Softwareentwicklung und Automatisierung. Moskau, remote

filippmikheev.com Telegram @filly_tt feelmetttt@gmail.com

Filipp Mikheev

Zehn Jahre habe ich die Abschlüsse großer Unternehmen geprüft. Heute baue ich Systeme, in denen Geld und Daten stimmen und Beträge nur sieht, wer sie sehen darf: Unternehmenssoftware und CRM, Umzüge von Altsystemen, KI im Arbeitsalltag, Websites.

Von der Prüfung zur Entwicklung

Ab 2017 habe ich in großen Unternehmen geprüft, ob das Geld aufgeht, wer welche Zahlen sieht und was beim Datenumzug verloren geht. Heute prüfe ich genau das in meinem eigenen Code vor dem Go-live.

Ausbildung
MEPhI (Moskau), Wirtschaftssicherheit, Abschluss mit Auszeichnung, 2017

Weiterbildung
EPAM, Systemanalyse, 2024

Wirtschaftsprüfung und interne Kontrolle

Prüfungen nach IFRS und russischer Rechnungslegung (RAS) bei großen Industrie-, IT- und Staatsunternehmen. Bewertung interner Kontrollen nach COSO und SOX, operative Prüfung, forensische Untersuchungen, Berichte an Finanzvorstände und Prüfungsausschüsse. Teams mit bis zu zehn Personen geleitet.

- 2017-2018 — **FBK Grant Thornton** Prüfungsassistent
- 2018-2019 — **Baker Tilly Russia** Erfahrener Prüfungsassistent
- 2019-2020 — **Produktionsunternehmen** Interner Revisor
- 2020-2021 — **FBK Grant Thornton** Prüfer
- 2021-2022 — **ADE Professional Solutions** Projektleiter Prüfung
- 2023 — **Haier** Senior-Spezialist interne Kontrollen
- 2023-2024 — **FRESH AUTO** Senior Auditor

Entwicklung

Kommerzielle .NET-Entwicklung in einem Produktteam und seit 2024 eigene Projekte, in denen ich die Entwicklung verantworte: vom ersten Kundengespräch bis zu Betrieb und Störungsbereitschaft.

- 2024-2026 — **Altenar** Backend-Entwickler, C# und .NET
- seit 2024 — **Selbstständig** Kundensysteme: Unternehmenssoftware, CRM, Umzüge, KI. Argos. «Сайты и точка».

Woran ich gerade arbeite

Für Kunden

Automatisierung aus einer Hand

Das typische Bild: Der Prozess läuft bereits, hängt aber an Chats, Tabellen und dem Gedächtnis einzelner Mitarbeiter, und ein Fehler bei Geld oder Daten kostet mehr als die Entwicklung selbst. Ich begleite das Projekt vom ersten Gespräch bis zum laufenden System und bleibe danach für den Betrieb da.

Unternehmenssoftware oder CRM für Ihren Prozess
Reinigung: Büro-CRM, 6 Rollen

Umzug von Altsystemen und Tabellen
11.055 alte Adressen funktionieren weiter, Rollback mit einem Befehl

Integrationen
Telefonanlage: Kundenprofil vor dem Abheben

KI im Tagesgeschäft
Argos: 500.000 Buchungszeilen in 85 s anonymisiert

Vorfallsanalyse und Stabilität
Ursache nach zwei erfolglosen Untersuchungen gefunden

Apps für Mitarbeiter und Kunden
~600 Käufer in zwei Messengern

Websites und Suchmaschinenoptimierung
Antwortzeit für Crawler 8,6 → 0,6 s nach der Reparatur, Lighthouse 98-100 bei Website-Prototypen

Eigenes Produkt

Argos

Ein KI-Assistent für Prüfungsgesellschaften. Die Agenten arbeiten in mehreren Schritten, das Ergebnis wird bei jedem Schritt geprüft, und ein eigener MCP-Server für externe KI-Assistenten protokolliert jeden Aufruf.

Ein Befehl prüft alle acht Pakete

Build	✓
Typprüfung	✓
API-Vertrag	✓
Tests im vollständigen Lauf	23.875 ✓

16	133.000	173.000
Prüfungen des Buchungsjournals	Zeilen Code	Zeilen Tests

argos-audit.ru ↗

Studio

Сайты и точка.

Der Name heißt etwa „Websites, Punkt“. Das Studio zieht Onlineshops um und gestaltet sie neu. Beispiel: by-home.ru, ein Shop mit 13.000 Produkten: alle 11.055 alten Produktadressen funktionieren weiter, die erste Bestellung auf der neuen Website kam am nächsten Tag.

sitesstudio.ru ↗

Zahlen und woher sie stammen

Wie in der Prüfung steht bei jeder Zahl ihre Quelle. Alle stammen aus den unten beschriebenen Projekten.

Automatische Tests, Stand 9.11.2026

Onlineshop	**1.023**
Lebensmittellieferung	**752**
Reinigungs-CRM	**1.990**
App fürs Landhaus	**70**
Argos	**23.875**
Fünf Systeme insgesamt	27.710 ✓

Projekte

Sieben Projekte aus dem Jahr 2026: vorher, umgesetzt, Ergebnis. Kundennamen, Domains und Daten nenne ich nicht, außer by-home.ru, das mit Zustimmung des Betreibers gezeigt wird.

● **Im Betrieb** Onlinehandel, 2026

by-home.ru: Reparatur eines Shops mit 13.000 Produkten und Umzug auf eine neue Plattform

Zuerst wurde die alte Website repariert: Eine Produktseite lud in 0,5 statt 11,6 Sekunden, Crawler bekamen ihre Antwort in 0,6 statt 8,6 Sekunden. Danach zog der Shop auf die neue Plattform um: alle 11.055 alten Produktadressen funktionieren weiter, die alte Website blieb für ein Rollback mit einem Befehl erhalten.

Vorher

Eine Plattform ohne Herstellersupport. Die Schadsoftware kehrte innerhalb eines Monats dreimal zurück, zwei frühere Untersuchungen fanden nichts. Eine Produktseite brauchte 11,6 Sekunden zum Laden.

Umgesetzt

- Die Prozessüberwachung zeigte, dass der Zugang am Webserver vorbei über einen offenen Dienstport lief. Der Port ist geschlossen, die Korrektur übersteht einen Neustart.
- Parallel eine neue Plattform aufgebaut; alle 11.055 alten Produktadressen öffnen das Produkt oder, wenn es nicht mehr verkauft wird, seine Rubrik.
- Suche: Crawler machten 62 % des Traffics aus und warteten 8,6 Sekunden auf eine Antwort. Jetzt sind es 0,6 Sekunden, und jede Adresse nach dem Umzug wurde eins zu eins mit der alten abgeglichen.

Technik

Python, FastAPI, SQLAlchemy, PostgreSQL, Alembic, React, TypeScript, Docker Compose, nginx, iptables, fail2ban, PrestaShop / MariaDB

13.532
Produkte übernommen

11.055
alte Adressen funktionieren

13
Tabellen statt 373

1.023
Tests

by-home.ru ↗

○ **Prototyp beim Kunden** Baustoffe, 2026

Website eines Herstellers: vom Audit zu einer Website, die das passende Produkt selbst auswählt

Lighthouse mobil: alte Website 57-92, neue 98-100 in allen Kategorien. Die Änderungswünsche des Kunden sind umgesetzt, als Nächstes folgt der Umzug in sein CMS.

170
Adressen im Audit gecrawlt

684
Profile im Rechner

Vorher

93 Seiten und kein einziges Anfrageformular: E-Mails landeten in den Postfächern von sieben Niederlassungen, Ziele in der Webanalyse waren nicht eingerichtet, also wusste niemand, wie viele Anfragen die Website bringt. Die Sitemap enthielt 9 von 93 Adressen, 21 Links waren defekt, 1.785 von 1.912 Bildern hatten keinen Alt-Text, 61 Titel kamen doppelt vor. Auf der Berechnungsseite stand statt einer Berechnung „wenden Sie sich an unsere Fachleute“.

Umgesetzt

- Audit von außen, ohne Zugänge: 170 Adressen gecrawlt, Lighthouse-Messungen, fünf Wettbewerber analysiert, ein Plan in fünf Schritten.
- Eine neue Website: Auswahl von Produkt und Schichtdicke für die geforderte Feuerwiderstandsklasse samt Zertifikatsnummer, ein Verbrauchsrechner über den Stahlprofilkatalog (684 Profile, gegen zwei Quellen abgeglichen), Anfragen gehen je nach Region an die richtige Niederlassung.
- Eine Zuordnung von 123 alten Adressen: 84 bleiben, 21 erhalten dauerhafte Weiterleitungen, 18 Grußbeiträge wandern in die News, die Regeln sind auf demselben Servertyp getestet. Eine Domainvariante und kanonische Adressen statt Duplikaten.

Außerdem gefunden

Fragen zu den Produktunterlagen und zu den Normtabellen, beides ging zur Prüfung an den Technologen.

Technik

HTML, CSS, JavaScript, Node.js, schema.org, Apache 301 / canonical, Caddy, Docker, Lighthouse, axe

---

● **Im Betrieb** Möbel, 2026

Händlerkatalog in zwei Tagen: 817 Artikel, keiner vergessen

Der Katalog läuft beim Hoster des Kunden. Beim Umzug wurden alle 2.750 Dateien nach Größe abgeglichen.

42
Rubriken geprüft

2
Tage von der Sprachnachricht zum Katalog

Vorher

Ein Händler brauchte einen eigenen Katalog des gesamten Sortiments einer Fabrik, deren Website 42 Rubriken und keinen Export hatte. Der Auftrag kam als zwei Sprachnachrichten.

Umgesetzt

- Die Sprachnachrichten wurden lokal transkribiert. Die Website der Fabrik wurde in zwei unabhängigen Durchläufen gecrawlt, die Vollständigkeit Rubrik für Rubrik geprüft.
- Eine Excel-Tabelle mit allen Merkmalen und einer Zusammenfassung per Formeln. Ein Onlinekatalog: Kollektionen, Katalogansicht mit Filtern, Produktseiten mit Galerie, Bestellformular auf jeder Produktseite. Von der Indexierung ausgeschlossen.

Technik

JavaScript, HTML, CSS, Python, Excel, whisper.cpp, Vercel

○ **Prototyp** Musiker, Deutschland, 2026

Prototyp für die Website eines Konzertmusikers: eine persönliche Einladung für jeden Gast

Sechs Ansichten; jeder Gast bekommt einen persönlichen Einladungslink in seinem Kanal und seiner Sprache.

6
Ansichten

3
Einladungskanäle

Aufgabe

Eine Website auf Deutsch und Russisch für einen in Deutschland lebenden Konzertmusiker. Das Wichtigste: jeden Gast persönlich zu Konzerten einladen.

Umgesetzt

- Jeder Gast bekommt eine Nachricht mit eigenem Namen und eigener Anrede in seiner Sprache („Liebe …“, „Sehr geehrte Frau Dr. …“, „Дорогая …“) über seinen Kanal: WhatsApp, SMS oder E-Mail. Ein persönlicher Link führt zur Antwortseite.
- Strukturierte Daten nach schema.org (MusicEvent), damit Google Datum und Ort direkt in den Suchergebnissen zeigt.

Prüfungen

Das Layout wurde bei 390, 768, 1024 und 1280 px geprüft: kein horizontales Scrollen.

Technik

HTML, CSS, JavaScript, schema.org MusicEvent, WhatsApp / SMS / E-Mail, Lighthouse, axe

Im Betrieb Lebensmittellieferung, 2026

Ein Lieferdienst ist von Chats und Tabellen in ein System umgezogen

Die Auftragsnummern stimmen zwischen Bot und Oberfläche überein, und riskante neue Funktionen bleiben abgeschaltet, bis der Kunde entscheidet.

383
alte Aufträge neu nummeriert

752
Tests

Vorher

Aufträge, Disponenten, Einkauf, Lieferung und Kundenguthaben liefen über einen Messenger und Tabellen. Bot und Admin-Oberfläche zeigten für denselben Auftrag unterschiedliche Nummern.

Umgesetzt

- Eine Büro-Oberfläche und zwei Shop-Bots mit gleichem Funktionsumfang: Telegram und ein zweiter Messenger, dessen API-Client ich von Grund auf geschrieben habe.
- Kundenguthaben werden als Buchungsjournal geführt, ein zweiter Klick bei der Bargeldabrechnung verdoppelt keinen Betrag.

Technik

Python 3.12, FastAPI, SQLAlchemy async, PostgreSQL, Redis, aiogram, React 18, TypeScript, Ant Design, TanStack Query, Docker Compose

---

An Kunden übergeben

Reinigungsdienste, 2026

Büro-CRM und App für Einsatzkräfte, in denen Beträge nur sieht, wer sie sehen darf

Das System hat vier Abnahmerunden bestanden. Büro und Einsatzkräfte sehen nur die Beträge, die ihnen zustehen, und das Kundenprofil erscheint, während das Telefon noch klingelt.

1.990
Tests

~550
Kunden übernommen

Vorher

Einzelaufträge und Vertragsobjekte lagen in Tabellen. Standard-CRMs passten weder zur Abrechnung mit den Einsatzkräften noch zur Rechtevergabe bei Beträgen.

Umgesetzt

- Der Zugriff auf Beträge ist serverseitig gesperrt, einschließlich aller Umgehungswege: Excel-Exporte, PDF-Angebote, Rechner, App für Einsatzkräfte.
- Zwei Jahre Daten und rund 550 Kunden übernommen, Anbindung an die Cloud-Telefonanlage in beide Richtungen.

Technik

Python, FastAPI, PostgreSQL, React, TypeScript, Ant Design, Telegram Mini App, API der Cloud-Telefonanlage, Excel / PDF

Im täglichen Einsatz

Immobilienverwaltung, 2026

Landhaus: eine App, die Belege liest und Sprache versteht

Die Spracheingabe wird auf dem Server erkannt und funktioniert auf iPhone und Android gleich, Benachrichtigungen sind auf einem echten Gerät geprüft. Die App wird täglich genutzt.

70
Tests

0
App-Stores nötig

Vorher

Eigentümer und Verwalter führten Ausgaben, Aufgaben und Einkäufe in Chats und auf Papier.

Umgesetzt

- Eine App, die ohne App Store als Symbol aufs Handy kommt, mit 3 Rollen.
- Belege laufen durch eine Kette von Erkennungsdiensten mit lokaler Reserve, die Positionen werden automatisch dem offenen Auftrag zugeordnet.

Technik

Python, FastAPI, PostgreSQL, PWA, Web Push, Tesseract, Whisper, Docker Compose

Wie Prüfungserfahrung Ihr Risiko senkt

Drei Regeln aus der Prüfung habe ich in den Code übernommen. Zu jeder gehört ein Fehler aus einem echten Projekt.

Geld stimmt auf Buchungsebene

Salden entstehen aus der Buchungshistorie. Jeder Betrag lässt sich bis zur Ursprungsbuchung zurückverfolgen, und ein zweiter Klick erzeugt nie eine zweite Zahlung.

aus der Praxis

Die Abgabe von Bargeld verdoppelte die Schuld eines Mitarbeiters, weil das Vorzeichen vom Button kam. Vor dem Go-live behoben.

Rechte werden auf dem Server durchgesetzt

Einen Button auszublenden reicht nicht. Ich prüfe die Umgehungswege: Exporte, Druckformulare, Rechner, den Modus „Ansicht als Rolle“.

aus der Praxis

Der Rollenmodus tauschte nur das Menü, Beträge kamen weiter mit den Rechten des Managers. Auf dem Server geschlossen.

Migrationen werden Zeile für Zeile abgestimmt

Vor dem Umzug: Sicherung und getestete Wiederherstellung. Danach: Anzahl und Summen mit der Quelle abgestimmt. Eine Abweichung stoppt den Go-live.

aus der Praxis

Beträge aus Tabellen kamen als 6094.999999999999 an und brachen den Wiederholungslauf ab. In der Migration abgefangen.

Technik

Backend	Frontend	Bots und KI	Infrastruktur
Python	React	aiogram	Docker Compose
FastAPI	TypeScript	Telegram Mini App	nginx, Caddy
SQLAlchemy, Alembic	Ant Design	LLM-Agenten	Linux
PostgreSQL	Vite	MCP	iptables, fail2ban
Redis	TanStack Query	Whisper	pytest
C# / .NET	PWA	Tesseract	Lighthouse, axe

Schreiben Sie mir

Per Telegram oder E-Mail. Geht es um eine Aufgabe, helfen ein paar Zeilen, damit das erste Gespräch konkret wird:

- wie der Prozess heute läuft und womit;
- wie viele Personen und Vorgänge pro Tag oder Monat;
- wo Geld, Zeit oder Kontrolle verloren gehen.

Ich antworte mit einem Lösungsweg, einer ersten Schätzung und den Risiken, die ich schon sehe.

Telegram
@filly_tt

E-Mail
feelmetttt@gmail.com

filippmikheev.com

© 2026 Filipp Mikheev

Keine Tracker und keine Anfragen an Dritte auf dieser Website.