

# Production Deployment Plan

---

## primariata.work

---

**Target Date:** TBD **Environment:** Production (Vercel Frankfurt + Supabase + Cloudflare)

---

## Phase 1: Pre-Deployment Preparation

---

### 1.1 Code Quality & Testing

- [ ] Run full test suite: `pnpm test && pnpm test:e2e`
- [ ] TypeScript validation: `pnpm type-check`
- [ ] Linting: `pnpm lint`
- [ ] Build verification: `pnpm build`
- [ ] Performance audit: Lighthouse CI (target: 90+ scores)
- [ ] Accessibility audit: `pnpm test:ally`

### 1.2 Environment Configuration

- [ ] Create `.env.production` with production values
- [ ] Verify all required env vars are set in Vercel dashboard
- [ ] Remove all debug/development flags
- [ ] Enable production error tracking (Sentry)
- [ ] Configure production analytics (Vercel Analytics)

### 1.3 Security Hardening

- [ ] Review and update all RLS policies in Supabase
- [ ] Rotate all API keys and secrets
- [ ] Enable rate limiting on API routes
- [ ] Configure CORS policies
- [ ] Review authentication flows for vulnerabilities
- [ ] Enable Cloudflare WAF rules
- [ ] Configure DDoS protection thresholds

## Phase 2: Database Setup (1-2 ore)

---

### 2.1 Supabase Production Instance

- [ ] Create new Supabase project for production
- [ ] Configure production database settings (connection pooling, timeouts)
- [ ] Run all migrations: `supabase db push`
- [ ] Verify schema integrity
- [ ] Enable Point-in-Time Recovery (PITR)
- [ ] Configure automated backups (daily @ 3 AM EET)

### 2.2 Data Import

- [ ] Import 13,851 localități: `pnpm run import:localitati`
- [ ] Verify data integrity with test queries
- [ ] Create database indexes for performance: `sql CREATE INDEX idx_localitati_judet ON localitati(judet_id); CREATE INDEX idx_cereri_user ON cereri(user_id); CREATE INDEX idx_cereri_status ON cereri(status);`
- [ ] Run VACUUM ANALYZE for statistics

### 2.3 Storage Setup

- [ ] Configure Supabase Storage buckets:
  - `documents` (private, authenticated users only)
  - `avatars` (public read, authenticated write)
  - [ ] Set up storage policies (RLS)
  - [ ] Configure file size limits (10MB per document)
- 

## Phase 3: Vercel Deployment (30 min)

---

### 3.1 Initial Setup

- [ ] Connect GitHub repo to Vercel
- [ ] Configure build settings:
- Framework: Next.js

- Build Command: `pnpm build`
- Output Directory: `.next`
- Install Command: `pnpm install`
- Node Version: 20.x
- ☐ Select Frankfurt region (EDGE network)

## 3.2 Environment Variables

```
# Supabase
NEXT_PUBLIC_SUPABASE_URL=https://xxx.supabase.co
NEXT_PUBLIC_SUPABASE_ANON_KEY=eyJxxx...
SUPABASE_SERVICE_ROLE_KEY=eyJxxx...

# Auth
NEXTAUTH_URL=https://primariata.work
NEXTAUTH_SECRET=[generated-secret]

# Google OAuth
GOOGLE_CLIENT_ID=xxx.apps.googleusercontent.com
GOOGLE_CLIENT_SECRET=xxx

# Analytics & Monitoring
NEXT_PUBLIC_VERCEL_ANALYTICS_ID=xxx
SENTRY_DSN=https://xxx@sentry.io/xxx

# Feature Flags
NEXT_PUBLIC_ENABLE_SURVEYS=true
NEXT_PUBLIC_MAINTENANCE_MODE=false
```

## 3.3 Domain Configuration

- ☐ Add custom domain: `primariata.work`
- ☐ Configure www redirect: `www.primariata.work` → `primariata.work`
- ☐ Enable automatic HTTPS (Let's Encrypt)
- ☐ Verify SSL certificate

## Phase 4: Cloudflare Setup

---

### 4.1 DNS Configuration

Type	Name	Content	TTL	Proxy
A	@	76.76.21.21 (Vercel IP)	Auto	Proxied
CNAME	www	primariata.work	Auto	Proxied

### 4.2 Security Rules

- ☐ Enable WAF (Web Application Firewall)
- ☐ Configure DDoS protection (Medium sensitivity)
- ☐ Set up rate limiting:
  - API routes: 100 req/min per IP
  - Auth endpoints: 10 req/min per IP
- ☐ Enable Bot Fight Mode
- ☐ Configure challenge page for suspicious traffic

### 4.3 Performance Optimization

- ☐ Enable Cloudflare CDN (caching rules)
- ☐ Configure Brotli compression
- ☐ Enable HTTP/3 (QUIC)
- ☐ Set cache TTLs:
  - Static assets: 1 year
  - API responses: 0 (no cache)
  - Pages: 1 hour (with revalidation)

---

## Phase 5: Monitoring & Logging (30 min)

---

### 5.1 Error Tracking (Sentry)

- ☐ Create Sentry project for production
- ☐ Configure source maps upload
- ☐ Set up error alerts (Slack/Email)

- [ ] Configure performance monitoring
- [ ] Set up issue grouping rules

## 5.2 Analytics

- [ ] Enable Vercel Analytics
- [ ] Configure custom events:
  - User registration
  - Request submission
  - Payment completion
  - Document download
- [ ] Set up conversion funnels
- [ ] Configure real-user monitoring (RUM)

## 5.3 Uptime Monitoring

- [ ] Configure health check endpoint: `/api/health`
- [ ] Set up external monitoring (UptimeRobot/Pingdom)
- [ ] Configure alerting (5xx errors > 1% → immediate alert)
- [ ] Set up status page (status.primariata.work)

---

# Phase 6: Launch Sequence

---

## 6.1 Final Verification

- [ ] Manual smoke test on staging environment
- [ ] Verify all critical user flows:
  - Registration + Email verification
  - Login with Google OAuth
  - Location selection
  - Submit request (cerere)
  - Upload document
  - Track request status
- [ ] Test from multiple devices (mobile, tablet, desktop)
- [ ] Test from different browsers (Chrome, Firefox, Safari, Edge)

## 6.2 Go-Live

1. [ ] Merge `develop` → `main` branch
2. [ ] Tag release: `git tag -a v1.0.0 -m "Production launch"`
3. [ ] Push to GitHub: `git push origin main --tags`
4. [ ] Vercel auto-deploys from `main` branch
5. [ ] Monitor deployment logs in Vercel dashboard
6. [ ] Verify deployment success (health check returns 200)

## 6.3 DNS Propagation

- [ ] Update Cloudflare DNS to point to Vercel
  - [ ] Wait for DNS propagation (usually < 5 minutes)
  - [ ] Verify site is accessible at <https://primariata.work>
  - [ ] Test SSL certificate (A+ rating on SSL Labs)
- 

# Phase 7: Post-Deployment Validation

---

## 7.1 Functional Testing

- [ ] Complete registration flow (new user)
- [ ] Login with Google OAuth
- [ ] Submit test request (cerere)
- [ ] Upload test document
- [ ] Verify email notifications work
- [ ] Test payment flow (sandbox mode)
- [ ] Download signed document

## 7.2 Performance Validation

- [ ] Run Lighthouse audit (target: 90+ all metrics)
- [ ] Check Core Web Vitals:
  - LCP < 2.5s
  - FID < 100ms
  - CLS < 0.1
- [ ] Verify CDN caching (check response headers)

- [ ] Test page load from different geographic locations

## 7.3 Security Validation

- [ ] Verify HTTPS is enforced (HTTP → HTTPS redirect)
  - [ ] Test rate limiting (attempt brute force)
  - [ ] Verify RLS policies (attempt unauthorized access)
  - [ ] Check security headers (CSP, HSTS, X-Frame-Options)
  - [ ] Run OWASP ZAP security scan
- 

## Phase 8: Rollback Plan

---

### If Critical Issues Detected:

**Immediate Actions (< 5 min):** 1. [ ] Enable maintenance mode: Set `NEXT_PUBLIC_MAINTENANCE_MODE=true` 2. [ ] Revert Vercel deployment to previous stable version 3. [ ] Notify team via Slack 4. [ ] Post status update on status page

**Database Rollback (if needed):** 1. [ ] Stop all write operations (enable read-only mode) 2. [ ] Restore from Point-in-Time Recovery (PITR) 3. [ ] Verify data integrity 4. [ ] Re-enable write operations

**Post-Incident:** 1. [ ] Document issue in incident report 2. [ ] Root cause analysis (RCA) 3. [ ] Update deployment checklist to prevent recurrence

---

## Post-Launch Monitoring

---

### Critical Metrics to Watch:

- **Error Rate:** < 0.1% (alert if > 1%)
- **Response Time:** p95 < 500ms (alert if > 1s)
- **Uptime:** 99.9% (alert if < 99%)
- **User Registration Rate:** Monitor for anomalies
- **Database Connections:** Monitor pool utilization

## Daily Tasks (First Week):

- [ ] Review Sentry errors (morning & evening)
  - [ ] Check Vercel Analytics for traffic patterns
  - [ ] Monitor database performance (query times, connection count)
  - [ ] Review user feedback (support emails, surveys)
  - [ ] Update changelog with any hotfixes deployed
- 

## Success Criteria

---

1. All automated tests pass (unit, integration, e2e)
  2. Lighthouse scores > 90 (Performance, Accessibility, Best Practices, SEO)
  3. Zero critical errors in first 24 hours
  4. User registration flow completes successfully
  5. Payment integration works (sandbox mode)
  6. Email notifications are delivered
  7. SSL certificate is valid (A+ rating)
  8. CDN caching is working (verified via headers)
  9. Monitoring and alerting are operational
  10. Team can access logs and metrics
- 

## Contacts & Resources

---

**Team:** - Octavian Mihai (Full-Stack, DevOps) - [mihai.g.octavian24@stud.rau.ro](mailto:mihai.g.octavian24@stud.rau.ro) - Bianca-Maria Abbasi Pazeyazd (Frontend, UI/UX) - [abbasipazeyazd.h.biancamaria24@stud.rau.ro](mailto:abbasipazeyazd.h.biancamaria24@stud.rau.ro)

**Services:** - Vercel Dashboard: <https://vercel.com/primariata-work> - Supabase Dashboard: <https://supabase.com/dashboard/project/xxx> - Cloudflare Dashboard: <https://dash.cloudflare.com> - Sentry: <https://sentry.io/organizations/primariata/> - GitHub Repo: <https://github.com/mihaigoctavian24/primariata.work>

**Emergency Rollback Command:**



```
# Revert to previous deployment
vercel rollback [deployment-url]

# Or via Vercel Dashboard:
# Deployments → Previous → Promote to Production
```

---

**Last Updated:** 2025-12-19