MatSelamat.com - Complete Deployment Guide

🚀 Production Deployment Checklist

Phase 1: Pre-Launch Setup (Week 1-2)

1. Domain & Hosting

```
bash

# Domain: matselamat.com (Already registered)
# DNS Configuration:
A     @          → Your server IP
CNAME www        → matselamat.com
TXT   _dmarc     → v=DMARC1; p=none; rua=mailto:dmarc@matselamat.com
```

Recommended Hosting:

- **Frontend**: Vercel (Free tier, automatic SSL, global CDN)
- **Backend**: Railway.app or Render (RM50/month, includes PostgreSQL)
- **Alternative**: AWS Lightsail (RM20/month for full control)

2. Stripe Setup

```
bash

# 1. Create Stripe Malaysia account: https://dashboard.stripe.com/register
# 2. Enable payment methods:
  - Cards (Visa, Mastercard, Amex)
  - FPX (Malaysian online banking)
  - GrabPay, Touch 'n Go (e-wallets)

# 3. Configure webhook endpoint:
  URL: https://api.matselamat.com/api/webhooks/stripe
  Events to listen:
  - checkout.session.completed
  - payment_intent.payment_failed
  - charge.refunded

# 4. Get your keys:
  STRIPE_PUBLIC_KEY=pk_live_...
  STRIPE_SECRET_KEY=sk_live_...
  STRIPE_WEBHOOK_SECRET=whsec_...
```

3. Anthropic API Setup

```bash
# 1. Get API key: https://console.anthropic.com
# 2. Set usage limits to control costs:
  Max tokens per request: 2000
  Monthly budget: RM500 (covers ~125 claims at RM4/claim)

# Environment variable:
ANTHROPIC_API_KEY=sk-ant-api03-...
```

4. Database Setup (PostgreSQL)

```sql
```

```sql
-- Database schema
CREATE TABLE claims (
    id VARCHAR(255) PRIMARY KEY,
    status VARCHAR(50) NOT NULL,

    -- Claimant details
    claimant_name VARCHAR(255) NOT NULL,
    claimant_ic VARCHAR(20) NOT NULL,
    claimant_address TEXT NOT NULL,
    claimant_phone VARCHAR(20),
    claimant_email VARCHAR(255) NOT NULL,

    -- Defendant details
    defendant_name VARCHAR(255) NOT NULL,
    defendant_address TEXT NOT NULL,

    -- Claim details
    claim_type VARCHAR(50) NOT NULL,
    claim_amount DECIMAL(10, 2) NOT NULL,
    claim_description TEXT NOT NULL,

    -- Payment tracking
    stripe_session_id VARCHAR(255),
    payment_intent_id VARCHAR(255),
    paid_at TIMESTAMP,
    refunded_at TIMESTAMP,

    -- Document
    document_url VARCHAR(500),
    document_generated_at TIMESTAMP,

    -- Metadata
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

CREATE INDEX idx_claims_email ON claims(claimant_email);
CREATE INDEX idx_claims_status ON claims(status);
CREATE INDEX idx_claims_created ON claims(created_at DESC);

-- Analytics table
CREATE TABLE events (
    id SERIAL PRIMARY KEY,
    event_type VARCHAR(100) NOT NULL,
    event_data JSONB,
    user_id VARCHAR(255),
```

```
    ip_address VARCHAR(45),
    user_agent TEXT,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

CREATE INDEX idx_events_type ON events(event_type);
CREATE INDEX idx_events_created ON events(created_at DESC);
```

Phase 2: Frontend Deployment

Vercel Deployment (Recommended)

bash

```
# 1. Install Vercel CLI
npm install -g vercel

# 2. Deploy
cd matselamat-frontend
vercel --prod

# 3. Configure environment variables in Vercel dashboard:
NEXT_PUBLIC_API_URL=https://api.matselamat.com
NEXT_PUBLIC_STRIPE_PUBLIC_KEY=pk_live_...
NEXT_PUBLIC_GOOGLE_ANALYTICS=G-...
```

Build Configuration (next.config.js)

javascript

```
module.exports = {
  reactStrictMode: true,
  i18n: {
    locales: ['en', 'ms', 'zh', 'ta'],
    defaultLocale: 'en',
  },
  images: {
    domains: ['matselamat.com'],
  },
  // Security headers
  async headers() {
    return [
      {
        source: '/:path*',
        headers: [
          {
            key: 'X-Frame-Options',
            value: 'DENY',
          },
          {
            key: 'X-Content-Type-Options',
            value: 'nosniff',
          },
          {
            key: 'Referrer-Policy',
            value: 'strict-origin-when-cross-origin',
          },
        ],
      },
    ];
  },
};
```

Phase 3: Backend Deployment

Railway.app Deployment (Recommended)

bash

```
# 1. Connect GitHub repository
# 2. Add PostgreSQL plugin (automatic)
# 3. Configure environment variables:

NODE_ENV=production
PORT=3001
DATABASE_URL=${{Postgres.DATABASE_URL}}  # Auto-populated
STRIPE_SECRET_KEY=sk_live_...
STRIPE_WEBHOOK_SECRET=whsec_...
ANTHROPIC_API_KEY=sk-ant-...
FRONTEND_URL=https://matselamat.com

# Email service (choose one):
SENDGRID_API_KEY=SG....
# OR
MAILGUN_API_KEY=key-...
MAILGUN_DOMAIN=mg.matselamat.com

# 4. Deploy
railway up
```

Phase 4: SSL & Security

1. SSL Certificate (Automatic with Vercel/Railway)

✅ Vercel provides automatic SSL
✅ Railway provides automatic SSL
✅ Force HTTPS redirect enabled

2. Security Headers

nginx

```
# Add to nginx if self-hosting
add_header Strict-Transport-Security "max-age=31536000; includeSubDomains" always;
add_header X-Frame-Options "DENY" always;
add_header X-Content-Type-Options "nosniff" always;
add_header X-XSS-Protection "1; mode=block" always;
add_header Referrer-Policy "strict-origin-when-cross-origin" always;
```

3. Rate Limiting (Prevent API abuse)

javascript

```
// Add to Express server
const rateLimit = require('express-rate-limit');

const limiter = rateLimit({
  windowMs: 15 * 60 * 1000, // 15 minutes
  max: 100, // limit each IP to 100 requests per windowMs
  message: 'Too many requests, please try again later.'
});

app.use('/api/', limiter);

// Stricter limit for AI endpoints
const aiLimiter = rateLimit({
  windowMs: 60 * 60 * 1000, // 1 hour
  max: 10, // 10 AI calls per hour per IP
});

app.use('/api/ai/', aiLimiter);
```

Phase 5: Monitoring & Analytics

1. Error Tracking (Sentry)

bash

```
# Install
npm install @sentry/node @sentry/react

# Configure (server.js)
const Sentry = require('@sentry/node');
Sentry.init({
  dsn: 'https://...@sentry.io/...',
  environment: process.env.NODE_ENV,
  tracesSampleRate: 1.0,
});

# Configure (frontend)
import * as Sentry from '@sentry/react';
Sentry.init({
  dsn: 'https://...@sentry.io/...',
  integrations: [new Sentry.BrowserTracing()],
  tracesSampleRate: 0.1,
});
```

2. Analytics (Mixpanel)

```javascript
// Track key events
mixpanel.track('Page View', { page: 'Landing' });
mixpanel.track('Eligibility Check Started');
mixpanel.track('Claim Builder Started');
mixpanel.track('Payment Initiated', { amount: 79 });
mixpanel.track('Payment Completed', { amount: 79, claimId: 'xxx' });
mixpanel.track('Document Downloaded', { format: 'PDF' });
```

3. Uptime Monitoring

```
Service: UptimeRobot (Free)
Monitor: https://matselamat.com (every 5 minutes)
Monitor: https://api.matselamat.com/health (every 5 minutes)
Alerts: Email + SMS when down
```

Phase 6: Email Setup

SendGrid Configuration (Recommended)

```javascript
const sgMail = require('@sendgrid/mail');
sgMail.setApiKey(process.env.SENDGRID_API_KEY);

async function sendDocumentEmail(email, documentData) {
  const msg = {
    to: email,
    from: 'noreply@matselamat.com',
    subject: 'Dokumen Tuntutan Kecil Anda Sudah Siap! 📄 ',
    templateId: 'd-...', // SendGrid dynamic template ID
    dynamicTemplateData: {
      claimantName: documentData.claimantName,
      claimAmount: documentData.claimAmount,
      downloadLink: `https://matselamat.com/documents/${documentData.claimId}`,
      expiryDate: '30 hari'
    }
  };

  await sgMail.send(msg);
}
```

Phase 7: Backup & Recovery

1. Database Backups

```bash
# Railway automatic daily backups ✅
# Manual backup script:
pg_dump $DATABASE_URL > backup_$(date +%Y%m%d).sql

# Restore:
psql $DATABASE_URL < backup_20250115.sql
```

2. Document Storage (AWS S3 or Cloudflare R2)

```javascript
const AWS = require('aws-sdk');
const s3 = new AWS.S3({
  accessKeyId: process.env.AWS_ACCESS_KEY,
  secretAccessKey: process.env.AWS_SECRET_KEY,
  region: 'ap-southeast-1'
});

async function uploadDocument(claimId, pdfBuffer) {
  const params = {
    Bucket: 'matselamat-documents',
    Key: `documents/${claimId}.pdf`,
    Body: pdfBuffer,
    ContentType: 'application/pdf',
    Metadata: {
      claimId: claimId,
      generatedAt: new Date().toISOString()
    }
  };

  const result = await s3.upload(params).promise();
  return result.Location;
}
```

💰 Cost Breakdown

Monthly Operating Costs (Conservative Estimate)

Service	Plan	Cost (RM)
Vercel Frontend	Hobby	RM0 (free)
Railway Backend + DB	Starter	RM50
Domain (GoDaddy)	Annual/12	RM10

Service	Plan	Cost (RM)
SendGrid Email	Essential	RM70
Stripe Fees	2.9% + RM1	Variable*
Anthropic API	Pay-as-go	Variable**
Sentry (Error tracking)	Developer	RM0 (free)
UptimeRobot	Free	RM0
Total Fixed		**RM130/month**

*Stripe fees: For 300 claims @ RM79 = ~RM790/month
**Anthropic API: 300 claims @ RM4 = RM1,200/month

Break-Even Analysis

```
Fixed costs: RM130/month
Variable costs per claim: RM27.70 (Stripe RM23.70 + AI RM4)
Revenue per claim: RM79

Contribution margin: RM51.30/claim
Break-even: 130 ÷ 51.30 = 3 claims/month (easily achievable)

At 300 claims/month:
Revenue: RM23,700
Total costs: RM9,440 (fixed + variable)
Net profit: RM14,260/month (60% margin)
```

🧪 Testing Checklist

Pre-Launch Testing

1. Functional Testing

- [ ] Eligibility checker flows correctly
- [ ] All form validations work
- [ ] AI refinement produces good output
- [ ] Payment integration works (test mode)
- [ ] Document generation works
- [ ] Email delivery works
- [ ] Download links work (PDF & DOCX)
- [ ] Mobile responsiveness (all screen sizes)
- [ ] Multi-language switching works

2. Payment Testing (Stripe Test Mode)

Test Cards:
✅ Success: 4242 4242 4242 4242
✅ Decline: 4000 0000 0000 0002
✅ 3D Secure: 4000 0027 6000 3184
✅ FPX: Use Stripe test mode banks

3. Security Testing

- [ ] SQL injection attempts blocked
- [ ] XSS attempts blocked
- [ ] Rate limiting works
- [ ] HTTPS enforced
- [ ] Headers properly set
- [ ] API keys not exposed in frontend
- [ ] Webhook signature validation works

4. Performance Testing

- [ ] Page load < 3 seconds
- [ ] API response < 500ms
- [ ] AI refinement < 5 seconds
- [ ] Document generation < 10 seconds
- [ ] Can handle 100 concurrent users

📊 Launch Strategy

Week 1: Soft Launch (Beta)

- **Target**: 20 beta users
- **Source**: Personal network, legal communities
- **Goal**: Collect feedback, fix bugs
- **Budget**: RM0 (organic)

Week 2-4: Public Launch

- **Target**: 100 paying customers
- **Tactics**:
  - Press release (Malaysiakini, The Star)
  - Facebook ads (RM1,000 budget)
  - Reddit r/malaysia post

    - Lowyat Forum thread
    - LinkedIn post (lawyer groups)
- **Goal**: Validate product-market fit

Month 2-3: Growth Phase

- **Target**: 300 customers/month
- **Tactics**:
    - Google Ads (RM2,000/month)
    - SEO content (50 blog posts)
    - YouTube tutorials
    - Influencer partnerships
    - Referral program launch

🔧 Maintenance Schedule

Daily

- Monitor error logs (Sentry)
- Check payment webhooks (Stripe dashboard)
- Respond to support emails

Weekly

- Review analytics (conversion rates)
- A/B test variations
- Update content (FAQ, blog)
- Backup database manually

Monthly

- Financial review (profitability)
- User feedback analysis
- Feature prioritization
- Security updates

🚨 Emergency Procedures

If Site Goes Down

1. Check status page: status.matselamat.com
2. Verify DNS (dig matselamat.com)
3. Check hosting provider (Vercel/Railway)
4. Rollback to previous deployment if needed
5. Notify users via social media

If Payments Fail

1. Check Stripe dashboard for issues
2. Verify webhook endpoint is reachable
3. Check SSL certificate validity
4. Contact Stripe support if needed
5. Manual payment reconciliation

If AI API Fails

1. Switch to fallback provider (OpenAI)
2. Queue requests for retry
3. Notify affected users
4. Manual refinement for urgent cases

📞 Support Infrastructure

Tier 1: Automated (Chatbot)

- FAQ answering
- Process guidance
- Document status check

Tier 2: Email Support

- Response time: < 24 hours
- Email: support@matselamat.com
- Team: 1-2 people

Tier 3: Urgent Issues

- Phone hotline (future)
- WhatsApp business (012-XXX-XXXX)
- Response time: < 2 hours

✅ Go-Live Checklist

Technical

- [ ] All tests passing
- [ ] Production environment variables set
- [ ] SSL certificate active
- [ ] Domain DNS configured
- [ ] Stripe live mode enabled
- [ ] Email delivery tested
- [ ] Backup systems tested
- [ ] Monitoring active
- [ ] Analytics tracking

Legal

- [ ] Terms of Service reviewed by lawyer
- [ ] Privacy Policy compliant with PDPA
- [ ] Refund Policy clear
- [ ] Disclaimer prominent
- [ ] Professional indemnity insurance obtained

Marketing

- [ ] Landing page optimized
- [ ] Social media accounts created
- [ ] Google Analytics configured
- [ ] Facebook Pixel installed
- [ ] Press kit prepared
- [ ] Launch announcement ready

Operations

- [ ] Support email setup
- [ ] Payment reconciliation process
- [ ] Refund process documented
- [ ] Emergency contacts list
- [ ] Runbook created

🎯 Success Metrics (First 90 Days)

Traffic

- Target: 10,000 unique visitors
- Conversion rate: 3% (300 customers)

Revenue

- Target: RM23,700 (300 claims × RM79)
- Actual profit: RM14,000+ (60% margin)

Customer Satisfaction

- NPS score: > 50
- 5-star reviews: > 80%
- Refund rate: < 5%

Technical

- Uptime: > 99.9%
- Page load: < 2s
- API response: < 500ms
- Error rate: < 0.1%

Ready for production. Launch when ready! 🚀