

AI Project Manager for teamwork

From chat and calls — straight to the board. No manual transfer.



Team KYSS

Problem: loss of focus

The growing pace of decision-making and product development requires full focus on the work itself, not on putting it onto a board.

1

Work has moved into communication

Decisions appear in chats, voice messages, and short meetings faster than teams can document them.

2

The gap between “discussed” and “done”

After a call or chat, employees have to switch context instead of moving straight to execution.

3

Manual documentation does not scale

Task volume is growing; information sources are scattered and the information itself is fragmented.

Teams need a bridge: every conversation should automatically become a task, meeting notes, and a clear summary.

Komandus — a bridge between conversation and board

The service analyzes live speech, extracts the task, deadline, and assignee, creates meeting notes, and sends the result to the board.

For example:

28

total tasks

87%

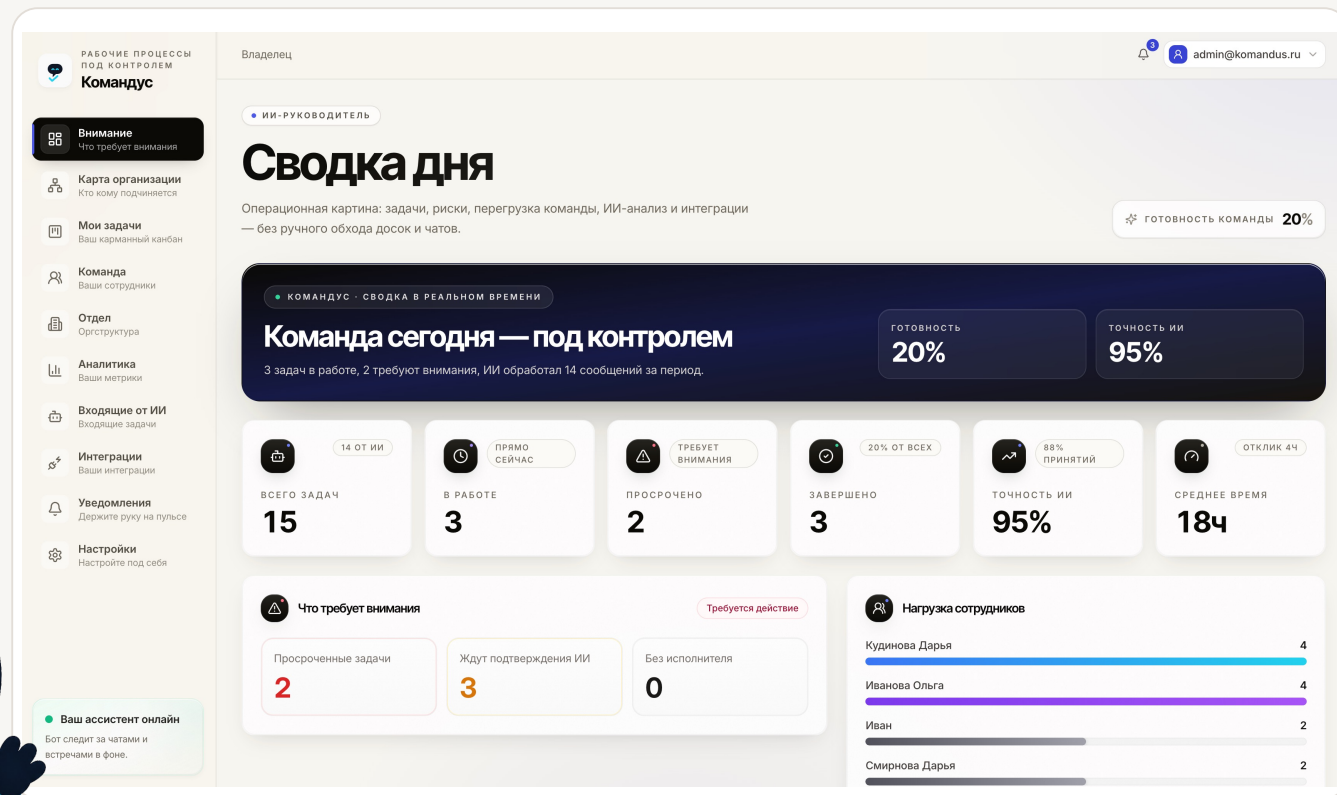
AI accuracy

18ч

average time

What the bot does

- reads messages
- listens to meetings
- creates notes + tasks
- shows risks



Long meeting notes without manual capture

Komandus understands natural speech: it turns a meeting into notes, tasks, deadlines, and assignees.

Not a command — natural speech

**“Pavel — API by tomorrow;
Ivan — board screen by evening”**

AI identifies what to create
on the board.

Tasks

Owners

Dates

Confirm

1

Speech → text

ASR transcribes Russian speech and voice messages.

2

Text → notes

The LLM creates a concise summary: what was decided
and what comes next.

3

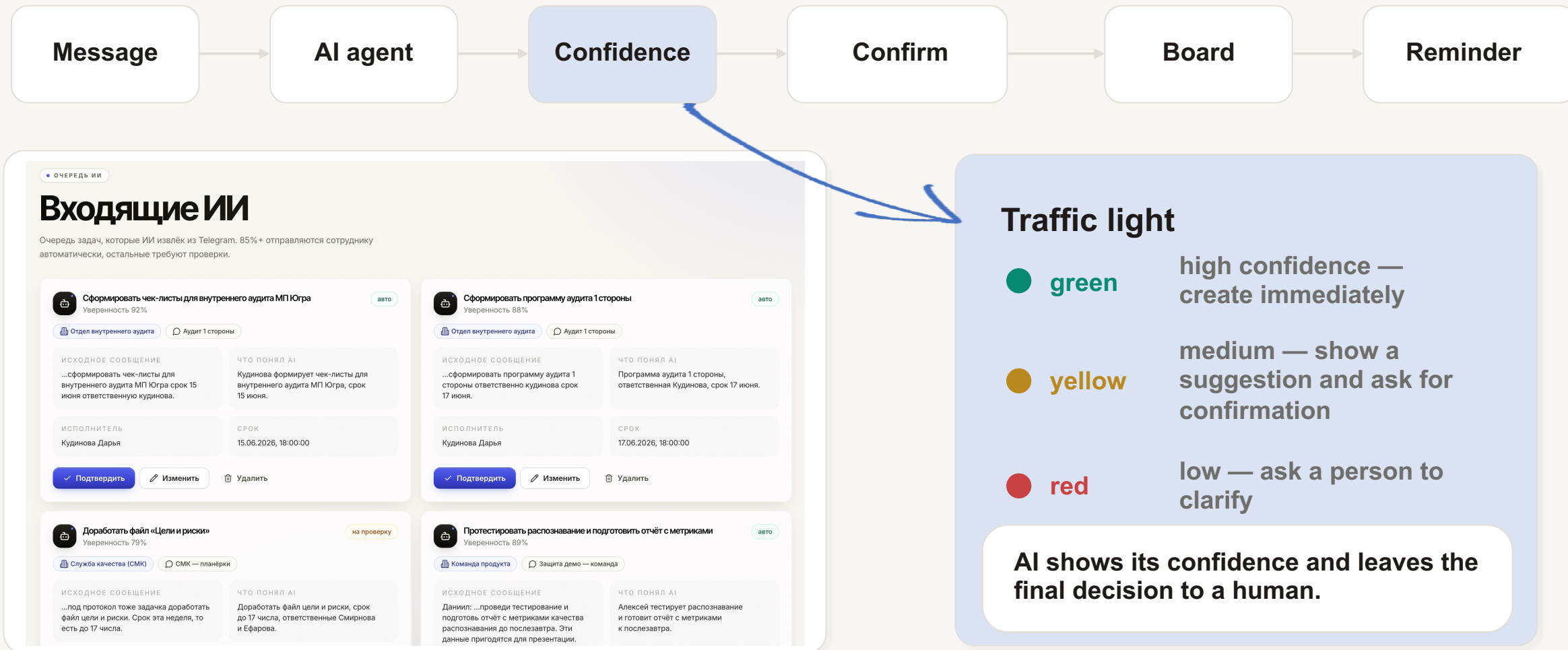
Notes → tasks

Deadlines, assignees, and checks are moved to the
board.

**The meeting ends — the notes and tasks are
already ready for confirmation.**

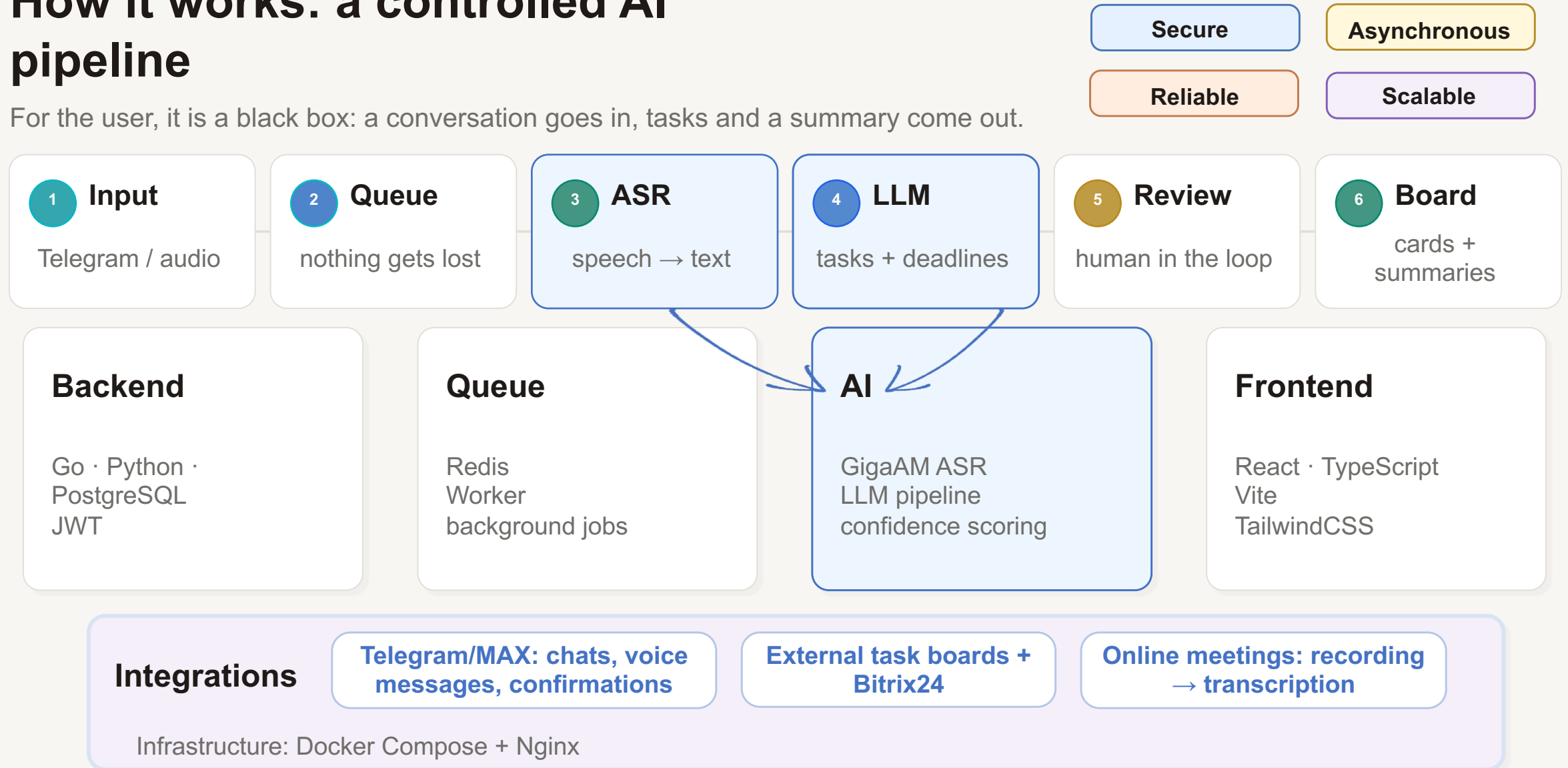
Client scenario: from phrase to card

The “confidence traffic light” shows where a task can be created automatically and where a human is needed.



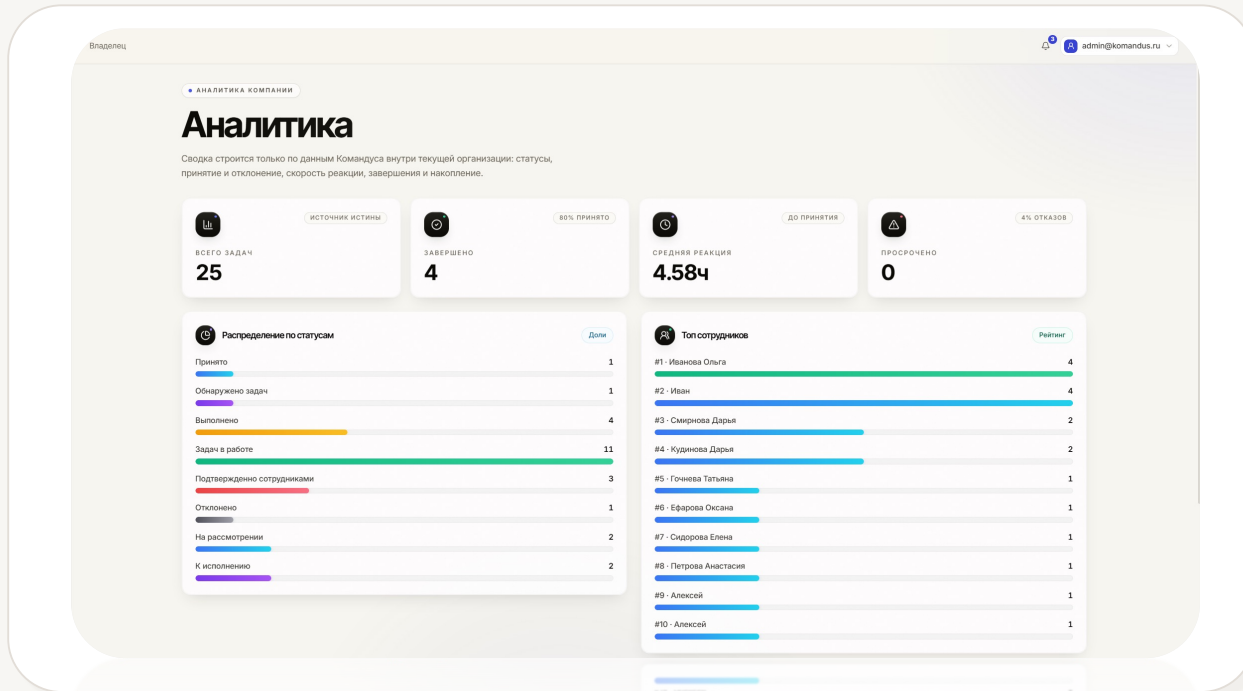
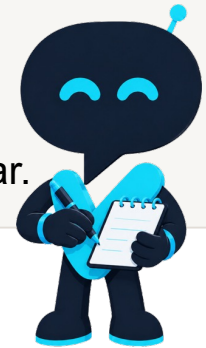
How it works: a controlled AI pipeline

For the user, it is a black box: a conversation goes in, tasks and a summary come out.



Statistics = achievements

Transparent and informative: after a meeting and at the end of the week, results, risks, and contributions are clear.



What you see without reports

- Achievements** closed tasks and team contribution
- Overdue tasks** what requires action
- Workload** who is overloaded before deadlines slip
- Productivity** team dynamics by day and week

The weekly summary arrives automatically: managers get the big picture, employees see clear progress.

MVP is ready

A conversation turns into a board task.

What already works

- ✓ Telegram bot in a team chat
- ✓ task extraction from text
- ✓ voice and long meeting transcription
- ✓ task confirmation
- ✓ cards on the board
- ✓ deadline reminders
- ✓ dashboard and analytics

MVP timeline

- 1 June 5** foundation: bot, API, board
- 2 June 8** AI pipeline and confirmation
- 3 June 9** voice, analytics, integrations

The MVP is ready for demo: the scenario runs from an incoming message or recording to a board task.

Development roadmap

Concrete dates, features, and business-pilot impact after the MVP.

June 12

- **Meeting by link**
- recording and transcription without manual upload
- impact: notes right after the call

June 19

- **Speaker recognition**
- who said it → who gets assigned
- impact: fewer assignment errors

July 4

- **Corporate pilot**
- on-premise + integrations
- impact: launch inside the company perimeter

Value

Estimated pilot impact for a 50-person team: less manual work, clearer control, faster execution of agreements.

For business

≈150–300K RUB/month

- savings on manual reports, meeting notes, and task transfer
- up to 10–15% fewer losses from forgotten tasks
- pilot payback — 1–2 months

For employees

3–5 h/week

- less manual follow-up after meetings
- deadlines and assignees are captured automatically
- tasks appear where the team already communicates

Estimate for the pilot: 50 employees, 2 meetings per day, 5 working days per week.

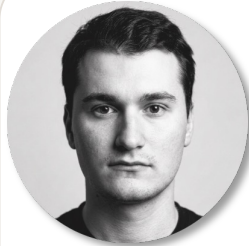
Team KYSS

Roles are aligned with the working prototype: backend, frontend, ASR, LLM, and product assembly.



Ex-Middle AI Engineer at Sber
Alexey Podkolzin

Captain / ML + ASR Engineer



Senior Go Engineer at MTS
Artem Vasilyev

Tech Lead /
Technical Product Manager /
Backend Go Developer



Pavel Yudenkov
LLM Engineer



Ivan Rzhevsky
Frontend Developer



Daniil Maslov
Python Developer



Komandus

AI Project Manager for teamwork

**From chat and calls — to the board.
No manual transfer.**

**Thank you for your attention.
Ready to answer your questions.**



GitHub