

# "תוכנית מערכת ניהול תקין לעמותות - "מהטפת"

---

## Context

מלאה לניהול תקין של עמותות בישראל עם אוטומציה של 100%. המערכת מספקת שירות SaaS בנית פלטפורמת וכוללת: ניהול כספי, תורמים, ועד הנהל, מתנדבים, דיווחים רגולטוריים, ומגוון (Multi-tenant) למספר עמותות אוטומציות מלא. הפרויקט מתחילה מאפס בתקיה ריקה.

---

## Tech Stack

שכבה	טכנולוגיה
Framework	Next.js 14+ (App Router), TypeScript
UI	Tailwind CSS + shadcn/ui (RTL)
Database	PostgreSQL + Prisma ORM
Auth	NextAuth.js (Auth.js v5)
State	React Query (TanStack Query) + Zustand
Jobs/Queue	BullMQ + Redis
File Storage	S3-compatible (MinIO dev / AWS S3 prod)
PDF	@react-pdf/renderer + Puppeteer
Email	Nodemailer + React Email templates
Monorepo	Turborepo

---

## מבנה פרויקט (Project Structure)

```

mahatefet/
├── turbo.json
├── package.json
├── .env.example
└── docker-compose.yml      # PostgreSQL, Redis, MinIO

└── apps/
    ├── web/                # Next.js App (Frontend + API)
    │   └── app/
    │       ├── (auth)/      # Login, Register, Forgot Password
    │       ├── (public)/    # Landing page, Pricing
    │       ├── (dashboard)/ # Main app (protected)
    │       │   └── layout.tsx # Dashboard shell + sidebar
    │       └── page.tsx     # Overview dashboard
    └── finance/             # ניהול כספי
        └── donors/         # ניהול תורמים

```

```

    └── compliance/ # ניהול תקין
    └── board/      # ועד מנהל
    └── volunteers/ # מתנדבים
    └── automation/ # אוטומציה
    └── reports/   # דוחות
    └── integrations/ # אינטגרציות
        └── settings/ # הגדרות ארגון
    └── api/v1/     # API Routes
    └── components/ # UI Components
    └── lib/         # Utils, hooks, services
    └── middleware.ts # Auth + Tenant + Locale

    └── worker/      # Background Job Processor
        └── src/
            └── workers/ # BullMQ workers
            └── jobs/    # Job definitions
            └── index.ts # Worker entry

    └── packages/
        └── db/          # Prisma schema + client
            └── prisma/
                └── schema.prisma
                └── seed.ts
                └── migrations/
            └── src/
                └── client.ts # Prisma client singleton
                └── tenant.ts # Multi-tenant Prisma extension

    └── shared/
        └── types/       # Shared types, validators, constants
        └── validators/ # Zod schemas
        └── constants/

    └── integrations/ # External service adapters
        └── src/
            └── base-adapter.ts
            └── invoice-green/
            └── icount/
            └── tranzila/
            └── cardcom/
            └── whatsapp/
            └── google-workspace/
            └── tax-authority/
            └── registrar/

```

## מודלים מרכזיים - Database Schema

Multi-Tenancy: Row-Level Security

query מוסיף פילטר אוטומטי לכל טבלה מכילן `organizationId` - Prisma Client Extension

## (מודלים עיקריים (~40 טבלאות):

### ליבת:

- עמותה (שם, מספר עמותה, כתובת, הגדרות - Organization)
- משתמש (אימייל, סיסמה, שם - User)
- Membership (תפקיד + User בOrganization (ADMIN/MANAGER/ACCOUNTANT/VIEWER))
- AuditLog - לוג מלא של כל פעולה במערכת

### כספי:

- חשבוניות (מספר רציף, סטטוס, סכום - Invoice)
- קבילות (כולל סעיף 46 - Receipt)
- העברות בנקאיות - BankTransfer
- תקציב שנתי - Budget
- BudgetLine - שורות תקציב
- תנועות כספיות - FinancialTransaction
- מספר רציף חוקי ללא פערים - DocumentSequence

### תורמים:

- פרופיל תורם - Donor
- תרומות - Donation
- הוראות קבוע - RecurringDonation
- קמפיינים - Campaign
- DonorCommunication - תקשורת עם תורמים

### עד מנהל:

- ישיבות ועד - BoardMeeting
- פרוטוקולים - MeetingProtocol
- החלטות - BoardResolution
- חברי ועד - BoardMember

### מתנדבים:

- מתנדב - Volunteer
- משמרות - VolunteerShift
- שעות - VolunteerHours

### ניהול תקין:

- ComplianceItem - פריטי צוות (עם deadline)
- ComplianceCertificate - אישורי ניהול תקין
- AnnualReport - דוחות שנתיים
- OrganizationDocument - מסמכים יסוד

### אוטומציות:

- Workflow - תהליכי עבודה (trigger → conditions → actions)
- הרצות - WorkflowExecution

- **WorkflowStep** - צעדים
- **Notification** - התראות
- **ScheduledTask** - משימות מתוכזנות

#### **:אינטגרציות**

- **IntegrationConfig** מוצפנים (credentials) הגדרות אינטגרציה - AES-256-GCM)
  - **IntegrationLog** לוגים של אינטגרציות
- 

## מנוע אוטומציה (Automation Engine)

ארQUITקטורה תלת-שכבותית:

1. **Trigger Layer** (דני, Cron) לוח זמנים (Event) מה מפעיל: אירוע -
2. **Matching Layer** (if/else) תנאים: סינון לפי כלליים -
3. **Execution Layer** (שליחת מייל, ייצור מסמך, עדכן רשומה) API call, פעולות: שליחת מייל, ייצור מסמך, עדכן רשומה

Event Bus (Redis Pub/Sub):

רלוונטיים workflows מנוע האוטומציה מАЗן ומפעיל → כל פעולה במערכת מפרסמת event.

tabנויות מוקנות מראש (Pre-built Templates):

1. קבלה אוטומטית בקבלת תרומה.
2. תזכורת לפני תום תקופה אישור ניהול תקין.
3. סיכום ישיבת ועד אוטומטי.
4. דוח חודשי אוטומטי לנציגי הארגון.
5. תזכורת להגשת דוח שנתי.
6. ברכת שנה טוביה/חג לתורמים.
7. אישור שעות מתנדבים חודשי.
8. התראה על חריגה מתקציב.
9. גיבוי מסמכים אוטומטי.
10. עדכן סטוס ניהול תקין מרשם העמותות.

UI - Visual Workflow Builder:

React Flow (לבנית תהליכי בגרירה drag & drop).

---

## שכבה אינטגרציות (Integration Layer)

Adapter Pattern:

כל אינטגרציה מממשת **BaseIntegrationAdapter**:

```
abstract class BaseIntegrationAdapter {
    abstract connect(config: IntegrationConfig): Promise<void>
    abstract disconnect(): Promise<void>
    abstract testConnection(): Promise<boolean>
```

```
abstract executeAction(action: string, params: any): Promise<any>
}
```

:אינטגרציות מתוכנות

אינטגרציה	סוג	זמן API?
חשבונית יroke	REST API	כ
iCount	REST API	כ
Tranzila	REST API	(כ  (סליה)
CardCom	REST API	(כ  (סליה)
WhatsApp Business	Cloud API	כ
Google Workspace	OAuth2 + APIs	כ
מס הכנסה	XML/SOAP	חלקי
רשם העמונות	Web Scraping + API	(בסיסי API חלקי (יש
בנקים	OFX/MT940	(ישיר API קבועים (לא

## שלבי מימוש (Implementation Phases)

שבועות 1-3 (Foundation) - שלב 1: בסיס

### קיצים קריטיים ליצירה:

- `turbo.json`, `package.json`, `docker-compose.yml`
- `packages/db/prisma/schema.prisma` כל הסכמה -
- `apps/web/middleware.ts` - Auth + Tenant
- `apps/web/app/(auth)/` - Login/Register
- `apps/web/app/(dashboard)/layout.tsx` - Shell + Sidebar
- `apps/web/components/ui/` - shadcn/ui components

### מה נבנה:

- Turborepo monorepo
- Docker Compose (PostgreSQL, Redis, MinIO)
- Prisma schema (מלא ~40 מודלים)
- NextAuth.js authentication
- Multi-tenant middleware + Prisma extension
- Dashboard layout עם sidebar (RTL)
- shadcn/ui components בסיסיים
- RBAC (Role-Based Access Control)
- Audit logging infrastructure

שלב 2: כספי + תורמים - שבועות 4-7

## **קבצים קרייטיים:**

- [apps/web/app/\(dashboard\)/finance/](#) - כל עמודי הכספיים
- [apps/web/app/\(dashboard\)/donors/](#) - ניהול תורמים
- [apps/web/app/api/v1/finance/](#) - API כספי
- [apps/web/app/api/v1/donors/](#) - API תורמים
- [packages/db/src/services/document-sequence.ts](#) - מספור רציף

## **מה נבנה:**

- ניהול חשבונות (יצירה, עריכה, שליחה PDF)
- ניהול קבלות (כולל קבלות סעיף 46)
- מספור רציף חוקי (DocumentSequence)
- ניהול העברות בנקאיות
- תקציב שנתי + מעקב
- דוחות כספיים בסיסיים
- תורמים (פרופיל, היסטוריה CRM)
- ניהול תרומות (חד-פעמי + הוראת קבע)
- קמפיינים
- הפקת קבלות אוטומטית בקבלה תרומה

שלב 3: ניהול תקין + ועד מנהל - שבועות 10-8

## **קבצים קרייטיים:**

- [apps/web/app/\(dashboard\)/compliance/](#) - ניהול תקין
- [apps/web/app/\(dashboard\)/board/](#) - ועד מנהל
- [apps/web/app/api/v1/compliance/](#)
- [apps/web/app/api/v1/board/](#)

## **מה נבנה:**

- צ'קליסט ניהול תקין אוטומטי
- מעקב אישורים ותעודות
- ניהול מסמכי יסוד
- יצירת דוח שנתי אוטומטי
- deadlines התראות
- ניהול ישיבות ועד
- יצירת פרוטוקולים אוטומטיות
- מעקב החלטות ומשימות
- הצבעות דיגיטליות

שלב 4: מתנדבים + מנוע אוטומציות - שבועות 11-14

## **קבצים קרייטיים:**

- [apps/web/app/\(dashboard\)/volunteers/](#)
- [apps/web/app/\(dashboard\)/automation/](#)
- [apps/worker/src/workers/automation.worker.ts](#) - הלב של האוטומציות
- [apps/worker/src/workers/](#) - כל-workers

**מה נבנה:**

- ניהול מתנדבים (רישום, שיבוץ, מעקב שעות)
- Event Bus (Redis Pub/Sub)
- Workflow engine (trigger → condition → action)
- Visual Workflow Builder (React Flow)
- תכניות אוטומציה מוכנות 10
- Scheduled tasks (cron)
- Notification system (email, push)
- BullMQ workers setup

**שלב 5: אינטגרציות + אנליטיקס - שבועות 19-15**

**কבצים קרייטיים:**

- [packages/integrations/src/](#) - כל adapters
- [apps/web/app/\(dashboard\)/integrations/](#)
- [apps/web/app/\(dashboard\)/reports/](#)

**מה נבנה:**

- Integration adapter base class
- iCount adapter / חשבונית יrokeה
- Tranzila / CardCom adapter (סליקה)
- WhatsApp Business adapter
- Google Workspace adapter
- Email automation (Nodemailer + React Email)
- adapter (API + scraping) רשם העחות
- דashboard אנליטיקס
- מותאם אישית Report builder

**יצור - שבועות 20-24: שלב 6**

**מה נבנה:**

- Landing page + Pricing page
- Subscription management (Stripe/PayPlus)
- Onboarding wizard
- **לעמותות חדשות בדיקות E2E (Playwright)**
- Performance optimization
- Security audit
- Deployment (Vercel + Railway/Render)
- Monitoring (Sentry + analytics)
- Documentation

**החלטות עיצוב מפתח**

1. **Multi-tenancy: Row-Level Security** - כל טבלה עם [organizationId](#), Prisma extension מוסיף פילטר על טבלה עם organizationId. מספק טובי ל-10,000 ארגונים, schema-per-tenant. פשטוט יותר מ

2. מבטיח - בטור transaction - טבלה נפרדת עם **מספר רציף (DocumentSequence)**: `SELECT FOR UPDATE` מספר רציף.
  3. **Credentials**: כל הרשאות אינטגרציה מוצפנות בenv vars.
  4. **Event Bus → Automation**: מנע אוטומציות מזין → מפעיל → event מפרשמת mutation כל event-driven. ארכיטקטורה workflows.
  5. **PDF Generation**: `@react-pdf/renderer`, Puppeteer, לקבילות/חשבוניות (מהיר).
- 

## אימות (Verification)

בדיקת שלב 1:

```
# הרץת DB + Redis
docker-compose up -d
# מיגרציה
npx prisma migrate dev
# הרצת אפליקציה
npm run dev
# בדיקה: כניסה, יצירה ארגון, הרשות
```

בדיקת כל שלב:

- Unit tests עם Vitest לכל service
- Integration tests ל-API routes
- E2E tests Playwright עם Cypress (לתחומים קרייטיים (יצירת חשבונית, קבלת תרומה
- דנית בכל עמוד RTL בדיקת

בדיקת אוטומציות:

- ידוא הפעלה → trigger event
- ידוא הרצה בזמן → Scheduled task
- Integration test → mock external API → ידוא adapter

בדיקת אינטגרציות:

- Sandbox/test environments
- Chiron Chironite sandbox
- Tranzila test mode