***devstegram***

**מסמך עיצוב**

**איתי קבנובסקי**

*<28.12.2020>*

**היסטוריית גרסאות המסמך**

|  |  |  |
| --- | --- | --- |
| **תאריך** | גרסה | **תקציר השינויים** |
| 28.12.2020 | 1 | מסמך ראשוני |
|  |  |  |
|  |  |  |
|  |  |  |



**1. הקדמה**

החלק הזה ישמש להצגה כללית של מסמך העיצוב. הסבר כללי, וקישור למסמכים קודמים כמו מסמך האפיון עליו אנו מתבססים

1.1 מטרה

*מה המטרה של המסמך ומי קהל היעד שלו*

*מטרת המסמך לרדת לפרטי התכנות של הפרוייקט ולהתחיל לפרק אותו לחלקים הקטנים שממנו הוא מורכב.*

1.2 המוצר

*חלק זה צריך להכיל את הדברים הבאים (ניתן לתמצת מתוך מסמך האפיון):*

* *שם המוצר אותו מפתחים*
* *הסבר כללי של מה המוצר עושה*

*שם המוצר הוא devstegram.*

*מטרתו היא לספק פלטפורמה חברתית ב theme של מתכנתים.  
היא תלמד על המשתמש ובהתאם תמליץ לו תכנים.*

1.3 קישור למסמכים קודמים

*קישור למסמכים / קבצים חיצוניים הרלוונטיים למסמך. לדוגמא: מסמך אפיון, עמוד web עם עיצוב הממשק הכללי שאתם מתכננים וכו'.*

<https://github.com/itaykbn/cyber-senior-project>

1.4 הגדרות

**למידת מכונה -** למידת מכונה היא תת-תחום במדעי המחשב ובבינה מלאכותית המשיק לתחומי הסטטיסטיקה והאופטימיזציה. התחום עוסק בפיתוח אלגוריתמים המיועדים לאפשר למחשב ללמוד מתוך דוגמאות, ופועל במגוון משימות חישוביות בהן התכנות הקלאסי אינו אפשרי.

1. ארכיטקטורת המערכת

חלק זה כולל את תיאור מבנה המערכת ופירוט המודולים השונים בה

2.1 מבט על

בסעיף זה יש להסביר את החלוקה העיקרית למודולים השונים במערכת, והיחסים/קשרים ביניהם. יש להגדיר באופן ברור את תחום האחריות של כל מודול או רכיב, ולאיזו פונקציונליות (כפי שהוגדרה במסמך האפיון שנכתב לפני כן) הוא מתקשר (בין אם מודול אחד בלבד או מספר מודולים יחד).  
חשוב להסביר לא רק מה/איך נעשה אלא גם מדוע דווקא כך.

**הפרוייקט יחולק לכמה חלקים כאשר לכל חלק תפקיד:**

החלק הראשון - הירשמות והתחברות של המשתמש לפלטפורמה. בהתחברות הראשונה יבחר לו קטגוריות רצויות. לאחר מכן המידע יוכנס לדאטאבייס.

החלק השני – מודל למידת המכונה יחשב את התכנים שהמשתמש מעויין בהם וישיג פוסטים שהוא לומד שמתאימים לקטגוריה הזו שהמשתמש מתעניין בה.

החלק השלישי – אפשרות לעקוב להתכתב לעשות לייק

החלק הרביעי – תחזוקה על בסיס מחזורי של תחומי העיניין של המשתמש ( עידכונם בדאטאבייס) וחישוב מחדש של המודל לפי תחומי העיניין.

החלק החמישי -המשתמש גולש בהום פייג ורואה את התכנים שהמודל המליץ לו

2.2 פירוט רכיבי המערכת

**כאן יופיע פירוט לכל רכיב (ניתן לעשות באיטרציות – כל פעם עבור הרכיבים הרלוונטיים, לפני תהליך הפיתוח שלהם)**  
בסעיף זה יש לפרט את המבנה הפנימי של כל מודול/רכיב - כלומר:

* מבנה המחלקות השונות המרכיבות אותו, המאפיינים שלהם (public/private/protected + אילו טיפוסי משתנים כל פונקציה מקבלת ו/או מחזירה) והקשרים ביניהם (מי קורא למי, משתמש במי ולאיזה צורך) – מומלץ לצייר גרף UML (לא חייבים לפרט משתנים/פונקציות פרטיות אבל שיהיה ברור מה קשור למה ובאיזה אופן או לאיזה צורך)
* יש לתאר את ה- data flow / use case של שימושים שונים בכל תת-רכיב או פונקציה (מה קורה אם מתקבלים סוגים שונים של ערכים, איך/מדוע/באילו מקרים הרכיבים מתנהגים בצורה מסוימת או אחרת וכן הלאה)

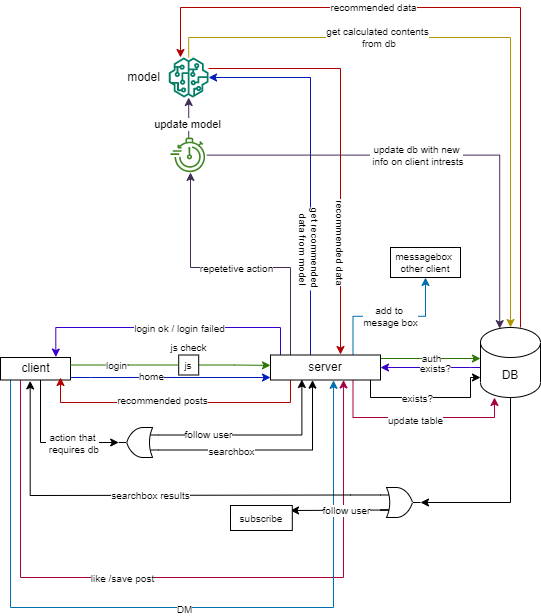
מצורף תרשים UML:

**המלצות:**

* כדאי לתכנן רכיבים תשתיתיים עבור צרכים גלובליים במערכת - כמו למשל יצירת תקשורת רשתית, כתיבה וקריאה לבסיס נתונים (או קבצים), כתיבת תיעוד (log) הפעולות של המערכת (מקובל להגדיר "רמה" לכל שורה שנכתבת - debug, info, warning, error כדי שיהיה קל לסנן בזמן ריצת התוכנית. בנושא זה – מוזמנים להסתכל על מודול logging בפייתון).
* מומלץ לתכנן את הרכיבים באופן שיאפשר אבסטרקציה ומודולריות – בכדי לאפשר הרחבה עתידית ותחזוקה קלה. אם המימוש הפנימי של רכיב או מודול מסוים משתנה אז שאר הרכיבים/מודולים אינם מושפעים או נפגעים מכך ואין צורך לשנות גם אותם, או כאשר רוצים להוסיף פונקציונליות/רכיב חדש אז אין צורך לשכתב את כל שאר המערכת במיוחד בשבילו.

2.3 דיון בנושא העיצוב הנבחר

כאן יש לפרט ובעיקר להסביר מדוע בחרתם דווקא בחלוקת הרכיבים/תפקידים הזו ולא אחרת. מה היתרונות שלכם מבחינתה, וגם מה החסרונות שאתם מודעים אליהם. במידה וחשבתם על חלופות אחרות, יש לציין אותן ולהסביר בקצרה כיצד הן שונות מהפתרון שנבחר - ומדוע החלטתם בסופו של דבר שלא לבחור בחלופות האלו.  
כאן יש להתייחס גם לגבי שפת התכנות שבה בחרתם לכתוב כל רכיב, ומדוע דווקא בשפה זו ולא אחרת.



החלק של עיבוד הנתונים ולמידת המכונה יתוכנת בפייתון מכיוון שהוא נוח לתכנות בשפה זו וישנן הרבה ספריות נוחות לעבודה אשר מאפשרות למידת מכונה.   
האתר יהיה מורכב בעזרת קוד - python javascript html css (php?)   
בגלל הנוחות והיעילות של כל שפה.

1. עיצוב נתונים ופרוטוקולים

כאן יופיע תיעוד של מבני נתונים שונים / פרוטוקולים המשמשים אותנו במערכת

בסעיף זה יש לפרט את כל סוגי המידע אשר מועבר או נשמר במערכת - בין אם בין רכיבים, בין מודולים או בכל צורה אחרת (כלומר גם אם זה משמש רק לתקשורת פנימית ולא ע"ג הרשת, או למשל אם זה משהו ששומרים לקובץ/בסיס נתונים).

* לכל סוג מידע כזה יש לפרט על השדות אותן הוא מכיל ומאיזה סוג כל שדה, מה טווח הערכים הרלוונטי לגביו וכל הגבלה או מידע נוסף אחר שאתם מוצאים לנכון (למשל: האם מותר שהשדה יהיה ריק ומה המשמעות של זה, אורך או ערך מינימלי/מקסימלי למספר או למחרוזת)
* יש לציין לאיזו מטרה משמש כל מבנה נתונים.
* עבור פרוטוקול תקשורת-נתונים בין רכיבים שונים:
* מה המצבים השונים לכל סוג של תקשורת (למשל אותנטיקציה, הרשמה, שליחת קובץ וכו')
* באילו ייצוגי מידע נעשה שימוש + תרשים זרימה של המצבים השונים שיכולים להיות וסדר השלבים שלהם (למשל כיצד נעביר מידע על מקרים של סיסמה שגויה, משתמש חסום, התחברות מוצלחת וכו')
* מומלץ להגדיר מראש קודים של בקשות/פעולות, תוצאות/תגובות ושגיאות אשר ישותפו בין כך הרכיבים הרלוונטיים
* יש להתייחס לאופי המידע הנשמר (האם הוא טקסטואלי או בינארי) ולייצגו בצורה נוחה והולמת בהתאם
* יש לפרט אודות מבנה בסיס הנתונים (חלוקה לטבלאות, השם והטיפוס של כל עמודה/שדה, קשרים בין טבלאות ואילוצים כלשהם על עמודות)

בין חלקי הפרוייקט ישנה העברת נתונים רבה.  
המשתמש צריך להירשם לבחור קטגוריות שמעניינות אותו, ולהתחבר.   
נתוני המשתמש ישמרו בבסיס הנתונים.  
תעבורה של משתמשים כאשר יש בקשות לשרת.  
תעבורה בין השרת לבסיס הנתונים כאשר צריך לעדכן נתונים (קובץ, טקסט,תמונה וכו')

המידע המועבר יהיה טקסטואלי ובינארי.

עמודות – שם, שם משתמש, סיסמה,טלפון, 5 קטגוריות מועדפות, מייל, עוקבים אחרייך, עוקב אחרי, חברים ועוד

4. ממשק משתמש

כאן יופיע פירוט ממשקי המשתמשים וכן תופיע הסקיצה עבורם

בסעיף זה יש לפרט את הפונקציונליות של המערכת כפי שהיא מתבטאת עבור משתמש חיצוני. במידה ויש סוגים שונים של משתמשים, יש להתייחס לכולם בהתאם - אילו רכיבים או נתונים רלוונטיים עבורם, וכיצד הם מתקשרים איתם.  
  
בנוסף, יש לצרף המחשות ויזואליות של המסכים השונים, ולהסביר את התוכן שלהם (למשל מה התפקיד של כל שדה או כפתור, מה קשור/תלוי במה - למשל כפתור שמכובה בהתאם לתנאים מסוימים במערכת וכן הלאה) והקשרים ביניהם (איזה מסך מוביל לאיזה מסך ובאילו מקרים)  
  
דוגמה (לא מלאה):  
תמונה שמכילה לוח ציור

התיאור נוצר באופן אוטומטי

5. נספחים

**כל דבר שהייתם רוצים להוסיף בנוגע לעיצוב המערכת**

**giuthub**

<https://github.com/itaykbn/cyber-senior-project>