

21/11/2017

שלי הפתרון:

1. יצירת בסיס הנתונים ב-mysql.
2. כתיבת צד שרת REST API ב-php, לצורך ביצוע פעולות CRUD מול בסיס הנתונים (יצירה, קריאה, עדכון, ומחיקה). בנוסף, כתיבת ממשק לפעולת חיפוש לפי שם פרטי של סטודנט.
3. כתיבת התוכנה לצד הלקוח ב-AngularJS Material.

הערות / נקודות לשיפור:

1. רצוי להוסיף routing ב-php או בשרת המארח עצמו (באמצעות קובץ htaccess) כדי להציג שורת URL מובנת יותר.
2. לשקול שימוש ב-Stored procedures ב-mysql לצורך שליפה ועדכון של הנתונים, במקום להשתמש בקוד קשיח שנכתב בדפי ה-php.
3. לממש בדיקת תקינות מספר ת.ז גם בצד השרת (php) בשלב של הוספת סטודנט חדש, כדי להבטיח את תקינות ומהימנות הנתונים.
4. להוסיף paging לדף שמציג את הסטודנטים ולהגביל את מספר התוצאות המוצגות בדף, כדי להתמודד טוב יותר עם מערכת שיש בה כמות גדולה של נתונים.
5. חיפוש לפי שם פרטי של סטודנט מומש רק בצד הלקוח – מבצע סינון על נתוני הטבלה המלאה שנשלפו מהמאגר. קיימת תשתית לחיפוש בצד השרת באמצעות api שנכתב ב-php, לצורך שימוש עתידי.

קובי שמיר