**חפיפת הכנה לטכנולוגיות אפליקציית אלמונסק**

**1 .הישגים נדרשים:**

* שליטה מקצועית בטכנולוגיות ושפות הפיתוח של הפרויקט.
* עבודה על פרויקט בעזרת Git ושימוש בכלל הכלים הרלוונטיים אשר הוא מציע.
* התיישרות לסטנדרט כתיבת הקוד של הצוות
* התנסות במתודולוגיית Agile על כל הכלים והטקסים שהיא מכילה.
* הכרת ענף "אבן מתגלגלת" על כלל מוצריו ומדוריו לעומק.
* הגברת תחושת המסוגלות לפיתוח תכולות מבצעיות.

**2 .הליך החפיפה:**

* תיאום ציפיות לחפיפה עם החונך.
* העמקה מקצועית בשפות הפיתוח JS , TS .
* עקרונות SOLID ו- .Clean Code
* תיאום ציפיות לפרויקט ולמידת עקרונות .Agile
* למידת טכנולוגיות צד שרת וDB-.
* כתיבת צד השרת לפרויקט החפיפה.
* למידת טכנולוגיות צד לקוח.
* כתיבת צד הלקוח לפרויקט החפיפה.
* סיכום החפיפה.

**3 .תיאום ציפיות לחפיפה:**

* לגוגל יש תשובה לכל שאלה, אם מושג או תחום לא מובן מספיק מצופה מכם לחקור ברשת לפני שתשאלו את הצוות.
* בניגודיות מוחלטת לנקודה שמעל, תרגישו חופשי לפנות לצוות בנוגע לכל שאלה, אנחנו כאן כדי לחנוך וללמד אותכם.
* עד כמה החפיפה תהייה עמוסה ומעמיקה תלוי לגמרי בכם, אם תרגישו שאתם בטוחים בנושא מסוים תרגישו חופשי לרפרף עליו, כמובן בשיח עם החונך.
* תגדילו ראש לגבי התכולות של הפרויקט חפיפה ותשקיעו בעיצוב.
* לעיתים זה ירגיש שלכאורה אין עלייכם לחץ. נהפוכו. הצוות מחכה שתצטרפו לעשייה, כמה שתיתנו מעצמכם בתקופה הזאת יהפוך אותה לקצרה יותר ויכניס אותכם לצוות בצורה מהירה הרבה יותר, מקווים שתשבורו את השיא!

**4 .העמקה מקצועית בשפות הפיתוח:**

**Javascript:**

ביצוע קורס Udemy שיסופק ע "י הצוות. (באזרחי לבקש מאלעד)

יש לבצע את פרקים 1 – 10 .לא כולל פרקים 5 ,7 ,8 .

יש לתת דגש וקשב לנושאים הבאים:

* Lexical scope, Lexical this
* Async/Await and Promise
* ES6 Features

**Typescript**

העמקה מקצועית בדוקומנטציה של Ts ומענה על השאלות הבאות:

1) למה צריך להשתמש בTs?

2) האם אנו מחויבים להשתמש בכלל החוקים שהשפה דורשת?

כיצד ניתן להימנע מחוק מסוים? ספק כמה דוגמאות.

3) סכם על סוגי הTypeים הקיימים

4) ספק הסבר קצר על המושגים הבאים:

* Union
* Generic
* Intersection
* Keyof
* Typeof
* Partial
* Required
* Readonly
* Record
* Pick
* Omit

**5. עקרונות SOLID ו- Clean Code:**

חפשו ביוטיוב את הסרטון הבא:

[Clean Code: The Next Chapter by Victor Rentea](https://www.youtube.com/watch?v=wY_CUkU1zfw)

עלייך לסכם בקצרה על כל אחד מעקרונות הSOLID .

**6****. פרויקט חפיפה – EvenSoldiers**

**הצורך המבצעי:**

מרוב כל התקנים שענף "אבן מתגלגלת" הרוויח ביושר בשנים האחרונות

נוצר מצב של חוסר שליטה באנשים המאיישים את התפקידים.

על כן, הדרג הניהולי הבכיר החליט לחפש תוכניתנ/ית אשר יוכלו לתת

מענה למיפוי כלל המשרתים בענף עם כמה שיותר מידע על כל אחד מהם.

**המענה המבצעי:**

עליך לייצר מערכת עם אפליקציות Web ו Mobile אשר בעזרתם נוכל

למפות את הענף על כלל בעלי תפקידיו ומיקומם.

המערכת תתוכנן באופן הבא:

**אפליקציית Mobile מבוססת React Native – אשר תכיל:**

אפשרות להתחברות, הרשמה, תיעוד וצפייה בתיעוד שיכילו:

1. מידע בסיסי אישי – שם, תאריך לידה, מין, כתובת, מספר טלפון, כתובת מייל
2. מידע הקשור לשירות הצה"לי – מספר אישי, דרגה, תאריך גיוס , תפקיד, מדור, צוות

**אפליקציית Web מבוססת React – אשר תכיל:**

אפשרות להתחברות, הרשמה, צפייה בכל התיעוד שצולם במובייל.

**צד שרת מבוסס NestJs – אשר יכיל:**

את כל הדגשים שנלמד בנושא ושמירת כל המידע ב Postgresql – תוך שימוש ב Liquibase.

**7. למידת טכנולוגיות צד שרת ו- DB:**

**NestJs**

קריאת הדוקומנטציה של הספרייה ומענה על השאלות

הבאות:

* מה זה ExpressJs ? מה הקשר בינו לבין NestJs?

**Controller**

* איך מגדירים Route ?
* לאילו אלמנטים של הבקשה ניתן לגשת מהפונקציות בController ? איך אפשר לבצע זאת?
* איך מחזירים סטטוס מסוים עבור Route?
* כמה מופעים נוצרים ל Controller באפליקציה? האם בקשות שונות חולקות את אותו המופע?

**Provider**

* אילו סוגים של אובייקטים יכולים להוות Provider ?
* מהו Provider Token?
* תן דוגמה לשימוש בuseFactory .

**Modules**

* הסבר בקצרה את המושג Module. למה הוא משמש?
* כמה Modules יש באפליקציה של Nest?
* ציין את ה properties הקיימים באובייקט ש @Moduleמקבל כפרמטר, מה משמעותם?
* הסבר על המושג ()Global@

**Tests**

* קרא באינטרנט על סוגי הטסטים השונים, הסבר בקצרה על . Unit Test, E2E Test, Intergration Test
* למה שנשקיע את הזמן שלנו בטסטים? ציין שלוש סיבות לפחות.
* קרא בהרחבה על טסטים ב –Nest, איזו ספריית טסטים הכי מומלצת?

**PostgreSQL ו- Liquibase**

PostgreSQL – חפיפת PostgreSQL מה GitLab האזרחי המדורי.

Liquibase – מהאתר הראשי + צפייה בשימוש בו במערכות המדור (למשל שביט).

**8. פרויקט חפיפה – צד שרת**

אחרי שלמדתם על טכנולוגיות צד השרת שלנו בצורה בסיסית, עליכם לייצר

את צד השרת של פרויקט החפיפה.

שימו לב שמצופה מהמערכת שלכם לדעת לקבל בקשות בהתאם לנדרש ממכם

בפרויקט ולנהל אותה.

ההגדרה של הסיום והמשך לשלב הבא בחפיפה היא האפשרות לפנות

לשרת ולקבל ממנו תשובה על כלל הבקשות הנדרשות.

דגשים לכתיבת צד השרת:

* שימוש מעמיק בכלי Typescript ,כולל תכנון נכון של כלל האובייקטים שידרשו גם לצד לקוח.
* שימוש ב Pipe Validation בכל Route של המערכת.
* מימוש Guard שאוכף JWT בכל Route של המערכת.
* מימוש Middleware של לוגים, תיעוד של כל פעולה שמתבצעת וסיווג הLog לפי קטגוריה: שגיאה, אזהרה, מידע.
* שמירה על השכבות ומימוש כל לוגיקה במקום ההגיוני שלה.
* E2E Test לכל Route.
* Unit Test לכל פונקציה ב Service
* יצירת סביבות פיתוח ומשתני סביבה בקבצים רלוונטיים.
* תפיסה, טיפול ותיעוד של כל שגיאה.
* כל הקוד מנוהל בGit .
* סטנדרט קוד לרבות:
  + שמירת Constים גלובאליים במקום רלוונטי.
  + קובץ Utils עם פונקציות עזר גלובאליות.
  + אין בשום מקום במערכת @ts-ignore ולא as (אופרטור של ts) בלי הסבר מפורט ללמה הייתם צריכים.
  + פונקציות קצרות שמבצעות פעולה אחת בלבד בלי אפקט צדדי.
  + שמות משתנים רלוונטיים.

אישורי תוכניות: (אחריותכם לקבוע עם החונך(

* אישור תכנון אובייקטים מול החונך.
* אישור תכנון מודולים מול החונך.
* אישור תכנון עץ תיקיות מול החונך.

**9. למידת טכנולוגיות צד לקוח:**

**React**

העמקה בדוקומנטציה של ריאקט ומענה על השאלות הבאות:

* מה הייחודיות והיתרון של ספריית ריאקט לפיתוח ?Web

מהם העקרונות לפיהם ריאקט עובד (virtual DOM, lifecycle)…

* הסבירו בקצרה לתפיסתכם את הקונספט של קומפוננטה.
* הסבירו בקצרה על Hooks, הסבירו בהרחבה על useState וuseEffect .
* הסבירו מה זה קומפוננטה שהיא Memo?
* תנו דוגמה למתי אשתמש ב useCallback וב useMemo הסבירו מה יקרה אם לא אשתמש.
* הסבירו בהרחבה על HOC תן דוגמה לHOC .
* הסבירו על Redux בהרחבה, איזה מענה אנו מקבלים דרכו לאיזו בעיה.
* הסבירו על redux-thunk,מה מטרתו ואיך הוא משתלב ב Store שלנו.
* מה ההבדל בין Redux State ל useContext ? ציינו דוגמאות לשימוש בכל אחת מהן.
* מה ההבדל בין props לstate בקומפוננטה?
* הסבירו בקצרה על המושג React Fragments .
* הסבירו על react-native-extended-stylesheet

**React Native**

תחילה יש להרים סביבת פיתוח של נייטיב במחשב. לאחר הקמת סביבת הפיתוח, הכינו מצגת לימודית להסברה לצוות על כלל התהליכים שמתבצעים המאפשרים כתיבת קוד Javascript והרצתו על מנוע Android וiOS במקביל.

**10. פרויקט חפיפה – צד לקוח**

עכשיו הגענו לחלק האמנותי, פיתוח צד הלקוח של הפרויקט שלכם.

לשם כך תצטרכו לפתח אפליקציות Web ו Mobile אשר כל אחת תבצע את

תפקידה לפי האפיון שיהיה קיים ב TFS ויסופק לכם ע"י החונך שלכם.

בסוף זמן הפיתוח המערכת צריכה להיות פונקציונלית ועובדת, ועליכם

להשתמש בה על מנת לתעד את כל הענף, אז לטובתכם עדיף שלפחות

החווית משתמש שלה תהייה ברמה גבוהה.😜

**דגשים לכתיבת צד הלקוח:**

* צרו את הפרויקט בעזרת create-react-app עם טמפלייטTypescript בלבד.
* כתיבת קומפוננטות פונקציונליות בלבד בסטנדרט הנהוג.
* שימוש בhooks בחוכמה ע"פ מה שהקומפוננטה דורשת, יהיה עלייך לדעת להסביר גם למה השתמשת.
* שימוש ב Redux Persist במובייל על מנת לשמור את ה State גם כשאפליקציה נסגרת.
* שימוש ב formikעם yup לכל הinputים כולל ולדיציות של הקלט.
* עיצוב קומפוננטה התבצע רק עם אובייקט Style ולא עם css .
* פיצול של קוד בקומפוננטות.
* בלי רנדורים מיותרים, נבדוק את זה.
* שימוש בRedux , Stores שבנויות בתבנית מוגדרת של הסטנדרט שקיים כרגע בצוות.
* בכל עבודה עם תאריכים וזמנים נשתמש ב date-fns
* להשתמש ב react-navigation במובייל וב react-router ב web .
* אין שום error/warning בconsole אלא אם כן יש הסבר משכנע לכך.
* כמה שפחות לוגיקה בעצמכם, לא לפחד להשתמש בספריות קיימות, יחד עם זאת להסביר למה דווקא הספרייה הנבחרת ולא המתחרים.
* קבצי Config ו Utils בדיוק כמו בשרת, כולל HttpConfig שמנהל את כל הבקשות.
* קבועי מערכת בקבצים מסודרים, כולל צבעי המערכת שנקבעו.
* סביבות פיתוח ומשתני סביבה.
* חיווים רלוונטיים למשתמש על כל שגיאה לרבות קלט לא נכון ופופאפים של שגיאות.
* שימוש בספריית Material-UI ב web וב React Native Paper במובייל.
* השקעה נרחבת בעיצוב וחווית משתמש.

אישורי תוכניות: (אחריותך לקבוע עם החונך(

* אישור תכנון קומפוננטות מול החונך.
* אישור תכנון עיצוב מול החונך.
* אישור תכנון עץ תיקיות מול החונך.