LSoft

A SHORT PRESENTATION OF THE NEW SYSTEM

LsBank on the web

The same banking system. Nothing to install on desks, no desktop-app limits.

Your database, PL/SQL packages and customizations stay as they are. Only the operator's tool changes.

What you put up with

Install on every PC

Each release: IT walks every office for a day, not a minute.

Oracle client on every desk

Each desk connects to the database directly; branches and home need VPN to it.

Windows only

Thin client, Linux laptop, executive's tablet — not supported.

Exchange needs an open window

BankMail and SWIFT run in a form on someone's PC: close it, they stop.

Raw Oracle errors

Operator sees “ORA-20110: LSB-0110-U_LOAN…”, a developer decodes it.

Settings tied to one PC

Switch PCs, and you have to set up columns and forms all over again.

DB activity unseen

You can't see which query is slow or which form loads the DB.

This is not your team's fault. It is how a 20-year-old desktop architecture works.

LSoft

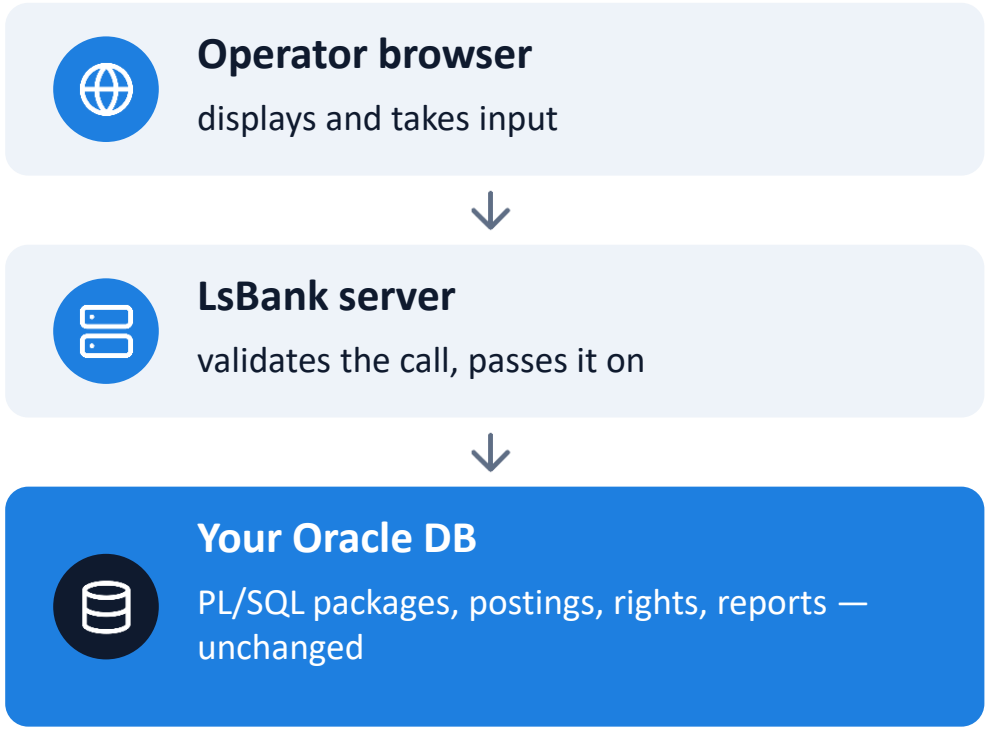
What changes

THEN	NOW
Install and update on every PC	**One browser address; new versions go on the server, operators just reload the page**
Oracle client and direct DB access from every desk	**Only the server connects to the DB; a desk needs just a browser**
Windows only	**Windows, Linux, macOS, thin clients, an executive's tablet**
Exchange needs an open window	**Exchange runs as a server service; closing a tab stops nothing**
Errors in Oracle-speak	**Errors in Armenian, Russian or English, with a reference number**
Settings on one PC	**Grid layouts are stored on the server and follow the user**
DB activity unseen	**Monitoring panel: every DB call with its duration**
DB alerts seen only in the open program	**A bell in the app and a browser notification, even with the tab minimized**

LSoft

The key: we rewrote nothing in your database

Postings, calculations, rights, checks, end of day — all remain in the same Oracle packages your bank has tuned and customized for twenty years.



Same procedures

The new app calls exactly what the old one did. No business rules on our side.

Your customizations stay

Packages changed for the regulator or a product work in the browser too.

Oracle checks access

Operators sign in with their own Oracle account, as today; we keep no user list.

Your reports

Report templates stay in the DB and run as before; layouts can be edited in the browser.

No second source of truth. The database decides — the program shows its answer.

LSoft

How it works

Four tiers; business logic lives only in the last. Anything added later connects to the same server and passes the same checks.

Workstations

- Browser only
- HQ and branches
- Armenian, Russian, English

Front server

- Single external address
- HTTPS
- Windows or Linux

LsBank server

- Sign-in with Oracle account
- Registry of allowed calls
- PDF, Excel, Word reports
- Background jobs, file exchange

Your Oracle DB

- PL/SQL packages unchanged
- Postings, rights, calculations
- Report templates

Next modules connect to the same server

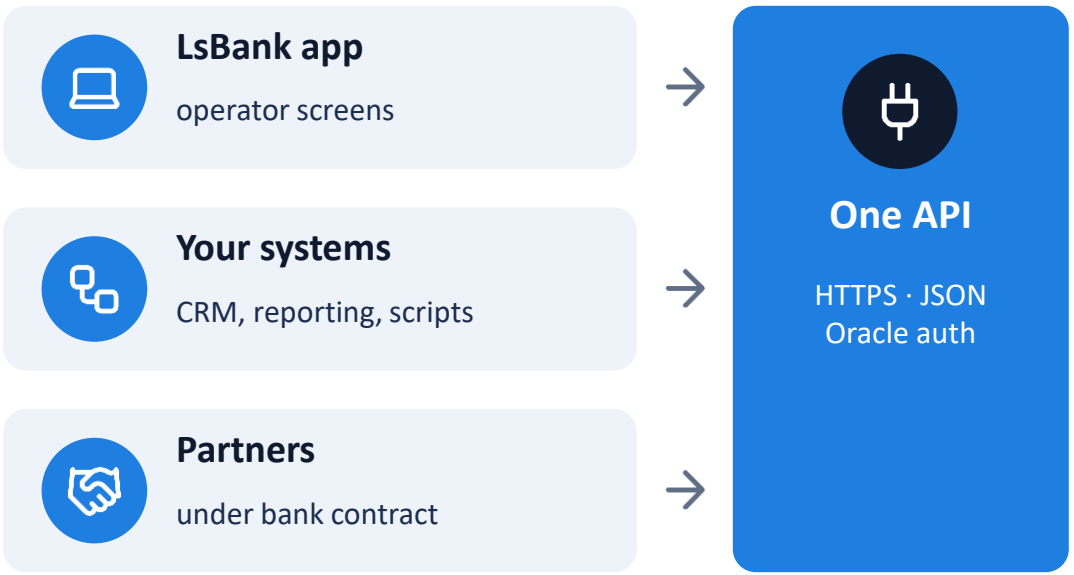
- Internet banking
- Mobile banking
- Partners: tax, transfers, payments
- Cards, SMS, email, alerts

No module goes to the database directly: rights, call log and rate limit are shared by all.

LSoft

One API — for the program and your systems

The program itself works through the same API the bank's systems can use. There is no separate “integration” entry with other rules.



Same sign-in, same rights

Oracle account: what an operator may do, a system under that account may do too.

Plain HTTPS and JSON

Connects from any language or platform; no Oracle driver needed.

Declared calls only

No ad-hoc query or procedure can run from outside.

Documented inside the app

Each call has a sample request and response. No waiting for documentation.

Whatever the program can do, your systems can too — same rights, same log.

LSoft

Proven by postings, not by words

The most critical part of a banking system is the document: it produces postings. So we check not that it “seems to work” but that it matches your current program.

1

Take a doc

A real one, posted by the old program last month.

2

Post the same one

We enter the same document in the new app and post it.

3

Compare postings

Operation, account, currency, amount — line by line.

A match means this document type posts in the browser just as it does today.

Reconciliation runs on the bank's test database by document type, most frequent first. On top of that, every build passes 928 automated checks.

LSoft

Numbers

277

screens (forms) in use

928

automated checks on every build

2,059

bank report templates in use

3 UI languages — Armenian, Russian, English

22 visual themes

4 report engines; Excel without Office installed

Checks: 483 server tests · 421 client tests · 24 build consistency checks. Figures as of 20 November 2026.

LSoft

For the security team

No password kept

Only in server memory for the session: not in the browser, token or logs.

No procedures in browser

It names a key; the server's registry decides what runs and with which rights.

No outside SQL

No endpoint accepts query text from the browser. An unknown key is not executed.

Oracle decides

As today. We built no rights system of our own that would need separate audit.

Workstation logged

The change log shows the operator's PC, not the server everyone goes through.

Session limits

3 hours idle, 12 hours from login; login attempts are rate-limited.

LSoft

Migration with no bank downtime

1

Both programs run

At once, on one database and the same packages. The old one untouched.

2

By branch and by form

One branch tries it for a week; the rest work as before.

3

Roll back any day

The old program stays; the data is shared.

4

Bank's call

The old program goes off when the bank decides it's ready.

No D-Day. Just a growing circle of people working in the browser.

LSoft

What we need

Application server — Windows or Linux	Hosts the website and API; nothing to install on workstations
Oracle 12 or later (client 12.2 on server)	Banks on 11g must upgrade to 12
DB copy for a test environment	Used for posting reconciliation and a branch pilot
One bank IT specialist	Access, exchange folders, certificate, questions on package changes
List of your package and form changes	So nothing is lost: we check these first

The test environment is deployed by scripts in one day — repeatable, and not dependent on one person's memory.

LSoft

Stages

1 · Assessment	2 · Test setup	3 · First branch	4 · Rollout	5 · Switch-off
Package and form changes, exchanges, legacy report formats	On a copy of your DB, postings checked by document type	Operators work in the browser, the old program alongside	Other branches and forms as they are ready	When the bank decides; the DB then moves to Unicode
us + your IT	us	your operators + us	bank sets pace	us + your IT



We name each stage's timeline after the assessment: every bank's customizations differ, and promising weeks unseen would be dishonest.

LSoft

What opens up next

Unicode in DB

Armenian text stops being single-byte: input limits and per-language column splits go away.

Oracle up to 23ai

The DB version is no longer tied to a desktop client.

AI inside your perimeter

Docs from invoice text, similar transactions found during review, attachment search by content.

Move external systems to the API

Those that hit the database directly today: shared rights, log, rate limit.

Branches and remote work

No VPN to the database: a desk needs only a browser.

LSoft

Next step

1

Live demo

1 hour, anonymized DB: login, operating day, document, report, exchange.

2

Assessment

Two weeks: changes, exchanges, forms. Result: plan and timing for your bank.

3

Test setup

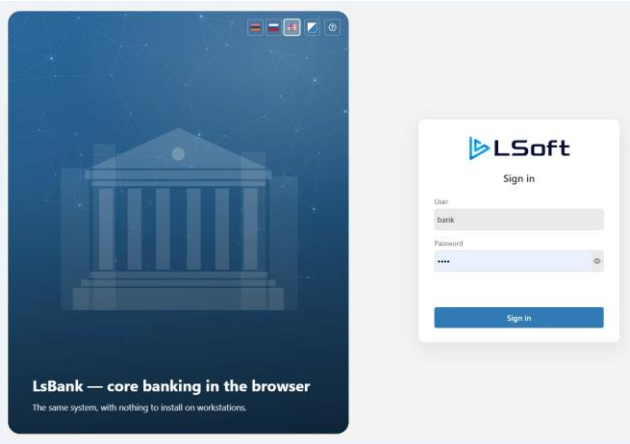
On a copy of your DB, with postings checked on your own docs.

The bank decides after step three — having seen its own documents posted in the browser.

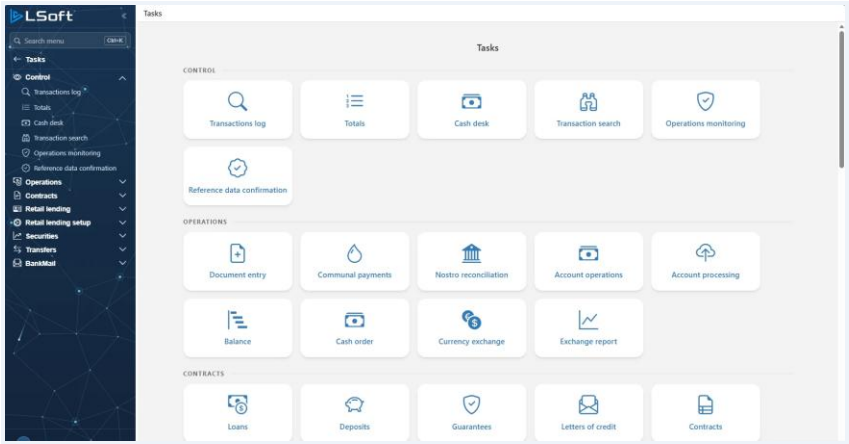
Architecture overview for IT — in the appendix.

LSoft

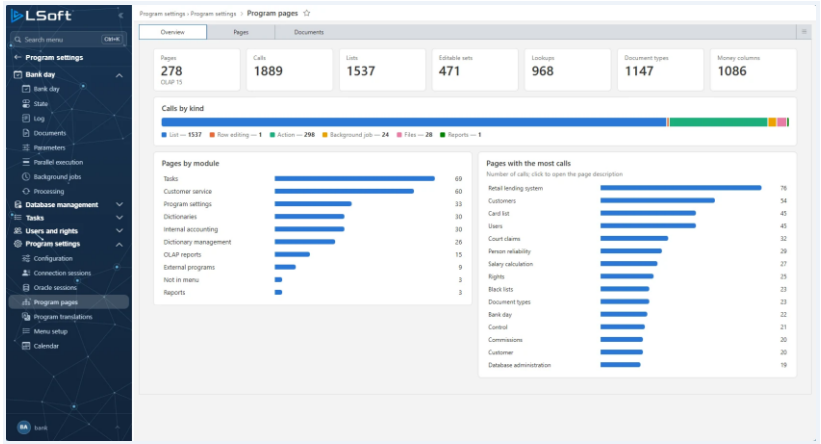
What it looks like



Sign-in



The Tasks module: control, operations, contracts



Program pages: screens, database calls, lookups

Screenshots from a test environment; bank name and customer data are hidden.

LSoft