

Production Readiness Checklist

Обзор

этот checklist поможет убедиться, что ваше приложение готово к production deployment. Отмечайте пункты по мере выполнения.

1. Безопасность

Критично

- [x] Удален автоматический debug логин
- Проверка: `app/auth/signin/page.tsx` не содержит `handleDebugLogin`
- [x] NextAuth DEBUG режим отключен
- Проверка: `lib/auth-options.ts` имеет `debug: false`
- [x] Credentials provider доступен только в dev/test
- Проверка: `lib/auth-options.ts` имеет условный provider
- `ENABLE_TEST_AUTH=true` НЕ установлена в production
- [x] Rate limiting middleware работает
- Проверка: `middleware.ts` существует и настроен

Важно

- [] HTTPS сертификат настроен
 - Let's Encrypt, Cloudflare, или другой провайдер
 - [] CORS настроен правильно
 - Только доверенные origins
 - [] Security headers настроены
 - CSP, X-Frame-Options, etc.
-

2. Environment Variables

Database

- [] `DATABASE_URL`

```
bash
# Проверка
echo $DATABASE_URL | grep "postgresql://"
```
- [] Production база данных (не localhost)

- [] SSL включен (если нужно)
- [] Connection pooling настроен

NextAuth

- [] **NEXTAUTH_SECRET**

```
bash
# Генерация
openssl rand -base64 32
```

- [] Минимум 32 символа
- [] Уникальный для production

- [] **NEXTAUTH_URL**

```
env
NEXTAUTH_URL=https://ваш-домен.com
```

- [] HTTPS (не HTTP)
- [] Правильный домен
- [] Без trailing slash

Email Provider

- [] **EMAIL_SERVER_HOST**

- [] SMTP сервер настроен (Resend, SendGrid, AWS SES)

- [] **EMAIL_SERVER_PORT**

- [] 587 (TLS) или 465 (SSL)

- [] **EMAIL_SERVER_USER**

- [] Правильный username/API key

- [] **EMAIL_SERVER_PASSWORD**

- [] API key или password

- [] **EMAIL_FROM**

- [] Верифицированный email address

- [] **DNS записи настроены**

- [] SPF record

- [] DKIM record

- [] DMARC record

Stripe

- [] **STRIPE_PUBLISHABLE_KEY**

```
bash
# Проверка
echo $STRIPE_PUBLISHABLE_KEY | grep "pk_live_"
• [ ] Live key (не pk_test_)
```

- [] **STRIPE_SECRET_KEY**

```
bash
# Проверка
echo $STRIPE_SECRET_KEY | grep "sk_live_"
```

- [] Live key (не sk_test_)

- [] Никому не показывали

- [] **STRIPE_WEBHOOK_SECRET**

```
bash
# Проверка
echo $STRIPE_WEBHOOK_SECRET | grep "whsec_"
```

- [] Live webhook secret

- [] Webhook endpoint настроен в Stripe Dashboard

AWS S3

- [] **AWS_BUCKET_NAME**

- [] Production bucket

- [] CORS настроен

- [] Permissions правильные

- [] **AWS_FOLDER_PREFIX** (опционально)

- [] Если используете, проверьте правильность

Abacus AI (опционально)

- [] **ABACUSA1_API_KEY**

- [] Если используете document-to-SOP generation
-

3. Stripe Configuration

Account Setup

- [] **Stripe account verified**

- [] Business details complete

- [] Bank account connected

- [] Identity verified

- [] **Payment methods enabled**

- [] Cards enabled

- [] Other methods as needed

Webhook Configuration

- [] **Production webhook endpoint created**

```
https://ваш-домен.com/api/webhooks/stripe
```

- [] HTTPS URL

- [] Events configured:
 - checkout.session.completed
 - payment_intent.payment_failed
- [] Signing secret copied to .env
- [] **Webhook tested**
- [] Test payment successful
- [] Webhook received (200 status)
- [] Database updated correctly

Testing

- [] **Test payment completed**
 - [] Small amount (\$0.50)
 - [] Verified in Stripe Dashboard
 - [] Purchase record created
 - [] Revenue record created
 - [] User can access SOP
-

4. Email Configuration

Provider Setup

- [] **Email provider chosen**
- Resend (рекомендуется)
- SendGrid
- AWS SES
- Mailgun
- Другой
- [] **Account created and verified**
- [] **Domain verified**
- [] DNS records added
- [] Verification complete

Testing

- [] **Magic link email works**

```
bash
# Test
# 1. Go to /auth/signin
# 2. Enter email
# 3. Check email inbox
# 4. Click magic link
# 5. Verify login successful
```

- [] **Email deliverability**

- [] Not in spam folder
 - [] Proper sender name
 - [] Links work correctly
-

5. Database

Production Database

- [] **Database created**
- [] Separate from development
- [] Backups configured

- [] **Migrations run**

```
bash
yarn prisma migrate deploy
```

- [] **Seeding**

- [] НЕ запускайте seed в production!
- [] Создайте categories вручную

Performance

- [] **Indexes настроены**
 - [] Prisma schema имеет @@index
 - [] **Connection pooling**
 - [] Configured in DATABASE_URL
 - [] Appropriate pool size
-

6. Build & Deployment

Pre-Deployment

- [] **TypeScript compilation successful**

```
bash
yarn tsc --noEmit
```

- [] **Production build successful**

```
bash
yarn build
```

- [] **No critical warnings**

- [] Check build output
- [] Review any errors

- [] **Environment validated**

```
bash
yarn validate-env --mode=production
```

Deployment

- [] Deployment method chosen

- Vercel
- Docker
- VPS
- Другой

- [] Environment variables set

- [] Все production variables
- [] Проверены на опечатки

- [] Deployment successful

- [] Application accessible
 - [] No 500 errors
 - [] Assets loading correctly
-

7. Post-Deployment Testing

Core Functionality

- [] Homepage loads

- [] No console errors
- [] Images load
- [] Styles correct

- [] Authentication

- [] Signup works
- [] Signin works (magic link)
- [] Logout works
- [] Protected routes redirect correctly

- [] SOP Creation (для sellers)

- [] Create new SOP
- [] Upload images
- [] Add steps
- [] Publish SOP

- [] Marketplace

- [] Browse SOPs
- [] Search works
- [] Filters work
- [] Categories work

- [] Shopping Cart

- [] Add to cart
- [] View cart
- [] Remove from cart
- [] Cart counter updates
- [] **Checkout**
- [] Stripe Checkout opens
- [] Payment succeeds
- [] Redirect to success page
- [] Purchase appears in dashboard
- [] Can download/access SOP
- [] **SOP Execution**
- [] Start session
- [] Navigate steps
- [] Timers work
- [] Questions work
- [] Complete session

Edge Cases

- [] **Failed payment handling**
 - [] Use test card 4000 0000 0000 9995
 - [] Error message shown
 - [] Purchase not created
 - [] **Free SOPs**
 - [] No payment required
 - [] Immediate access
 - [] **Already owned SOPs**
 - [] Can't add to cart
 - [] Can access directly
-

8. Monitoring & Analytics

Error Tracking

- [] **Sentry configured** (рекомендуется)
- [] DSN configured
- [] Source maps uploaded
- [] Alerts set up

Application Monitoring

- [] **Logs accessible**
- [] Server logs

- [] Application logs
- [] Error logs
- [] **Uptime monitoring**
- [] Uptime Robot
- [] Pingdom
- [] Или другой сервис

Stripe Monitoring

- [] **Stripe Dashboard alerts**
 - [] Failed payments
 - [] Disputes
 - [] Unusual activity
 - [] **Webhook monitoring**
 - [] Check delivery status
 - [] Review failed webhooks
-

9. Performance

Frontend

- [] **Lighthouse score**
- [] Performance: >80
- [] Accessibility: >90
- [] Best Practices: >90
- [] SEO: >90
- [] **Images optimized**
- [] Next.js Image component used
- [] WebP format where possible

Backend

- [] **API response times**
 - [] < 500ms for most endpoints
 - [] Database queries optimized
 - [] **Rate limiting working**
 - [] Test hitting limits
 - [] 429 responses return correctly
-

10. Legal & Compliance

Policies

- [] **Privacy Policy**
- [] Mentions data collection
- [] Explains cookie usage
- [] Mentions Stripe
- [] Accessible from footer

- [] **Terms of Service**
- [] Payment terms clear
- [] Refund policy stated
- [] User responsibilities
- [] Accessible from footer

- [] **Refund Policy**

- [] Clear guidelines
- [] Timeframe specified
- [] Process explained

Contact

- [] **Support email configured**

- [] Monitored regularly
- [] Response time defined

- [] **Contact page/form**

- [] Easy to find
 - [] Works correctly
-

11. Security Audit

Code Review

- [] **No hardcoded secrets**

- [] All secrets in .env
- [] .env in .gitignore

- [] **No console.logs with sensitive data**

- [] Review all console.log statements
- [] Remove or guard with conditions

- [] **SQL injection prevention**

- [] Using Prisma (parameterized queries)
- [] **XSS prevention**

- [] React escaping (built-in)
- [] Validate user input

Dependencies

- [] **Dependencies up to date**

```
bash
yarn outdated
```

- [] **No critical vulnerabilities**

```
bash
yarn audit
```

- [] **Unused packages removed**
-

12. Documentation

Internal

- [] **README.md updated**
- [] Current features listed
- [] Setup instructions clear
- [] **DEPLOYMENT.md reviewed**
- [] Deployment process documented
- [] Troubleshooting guide
- [] **Environment variables documented**
- [] All required variables listed
- [] Examples provided

External

- [] **User documentation** (если нужно)
 - [] How to create SOPs
 - [] How to buy SOPs
 - [] FAQ
-

13. Backup & Recovery

Database Backup

- [] **Automated backups configured**
- [] Daily backups
- [] Retention policy set
- [] **Backup restoration tested**
- [] Test restore process

- [] Verify data integrity

Application Backup

- [] **S3 versioning enabled**
 - [] File versioning on
 - [] Lifecycle policies set
 - [] **Code repository**
 - [] Latest code pushed
 - [] Tags for releases
-

14. Final Checks

Pre-Launch

- [] **All checkboxes above completed**
- [] **Team notified**
- [] Launch time communicated
- [] Responsibilities assigned
- [] **Support ready**
- [] Support email monitored
- [] Response process defined

Launch

- [] **DNS updated** (если используете custom domain)
 - [] A record
 - [] CNAME record
 - [] Propagation complete
- [] **SSL certificate active**
 - [] HTTPS working
 - [] No certificate warnings
- [] **Application accessible**
 - [] Main domain loads
 - [] All pages accessible
 - [] No broken links

Post-Launch

- [] **Monitor for 24 hours**
- [] Check error rates
- [] Review logs
- [] Monitor performance

- [] **First real transaction**

- [] Monitor closely

- [] Verify all steps

- [] Confirm revenue split

- [] **Celebrate!** 🎉

- [] You launched!

- [] Document lessons learned

- [] Plan next iteration

Автоматическая Проверка

Запустите скрипт для автоматической проверки environment variables:

```
# Development  
yarn validate-env
```

```
# Production  
yarn validate-env --mode=production
```

Документ обновлен: 26 ноября 2025

Версия: 1.0

Статус:  Ready for Use