

הנדסת תוכנה לאינטרנט – פרויקט מסכם תשפ"ב

כללי:

מוסד הבנקאות כגורם מתווך פיננסי קיים מאות שנים במתכונתו הנוכחית, ובין תפקידיו המרכזיים קבלת פקדונות במסגרת חשבונות עו"ש, חיסכון והשקעה מחד, ומאידך מתן הלוואות. בנק משמש גורם חשוב בכלכלה מודרנית על ידי ניתוב והקצאת מקורות לגורמים הזקוקים לו לצורך פיתוח וצמיחה.

בשנים האחרונות מקבלים מוסדות מסוג זה ציביון וירטואלי (כמעט) מלא, והצורך בסניפים/פקידים/כסף ממשי הופך למיותר, ואיתו ההכרה כי גורם מתווך אינו נחוץ בעידן הנוכחי ואף הצורך בכסף כאמצעי חליפין שמקורו בבנקים מרכזים אינו ברור מאליה ומאותגר על ידי מגוון מטבעות קריפטו (crypto currency).

יחד עם זאת הצורך בקביעת מנגנון ביסוס ערך וואלידציה טרנסאקציות נותר בעינו, והנטיה כיום היא ליצור מנגנון מבוצר העוקב ומאשר נכונות תנועות בצורה מאובטחת ומכונה Blockchain.

בפרויקט זה ניצור בנק וירטואלי חברתי המספק במה להתקשרות בין אנשים המנהלים בו חשבון (ארנק) ומאפשר ביצוע העברת כספים ביניהם. ולידציית הטרנסאקציות תעשה באמצעות שירות ייחודי מבוסס Blockchain.
* המערכת המוגדרת מפשטת מרכיבים במנגנון ניהול שהיה ממומש במציאות.

דרישות פונקציונאליות:

1. למערכת 2 שחקנים : לקוח ומנהל.
2. לקוח יכול לבקש לפתוח חשבון.
3. מנהל יכול לפתוח חשבון ללקוח לרבות הגדרת סכום כסף ראשוני העומד לרשותו ומתורגם מדולר אמריקאי למטבע המכונה LevCoin. אפשר לבקש להגדיר סכום ראשוני גם באמצעות שקלים.
4. שווי LevCoin נגזר ממספר המטבעות שחולקו ללקוחות. ללקוח ראשון שווי LevCoin הינו דולר, ללקוח שני 0.99, לשלישי 0.98 בסדרה יורדת (שימו לב כי יש לברר את שער הדולר בעת התרגום).
5. לקוח יכול :
 - א. לעדכן פרטיו (למעט יתרה).
 - ב. להעביר סכום כסף ללקוח אחר.
 - ג. לצפות בתנועות החשבון בחתך שבועי/חודשי/שנתי.
 - ד. ללוות או להלוות כסף מ- או ל- לקוח אחר לתקופה מוגדרת.
 - ה. לצפות ברשימת ההלוואות.
 - ו. ניתן להציג סטטוס בשקלים או LevCoin.
6. לא ניתן להלוות כסף יותר מחצי הסכום הקיים ביתרה.
7. לא ניתן ללוות כסף יותר מ-60% מהמופיע ביתרה.
8. אם יתרת לווה ירדה מתחת לסף 60% כמוגדר בסעיף 5. תשלח הודעה מתפרצת למלווה, הרשאי לבקש החזר מיד.
9. אם יתרת לקוח מגיעה ל-0 תשלח הודעה מתפרצת למנהל.
10. לקוח יכול לנהל Chat עם מנהל הבנק.
11. מנגנון ולידאציות פעולות יעודק עם כל טרנסאקציה וניתן יהיה לתשאלו אודות מצב העולם מבחינת יתרות.

דרישות לא פונקציונאליות:

1. המערכת תתבסס על MERN Stack (להבהיר MongoDB)
2. כל User Task ימודל באמצעות MVC.
3. ניהול משתמשים ייעשה באמצעות חבילת npm סטנדרטית או באמצעות שירות Authentication של Firebase.
4. כדי לפתוח חשבון יש לשלוח מייל למנהל, תוכן המייל ופרטיו יופיעו למנהל לא "כמייל" אלא כהודעה במסך הזנה\עדכון, מסך שבו המנהל יחליט האם לאשר פתיחה ואם כן יזין את הסכום הראשוני בחשבון (ארנק). השתמשו ב- Google Developer API לקריאה\כתיבת מיילים במסגרת המערכת.
5. הודעות מתפרצות ימומשו באמצעות פרוטוקול Web Socket.
6. כל נתוני המשתמש יישמרו לוקאלית בדפדפן באמצעות LocalStorage ורק במידה והיתה תקלה ניתן יהיה לשחזרם.
7. שירות הוואלידציות ימומש באמצעות Node.js ו- Blockchain (לדוגמא : <https://medium.com/codesphere-cloud/how-to-build-your-own-blockchain-in-nodejs-b4506294c9d4>)
8. שירות המרת דולר לשקל - כל שירות הכולל API מתקבל, או אלטרנטיבית חילוץ שער דולר באמצעות Web Scrapping.
9. אופציה: העלאת המערכת לשירות אירוח כדוגמת <https://www.heroku.com> או הכנת Docker Image והרצת השרת מתוכו.
10. אופציה: אירגון הפרויקט ותוצריו בסביבת GitHub.

הערות:

1. את הפרויקט יש להגיש בזוגות או שלשות. יש להגדיר לכל שותף מה חלקו בפרויקט.
2. נא להרשם להגשה כאן : https://docs.google.com/spreadsheets/d/1_pUdT8KJlIkjVZjEDjzLujHi5i2dkaqKefwWZdoE8/edit#gid=0
3. במסגרת ההגשה יישאלו שאלות הנוגעות לחומר שנלמד בקורס ודנו בו בהרצאות הפרונטליות.
4. על התצוגה להיות מקצועית ואסתטית ככל האפשר, נא לחפש בגוגל רעיונות לעיצובים באפליקציות דומות. לתצוגה המחוזה מקצועיות יינתן משקל בהערכת הפרויקט הכללית.
5. ההגשה תהיה בזום בתאריך 10.7.22 (נא לעקוב במידה ויהיו שינויים).

בהצלחה.