

ארכיטקטורת המערכת

אופן הפניה בין הסרוויסים:

המערכת מחולקת לשני חלקים עיקריים: צד שרת וצד לקוח. בין החלקים קיימת תקשורת, ואופן פעולתה תפורט בשורות הבאות:

צד השרת:

נכתב בשפת Python בטכנולוגיית **WebApi**:

Web API (Application Programming Interface) היא טכנולוגיה המשמשת לבניית יישומי אינטרנט ושירותים שניתן לגשת אליהם באמצעות פרוטוקולי אינטרנט סטנדרטיים כמו HTTP. זוהי מסגרת לבניית שירותים מבוססי HTTP שיכולים לשמש לקוחות שונים, כולל דפדפני אינטרנט, מכשירים ניידים ויישומי שולחן עבודה.

Web API מאפשר למפתחים ליצור שירותי RESTful, שהם שרתי אינטרנט קלים הניתנים להרחבה, שמשתמשים בשיטות HTTP סטנדרטיות כמו GET, POST, PUT ו-DELETE כדי לתקשר עם לקוחות. ניתן להשתמש בשירותים אלו לביצוע פעולות CRUD (יצירה, קריאה, עדכון ומחיקה) על נתונים, כמו גם ביצוע לוגיקה עסקית והחזרת נתונים בפורמטים שונים כמו JSON (סימון אובייקט Javascript) או XML (שפת סימון הרחבה), או פורמטים אחרים.

בצד זה מנוהל גם ה Database שמאוחסן ב SQL Server, ונכתב באמצעות טכנולוגיית **SQLAlchemy** בשפת Python:

SQLAlchemy היא ערכת כלים של SQL בקוד פתוח וספריית מיפוי יחסי אובייקט (ORM) לשפת התכנות Python. הוא מספק קבוצה של כלים לעבודה עם מסדי נתונים יחסיים, מה שמקל על אינטראקציה עם מסדי נתונים באמצעות קוד Python.

צד לקוח:

נכתב בשפת JavaScript בטכנולוגיית **React**:

React היא ספריית JavaScript בקוד פתוח המשמשת לבניית ממשקי משתמש (UIs) עבור אפליקציות אינטרנט ונייד. הוא פותח על ידי פייסבוק ושחרר לציבור בשנת 2013. React מספק גישה הצהרתית לבניית ממשקי משתמש, מה שמקל על מפתחים ליצור ממשקי משתמש מורכבים ואינטראקטיביים עם פחות קוד.

React משתמשת בארכיטקטורה מבוססת רכיבים, מה שאומר שממשקי המשתמש נבנים באמצעות שילוב של רכיבים הניתנים לשימוש חוזר שניתן להרכיב בקלות ליצירת ממשקי משתמש מורכבים. כל רכיב מייצג חלק קטן מממשק המשתמש, וניתן לעשות בו שימוש חוזר בכל האפליקציה. לרכיבים יכולים להיות גם מצב והתנהגות משלהם, מה שמקל על ניהול אינטראקציות מורכבות ושינויים בממשק המשתמש.

פרוטוקול התקשורת:

מערכת זו מיישמת את פרוטוקול התקשורת HTTP.

HTTP הוא פרוטוקול המשמש להעברת נתונים דרך האינטרנט. זהו הבסיס לתקשורת נתונים ברשת העולמית והוא אחראי על אספקת דפי אינטרנט ומשאבים מבוססי אינטרנט אחרים משרתים ללקוחות, כגון דפדפני אינטרנט. דרך פעולתו היא על ידי יצירת חיבור בין הלקוח לשרת ולאחר מכן החלפת בקשות ותגובות ביניהם. כאשר לקוח מבקש משאב, הוא שולח בקשת HTTP לשרת. לאחר מכן השרת מעבד את הבקשה ושולח תגובת HTTP חזרה ללקוח, הכוללת את המשאב המבוקש.

פרוטוקול זה מיושם בפרויקט עבור רוב החיבורים בין השרת ללקוח או ל solver, באמצעות פונקציות CRUD- פונקציות בסיסיות לתפעול הנתונים.

תיאור מסד הנתונים ויחסי גומלין:

