TANA CORP

# Findings

Fictional example · Order management application of an industrial supplies distributor

| PROJECT | SHEET | REVISION | STRENGTHS | ITEMS TO ADDRESS |
|---|---|---|---|---|
| Order management | 2/4 | A | 3 | 6 |

Scope: a 42-screen application written in 2015 by a supplier, maintained since 2019 by a freelance developer, and its hosting. We read the code, installed the application on a test machine and met four users.

## What holds up

**The software covers the business.**
All 42 screens are used every week and the pricing rules give the right results.

**The database is well structured.**
It can take new modules without a redesign.

**Users know what they want.**
They have written 23 change requests. The three most cited are in the roadmap.

## Six items to address so it can keep evolving

| REF. | TOPIC | WHAT WE FOUND | WHAT WE PROPOSE | WHAT YOU GAIN | PRIORITY |
|---|---|---|---|---|---|
| C1 | **Access to the code**<br>SOURCE CODE | The code sits in the freelance developer's personal account, with no other access. No document explains how to install the application. | Move the repository to a company account, open three named accesses and write the installation guide. | Your company holds its own software. Another team can step in within days. | 1<br>30 DAYS |
| C2 | **Tests on pricing rules**<br>BUSINESS SERVICES | No automated tests. Discount calculation lives in a single 3,800-line file, changed 41 times in twelve months. | Write tests for the twelve pricing and invoicing rules, then split the file step by step. | Every change is checked before it goes live. Releases can become more frequent. | 2<br>90 DAYS |
| C3 | **Accounts and data access**<br>DATABASE | One administrator account is shared by the application, the three support staff and the reporting tool. Its password is written in plain text in the code. | Create named accounts with limited rights, change the password and take it out of the code, log every access. | Every access to the 12,400 customer records is named and logged, as the data protection authority recommends. | 1<br>30 DAYS |
| C4 | **Invoicing rebuild**<br>ONGOING PROJECT | The rebuild of the invoicing module has been running for seven months without written requirements. Delivery has slipped twice. | Write the requirements in two weeks, then split the rebuild into three batches delivered and approved one by one. | A date and a budget for each batch, and a first batch live before the end of the quarter. | 1<br>30 DAYS |
| C5 | **Server versions**<br>HOSTING | The server runs PHP 7.2, without fixes since November 2020, and MySQL 5.7, discontinued in October 2023. | Upgrade in two steps on a test server, then switch outside business hours with a rollback ready. | Security fixes apply again and recent components can be installed. | 2<br>90 DAYS |
| C6 | **Backups**<br>HOSTING | A backup runs every night, on the same server as the application. No restore has been tried yet. | Copy the backups to a second hosting provider, restore once a quarter and record how long it takes. | After a server failure, the application restarts with the previous day's orders. | 1<br>30 DAYS |

TANA CORP

# Roadmap

First consolidate, then evolve. Every line has an estimated effort: you decide where to stop.

| PROJECT | SHEETS | REVISION | CONSOLIDATE | EVOLVE |
|---|---|---|---|---|
| Order management | 3/4 and 4/4 | A | 40 days | 58 days |

| ORDER | ACTION | SOURCE | EFFORT | DEADLINE |
|---|---|---|---|---|
| CONSOLIDATE · WITHIN 90 DAYS | | | | 40 D |
| 1 | Move the repository to a company account, open three accesses, write the installation guide | C1 | 3 d | D+15 |
| 2 | Create the named accounts, change the password and take it out of the code | C3 | 4 d | D+30 |
| 3 | Copy the backups to a second hosting provider and complete a first restore | C6 | 2 d | D+30 |
| 4 | Write the invoicing requirements and split the rebuild into three batches | C4 | 5 d | D+30 |
| 5 | Write the tests for the twelve pricing and invoicing rules | C2 | 12 d | D+60 |
| 6 | Upgrade in two steps, then switch with a rollback ready | C5 | 14 d | D+90 |
| EVOLVE · WITHIN A YEAR | | | | 58 D |
| 7 | Customer portal: online tracking of orders and invoices | E1 | 25 d | M+6 |
| 8 | Order entry on tablets for sales staff on customer visits | E2 | 18 d | M+9 |
| 9 | Automatic reading of purchase orders received by email | E3 | 15 d | M+12 |
| | Total effort | | 98 d | |

## 90-day review

At 90 days, we go through the roadmap with you. This is what should be in place:

- The code is in a company account, with three accesses
- Every access to the data is named and logged
- A full restore has been completed
- The first invoicing batch is live
- The twelve pricing rules are covered by tests
- The server runs on supported versions

The year's evolutions are decided at that point, one by one.

## Reading the roadmap

**Priority 1**
To start within 30 days: what everything else depends on.

**Priority 2**
To address within 90 days: what makes changes safer and faster.

**Effort**
In working days. It is an estimate, confirmed by a quote before each batch.

## What an assessment contains

- Sheet 1/4 · Map of the application and its hosting
- Sheet 2/4 · Findings
- Sheet 3/4 · Costed roadmap
- Sheet 4/4 · 90-day review

Ten working days for one application and its hosting. Fixed fee from €10,000 excl. VAT, deducted from the first invoice if you entrust us with the roadmap.

## What about your software?

This example describes a fictional company. To draw up yours, start with a free 30-minute call, with no commitment.

+33 4 78 29 69 39
contact@tanacorp.com

tanacorp.com/en →