

Learning Management System (LMS)

Complete Database Schema Documentation

Total Count

- **PostgreSQL Tables:** 28
- **MongoDB Collections:** 3
- **Total Entities:** 31

Tables by Module

1. **User & Team Management:** 5 tables
2. **Courses & Modules:** 7 tables
3. **Assignments & Tests:** 10 tables
4. **Progress & Gamification:** 6 tables
5. **Notifications & Audit:** 2 tables
6. **Feedback & Communication:** 1 table

Key Features

- All tables have UUIDs as primary keys
 - All tables have created_at and updated_at timestamps
 - All foreign keys properly defined with CASCADE rules
 - All status/enum fields use CHECK constraints
 - All score/percentage fields have range validation
 - 32 strategic indexes for performance
 - Complete audit trail support
 - Soft delete support via status fields
 - Many-to-many relationships properly implemented
-

Database Design Principles Applied

1. **Normalization:** 3rd Normal Form maintained throughout
2. **Referential Integrity:** All foreign keys enforced
3. **Data Validation:** CHECK constraints on all enums and ranges
4. **Audit Trail:** Created/updated timestamps everywhere
5. **Soft Deletes:** Status fields for preserving history
6. **Performance:** Strategic indexes on frequently queried columns
7. **Scalability:** UUID primary keys for distributed systems

Hybrid Database Strategy

- **PostgreSQL:** Relational data with clear relationships
- **MongoDB:** Large files, flexible content metadata
- **Object Storage (S3):** Actual media files (videos, PDFs, PPTs)

Security Considerations

- Password stored as hash only
- Role-based access control via user_roles
- Audit logs track all critical actions
- IP address and user agent tracking
- Soft deletes preserve compliance history

MODULE 1: USER & TEAM MANAGEMENT

Table 1: roles

Purpose: Define system roles (Admin, Trainer, Manager, Trainee)

Column Name	Data Type	Constraints	Description
role_id	UUID	PRIMARY KEY, DEFAULT gen_random_uuid()	Unique role identifier
role_name	VARCHAR(50)	UNIQUE, NOT NULL, CHECK IN ('admin', 'trainer', 'manager', 'trainee')	Role name
description	TEXT		Role description
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Record creation timestamp
updated_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Last update timestamp

Indexes:

- PRIMARY KEY on role_id
- UNIQUE INDEX on role_name

Table 2: users

Purpose: Store all user information and authentication details

Column Name	Data Type	Constraints	Description
user_id	UUID	PRIMARY KEY, DEFAULT gen_random_uuid()	Unique user identifier
first_name	VARCHAR(100)	NOT NULL	User's first name

Column Name	Data Type	Constraints	Description
last_name	VARCHAR(100)	NOT NULL	User's last name
email	VARCHAR(255)	UNIQUE, NOT NULL	User's email address
password_hash	VARCHAR(255)	NOT NULL	Hashed password
primary_role	VARCHAR(50)	NOT NULL, CHECK IN ('admin', 'trainer', 'manager', 'trainee')	Primary role
status	VARCHAR(20)	DEFAULT 'active', CHECK IN ('active', 'deactivate', 'archived')	Account status
profile_image_url	TEXT		Profile image URL
last_login	TIMESTAMP		Last login timestamp
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Account creation date
updated_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Last update timestamp

Indexes:

- PRIMARY KEY on user_id
- UNIQUE INDEX on email
- INDEX on primary_role (idx_users_role)
- INDEX on status (idx_users_status)
- INDEX on email (idx_users_email)

Table 3: user_roles

Purpose: Many-to-many mapping between users and roles (allows multiple roles per user)

Column Name	Data Type	Constraints	Description
user_id	UUID	NOT NULL, REFERENCES users(user_id) ON DELETE CASCADE	User reference
role_id	UUID	NOT NULL, REFERENCES roles(role_id) ON DELETE CASCADE	Role reference
assigned_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Assignment timestamp
assigned_by	UUID	REFERENCES users(user_id)	Who assigned the role

Constraints:

- PRIMARY KEY (user_id, role_id)
- FOREIGN KEY user_id → users(user_id) CASCADE
- FOREIGN KEY role_id → roles(role_id) CASCADE
- FOREIGN KEY assigned_by → users(user_id)

Table 4: teams

Purpose: Define organizational teams/batches

Column Name	Data Type	Constraints	Description
team_id	UUID	PRIMARY KEY, DEFAULT gen_random_uuid()	Unique team identifier
team_name	VARCHAR(255)	UNIQUE, NOT NULL	Team name
description	TEXT		Team description

Column Name	Data Type	Constraints	Description
status	VARCHAR(20)	DEFAULT 'active', CHECK IN ('active', 'inactive', 'archived')	Team status
manager_id	UUID	REFERENCES users(user_id) ON DELETE SET NULL	Assigned manager
created_by	UUID	REFERENCES users(user_id)	Creator user
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Creation timestamp
updated_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Last update timestamp

Indexes:

- PRIMARY KEY on team_id
- UNIQUE INDEX on team_name
- INDEX on manager_id (idx_teams_manager)

Constraints:

- FOREIGN KEY manager_id → users(user_id) SET NULL
- FOREIGN KEY created_by → users(user_id)

Table 5: team_members

Purpose: Many-to-many mapping between teams and users

Column Name	Data Type	Constraints	Description
team_id	UUID	NOT NULL, REFERENCES teams(team_id) ON DELETE CASCADE	Team reference

Column Name	Data Type	Constraints	Description
user_id	UUID	NOT NULL, REFERENCES users(user_id) ON DELETE CASCADE	User reference
is_primary_team	BOOLEAN	DEFAULT true	Is this user's primary team
assigned_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Assignment timestamp
assigned_by	UUID	REFERENCES users(user_id)	Who assigned member

Constraints:

- PRIMARY KEY (team_id, user_id)
- FOREIGN KEY team_id → teams(team_id) CASCADE
- FOREIGN KEY user_id → users(user_id) CASCADE
- FOREIGN KEY assigned_by → users(user_id)

Indexes:

- INDEX on user_id (idx_team_members_user)
 - INDEX on team_id (idx_team_members_team)
-

MODULE 2: COURSES & MODULES

Table 6: courses

Purpose: Define learning courses

Column Name	Data Type	Constraints	Description
course_id	UUID	PRIMARY KEY, DEFAULT gen_random_uuid()	Unique course identifier

Column Name	Data Type	Constraints	Description
title	VARCHAR(500)	NOT NULL	Course title
description	TEXT		Course description
about	TEXT		About the course
outcomes	TEXT		Expected learning outcomes
course_type	VARCHAR(30)	DEFAULT 'self_paced', CHECK IN ('self_paced', 'instructor_led', 'blended')	Course type
status	VARCHAR(20)	DEFAULT 'draft', CHECK IN ('draft', 'published', 'archived')	Course status
is_mandatory	BOOLEAN	DEFAULT false	Is course mandatory
estimated_duration_hours	INTEGER		Estimated completion time
passing_criteria	INTEGER	DEFAULT 70, CHECK (passing_criteria >= 0 AND passing_criteria <= 100)	Minimum passing score
created_by	UUID	NOT NULL, REFERENCES users(user_id)	Course creator
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Creation timestamp
updated_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Last update timestamp

Indexes:

- PRIMARY KEY on course_id
- INDEX on created_by (idx_courses_created_by)
- INDEX on status (idx_courses_status)

Constraints:

- FOREIGN KEY created_by → users(user_id)

Table 7: course_prerequisites

Purpose: Define prerequisite relationships between courses

Column Name	Data Type	Constraints	Description
course_id	UUID	NOT NULL, REFERENCES courses(course_id) ON DELETE CASCADE	Current course
prerequisite_course_id	UUID	NOT NULL, REFERENCES courses(course_id) ON DELETE CASCADE	Required prerequisite course
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Creation timestamp

Constraints:

- PRIMARY KEY (course_id, prerequisite_course_id)
- CHECK (course_id != prerequisite_course_id) - Cannot be prerequisite of itself
- FOREIGN KEY course_id → courses(course_id) CASCADE
- FOREIGN KEY prerequisite_course_id → courses(course_id) CASCADE

Table 8: course_assignments

Purpose: Track who assigned which course to whom

Column Name	Data Type	Constraints	Description
assignment_id	UUID	PRIMARY KEY, DEFAULT gen_random_uuid()	Unique assignment identifier
course_id	UUID	NOT NULL, REFERENCES courses(course_id) ON DELETE CASCADE	Course being assigned
assigned_to_user_id	UUID	REFERENCES users(user_id) ON DELETE CASCADE	Individual user assignment
assigned_to_team_id	UUID	REFERENCES teams(team_id) ON DELETE CASCADE	Team assignment
assigned_by	UUID	NOT NULL, REFERENCES users(user_id)	Who assigned the course
due_date	TIMESTAMP		Assignment due date
assigned_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Assignment timestamp

Constraints:

- CHECK: Either assigned_to_user_id OR assigned_to_team_id must be set, not both
- FOREIGN KEY course_id → courses(course_id) CASCADE
- FOREIGN KEY assigned_to_user_id → users(user_id) CASCADE
- FOREIGN KEY assigned_to_team_id → teams(team_id) CASCADE
- FOREIGN KEY assigned_by → users(user_id)

Indexes:

- INDEX on assigned_to_user_id (idx_course_assignments_user)
- INDEX on assigned_to_team_id (idx_course_assignments_team)

Table 9: modules

Purpose: Define course modules/chapters

Column Name	Data Type	Constraints	Description
module_id	UUID	PRIMARY KEY, DEFAULT gen_random_uuid()	Unique module identifier
course_id	UUID	NOT NULL, REFERENCES courses(course_id) ON DELETE CASCADE	Parent course
title	VARCHAR(500)	NOT NULL	Module title
description	TEXT		Module description
module_type	VARCHAR(30)	CHECK IN ('video', 'pdf', 'ppt', 'document', 'quiz', 'mixed')	Content type
sequence_order	INTEGER	NOT NULL	Order in course
is_mandatory	BOOLEAN	DEFAULT true	Is module mandatory
estimated_duration_minutes	INTEGER		Estimated completion time
video_count	INTEGER	DEFAULT 0	Number of videos
has_quizzes	BOOLEAN	DEFAULT false	Contains quizzes
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Creation timestamp

Column Name	Data Type	Constraints	Description
updated_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Last update timestamp

Constraints:

- UNIQUE (course_id, sequence_order)
- FOREIGN KEY course_id → courses(course_id) CASCADE

Indexes:

- PRIMARY KEY on module_id
- INDEX on course_id (idx_modules_course)
- INDEX on (course_id, sequence_order) (idx_modules_sequence)

Table 10: module_sequencing

Purpose: Define module prerequisites and drip-feed rules

Column Name	Data Type	Constraints	Description
sequence_id	UUID	PRIMARY KEY, DEFAULT gen_random_uuid()	Unique sequence identifier
course_id	UUID	NOT NULL, REFERENCES courses(course_id) ON DELETE CASCADE	Parent course
module_id	UUID	NOT NULL, REFERENCES modules(module_id) ON DELETE CASCADE	Current module
preceding_module_id	UUID	REFERENCES modules(module_id) ON DELETE CASCADE	Previous module

Column Name	Data Type	Constraints	Description
drip_feed_rule	VARCHAR(30)	DEFAULT 'none', CHECK IN ('none', 'time_based', 'completion_based')	Release rule
drip_feed_delay_days	INTEGER	DEFAULT 0	Days to wait before release
prerequisite_completed	BOOLEAN	DEFAULT false	Must complete previous
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Creation timestamp
updated_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Last update timestamp

Constraints:

- UNIQUE (course_id, module_id)
- FOREIGN KEY course_id → courses(course_id) CASCADE
- FOREIGN KEY module_id → modules(module_id) CASCADE
- FOREIGN KEY preceding_module_id → modules(module_id) CASCADE

Table 11: module_completions

Purpose: Track module-level completion progress

Column Name	Data Type	Constraints	Description
completion_id	UUID	PRIMARY KEY, DEFAULT gen_random_uuid()	Unique completion identifier

Column Name	Data Type	Constraints	Description
module_id	UUID	NOT NULL, REFERENCES modules(module_id) ON DELETE CASCADE	Module reference
user_id	UUID	NOT NULL, REFERENCES users(user_id) ON DELETE CASCADE	User reference
completion_percentage	INTEGER	DEFAULT 0, CHECK (completion_percentage >= 0 AND completion_percentage <= 100)	Progress percentage
is_completed	BOOLEAN	DEFAULT false	Completion status
time_spent_minutes	INTEGER	DEFAULT 0	Time spent on module
completed_at	TIMESTAMP		Completion timestamp
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Record creation
updated_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Last update

Constraints:

- UNIQUE (module_id, user_id)
- FOREIGN KEY module_id → modules(module_id) CASCADE
- FOREIGN KEY user_id → users(user_id) CASCADE

Indexes:

- INDEX on user_id (idx_module_completions_user)

Table 12: notes

Purpose: Store trainee notes for modules

Column Name	Data Type	Constraints	Description
note_id	UUID	PRIMARY KEY, DEFAULT gen_random_uuid()	Unique note identifier
user_id	UUID	NOT NULL, REFERENCES users(user_id) ON DELETE CASCADE	Note owner
module_id	UUID	NOT NULL, REFERENCES modules(module_id) ON DELETE CASCADE	Related module
content	TEXT	NOT NULL	Note content
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Creation timestamp
updated_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Last update timestamp

Constraints:

- FOREIGN KEY user_id → users(user_id) CASCADE
- FOREIGN KEY module_id → modules(module_id) CASCADE

MODULE 3: ASSIGNMENTS & TESTS

Table 13: assignments

Purpose: Define course/module assignments

Column Name	Data Type	Constraints	Description
assignment_id	UUID	PRIMARY KEY, DEFAULT gen_random_uuid()	Unique assignment identifier
course_id	UUID	REFERENCES courses(course_id) ON DELETE CASCADE	Parent course
module_id	UUID	REFERENCES modules(module_id) ON DELETE CASCADE	Parent module
title	VARCHAR(500)	NOT NULL	Assignment title
description	TEXT		Assignment description
assignment_type	VARCHAR(30)	CHECK IN ('task', 'role_play', 'written', 'project', 'other')	Type of assignment
due_date	TIMESTAMP		Due date
max_attempts	INTEGER	DEFAULT 1, CHECK (max_attempts > 0)	Maximum attempts allowed
points_possible	INTEGER	DEFAULT 100	Maximum points
is_mandatory	BOOLEAN	DEFAULT true	Is assignment mandatory
created_by	UUID	NOT NULL, REFERENCES users(user_id)	Assignment creator
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Creation timestamp
updated_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Last update timestamp

Constraints:

- FOREIGN KEY course_id → courses(course_id) CASCADE
- FOREIGN KEY module_id → modules(module_id) CASCADE
- FOREIGN KEY created_by → users(user_id)

Indexes:

- INDEX on course_id (idx_assignments_course)

Table 14: assignment_targeting

Purpose: Define who is assigned which assignment

Column Name	Data Type	Constraints	Description
assignment_target_id	UUID	PRIMARY KEY, DEFAULT gen_random_uuid()	Unique target identifier
assignment_id	UUID	NOT NULL, REFERENCES assignments(assignment_id) ON DELETE CASCADE	Assignment reference
assigned_to_user_id	UUID	REFERENCES users(user_id) ON DELETE CASCADE	Individual user assignment
assigned_to_team_id	UUID	REFERENCES teams(team_id) ON DELETE CASCADE	Team assignment
assigned_by	UUID	NOT NULL, REFERENCES users(user_id)	Who assigned
assigned_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Assignment timestamp

Constraints:

- CHECK: Either assigned_to_user_id OR assigned_to_team_id must be set, not both
- FOREIGN KEY assignment_id → assignments(assignment_id) CASCADE
- FOREIGN KEY assigned_to_user_id → users(user_id) CASCADE
- FOREIGN KEY assigned_to_team_id → teams(team_id) CASCADE
- FOREIGN KEY assigned_by → users(user_id)

Table 15: assignment_submissions

Purpose: Store assignment submissions from trainees

Column Name	Data Type	Constraints	Description
submission_id	UUID	PRIMARY KEY, DEFAULT gen_random_uuid()	Unique submission identifier
assignment_id	UUID	NOT NULL, REFERENCES assignments(assignment_id) ON DELETE CASCADE	Assignment reference
user_id	UUID	NOT NULL, REFERENCES users(user_id) ON DELETE CASCADE	Submitter
attempt_number	INTEGER	NOT NULL DEFAULT 1	Attempt number
submission_text	TEXT		Text submission
submission_files	JSONB		File references array
submitted_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Submission timestamp
status	VARCHAR(30)	DEFAULT 'submitted', CHECK IN ('draft', 'submitted', 'graded', 'returned')	Submission status
score	INTEGER	CHECK (score >= 0)	Points scored
points_earned	INTEGER	DEFAULT 0	Points earned
feedback	TEXT		Trainer feedback
graded_by	UUID	REFERENCES users(user_id)	Who graded
graded_at	TIMESTAMP		Grading timestamp

Column Name	Data Type	Constraints	Description
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Creation timestamp
updated_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Last update timestamp

Constraints:

- FOREIGN KEY assignment_id → assignments(assignment_id) CASCADE
- FOREIGN KEY user_id → users(user_id) CASCADE
- FOREIGN KEY graded_by → users(user_id)

Indexes:

- INDEX on user_id (idx_assignment_submissions_user)
- INDEX on assignment_id (idx_assignment_submissions_assignment)

Table 16: test_bank

Purpose: Centralized question bank for randomization

Column Name	Data Type	Constraints	Description
test_bank_id	UUID	PRIMARY KEY, DEFAULT gen_random_uuid()	Unique bank identifier
name	VARCHAR(255)	NOT NULL	Bank name
description	TEXT		Bank description
category	VARCHAR(100)		Question category
created_by	UUID	NOT NULL, REFERENCES users(user_id)	Bank creator

Column Name	Data Type	Constraints	Description
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Creation timestamp
updated_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Last update timestamp

Constraints:

- FOREIGN KEY created_by → users(user_id)

Table 17: tests

Purpose: Define tests/quizzes/exams

Column Name	Data Type	Constraints	Description
test_id	UUID	PRIMARY KEY, DEFAULT gen_random_uuid()	Unique test identifier
course_id	UUID	REFERENCES courses(course_id) ON DELETE CASCADE	Parent course
module_id	UUID	REFERENCES modules(module_id) ON DELETE CASCADE	Parent module
test_bank_id	UUID	REFERENCES test_bank(test_bank_id)	Source question bank
title	VARCHAR(500)	NOT NULL	Test title
description	TEXT		Test description
test_type	VARCHAR(30)	CHECK IN ('quiz', 'test', 'exam', 'assessment')	Test type

Column Name	Data Type	Constraints	Description
time_limit_minutes	INTEGER		Time limit
passing_score	INTEGER	DEFAULT 70, CHECK (passing_score >= 0 AND passing_score <= 100)	Minimum passing score
max_attempts	INTEGER	DEFAULT 1, CHECK (max_attempts > 0)	Maximum attempts
randomize_questions	BOOLEAN	DEFAULT false	Randomize question order
show_correct_answers	BOOLEAN	DEFAULT false	Show answers after submission
points_possible	INTEGER	DEFAULT 100	Maximum points
is_mandatory	BOOLEAN	DEFAULT true	Is test mandatory
created_by	UUID	NOT NULL, REFERENCES users(user_id)	Test creator
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Creation timestamp
updated_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Last update timestamp

Constraints:

- FOREIGN KEY course_id → courses(course_id) CASCADE
- FOREIGN KEY module_id → modules(module_id) CASCADE
- FOREIGN KEY test_bank_id → test_bank(test_bank_id)
- FOREIGN KEY created_by → users(user_id)

Indexes:

- INDEX on course_id (idx_tests_course)

Table 18: test_questions

Purpose: Store test questions

Column Name	Data Type	Constraints	Description
question_id	UUID	PRIMARY KEY, DEFAULT gen_random_uuid()	Unique question identifier
test_id	UUID	REFERENCES tests(test_id) ON DELETE CASCADE	Parent test
test_bank_id	UUID	REFERENCES test_bank(test_bank_id)	Source bank
question_text	TEXT	NOT NULL	Question text
question_type	VARCHAR(30)	NOT NULL, CHECK IN ('mcq', 'true_false', 'short_answer', 'essay', 'fill_blank')	Question type
options	JSONB		MCQ options with correct flag
correct_answer	TEXT		Correct answer
points	INTEGER	DEFAULT 1	Points for question
difficulty	VARCHAR(20)	CHECK IN ('easy', 'medium', 'hard')	Difficulty level
explanation	TEXT		Answer explanation
sequence_order	INTEGER		Question order
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Creation timestamp
updated_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Last update timestamp

Constraints:

- FOREIGN KEY test_id → tests(test_id) CASCADE
- FOREIGN KEY test_bank_id → test_bank(test_bank_id)

Table 19: test_attempts

Purpose: Track test attempts by users

Column Name	Data Type	Constraints	Description
attempt_id	UUID	PRIMARY KEY, DEFAULT gen_random_uuid()	Unique attempt identifier
test_id	UUID	NOT NULL, REFERENCES tests(test_id) ON DELETE CASCADE	Test reference
user_id	UUID	NOT NULL, REFERENCES users(user_id) ON DELETE CASCADE	Test taker
attempt_number	INTEGER	NOT NULL	Attempt number
started_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Start time
submitted_at	TIMESTAMP		Submission time
time_spent_minutes	INTEGER		Time spent
status	VARCHAR(30)	DEFAULT 'in_progress', CHECK IN ('in_progress', 'completed', 'abandoned', 'timed_out')	Attempt status
score	INTEGER	CHECK (score >= 0 AND score <= 100)	Score percentage
points_earned	INTEGER	DEFAULT 0	Points earned
passed	BOOLEAN		Pass/fail status
graded_by	UUID	REFERENCES users(user_id)	Who graded

Column Name	Data Type	Constraints	Description
graded_at	TIMESTAMP		Grading timestamp
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Creation timestamp
updated_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Last update timestamp

Constraints:

- UNIQUE (test_id, user_id, attempt_number)
- FOREIGN KEY test_id → tests(test_id) CASCADE
- FOREIGN KEY user_id → users(user_id) CASCADE
- FOREIGN KEY graded_by → users(user_id)

Indexes:

- INDEX on user_id (idx_test_attempts_user)
- INDEX on test_id (idx_test_attempts_test)

Table 20: test_answers

Purpose: Store individual question answers

Column Name	Data Type	Constraints	Description
answer_id	UUID	PRIMARY KEY, DEFAULT gen_random_uuid()	Unique answer identifier
attempt_id	UUID	NOT NULL, REFERENCES test_attempts(attempt_id) ON DELETE CASCADE	Attempt reference

Column Name	Data Type	Constraints	Description
question_id	UUID	NOT NULL, REFERENCES test_questions(question_id) ON DELETE CASCADE	Question reference
answer_text	TEXT		Text answer
selected_options	JSONB		Selected MCQ options
is_correct	BOOLEAN		Correct/incorrect flag
points_earned	INTEGER	DEFAULT 0	Points earned
feedback	TEXT		Question feedback
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Creation timestamp
updated_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Last update timestamp

Constraints:

- UNIQUE (attempt_id, question_id)
 - FOREIGN KEY attempt_id → test_attempts(attempt_id) CASCADE
 - FOREIGN KEY question_id → test_questions(question_id) CASCADE
-

MODULE 4: PROGRESS & GAMIFICATION

Table 21: user_progress

Purpose: Track overall course progress for users

Column Name	Data Type	Constraints	Description
progress_id	UUID	PRIMARY KEY, DEFAULT gen_random_uuid()	Unique progress identifier
user_id	UUID	NOT NULL, REFERENCES users(user_id) ON DELETE CASCADE	User reference
course_id	UUID	NOT NULL, REFERENCES courses(course_id) ON DELETE CASCADE	Course reference
completion_percentage	INTEGER	DEFAULT 0, CHECK (completion_percentage >= 0 AND completion_percentage <= 100)	Overall progress
total_points_earned	INTEGER	DEFAULT 0	Total points
average_score	INTEGER	DEFAULT 0	Average score
time_spent_minutes	INTEGER	DEFAULT 0	Total time spent
modules_completed	INTEGER	DEFAULT 0	Modules completed count
total_modules	INTEGER	DEFAULT 0	Total modules count
tests_passed	INTEGER	DEFAULT 0	Tests passed count
tests_attempted	INTEGER	DEFAULT 0	Tests attempted count
assignments_submitted	INTEGER	DEFAULT 0	Assignments submitted

Column Name	Data Type	Constraints	Description
assignments_graded	INTEGER	DEFAULT 0	Assignments graded
started_at	TIMESTAMP		Course start date
completed_at	TIMESTAMP		Course completion date
last_activity	TIMESTAMP		Last activity timestamp
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Record creation
updated_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Last update

Constraints:

- UNIQUE (user_id, course_id)
- FOREIGN KEY user_id → users(user_id) CASCADE
- FOREIGN KEY course_id → courses(course_id) CASCADE

Indexes:

- INDEX on user_id (idx_user_progress_user)
- INDEX on course_id (idx_user_progress_course)

Table 22: badge_rules

Purpose: Define rules for earning badges

Column Name	Data Type	Constraints	Description
rule_id	UUID	PRIMARY KEY, DEFAULT gen_random_uuid()	Unique rule identifier
rule_name	VARCHAR(255)	NOT NULL	Rule name
rule_type	VARCHAR(50)	CHECK IN ('points_threshold', 'completion', 'score', 'streak', 'deadline', 'custom')	Rule type
description	TEXT		Rule description
criteria	JSONB	NOT NULL	Rule criteria JSON
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Creation timestamp
updated_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Last update timestamp

Example criteria JSON:

```
{
  "threshold": 1000,
  "type": "points"
}
```

or

```
{
  "modules_completed": 10,
  "within_days": 30
}
```

Table 23: badges

Purpose: Define available badges

Column Name	Data Type	Constraints	Description
badge_id	UUID	PRIMARY KEY, DEFAULT gen_random_uuid()	Unique badge identifier
badge_name	VARCHAR(255)	NOT NULL	Badge name
description	TEXT		Badge description
badge_type	VARCHAR(30)	CHECK IN ('gold', 'silver', 'bronze', 'positive', 'negative', 'custom')	Badge type
badge_icon_url	TEXT		Badge icon URL
rule_id	UUID	REFERENCES badge_rules(rule_id)	Associated rule
points_threshold	INTEGER	DEFAULT 0	Points required
visibility	VARCHAR(20)	DEFAULT 'public', CHECK IN ('public', 'private')	Visibility level
is_active	BOOLEAN	DEFAULT true	Is badge active
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Creation timestamp
updated_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Last update timestamp

Constraints:

- FOREIGN KEY rule_id → badge_rules(rule_id)

Table 24: badge_assignments

Purpose: Track badge awards to users

Column Name	Data Type	Constraints	Description
badge_assignment_id	UUID	PRIMARY KEY, DEFAULT gen_random_uuid()	Unique assignment identifier
badge_id	UUID	NOT NULL, REFERENCES badges(badge_id) ON DELETE CASCADE	Badge reference
user_id	UUID	NOT NULL, REFERENCES users(user_id) ON DELETE CASCADE	User reference
course_id	UUID	REFERENCES courses(course_id)	Related course
assigned_by	UUID	REFERENCES users(user_id)	Manual assignment by
reason	TEXT		Award reason
earned_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Award timestamp
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Record creation

Constraints:

- FOREIGN KEY badge_id → badges(badge_id) CASCADE
- FOREIGN KEY user_id → users(user_id) CASCADE
- FOREIGN KEY course_id → courses(course_id)
- FOREIGN KEY assigned_by → users(user_id)

Indexes:

- INDEX on user_id (idx_badge_assignments_user)
- INDEX on badge_id (idx_badge_assignments_badge)

Table 25: leaderboard

Purpose: Track user rankings and points

Column Name	Data Type	Constraints	Description
leaderboard_id	UUID	PRIMARY KEY, DEFAULT gen_random_uuid()	Unique leaderboard entry
scope	VARCHAR(30)	NOT NULL, CHECK IN ('global', 'team', 'course', 'batch', 'module')	Leaderboard scope
scope_id	UUID		Related entity ID (team/course)
user_id	UUID	NOT NULL, REFERENCES users(user_id) ON DELETE CASCADE	User reference
points	INTEGER	DEFAULT 0	Total points
rank	INTEGER		User rank
calculated_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Last calculation time
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Record creation
updated_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Last update

Constraints:

- FOREIGN KEY user_id → users(user_id) CASCADE

Indexes:

- INDEX on (scope, scope_id) (idx_leaderboard_scope)
- INDEX on user_id (idx_leaderboard_user)
- INDEX on (scope, rank) (idx_leaderboard_rank)

MODULE 5: NOTIFICATIONS & AUDIT

Table 26: notifications

Purpose: Store user notifications

Column Name	Data Type	Constraints	Description
notification_id	UUID	PRIMARY KEY, DEFAULT gen_random_uuid()	Unique notification identifier
user_id	UUID	NOT NULL, REFERENCES users(user_id) ON DELETE CASCADE	Recipient user
notification_type	VARCHAR(50)	CHECK IN ('assignment', 'test', 'badge', 'deadline', 'course', 'grade', 'system', 'reminder')	Notification type
title	VARCHAR(500)		Notification title
message	TEXT	NOT NULL	Notification message
link_url	TEXT		Deep link URL
priority	VARCHAR(20)	DEFAULT 'normal', CHECK IN ('low', 'normal', 'high', 'urgent')	Priority level
status	VARCHAR(20)	DEFAULT 'unread', CHECK IN ('unread', 'read', 'archived')	Read status
sent_via	VARCHAR(30)	DEFAULT 'in_app', CHECK IN ('in_app', 'email', 'both')	Delivery method
read_at	TIMESTAMP		Read timestamp
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Creation timestamp

Constraints:

- FOREIGN KEY user_id → users(user_id) CASCADE

Indexes:

- INDEX on user_id (idx_notifications_user)
- INDEX on status (idx_notifications_status)
- INDEX on created_at DESC (idx_notifications_created)

Table 27: audit_logs

Purpose: Track all system actions for compliance

Column Name	Data Type	Constraints	Description
log_id	UUID	PRIMARY KEY, DEFAULT gen_random_uuid()	Unique log identifier
user_id	UUID	REFERENCES users(user_id)	User who performed action
action_type	VARCHAR(100)	NOT NULL	Action type (CREATE, UPDATE, DELETE, VIEW)
entity_type	VARCHAR(100)		Entity type (user, course, assignment)
entity_id	UUID		Entity identifier
details	JSONB		Additional action details
ip_address	INET		User IP address
user_agent	TEXT		Browser user agent
timestamp	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Action timestamp

Constraints:

- FOREIGN KEY user_id → users(user_id)

Indexes:

- INDEX on user_id (idx_audit_logs_user)
- INDEX on timestamp DESC (idx_audit_logs_timestamp)
- INDEX on (entity_type, entity_id) (idx_audit_logs_entity)

Example details JSON:

```
{  
  "action": "user_role_changed",  
  "old_value": "trainee",  
  "new_value": "trainer",  
  "changed_by": "admin_user_id"  
}
```

MODULE 6: FEEDBACK & COMMUNICATION

Table 28: feedback

Purpose: Store user feedback and reviews

Column Name	Data Type	Constraints	Description
feedback_id	UUID	PRIMARY KEY, DEFAULT gen_random_uuid()	Unique feedback identifier
user_id	UUID	NOT NULL, REFERENCES users(user_id) ON DELETE CASCADE	Feedback author
course_id	UUID	REFERENCES courses(course_id)	Related course
module_id	UUID	REFERENCES modules(module_id)	Related module

Column Name	Data Type	Constraints	Description
feedback_type	VARCHAR(30)	CHECK IN ('course', 'module', 'trainer', 'system', 'general')	Feedback type
rating	INTEGER	CHECK (rating >= 1 AND rating <= 5)	Star rating
content	TEXT	NOT NULL	Feedback content
is_anonymous	BOOLEAN	DEFAULT false	Anonymous feedback
status	VARCHAR(20)	DEFAULT 'pending', CHECK IN ('pending', 'reviewed', 'resolved')	Review status
reviewed_by	UUID	REFERENCES users(user_id)	Who reviewed
reviewed_at	TIMESTAMP		Review timestamp
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Creation timestamp
updated_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Last update timestamp

Constraints:

- FOREIGN KEY user_id → users(user_id) CASCADE
- FOREIGN KEY course_id → courses(course_id)
- FOREIGN KEY module_id → modules(module_id)
- FOREIGN KEY reviewed_by → users(user_id)

MONGODB COLLECTIONS

Collection 1: module_content_items

Purpose: Store module content metadata

Document Structure:

```
{  
  "_id": "ObjectId",  
  "module_id": "UUID (reference to PostgreSQL modules table)",  
  "content_type": "String (video, pdf, ppt, document, link)",  
  "title": "String",  
  "description": "String",  
  "file_reference": "String (S3 URL or file path)",  
  "file_size_bytes": "Number",  
  "duration_seconds": "Number",  
  "thumbnail_url": "String",  
  "sequence_order": "Number",  
  "metadata": {  
    "format": "String (mp4, pdf, pptx)",  
    "resolution": "String (1080p, 720p)",  
    "mime_type": "String (video/mp4, application/pdf)"  
  },  
  "created_at": "Date",  
  "updated_at": "Date"  
}
```

Indexes:

- Index on module_id
- Index on sequence_order
- Compound index on (module_id, sequence_order)

Collection 2: media_files

Purpose: Store media file metadata and processing status

Document Structure:

```
{  
  "_id": "ObjectId",  
  "file_type": "String (video, audio, pdf, ppt, image)",  
  "title": "String",  
  "file_path": "String (S3 bucket path)",  
  "file_size_bytes": "Number",  
  "duration_seconds": "Number",  
  "thumbnail_path": "String",  
  "upload_metadata": {  
    "uploaded_by": "UUID (reference to PostgreSQL users)",  
    "upload_date": "Date",  
    "original_filename": "String"  
  },  
  "encoding_status": "String (pending, processing, completed, failed)",  
  "created_at": "Date",  
  "updated_at": "Date"  
}
```

Indexes:

- Index on file_type
- Index on encoding_status
- Index on upload_metadata.uploaded_by

Collection 3: test_question_media

Purpose: Store media attachments for test questions

Document Structure:

```
{  
  "_id": "ObjectId",  
  "question_id": "UUID (reference to PostgreSQL test_questions)",  
  "media_type": "String (image, video, audio)",  
  "file_reference": "String (S3 URL or file path)",  
  "file_size_bytes": "Number",  
  "metadata": {  
    "format": "String",  
    "dimensions": "String (1920x1080)",  
    "duration_seconds": "Number (for video/audio)"  
  },  
  "created_at": "Date"  
}
```

Indexes:

- Index on question_id
- Index on media_type