kommitdagi kodi o‘zgartirishsiz ishlatildi, tashqi xizmatlar (Telegram, to‘lov, pochta) soxta serverlar yoki yopiq portlar bilan almashtirildi va ishchi serverlarga so‘rov yuborilmadi. Asosiy konfiguratsiya ishchi muhitga yaqinlashtirildi: API uchta nusxada, har biri bazaga 20 tagacha ulanish bilan ishladi, nusxalar umumiy Redis orqali bog‘landi, mijozlar nusxalarga navbatma-navbat taqsimlandi (nginx’siz); taqqoslash uchun bitta nusxa ham sinaldi.

Har bir sinov uchun 25 savolli (20 ta bitta javobli va 5 ta sonli javobli) yangi imtihon yaratildi. Virtual ishtirokchilar haqiqiy kabinet foydalanadigan socket.io-client orqali soniyasiga 500 ta ulanish tezligida ulandi; barchasi ulangach imtihon bir vaqtda boshlandi, so‘ng har bir savolga 0–2 soniyalik tasodifiy kutishdan (tasodifiy sonlar generatorining boshlang‘ich qiymati — 20261010) keyin oldindan belgilangan javob yuborildi va imtihon yakunlandi. Kechikish so‘rovdan tasdiqqacha o‘lchandi (yuklama generatorining kutish chegarasi 120 s; haqiqiy kabinet boshlash va yakunlashni 20 s, javobni 10 s kutadi). N = 100, 500, 1 000, 2 000, 3 000, 4 000 va 5 000 (bitta nusxada 1 000, 2 000 va 3 000) darajalari uch martadan takrorlandi, isitish uchun o‘tkazilgan dastlabki sinovlar hisobga olinmadi. Har bir sinovdan keyin har bir ishtirokchida aynan bitta natija va yuborilganiga mos 25 ta javob yozuvi borligi, ball kutilganiga mosligi hamda yakunlash xabarnomasi aynan bir marta yuborilgani tekshirildi. Persentillar eng yaqin rang (nearest-rank) usulida, takrorlashlar birlashtirilib hisoblandi; generator yakunlash tasdig‘idan keyin ulanishni o‘zi yopganda javobi kutilayotgan 28 ta heartbeat so‘rovi o‘lchovdan chiqarildi.

Birinchi o‘lchov seansining bir qismi mashinada boshqa jarayonlar ishlagan vaqtga to‘g‘ri kelgani uchun bu seans tahlildan chiqarildi (begona yuklama ostida o‘tgan bu seansda N = 5 000 da bitta ulanish xatosi qayd etilgan); barcha raqamlar ikkinchi seansdan olingan. Unda har bir konfiguratsiyadan oldin fon yuklamasi o‘lchandi (protsessor quvvatining o‘rtacha 4 foizidan kam), sinov davomida esa begona sinov jarayonlari yo‘qligi har 15 soniyada tekshirildi.

Sun’iy intellektdan foydalanish

Tadqiqot va maqolani tayyorlashda Anthropic kompaniyasining Claude (Opus 5.5) generativ sun’iy intellekt modelidan quyidagilar uchun foydalanildi: manba kodi va versiyalar tarixini tahlil qilish (jumladan, nosozliklarni dastlabki tizimlashtirish), tahlil, nosozliklarni

CC BY 4.0 https://doi.org/10.5281/zenodo.23947589

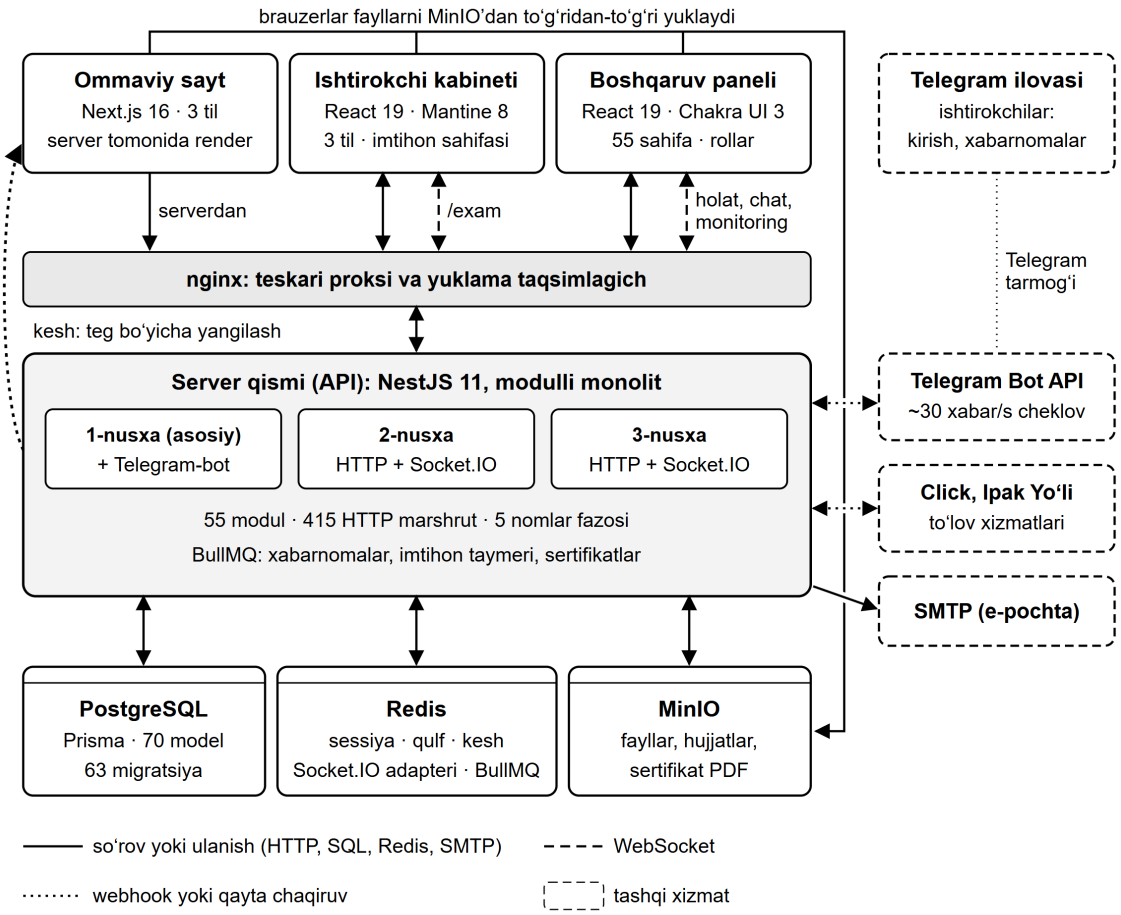
qayta hosil qilish va yuklama sinovi skriptlarini yozish va ishga tushirish, adabiyotlarni izlash, 1–3-rasmlarni tayyorlash (diagrammalarning SVG kodi va grafik skripti) hamda matn qoralamasini yozish. Rasmlar faqat maqoladagi ma’lumotlar asosida chizilgan; platformaga oid raqamlar kod, versiyalar tarixi va o‘lchov skriptlaridan, adabiyotdagi raqamlar manbalarning o‘zidan olingan; bibliografik ma’lumotlar Crossref, rfc-editor.org, lex.uz va nashriyotlarning sahifalari orqali tekshirilgan. Matn, ma’lumotlar va xulosalarning to‘g‘riligi uchun javobgarlik muallif zimmasida. Platformani ishlab chiqishda ham sun’iy intellektga asoslangan dasturlash yordamchilaridan foydalanilgani versiyalar tarixida qayd etilgan.

Natijalar

Platforma tarkibi va arxitekturasi

Platforma ro‘yxatdan o‘tish va hujjat tekshiruvidan tortib olimpiadaga yozilish, to‘lov, onlayn imtihon, natijalarni import qilish va e’lon qilish, sertifikat berish va uni ochiq tekshirishgacha bo‘lgan to‘liq siklni qamrab oladi. Umumiy arxitektura 1-rasmda, tarkibiy qismlar va hajm ko‘rsatkichlari 1-jadvalda keltirilgan.

CC BY 4.0 https://doi.org/10.5281/zenodo.23947589



1-rasm. TASIMO platformasining umumiy arxitekturasi. HTTPS shifrlash nginx oldidagi alohida proksi serverda tugatiladi

Server qismi — NestJS 11 freymvorkida yozilgan, 55 ta moduldan iborat modulli monolit (beshta Socket.IO nomlar fazosidan biri — /mafia — administratorlar uchun ichki o‘yinga tegishli). Ma’lumotlar PostgreSQL bazasida Prisma orqali saqlanadi; ishchi muhitda API PM2 boshqaruvidagi uchta nusxada ishlaydi. Nusxalar o‘rtasidagi muvofiqlik Redis orqali ta’minlanadi: Socket.IO adapteri xabarlarni barcha nusxalarga yetkazadi, BullMQ navbatlarining (xabarnomalar, imtihon taymeri, sertifikatlar) ishlov beruvchilari har bir nusxada ishlaydi, lekin har bir vazifani ulardan faqat bittasi bajaradi, kritik bo‘limlar esa muddatli qulflar bilan himoyalanadi. Telegram webhook’ini ro‘yxatdan o‘tkazish va davriy eslatmalar kabi faqat bitta nusxada bajarilishi kerak bo‘lgan amallarni «asosiy» nusxa bajaradi; webhook uning lokal portiga ko‘rsatilgani uchun Telegram yangilanishlariga ham faqat unda ishlov beriladi.

Ommaviy sayt Next.js 16 asosida serverda shakllantiriladi: brauzer API’ga to‘g‘ridan-to‘g‘ri murojaat qilmaydi, rasm va PDF fayllar esa MinIO omborining havolalari orqali yuklanadi. API javoblari keshda 15 ta teg bilan saqlanadi; administrator ma’lumotni

CC BY 4.0 https://doi.org/10.5281/zenodo.23947589

o‘zgartirganda API saytga maxfiy kalit bilan himoyalangan webhook yuboradi va faqat tegishli teglarga bog‘langan yozuvlar yangilanadi. Teg bo‘yicha yangilash — Next.js imkoniyati, bir soatlik zaxira muddati esa HTTP keshlashdagi yaroqlilik muddatiga mos keladi (Fielding et al., 2022). Kabinet (React 19, Mantine 8) va panel (React 19, Chakra UI 3) bir sahifali ilovalardir; sayt va kabinet uch tilda ishlaydi. Paneldagi 55 ta sahifadan 53 tasiga kirish rol ruxsatlari bilan cheklanadi, ammo bu cheklov faqat interfeysda tekshiriladi; asosiy nazorat server qismidagi administrator marshrutlarida amalga oshiriladi.

1-jadval. Platformaning tarkibiy qismlari va hajm ko‘rsatkichlari

Qism	Asosiy texnologiyalar	Vazifasi	Hajm ko‘rsatkichlari
Server qismi (API)	NestJS 11, Prisma 6, PostgreSQL, Redis, Socket.IO, BullMQ, grammY	Biznes mantig‘i, HTTP (REST) va WebSocket interfeyslari, Telegram-bot, navbatlar, sertifikatlar	97 136 qator 415 HTTP marshrut 5 WebSocket nomlar fazosi 70 model, 63 migratsiya
Ommaviy sayt	Next.js 16, React 19, next-intl	Olimpiada haqida axborot, natijalar, sertifikatni tekshirish	16 408 qator 24 sahifa 3 til × 439 kalit
Ishtirokchi kabineti	React 19, Mantine 8, TanStack Query, Socket.IO mijozi	Ro‘yxatdan o‘tish, olimpiadaga yozilish, to‘lov, onlayn imtihon, natija va sertifikatlar	16 666 qator 14 sahifa 3 til × 693 kalit
Boshqaruv paneli	React 19, Chakra UI 3, TanStack Query, Recharts	Olimpiada, imtihon, natija, foydalanuvchi va statistika boshqaruvi	99 812 qator 55 sahifa 160 ruxsat (43 guruh)
Jami			**230 022 qator** 435 ta kommit (2025-yil 26-sentabr — 2026-yil 9-oktabr)

Izoh: «qator» — git kuzatuvidagi kod fayllaridagi (.ts, .tsx, .js, .mjs, .prisma, .css) bo‘sh bo‘lmagan qatorlar; JSON (JavaScript Object Notation), Markdown, migratsiya SQL fayllari va avtomatik generatsiya qilingan xarita fayllari hisobga olinmagan. Kabinet uchun yo‘naltiruvchi va 404 marshrutlari, boshqaruv paneli uchun kirish va 404 sahifalari sanalmagan. REST — Representational State Transfer. Ko‘rsatkichlar «Materiallar va usullar» bo‘limida keltirilgan kommitlar holatida hisoblangan.

Server qismi, sayt va panel kodi asosiy git-tarmoqqa kiritilishi bilan GitHub Actions orqali avtomatik joylashtiriladi (kabinetda bunday jarayon yo‘q): server qismida migratsiyalar qo‘llanadi, loyiha yig‘iladi, boshlang‘ich ma’lumotlar idempotent tarzda yangilanadi va nusxalar birma-bir qayta ishga tushiriladi — keyingisiga oldingisi HTTP javob bera boshlagach o‘tiladi. Avtomatlashtirilgan testlar deyarli yo‘q: server qismi, sayt va kabinetda test fayllari mavjud emas, panelda esa faqat interfeys mavzusini tekshiruvchi 101 qatorli ikkita test fayli bor.

Onlayn imtihon sessiyasi

Imtihon sessiyasi 2-rasmda ko‘rsatilgan. Mijoz /exam nomlar fazosiga JSON Web Token (JWT) bilan ulanadi, token ulanishda bir marta tekshiriladi. Boshlash so‘rovida server

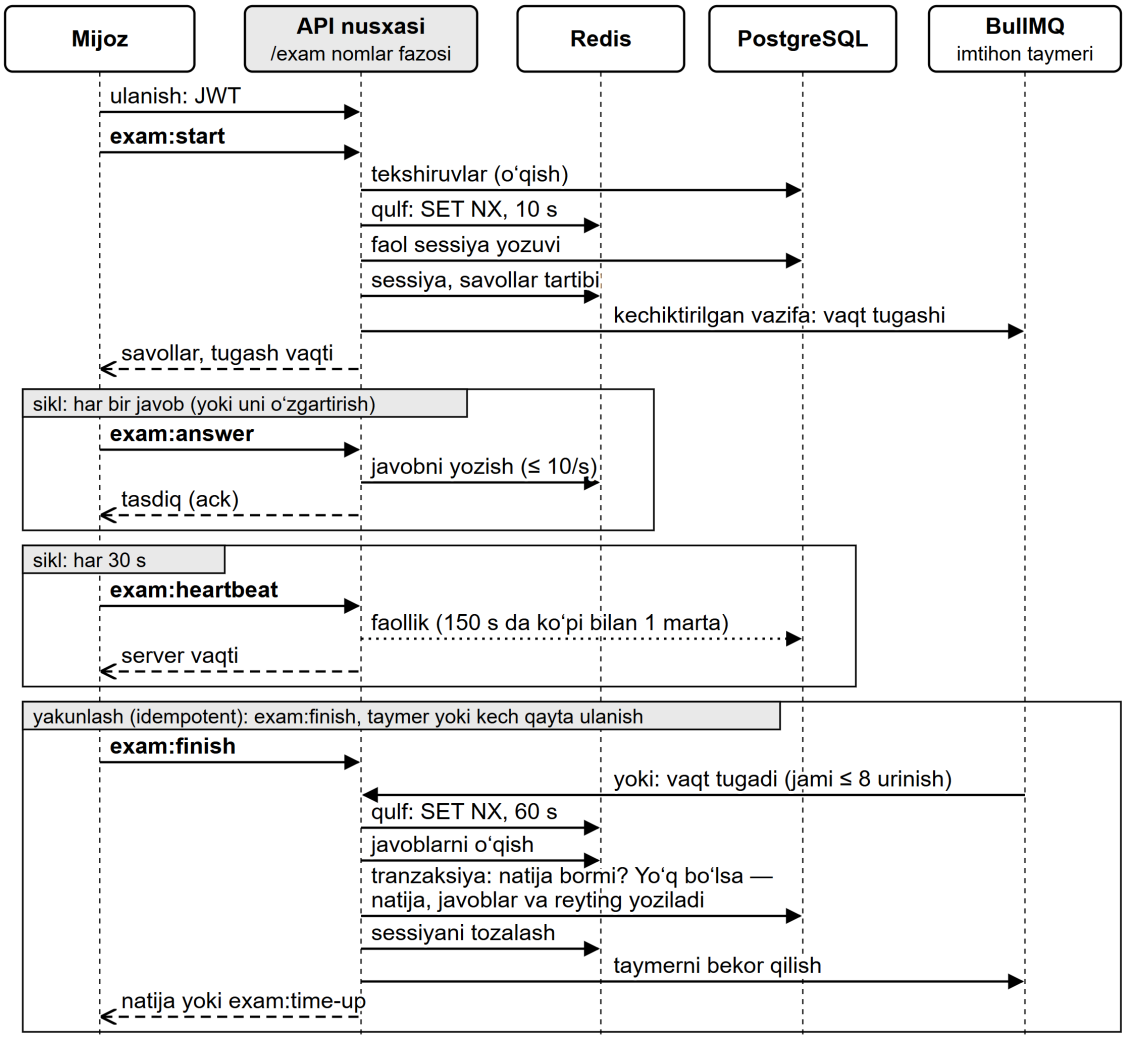
CC BY 4.0 https://doi.org/10.5281/zenodo.23947589

foydalanuvchi tasdiqlanganini, vaqt oynasi ochiqligini (5 soniyalik erta boshlash ruxsati bilan), rol, til va sinf mosligini, ro‘yxatdan o‘tish yozuvi va (agar bo‘lsa) to‘lov holatini hamda imtihon avval topshirilmaganini tekshiradi; so‘ng 10 soniyalik qulf ostida faol sessiya yozuvi bazaga, sessiya va savollar tartibi Redis’ga yoziladi, vaqt tugashi uchun kechiktirilgan vazifa navbatga qo‘yiladi. Boshqa qurilmadan boshlanganda avvalgi qurilma majburan uziladi; savollar teng ball beriladigan guruhlar ichida, variantlar esa har bir ishtirokchi uchun alohida aralashtiriladi.

Javoblar yakunlashgacha faqat Redis’ga yoziladi (foydalanuvchidan soniyasiga ko‘pi bilan 10 ta). Mijoz har 30 soniyada server vaqtini so‘raydi, faollik ma’lumoti bazaga ko‘pi bilan 150 soniyada bir marta ko‘chiriladi. Tugash vaqti boshlashda serverda hisoblanadi va undan keyingi javoblar rad etiladi; vaqt tugaganda sessiyani BullMQ’dagi kechiktirilgan vazifa yakunlaydi (xatoda jami sakkiz martagacha, 5 soniyadan boshlab ikki baravar ortuvchi oraliq bilan urinadi).

Yakunlash ishtirokchi, taymer yoki vaqt o‘tgach qayta ulanish orqali ishga tushirilishi mumkin, shuning uchun u idempotent qilib loyihalangan (Helland, 2012): 60 soniyalik qulf ostida server javoblarni Redis’dan o‘qiydi, bitta tranzaksiya natija allaqachon yozilganini tekshiradi va natija bo‘lmasagina uni, savollar bo‘yicha javob yozuvlarini va reyting ballini saqlaydi. Bazada ishtirokchi–imtihon juftligi uchun noyoblik cheklovi yo‘q, ya’ni takroriy natijadan himoya faqat ilova darajasida. Mijoz javoblarni optimistik navbatga qo‘yib, aloqa tiklanganda qayta yuboradi; tarmoq uzilganda qayta ulanish cheklanmagan va 1–5 soniyalik, ikki baravar ortuvchi, ±50 foiz tasodifiy tebranishli (jitter) oraliqlar bilan bajariladi. Kod tahlili imtihon halolligi mexanizmlarida bir nechta kamchilikni ham aniqladi; ular tuzatilgunga qadar mas’uliyatli oshkor qilish tamoyiliga ko‘ra yoritilmaydi.

CC BY 4.0 https://doi.org/10.5281/zenodo.23947589



2-rasm. Onlayn imtihon sessiyasining soddalashtirilgan ketma-ketlik diagrammasi

Mahalliy sharoitga moslashtirish

Telegram orqali kirish. Bot foydalanuvchiga 7 kun amal qiladigan JWT joylangan shifrlangan havola beradi, server esa havola ochilganda tokenning imzosi va muddatini qayta tekshiradi. Telegram ommaviy jo‘natmalarni taxminan soniyasiga 30 xabar bilan cheklaydi (Telegram, n.d.), shuning uchun «Profilga kirish» tugmasi orqali so‘ralgan havolalar jarayon ichidagi FIFO navbat orqali soniyasiga 25 tadan yuboriladi (boshqa holatlarda havola navbatsiz jo‘natiladi); HTTP 429 («Too Many Requests»; Nottingham & Fielding, 2012) javobida butun navbat ko‘rsatilgan muddatga to‘xtatiladi, takroriy bosishlarni esa Redis’dagi qisqa muddatli kalit filtrlaydi. **To‘lovlar.** O‘zbekistondagi ishtirokchilar uchun Click, barcha ishtirokchilar uchun «Ipak Yo‘li» banki to‘lov xizmati ulangan; xorijiy ishtirokchilar uchun onlayn to‘lov o‘chirilsa, to‘lov olimpiada joyida amalga oshiriladi.

Ikki yozuvli qidiruv. Yagona SQL (Structured Query Language) funksiyasi kirill harflarini lotin harflariga o‘giradi, tutuq belgisi va o‘, g‘ harflarining turli shakllarini yagona

CC BY 4.0 https://doi.org/10.5281/zenodo.23947589

shaklga keltiradi, x va h kabi tez-tez almashtiriladigan harflarni tenglashtiradi hamda saqlanadigan kalitga ham, so‘rovga ham qo‘llanadi. Aniq moslik topilmasa, PostgreSQL pg_trgm kengaytmasining so‘z darajasidagi trigram o‘xshashligi (word_similarity, chegara 0,35) bo‘yicha taxminiy (fuzzy) qidiruv ishlaydi (Navarro, 2001).

Nazorat belgili identifikator. Excel orqali natija importida identifikatordagi bitta belgi xatosi natijani boshqa o‘quvchiga yozib yuborishi, ketma-ket raqamlar esa foydalanuvchilar sonini oshkor qilishi mumkin edi. Shu sababli CC-YY-RRRRRK formati joriy etildi: mamlakat kodi, yil, Crockford Base32 alifbosidagi (Crockford, 2019) beshta tasodifiy belgi va nazorat belgisi (Gallian, 1991):

$$K = A\left[\left(\sum_{i=1}^{9} i \cdot v_i\right) \bmod 31\right] \tag{1}$$

bu yerda A — Crockford Base32 alifbosi, v_i — CC, YY va RRRRR qismlaridagi i-belgining son qiymati (harflar uchun 0–25, yil raqamlari uchun 0–9, tasodifiy belgilar uchun 0–31). Tasodifiy qism kriptografik generator bilan yaratiladi, har bir mamlakat–yil juftligi uchun $32^5 = 33\,554\,432$ ta qiymat mavjud. 20 000 ta tasodifiy identifikatorning har biri uchun bitta belgini almashtirishning barcha variantlari (254 tadan) va qo‘shni belgilarning barcha o‘rin almashinuvlari sanab chiqildi: ularning mos ravishda 99,88 foizi (5 080 000 tadan 6 240 tasi aniqlanmagan) va 99,91 foizi (172 352 tadan 153 tasi aniqlanmagan) aniqlandi. Aniqlanmagan barcha holatlar tasodifiy qismdagi 0 va Z juftligiga to‘g‘ri keldi; bu istisno loyiha rejasida oldindan qayd etilgan edi.

Yuklama sinovi natijalari

Natijalar 2-jadvalda va 3-rasmda keltirilgan. Ikkinchi seansda jami 1 850 990 ta amal o‘lchandi; ularning birortasi ham xato bilan yakunlanmadi va haqiqiy mijozning kutish chegarasidan oshmadi.

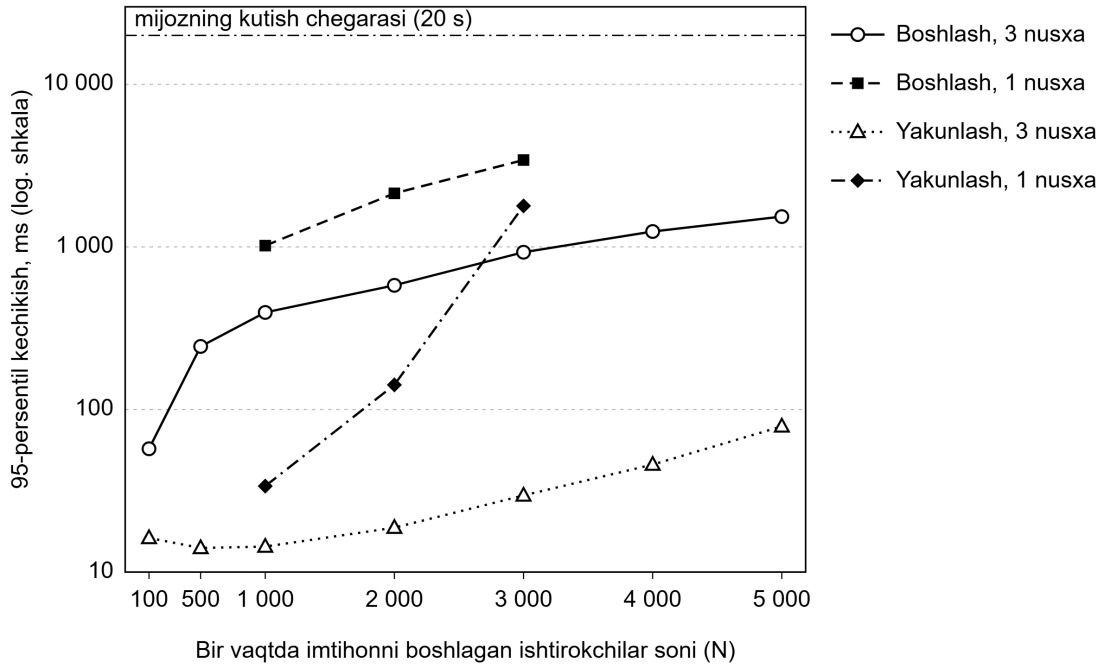
2-jadval. Yuklama sinovi natijalari: amallarning kechikishi (mediana / 95-persentil, ms)

API nusxalari	N	Takrorlash	Boshlash	Javob	Yakunlash	Xato / taymaut
3 nusxa	100	3	53 / 57	2,0 / 2,9	11 / 16	0 / 0
3 nusxa	500	3	156 / 244	1,9 / 2,6	10 / 14	0 / 0
3 nusxa	1 000	3	298 / 395	2,0 / 2,8	11 / 14	0 / 0
3 nusxa	2 000	3	481 / 580	2,2 / 3,5	14 / 19	0 / 0
3 nusxa	3 000	3	725 / 928	2,4 / 5,2	18 / 30	0 / 0
3 nusxa	4 000	3	1 056 / 1 244	2,4 / 8,4	25 / 46	0 / 0
3 nusxa	5 000	3	1 207 / 1 536	2,4 / 16	36 / 79	0 / 0

CC BY 4.0 https://doi.org/10.5281/zenodo.23947589

API nusxalari	N	Takrorlash	Boshlash	Javob	Yakunlash	Xato / taymaut
1 nusxa	1 000	3	810 / 1 019	2,6 / 13	16 / 34	0 / 0
1 nusxa	2 000	3	1 707 / 2 136	3,5 / 21	50 / 142	0 / 0
1 nusxa	3 000	3	2 909 / 3 425	6,8 / 103	1 408 / 1 788	0 / 0

Izoh: N — imtihonni bir vaqtda boshlagan ishtirokchilar soni; har bir daraja uchun barcha takrorlashlar namunalari birlashtirilgan. «Xato / taymaut» — muvaffaqiyatsiz amallar soni va haqiqiy mijozning kutish chegarasidan (boshlash va yakunlash uchun 20 s, javob uchun 10 s) oshgan javoblar soni; ulanish va heartbeat amallari ham hisobga olingan.



3-rasm. Imtihonni boshlash va yakunlash kechikishining 95-persentili ishtirokchilar soniga bog‘liq ravishda (logarifmik shkala)

Uch nusxada N = 5 000 bo‘lganda boshlash kechikishining medianasi 1,21 s, 95-persentili 1,54 s, javob tasdig‘iniki 2,4 va 16 ms, yakunlashniki 36 va 79 ms bo‘ldi. Boshlash kechikishining 95-persentili N bilan deyarli chiziqli o‘sdi (N = 100 da 57 ms, N = 1 000 da 395 ms, N = 3 000 da 0,93 s): boshlash o‘tkazuvchanligi N ≥ 2 000 da soniyasiga taxminan 3 100–3 300 ta darajasida to‘yindi va keyingi o‘sish navbatda kutish hisobiga bo‘ldi. Javoblar faqat Redis’ga yozilgani uchun ularning kechikishi barcha darajalarda past qoldi.

Bitta nusxada navbatda kutish sezilarli bo‘ldi: N = 3 000 da boshlash kechikishining 95-persentili 3,42 s ga (uch nusxada 0,93 s), yakunlashniki 1,79 s ga (uch nusxada 30 ms) yetdi. Bu konfiguratsiyada birinchi N = 3 000 sinovidan boshlab bir xil ish uchun protsessor vaqti taxminan 30 foizga oshib, keyingi takrorlashlar sekinlashdi (sababi aniqlanmadi), shuning uchun bitta va uchta nusxa o‘rtasidagi farq faqat sifat jihatidan talqin qilinadi. Barcha sinovlarda invariantlar bajarildi: har bir ishtirokchida aynan bitta natija saqlandi va uning 25 ta javobi

CC BY 4.0 https://doi.org/10.5281/zenodo.23947589

yuborilganiga aynan mos yozildi, ballar kutilganiga mos keldi, yakunlash xabarnomasi esa har bir natija uchun aynan bir marta yuborildi.

Ekspluatatsiyada aniqlangan nosozliklar

Katalogga 10 ta muhim nosozlik va loyihaviy kamchilik kiritildi; ulardan yettitasi 3-jadvalda umumlashtirilgan (imtihon arafasidagi Telegram orqali kirish holati ikki qatorga ajratilgan). Qolgan uchtasi jadvalga kiritilmagan: ketma-ket identifikatorlar muammosi «Nazorat belgili identifikator» qismida, saytning avvalgi holati (ma’lumotlarni brauzer API’dan to‘g‘ridan-to‘g‘ri olgan, kesh faqat vaqt bo‘yicha yangilangan) o‘rniga joriy etilgan yechim esa arxitektura tavsifida yoritilgan; panel interfeysining qotib qolishi sahifalash, memoizatsiya, og‘ir komponentlarni kechiktirib yuklash va effekt ichidagi cheksiz navigatsiya siklini to‘xtatish orqali bartaraf etilgan.

3-jadval. Ekspluatatsiya davrida aniqlangan asosiy nosozliklar, ularning sabablari va yechimlari

Nosozlik (davr)	Asosiy sabab	Yechim	Dalil
Ishga tushishda barcha nusxalar Telegram webhook’ini o‘rnatgan, Telegram 429 qaytargan (2025-yil dekabr)	Global amal har bir nusxada bajarilgan	Faqat «asosiy» nusxa bajaradi	Ishchi tizim jurnali
Telegram orqali kirishda sahifa uzoq «osilib» qolgan (2026-yil fevral — oktabr)	Ikkinchi darajali Telegram chaqiruvi so‘rov ichida qayta urinishlar bilan kutilgan (har urinishda 60 s gacha)	Chaqiruv fonga o‘tkazildi; havolalar soniyasiga 25 xabarlik navbat orqali yuboriladi	Ishlab chiquvchi kuzatuvi; skriptda qayta hosil qilindi
Bitta bosishga ikki marta ishlov berilgan (2025-yil oktabr — 2026-yil oktabr)	Webhook taymautida xato kodi qaytgan, ishlov to‘xtamagan, Telegram qayta yuborgan	Taymautda muvaffaqiyat kodi; ishlov fonda	Kod izohi; sinov stendida qayta hosil qilindi
Vaqti-vaqti bilan 502 xatolari (2025-yil dekabr — 2026-yil oktabr)	Node.js keep-alive taymauti (5 s) nginx’nikidan (65 s) qisqa	66 va 67 s ga oshirildi	Kod izohi; qayta hosil qilinmadi (0/60)
Qisqa Redis uzilishi yoki ishlov berilmagan promise rad etilishidan keyin nusxa ishlamay qolgan (2025-yil oktabr — 2026-yil oktabr)	Qayta ulanish 3 urinishdan keyin to‘xtagan; rad etish jarayonni to‘xtatgan	Cheksiz qayta ulanish; rad etish faqat jurnalga yoziladi	Kod izohi; Redis mexanizmi qayta hosil qilindi
Har bir joylashtirishda xizmat bir necha soniya uzilgan (2026-yil yanvar — oktabr)	Barcha nusxalar bir vaqtda qayta ishga tushirilgan	Nusxalar birma-bir qayta ishga tushiriladi	Joylashtirish skripti va kod izohi
Natijalarni Excel’dan import qilish qariyb 5 oy ishlamagan (2026-yil may — sentabr)	Panel parametrni yubormay qo‘ygan, server qismi uni majburiy deb hisoblagan	Parametr ixtiyoriy qilindi	Kod farqlari; qayta hosil qilindi (HTTP 400)
Ustunlarni o‘chiruvchi migratsiya avtomatik joylashtiriladigan git-tarmoqqa kiritilgan (2026-yil mart)	git revert qo‘llangan sxema o‘zgarishini qaytarmaydi	Ustunlarni qayta yaratuvchi migratsiya	Git tarixi

CC BY 4.0 https://doi.org/10.5281/zenodo.23947589

Izoh: davrlar kommit sanalari bo‘yicha; 3–6-qatorlardagi alomatlar 2026-yil 9-oktabrdagi tuzatish kommitlari izohlaridan olingan va ularning chastotasi hujjatlashtirilmagan. Joriy holat «Materiallar va usullar» bo‘limida keltirilgan kommitlar bo‘yicha tekshirilgan.

Dalil darajasi turlicha (3-jadvaldagi «Dalil» ustuni). To‘rtta nosozlik mexanizmi loyihadagi kutubxona versiyalari bilan skriptlarda qayta hosil qilindi: doimiy 429 qaytaruvchi soxta Bot API serverida eski kirish so‘rovi kutgan chaqiruv retry_after 2 va 5 soniya bo‘lganda 8,05 va 20,05 soniya davom etdi (to‘rtta so‘rov va to‘rt marta kutish), tuzatilgani esa birinchi javobdayoq yakunlandi; yangilanishni ko‘pi bilan uch marta yetkazuvchi stendda webhook taymautining standart sozlamasi bitta yangilanishga uch marta, tuzatishdan keyin bir marta ishlov berilishiga olib keldi; eski so‘rov sxemasi panelning import so‘rovini HTTP 400 bilan rad etdi; eski siyosatda Redis mijozi uch urinishdan keyin qayta ulanishni to‘xtatdi. 502 xatolari mexanizmi esa qayta hosil qilinmadi (60 urinishda xato yo‘q).

Nosozliklar to‘rt guruhga bo‘linadi: (1) uchinchi tomon cheklovlari va yetkazish qoidalarining foydalanuvchi so‘rovi yo‘liga ta’siri; (2) komponentlar o‘rtasidagi faqat bir tomonda qayd etilgan yoki yozilmagan kelishuvlarning buzilishi (keep-alive taymautlari, panel va server qismi o‘rtasidagi shartnoma); (3) ko‘p nusxali va avtomatik joylashtiriladigan muhitda bir martalik yoki qaytarib bo‘lmaydigan amallarning nazoratsiz bajarilishi; (4) vaqtinchalik nosozlikka jarayonni to‘xtatish bilan javob beruvchi siyosat.

Muhokama

Arxitektura va yuklama ostidagi xatti-harakat

Natijalar olimpiadaning to‘liq siklini mikroservislarsiz, modulli monolit va bir nechta ilova nusxasi bilan qamrab olish mumkinligini ko‘rsatadi. Mikroservislar mustaqil joylashtirish imkonini bersa-da, taqsimlangan tizimning murakkabligi va xarajatini olib keladi (Dragoni et al., 2017), modulli monolit esa aniq ajratilgan modullarni bitta ilovada saqlovchi muqobil sifatida tobora ko‘proq muhokama qilinmoqda (Su & Li, 2024). TASIMO’da nusxalar birgalikda foydalanadigan holat Redis’da jamlangan. Buning narxi ham bor: Redis yagona nosozlik nuqtasiga aylanadi, javoblar yakunlashgacha faqat xotirada turadi, yakunlanmay qolgan sessiyalarni tiklovchi (reconciliation) jarayon yo‘q, Telegram yangilanishlari esa bitta asosiy nusxaga bog‘liq.

Sinovdagi yuklama modelida (barcha boshlashlar bir vaqtda, yakunlashlar taxminan 20 soniyaga tarqalgan) eng og‘ir bosqich boshlash bo‘ldi: har bir boshlash bazaga yozuv va Redis amallarini talab qiladi, shuning uchun uning o‘tkazuvchanligi cheklangan va kechikish N ortgan sari navbatda kutish hisobiga o‘sadi, javoblar esa xotiraga yozilgani uchun tez qoladi.

CC BY 4.0 https://doi.org/10.5281/zenodo.23947589

Uch nusxa bitta nusxaga nisbatan navbatda kutishni sezilarli qisqartirdi, uch nusxadagi eng katta boshlash kechikishi (N = 5 000 da 1,59 s) esa mijozning 20 soniyalik kutish chegarasidan o‘n martadan ortiq past qoldi. Repozitoriydagi avvalgi yuklama sinovi faqat WebSocket ulanishlari va heartbeat so‘rovlarini o‘lchagan edi; Italiya olimpiadasida boshlanishdagi cho‘qqi yuklama noto‘g‘ri sozlangan proksi sababli ulanishlarni uzgani (Audrito et al., 2021) hisobga olinsa, to‘liq foydalanuvchi yo‘lini realistik yuklama bilan sinash zarur (Jiang & Hassan, 2015).

Nosozliklardan saboqlar

Nosozliklar tahlili empirik tadqiqotlarga mos keladi. 198 ta nosozlik tahlilida halokatli nosozliklarning aksariyatini xatolarga ishlov beruvchi kodni oddiy sinash orqali oldindan bartaraf etish mumkin bo‘lgani ko‘rsatilgan (Yuan et al., 2014): TASIMO’dagi Redis qayta ulanishi va ishlov berilmagan promise rad etilishi holatlari shu turga kiradi va testlar yo‘q sharoitda yuzaga kelgan. 32 ta internet xizmatidagi 597 ta uzilish tahlilida dasturiy yangilanishlar eng keng tarqalgan sabablardan biri bo‘lgan (Gunawi et al., 2016), yangilanish nosozliklari esa alohida sinf sifatida o‘rganilgan (Zhang et al., 2021) — bir vaqtda qayta ishga tushirish va qaytarib bo‘lmaydigan migratsiya holatlari shu sinfga tegishli. Yirik bulut xizmatidagi yuzlab nosozliklar tahlili ham avtomatlashtirish va turli darajadagi testlarni tavsiya qiladi (Ghosh et al., 2022).

Bundan oltita saboq kelib chiqadi. Birinchidan, uchinchi tomon API’lariga bog‘liq ikkinchi darajali amallar foydalanuvchi so‘rovi yo‘lidan chiqarilishi, har bir tashqi chaqiruvga aniq taymaut va qayta urinish siyosati berilishi, 429 orqali bildirilgan cheklovlarga (Nottingham & Fielding, 2012) esa butun navbatni to‘xtatib turish bilan rioya qilinishi kerak. Ikkinchidan, kamida bir marta yetkaziladigan webhook’lar va bir necha yo‘l bilan ishga tushiriladigan yakunlash idempotent bo‘lishi shart (Helland, 2012): TASIMO’da yakunlash idempotent, webhook ishlov beruvchisida esa takroriy yangilanishlarni filtrlash hali yo‘q, ilova darajasidagi himoyani bazadagi noyoblik cheklovi bilan mustahkamlash ham maqsadga muvofiq. Uchinchidan, komponentlararo invariantlar ikkala tomonda hujjatlashtirilishi va avtomatik tekshirilishi kerak: nginx qiymatiga teng (65 soniya) qo‘yilgan keep-alive sozlamasi boshqa xatoni tuzatuvchi kommitda izohsiz olib tashlangan va Node.js’ning 5 soniyalik standart qiymati invariantni to‘qqiz oydan ortiq buzib turgan. To‘rtinchidan, global bir martalik amallar tayinlangan nusxada yoki qulf ostida bajarilishi, joylashtirish esa nusxalarni birma-bir qayta ishga tushirish va sxemani «kengaytirish — keyin qisqartirish» tamoyili bo‘yicha o‘zgartirish orqali xavfsiz qilinishi kerak. Beshinchidan, mijoz va server o‘rtasidagi API shartnomasi

CC BY 4.0 https://doi.org/10.5281/zenodo.23947589

yagona manbadan (masalan, OpenAPI tavsifidan) generatsiya qilingan turlar va shartnoma testlari bilan himoyalanishi lozim: Excel’dan import qariyb besh oy ishlamagan, yetishmayotgan parametrni nomlamaydigan umumiy xato xabari esa sababni yashirgan. Oltinchidan, ilova tayanadigan xizmatning qisqa uzilishiga jarayonni to‘xtatish bilan emas, cheksiz qayta ulanish va so‘rovlarni tez rad etish bilan javob berish kerak; bu siyosat hujjatlashtirilmagani uchun TASIMO’da ishlov berilmagan promise rad etilishiga munosabat uch marta o‘zgargan.

Identifikator tekshiruvi hujjatlashtirish bo‘yicha ham saboq berdi: 32 belgili alifboda 31 moduli tanlanganda 0 va 31 bir xil qoldiq beradi, shuning uchun 0 va Z o‘rtasidagi almashtirish ham, ularning o‘rin almashinuvi ham aniqlanmaydi. Bu istisno rejada oldindan qayd etilgan, biroq kod izohida xossa istisnosiz yozilgan. ISBN-10 dagi 11 moduli va qo‘shimcha «X» belgisi kabi (Gallian, 1991), alifbodan katta tub modul (Crockford sxemasidagi 37 va beshta qo‘shimcha belgi; Crockford, 2019) bu bo‘shliqni yopadi; bunday xossalar esa izohda istisnolari bilan yozilishi va barcha bitta belgili xatolarni sanab chiquvchi avtomatik test bilan tasdiqlanishi kerak.

Akademik halollik uchun platforma masofaviy kuzatuvsiz yondashuvni tanlagan: server tomonida baholash, javob kalitini mijozga yubormaslik tamoyili, savol va variantlarni aralashtirish, bitta urinish va bitta faol qurilma qoidalari ko‘zda tutilgan. Italiya olimpiadasida ham mingdan ortiq o‘quvchini uydan real vaqtda kuzatish mavjud resurslar bilan amalga oshmasligi qayd etilib, mintaqaviy bosqich musobaqadan keyingi tekshiruvlar bilan cheklangan, yuzga yaqin ishtirokchili milliy bosqichda esa videoaloqa orqali kuzatuv qo‘llangan (Audrito et al., 2021). Kuzatuvsiz yondashuv proktoringning axloqiy muammolarini chetlab o‘tadi (Coghlan et al., 2021), biroq ko‘chirmachilikning barcha shakllarini, masalan, tashqi yordamni istisno qilmaydi (Noorbehbahani et al., 2022); aniqlangan kamchiliklar esa halollik xossalari ham avtomatik testlar bilan himoyalanishi kerakligini ko‘rsatadi.

Cheklovlar va amaliy ahamiyati

Tadqiqotning cheklovlari: (1) bu yagona keys, xulosalarni umumlashtirishda ehtiyotkorlik zarur; (2) muallif loyihaga bevosita aloqador, shuning uchun xulosalar qayta tekshirish mumkin bo‘lgan artefaktlarga tayandi; (3) yuklama sinovi bitta stansiyada, nginx’siz va Docker’dagi baza bilan o‘tkazildi, generator va server bir xil resurslardan foydalandi, fon jarayonlari protsessor quvvatining bir necha foizini band qildi va Telegram orqali kirish sinalmadi; (4) yuklama modeli soddalashtirilgan: haqiqiy kabinetda ulanish va avtorizatsiya «Boshlash» bosilganda bajariladi, sinovda esa ulanishlar oldindan o‘rnatildi, imtihon vaqti

CC BY 4.0 https://doi.org/10.5281/zenodo.23947589

siqildi (25 savol taxminan 25 soniyada), vaqt tugaganda ommaviy avtomatik yakunlash va qayta ulanish to‘lqinlari sinalmadi — shuning uchun raqamlar ishchi serverlar sig‘imini emas, arxitekturaning nisbiy xatti-harakatini ko‘rsatadi; (5) ishchi muhit monitoringi, joylashtirish jurnallari, amaldagi nginx sozlamasi va foydalanish statistikasi repozitoriylarda yo‘q, nosozlik alomatlari esa asosan tuzatish bilan yozilgan kod izohlariga tayanadi (ayrim tuzatishlar sun’iy intellekt asosidagi dasturlash yordamchisi ishtirokida kiritilgan); (6) manba kodi ochiq emas; (7) xavfsizlikka oid qo‘shimcha kamchiliklar tuzatilgunga qadar yoritilmaydi.

Amaliy jihatdan, O‘zbekistonda olimpiada yoki onlayn test o‘tkazuvchi tashkilotlarga quyidagilarni minimal talab sifatida tavsiya qilish mumkin: to‘liq foydalanuvchi yo‘li bo‘yicha yuklama sinovi; idempotent yakunlash va serverda hisoblanadigan tugash vaqti; tashqi xizmatlar uchun navbat, aniq taymautlar va takroriy yetkazishni filtrlash; ikki yozuvli qidiruv va nazorat belgili identifikatorlar; API shartnomasi va komponentlararo invariantlar uchun avtomatik testlar; nusxalarni birma-bir joylashtirish va qaytarib bo‘lmaydigan migratsiyalarni qo‘lda tasdiqlash.

Xulosa

Tadqiqot Toshkent xalqaro matematika olimpiadasi platformasi misolida fan olimpiadasining to‘liq siklini qamrab oluvchi onlayn tizimni qurish va ishlatish tajribasini tahlil qildi. Birinchidan, NestJS asosidagi modulli monolit, umumiy holat uchun Redis va uchta ilova nusxasi barcha jarayonlarni qamrab olish uchun yetarli bo‘ldi, Telegram orqali parolsiz kirish, ikki yozuvli qidiruv va nazorat belgili identifikator kabi mahalliy moslashuvlar esa platformaning muhim qismiga aylandi. Ikkinchidan, mahalliy test muhitida 5 000 nafar ishtirokchi imtihonni bir vaqtda boshlaganda xato kuzatilmadi, uch nusxada boshlash kechikishining 95-persentili 1,54 s ni tashkil etdi, uchta nusxa esa bitta nusxaga nisbatan navbatda kutishni sezilarli qisqartirdi. Uchinchidan, nosozliklarning asosiy manbai biznes mantig‘i emas, balki tizim chegaralari bo‘ldi: uchinchi tomon cheklovlari, komponentlararo taymautlar, ko‘p nusxali joylashtirish va mijoz–server shartnomalari. Kelgusida ishchi muhitda telemetriya yig‘ish, taqsimlangan generatorlar va proksi server bilan to‘liq foydalanuvchi yo‘lini (Telegram orqali kirishni ham) sinash, shartnoma va halollik uchun avtomatik testlarni joriy etish hamda internet sifati past hududlardagi ishtirokchilar tajribasini baholash rejalashtirilgan.

CC BY 4.0 https://doi.org/10.5281/zenodo.23947589

Tasdiqlar

Manfaatlar to‘qnashuvi. Muallif tahlil qilingan loyihaga bevosita aloqador; manfaatlar to‘qnashuvining to‘liq tavsifi taqriz anonimligini saqlash maqsadida platformadagi tegishli maydonda tahririyatga taqdim etilgan.

Etika. Tadqiqotda odamlar yoki hayvonlar ishtirok etmagan, shu sababli etika qo‘mitasining ruxsati talab etilmagan. Yuklama sinovi sun’iy test foydalanuvchilari bilan, mahalliy test muhitida o‘tkazilgan; ishchi tizimning ma’lumotlar bazasi va shaxsiy ma’lumotlardan foydalanilmagan.

Sun’iy intellektdan foydalanish. Generativ sun’iy intellektdan (Anthropic Claude, Opus 5.5) foydalanish tartibi «Materiallar va usullar» bo‘limida bayon etilgan.

Ma’lumotlar mavjudligi. Kod ko‘rsatkichlarini hisoblash, nosozliklarni qayta hosil qilish, identifikator nazorat belgisini tekshirish va yuklama sinovi skriptlari hamda xom natijalar so‘rov asosida taqdim etiladi. Platformaning manba kodi ochiq emas.

Adabiyotlar

Agarwal, R., & Gaule, P. (2020). Invisible geniuses: Could the knowledge frontier advance faster? *American Economic Review: Insights, 2*(4), 409–424. https://doi.org/10.1257/aeri.20190457

Akhmedjanova, D., & Kerimova, I. (2024). University students and teachers’ experiences with distance education in Uzbekistan. *Journal of Eastern European and Central Asian Research, 11*(1), 156–175. https://doi.org/10.15549/jeecar.v11i1.1282

Audrito, G., Di Luigi, W., Laura, L., Morassutto, E., & Ostuni, D. (2021). The Italian job: Moving (massively) online a national olympiad. *Olympiads in Informatics, 15*, 3–12. https://doi.org/10.15388/ioi.2021.01

Butler-Henderson, K., & Crawford, J. (2020). A systematic review of online examinations: A pedagogical innovation for scalable authentication and integrity. *Computers & Education, 159*, Article 104024. https://doi.org/10.1016/j.compedu.2020.104024

Campbell, J. R., & Walberg, H. J. (2011). Olympiad studies: Competitions provide alternatives to developing talents that serve national interests. *Roeper Review, 33*(1), 8–17. https://doi.org/10.1080/02783193.2011.530202

Coghlan, S., Miller, T., & Paterson, J. (2021). Good proctor or “Big Brother”? Ethics of online exam supervision technologies. *Philosophy & Technology, 34*(4), 1581–1606. https://doi.org/10.1007/s13347-021-00476-1

Crockford, D. (2019, March 4). *Base 32*. https://www.crockford.com/base32.html

CC BY 4.0 https://doi.org/10.5281/zenodo.23947589

Dragoni, N., Giallorenzo, S., Lluch Lafuente, A., Mazzara, M., Montesi, F., Mustafin, R., & Safina, L. (2017). Microservices: Yesterday, today, and tomorrow. In M. Mazzara & B. Meyer (Eds.), *Present and ulterior software engineering* (pp. 195–216). Springer. https://doi.org/10.1007/978-3-319-67425-4_12

Fielding, R., Nottingham, M., & Reschke, J. (Eds.). (2022). *HTTP caching* (RFC 9111). RFC Editor. https://doi.org/10.17487/RFC9111

Gallian, J. A. (1991). The mathematics of identification numbers. *The College Mathematics Journal, 22*(3), 194–202. https://doi.org/10.1080/07468342.1991.11973381

Ghosh, S., Shetty, M., Bansal, C., & Nath, S. (2022). How to fight production incidents? An empirical study on a large-scale cloud service. In *Proceedings of the 13th Symposium on Cloud Computing* (pp. 126–141). Association for Computing Machinery. https://doi.org/10.1145/3542929.3563482

Gunawi, H. S., Hao, M., Suminto, R. O., Laksono, A., Satria, A. D., Adityatama, J., & Eliazar, K. J. (2016). Why does the cloud stop computing? Lessons from hundreds of service outages. In *Proceedings of the Seventh ACM Symposium on Cloud Computing* (pp. 1–16). Association for Computing Machinery. https://doi.org/10.1145/2987550.2987583

Helland, P. (2012). Idempotence is not a medical condition. *Communications of the ACM, 55*(5), 56–65. https://doi.org/10.1145/2160718.2160734

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS Quarterly, 28*(1), 75–105. https://doi.org/10.2307/25148625

Jiang, Z. M., & Hassan, A. E. (2015). A survey on load testing of large-scale software systems. *IEEE Transactions on Software Engineering, 41*(11), 1091–1118. https://doi.org/10.1109/TSE.2015.2445340

Navarro, G. (2001). A guided tour to approximate string matching. *ACM Computing Surveys, 33*(1), 31–88. https://doi.org/10.1145/375360.375365

Noorbehbahani, F., Mohammadi, A., & Aminazadeh, M. (2022). A systematic review of research on cheating in online exams from 2010 to 2021. *Education and Information Technologies, 27*(6), 8413–8460. https://doi.org/10.1007/s10639-022-10927-7

Nottingham, M., & Fielding, R. (2012). *Additional HTTP status codes* (RFC 6585). RFC Editor. https://doi.org/10.17487/RFC6585

O‘zbekiston Respublikasi Prezidentining «“Raqamli O‘zbekiston — 2030” strategiyasini tasdiqlash va uni samarali amalga oshirish chora-tadbirlari to‘g‘risida»gi Farmoni, PF-6079-son. (2020, 5-oktabr). Qonun hujjatlari ma’lumotlari milliy bazasi, 06/20/6079/1349-son. https://lex.uz/docs/5030957

CC BY 4.0 https://doi.org/10.5281/zenodo.23947589

Runeson, P., & Höst, M. (2009). Guidelines for conducting and reporting case study research in software engineering. *Empirical Software Engineering, 14*(2), 131–164. https://doi.org/10.1007/s10664-008-9102-8

Salaev, U., Kuriyozov, E., & Gómez-Rodríguez, C. (2022). A machine transliteration tool between Uzbek alphabets. In J. Vičič & B. Kavšek (Eds.), *Proceedings of the International Conference and Workshop on Agglutinative Language Technologies as a Challenge of Natural Language Processing (ALTNLP 2022)* (CEUR Workshop Proceedings, Vol. 3315, pp. 42–50). CEUR-WS.org. https://ceur-ws.org/Vol-3315/paper05.pdf

Su, R., & Li, X. (2024). Modular monolith: Is this the trend in software architecture? In *Proceedings of the 1st International Workshop on New Trends in Software Architecture* (pp. 10–13). Association for Computing Machinery. https://doi.org/10.1145/3643657.3643911

Telegram. (n.d.). *Bots FAQ*. Retrieved October 10, 2026, from https://core.telegram.org/bots/faq

Yin, R. K. (2018). *Case study research and applications: Design and methods* (6th ed.). SAGE Publications.

Yuan, D., Luo, Y., Zhuang, X., Rodrigues, G. R., Zhao, X., Zhang, Y., Jain, P. U., & Stumm, M. (2014). Simple testing can prevent most critical failures: An analysis of production failures in distributed data-intensive systems. In *Proceedings of the 11th USENIX Symposium on Operating Systems Design and Implementation* (pp. 249–265). USENIX Association. https://www.usenix.org/conference/osdi14/technical-sessions/presentation/yuan

Zhang, Y., Yang, J., Jin, Z., Sethi, U., Rodrigues, K., Lu, S., & Yuan, D. (2021). Understanding and detecting software upgrade failures in distributed systems. In *Proceedings of the ACM SIGOPS 28th Symposium on Operating Systems Principles* (pp. 116–131). Association for Computing Machinery. https://doi.org/10.1145/3477132.3483577

CC BY 4.0 https://doi.org/10.5281/zenodo.23947589