

ניהול מבחנים

פרוייקט FULLSTACK - מדעי המחשב תל-חי

לוגאין\לוגאאוט

- מסך לוגאין עבור מרצה או סטודנט
- לאחר לוגאין כל הפעולות יהיו תחת שם המשתמש

בניית מבחן עי מרצה\מורה

המרצה יוסיף בחינה ואת השאלות שבבחינה
כך שכל נתוני הבחינה והשאלות ישמרו כקובץ JSON

נתוני בחינה:

- שם
- ID
- תאריך
- שם מורה\מרצה
- שעת התחלה
- זמן כולל
- האם סידור ראנדומאלי
- ציון בחינה

כל בחינה תכלול מספר שאלות וכל שאלה תכלול רשימה של תשובות אפשריות (מבחן אמריקאי)

ומתוכן התשובה הנכונה

- המבחן יכיל מספר רשימת שאלות
- כל שאלה תכלול מספר אפשרויות לבחירת תשובה ותשובה אחת נכונה
- כל שאלה תאפשר הצגה של שאלת טקסט או שאלת תמונה
- האם סידור ראנדומאלי של אפשרויות בשאלה
- האם הסידור ראנדומאלי בסידור השאלות

אפשרויות מרצה:

- הוספת בחינה חדשה ואפשרות לשמור את המבחן לוקאלית עד אשר מעדכנים אותו בשרת
- המרצה יוכל לערוך ולעדכן את נתוני הבחינה והשאלות
- בסיום בניית השאלות יעלה המרצה את נתוני הבחינה לשרת
- לאחר ביצוע בחינה לא יוכל המרצה לעדכן בחינה
- חיפוש בחינה בשרת לפי שם והצגתו
- הצגת סטטיסטיקה בבחינה של כל הסטודנטים

הרצת בחינה עי סטודנט

- סטודנט יראה רשימה של כל המבחנים שהוא ביצע
- סטודנט יתחיל ביצוע בחינה לפי התאריך והמועד המתאים שנקבע לבחינה בלבד
- לפני ביצוע תחילת בחינה הסטודנט יזין את הפרטים שלו ואת תעודת הזהות שלו
- כל שאלה תוצג בדף מסך נפרד ויהיו כפתורים קדימה ואחורה לניווט השאלות
- בכל שלב בבחינה יוצג כמה שאלות יש סהכ, ואיזה שאלות קיים פתרון וכמה שאלות נותרו ללא פתרון
- יש להציג כמה דקות נותרו לבחינה

בסוף ביצוע הבחינה יש לחשב את הציון שהתקבל ולשמור את הפרטים הבאים בבסיס הנתונים בשרת:

- תעודת זהות הנבחן
- שם הנבחן
- ID הבחינה
- ציון
- רשימת שגיאות
 - שם שאלה
 - תשובת טעות שנבחרה
 - תשובה נכונה

דרישות טכנולוגיות צד שרת

- ASP.NET API OR NODEJS

דרישות טכנולוגיות צד CLIENT

- JS OR REACT