

# מסמך פרויקט React

## פרק 1: ייזום

o תיאור כללי - הרעיון הוא דף בית באתר אינטרנט. דף הבית מתקשר עם מגוון API's לקבלת הנתונים שאותם הוא מציג למשתמש. דף הבית כולל כמה פיצ'רים שאותם אסביר בהמשך.

o מטרת הפרויקט - פרויקט ב-React.

o טכנולוגיות עיקריות וספריות חיצוניות -

- a. המתודולוגיה העיקרית שעליה מושתת הפרויקט היא "Class-based Component". בעצם הרכיב המרכזי של האפליקציה כולל בתוכו states שהם תאי המידע שחוזר מהAPI. בתוך class הראשי "App" ישנן פונקציות שקוראות לAPI ומגדירות את המידע על גבי הstates. בחרתי בשיטה זו כדי להשתמש במחזור החיים של הקומפוננטה שלאחר הרינדור הראשון (componentDidMount) ייקראו הפונקציות שמקבלות נתונים מAPI ויעדכנו את הstates בהתאם.
- b. בפיצ'ר Todo List כדי להשתמש בניהול states יותר יעיל החלטתי להיעזר בספריה redux. ראשית נסתכל על TodoListSlice.js שם יצרתי slice שבו הגדרתי את המצב הראשוני והגדרתי פעולות שמכילות את ה-action שהם בעצם reducers. כל הנתונים האלה מוחזקים בעצם באובייקט store שבמידת הצורך מזרים אותם לשאר חלקי האפליקציה שצריכים את אותו מידע.
- c. עוד טכנולוגיה שהשתמשתי בה בפרויקט היא hooks. ודוגמא לכך היא באובייקט imageUrl. בעזרת useState הגדרתי מספר שמייצג את מספר התמונה במערך התמונות. כדי שבפני המשתמש יוצג מערך התמונות אותו מספר שהגדרתי מתעדכן כל 10 שניות למספר רנדומלי אחר בעזרת useEffect.

## פרק 2: אפיון

o פיצ'רים ותכונות -

1. המשתמש יוכל להתעדכן במזג האוויר - על ידי שימוש

בOpenWeatherMap API.

2. מוצג למשתמש תמונות מעוררות השראה - על ידי שימוש בUnsplash

API

3. המשתמש יוכל לראות ציטוט מעורר השראה - על ידי שימוש

בAdviceSlip API.

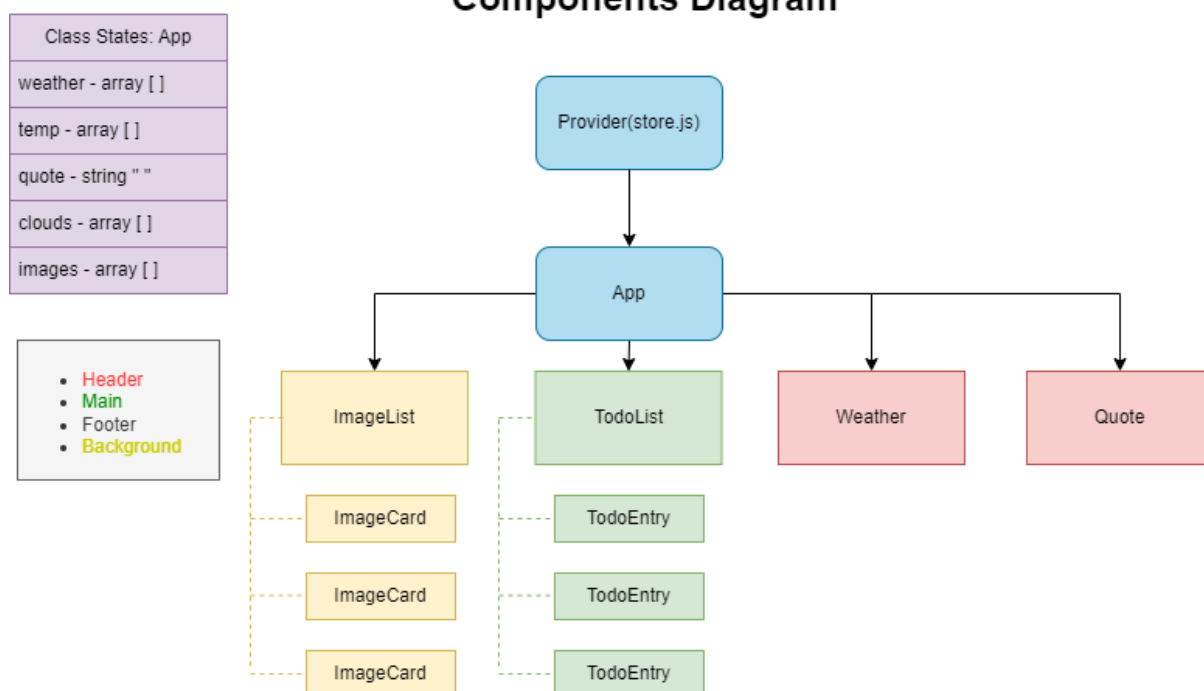
4. המשתמש יוכל לרשות את המטלות שלו

5. המשתמש יוכל למחוק את המטלות שלו

6. המשתמש יוכל לסמן שמטלות נעשו

o מצורף תרשים ארכיטקטורה של הקומפוננטות וStates:

### Components Diagram



### פרק 3: נקודות לשיפור

אני תמיד חושב שיש מקום לשיפור לכן הנה כמה נקודות לשיפור ולמידה:

- מבנה Components Tree - בחרתי בשיטת class based components למרות השיטה המקובלת יותר שהיא function-based components כי חשבתי שזה בסיס טוב יותר למקרה ויהיה צריך להרחיב את האפליקציה ולהוסיף לה עוד פיצ'רים ופונקציונליות. במקרה הזה אולי להשתמש בשיטת function-based components היה יותר יעיל גם מבחינת ניהול ה States (שהיה אפשר להגדיר לכל פיצ'ר בנפרד במקום שכל ה States יהיו ב App) וגם מבחינת הבנת הקוד.