**מפרט אריטקטוני**

הפרויקט מחולק לשני חלקים- לצד השרת וצד הלקוח

. 1 צד שרת שנכתב בשפת #C בטכנולוגית asp.Net Core

2 . צד לקוח שנכתב בשפת Script Type, Html ובטכנולוגית REACT

**צד שרת:**

הקוד ב Dot.Net מאורגן בתוך solution שמכיל את כל הקוד של המערכת.

ובתוכו אפשר להוסיף פרויקטים רבים. בין הפרויקטים קיים קשר והם יכולים להשתמש זה בזה.

נארגן את הקוד בפרויקטים נפרדים בתוך ה- solution כל פרויקט מכיל קוד עם תפקיד מסוים והוא בעצם שכבה אחת של המערכת. ארגון של הקוד בשכבות נועד כדי לאפשר ניהול טוב יותר של הקוד, יכולת לעדכן חלקים ממנו בלי שכל העדכון ישפיע על כל הקוד כולו ועוד.

שם ה- solution  הוא שם הפרויקט HMO

בתוך ה- solution  יש לנו את הפרויקטים הבאים:

1. HMO.Repositories – מסוג Class libery שמתקשר עם מקור הנתונים, מחזיק את הנתונים ומטפל בפעולות שליפה , עדכון, הוספה ומחיקה.
2. HMO.Services - מסוג Class libery ובו הלוגיקה העסקית.
3. HMO.Common - מסוג Class libery בו אובייקטים להמרה מטיפוס של מסד נתונים למחלקה בC#.
4. HMO.WebAPI – מסוג Web Api ובו הApi שחשוף החוצה ומספק לצד לקוח גישה לפונקציות של שליפת נתונים, עדכון, הוספה ומחיקה וכל מה שנדרש.

**פירוט:**

**HMO.Repositories**

בפרויקט Repositories ננהל את הנתונים ואת התקשורת עם מסד הנתונים.

נארגן את הקוד ב 3 תיקיות:

Entities

בתיקיה זו יהיו ה-CLASSים שמחזיקים את הנתונים. כל Class  ממופה לטבלה אחרת ובו properties שתואמים לשדות הטבלה.

הEntities הם מחלקות פשוטות שמגדירות את מבני הנתונים בלבד ולכן יש בהם רק properties

הפונקציונאליות של קישורים ממסד נתונים מוגדרים במחלקות נפרדות שנקראות Repositories.

Interfaces

בשביל שנוכל להשתמש בהזרקות קוד לצורך גמישות המערכת אנו יוצרים לכל מחלקה פונקציונלית ממשק ובמהלך כל הקוד נשתמש בטיפוסי ממשק, בבנית האפליקציה נזריק את הסרברים המתאימים.

בתיקיה זו שמורים הממשקים של המחלקות Repositories שהם המחלקות הפונקציונלית של קישור למסד נתונים.

Repositories

מחלקות ובהם פונקציות לקישור עם מסד נתונים כמו שליפה, מחיקה , עדכון, הוספה וכו.

**HMO.Common**

שכבה זו אחראית על המרה של נתונים מה Repositories שבה נתונים מסוג של מסד הנתונים.

ל services שבה נתונים מסוג מחלקות רגילות.

**HMO.Services**

אחראית על הלוגיקה העסקית.

בשכבה זו גם קיימיםInterfacses ומחלקות פונקציונאליות שמבצעות את הלוגיקה העסקית בפרויקט.

**HMO.WebAPI**

פרויקט הWeb Api הוא השכבה שחשופה החוצה ומקבלת את בקשות צד הלקוח.

בפרויקט זה יש תיקיית controllers שבתוכה classים מסוג Controller שהם בעצם הendPointים שחשופים החוצה וניתן לשלוח אליו קריאה באמצעות ניתוב ייחודי משלו.

בפרויקט זה מבוצעות הזרקות תלויות שמטרתם להפוך את הקוד ל "טסטבילי" וגמיש לשינויים.

ההזרקות מבוצעות בקובץ program.cs שנמצא בפרויקט HMO.WebAPi.

ההזרקות מנוהלות באמצעות פונקציית הרחבה Extension Method.

בכל שכבה קיים Extension ל IServiceController וכל שכבה מזריקה לעצמה את הסרוויסים וקוראת ל Extension של השכבה מתחתיה.

שכבת Services מזריקה גם את ה AutoMapper שדואג להמרות של שכבת הCommon.