**דרישות פרויקט גמר בקורס אנדרואיד צד לקוח 2**

**כללי:** עליכם לממש חנות אינטרנטית בהתאם לדרישות ותוך שילוב הטכנולוגיות המופיעות בסעיף הדרישות.

**דרישות מנהליות:**

1. ניתן לעבוד בקבוצות של עד שלשות סטודנטים עם זאת, כל אחד מחברי הקבוצה **חייב להכיר את כל הקוד** שנכתב ועליו להיות מסוגל לענות על שאלות מכל חלקי הקוד במהלך ההגנה. **לא ניתן** לשלב בקבוצה סטודנטים אשר רשומים לקורס אצל **מרצים שונים. כל חברי הקבוצה חייבים להיות רשומים כולם לאותו המרצה**. באחריותכם לבדוק ולוודא לאיזה מרצה אתם רשומים ולהרכיב קבוצות בהתאם. שימו לב שבמודל כל הסטודנטים רואים את אותו האתר (לא משנה לאיזה מרצה הם רשומים), אבל כל סטודנט רשום רק לאחד מהמרצים ויכול להיבחן אך ורק אצלו. **לא יתקבלו בקשות לערבב קבוצות, ובפרט כתוצאה מטעות ואי ההבנה לאיזה מרצה אתם שייכים.**
2. מועדי ההגנות מפורטים בלוח הבחינות באתר המכללה.
3. כל סטודנט רשאי להבחן בהגנה אחת בלבד. בפרט, לא ניתן לגשת שוב או לתקן או לשפר לאחר שניגשתם כבר.
4. אישור לגשת למועד המיוחד יאושר אך ורק ע"י הממונה לענייני סטודנטים, בהצגת סיבת היעדרות משני המועדים ובצירוף אישורים וסימוכין רלוונטיים.
5. הפרויקט יוגש ע"י כל הסטודנטים עד 24 שעות לפני מועד ההגנה אליה אתם ניגשים. תיבת ההגשה תיסגר בשעה 23:59 ביום שלפני ההגנה.
6. סטודנט שלא הגיש במועד את הפרויקט – ירדו לו 20 נקודות מהציון הסופי בקורס.
7. יש להגיש את הקוד של פרויקט הקורס (לבצע Zip לקוד הפרויקט) בתיבה שנפתחה לשם כך במודל, **וכן יש להוסיף הערה להגשה במודל המכילה קישור לגיט של הפרויקט**. יש לוודא שהגיט פתוח (מבחינת גישת צפיה) לכל מי שיש לו גישה לקישור. לכן, וודאו בזמן הפיתוח שאינכם מעלים לגיט שום דבר המכיל מידע רגיש כלשהו, כמו סיסמאות, API-keys, טוקנים וכו'.
8. במעמד ההגנה, יש להפעיל את המערכת בגירסת הקוד שהועלתה למודל בלבד. אין לבצע שינויים לגרסה זו.
9. בהגנה יבדק הפרויקט, כאשר המרצה מתפעל ובודק את המערכת, את התכולה, והאם יש נפילות / באגים. המרצה יכול לשאול את כל הסטודנטים שאלות (לקבוצה ככלל ולסטודנטים ספציפיים בפרט) כשקוד המערכת פתוח בפניו, ובפרט, שאלות אודות הקוד והמימוש. מעמד ההגנה מהווה רק חלק מהציון הקשור למראה הכללי של המערכת, נפילות/באגים, שליטת הסטודנטים בקוד וחלוקת העבודה בקבוצה והעבודה כצוות. חלק נוסף במתן הציון מבצע המרצה לאחר המפגש עם הקבוצה, כאשר הוא בודק את הפרויקט בכלל ואת קוד המערכת בפרט באופן עצמאי על מנת לתת ציון סופי – על כן, לא ניתן לבשר על ציון סופי במהלך מפגש ההגנה.
10. עליכם לוודא שניתן להדגים את המערכת במעבדה שנקבעה טרם מועד ההגנה. מנסיוננו – עשויות להיות בעיות הרשאה/התקנת DB וכד', על כן וודאו זאת מספר ימים לפני ההגנה. ניתן להגיע להגנה עם לפטופ אישי, אך תקינות המחשב הפרטי והתוכנות בו – באחריות הסטודנט – על כן וודאו כי המערכת עובדת גם במחשב המעבדה למקרה של תקלה. **עליכם לוודא שהמערכת רצה על כל המחשבים האישיים שלכם, ובנוסף, שהיא מוכנה להצגה במידת הצורך במהלך ההגנה.**
11. חובה להשתמש בטכנולוגיות, שפות וארכיטקטורות שנלמדות בקורס (ובהן בלבד) ובאופן שבו הן נלמדות – ולא ניתן להשתמש בחלופות אחרות במקומן. סעיף זה רלוונטי במיוחד עבור סטודנטים הבוחרים לעבוד על הפרויקט שלא במקביל לקצב העברת החומר בכיתה. לא תוכלו להשתמש בטכנולוגיות שלא נלמדו בכיתה או שלא אושרו לשימוש.
12. חלק מדרישות הפרויקט הינן דרישות במסגרת למידה עצמית. כלומר, בעזרת החומר הנלמד בכיתה ניתנו לכם כלים, ובעזרתם עליכם יהיה ללמוד עצמאית כיצד לבצע דרישות אלו.
13. הפרויקט והקוד בו חייבים להיות פרי עבודה אישי של הקבוצה שלכם, ולהיכתב על ידכם. במהלך בדיקת הפרויקט יהיו בשימוש כלים ידניים ואוטומטיים לזיהוי קוד שאינו פרי עבודה מקורית של חברי הקבוצה.
14. דרישות המינימום עבור הפרוייקט מפורטות בהמשך. עבור ציון גבוה יש להגיש פרויקט מלא המציג מערכת שלמה ומתפקדת (מעשית ולוגית), **אשר מכיל מעבר לדרישות המינימום**.

**דרישות טכניות:**

1. תשתית המערכת תבוסס Node.JS בצד השרת בשימוש ב Express ואילו בצד הלקוח נבצע את הפרויקט ב react או react native או android kotlin ( שילוב של הפלטפורמות יזכה את הקבוצה בניקוד בונוס )
2. אחסון ושליפת הנתונים מתוך MongoDB או מתוך REST API – שילוב ביניהם יזכה בבונוס
3. על המערכת לתמוך בלפחות 3 מודלים שונים. (למשל: מודל להצגת הפריטים, מודל להכנסת שם משתמש וסיסמא, מודל לעגלת קניות וכד ... )
4. על **כל אחד** מהמודלים, המערכת צריכה לתמוך ב Create (יצירת אובייקט חדש), Update (עדכון תוכן האובייקט – למשל עדכון פרטי מוצר), Delete (מחיקת אובייקט – למשל מחיקת מוצר), List (רשימת אובייקטים: למשל רשימת מוצרים), Search (חיפוש אובייקט על פי קטגוריות שונות – למשל מצא את המוצרים במחיר מסויים ). שימו לב, **לא** מספיק שהקוד יכיל את המימוש, אלא האפליקציה צריכה לאפשר למשתמש לבצע את **כל אחת** מהפעולות הללו על כל אחד מהמודלים.
5. המערכת תתמוך בלפחות 2 שאילתות חיפוש המאפשרות ללקוח להגדיר פרמטרים (לפחות 3 פרמטרים) לחיפוש באמצעות ממשק המשתמש. (לדוגמא, כמו בזאפ: חיפוש טלויזיה על פי גודל מסך, רזולוציה, ושם היצרן). סעיף זה הוא **בנוסף** לדרישת החיפוש של סעיף 21.
6. בונוס : המערכת תכיל את היכולת למשתמשים מסויימים לנהל משתמש . הרשאות הגישה של המשתמשים השונים ינוהלו באמצעות שם משתמש וסיסמא. למנהל קבוצה יהיו יכולות עריכה וחיפוש מורחבות לעומת משתמש רגיל. יש להקפיד על מנגנון הרשאת הגישה כך שעמודים ופונקציונאליות של ניהול יהיו נגישים רק למי שיש לו הרשאות. בפרט, כל משתמש יכול לראות רק את המידע הפרטי שלו ולא של משתמש אחר. למשל: רק מנהל יכול להוסיף לאשר הצטרפות לקבוצה, כל משתמש יכול לערוך/למחוק רק פוסטים שהוא פירסם, משתמש לא יכול לראות פוסטים בקבוצה פרטית שהוא לא חבר בה וכו׳.
7. כל משתמש יוכל לראות את כל המוצרים שקנה, ומה שעדיין נמצא אצלו בעגלת הקניות – ממתין לקניה
8. **חובה** לטפל במקרי קצה, שגיאות, ולידציות וכו' – הן בצד השרת והן בצד הלקוח, ובפרט, להימנע מקריסה של השרת, גם בעת הזנה של קלט חריג והתנהגות לא צפויה של המשתמש.
9. המערכת תתמוך ב react ותעשה שימוש ביכולות הבאות:
   1. Video
   2. Canvas
10. המערכת תתמוך ב CSS 3 ותעשה שימוש ביכולות הבאות:
    1. Text-shadow
    2. Transition
    3. Multiple-columns
    4. Font-face
    5. Border-radius
11. בונוס: להכיל רכיב במערכת צ׳אט בין משתמשים, אשר מעביר את הנתונים בין השרת ללקוח באמצעות Socket.io / WebSockets
12. ​​כמו כן:
    1. המערכת תציג נתונים סטטיסטיים בלפחות 2 גרפים (לדוגמא – מספר הפוסטים הממוצע בקבוצה פר חודש) באמצעות החבילה: D3.JS. על הנתונים המוצגים להגיע מה DB ולהציג מידע דינמי בהתאם למידע הקיים ב DB ברגע נתון.

**בהצלחה!**