**מבוא**

הדרישות כוללות מערכת מבוססת Node.js עם Express לצד השרת, ובסיס נתונים MongoDB. היכולות כוללות מודלים שונים, שאילתות חיפוש, יכולות למנהל ולמשתמשים, וכן הרבה פיצ'רים אחרים.

**Node.js ו-Express**

* יצירת שרת באמצעות Node.js ו-Express שיכיל את כל הנתבים הדרושים לטיפול במשימות CRUD (Create, Read, Update, Delete) עבור מודלים שונים.

**MongoDB**

* השימוש ב-Mongoose ליצירת מודלים בסיסיים שיחזקו בפונקציונליות של CRUD, וכן בעריכת שאילתות מתקדמות.

**MVC**

יישום מודל MVC כך שיהיה ניתן להפריד בין המודלים (M), התצוגות (V) והבקרים (C).

* Models (M)- מגדירים את המבנים של הנתונים, העסקים והחוקים התקנים של   
  האפליקציה. הם אחראים לקריאות וכתיבות מהבסיס נתונים ולכך שהמידע יתעדכן כראוי באפליקציה.
* Views (V)- מציגים את המידע למשתמש ואחראים לכל סוגי הפלט: טקסט, גרפים, כפתורים וכו'. בדרך כלל, התצוגות לא מבצעות פעולות על הנתונים עצמם, אלא רק מציגות אותם.
* Controllers (C)- משמשים כגשר בין התצוגות והמודלים. הם מטפלים בקלט שנקלט מהמשתמש ובוחרים איזה פעולות יש לבצע במודל ואיזו תצוגה יש לעדכן כתגובה.

**ממשק משתמש**

* השימוש ב-jQuery ו-AJAX לטעינה דינמית של תוכן ועדכון הממשק ללא צורך בטעינה מחודשת של הדף.
* השימוש ב-HTML5 ו-CSS3 לפיצ'רים כמו וידאו, כותרות, ועיצוב דינמי.

**הרשאות ואבטחה**

* יישום מנגנון התחברות מאובטח, השימוש בmiddleware לבקרה בהרשאות.

**פיצ'רים נוספים**

* + הטמעת Google Maps או Bing Maps, התממשקות ל-API של טוויטר או פייסבוק, ומידע סטטיסטי באמצעות גרפים מ-D3.js.

**ולידציה ובדיקות**

* + הוספת ולידציות ובדיקות יחידה ואינטגרציה לוודא תקינות ואיכות הקוד.