

Mongoose Models & Relationships

Model	Fields & Constraints
User	name: String, required, trim email: String, required, unique, lowercase phone: String role: 'ADMIN' 'STUDENT' 'SUPERVISOR' (default 'STUDENT') passwordHash: String, required emailVerifiedAt: Date otp: { code, expiresAt } step1LockedAt: Date refreshTokens: [{ token, createdAt, userAgent, revokedAt }] Meta: timestamps
Exam	userId: ObjectId ref 'User', required, indexed step: Number enum [1,2,3], required levels: ['A1'..'C2'], required status: 'CREATED'..'EXPIRED' (default 'CREATED') startedAt, expiresAt, submittedAt: Date questions: [ObjectId ref 'Question'], required attempts: [{ questionId, selectedKey, isCorrect }] scorePercent: Number awardedLevel: 'A1'..'C2' null nextStepUnlocked: Boolean (default false) Indexes: { userId, step, status } Meta: timestamps
Question	competency: String, required (soft ref to Competency.name) level: 'A1'..'C2', required text: String, required options: [{ key, text }] correctKey: String, required tags: [String] active: Boolean (default true) Indexes: { competency, level, active } Meta: timestamps
Competency	name: String, required, unique description: String active: Boolean (default true) Meta: timestamps
Certificate	userId: ObjectId ref 'User', required, indexed highestLevel: 'A1'..'C2', required sourceExamId: ObjectId ref 'Exam', required serial: String, required, unique pdfPath: String Meta: timestamps
Policy	step1, step2, step3: { thresholds: [{min, award}], lockStep1Below, unlockNextAt } Meta: timestamps

Relationships & Cardinality

User 1 — * Exam
User 1 — * Certificate
Exam 1 — * Attempt (embedded, references Question)
Exam * — * Question (via Exam.questions[])
Certificate * — 1 Exam
Question * — 1 Competency (soft link by name)
Policy — Standalone