

Base URL: <http://localhost:3000/api>

1. Create User (POST /users) ? body: authUid, email, name, profile
2. Get User (GET /users/:authUid) ? returns user by authUid
3. Update User Profile (PUT /users/:authUid) ? body: name, profile
4. Update User Streak (PUT /users/:authUid/streak) ? no body
5. Add Achievement to User (POST /users/:authUid/achievements) ? body: achievementKey
6. Delete User (DELETE /users/:authUid) ? cascades related data

7. Create Assessment Template (POST /assessment-templates) ? body: title, scale, items
8. Get All Templates (GET /assessment-templates)
9. Get Template by ID (GET /assessment-templates/:id)
10. Update Template (PUT /assessment-templates/:id)
11. Delete Template (DELETE /assessment-templates/:id)

12. Submit Assessment Response (POST /assessment-responses) ? body: userId, templateId, responses, derived, notes
13. Get User Assessment Responses (GET /assessment-responses/user/:userId)
14. Get Latest Assessment Response (GET /assessment-responses/user/:userId/latest)

15. Create Scenario (POST /scenarios) ? body: title, description, levels, status
16. Get All Scenarios (GET /scenarios?status=optional)
17. Get Published Scenarios (GET /scenarios/published)
18. Get Scenario by ID (GET /scenarios/:id)
19. Update Scenario (PUT /scenarios/:id)
20. Delete Scenario (DELETE /scenarios/:id)

21. Create Level Prompt (POST /level-prompts) ? body: scenarioId, level, introScript, questionSet, rubric, ai prompts
22. Get Level Prompt (GET /level-prompts/scenario/:scenarioId/level/:level)
23. Get All Level Prompts (GET /level-prompts/scenario/:scenarioId)
24. Update Level Prompt (PUT /level-prompts/:id)
25. Delete Level Prompt (DELETE /level-prompts/:id)

26. Create Media Job (POST /media-jobs) ? body: scenarioId, level, questionOrder, provider, jobId, source
27. Get Media Job by ID (GET /media-jobs/:id)
28. Get Media Job by Job ID (GET /media-jobs/job/:jobId)
29. Update Media Job Status (PUT /media-jobs/job/:jobId/status)
30. Get Media Jobs by Status (GET /media-jobs/status/:status)
31. Get Media Jobs for Scenario (GET /media-jobs/scenario/:scenarioId)

32. Start Practice Session (POST /practice-sessions) ? body: userId, scenarioId, level
33. Add Step to Session (POST /practice-sessions/:sessionId/steps) ? body: step details
34. Complete Session (PUT /practice-sessions/:sessionId/complete) ? body: aggregate, feedback, achievementsUnlocked
35. Abandon Session (PUT /practice-sessions/:sessionId/abandon)
36. Get User Sessions (GET /practice-sessions/user/:userId?status&limit)
37. Get Session by ID (GET /practice-sessions/:sessionId)
38. Delete Session (DELETE /practice-sessions/:sessionId)

39. Initialize Progress (POST /progress) ? body: userId, scenarioId
40. Get User Progress (GET /progress/user/:userId)

- 41. Get Progress for Scenario (GET /progress/user/:userId/scenario/:scenarioId)
- 42. Unlock Level (PUT /progress/user/:userId/scenario/:scenarioId/unlock) ? body: level

- 43. Create Encouragement Note (POST /encouragement-notes) ? body: userId, date, title, body, tags, linkedSessionId
- 44. Get User Notes (GET /encouragement-notes/user/:userId?tags&date)
- 45. Get Note by ID (GET /encouragement-notes/:id)
- 46. Update Note (PUT /encouragement-notes/:id)
- 47. Delete Note (DELETE /encouragement-notes/:id)

- 48. Create Achievement (POST /achievements) ? body: key, title, description, icon, category
- 49. Get All Achievements (GET /achievements?category=optional)
- 50. Get Achievement by ID (GET /achievements/:id)
- 51. Get Achievement by Key (GET /achievements/key/:key)
- 52. Update Achievement (PUT /achievements/:id)
- 53. Delete Achievement (DELETE /achievements/:id)

All endpoints expect 'Content-Type: application/json'. Use 'npm run dev' to start the server before testing. Tools like Postman or