PrepNexa AI

AI-Powered Exam Question Prediction Platform

---

1. Document Overview

Product Name: PrepNexa AI
Document Type: Product Requirements Document (PRD)
Version: 1.0

---

2. Product Vision

To democratize exam success by leveraging AI to predict high-probability questions and answers for any exam—competitive, university, or professional—using historical question patterns, syllabus intelligence, and data-driven insights.

---

3. Problem Statement

Exam aspirants face critical challenges:

- Extremely large syllabi with limited preparation time
- No clarity on high-weightage or frequently asked topics
- Preparation driven by guesswork instead of data
- Lack of personalization based on exam, subject, or trends

PrepNexa AI aims to replace guesswork with predictive intelligence.

---

4. Target Users

Primary Users

- Competitive exam aspirants (UPSC, SSC, Banking, NEET, JEE, etc.)
- University and college students , school students, coaching classes (semester, internal, final exams)
- Professional certification aspirants

Secondary Users

- Coaching institutes
- Faculty and content creators
- Academic institutions

---

5. User Personas

Student / Aspirant (Primary User)

- Can upload previous year question papers
- Can generate predicted papers
- Can practice, revise, and track preparation
- Acts as both contributor and consumer of content

---

6. Functional Requirements

6.1 User Authentication & Profile Management

FR-01: Users shall be able to sign up and log in using:

- Email & password
- Mobile number (OTP-based)
- Google OAuth

FR-02: Users shall be able to configure their profile by selecting:

- Exam type (Competitive / University / Professional)
- Exam name
- University / Board (if applicable)
- Course / Subject
- Exam year / semester

6.2 Exam, Syllabus & Content Management (User-Driven)

FR-03: Any authenticated user shall be able to upload:

- Previous Year Question Papers (PDF, DOC, Images)
- Syllabi (PDF / Text)

FR-04: System shall automatically extract and map questions to:

- Subject
- Topic
- Sub-topic
- Marks
- Question type

FR-05: System shall use AI confidence scoring and community validation instead of manual admin correction.

6.3 User-Uploaded Question Papers

FR-06: Students/aspirants shall be able to upload previous year question papers.

FR-07: System shall:

- Extract questions using OCR and NLP
- Detect exam pattern and structure
- De-duplicate existing questions

FR-08: Uploaded papers shall contribute to:

- User-specific question banks
- Global dataset (after admin validation)

6.4 Question Bank & Browsing

FR-09: Users shall be able to browse:

- PYQs by year
- Questions by subject/topic
- High-frequency questions
- Recently repeated questions

FR-10: Each question shall store metadata:

- Year
- Exam type
- Subject
- Topic
- Marks
- Difficulty level

6.5 AI Prediction Engine

FR-11: System shall analyze:

- Question repetition frequency
- Topic weightage trends
- Difficulty distribution
- Recent exam trends

FR-12: System shall generate:

- Important question lists
- Topic-wise priority lists
- Predicted question papers

FR-13: Each predicted question shall include:

- Probability score (%)
- Topic linkage
- Historical justification

6.6 Personalized Learning & Insights

FR-14: System shall provide personalized insights based on:

- User's selected exam
- Uploaded papers
- Preparation timeline

FR-15: Users shall receive:

- Study recommendations
- High-ROI topics
- Revision alerts

6.7 Search & Semantic Query

FR-16: Users shall be able to search questions using:

- Keywords
- Natural language queries

FR-17: System shall use semantic search to return relevant results even if keywords differ.

6.8 Reports & Export

FR-18: Users shall be able to:

- Download predicted papers (PDF)

- Export important questions

---

7. Non-Functional Requirements

7.1 Performance

- API response time < 500 ms for standard queries
- Support 10,000+ concurrent users
- AI predictions generated within acceptable batch limits

7.2 Scalability

- System shall support millions of questions
- Horizontally scalable backend and vector database

7.3 Security

- JWT-based authentication
- Encrypted user data
- Role-based access (Admin / User)

7.4 Reliability

- 99.5% uptime SLA
- Automated backups

7.5 Usability

- Mobile-first design
- Simple onboarding
- Low learning curve for students

7.6 Compliance

- GDPR-aligned data handling
- Secure storage of user-uploaded documents

---

8. Technical Requirements

Frontend

- Flutter / React Native
- Android-first

Backend

- FastAPI (preferred)
- REST APIs

AI / ML Stack

- OCR: Tesseract / Vision APIs
- LLMs (RAG)
- Embeddings for similarity

Databases

- PostgreSQL
- Vector DB (Pinecone)
- Object storage (AWS S3)

---

9. API Endpoints (Examples)

Authentication

- `POST /auth/register`
- `POST /auth/login`

Exam & Profile

- `GET /exams`

- `POST /user/profile`

Upload Question Papers

- `POST /papers/upload`
- `GET /papers/{paper_id}`

Question Bank

- `GET /questions?topic=&difficulty=`

AI Prediction

- `POST /predict/paper`
- `GET /predict/important-questions`

Practice & Analytics

- `POST /practice/start`
- `POST /practice/submit`
- `GET /analytics/weak-topics`

---

10. Assumptions & Dependencies

- Availability of quality historical question papers
- OCR accuracy for scanned documents
- Regular syllabus updates

---

10. Future Enhancements (Out of Scope v1)

- AI-generated model answers
- Mock test creation & evaluation
- Faculty dashboards
- Monetization & subscription plans
- Multi-language support

---

11. Success Metrics

- Prediction accuracy improvement over time
- Active users per exam
- User retention rate
- Reduction in preparation time

End of Document