

# Tech Debt Log

## Tech Debt Log: Bilko

“ **Project:** Bilko — Balkan Accounting SaaS **Version:** 0.1 **Date:** 2026-02-23  
**Author:** John (AI Director) **Status:** Draft **Reviewers:** Alem Bašić (CEO)

### Document History

Version	Date	Author	Changes
0.1	2026-02-23	John (AI Director)	Initial population — known debt from Phase 1 kickoff

## 1. Tech Debt Philosophy

Technical debt is not inherently bad. The Bilko MVP was built quickly (by design) to validate market fit before spending full engineering resources. The known debts below were **conscious decisions** — the correct ones at the time. This log exists to make them visible, prioritizable, and tracked to resolution.

#### Bilko-specific debt policy:

- **Financial logic debt (double-entry, PDV, SEF):** Zero tolerance — fix immediately. Accounting errors have legal consequences for users.
- **UI/UX debt (mock data, placeholder pages):** High priority — block beta launch if not resolved.
- **Infrastructure debt (CI/CD, monitoring):** High priority — block production launch if not resolved.
- **Documentation debt:** Medium — resolve before Croatia launch.

## 2. Summary Dashboard

Metric	Current	Last Sprint	Trend
Total Items	11	0	↑ (initial population)
Open	11	0	↑
In Progress	0	0	→
Resolved	0	0	→
Critical / High (P1/P2)	5	0	↑
Debt in current sprint	0	0	→
Avg age of open items (days)	3 (project just started)	—	—
Debt resolved this sprint	0	0	→

**Sprint Debt Budget:** 20% of sprint capacity allocated to debt reduction starting Sprint 2.

### 3. Tech Debt Categories

Category	Description	Common Causes	Detection Method
<b>Design</b>	Wrong abstractions, poor separation	Speed, evolving requirements	Code review
<b>Code Quality</b>	Duplication, complexity, poor naming	Speed over quality	Static analysis
<b>Infrastructure</b>	Manual processes, missing monitoring	Growth outpacing ops maturity	Ops review
<b>Documentation</b>	Missing or outdated docs	Documentation deferred	New team feedback
<b>Dependency</b>	Outdated libraries, deprecated APIs	Insufficient update cadence	Snyk/Dependabot
<b>Testing</b>	Low coverage, missing tests, flaky tests	Speed over quality	Coverage reports
<b>Security</b>	Known vulnerabilities, security shortcuts	Urgency	Security scans
<b>Performance</b>	N+1 queries, no caching, bottlenecks	Optimizations deferred	Performance monitoring

### 4. Impact Assessment Methodology

## 4.1 Impact Score (1–10)

Score	Impact Level	Effect on Team/Business
1-2	Negligible	Cosmetic; no measurable effect
3-4	Minor	Slightly slows specific tasks; <5% velocity impact
5-6	Moderate	Slows multiple tasks; 5-15% velocity impact
7-8	High	Significantly slows development; recurring bugs; 15-30% impact
9-10	Critical	Blocks progress or creates legal/financial risk for users

## 4.2 Effort to Fix (Story Points)

Size	Points	Typical Scope
XS	1	Single line / config change
S	2-3	Single function / component refactor
M	5-8	Module-level changes
L	13	Multi-module refactoring
XL	21+	Major refactor; break into sub-tasks first

## 4.3 Priority Calculation

Priority	Criteria	Sprint Allocation Target
P1 — Critical	Impact $\geq$ 8 OR blocks beta/launch	Schedule within 1 sprint
P2 — High	Impact 6-7 OR blocks a planned feature	Schedule within 2 sprints
P3 — Medium	Impact 4-5	Schedule within 4 sprints
P4 — Low	Impact $\leq$ 3	Schedule when convenient

## 5. Tech Debt Register

ID	Description	Category	Introduced	Impact (1-10)	Effort	Priority	Owner	Status	Target Resolution	Notes
TD-001	<p><b>ALL frontend data served from mock-data.ts</b></p> <p>— 8 pages (Dashboard, Invoices, Expenses, Purchases, Banking, Reports, VAT, Settings) use hardcoded mock data instead of real API calls</p>	Code Quality	2026-02-19 (MVP build)	10	L (13)	P1	John	Open	2026-03-14	BLOCKS beta launch. CI must grep for mock-data.ts imports and fail build in staging/production

ID	Description	Category	Introduced	Impact (1-10)	Effort	Priority	Owner	Status	Target Resolution	Notes
TD-002	<b>Backend API 0/50 core endpoints</b> — Only 4 auth endpoints built. Invoice, expense, contact, account, transaction, report, banking endpoints all missing	Design	2026-02-20 (MVP build)	10	XL (50 endpoints)	P1	John	In Progress	2026-03-14	Primary Sprint 1-2 work. 4/50 done (auth).
TD-003	<b>packages/ui empty scaffold</b> — Shared UI component library initialized but empty. All components live inside apps/web directly, causing potential duplication if mobile app added	Design	2026-02-19	4	M (8)	P3	John	Open	2026-05-01	Not urgent for Phase 1. Becomes critical if React Native added in Phase 2

ID	Description	Category	Introduced	Impact (1-10)	Effort	Priority	Owner	Status	Target Resolution	Notes
TD-004	<b>No unit tests written</b> — 0% test coverage on the entire codebase. Double-entry logic, PDV calculation, and SEF integration have zero automated verification	Testing	2026-02-20	9	L (13 for critical paths)	P1	John	Open	2026-03-07	Financial software without tests is a legal risk. CI must enforce $\geq 80\%$ coverage before merge to main
TD-005	<b>No CI/CD pipeline</b> — No GitHub Actions or equivalent. All deployment is manual. No automated test runs on PR. No staging environment defined	Infrastructure	2026-02-20	8	M (8)	P1	John	Open	2026-03-07	Blocks safe iteration. Must be operational before any team member merges code

ID	Description	Category	Introduced	Impact (1-10)	Effort	Priority	Owner	Status	Target Resolution	Notes
TD-006	<b>No monitoring or alerting</b> — No uptime monitor, no error tracking (Sentry), no APM tool configured. SEF submission failures would go undetected	Infrastructure	2026-02-20	8	S (3)	P2	John	Open	2026-03-21	Minimum: uptime monitor + Sentry for production launch
TD-007	<b>SEF integration not built</b> — SefService placeholder exists conceptually but no actual SEF API integration implemented. No UBL 2.1 XML generation	Design	2026-02-20	10	L (13)	P1	John	Open	2026-03-21	Serbia launch blocker. Must be fully tested in SEF sandbox before launch

ID	Description	Category	Introduced	Impact (1-10)	Effort	Priority	Owner	Status	Target Resolution	Notes
TD-008	<p><b>No database indexes on high-frequency queries</b> — Prisma schema has no explicit indexes . Invoice list query filtering by organizationId + status will be slow at scale (&gt;10K records)</p>	Performance	2026-02-20	6	S (3)	P2	John	Open	2026-03-28	Add indexes on: Invoice(organizationId, status), Transaction(organizationId, date), Expense(organizationId)

ID	Description	Category	Introduced	Impact (1-10)	Effort	Priority	Owner	Status	Target Resolution	Notes
TD-009	<b>No environment configuration validation</b> — Application starts even if critical env vars missing (DATABASE_URL, JWT_SECRET, SEF_API_KEY). Silent failures in production	Infrastructure	2026-02-20	7	XS (1)	P2	John	Open	2026-03-07	Add startup validation: fail fast if required env vars absent
TD-010	<b>i18n not implemented</b> — All text hardcoded in English in frontend components. Serbian language support (core requirement BR-011) not yet implemented	Code Quality	2026-02-19	7	M (8)	P2	John	Open	2026-03-28	Need next-intl or similar. Serbian (Latin) required for beta. Cyrillic optional initially

ID	Description	Category	Introduced	Impact (1-10)	Effort	Priority	Owner	Status	Target Resolution	Notes
TD-011	<b>API-REFERENCE. md not created</b> — Backend API contract documentation referenced in CLAUDE.md does not exist yet. API endpoints have no documented request/response schemas	Documentation	2026-02-20	5	M (5)	P3	John	Open	2026-03-14	Should be created alongside API endpoints. Prevents frontend/backend sync issues

**Status Values:** Open | In Progress | Resolved | Accepted (deferred) | Won't Fix (justified)

## 6. Prioritization Framework

### 6.1 Cost of Delay Analysis

ID	Description	Cost Per Sprint (velocity loss %)	Sprints Open	Accumulated Risk
TD-001	Mock data in frontend	30% — all features untestable with real data	1	HIGH — beta shows fake data
TD-004	No unit tests	15% — regressions catch later, cost 3x to fix	1	HIGH — accounting bugs undetected

ID	Description	Cost Per Sprint (velocity loss %)	Sprints Open	Accumulated Risk
TD-005	No CI/CD	20% — manual deploy risk of breaking production	1	CRITICAL — no safety net
TD-007	SEF not built	100% (Serbia launch impossible)	1	CRITICAL — launch blocker

## 6.2 Debt Payoff Priority Order

1. **TD-002** (Backend endpoints) — Sprint 1-2 focus, primary sprint work
2. **TD-007** (SEF integration) — Sprint 2-3, Serbia launch blocker
3. **TD-001** (Mock data replacement) — Sprint 2-3, coincides with TD-002
4. **TD-004** (Unit tests) — Ongoing from Sprint 1; written alongside features
5. **TD-005** (CI/CD) — Sprint 1, early; safety net for all subsequent work
6. **TD-009** (Env validation) — XS, quick win Sprint 1
7. **TD-008** (DB indexes) — S, pre-launch
8. **TD-010** (i18n) — Sprint 2-3, beta blocker
9. **TD-006** (Monitoring) — Sprint 3, pre-launch
10. **TD-011** (API docs) — Ongoing, Sprint 2
11. **TD-003** (Shared UI package) — Deferred to Phase 2

## 7. Sprint Debt Allocation Guidelines

Sprint	Debt Items to Address	Rationale
Sprint 1 (Feb 23 - Mar 7)	TD-005 (CI/CD), TD-009 (env validation), TD-004 (test framework setup)	Safety net before building features
Sprint 2 (Mar 7-14)	TD-001 (mock → real API), TD-002 (backend endpoints), TD-010 (i18n scaffold)	Primary feature work + debt concurrent
Sprint 3 (Mar 14-21)	TD-007 (SEF), TD-008 (DB indexes)	Serbia launch readiness
Sprint 4 (Mar 21-28)	TD-006 (monitoring), TD-011 (API docs)	Production quality
Sprint 5 (Alpha/Beta)	TD-004 (remaining test coverage to ≥80%)	Pre-launch quality gate

## 8. Debt Reduction Tracking

Sprint	Debt Items Added	Debt Items Resolved	Net Change	Running Total Open
Sprint 0 (kickoff)	11	0	+11	11
Sprint 1	TBD	TBD	TBD	TBD
Sprint 2	TBD	TBD	TBD	TBD

**Debt Trend:** INCREASING (initial population) → Target: DECREASING by Sprint 2

---

## 9. Trend Analysis

### 9.1 Debt by Category (initial state)

Category	Items Added	Items Resolved	Net
Code Quality	2	0	+2
Testing	1	0	+1
Design	3	0	+3
Dependencies	0	0	0
Infrastructure	3	0	+3
Documentation	1	0	+1
Security	0	0	0
Performance	1	0	+1

### 9.2 Root Cause Analysis

**Dominant category:** Design + Infrastructure (3 each) **Root cause:** Prototype-speed MVP build — all design/infrastructure decisions deferred in favor of demonstrating UI **Proposed systemic fix:** For future Bilko features, CI/CD + tests must be in place before any feature PR is mergeable. Hook: PR blocked without test file alongside new service.

---

## 10. Resolved / Accepted Debt Archive

ID	Description	Resolution	Resolution Date	Sprint	Resolved By
----	-------------	------------	-----------------	--------	-------------

---

—	No resolved items yet	—	—	—	—
---	-----------------------	---	---	---	---

---

# Approval

Role	Name	Date	Signature
Author	John (AI Director)	2026-02-23	
Reviewer			
Tech Lead	John	2026-02-23	
AI Director (John)	John	2026-02-23	

---

Revision #3

Created 2026-02-24 23:11:29 UTC by John

Updated 2026-05-31 20:04:24 UTC by John