

CaregiveFlow — אפיון מלא מפורט (Master Spec v2)

גרסה: 2.0 • מצב: מאושר בסיס • שפה: עברית (RTL)
עקרונות: **לוגיקה ב-PHP בלבד**, ה-DB מיועד **אך ורק** לאחסון/שליפה/עדכון/מחיקה (CRUD) + אינדקסים
לביצועים. להצמד לסגנון הקוד וה-MVC הקיים בפרויקט. שמות קבצים/פונקציות/תבניות — זהים לקיים ככל האפשר.

1) מבוא ומטרות

CaregiveFlow היא מערכת ניהול לסוכנויות/תאגידים העוסקים בהשמות עובדי סיעוד. המערכת מרכזת נתוני **עובדים**, **מעסיקים/מטופלים**, **השמות**, **דרכונים/ויזות/ביטוחים**, **דמי תאגיד**, **היתרי העסקה**, **קבצים ומסמכים**, וכוללת **ייצוא קבצים** **רגולטוריים** (PIBA XML, BAFI TXT) וכן **דוחות רבעוניים באקסל** עבור ביקורי בית. המערכת בנויה MVC מינימלי ב-PHP, RTL, וידידותית לתפעול יומיומי.

1.1 בעלי עניין

- הנהלת התאגיד/לשכה
- רכזים/עו"סים/צוות שטח
- משתמשי Back-office (גבייה/דוחות)
- רגולטור/רשויות (דוחות, קובצי ייצוא)

1.2 יעדים

- בסיס נתונים אמין עם עקביות ואילוצים
- UI מהיר עם ממשק עברי מלא (RTL)
- תהליכי CRUD פשוטים ובטוחים
- ייצואים רגולטוריים תואמי-מפרט
- דוחות ניהוליים ותזכורות (expirations, חובות וכו')

2) ארכיטקטורה ומבנה פרויקט

2.1 טכנולוגיות

- PHP 8.x על Apache (XAMPP בסביבת פיתוח)
- MariaDB/MySQL
- Bootstrap 5 + RTL
- אין Composer; מחלקות שירות יושבות תחת `src/services`

2.2 מבנה תיקיות (תמציתי)

```
public/          # קבצי הורדה זמניים, CSS/JS, נכסי index.php
src/
```

```

controllers/    # EmployeesController.php, EmployersController.php, ...
models/         # Employee.php, Employer.php, Placement.php, ...
services/       # PibaXmlBuilder.php, BafiFileBuilder.php,
HomeVisitQuarterlyExporter.php
helpers/        # פונקציות עזר: e(), csrf(), flash(), auth()
views/
  <module>/     # index.php, form.php, show.php
sql/            # שאילתות עזר/seed/מיגרציות
docs/           # אפיונים ודוקומנטציה
storage/
  uploads/      # קבצים מצורפים
  exports/      # פלטים רגולטוריים/דוחות

```

MVC 2.3 — מוסכמות

- **Models:** מחלקות עם API עקבי: `count($pdo, $filters = [])`, `all($pdo, $filters = [])`, `delete($pdo, $id)`, `update($pdo, $id, $data)`, `create($pdo, $data)`, `find($pdo, $id)`.
- **Controllers:** פעולות סטנדרט `index/create/edit/show/delete` + פעולות ייעודיות (למשל `export()`), שימוש ב-PDO, ולידציה בסיסית, הודעות פלאש.
- **Views:** שלושה קבצים עיקריים לכל מודול (`index/form/show.php`), RTL, Bootstrap, אותם אייקונים/כפתורים, קוד HTML/PHP תואם לסגנון הקיים.

2.4 עקרונות קוד

- הערות דו-לשוניות (HE/EN) כשנדרש.
- פונקציית `e()` לכל הדפסה למסך (XSS).
- CSRF בכל טופס שינוי (input מוסתר + בדיקה בקונטרולר).
- `try/catch` סביב פעולות DB להודעת שגיאה נקייה למשתמש ורישום לוג מפורט.
- **אין לוגיקה ב-DB:** ללא Stored Procedures/Triggers. חישובים/מיפויים/פורמטים — אך ורק ב-PHP (`services`).

(3) אבטחה, הרשאות, תאימות

- תפקידי ברירת מחדל: `viewer` (קריאה בלבד), `staff` (CRUD ללא מחיקה), `admin` (מלא).
- בדיקות הרשאה בכל פעולה מתערכת.
- הצפנת סיסמאות (אם קיים מודול משתמשים) ב-bcrypt/argon2.
- GDPR/תקנות פרטיות: מידע מזוהה מוצג רק היכן שנדרש; ייצוא/דוחות כוללים שדות חובה בלבד.

(4) בסיס נתונים — ישויות מרכזיות

דוגמאות שמיות/שדות — יתאימו לשמות אצלך, עם מיפוי בשכבת השירות במידת הצורך. אילוצים ואינדקסים — חובה.

4.1 Employees (עובדים)

- שדות: id, first_name_en, last_name_en, first_name_he, last_name_he, id_number, created_at, notes, birth_date, country_code, email, cellphone, updated_at (אם קיים).
- אינדקסים: (last_name_en, first_name_en, id_number), (cellphone).

4.2 Employers (מטופלים/מעסיקים)

- שדות: id, first_name, last_name, id_number, city_name / city_code, street, updated_at, created_at, notes, current_employee_id, email, phone (אופציונלי).
- אינדקסים: (last_name, first_name, id_number).

4.3 Placements (השמות)

- שדות: id, employee_id, employer_id, placement_type_code, start_date, end_date, status_code, notes.
- אינדקסים: (employee_id, employer_id, start_date, status_code).

4.4 Passports / Visas / Insurance (לעובד ומעסיק)

- Passports (עובד): employee_passports עם passport_number, country_code, issue_date, expiry_date, is_primary וכו'.
- Passports (מעסיק): employer_passports (אם קיים אצלך) — שדות דומים.
- Visas: employee_visas — visa_number, issue_date, expiry_date, status_code.
- Insurance: employee_insurances — policy_number, coverage_type, issue_date, expiry_date, provider, status_code.
- אינדקסים על (employee_id, expiry_date) ועוד לתזכורות.

4.5 Employer Fees (דמי תאגיד)

- employer_fees: מזהה מעסיק, סכום, תקופת חיוב (חודש/שנה), תאריך תשלום בפועל, אמצעי תשלום, סטטוס.

4.6 Employer Permits (היתרי העסקה)

- employment_permits (למעסיק): מספר היתר, סוג, סטטוס, תוקף, קבצים מצורפים.

4.7 קבצים ומסמכים

- טבלת מסמכים לכל מודול או טבלת על מאוחדת עם entity_type, entity_id, doc_type_code, path, original_name, size, uploaded_at, uploaded_by.

4.8 טבלאות קודים

- ערי ישראל, ארצות, סוגי השמה, סטטוס ויזה/ביקור/ביטוח/דרכון, סוגי מסמך — read-only, נטענים ב-seed ל-SQL.

5) מודולים ותהליכים (UI/Flows)

Employees 5.1

- **Index:** חיפוש (שם/דרכון/טלפון/אימייל), עמודות עיקריות + אינדיקטורים תוקף (תאריכים בצבע: אדום פג/ירוק בתוקף).
- **Form:** יצירה/עריכה, הכול כולל קבצים (אותו רכיב העלאה). ולידציות שדות חובה.
- **Show:** כרטיס עובד עם טאבים: פרטים • דרכונים • ויזות • ביטוחים • השמות. פעולות מהירות (כפתורי שיבוץ/יצוא).


Employers 5.2

- **Index/Form/Show** במבנה זהה; עמודת "מעסיק נוכחי" ברשימת עובדים תילקח מ-Placements פעיל.

Placements 5.3

- יצירה משורת עובד: פעולה "שיבוץ למעסיק" פותחת `placements/form.php` עם employee מוגדר מראש.
- ולידציות: לא לאפשר שני Placements פעילים במקביל לעובד.

Passports/Visas/Insurance 5.4

- מבנה 1:1 views עם מודול הוויזות הקיים; אותו UI/אייקונים; `index/form/show` עם RTL.
- שדה `is_primary` — הצגה כ- בלבד בעמודת רשימה.

Employer Fees 5.5

- מסך סיכום לפי תקופת תשלום נבחרת, כולל פילטר "מעסיקים שלא שילמו בתקופה X" (שאלות אופטימליות; הקפדה על `count()` מחזיר int).

Employer Permits 5.6

- מודול זהה בסגנון ל-Passports אך במודל המעסיק (קבצים/סטטוסים/תוקפים).

File System Integration 5.7

- בכל מודול ניתן לצרף קבצים — UI אחיד, טבלת מסמכים, שמירה ל- `storage/uploads/<module>/<id>/`.

5.8 דוחות/התראות

- עמוד בית עם התראות: מסמכים/ויזות/ביטוחים/דרכונים שקרובים לפקיעה; דמי תאגיד לא משולמים.

6) ייצוא רגולטורי

PIBA (XML) 6.1

- **Service:** `services/PibaXmlBuilder.php` — בניה מלאה ב-PHP; טענת קודי טבלאות; ולידציות טרם יצוא; שם קובץ וארכוב.

BAFI (TXT Fixed-Width) 6.2

- Service: `services/BafiFileBuilder.php` — Control Record, רשומות עובדים, ספירת רשומות (ללא שורת בקרה), ריווח/ריפוד.
- שם קובץ: `oz_siud_manot_<bureau>_<employee_id>_<unique>.txt` כאשר `<unique>` = יוניק ע"פ timestamp+random או מס' רץ ממחולל טבלאי.

Home Visits – Excel 6.3 (רבעוני)

- Service: `services/HomeVisitQuarterlyExporter.php` — ייצוא דו-גיליוני לפי `V2.xlsx` דוחות רבעוניים - פורמט דיגיטלי/`docs`.
- גליון "דוח רבעוני" (18 עמודות מדויקות) + גליון "הצהרה מנהלים" (מילוי שם לשכה/שנה/רבעון; חתימות ידני/USB).
- ערכי הרשימות — **בדיוק** כפי שמוגדר באקסל (כולל Q1-Q4, סוגי/סטטוס ביקור/השמה).
- פורמט: RTL, תאריכים אמיתיים, טלפון/דרכון כ-Text, מספור #, דו"ח שגיאות לפני הורדה.

(7) מודול ביקורי בית (Home Visits) — אפיון מלא

7.1 מטרות

- תיעוד ביקורי בית אצל מעסיקים סביב השמות פעילות.
- שקיפות מול הנהלה/רגולטור באמצעות דוח רבעוני תקני.

DB 7.2

: `home_visits`

- PK `id`
- `employee_id` FK → `employees.id` (חובה)
- `employer_id` FK → `employers.id` (חובה)
- `placement_id` FK → `placements.id` (אופציונלי)
- `scheduled_at` DATETIME (מועד מתוכנן)
- `visited_at` DATETIME (מועד בפועל) **נכנס לדוח**
- `visit_status` VARCHAR(50) (ערך מהרשימה שבאקסל)
- `visit_type` VARCHAR(50) (פנימי)
- `notes` TEXT
- `followup_needed` TINYINT(1) DEFAULT 0
- `followup_due` DATE
- `followup_notes` TEXT
- `risk_level` TINYINT UNSIGNED DEFAULT 0
- `visitor_name` VARCHAR(120) (חובה)
- `visitor_role` VARCHAR(50) DEFAULT 'עו"ס' (**קבוע**)
- `updated_by` INT, `created_by`
- `created_at` TIMESTAMP DEFAULT CURRENT_TIMESTAMP
- `updated_at` TIMESTAMP ON UPDATE CURRENT_TIMESTAMP

אינדקסים: (employee_id), (placement_id), (visited_at), (visited_at, visit_status)

home_visit_documents (אופציונלי): id, home_visit_id, FK CASCADE, doc_type_code, path, uploaded_at, uploaded_by, size_bytes, mime_type, original_name

7.3 ולידציות

- (visitor_name, employer_id, employee_id, visited_at, visit_status, visitor_role) — חובה. מתוך הרשימה בלבד.
- תמיד "עו"ס".

UI 7.4

- **Index**: פילטרים (שנה/רבעון/טווח/סטטוס/עובד/מעסיק/רק באיחור), טבלת רשומות, פעולות.
- **Form**: בחירת עובד/מעסיק (Select2/כמו הקיים), תאריך ביקור, סטטוס (Select מהרשימה), השמה, מבצע/תפקיד (תפקיד נעול ל-"עו"ס"), הערות, קבצים.
- **Show**: פרטי ביקור מלאים + קבצים + "שכפל ביקור".

7.5 יצוא רבעוני (Excel)

- ראה 6.3§; מפיק את 18 העמודות המדויקות, מילוי גליון הצהרה, שמירת שם לשנה/שנה/רבעון.
- שם קובץ: home_visits_quarterly_<year>_Q<quarter>_<YYYYMMDD_HHMM>.xlsx

8 כללים לשמות, מיפוי ופורמטים

- שמות קבצים: באותיות לטיניות כאשר נדרש (למשל לצורך הורדות).
- תאריכים: YYYY-MM-DD ב-DB; באקסל dd/mm/yyyy
- ערכי קוד: שמורים בטבלאות קוד + JOIN להצגה ידידותית.

9 ביצועים ויציבות

- אינדקסים לפי שימוש בשאלות.
- טעינת נתונים מרוכזת בשירותים (הימנעות מ-N+1).
- Healthcheck (sql/007_self_check.sql) וכד' לבדיקות עקביות (FKs, ערכים חסרים בחשובים).

10 שגיאות, לוגים ומעקב

- כל חריגת PDO — רישום ללוג (קובץ טקסט ב-storage/logs/app.log) + הודעת פלאש ידידותית.
- רישום אירועים עסקיים (יצוא/הורדות/מחיקות) עם timestamp ומשתמש.

11) בדיקות

- בדיקות ידניות לכל מודול: תרחישי CRUD, ולידציות, הרשאות.
- בדיקות ייצוא: השוואה מול קבצי דוגמה/מפרטים (PIBA/BAFI/Excel רבעוני).

12) פריסה ו-DevOps

- DEV: XAMPP מקומי; PRs/Commits לפי מודולים; ענף main יציב.
- ארכוב יצואים תחת storage/exports/<module>/.
- גיבויים: דמפ DB יומי, גיבוי storage/ שבועי.

13) Roadmap (הצעה)

- **סבב A:** Home Visits (DB + Views + Export), חיזוק דמי תאגיד (פילטרים/סיכומים), כפתורי ניווט עקביים בכל המסכים.
- **סבב B:** Dashboard התראות, חוויית קבצים אחידה בכל המודולים.
- **סבב C:** שיפורי UX (אייקונים+tooltips), שיפורי חיפוש מתקדם, דוחות ניהוליים נוספים (CSV/PDF).

14) קונבנציות UI/UX

- RTL בכל מקום; פונט מערכת אחיד; כפתורי פעולה בצד ימין של טבלאות.
- צבעי תאריכי תוקף: ירוק בתוקף, אדום פג; אפור לחסר.
- breadcrumbs/כפתור "חזרה" לעמוד הקודם.

15) נספחים: מיפויי שדות (תמצית)

פירוט מלא לפי מודולים:

15.1 דוח רבעוני (Home Visits Excel) — 18 עמודות

1.	# (מספור)
2.	תעודת זהות (employers.id_number)
3.	שם משפחה (employers.last_name)
4.	שם פרטי (employers.first_name)
5.	יישוב (employers.city_name/קוד→שם)
6.	רחוב (employers.street)
7.	טלפון (מטופל/בן משפחה) (employers.phone)
8.	שם משפחה עובד (employees.last_name_en)
9.	שם פרטי עובד (employees.first_name_en)
10.	פלאפון עובד (employees.cellphone)

11.	מספר דרכון	(employees.passport_number)
12.	ארץ מוצא	(employees.country_name/קוד→שם)
13.	תאריך הביקור	(home_visits.visited_at)
14.	סטטוס ביקור	(home_visits.visit_status — ערכי הרשימה מהאקסל)
15.	סוג השמה	(placements.placement_type — מיפוי לטקסט שבאקסל)
16.	תאריך השמה	(placements.start_date)
17.	מבצע הביקור	(home_visits.visitor_name)
18.	תפקיד מבצע הביקור	(קבוע: "עו"ס")

BAFI / PIBA 15.2 — עקרונות מיפוי

- BAFI: שדה אורך/יישור/ריפוד לפי המיפוי באקסל הייעודי של BAFI (קיים בפרויקט); Control Record מחשב ספירת רשומות **ללא** שורת הבקרה.
- PIBA: XML Nodes, קודי טבלאות, שמירת פורמטי תאריך ומזהים כנדרש.

16) קווים מנחים לפיתוח מול ChatGPT (Best Practices)

- לצרף תמיד קובץ/קטע קוד רלוונטי → קבל Patch "אחד לאחד" התואם לסגנון.
- להתחיל בכל מודול מ-SQL + שלד קבצים (models/controllers/services/views) ואז להשלים פרטים.
- בייצוא/דוחות: קודם מיפוי מלא באקסל, אחר כך Service, ואז מסך UI קטן לטריגר/הורדה.