**ארכיטקטורת התוכנה**

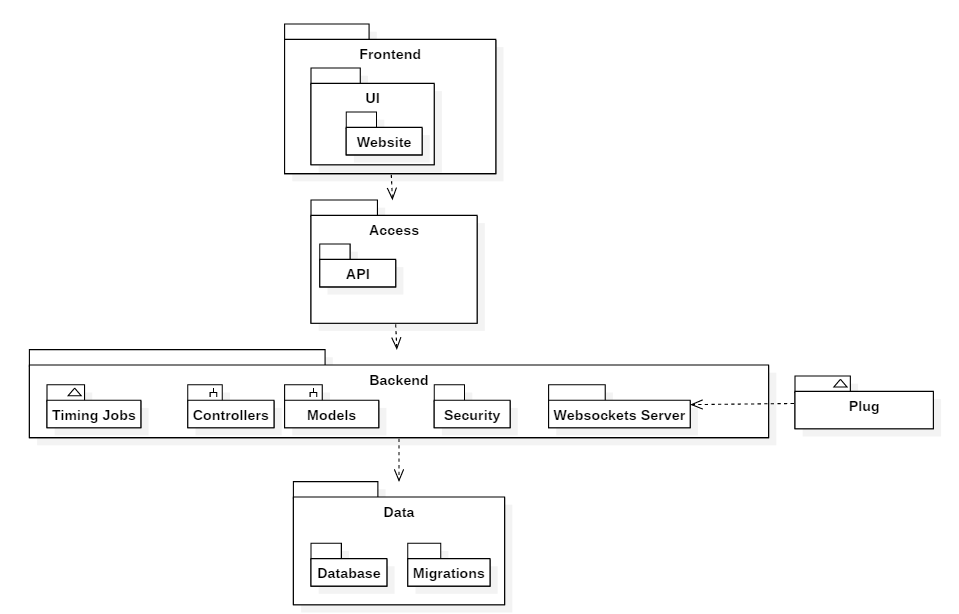
בצד הלקוח יש אתר בו המשתמש מבצע פעולות כמו התחברות, צפייה במכשירים ובאייקונים שלהם, הדלקת/כיבוי מכשירים, קביעת זמני הדלקה וכיבוי, הוספת מכשירים, צפייה בצריכת חשמל ובעלות כספית מצטברת.

ב-API נמצאות הפונקציות בהן צד הלקוח משתמש כדי לגשת לאפליקציה ולבצע את הפעולות שתוארו לעיל. בשכבה זו נמצאות בקשות משיכת נתוני מכשירים, דגימות מתח וזרם, תזמונים; בנוסף גם הפעלת בקשות הדלקה, כיבוי ותזמון מאוחר יותר של פעולות אלו.

צד השרת מנהל את תזמוני ההדלקות, מספק גישה ל-API המוזכר לעיל לנתונים ממסד הנתונים, מתקשר עם המכשיר החכם ומקבל דרכו את הדגימות, מעביר פקודות הדלקה וכיבוי למכשיר ומספק שירותי אבטחה ל-API (משתמש לא יכול לשנות נתוני מכשיר שלא ברשותו).

השקע החכם בעל יכולת להדליק ולכבות את המכשיר שמחובר אליו, כמו גם למדוד ולשלוח נתוני צריכה.

מסד הנתונים מחזיק בכל המידע שיש לשמור לגבי המצב העכשווי וההיסטוריה של המכשירים, ביניהם תזמונים ודגימות. בנוסף, מסד הנתונים שומר מידע מוצפן לגבי סיסמאות משתמשים ו-tokens עבור התנהלות השרת מול ה-API.



**ספריות בולטות**

ה-Websockets Server ממומש בעזרת ספריית ה-Open Source בשם Fleck

תזמוני ההדלקות וכיבויים ממומשים בעזרת ספריית Hangfire

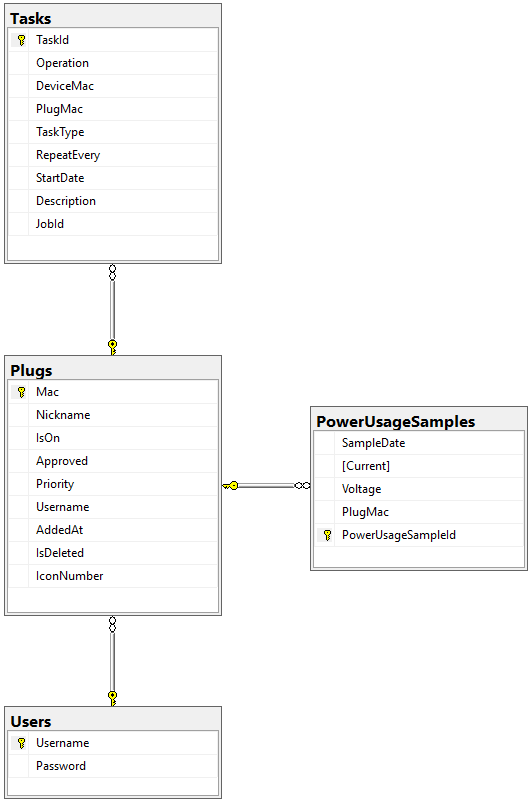
גישה למסד הנתונים מתבצעת בעזרת Entity Framework

התנהלות אוטמוטית למעבר בין DTOs לישות האמיתית בעזרת AutoMapper

אבטחה ע"י Identity

גרף לשם צריכת חשמל ועלות מצטברת ע"י Plotly

**המבנה הטבלאי במסד הנתונים**

****