

תיעוד עבודה בפרויקט LR-ANGUS עם כלי בינה מלאכותית

מגישים: רותם סויסה ת"ז 207788027 ולידור בן סימון ת"ז 207925330

שימוש בבינה מלאכותית במהלך ביצוע הפרויקט

במהלך הפרויקט שלנו, נעזרנו רבות בכלים של בינה מלאכותית כמו – ChatGPT, Claude ו-Ideogram אלו כלים שמלווים את תהליך הפיתוח המודרני, ובמקום להימנע מהם – בחרנו ללמוד דרכם, להבין טוב יותר תהליכים מורכבים, ולמנף אותם כדי לשפר את איכות הפרויקט והקוד שכתבנו.

שלבים מרכזיים בהם נעזרנו בבינה המלאכותית:

1. תכנון מבנה הפרויקט (Client + Server)

לאחר קבלת הדרישות מהמרצה ומסמך המחווה, השתמשנו ב- ChatGPT כדי להבין איך עלינו לבנות את מבנה הפרויקט מבחינת התיקיות, הקבצים, חלוקה בין צד לקוח לצד שרת, ו-MVC ב- Node.js. הוא יצר עבורנו תרשים ברור שנשענו עליו לאורך כל הדרך:

מבנה הפרויקט הכללי

Edit Copy

plaintext

```
restaurant-project/
├── lr-angus-client/      ← צד לקוח (Angular)
│   ├── src/
│   │   ├── app/
│   │   │   ├── pages/    ← (menu, order, reservation, home, payment) כל עמוד בפרויקט
│   │   │   ├── models/   ← 'וכן Reservation, Table מגדיר ממשקים כמו
│   │   │   ├── services/ ← (HTTP) שירותים שאחראים לשליחת בקשות לשרת
│   │   │   ├── shared/   ← (כמו navbar, footer) רכיבים/מודולים לשימוש חוזר
│   │   └── assets/       ← 'קבצי תמונות, לוגו, רקעים וכו'
│   └── angular.json      ← קובץ קונפיגורציה של פרויקט
├── lr-angus-server/     ← צד שרת (Node.js + Express)
│   ├── controllers/     ← קבצים שמקבלים בקשות מהלקוח (כמו ReservationController)
│   ├── routes/          ← (reservationRouter, orderRouter...) לכל נושא router קובץ
│   ├── services/        ← הלוגיקה העסקית: גישה למסד נתונים, ולידציות ועוד
│   ├── models/          ← לא מייצג טבלאות, אלא לעתיד - (Dish.js בשלב זה רק)
│   ├── config/          ← MySQL-שמנהל את החיבור ל database.js קובץ
│   └── server.js         ← והנתיבים Express-נקודת הכניסה לשרת - מפעיל את ה
└── database/
    └── restaurant_db.sql ← לבניית הטבלאות והנתונים במסד הנתונים SQL קובץ
```

תיעוד עבודה בפרויקט LR-ANGUS עם כלי בינה מלאכותית

2. יצירת גלריית תמונות אינטראקטיבית בעמוד הבית (חדשנות בפרויקט)

החלק החדשני ביותר בפרויקט שלנו היה גלריית התמונות בעמוד הבית — גלילה אוטומטית של תמונות עם עצירה כאשר המשתמש עובר עם העכבר. קיבלנו השראה מהצ'ט כיצד לבנות זאת באמצעות HTML, CSS ו- ts פשוט. הנה החלקים המרכזיים של הקוד:

```
//לוגיקת התמונות במערך/
carouselImages: string[] = [
  '/assets/img/0.avif', '/assets/img/15.jpg', '/assets/img/14.jpg',
  '/assets/img/4.png', '/assets/img/1.png', '/assets/img/6.png',
  '/assets/img/3.png', '/assets/img/5.jpg', '/assets/img/2.png',
  '/assets/img/7.webp', '/assets/img/8.jpg', '/assets/img/9.webp',
  '/assets/img/10.webp', '/assets/img/11.jpg', '/assets/img/12.jpg',
  '/assets/img/16.webp', '/assets/img/13.jpg', '/assets/img/17.png',
];

ngAfterViewInit(): void {
  this.initializeCarousel();
}

//עצירה אוטומטית במעבר העכבר/
private initializeCarousel(): void {
  if (typeof window === 'undefined') return;

  const container = document.querySelector('.image-carousel-container');
  const wrapper = document.getElementById('carouselWrapper');

  if (container && wrapper) {
    container.addEventListener('mouseenter', () => {
      wrapper.style.animationPlayState = 'paused';
    });
    container.addEventListener('mouseleave', () => {
      wrapper.style.animationPlayState = 'running';
    });
  }
}
```

מקובץ ts

```
<!-- גלריית תמונות -->
<div class="container mt-5">
  <h2 class="text-center gallery-title"><i class="fas fa-images"></i> גלריית תמונות </h2>
  <div class="image-carousel-container">
    <div class="image-carousel-wrapper" id="carouselWrapper">
      <ng-container *ngFor="let src of carouselImages.concat(carouselImages)">
        <div class="carousel-item">
          <img [src]="src" alt="תמונת מנה" />
        </div>
      </ng-container>
    </div>
  </div>
</div>
```

מקובץ html

תיעוד עבודה בפרויקט LR-ANGUS עם כלי בינה מלאכותית

```
/* גלריית תמונות - עיצוב + אנימציה */
.gallery-title {
  color: #b07a5b;
  font-weight: bold;
  margin-bottom: 30px;
  position: relative;
  text-shadow: 2px 2px 4px rgba(0, 0, 0, 0.3);
  letter-spacing: 1px;}

.gallery-title::after {
  content: '';
  position: absolute;
  width: 100px;
  height: 3px;
  background-color: #b07a5b;
  bottom: -10px;
  left: 50%;
  transform: translateX(-50%);}

.image-carousel-container {
  width: 100%;
  overflow: hidden;
  margin: 50px 0;
  position: relative;
  height: 350px;}

.image-carousel-wrapper {
  display: flex;
  animation: scroll 30s linear infinite;
  width: fit-content;}

@keyframes scroll {
  0% {
    transform: translateX(0%);}
  100% {
    transform: translateX(50%);}}

.carousel-item {
  min-width: 450px;
  height: 300px;
  margin: 0 15px;
  border-radius: 15px;
  overflow: hidden;
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.5);
  transition: all 0.3s ease;
  opacity: 1 !important;
  visibility: visible !important;
  display: block !important;}

.carousel-item:hover {
  transform: scale(1.05);
  box-shadow: 0 8px 16px rgba(0, 0, 0, 0.7);}
```

מקובץ CSS

תיעוד עבודה בפרויקט LR-ANGUS עם כלי בינה מלאכותית

3. צד השרת – (Express + MySQL) חיבורים בין הרכיבים

הבינה המלאכותית עזרה לנו להבין איך בונים צד שרת מלא עם:

- שירותים (services)
- קונטרולרים (controllers)
- ראוטרים (routes)
- התחברות ל-MYSQL
- שאילתות SELECT/INSERT והכנסת מידע אמיתי מהפורמים

לדוגמה, היא עזרה לנו לוודא שהקובץ server.js מכיל את ההגדרות הנכונות:

```
const express = require('express');
const cors = require('cors');
const bodyParser = require('body-parser');
require('dotenv').config();

// ייבוא מודולים פנימיים
const database = require('./config/database');
const menuRoutes = require('./routes/menuRoutes');

// ייבוא נתיבים נוספים (רק אם הקבצים קיימים)
let orderRoutes = null;
let reservationRoutes = null;
let paymentRoutes = null;

try {
  orderRoutes = require('./routes/orderRoutes');
  app.use('/api/orders', orderRoutes);
  console.log('✅ נתיבי הזמנות נטענו בהצלחה');
} catch (error) {
  console.log('❗ נתיבי הזמנות לא נמצאו - ניתן ליצור מאוחר יותר');
}

try {
  reservationRoutes = require('./routes/reservationRoutes');
  console.log('✅ נתיבי הזמנות שולחן נטענו בהצלחה');
} catch (error) {
  console.log('❗ נתיבי הזמנות שולחן לא נמצאו - ניתן ליצור מאוחר יותר');
}

try {
  paymentRoutes = require('./routes/paymentRoutes');
  console.log('✅ נתיבי תשלומים נטענו בהצלחה');
} catch (error) {
  console.log('❗ נתיבי תשלומים לא נמצאו - ניתן ליצור מאוחר יותר');
}
```

תיעוד עבודה בפרויקט LR-ANGUS עם כלי בינה מלאכותית

4. פתרון באגים ותקלות קומפילציה

במהלך העבודה על Angular ו-Node.js נתקלנו בהרבה באגים קטנים שעצרו את העבודה החל ממבנים לא תקינים, שגיאות JSON, TypeScript, ngModel, ngFor ועד לחיבורים למסד נתונים. במקום לבזבז זמן יקר על ניחושים, נעזרנו בצ'ט כדי להבין במהירות מה גורם לתקלה ולקבל תיקון מדויק.

5. יצירת לוגו ותמונה ראשית עם שם המסעדה

השתמשנו גם בכלי בינה חזותיים כמו [Ideogram.ai] כדי לעצב את לוגו המסעדה שכלל את ראשי התיבות שלנו (לידור ורותם) בשילוב השם "Angus" כסוג בשר יוקרתי. הכלי נתן לנו לוגו במראה מודרני שהתאים בדיוק לקונספט.



כלים בהם השתמשנו:

- ChatGPT - הסבר קוד, פתרון תקלות, בניית מבנה פרויקט
- Claude - עזרה ביצירת השרת, תיעוד, טיפים לעיצוב
- Ideogram.ai - יצירת לוגו, תמונה ראשית מותאמת לעמוד הבית

לסיכום:

הבינה המלאכותית לא עשתה את הפרויקט עבורנו, אלא הייתה חלק מתהליך הלמידה שלנו מלווה, מסבירה, מתקנת ומאפשרת להתקדם בצורה חכמה יותר. בזכות העבודה איתה למדנו יותר על כתיבת קוד תקני, ניתוח בעיות, חוויית משתמש, ושיטות עבודה נכונה בפרויקט כצוות.