Vitest Test Report

Membership Payment System - Backend
Generated on 17 February 2026 at 3:17:39 pm

200	0	200	12	1.57s
Passed	Failed	Total	Test Files	Duration

Test Suites

✓ admin.test.js   18/18 passed • 159ms

POST /api/admin/login

✓ POST /api/admin/login should return token for valid credentials	49.17ms
✓ POST /api/admin/login should return 401 for invalid credentials	6.53ms
✓ POST /api/admin/login should return 401 for empty credentials	4.38ms

GET /api/admin/verify-token

✓ GET /api/admin/verify-token should return 401 without token	5.16ms
✓ GET /api/admin/verify-token should return 401 with invalid token	4.59ms
✓ GET /api/admin/verify-token should return success with valid token	9ms

GET /api/admin/members

✓ GET /api/admin/members should return 401 without auth header	4.5ms
✓ GET /api/admin/members should return paginated member list	7.01ms
✓ GET /api/admin/members should support search parameter	6.52ms
✓ GET /api/admin/members should support status filter	5.29ms

POST /api/admin/members

✓ POST /api/admin/members should return 400 for missing required fields	5.47ms
✓ POST /api/admin/members should return 400 for duplicate folio/email	10.43ms
✓ POST /api/admin/members should create member successfully	6.63ms
✓ POST /api/admin/members should calculate starting_period correctly for Jan-Mar join	5.12ms

PUT /api/admin/members/:id

✓ PUT /api/admin/members/:id should update member fields	6.61ms
✓ PUT /api/admin/members/:id should return 400 when no fields to update	10.96ms
✓ PUT /api/admin/members/:id should handle payment data updates	6.03ms

GET /api/admin/stats

✓ GET /api/admin/stats should return dashboard statistics	5.45ms

✓ health.test.js   3/3 passed • 41ms

Health & Root Endpoints

✓ Health & Root Endpoints GET / should return API info	30.12ms
✓ Health & Root Endpoints GET /health should return healthy status	5.06ms
✓ Health & Root Endpoints GET /nonexistent should return 404	5.08ms

✔ members.test.js · 17/17 passed · 170ms

POST /api/members/search
✔ POST /api/members/search should return matching members · 43.58ms
✔ POST /api/members/search should return empty array for empty name · 11.74ms
✔ POST /api/members/search should return empty array for missing name · 5.27ms
✔ POST /api/members/search should return 500 on database error · 6.58ms

GET /api/members/:id
✔ GET /api/members/:id should return member when found · 6.08ms
✔ GET /api/members/:id should return 404 when member not found · 4.1ms

GET /api/members/:id/payment-calculation
✔ GET /api/members/:id/payment-calculation should return payment calculation for member · 5.61ms
✔ GET /api/members/:id/payment-calculation should return 404 for non-existent member · 10.88ms
✔ GET /api/members/:id/payment-calculation should skip exempt periods (amount === 0) · 4.68ms

POST /api/members/send-otp
✔ POST /api/members/send-otp should send OTP and return masked email · 5.76ms
✔ POST /api/members/send-otp should return 400 when memberId is missing · 5.12ms
✔ POST /api/members/send-otp should return 404 for non-existent member · 4.64ms
✔ POST /api/members/send-otp should return 400 when member has no email · 4.45ms

POST /api/members/verify-otp
✔ POST /api/members/verify-otp should verify OTP successfully · 4.68ms
✔ POST /api/members/verify-otp should return 400 for invalid OTP · 31.05ms
✔ POST /api/members/verify-otp should return 400 when memberId or otp missing · 4.74ms
✔ POST /api/members/verify-otp should return 404 for non-existent member · 10.32ms

✔ payments.test.js · 13/13 passed · 183ms

GET /api/payments/history/:memberId
✔ GET /api/payments/history/:memberId should return payment history for all periods · 38.87ms
✔ GET /api/payments/history/:memberId should return 404 for non-existent member · 16.09ms

POST /api/payments/calculate
✔ POST /api/payments/calculate should return payment calculation with options · 26.54ms
✔ POST /api/payments/calculate should return 400 when memberId is missing · 4.84ms
✔ POST /api/payments/calculate should return 404 for non-existent member · 42.07ms
✔ POST /api/payments/calculate should return canPay false when all periods are paid · 5.63ms

POST /api/payments/initiate
✔ POST /api/payments/initiate should create Razorpay order on success · 5.05ms
✔ POST /api/payments/initiate should return 400 when required fields missing · 4.38ms
✔ POST /api/payments/initiate should return 404 for non-existent member · 4.78ms
✔ POST /api/payments/initiate should return 500 when Razorpay order creation fails · 4.8ms

POST /api/payments/verify
✔ POST /api/payments/verify should verify valid payment and update database · 20.21ms
✔ POST /api/payments/verify should return 400 for invalid signature · 4.46ms
✔ POST /api/payments/verify should return 400 when signature data is missing · 4.7ms

✔ api.security.test.js   20/20 passed • 185ms

Error Information Disclosure

✔ Error Information Disclosure should not expose SQL error details in member search error   40.58ms

✔ Error Information Disclosure should not expose stack traces in error responses   6.72ms

✔ Error Information Disclosure should not expose database credentials in error response   9.63ms

✔ Error Information Disclosure should not expose JWT secret in any error response   14.42ms

✔ Error Information Disclosure should return proper JSON error format on 404   4.45ms

HTTP Method Security

✔ HTTP Method Security should reject PUT on member search endpoint   5.42ms

✔ HTTP Method Security should reject DELETE on admin members endpoint   6.59ms

✔ HTTP Method Security should reject PATCH on admin members endpoint   6.88ms

✔ HTTP Method Security should handle OPTIONS request (CORS preflight)   6.17ms

Content-Type & Malformed Input

✔ Content-Type & Malformed Input should handle request with text/plain content-type   4.34ms

✔ Content-Type & Malformed Input should handle malformed JSON body gracefully   5.64ms

✔ Content-Type & Malformed Input should handle empty body on POST endpoint   4.55ms

✔ Content-Type & Malformed Input should handle content-type mismatch (form data sent as JSON)   6.9ms

Response Headers Security

✔ Response Headers Security should check X-Powered-By header (information disclosure)   4.43ms

✔ Response Headers Security should return proper Content-Type for JSON responses   4.11ms

✔ Response Headers Security should have CORS headers on responses   4.43ms

✔ Response Headers Security should not have CORS headers for unknown origin   4.94ms

Large Payload Protection

✔ Large Payload Protection should reject extremely large JSON payload   23.12ms

✔ Large Payload Protection should handle deeply nested JSON object   5.27ms

✔ Large Payload Protection should handle array with many elements   14.5ms

✔ **auth.security.test.js**   26/26 passed • 184ms

JWT Token Manipulation

✔ JWT Token Manipulation should reject expired JWT tokens   42.84ms
✔ JWT Token Manipulation should reject token signed with wrong secret   7.35ms
✔ JWT Token Manipulation should reject malformed JWT string   4.71ms
✔ JWT Token Manipulation should reject empty Bearer token   5.93ms
✔ JWT Token Manipulation should reject token without Bearer prefix   5.1ms
✔ JWT Token Manipulation should reject null authorization header   3.85ms
✔ JWT Token Manipulation should reject token with tampered payload   6.93ms
✔ JWT Token Manipulation should reject token with "none" algorithm (alg:none attack)   4.66ms
✔ JWT Token Manipulation should reject token with only header and payload (no signature)   4.44ms

Admin Login Security

✔ Admin Login Security should reject SQL injection in username   16.79ms
✔ Admin Login Security should reject SQL injection with UNION in username   4.71ms
✔ Admin Login Security should handle extremely long username without crashing   4.29ms
✔ Admin Login Security should handle extremely long password without crashing   4.83ms
✔ Admin Login Security should return same error message for wrong username vs wrong password   10.04ms
✔ Admin Login Security should not return token in error response   4.64ms
✔ Admin Login Security should not return password or secret in any response   4.2ms
✔ Admin Login Security should reject login with empty strings   4.13ms
✔ Admin Login Security should reject login with null values   4.14ms
✔ Admin Login Security should reject login with numeric values   4.19ms

Authorization Bypass

✔ Authorization Bypass should return 401 for GET /api/admin/members without token   4.16ms
✔ Authorization Bypass should return 401 for POST /api/admin/members without token   4.03ms
✔ Authorization Bypass should return 401 for PUT /api/admin/members/1 without token   3.69ms
✔ Authorization Bypass should return 401 for GET /api/admin/stats without token   3.73ms
✔ Authorization Bypass should reject token with non-admin role   8ms
✔ Authorization Bypass should reject request with Authorization header but no Bearer scheme   4.84ms
✔ Authorization Bypass should handle request with multiple Bearer tokens   6.07ms

✔ **injection.security.test.js**   17/17 passed • 189ms

SQL Injection - Member Search

✔ SQL Injection - Member Search should safely handle DROP TABLE injection in name   41.58ms

✔ SQL Injection - Member Search should safely handle OR 1=1 injection   12.31ms

✔ SQL Injection - Member Search should safely handle UNION SELECT injection   27.99ms

✔ SQL Injection - Member Search should safely handle DELETE injection   5.96ms

✔ SQL Injection - Member Search should safely handle comment-based injection   4.48ms

SQL Injection - Admin Routes

✔ SQL Injection - Admin Routes should safely handle SQL injection in search query param   13.26ms

✔ SQL Injection - Admin Routes should safely handle SQL injection in status query param   7.8ms

✔ SQL Injection - Admin Routes should handle non-numeric member ID gracefully   5.68ms

✔ SQL Injection - Admin Routes should handle SQL injection in member ID path param   4.43ms

XSS Payload Testing

✔ XSS Payload Testing should store XSS payload in name as plain text (not execute)   7.65ms

✔ XSS Payload Testing should handle img onerror XSS in email field   7.86ms

✔ XSS Payload Testing should handle XSS payload in search name   5.01ms

✔ XSS Payload Testing should handle HTML entities in member name   5.27ms

Dynamic Column Injection - Payment Verify

✔ Dynamic Column Injection - Payment Verify should neutralize SQL injection in period values via parseInt   22.98ms

✔ Dynamic Column Injection - Payment Verify should handle non-numeric period values gracefully   6.2ms

✔ Dynamic Column Injection - Payment Verify should handle negative period numbers   5.18ms

✔ Dynamic Column Injection - Payment Verify should handle extremely large period numbers   4.67ms

✔ input-validation.security.test.js

28/28 passed • 267ms

Boundary Testing - Member Search

✔ Boundary Testing - Member Search should return empty array for empty string name	65.75ms
✔ Boundary Testing - Member Search should handle single character name	6.29ms
✔ Boundary Testing - Member Search should handle name with 1000+ characters without crashing	5.4ms
✔ Boundary Testing - Member Search should handle name with only special characters	4.42ms
✔ Boundary Testing - Member Search should handle unicode and emoji in name	4.78ms
✔ Boundary Testing - Member Search should handle null byte in name	18.11ms
✔ Boundary Testing - Member Search should handle name as number type	20.93ms
✔ Boundary Testing - Member Search should handle name as boolean type	5.51ms
✔ Boundary Testing - Member Search should handle name as array	5.36ms
✔ Boundary Testing - Member Search should handle name as object	5.16ms

Admin Member Creation Validation

✔ Admin Member Creation Validation should return 400 for missing folio_number	21.49ms
✔ Admin Member Creation Validation should return 400 for missing name	6.9ms
✔ Admin Member Creation Validation should handle invalid email format	8.98ms
✔ Admin Member Creation Validation should handle phone number with letters	7.29ms
✔ Admin Member Creation Validation should not leak DB details on duplicate folio error	6.71ms
✔ Admin Member Creation Validation should handle invalid date format for join_date	5.9ms
✔ Admin Member Creation Validation should handle extremely long name	7.17ms
✔ Admin Member Creation Validation should handle far future join_date	6.13ms

Admin Member Update Validation

✔ Admin Member Update Validation should return 400 for empty body	6.74ms
✔ Admin Member Update Validation should handle id = 0	6.13ms
✔ Admin Member Update Validation should handle negative id	6.08ms
✔ Admin Member Update Validation should handle string id	7.17ms

OTP Input Validation

✔ OTP Input Validation should handle memberId = 0 for send-otp	5.35ms
✔ OTP Input Validation should handle OTP with special characters	5.02ms
✔ OTP Input Validation should handle OTP longer than 6 digits	4.84ms
✔ OTP Input Validation should handle OTP as number type	4.23ms
✔ OTP Input Validation should handle missing OTP field	4.19ms
✔ OTP Input Validation should handle missing memberId field	4.26ms

✔ payment.security.test.js
17/17 passed • 130ms

Amount Tampering - Payment Initiation

- ✔ Amount Tampering - Payment Initiation should handle zero amount — 40.77ms
- ✔ Amount Tampering - Payment Initiation should handle negative amount — 8.28ms
- ✔ Amount Tampering - Payment Initiation should handle absurdly large amount — 5.07ms
- ✔ Amount Tampering - Payment Initiation should handle non-numeric amount — 4.57ms
- ✔ Amount Tampering - Payment Initiation should handle float amount (underpayment) — 4.87ms

Razorpay Signature Bypass

- ✔ Razorpay Signature Bypass should reject empty signature — 4.83ms
- ✔ Razorpay Signature Bypass should reject random signature string — 4.2ms
- ✔ Razorpay Signature Bypass should reject with missing razorpay_order_id — 4.34ms
- ✔ Razorpay Signature Bypass should reject with missing razorpay_payment_id — 6.84ms
- ✔ Razorpay Signature Bypass should reject with missing razorpay_signature — 4.36ms
- ✔ Razorpay Signature Bypass should reject with all fields missing — 4.12ms

Payment Replay & Period Manipulation

- ✔ Payment Replay & Period Manipulation should handle replay of same payment data — 10.25ms
- ✔ Payment Replay & Period Manipulation should handle empty periods array — 4.46ms
- ✔ Payment Replay & Period Manipulation should handle null periods — 4.24ms
- ✔ Payment Replay & Period Manipulation should handle non-existent memberId with valid signature — 7.13ms
- ✔ Payment Replay & Period Manipulation should handle payment with totalAmount as null — 4.82ms
- ✔ Payment Replay & Period Manipulation should not expose Razorpay key secret in response on failure — 5.8ms

✔ otpService.test.js
14/14 passed • 8ms

OTP Service

- ✔ generateOTP > OTP Service generateOTP should return a 6-digit string — 1.99ms
- ✔ generateOTP > OTP Service generateOTP should return different OTPs on multiple calls — 0.32ms
- ✔ maskEmail > OTP Service maskEmail should mask email with long name — 0.4ms
- ✔ maskEmail > OTP Service maskEmail should mask email with 3-char name — 0.48ms
- ✔ maskEmail > OTP Service maskEmail should mask email with 2-char name — 0.18ms
- ✔ maskEmail > OTP Service maskEmail should mask email with 1-char name — 0.23ms
- ✔ maskEmail > OTP Service maskEmail should preserve domain — 0.3ms
- ✔ storeOTP + verifyOTP > OTP Service storeOTP + verifyOTP should verify correct OTP successfully — 1.54ms
- ✔ storeOTP + verifyOTP > OTP Service storeOTP + verifyOTP should be case-insensitive for email — 0.39ms
- ✔ storeOTP + verifyOTP > OTP Service storeOTP + verifyOTP should reject wrong OTP and show remaining attempts — 0.37ms
- ✔ storeOTP + verifyOTP > OTP Service storeOTP + verifyOTP should reject after max verification attempts — 0.25ms
- ✔ storeOTP + verifyOTP > OTP Service storeOTP + verifyOTP should reject non-existent email — 0.29ms
- ✔ storeOTP + verifyOTP > OTP Service storeOTP + verifyOTP should reject expired OTP — 0.28ms
- ✔ storeOTP + verifyOTP > OTP Service storeOTP + verifyOTP should delete OTP after successful verification (one-time use) — 0.31ms

✔ paymentUtils.test.js

23/23 passed • 45ms

Payment Utilities

✔ MEMBERSHIP_FEE > Payment Utilities MEMBERSHIP_FEE should be 1200 — 1.84ms

✔ getCurrentMembershipYear > Payment Utilities getCurrentMembershipYear should return current year as start when in Apr-Dec — 3.99ms

✔ getCurrentMembershipYear > Payment Utilities getCurrentMembershipYear should return previous year as start when in Jan-Mar — 0.52ms

✔ getCurrentMembershipYear > Payment Utilities getCurrentMembershipYear should return correct year when in April (boundary) — 0.45ms

✔ getCurrentMembershipYear > Payment Utilities getCurrentMembershipYear should return correct year when in March (boundary) — 0.43ms

✔ generateMembershipYears > Payment Utilities generateMembershipYears should generate correct range of years — 2.22ms

✔ generateMembershipYears > Payment Utilities generateMembershipYears should return single year when start equals end — 0.23ms

✔ generateMembershipYears > end — 0.19ms

✔ parseMembershipYear > Payment Utilities parseMembershipYear should extract year from date string — 0.27ms

✔ parseMembershipYear > Payment Utilities parseMembershipYear should handle ISO date strings — 0.13ms

✔ calculatePayableYears > Payment Utilities calculatePayableYears should return current year for empty history (first-time) — 0.96ms

✔ calculatePayableYears > Payment Utilities calculatePayableYears should return current year for null history — 0.58ms

✔ calculatePayableYears > Payment Utilities calculatePayableYears should detect all paid up to current year — 0.72ms

✔ calculatePayableYears > Payment Utilities calculatePayableYears should detect gap in payment history — 0.56ms

✔ calculatePayableYears > Payment Utilities calculatePayableYears should calculate sequential unpaid years — 0.74ms

✔ calculatePayableYears > Payment Utilities calculatePayableYears should handle history with no successful payments — 0.66ms

✔ formatAmount > Payment Utilities formatAmount should format 1200 as INR currency — 26.41ms

✔ formatAmount > Payment Utilities formatAmount should format 0 — 1.26ms

✔ formatAmount > Payment Utilities formatAmount should format large amounts with Indian grouping — 1.45ms

✔ isValidMembershipYear > Payment Utilities isValidMembershipYear should return true for valid April-March period — 0.26ms

✔ isValidMembershipYear > Payment Utilities isValidMembershipYear should return false for non-April start — 0.13ms

✔ isValidMembershipYear > Payment Utilities isValidMembershipYear should return false for non-March end — 0.23ms

✔ isValidMembershipYear > Payment Utilities isValidMembershipYear should return false for same year start and end — 0.15ms

✔ razorpay.test.js

4/4 passed • 4ms

Razorpay Signature Verification

✔ Razorpay Signature Verification should return true for valid signature — 2.98ms

✔ Razorpay Signature Verification should return false for tampered signature — 0.3ms

✔ Razorpay Signature Verification should return false for wrong orderId — 0.26ms

✔ Razorpay Signature Verification should return false for wrong paymentId — 0.19ms