

פרויקט קורס: איפיון ופיתוח שרת מבוסס nodejs ו-MongoDB ושימוש ברכיבי חומרה בטלפון חכם

מטרתנו של פרויקט זה הוא לבנות אפליקציה רשת חברתית מבוססת מיקום למכשירים ניידים. התוצר הוא מערכת end-to-end הכוללת שרת מסוג nodejs מסד נתונים מסוג MongoDB ואפליקציה טלפון חכם העושה שימוש ברכיבי החומרה של הטלפון. פרויקט זה תיבנה בשלבים כאשר כל שלב הוא תרגיל הגשה. המערכת, ככל מערכת, תאופיין (חלק א' של הפרויקט) ותכיל פיתוח צד שרת (חלק ב'), פיתוח צד לקוח (חלק ג') ושימוש ברכיבי חומרה של הטלפון החכם (חלק ד').

הנחיות כלליות לביצוע הפרויקט:

- הדרישות המופיעות להלן הינן דרישות **המינימום**. ישנו מקום נרחב ליצירתיות, הרחבה והעמקה על מנת לשרת טוב יותר את הארגון לו אתם בונים את האפליקציה.
- הפרויקט יבוצע בקבוצות של שלושה סטודנטים.
- היישום יכול להבנות באנגלית או בעברית אולם יש חשיבות לשמירה על אחידות שפתית לכל אורך הפרויקט. בבחירת השפה יש להקפיד על קריאות ועיצוב נכון של הדפים.
- לכל אורך הפרויקט ניתן להניח הנחות במידת הצורך (למשל "חלק מהמידע מגיע משירותים מרשת האינטרנט, את החלק הזה לא נממש, במקומו נשתמש בפונקציה המכילה מספר סופי של נתונים מוכנים מראש"). בהנחת הנחות יש לציין אותן במפורש ולשמור על עקביות לאורך הפרויקט.
- בכל שלב בפרויקט צרפו מסמך המסביר בקצרה את ההנחות שלכם וההערות לבודק (במסמך WORD).
- ההערכה מבוססת על מקצועיות, מקוריות ויצירתיות המשרתות את מטרת היישום.

בהצלחה!

ד"ר עוזי חדד

חלק 0 (הגשה עד 12.7.13) – ללא ציון

יש להתחלק לקבוצות של שלושה סטודנטים, לבחור רעיון לאפליקציה רשת חברתית בחרים לפתח, לשייך את זה לחברה/ארגון, ולקבל אישור לכך מבודק התרגילים בקורס.

הנחיות:

1. לא יאושרו קבוצות של פחות או יותר משלושה סטודנטים.
2. רצוי שהייעוץ שנבחר יהיה יישום אמיתי. ניתן לבחור יישום דמיוני, כזה שעדיין לא קיים (למיטב ידיעתכם) אולם חשוב שהוא יהיה בעל פוטנציאל קיום.
3. הייעוץ צריך לכלול בתוכו מרכיבים של מסדי-נתונים מסוג MongoDB.
4. יש לשלוח לדוא"ל של בודק הקורס (mtadevnode@gmail.com) את בחירתכם הכוללת את שמות חברי הקבוצה, מספרי ת.ז., והייעוץ שבחרתם. יש לתאר את הייעוץ בהיקף של 2-3 משפטים.
5. כל קבוצה תבצע את הפרויקט על רעיון אחר, בשיטת כל הקודם זוכה.
6. אין להמשיך לשאר חלקי הפרויקט ללא אישור בדוא"ל חוזר מבודק התרגילים.

חלק א' (הגשה 14.7.13) – 20% מהציון הסופי - איפיון היישום

בחלק זה נבצע איפיון ראשוני של היישום. על האיפיון להיות באורך של לפחות חמישה עמודים ולתאר את הפריטים הבאים:

- דרישות פונקציונליות – מה הייעוץ אמור לעשות.
- דרישות לא פונקציונליות כולל:
 - Performance (response time estimations), Accessibility, Usability (easy, efficient to use and easy to learn)
- הסבר על הרשת החברתית של האפליקציה.
- קהל היעד והגדרת משתמשים: מאפייני תרבות, גיל, ניסיון בישומים דומים.
- הסביבה של החברה שעבורה אתם בונים את היישום.
- 2-3 תהליכים ותיאורם מבחינת התהליך וצרכי המידע שלו. יש לשלב תרחישי שימוש והגדרה של משתמשים והקשרי שימוש כחלק משלב זה.
- מבנה מסדי הנתונים של MongoDB
- ניתוח של אפליקציות חברתיות דומות, וההבדל בין האפליקציה שלכם לאלו של המתחרים.

חלק ב' (הגשה: 16.8.13) – 25% מהציון הסופי

בחלק זה נבנה את:

- א. נשתמש באפיון אותו ביצעתם בחלק א' של הפרויקט.
- ב. נבנה את בסיס הנתונים MongoDB.
- ג. נבנה את השרת עם הקוד שמבצע תהליכים עיסקאים מול MongoDB.
- ד. ממשק Rest לגישה לשרת nodejs. הנתונים ישלחו ויחזרו בפורמט של JSON.
- ה. בסיום חלק זה ישאר לכם לממש בחלק ד עבודה עם WebSocket. מעבר לזה אין צורך בעבודה נוספת בשרת.

בניית בסיס הנתונים והטבלאות יבוצע בצורה דינמית מהקוד nodejs

- בנו את בסיסי הנתונים (על MongoDB) לפי מה שהגדרתם בחלק א' (אם במהלך כתיבת חלק זה התברר לכם שיש צורך לשנות את מה שהגדרתם בחלק א', אתם רשאים לתקן זאת אך לרשום עובדה זו במסמך המצורף לחלק זה).
1. הזינו לבסיס הנתונים נתוני בדיקה (על מנת שלישום שלכם יהיה תוכן).
 2. עשו שימוש בכלי curl (או התקינו תוסף ל chrome מסוג rest client) על מנת לבדוק את הקוד שכתבתם. כמו כן שלחו לבדוק מסמך עם כל השאילתות curl מול המערכת שלכם (עם ה-DATA הרלוונטי-יוסבר בכיתה).
 3. הגדירו 10 שאילתות התומכות בתהליכים עסקיים עבור היישום שלכם (בפרט ניתן לשנות קצת מההגדרות הראשונים שכתבתם בחלק א'). ובנו דפים התומכים במערכת זו. השאילתות צריכות לכלול:
 1. 2 שאילתות find פשוטות.
 2. 2 שאילתות find מורכבות.
 3. 1-2 שאילתות UPDATE.
 4. 1-2 שאילתות save or insert.
 5. 2 שאילתות remove.
 6. 1 ensureIndex.
 4. העלו את האתר שלכם לענן של Heruko.

הנחיות כלליות:

- כל סעיף עומד בפני עצמו.
- יש לשים דגש על יעילות השאילתה, אופן ביצוע השאילתה והנתונים אותם שולפים.

קריטריוני הערכה:

- יעילות השאילתות.
- מודלריות הקוד.
- צמידות נמוכה.
- מורכבות האפליקציה.
- פונקציונליות.
- ניהול נכון של המסמכים (documents, collections) שישרו ב- MongoDB
- שימוש ועבודה יעילה עם מסד הנתונים MongoDB

חלק ג' (תאריך הגשה: 2.9.13) – 25% מהציון הסופי

עיצוב הממשק

1. בחלק זה עליכם להגדיר ולבנות את ממשק המשתמש עבור דפדפן במוביל. ממומלץ לעשות שימוש ב-jQueryMobile אך אפשר להשתמש בטכנולוגיות אחרות.
2. יש לקשר את כל הממשקים / שאילתות לדף המופיע בפתיחה האתר.
3. בדיקת נכונות השאילתות תתבצע על ידי נתונים שיוזנו על ידכם לבסיס הנתונים וכן על ידי נתונים חדשים.

הנחיות:

- עליכם לבנות היישום אשר מציג את הדפים המרכזים של היישום (כ- 10 דפים).
- יש להשתמש **נכון** באלמנטים שנלמדו ובהתאם למתודולוגיה המוצגת בקורס. אין צורך להשתמש בכל האלמנטים שנלמדו, השיקול העיקרי בבחירת האלמנטים הוא שמישות ומימוש המטרה המבוקשת.
- הדגש בחלק זה הוא על הקוד ושימושיות של האפליקציה.
- עיצוב נאה ומקצועי תמיד "ימשוך את העין" ויצור רושם ראשוני חזק.
- יש להקפיד על כתיבת קוד קריא, והפרדת עיצוב (CSS) וסקריפט (JS) מהתוכן.



קריטריוני הערכה

- פונקציונליות האפליקציה (האם כל הקישורים עובדים וכ"ו) וחוויית משתמש.
- קוד נקי ומובן.
- מודלריות.
- צמידות נמוכה.
- שימוש בסטנדרטים ומתודולוגיות פיתוח שלמדנו.
- מקצועיות.

חלק ד' (תאריך הגשה 17.9.13) – 20% מהציון הסופי -

- בחלק זה תממשו רכיבי חומרה
- (1) שילוב של מערכת לזיהוי מיקום.
 - (2) שילוב של רכיבי חומרה נוספים כגון (accelerometer, Gyroscope), יש לשלב לפחות רכיב אחד.
 - (3) שילוב WebSocket .

קריטריוני הערכה:

- יעילות השאילתות.
- מודלריות הקוד.
- צמידות נמוכה.
- מורכבות האפליקציה.
- פונקציונליות.
- עיצוב הממשק וחוויית המשתמש.
- שימוש בטכנולוגיות/ספריות שלא נלמדו.

אופן הגשה

- את כל חלקי הפרויקט באימייל לבודק
- כותרת האימייל תהיה בפורמט הבא: <id1>, <id2>, <id3> D<drill number>
- תוכן האימייל יכלול את:



- לינק לאפליקציה שלכם בענן
- שמות ות.ז של הסטודנטים המגשים
- קובץ ZIP שמכיל את כל הקבצים הדרושים להפעלת האפליקציה לאותו החלק, שבתוכו קובץ הערות לבודק.
- ה-ZIP שאתם שולחים לבודק צריך לכלול גם את הקוד וגם את הקוד כך שהבודק יוכל להריץ את התוכנית שלכם.
- בנוסף מחלק ב(כולל) עליכם לעלות את האתר שלכם לענן של Heroku

בהצלחה 😊