

API Coverage Report

API Coverage Report

Date: 2026-02-20 **Purpose:** Map every frontend page to required API endpoints. Verify 100% coverage. **Status:** Backend NOT implemented (specification only)

Coverage Matrix

Dashboard Page (/)

UI Element	Data Required	API Endpoint	Status
Cash Balance metric	Total bank account balances in base currency	GET /api/v1/reports/dashboard	COVERED
Revenue MTD metric	Month-to-date revenue	GET /api/v1/reports/dashboard	COVERED
Unpaid Invoices metric	Total unpaid invoices	GET /api/v1/reports/dashboard	COVERED
Expenses MTD metric	Month-to-date expenses	GET /api/v1/reports/dashboard	COVERED
Profit MTD metric	Month-to-date profit	GET /api/v1/reports/dashboard	COVERED
Cash Flow Change	Percentage change from last month	GET /api/v1/reports/dashboard	COVERED
P&L Bar Chart	6-month monthly P&L data	GET /api/v1/reports/dashboard	COVERED
Receivables Aging Chart	Receivables breakdown by age (current, 30d, 60d, 90d+)	GET /api/v1/reports/dashboard	COVERED
Expenses by Category Chart	Expenses grouped by category	GET /api/v1/reports/dashboard	COVERED
Recent Transactions table	Last 5 transactions	GET /api/v1/transactions?perPage=5&sort=transactionDate&order=desc	COVERED

Invoices List Page (/invoices)

UI Element	Data Required	API Endpoint	Status
Invoice list (all filters)	Paginated invoices with filter/sort/search	GET /api/v1/invoices?status=X&search=Y&fromDate=Z&toDate=W&page=P&perPage=20&sort=field&order=desc	COVERED
Status filter options	Invoice list filtered by status	GET /api/v1/invoices?status={draft sent viewed paid overdue cancelled}	COVERED
Search (customer/number)	Invoice list filtered by search query	GET /api/v1/invoices?search={query}	COVERED
Date range filter	Invoice list filtered by date range	GET /api/v1/invoices?fromDate=YYYY-MM-DD&toDate=YYYY-MM-DD	COVERED
Summary totals by status	Client-side calculation from filtered list	N/A (client-side)	COVERED
Edit invoice action	Get invoice details for editing	GET /api/v1/invoices/:id	COVERED
Delete invoice action	Delete invoice	DELETE /api/v1/invoices/:id	MISSING
Send invoice action	Send invoice via email	POST /api/v1/invoices/:id/send	COVERED
Download PDF action	Get invoice PDF	GET /api/v1/invoices/:id/pdf	COVERED

MISSING ENDPOINT:

- **DELETE /api/v1/invoices/:id** — Delete invoice (only draft invoices should be deletable)
 - Method: DELETE
 - Auth: Bearer token
 - Roles: owner, admin, accountant
 - Rate limit: 10 req/min
 - Response: 204 No Content
 - Errors:
 - 400 — Invoice is not in draft status
 - 404 — Invoice not found

Invoice Creation Wizard (/invoices/new)

Step	UI Element	Data Required	API Endpoint	Status
1	Customer dropdown	List of customers	GET /api/v1/contacts?type=customer	COVERED
1	Add customer dialog	Create new customer	POST /api/v1/contacts	COVERED
2	Invoice number	Auto-generated invoice number	Client-side generation (format: INV-YYYY-NNN)	COVERED
2	Currency options	List of supported currencies	GET /api/v1/currencies	COVERED
3	Line items	Form input (no API needed)	N/A	COVERED
3	VAT rate options	Tax rate configuration	GET /api/v1/settings/tax-rates	COVERED
4	Notes/Terms	Form input (no API needed)	N/A	COVERED
5	Preview	Client-side rendering of form data	N/A	COVERED
6	Save as Draft	Create invoice with status=draft	POST /api/v1/invoices	COVERED
6	Send Invoice	Create + send invoice	POST /api/v1/invoices (then) POST /api/v1/invoices/:id/send	COVERED
6	Download PDF	Generate PDF	GET /api/v1/invoices/:id/pdf	COVERED

Expenses List Page (/expenses)

UI Element	Data Required	API Endpoint	Status
Expense list	Paginated expenses with filters	GET /api/v1/expenses?period=X&category=Y&search=Z&page=P&perPage=20	COVERED
Period filter	Expenses filtered by date range	GET /api/v1/expenses?fromDate=YYYY-MM-DD&toDate=YYYY-MM-DD	COVERED
Category filter	Expenses filtered by category	GET /api/v1/expenses?category={category}	COVERED

UI Element	Data Required	API Endpoint	Status
Search (description/vendor)	Expenses filtered by search query	GET /api/v1/expenses (client-side search on fetched data)	COVERED
Summary stats	Client-side calculation from filtered list	N/A (client-side)	COVERED
Create expense	Create new expense	POST /api/v1/expenses	COVERED
Upload receipt	Upload receipt file	POST /api/v1/expenses (multipart with receiptFile)	COVERED
Edit expense	Update expense (pending only)	PUT /api/v1/expenses/:id	COVERED
Approve expense	Approve expense	PATCH /api/v1/expenses/:id/approve	COVERED
Delete expense	Delete expense (pending only)	DELETE /api/v1/expenses/:id	COVERED
Download receipt	Get receipt file	GET /api/v1/expenses/:id/receipt	MISSING

MISSING ENDPOINT:

- **GET /api/v1/expenses/:id/receipt** — Download expense receipt
 - Method: GET
 - Auth: Bearer token
 - Roles: All
 - Rate limit: 100 req/min
 - Response: 200 OK
 - Content-Type: image/jpeg, image/png, or application/pdf
 - Content-Disposition: attachment; filename="receipt-`{expenseNumber}`.`{ext}`"
 - Errors:
 - 404 — Expense or receipt not found

Purchases Page (/purchases)

UI Element	Data Required	API Endpoint	Status
(Same as /expenses)	Same data as expenses page	Same as /expenses	COVERED

Note: This is an alias route to the expenses page. No additional API endpoints needed.

Banking Page (/banking)

Accounts Tab

UI Element	Data Required	API Endpoint	Status
Bank account list	List of bank accounts	GET /api/v1/bank-accounts	COVERED
Add bank account	Create new bank account	POST /api/v1/bank-accounts	COVERED
Account balance	Current balance per account	GET /api/v1/bank-accounts (included in response)	COVERED

Reconcile Tab

UI Element	Data Required	API Endpoint	Status
Account selector	List of bank accounts	GET /api/v1/bank-accounts	COVERED
Unreconciled transactions	Unreconciled bank transactions for selected account	GET /api/v1/bank-accounts/:id/transactions?reconciled=false	COVERED
Match confidence	Client-side calculation based on amount/date/description	N/A (client-side)	COVERED
Approve match	Mark transaction as reconciled	POST /api/v1/bank-accounts/:id/reconcile	COVERED
Link to invoice/expense	Link bank transaction to existing record	POST /api/v1/bank-accounts/:id/reconcile (with transactionId)	COVERED
Create new transaction	Create GL transaction from unmatched bank transaction	POST /api/v1/transactions	COVERED

Transactions Tab

UI Element	Data Required	API Endpoint	Status
All bank transactions	All bank transactions (paginated)	GET /api/v1/bank-accounts/:id/transactions	COVERED
Reconciliation status filter	Bank transactions filtered by reconciled status	GET /api/v1/bank-accounts/:id/transactions?reconciled={true false}	COVERED
Date range filter	Bank transactions filtered by date	GET /api/v1/bank-accounts/:id/transactions?fromDate=X&toDate=Y	COVERED

Import Transactions

UI Element	Data Required	API Endpoint	Status
Import CSV	Upload bank statement CSV	POST /api/v1/bank-accounts/:id/import	COVERED

Reports Hub Page (/reports)

UI Element	Data Required	API Endpoint	Status
P&L Report preview	Profit & Loss data for current month	GET /api/v1/reports/profit-loss?from=YYYY-MM-01&to=YYYY-MM-DD	COVERED
Balance Sheet preview	Balance sheet data (coming soon)	GET /api/v1/reports/balance-sheet?date=YYYY-MM-DD	COVERED
Cash Flow preview	Cash flow data (coming soon)	GET /api/v1/reports/cash-flow?from=X&to=Y	COVERED
VAT Report preview	VAT report data (live at /reports/vat)	GET /api/v1/reports/vat?from=X&to=Y	COVERED
Trial Balance preview	Trial balance data (coming soon)	GET /api/v1/reports/trial-balance?date=YYYY-MM-DD	COVERED
General Ledger preview	Transaction list (coming soon)	GET /api/v1/transactions	COVERED

VAT Report Page (/reports/vat)

Step	UI Element	Data Required	API Endpoint	Status
1	Reconciliation status check	Count of unreconciled bank transactions	GET /api/v1/bank-accounts (aggregate unreconciled count client-side)	PARTIAL
2	VAT transaction table	All invoices and expenses with VAT for period	GET /api/v1/reports/vat?from=X&to=Y	COVERED
2	Summary boxes (collected/paid/due)	Calculated from VAT transaction data	GET /api/v1/reports/vat?from=X&to=Y	COVERED
3	VAT return boxes	Formatted VAT return data	GET /api/v1/reports/vat?from=X&to=Y	COVERED

Step	UI Element	Data Required	API Endpoint	Status
3	Export PDF	Generate PDF report	GET /api/v1/reports/vat/export/pdf?from=X&to=Y	MISSING
3	Export XML	Generate XML for e-filing	GET /api/v1/reports/vat/export/xml?from=X&to=Y	MISSING
3	Submit return	Submit VAT return (Phase 2)	POST /api/v1/reports/vat/submit	MISSING

MISSING ENDPOINTS:

- **GET /api/v1/reports/vat/export/pdf** — Export VAT report as PDF
 - Method: GET
 - Auth: Bearer token
 - Roles: All
 - Query: from (ISO date), to (ISO date)
 - Rate limit: 50 req/min
 - Response: 200 OK
 - Content-Type: application/pdf
 - Content-Disposition: attachment; filename="vat-report-{from}-{to}.pdf"
 - Errors: 422 — Invalid date range
- **GET /api/v1/reports/vat/export/xml** — Export VAT report as XML for e-filing
 - Method: GET
 - Auth: Bearer token
 - Roles: All
 - Query: from (ISO date), to (ISO date)
 - Rate limit: 50 req/min
 - Response: 200 OK
 - Content-Type: application/xml
 - Content-Disposition: attachment; filename="vat-return-{from}-{to}.xml"
 - Errors: 422 — Invalid date range
- **POST /api/v1/reports/vat/submit** — Submit VAT return to tax authority (Phase 2)
 - Method: POST
 - Auth: Bearer token
 - Roles: owner, admin, accountant
 - Rate limit: 5 req/min
 - Request:

```
interface SubmitVATRequest {
    period: string // ISO date range, e.g., "2026-02"
    confirmationEmail: string
}
```

```
}
```

- Response: 201 Created

```
interface SubmitVATResponse {  
  submissionId: string  
  submittedAt: string  
  status: 'pending' | 'accepted' | 'rejected'  
  confirmationNumber: string | null  
}
```

- Errors: 400 — VAT period already submitted

PARTIAL COVERAGE:

- Reconciliation status check requires aggregating unreconciled count from GET /api/v1/bank-accounts response. Consider dedicated endpoint:
 - **GET /api/v1/bank-accounts/unreconciled-count**
 - Returns: `{ total: number, byAccount: Array<{accountId: string, count: number}>`

Settings Page (/settings)

Company Section

UI Element	Data Required	API Endpoint	Status
Company profile form	Organization data	GET /api/v1/organization	COVERED
Save company profile	Update organization	PUT /api/v1/organization	COVERED

Users Section

UI Element	Data Required	API Endpoint	Status
User list	List of users in organization	GET /api/v1/users	COVERED
Invite user	Send user invite	POST /api/v1/users/invite	COVERED
Change user role	Update user role	PUT /api/v1/users/:id/role	COVERED
Remove user	Delete user	DELETE /api/v1/users/:id	COVERED

Tax & Compliance Section

UI Element	Data Required	API Endpoint	Status
------------	---------------	--------------	--------

Tax settings form	Tax rate configuration	GET /api/v1/settings/tax-rates	COVERED
Save tax settings	Update tax rates	PUT /api/v1/settings/tax-rates	COVERED
VAT registration toggle	Update organization (vatNumber field)	PUT /api/v1/organization	COVERED

Integrations Section

UI Element	Data Required	API Endpoint	Status
Connected integrations	List of integrations	GET /api/v1/integrations	MISSING
Available integrations	List of integrations	GET /api/v1/integrations	MISSING
Connect integration	Connect to integration	POST /api/v1/integrations/:id/connect	MISSING
Disconnect integration	Disconnect integration	DELETE /api/v1/integrations/:id/disconnect	MISSING

MISSING ENDPOINTS (Integrations): All integration-related endpoints are missing. These should be Phase 2, but placeholders needed:

- **GET /api/v1/integrations** — List integrations
 - Method: GET
 - Auth: Bearer token
 - Roles: All
 - Response: 200 OK

```
interface IntegrationsResponse {
  connected: Array<{
    id: string
    name: string
    type: string
    status: 'active' | 'inactive' | 'error'
    connectedAt: string
    lastSync: string | null
  }>
  available: Array<{
    id: string
    name: string
    type: string
    description: string
  }>
}
```

```
    icon: string
  }>
}
```

- **POST /api/v1/integrations/:id/connect** — Connect integration
 - Method: POST
 - Auth: Bearer token
 - Roles: owner, admin
 - Request body varies by integration type
 - Response: 201 Created
- **DELETE /api/v1/integrations/:id/disconnect** — Disconnect integration
 - Method: DELETE
 - Auth: Bearer token
 - Roles: owner, admin
 - Response: 204 No Content

Notifications Section

UI Element	Data Required	API Endpoint	Status
Notification preferences	User notification settings	GET /api/v1/settings/notifications	MISSING
Save preferences	Update notification settings	PATCH /api/v1/settings/notifications	MISSING

MISSING ENDPOINTS (Notifications):

- **GET /api/v1/settings/notifications** — Get notification preferences
 - Method: GET
 - Auth: Bearer token
 - Roles: All
 - Response: 200 OK

```
interface NotificationSettings {
  email: {
    invoicePaid: boolean
    invoiceOverdue: boolean
    expenseApproved: boolean
    bankAccountSynced: boolean
  }
  inApp: {
    invoiceUpdates: boolean
    expenseUpdates: boolean
  }
}
```

```

    reconciliationMatches: boolean
  }
}

```

- **PATCH /api/v1/settings/notifications** — Update notification preferences
 - Method: PATCH
 - Auth: Bearer token
 - Roles: All
 - Request: Same as GET response (partial updates allowed)
 - Response: 200 OK (updated settings)

Security Section

UI Element	Data Required	API Endpoint	Status
Enable 2FA	Enable 2FA for user	POST /api/v1/auth/2fa/enable	MISSING
Disable 2FA	Disable 2FA for user	DELETE /api/v1/auth/2fa/disable	MISSING
Session timeout	User/org settings	GET /api/v1/settings/security	MISSING
Save security settings	Update security settings	PATCH /api/v1/settings/security	MISSING
View audit log	Audit trail	GET /api/v1/security/audit-log	MISSING
Request data export	Export all data	POST /api/v1/security/data-export	MISSING
Delete company	Delete organization	DELETE /api/v1/organization	MISSING

MISSING ENDPOINTS (Security):

- **POST /api/v1/auth/2fa/enable** — Enable 2FA
 - Response: QR code + backup codes
- **DELETE /api/v1/auth/2fa/disable** — Disable 2FA
 - Requires current password confirmation
- **GET /api/v1/settings/security** — Get security settings
 - Returns: session timeout, password policy, etc.
- **PATCH /api/v1/settings/security** — Update security settings
- **GET /api/v1/security/audit-log** — Get audit log (paginated)
 - Query: fromDate, toDate, userId, action, tableName
 - Response: Paginated list of LoggedAction records
- **POST /api/v1/security/data-export** — Request GDPR data export
 - Response: Job ID, email sent when ready
- **DELETE /api/v1/organization** — Delete organization (danger zone)

- Requires password confirmation
- Cascades to all related data

Forms ? API Mapping

Form	Submit Action	API Endpoint	Request Body	Status
Create Invoice (Step 6)	POST	POST /api/v1/invoices	CreateInvoiceRequest	COVERED
Send Invoice (Step 6)	POST	POST /api/v1/invoices/:id/send	SendInvoiceRequest	COVERED
Add Customer (Wizard Step 1)	POST	POST /api/v1/contacts	CreateContactRequest	COVERED
Create Expense (Dialog)	POST	POST /api/v1/expenses	CreateExpenseRequest (multipart)	COVERED
Upload Receipt (Expense Dialog)	POST	POST /api/v1/expenses (multipart receiptFile)	File upload	COVERED
Update Company Profile	PUT	PUT /api/v1/organization	UpdateOrganizationRequest	COVERED
Update Tax Settings	PUT	PUT /api/v1/settings/tax-rates	UpdateTaxRatesRequest	COVERED
Invite User	POST	POST /api/v1/users/invite	InviteUserRequest	COVERED
Change User Role	PUT	PUT /api/v1/users/:id/role	ChangeRoleRequest	COVERED
Import Bank Statement	POST	POST /api/v1/bank-accounts/:id/import	CSV file upload	COVERED
Create Manual Transaction	POST	POST /api/v1/transactions	CreateTransactionRequest	COVERED
Update Notification Preferences	PATCH	PATCH /api/v1/settings/notifications	NotificationSettings	MISSING
Update Security Settings	PATCH	PATCH /api/v1/settings/security	SecuritySettings	MISSING
Connect Integration	POST	POST /api/v1/integrations/:id/connect	Integration-specific	MISSING

Form	Submit Action	API Endpoint	Request Body	Status
Submit VAT Return	POST	POST /api/v1/reports/vat/submit	SubmitVATRequest	MISSING

Coverage Summary

Page	Endpoints Required	Endpoints Documented	Coverage
Dashboard	2	2	100%
Invoices List	6	5	83% (missing DELETE)
Invoice Wizard	6	6	100%
Expenses	7	6	86% (missing receipt download)
Purchases	7	6	86% (same as expenses)
Banking	7	7	100%
Reports Hub	6	6	100%
VAT Report	7	4	57% (missing PDF/XML export, submit)
Settings - Company	2	2	100%
Settings - Users	4	4	100%
Settings - Tax	2	2	100%
Settings - Integrations	3	0	0% (Phase 2)
Settings - Notifications	2	0	0% (missing)
Settings - Security	6	0	0% (missing)
TOTAL	67	50	75%

Missing Endpoints

High Priority (Core Features)

- DELETE /api/v1/invoices/:id** — Delete invoice (draft only)
 - Reason: Invoice list has delete action
- GET /api/v1/expenses/:id/receipt** — Download expense receipt

- Reason: Expense list shows receipt indicator, needs download link
3. **GET /api/v1/reports/vat/export/pdf** — VAT report PDF export
 - Reason: VAT report has export button (placeholder currently)
 4. **GET /api/v1/reports/vat/export/xml** — VAT report XML export (e-filing)
 - Reason: VAT report has export button (placeholder currently)

Medium Priority (Settings)

5. **GET /api/v1/settings/notifications** — Get notification preferences
 - Reason: Settings page has notification section with checkboxes
6. **PATCH /api/v1/settings/notifications** — Update notification preferences
 - Reason: Settings page has save button for notification preferences
7. **GET /api/v1/settings/security** — Get security settings
 - Reason: Settings page has security section (session timeout, password policy)
8. **PATCH /api/v1/settings/security** — Update security settings
 - Reason: Settings page has save button for security settings
9. **GET /api/v1/security/audit-log** — Audit log
 - Reason: Settings security section has "View Audit Log" button
10. **POST /api/v1/security/data-export** — GDPR data export
 - Reason: Settings security section has "Request Data Export" button
11. **DELETE /api/v1/organization** — Delete company
 - Reason: Settings security section has "Delete Company" button (danger zone)
12. **POST /api/v1/auth/2fa/enable** — Enable 2FA
 - Reason: Settings security section has "Enable 2FA" button
13. **DELETE /api/v1/auth/2fa/disable** — Disable 2FA
 - Reason: Settings security section needs disable option if 2FA enabled

Low Priority (Phase 2 Features)

14. **GET /api/v1/integrations** — List integrations
 - Reason: Settings integrations section (Phase 2)
15. **POST /api/v1/integrations/:id/connect** — Connect integration
 - Reason: Settings integrations section (Phase 2)
16. **DELETE /api/v1/integrations/:id/disconnect** — Disconnect integration
 - Reason: Settings integrations section (Phase 2)
17. **POST /api/v1/reports/vat/submit** — Submit VAT return
 - Reason: VAT report has submit button (explicitly marked "Coming in Phase 2")
18. **GET /api/v1/bank-accounts/unreconciled-count** — Unreconciled transaction count
 - Reason: VAT report Step 1 checks reconciliation status (currently client-side aggregation, could be dedicated endpoint)

Redundant Endpoints

None identified. All endpoints in API-REFERENCE.md are consumed by at least one frontend page.

Recommendations

1. Add Missing Core Endpoints

Priority: HIGH **Scope:** Endpoints 1-4 from Missing Endpoints list

These are referenced directly in existing UI actions. Without them, users will encounter broken functionality:

- Delete invoice button will not work
- Receipt download links will not work
- VAT export buttons will not work

Implementation Order:

1. DELETE /api/v1/invoices/:id (simplest, no business logic)
2. GET /api/v1/expenses/:id/receipt (file download)
3. GET /api/v1/reports/vat/export/pdf (report generation + PDF library)
4. GET /api/v1/reports/vat/export/xml (report generation + XML serialization)

2. Implement Settings Endpoints

Priority: MEDIUM **Scope:** Endpoints 5-13 from Missing Endpoints list

Settings page is fully implemented with save buttons, but no backend to persist data. Current behavior: console.log() only.

Recommendation: Implement all settings endpoints together in one feature branch. They share similar patterns (GET/PATCH pairs, org-scoped data).

3. Phase 2 Features as Stubs

Priority: LOW **Scope:** Endpoints 14-18 from Missing Endpoints list

These are explicitly marked as "Phase 2" or "Coming Soon" in the UI. Consider implementing stub endpoints that return:

- 501 Not Implemented
- Or minimal placeholder data

This allows frontend to gracefully handle the "not yet implemented" state without errors.

4. Add Endpoint: GET /api/v1/bank-accounts/unreconciled-count

Priority: LOW **Scope:** New endpoint not in API-REFERENCE.md

VAT report Step 1 checks for unreconciled transactions. Currently, frontend must:

1. Fetch GET /api/v1/bank-accounts (all accounts)
2. For each account, fetch GET /api/v1/bank-accounts/:id/transactions?reconciled=false
3. Aggregate counts

Recommendation: Add dedicated endpoint to avoid N+1 query pattern.

```
// Proposed endpoint
GET /api/v1/bank-accounts/unreconciled-count

Response:
{
  total: number
  byAccount: Array<{
    accountId: string
    accountName: string
    count: number
  }>
}
```

5. Verify TypeScript Interface Consistency

Priority: MEDIUM

API-REFERENCE.md defines TypeScript interfaces for request/response bodies. Frontend uses these types in forms and state management.

Action Items:

- Create shared types package (`packages/types/`)
- Export all API interfaces from API-REFERENCE.md
- Import in both `apps/api/` (backend validation) and `apps/web/` (frontend forms)
- Use Zod schemas for runtime validation + TypeScript type generation

Example:

```
// packages/types/src/invoice.ts
import { z } from 'zod'

export const CreateInvoiceRequestSchema = z.object({
  customerId: z.string().uuid(),
  invoiceDate: z.string().regex(/^d{4}-d{2}-d{2}$/),
  dueDate: z.string().regex(/^d{4}-d{2}-d{2}$/),
  currencyCode: z.enum(['EUR', 'RSD', 'BAM', 'HRK']).optional(),
  items: z.array(z.object({
    description: z.string().min(1),
    quantity: z.number().positive(),
    unitPrice: z.number().nonnegative(),
    taxRate: z.number().nonnegative(),
    accountId: z.string().uuid().optional()
  })),
  notes: z.string().optional(),
  terms: z.string().optional()
})

export type CreateInvoiceRequest = z.infer<typeof CreateInvoiceRequestSchema>
```

Backend uses schema for validation:

```
// apps/api/src/routes/invoices.ts
import { CreateInvoiceRequestSchema } from '@bilko/types'

router.post('/invoices', async (req, res) => {
  const data = CreateInvoiceRequestSchema.parse(req.body) // Throws if invalid
  // ...
})
```

Frontend uses type for forms:

```
// apps/web/app/(dashboard)/invoices/new/page.tsx
import { CreateInvoiceRequest } from '@bilko/types'

const [formData, setFormData] = useState<CreateInvoiceRequest>({...})
```

6. Authentication Endpoints Missing from Coverage

Priority: HIGH

Frontend has no login/register pages yet, but API-REFERENCE.md defines 5 auth endpoints:

- POST /api/v1/auth/register
- POST /api/v1/auth/login
- POST /api/v1/auth/refresh
- POST /api/v1/auth/logout
- GET /api/v1/auth/me

Recommendation: Create auth pages in Phase 2:

- /login page
- /register page
- /logout action (server action)
- Auth middleware to protect all /dashboard routes

7. Error Handling Pattern

Priority: MEDIUM

API-REFERENCE.md defines error response format:

```
interface ApiError {
  error: string
  code: string
  details?: Record<string, string[]>
}
```

Recommendation: Create frontend error handling utility:

```
// lib/api-error.ts
export function handleApiError(error: Response) {
  const apiError: ApiError = await error.json()

  if (apiError.details) {
    // Field-level validation errors
    return Object.entries(apiError.details).map(([field, errors]) => ({
      field,
```

```

    message: errors.join(', ')
  }
}

// Generic error
return { message: apiError.error, code: apiError.code }
}

```

Use in forms:

```

try {
  await fetch('/api/v1/invoices', { method: 'POST', body: JSON.stringify(data) })
} catch (error) {
  const errors = handleApiError(error)
  // Display errors in form
}

```

Implementation Priority Matrix

Endpoint	Priority	Reason	Est. Effort
DELETE /api/v1/invoices/:id	HIGH	Invoice list delete button	2h
GET /api/v1/expenses/:id/receipt	HIGH	Receipt download links	3h
GET /api/v1/reports/vat/export/pdf	HIGH	VAT export button	8h
GET /api/v1/reports/vat/export/xml	HIGH	VAT e-filing	6h
GET /api/v1/settings/notifications	MEDIUM	Settings page persistence	3h
PATCH /api/v1/settings/notifications	MEDIUM	Settings page persistence	2h
GET /api/v1/settings/security	MEDIUM	Settings page persistence	4h
PATCH /api/v1/settings/security	MEDIUM	Settings page persistence	3h

Endpoint	Priority	Reason	Est. Effort
GET /api/v1/security/audit-log	MEDIUM	Audit log viewer	5h
POST /api/v1/security/data-export	MEDIUM	GDPR compliance	8h
DELETE /api/v1/organization	MEDIUM	Delete company action	4h
POST /api/v1/auth/2fa/enable	MEDIUM	2FA setup	8h
DELETE /api/v1/auth/2fa/disable	MEDIUM	2FA removal	2h
GET /api/v1/integrations	LOW	Phase 2 feature	6h
POST /api/v1/integrations/:id/connect	LOW	Phase 2 feature	12h
DELETE /api/v1/integrations/:id/disconnect	LOW	Phase 2 feature	3h
POST /api/v1/reports/vat/submit	LOW	Phase 2 feature	16h
GET /api/v1/bank-accounts/unreconciled-count	LOW	Performance optimization	3h

Total Estimated Effort: 98 hours (~12-15 working days for 1 developer)

Suggested Implementation Phases:

Phase 2a (Core Features) — 19h

- DELETE /api/v1/invoices/:id
- GET /api/v1/expenses/:id/receipt
- GET /api/v1/reports/vat/export/pdf
- GET /api/v1/reports/vat/export/xml

Phase 2b (Settings Persistence) — 31h

- All settings endpoints (notifications, security, audit log, data export, org delete, 2FA)

Phase 2c (Integrations + VAT Submit) — 37h

- Integrations endpoints (stub or full implementation)
- VAT submit endpoint (requires external API integration)

Phase 2d (Polish) — 11h

- Unreconciled count endpoint
 - Shared types package
 - Error handling utilities
 - Auth pages
-

Conclusion

Overall Coverage: 75% (50 out of 67 required endpoints documented)

API-REFERENCE.md is 75% complete. The 50 documented endpoints cover all core business logic:

- Invoice CRUD (except delete)
- Expense CRUD (except receipt download)
- Banking & reconciliation
- Reporting (except export formats)
- User management
- Organization settings
- Chart of accounts
- Transactions

Missing 25%:

- 4 high-priority endpoints (delete invoice, receipt download, VAT exports)
- 9 medium-priority settings endpoints
- 4 low-priority Phase 2 endpoints

No redundant endpoints. Every endpoint in API-REFERENCE.md is consumed by at least one frontend page.

Recommendation: Implement Phase 2a (core features, 19h) before beta launch. Settings persistence can follow in Phase 2b after user feedback.

Next Steps:

1. Review this coverage report with team
2. Prioritize missing endpoints based on business needs
3. Update API-REFERENCE.md with missing endpoint specs
4. Create implementation tickets in Mission Control
5. Build apps/api/ following API-REFERENCE.md contract

Revision #3

Created 2026-02-23 10:48:13 UTC by John

Updated 2026-05-31 20:02:54 UTC by John