

DETALJNO TEHNOLOŠKO IZVJEŠTAJE O IASOQAR APLIKACIJI

Analiza datum: 19. novembar 2025

Trenutna grana: claude/analyze-application-01Xvp6SajZg8ky79GfFX8L5v

Status: Clean (bez nezahitjevanih izmjena)

1. OPĆA STRUKTURA PROJEKTA

Direktorijumska organizacija:

```

/home/user/iasoqar-app/
├── isoqar_app/                # Django projekt konfiguracija
│   ├── settings.py           # Glavna Django konfiguracija
│   ├── urls.py               # Root URL routing
│   ├── asgi.py               # ASGI konfiguracija
│   ├── wsgi.py               # WSGI konfiguracija
│   ├── middleware.py         # Custom middleware za autentifikaciju
│   └── data/                 # Podatkovne datoteke
├── company/                  # Glavni business app (~1.1MB)
│   ├── models.py             # Centralni import svih modela
│   ├── views.py              # Glavne views (1,888 linija)
│   ├── views_ajax.py         # AJAX endpoints (849 linija)
│   ├── views_cycles.py       # Certification cycle views (370 linija)
│   ├── urls.py               # URL routing (189 linija)
│   ├── forms.py              # Django forms (787 linija)
│   ├── admin.py              # Admin panel konfiguracija (199 linija)
│   ├── migrations/           # 62+ migracije baze podataka
│   ├── tests/                # Unit i integracijsko testiranje
│   ├── admin/                # Specijalne admin klase
│   ├── management/commands/  # Custom Django komande
│   ├── fixtures/             # Test data
│   └── data/                 # Statički podaci (gradovi, industrije, zemlje)
├── accounts/                 # Autentifikacija i upravljanje korisnicima
│   ├── views.py
│   ├── urls.py
│   ├── models.py
│   └── migrations/
├── templates/                # HTML template datoteke (59 fajlova)
│   ├── company/
│   ├── auditor/
│   ├── calendar/
│   ├── accounts/
│   ├── certification_cycles/
│   ├── iaf/
│   ├── srbija_tim/
│   ├── layouts/              # Base template-i
│   └── dashboard.html
├── static/                   # Statički resursi (5.6MB)
│   └── js/                   # JavaScript fajlovi

```

```

|   |   |─ calendar.js
|   |   |─ jquery.min.js
|   |   |─ bootstrap.bundle.min.js
|   |   |─ company/          # Kompanija-specifični JS
|   |   |─ auditor/          # Auditor-specifični JS
|   |   |─ accounts/         # Accounts-specifični JS
|   |   |   └─ auth/         # Autentifikacija JS
|   |─ css/                  # CSS datoteke
|   |   |─ adminlte.min.css (1.3MB)
|   |   |─ custom.css
|   |   |─ calendar.css
|   |   |─ company/
|   |   |   └─ accounts/
|   |─ plugins/              # Treće strane biblioteke
|   |   └─ webfonts/
|─ nginx/                    # Nginx konfiguracija
|   |─ Dockerfile
|   |─ default.conf
|   |   └─ SSL_SETUP.md
|─ .github/workflows/        # GitHub Actions CI/CD
|   |─ ci-cd.yml             # Glavni pipeline
|   |─ django.yml            # Django testiranje
|   |─ manual-deploy.yml     # Ručni deployment
|   |   └─ summary.yml
|─ docker-compose.yml        # Production setup
|─ docker-compose.dev.yml    # Development setup
|─ Dockerfile                # Django aplikacija
|─ entrypoint.sh             # Docker entrypoint
|─ manage.py                 # Django CLI
|─ requirements.txt           # Python zavisnosti
|─ package.json              # Node.js zavisnosti
|─ pyproject.toml            # Python project config
|─ .flake8                   # Flake8 konfiguracija
|─ .gitignore                 # Git ignore pravila
|─ .env.example               # Environment template
└─ Documentation/            # Markdown dokumentacija
    |─ CI-CD-SETUP.md
    |─ DEPLOYMENT.md
    |─ IMPORT_AUDITOR_ASSIGNMENTS.md
    └─ IMPORT_IAF_CODES.md

```

2. TEHNOLOŠKI STACK

Backend Stack:

- **Jezik:** Python 3.13 - 3.14
- **Web Framework:** Django 5.2.8 (najnovija verzija)
- **Web Server:** Gunicorn 23.0.0
- **Reverse Proxy:** Nginx
- **Baza podataka:**
 - PostgreSQL 14+ (production)
 - SQLite3 (development)
- **ORM:** Django ORM
- **API Framework:** Django REST Framework 3.16.1

Frontend Stack:

- **Template Engine:** Django Templates
- **UI Framework:** AdminLTE (Tabler-based, ~1.3MB CSS)
- **Calendar Library:** FullCalendar 6.1.17 (sa Bootstrap, DayGrid, TimeGrid, Locales)
- **jQuery:** 3.x (minifizirano)
- **Bootstrap:** Bootstrap Bundle 5.x
- **JavaScript:** Vanilla JS + jQuery

Deployment:

- **Docker:** Multi-container setup
 - Django app (Python 3.14-slim)
 - PostgreSQL 14
 - Nginx reverse proxy
- **Orkestracija:** Docker Compose v2.23.0
- **CI/CD:** GitHub Actions

Baze podataka:

PostgreSQL 14:

- Proizvodni setup sa health check
- psycopg3 (binarni) kao driver
- Podrška za Postgis extension

SQLite3:

- Development okruženje
- Datoteka: db.sqlite3

3. ZAVISNOSTI I SKRIPTE

Requirements.txt (37 Python paketa)

Core Django:

- asgiref==3.10.0
- Django==5.2.8
- djangorestframework==3.16.1
- djangochannelsrestframework==1.3.0

Admin & UI:

- django-jazzmin==3.0.1 (Modernizovani admin panel)
- django-admin-rangefilter==0.13.3
- django-mass-edit==3.5.0
- django-nested-admin==4.1.6

Data Processing:

- pandas>=2.0.0 (Excel/CSV procesiranje)
- openpyxl>=3.1.0 (XLSX podrška)
- python-dateutil==2.9.0.post0
- pillow==12.0.0 (Image processing)
- python-slugify==8.0.4
- python-monkey-business==1.1.0
- text-unidecode==1.3
- sqlparse==0.5.3

PDF Export:

- weasyprint==66.0 (PDF generator, zamjena za xhtml2pdf)
- reportlab==4.4.4

Datetime & Timezone:

- pytz==2025.2
- tzdata==2025.2

Production:

- gunicorn==23.0.0
- psycpg[binary]>=3.2,<3.3

Package.json (Frontend):

```
{
  "@fullcalendar/bootstrap": "^6.1.17",
  "@fullcalendar/core": "^6.1.17",
  "@fullcalendar/daygrid": "^6.1.17",
  "@fullcalendar/interaction": "^6.1.17",
  "@fullcalendar/timegrid": "^6.1.17",
  "@fullcalendar/locales": "^6.1.17"
}
```

Pyproject.toml konfiguracija:

Black:

- line-length: 127
- target-version: ['py313']

isort:

- profile: black
- Django i custom app imports organizovani

Pylint:

- max-line-length: 127
- Ignorirani specifični kodovi (C0111, C0103, R0903, itd.)

Bandit (Security):

- Isključene određene provjere
- Ignorirani test direktorijumi

4. BACKEND STRUKTURA

Model struktura (7 glavnih datoteka sa modelima)

company_models.py (211 linija):

- Company (Kompanija)
- name, pib, mb, industry, number_of_employees
- certificate_status, is_active
- street, city, postal_code, country
- phone, email, website, notes
- KontaktOsoba (Contact person)
- OstalaLokacija (Other locations)

iaf_models.py:

- IAFScopeReference (IAF scope referencije)
- IAFEACCode (IAF/EAC certifikacijski kodovi)
- CompanyIAFEACCode (Povezanost kompanije i EAC kodova)

standard_models.py:

- StandardDefinition (Definicije standarda)

- Standard/CompanyStandard (Standardi kompanije)
- StandardIAFScopeReference

auditor_models.py (316 linija):

- Auditor (Revizor)
- Qualifications, kategorije, standardi
- Direktni EAC kodovi
- AuditorStandard (Standardi auditora)
- AuditorStandardIAFEACCode (Auditor IAF/EAC povezanost)
- AuditorIAFEACCode (Direktni IAF kodovi)

cycle_models.py (715 linija - najveći model fajl):

- CertificationCycle (Sertifikacijski ciklus)
- planirani_datum, is_integrated_system
- broj_dana_nadzora, status
- datum_sprovođenja_inicijalne
- CycleStandard (Standardi u ciklusu)
- AuditDay (Dani revizije)
- AuditorReservation (Rezervacije auditora)
- CycleAudit (Revizije u ciklusu)
- audit_type, audit_status
- planirani_datum, actual_date
- lead_auditor

calendar_models.py:

- CalendarEvent (Kalendarski događaji)
- Appointment (Sastanci/sastanke)

srbija_tim_models.py:

- SrbijaTim (Srbija tim model za lokalne revizije)

Views struktura (3,107 linija ukupno)

views.py (1,888 linija):

- Dashboard view
- Company CRUD views:
 - CompanyListView
 - CompanyDetailView
 - CompanyCreateView
 - CompanyUpdateView
 - CompanyDeleteView
- Audit views:
 - AuditListView

- CompanyAuditsView
- CompanyAuditDetailView
- AuditCreateView/Update/Delete
- Calendar views:
- CalendarView
- CalendarEventsView
- appointment_calendar_json (API)
- Appointment CRUD:
- appointment_detail/create/update/delete
- Standard management:
- company_standard_create/detail/update/delete
- AJAX endpoints:
- get_company_contacts
- get_companies

views_ajax.py (849 linija):

- delete_standard
- add_iaf_eac_code
- delete_iaf_eac_code
- update_iaf_eac_primary
- list_iaf_eac_codes
- certification_cycle_json
- audit_days_by_audit_id
- update_event_date (drag-and-drop)
- validate_auditor_reservation

auditor_views.py (652 linija):

- AuditorListView/DetailView/CreateView/UpdateView/DeleteView
- auditor_standard_create/update/delete
- auditor_standard_iaf_eac_create/update/delete
- get_auditor_details (API)
- get_qualified_auditors (API)

views_cycles.py (370 linija):

- CertificationCycleListView/DetailView/CreateView/UpdateView/DeleteView
- CycleAuditCreateView/UpdateView/DeleteView

auditor_direct_iaf_views.py (185 linija):

- auditor_direct_iaf_eac_create/update/delete

srbija_tim_views.py (396 linija):

- SrbijaTimCalendarView
- SrbijaTimListView
- SrbijaTimAuditorScheduleView

- CRUD operacije (Create/Update/Delete/Detail)
- srpbija_tim_calendar_json
- srpbija_tim_update_date

contact_views.py:

- kontakt_osoba_create/update/delete

location_views.py:

- LocationListView/DetailView/CreateView/UpdateView/DeleteView

iaf_views.py:

- IAFEACCodeListView

health_views.py:

- health_check
- readiness_check
- liveness_check (za monitoring)

URL routing (189 linija, 75+ ruta):

```
# Calendar i Appointments
- /company/calendar/
- /company/calendar/events/
- /company/calendar/api/events/
- /company/appointments/*

# Companies
- /company/companies/ (CRUD)
- /company/companies/<id>/
- /company/companies/<id>/update/
- /company/companies/<id>/delete/

# Auditors
- /company/auditors/ (CRUD)
- /company/auditors/<id>/standards/add/
- /company/auditors/<id>/direct-iaf-eac/

# Certification Cycles
- /company/cycles/ (CRUD)
- /company/cycles/<id>/audits/create/

# Srbija Tim
- /company/srbija-tim/ (CRUD)
- /company/srbija-tim/auditor-schedule/

# API endpoints
- /company/api/company-contacts/
- /company/api/companies/
- /company/api/auditors/<id>/details/
- /company/api/qualified-auditors/
- /company/api/standards/delete/
- /company/api/validate-auditor-reservation/
```

Forms struktura (787 linija):

- LocationForm
- CompanyForm

- AuditorForm
- auditor_forms.py (2,951 linija)
- auditor_direct_iaf_form.py
- contact_forms.py
- Sve koriste Django ModelForm

Admin interfejs:

admin.py (199 linija):

- CompanyAdmin sa inline-ovima
- KontaktOsobaAdmin
- CertificationCycleAdmin
- Custom tabularni inline-ovi za relacije

admin_register.py (243 linija):

- Registracija svih modela u Jazzmin admin panelu

admin/cycle_admin.py:

- Specijalizovana admin konfiguracija za cycle-e

admin/iaf_eac_admin.py:

- IAF/EAC kod admin panel

5. FRONTEND STRUKTURA

Template datoteke (59 HTML fajlova)

Layouts:

- base.html (Osnovna struktura)
- navbar.html (Navigacijska traka)
- sidebar.html (Bočna traka)
- footer.html

Company templates (10 datoteka):

- company-form.html (111KB, kompleksna forma)
- company-detail.html (45KB)
- company-list.html (14KB)
- company-confirm-delete.html
- standard-detail.html
- standard-update-form.html

- delete-standard-modal.html
- standard-validation-modal.html
- standard-success-modal.html
- script-tags.html

Auditor templates (10 datoteka):

- auditor-list.html
- auditor-detail.html
- auditor-form.html
- auditor_standard_form.html
- auditor_standard_iaf_eac_form.html
- auditor_direct_iaf_eac_form.html
- auditor_standard_manual_edit.html
- auditor_direct_iaf_eac_manual_edit.html
- confirm delete fajlovi

Calendar templates:

- calendar.html
- appointment_detail.html

Certification Cycles templates:

- cycle-list.html
- cycle-detail.html
- cycle-form.html
- audit-form.html
- audit_confirm_delete.html
- cycle_confirm_delete.html

Srbija Tim templates:

- form.html (za CRUD operacije)

Other templates:

- dashboard.html (52KB)
- debugging.html (3.3KB)
- Accounts templates (login, register, profile)
- IAF template-i

JavaScript datoteke (20+ fajlova):

Core:

- jquery.min.js
- bootstrap.bundle.min.js
- jquery-ui.min.js

- adminlte.min.js
- chart.min.js

Calendar:

- calendar.js (FullCalendar integracijska skripta)

Company-specifični:

- company-detail.js
- company-form-submit.js
- company-form-debug.js
- standards-validation.js
- company-standards.js
- standards-form.js
- standard-direct-add.js
- iaf-eac-add-clean.js
- iaf-eac-table.js
- auditor-list.js

Accounts:

- login.js
- register.js
- profile.js

Auth:

- session-handler.js

CSS datoteke:

Core:

- adminlte.min.css (1.3MB - glavni styling)
- all.min.css (FontAwesome, 59KB)
- jquery-ui.min.css (36KB)

Custom:

- custom.css
- calendar.css (4.8KB)
- cycle_detail_fix.css (1.4KB)

Component-specifični:

- accounts/ css
- company/ css

6. KONFIGURACIJE

Django Settings (isoqar_app/settings.py, 271 linija):

Database:

- PostgreSQL za production (konfiguracija kroz environment)
- SQLite3 za development
- psycopg3 binary driver

Installed Apps:

1. jazzmin (Modernizovani admin)
2. django.contrib.admin
3. django.contrib.auth
4. django.contrib.contenttypes
5. django.contrib.sessions
6. django.contrib.messages
7. django.contrib.staticfiles
8. nested_admin
9. company.apps.CompanyConfig
10. accounts

Middleware:

- SecurityMiddleware
- SessionMiddleware
- CommonMiddleware
- CsrfViewMiddleware
- AuthenticationMiddleware
- MessageMiddleware
- XFrameOptionsMiddleware
- RequireLoginMiddleware (custom - zahtijeva login)

Security Settings:

- SECRET_KEY iz env variable
- ALLOWED_HOSTS: ['localhost', '127.0.0.1', 'web', 'isoqar.geo-biz.com', ...]
- CSRF_TRUSTED_ORIGINS konfiguriran za Nginx proxy
- TIME_ZONE = 'Europe/Belgrade'
- USE_TZ = False

Static & Media:

- STATIC_URL = '/static/'
- STATIC_ROOT = '/app/static'
- MEDIA_URL = 'media/'

- MEDIA_ROOT = '/app/media'
- STATICFILES_STORAGE = ManifestStaticFilesStorage

Authentication:

- LOGIN_URL = '/accounts/login/'
- LOGIN_REDIRECT_URL = '/'
- LOGOUT_REDIRECT_URL = '/accounts/login/'

Jazzmin Settings:

- Site title: "ISOQAR Admin"
- Prilagođene ikonice za modele
- Horizontal tabs za forme
- Sidebar navigation

Environment Variables (.env.example):

```
DEBUG=False
DJANGO_SECRET_KEY=your_secure_secret_key_here
POSTGRES_DB=isoqar
POSTGRES_USER=postgres
POSTGRES_PASSWORD=secure_password_here
```

Docker konfiguracija:

Dockerfile (Django app):

- Base image: python:3.14-slim
- Sistemske zavisnosti:
- build-essential, gcc, g++
- libpq-dev (PostgreSQL)
- libcairo2, libpango, libgdk-pixbuf (za PDF)
- shared-mime-info
- Collect static files
- Expose port 8000
- Entrypoint: entrypoint.sh
- CMD: gunicorn

docker-compose.yml (Production):

Services:

web (Django):

- Volumes: static_volume, media_volume
- Expose: 8000 (internal)
- Depends on: PostgreSQL sa health check
- Env: DEBUG, DATABASE_URL, POSTGRES_*

nginx (Reverse proxy):

- Volumes: static_volume, media_volume
- Ports: 80:80 (i 443:443 za SSL)
- Depends on: web

db (PostgreSQL 14):

- Ports: 5432:5432 (development access)
- Health check: pg_isready
- Volume: postgres_data

Volumes: postgres_data, static_volume, media_volume

entrypoint.sh:

- Čeka PostgreSQL da bude dostupan
- Pokretanje migracija
- Prikupljanje statičkih fajlova
- Izvršavanje CMD komande

Nginx konfiguracija (nginx/default.conf):

- Upstream proxy ka Django app
 - Staticke datoteke serviranje
 - Media datoteke serviranje
 - GZIP kompresija
 - SSL setup dokumentacija dostupna
-

7. TESTING

Test datoteke (3 fajla):

test_location_views.py:

- LocationCreateViewTest klasa
- Tests:
 - location_form bez company_id pokazuje dropdown
 - location_form sa company_id skriva dropdown
 - LocationCreateView sa company parametrom
 - LocationCreateView bez company parametra
 - POST zahtijevanja za kreiranje lokacije

test_update_event_date.py:

- Testiranje drag-and-drop kalendara

init.py:

- Test package inicijalizacija

Test Framework:

- Django TestCase
- Django Client za HTTP zahtieve
- django.contrib.auth.models.User
- ModelForm testiranje

Code Quality Tools (iz CI-CD pipeline-a):

Linting:

- Black (code formatting check)
- isort (import sorting check)
- Flake8 (style guide enforcement)
- Pylint (code analysis)
- Bandit (security linting)
- Safety (dependency vulnerabilities)

Database Testing:

- postgres/postgis:16-3.4 service

Coverage:

- Django test runner
- pytest integracija

Performance:

- Max complexity: 10
- Max line length: 127

8. CI/CD PIPELINE

GitHub Actions Workflows:

.github/workflows/ci-cd.yml (Glavni pipeline):

Triggers:

- Push na: master, main, develop
- Pull request na: master, main

Environment:

- PYTHON_VERSION: "3.13"
- DOCKER_COMPOSE_VERSION: "2.23.0"

Jobs:

1. LINT Stage:

- Code Quality & Linting
- Black, isort, Flake8, Bandit checks
- Dependency vulnerability scan

2. BUILD & TEST Stage (depends on: lint):

- Django test suite
- PostgreSQL 14 sa PostGIS
- System dependencies instalacija
- Test pokretanje
- Build Docker image

3. DEPLOY Stage (manual):

- SSH deployment
- Docker Compose pull i restart
- Health checks

.github/workflows/django.yml:

- Specijalan Django test workflow

.github/workflows/manual-deploy.yml:

- Ručni deployment workflow

Security Checks:

- Bandit za sigurnosne propuste
- Safety za zavisnosti
- SSH key-based deployment autentifikacija

9. DOKUMENTACIJA

Markdown datoteke:

CI-CD-SETUP.md (6.7KB):

- GitHub Actions CI/CD pipeline pregled
- Konfiguracija GitHub Secrets
- SSH ključ generisanje
- Deployment setup instrukcije

DEPLOYMENT.md (7.7KB):

- Production deployment guide
- GitHub Secrets konfiguracija
- SSH setup procedura
- Server konfiguracija
- SSL/HTTPS setup

IMPORT_AUDITOR_ASSIGNMENTS.md (7.1KB):

- Import auditora iz Excel-a
- Komande za import
- Dry-run testiranje
- Normalizacija EAC kodova

IMPORT_IAF_CODES.md (1.9KB):

- Import IAF/EAC kodova
- Očekivani format Excel-a
- Procedura importa

Dostupna dokumentacija:

- README odsustan (trebalo bi kreirati)
- Setup instrukcije dostupne kroz .md fajlove
- Docker setup dokumentovan
- Deployment i CI/CD detaljno dokumentovani

10. NAPREDNE KARAKTERISTIKE

Custom Features:

- Certificacijski ciklus management
- Auditor management sa specijalističkim standardima
- IAF/EAC kodovi sa scope referencama
- Drag-and-drop kalendar sa событјjem
- Srbija Tim modul za lokalne revizije
- Multi-location podrška kompanije
- Contact person management
- PDF export (weasyprint)
- Excel import/export (pandas, openpyxl)

API Endpoints:

- RESTful endpoints za sve CRUD operacije
- AJAX endpoints za async operacije
- Calendar API za FullCalendar integraciju
- Auditor qualification API
- Validation endpoints

Security Features:

- Login middleware zahtijeva autentifikaciju
 - CSRF protection
 - Secure password handling
 - Custom authorization per-model
 - Health check endpoints (health, readiness, liveness)
-

11. STATISTIKA PROJEKTA

Veličina koda:

- **Company app:** ~8,427 linija Python koda
- **Najveći Python fajl:** views.py (1,888 linija)
- **Template datoteke:** 59 HTML fajlova
- **JavaScript datoteke:** 20+ fajlova
- **CSS datoteke:** 8+ fajlova
- **Migracije:** 62+ migracije baze

Disk prostor:

- **staticfiles/:** 18MB
- **static/:** 5.6MB
- **node_modules/:** 4.0MB
- **company app/:** 1.1MB
- **templates/:** 742KB
- **isoqar_app/:** 37KB

Database:

- **SQLite3 (dev):** 684KB (db.sqlite3)
- **20 Django modela** sa relacijama

12. ZAKLJUČAK

Aplikacija **ISOQAR** je moderan, dobro strukturirani Django projekat sa:

- ✓ **Skalabilnom arhitekturom** - Modularan design sa jasnom separacijom
- ✓ **Production-ready deployment** - Docker, Nginx, PostgreSQL
- ✓ **CI/CD pipeline** - GitHub Actions sa linting, build, test, deploy
- ✓ **Kompleksnom business logikom** - Certifikacijski ciklusi, auditor management
- ✓ **User-friendly UI** - AdminLTE theme sa custom JavaScript
- ✓ **API-first pristup** - RESTful endpoints za integraciju

- ✓ **Security-focused** - Middleware za autentifikaciju, CSRF proteksija
- ✓ **Dokumentovana** - Deployment, CI/CD, import proceduru jasno dokumentovane

Preporuke:

1. Kreirati README.md sa quick-start instrukcijama
2. Dodati integration testove za critical workflows
3. Dokumentovati API endpoints (možda OpenAPI/Swagger)
4. Implementirati logging i monitoring za production
5. Backup strategija za PostgreSQL bazu
6. Rate limiting za API endpoints