

Legacy App X-ray: dealer warranty system

Microsoft Access (.accdb), built 2003–2012 · Public open-source app, personal data replaced with test values

Technical demonstration, not a client engagement. The app is ob080270/Husqvarna-Warranty on GitHub (MIT licence). Nobody commissioned this report, and Legacy X-ray has no connection with the app's author, its users, or the brand in the repository's name. Every finding was checked in the database file; the ones marked *confirmed* were reproduced with the app running. A real X-ray follows this format and is produced two business days after we accept the app.

1. Executive summary

This app runs the warranty desk of a motorcycle dealer's service department: customers and their bikes, repair orders ("acts"), warranty claims with part and labour lines, warranty tags for returned parts, and emails to the warranty administrator. It still works, on 32-bit Access with Outlook installed, and it holds about 45 business rules that exist nowhere else.

Verdict: worth fixing, and small enough to rebuild safely. Three defects are confirmed, one of which prints the wrong parts on warranty tags today. Ten more are present in the file but their consequences weren't reproduced in this data. Two behaviours need a decision from the business rather than a fix. The whole app is about 45 rules, 8 forms and 3 reports, which is our smallest size class.

FINDING	STATUS	WHAT IT COSTS
Warranty tags always print claim #15's parts	Confirmed	Wrong parts tagged for return: 9 of 21 claims in the data
Exchange-rate table is a link to a file on one PC	Confirmed	No euro rate on any other machine
That file holds every rate divided by 100	Confirmed	New repair orders default to 0.435 per euro instead of 43.51 wherever the link works
Ten further defects: 32-bit floats, totals across orders, rounding, audit trail, numbering, 2-digit years, no line key, three input-loss bugs	Potential	Each explained in section 4
Reimbursement entry; lowercase codes	Decision	Business choices, not defects

Recommended path: rebuild as a web app, with the three confirmed defects and the input-loss bugs fixed, and the business deciding the rest one by one. The repair path (fixing the confirmed defects inside Access) is also viable and is listed in section 8.

2. At a glance

ITEM	FINDING
App	Dealer warranty claims and repair orders, Microsoft Access (.accdb), 2003–2012
Size	8 forms, 25 queries, 20 tables plus 1 linked file, 3 reports, 1 macro, about 2,900 lines of VBA (149 objects)
Data in the copy	14 repair orders, 21 warranty claims, 93 part and labour lines, 22 customers, 24 motorcycles
Business rules	About 45 live rules, each catalogued with where it lives in the app (excerpt in section 5); 9 interface rules; 7 functions in code nothing calls
Findings	3 confirmed defects, 10 potential, 2 decisions; plus 3 usability bugs and 5 defects in dead code
Runs only on	32-bit Access (a 32-bit tree control), with Outlook installed (email drafts)
Depends on	A linked Excel workbook for exchange rates; Outlook; four hard-coded email addresses
Size class	Smallest: up to about 50 rules, 10 forms and reports, one database

3. What it does

Staff record customers and their motorcycles, open a repair order per visit, and file one or more warranty claims against it. Each claim carries part and labour lines; labour is priced at the order's hourly rate and parts at dictionary prices. Defect codes come from a 24-group tree with Russian and English names.

The app prints warranty tags for returned parts and claim forms for the manufacturer, converts euro reimbursements to hryvnia at the order's exchange rate, and prepares Outlook emails to the warranty administrator at each step: a motorcycle activated, a new claim filed, a motorcycle not found, a claim entered in the manufacturer's system.

4. Findings

Confirmed reproduced with the app running, consequence shown · **Potential** present in the file, consequence not reproduced in this data · **Decision** behaviour the owner should choose to keep or change.

#	FINDING	WHAT IT COSTS	EVIDENCE	IN A REBUILD
1	Confirmed Warranty tags always print claim #15. The claim number is hard-coded in the tag query (qsBirka).	Wrong parts tagged for return. 9 of 21 claims have parts ticked for a tag; all would print claim 15's.	The query, run on the shipped data; the print preview	Tags for the claim on screen

2	Confirmed The exchange-rate table (EUR_Cur) is a link to Currency.xls on the original developer's PC.	On any other machine the link fails, so a new repair order has no euro rate and converted reimbursements are blank.	Link path in the file; the error when the app runs elsewhere	Rates stored in the database, maintained on a settings screen
3	Confirmed That workbook's "Export Rate" column, the one the app reads, is the published rate divided by 100 (euro 0.435093 against 43.5093).	Where the link works, every new repair order defaults to a rate 100x too small. A wrong reimbursement was not reproduced: see finding 14.	Reproduced with the link pointed at the same workbook in our test machine	Rates stored as published
4	Potential Every price, rate and quantity is stored as a 32-bit float.	Stored amounts drift from what was typed (a printed total of 10,433.48 is held as 10,433.4801828861). No wrong total found in this data.	Table definitions; totals recomputed by Access	Exact decimals
5	Potential Repair-order totals add lines from any other order with the same number (orders are keyed by number <i>and</i> date, totals filter on number only).	Wrong totals once an order number repeats on another date. None repeat in this data.	Form expressions	Totals per order
6	Potential Screen and printout round differently (round the sum vs. sum the rounded lines).	Screen and paper can disagree by 0.01. They agree on every shipped claim.	Form and report expressions	One rounding rule, the printed one
7	Potential The audit field records the Access user, which is "Admin" once user-level security is gone.	Nobody can tell who marked a claim as filed. Older records still show real users: the trail broke when the app was converted to the newer file format.	Data: 2 real usernames on old records	The signed-in user
8	Potential Claim and customer numbers are "highest + 1".	Two people adding at once get the same number.	Form defaults	Database sequences
9	Potential Dates are typed with 2-digit years.	From 2030, a typed "30" becomes 1930. Birthdays before 1930 can't be entered.	Input masks	4-digit years
10	Potential Part and labour lines have no primary key.	Duplicate lines can't be prevented or told apart.	Table definition	Unique line IDs
11–13	Potential Three input-loss bugs: ticking a tag on a labour line undoes every unsaved edit on that line; an unknown defect code erases the description; clearing a reimbursement leaves the old converted amount.	Lost typing; a stale amount. (The third can't happen through the shipped screens: see 14.)	Form code in the database file	Reject only the tick; keep the description; clear both amounts

14	Decision Nobody can type a reimbursement on the repair-order screen: the field is disabled. Yet all 93 lines in the data carry one.	Reimbursements are entered some other way. Who enters them, and where, is a question for the business.	Form property; confirmed running	Kept read-only until decided
15	Decision Codes are compared without regard to case; a quarter of the defect categories and 7 of 21 claims carry lowercase codes.	Nothing today. Changing it would split existing data.	Data	Keep case-insensitive matching

Also found: the repair-order form opens editable when its own code says it should be locked (confirmed running); in the form code, a crash when searching for a claim that doesn't exist and an empty warning box after every customer save (potential). Five more defects sit in functions nothing calls, including the amount-in-words routine.

5. Business-rule catalogue (excerpt)

The full catalogue lists every live rule in plain language with the object it lives in. On a real project you sign it, and it becomes the acceptance matrix. A selection:

RULE	TYPE	PLAIN LANGUAGE	WHERE IT LIVES
HQ-006	Search	A VIN search under 17 characters is an ends-with match; exactly 17 is an exact match; longer is refused	frCust: swFindMoto
HQ-007	Search	A plate search under 8 characters is a contains match	frCust: swFindMoto
HQ-012	Notify	Saving a new motorcycle with both a VIN and a warranty start prepares the "Warranty Activation" email	sfVhc: Form_AfterInsert
HQ-025	Notify	A VIN entered on a repair order that isn't on file prepares the "Not Found in Database" email to Sales, reminder in 2 hours	frAP: apVIN_AfterUpdate
HQ-028	Notify	Claims added in an editing session are listed in one "New Warranty Claim(s)" email when editing is turned off	sfC1m.Tag, GlobalFn.CheckC1mTag
HQ-030	Lookup	A typed conclusion is uppercased; a new defect code offers to save it to the dictionary, a known one offers to update it	sfC1m: clTechn_AfterUpdate
HQ-039	Calc	Order totals: charge = Σ rate $\times$ quantity; reimbursement = Σ converted lines; balance = reimbursement – charge; all to 2 dp	frAP: apChrg, apRmb, clcBln
HQ-041	Calc	An item number of 0 becomes "Ore" (hours): a labour line with the description "Time"	sfItm: itNr_AfterUpdate
HQ-043	Calc	Entering hours on a labour line sets its rate to the order's labour rate	sfItm: itQ_AfterUpdate
HQ-044	Calc	Reimbursement in hryvnia = euro amount $\times$ the order's exchange rate, kept to 4 decimals; only totals round	sfItm: itRmb_AfterUpdate
HQ-049	Report	A warranty tag shows claim number (5 digits), repair year, model, VIN, engine, mileage, part, defect code, complaint, conclusion and order number; parts only, ticked lines only	rpBirka $\leftarrow$ qsBirka
HQ-055	Data	Claim numbers are one sequence across all orders; customer IDs likewise	sfC1m.clNr, frCust.cstID defaults
HQ-057	Data	Repair orders are listed newest sent-date first, then highest number; the admin screen lists claims highest number first	frAP, frWA record sources
HQ-058	Report	The printed claim total is the sum of each line's cost rounded to 2 dp	rpC1mPrn $\leftarrow$ srDlrFriday

6. Dependencies and exceptions

- **Exchange rates** come from a linked Excel workbook on one PC (findings 2 and 3). Rates are a dependency the app cannot satisfy on its own.
- **Outlook** must be installed: the four notification emails are built through Outlook automation and displayed for the user to send. Recipient addresses are hard-coded in the code.
- **32-bit Office only:** the defect picker uses a 32-bit tree control; the app warns on every open under 32-bit Access and won't run the picker under 64-bit.
- **Data entered outside the screens:** reimbursements (finding 14), and two tables (tblDfSe1, tblCM) no screen or query reads.
- **Not in the app:** no scheduled jobs, no other databases, no external systems beyond Outlook were found. The manufacturer's system ("ARCO") is reached by email only.

7. What breaks, and when

TRIGGER	WHAT HAPPENS
Any day: two people enter claims at once	Duplicate claim numbers (finding 8), and either a rejected save or two claims with one number
Any day: a part is ticked for a tag	Claim 15's tag prints (finding 1)
The app is moved to another PC	No euro rate on new orders (finding 2); the tree control may need registering by hand
An Office update, or a move to 64-bit Office	The defect picker's tree control stops working; 64-bit Access won't run it at all
13 October 2026	Microsoft support for Access 2021 ends (only matters if the app runs on Access 2021; it continues on Microsoft 365 and Access 2024)
1 January 2030	A typed year "30" becomes 1930 (finding 9)

8. Recommendation and the three paths

PATH	WHAT IT INVOLVES FOR THIS APP	FIT
Repair	Inside Access: fix the tag query (1), replace the linked workbook with a rates table (2, 3), add 4-digit date masks (9), guard the find-by-claim crash and the three input-loss bugs (11–13). Leaves the floats, the shared-number totals, the audit trail and the numbering as they are.	Good if the desk is one person and the app otherwise suits them
In stages	Move the 20 tables to a database server (which also fixes the numbering and the audit user), keep the Access screens, then replace screens one at a time.	Good if several people use it at once

Rebuild	A web app with every catalogued rule, the three confirmed defects fixed, each potential defect kept or fixed as the business decides, reimbursement entry decided (14), data migrated with reconciled totals, acceptance tests from recorded behaviour. The acceptance guarantee applies to this path.	Recommended. Nobody can change this app safely, and it only runs on 32-bit Office with Outlook
----------------	---	---

Price: this sample has not been priced as a project. For a real app, the \$1,500 diagnostic establishes the full scope, timeline and fixed price before any build work; rebuilds start at \$6,000 for smaller applications. Repair and staged work are quoted after the X-ray.

9. What we checked, and what we didn't

1. **Read the published code:** 2,837 lines of VBA flagged 15 suspected problems.
2. **Extracted the real database file:** all 149 objects, including the queries and form settings the code depends on. Two suspicions were false alarms (plate search and labour billing both work) and 10 new defects surfaced.
3. **Ran the original in a dedicated virtual machine and recorded it:** Access itself recomputed every total for all 14 orders and 21 claims, and ran the app's own queries, lookups and reports. These recordings are what a rebuild is tested against.
4. **Rebuilt it and compared** (beyond a normal X-ray): findings 3 and 14 surfaced only then.

Not done: no interviews with the people who used the app, so the catalogue is what the code does, not what the business intended; no check of printed layouts against paper; no load or multi-user testing of the original.

10. What the diagnostic adds

A free X-ray stops here. The \$1,500 diagnostic produces the documents a fixed price rests on: the complete rules catalogue for your sign-off (every rule, with dependencies and exceptions listed separately), a keep-or-fix decision for every finding, recordings of the app's behaviour taken by Access itself, the acceptance matrix the rebuild must pass, data notes for the migration, and a fixed price, timeline and exclusions. The fee is credited in full to a rebuild signed within 90 days.