Here’s a comprehensive **Requirements Document** for your system — covering both **development** and **production environments**, ideal for guiding your engineering team and deployment planning.

**📑 Requirements Document: Outmatch — CBE-Based School Recommender System**

**1. Overview**

**Outmatch** is a web platform that provides personalized secondary school recommendations for Kenyan Grade 9 students based on psychometric assessments, academic performance, and CBE career pathway preferences. The system also allows schools to adopt the platform by managing student access through licensing and vouchers.

**2. Functional Requirements**

**🔹 2.1 Student-Side Features**

| **Feature** | **Description** |
| --- | --- |
| Sign Up / Login | Students create or access accounts |
| Personality Test | MBTI-like questions with local context |
| Test Result Processing | Personality → Pathway mapping |
| Student Profile | Location, gender, KCPE score, preferences |
| Recommender Engine | Matches schools using multi-criteria |
| Report Access | View/download school report (14 best-fit schools) |
| M-PESA Payment | Unlock full report access |
| Voucher Support | Unlock report using school-issued voucher |

**🔹 2.2 School-Side Features**

| **Feature** | **Description** |
| --- | --- |
| School Registration | Create school admin account |
| Dashboard Access | Log in and manage students |
| Buy Licenses | Pay via M-PESA to purchase seats |
| Generate Vouchers | Auto-create and assign vouchers to students |
| Upload Students | CSV upload or manual entry |
| Track Student Progress | Test status, performance reports |
| Export Reports | Individual or batch report downloads |

**🔹 2.3 Admin Features**

| **Feature** | **Description** |
| --- | --- |
| View All Users | Manage student and school accounts |
| Manage Schools | Add/edit schools and metadata |
| Manage Questions | Update psychometric questions |
| Approve Payments | Verify and manage payment records |
| Analytics Dashboard | Regional stats, pathway trends, school usage |

**3. Non-Functional Requirements**

| **Category** | **Requirement** |
| --- | --- |
| Performance | Response time < 2s for API; < 5s for reports |
| Scalability | Handle 10K+ concurrent users (especially during Form 1 selection) |
| Security | HTTPS, hashed passwords, token-based auth |
| Localization | Kenyan context, Swahili/English support optional |
| Uptime | 99.5% uptime target on production |
| Backup | Daily backups with 7-day retention |
| Mobile-Friendly | Responsive UI for smartphones & tablets |

**4. Technical Requirements**

**🔹 4.1 Frontend (Client)**

| **Item** | **Stack** |
| --- | --- |
| Framework | React.js (or Vue.js) |
| Styling | Tailwind CSS |
| Authentication | JWT or Session-based |
| Payment UI | M-PESA STK Integration |
| PDF Reports | jsPDF / Print-to-PDF support |
| Deployment | Netlify / Vercel / Render (for static) |

**🔹 4.2 Backend (Server)**

| **Item** | **Stack** |
| --- | --- |
| Language | Python |
| Framework | Django + Django REST Framework |
| Database | PostgreSQL |
| File Storage | Cloudinary / AWS S3 |
| Background Jobs | Celery + Redis (for email, report generation) |
| Authentication | Django Auth (Token or JWT) |
| Payment API | Safaricom Daraja API |
| Deployment | Render / Railway / DigitalOcean / AWS EC2 |

**5. Development Environment Setup**

| **Component** | **Spec** |
| --- | --- |
| Python | 3.10+ |
| Node.js | 18+ |
| Database | PostgreSQL (v14 or higher) |
| Virtual Env | venv or poetry |
| Package Manager | pip / poetry |
| Tools | Postman, Docker (optional), Git |
| Email | Mailtrap (dev), Gmail SMTP (prod) |
| Secrets | .env with django-environ or decouple |

**6. Production Environment**

| **Component** | **Spec** |
| --- | --- |
| Domain | outmatch.co.ke or custom school domains |
| Web Server | Gunicorn + Nginx |
| SSL | Let’s Encrypt via Certbot |
| Database | Managed PostgreSQL (e.g. Supabase, Railway, Heroku) |
| Object Storage | AWS S3 or Cloudinary |
| CI/CD | GitHub Actions or Railway Auto-deploy |
| Monitoring | UptimeRobot, Sentry (for errors), Grafana (optional) |
| Payment API | Safaricom M-PESA Daraja (live credentials) |

**7. Security Considerations**

* Password hashing using Django’s PBKDF2
* Token expiration and refresh support
* Secure CORS policies and CSRF tokens
* Input validation (forms and API)
* Rate limiting login and payment endpoints
* Voucher codes expire or become single-use

**8. Reporting & Analytics**

* Top pathways by region, gender, or school
* Test completion stats
* License/voucher usage per school
* Downloadable student-level reports (PDF)
* School ranking breakdown by fit factors

**9. System Architecture Diagram (Simplified)**

[Student / School UI] --> [React Frontend]

|

v

[Django API Server]

| | |

v v v

[PostgreSQL] [S3] [M-PESA API]

|

v

[Recommender Engine]

|

v

[School & Pathway DB]