קורס מיני פרויקט במערכת חלונות

מסמך הפרויקט

1. פתיחת התוכנית :

* נפתח חלון בו ישנה אפשרות להיכנס כמנהל / משתמש.

מנהל יכול לשנות ולעדכן ולהוסיף את מה שנצרך במערכת שלנו.

למשתמש ישנה אפשרות רק לצפות בקווים הקיימים ובתחנות הקיימות.

באותו חלון ישנה גם אפשרות להתחיל את הסימולציה ע"י בחירת זמן התחלה וקצב ולחיצה על כפתור Start (הסימולציה תיסגר בלחיצה על הכפתור Stop באותו חלון).

* חלון מנהל:
* [שמות משתמשים וסיסמאות : (yona,yona),(elyassaf,elyassaf).]
* נפתח חלון כללי בו אפשר לבחור מחלקת קווים, אוטובוסים או תחנות.
* כפתור להוספת ישות.
* בלחיצה על שם המחלקה נפתחת רשימה של הנתונים הקיימים במערכת כאשר לחיצה כפולה על פריט פותח חלון של הנתונים של אותו פריט.
* בכל פריט קיימים כפתור לעדכון והסרה של אותו פריט.
* בחלון המידע של הקוים ושל אוטובוסים מוצגת מפה של מיקומי התחנות תוך שימוש ב Api של Microsoft.
* חלון משתמש:
* [שמות משתמשים (Yona, Elyassaf).]
* [סיסמאות: 1234.]
* נפתח חלון בו מוצגות רשימת הקווים והאוטובוסים הקיימים במערכת.
* לחיצה כפולה על פריט פותח חלון של הנתונים של אותו פריט.

1. מנגנון סימולטור הנסיעות:
2. בלחיצת start בשעון הסימולציה מופעל פועל רקע (כל עוד לא נלחץ stop ואז התהליכון מבצע cancel) ששולח בקשה לשכבת הלוגיקה אשר מאתחלת את שעון הסימולציה על פי הנתונים שהוזנו ומעדכנת את הנתונים לפי קצב התקדמות השעון.
3. שעון הסימולציה הינו מחלקת singleton המייצג את שעת המערכת.
4. באותה המחלקה מוגדר אירוע שאליו נרשמת פונקציה (UpdateTime) שנשלחת על ידי שכבת התצוגה עבור עדכון התצוגה של השעון.
5. עם תחילת הסימולציה שכבת הלוגיקה מפעילה פונקציה (SendLinesToTravel) שתפקידה הוא להתחיל את הסימולציה של שליחת הקווים לנסיעה. הפונ' מגדירה פועל רקע עבור כל יציאת קו של כל קו בנפרד.
6. כל פועל רקע שיש לכל יציאת קו יוציא קו לנסיעה על פי תדירות יציאת קו זו.
7. כל קו שיוצא מופעל בתהליכון נפרד בעל מזהה ייחודי key אשר מוגדר לו ע"י קובץ קונפיגורציה בשכבת הלוגיקה.
8. עבור כל קו בנסיעה, עם הגעתה לתחנה מתבצע חישוב של כל זמני ההגעה לתחנות שבהמשך המסלול, ובמידה והתחנה המושקפת (התחנה שבמעקב) הינה אחת מהן אזי נשלח עדכון של זמן ההגעה ע"י השמה של LineTiming חדש המכיל את זמן ההגעה לתחנה וכן מזהה ייחודי של נסיעת הקו הספציפית לתוך שדה LineTiming של מחלקת TravelSimulator (שהיא בעצם המושקף של כל נסיעות הקו וכן המשקיף של הפאנל הדיגיטלי של התחנה שבמעקב).
9. מחלקת TravelSimulator הינה singleton.
10. השמה זו מפעילה את האירוע של TravelSimulator אשר מפעיל פונ' שנשלחה ע"י שכבת התצוגה בהגדרת התחנה למעקב.
11. בפתיחת חלון מידע של תחנה בשכבת התצוגה, אם הסימולטור פועל ויש לתחנה פאנל דיגיטלי (ע"פ המסומן ב CheckBox, נשלחת בקשה לשכבת הלוגיקה להגדיר תחנה זו להיות התחנה שבמעקב ע"י שליחת מזהה התחנה וכן פונקציה Action<LineTiming> אשר תבדוק את נתוני LineTiming הנשלחים משכבת הלוגיקה ותעדכן את התצוגה של הפאנל הדיגיטלי.
12. בונוסים:
13. ניהול המידע על תחזוקת כלי התחבורה (האוטובוסים) המשמשים לנסיעות.
14. ממשק חלון נוסע עבור משתמש user נוסף על ממשק המנהל.
15. תבנית מתקדמת של מודל השכבות.
16. שימוש ב DataGrid.
17. שימוש ב IValueConverter אשר מציג בסימולציה הגעת אוטובוס ברקע אדום ומעפיל צפצוף רקע.
18. שימוש ב MaterialDesign.
19. מחיקת ישות משנה את שדה הפעילות שלו ל OFF. (כמובן שיישות מחוקה לא תתקבל בביצוע בקשות משכבת הנתונים).
20. הוספה, גריעה  משירות, עדכון פרטים של אוטובוסים.
21. עדכון מיקום תחנה הכולל עדכון מרחקים וזמנים מתחנות קודמות ואל תחנות עוקבות.
22. גריעה של תחנת אוטובוס תוך עדכון כל הנתונים בהתאם.
23. זמן התחלת תווך זמנים עבור ישות יציאת קו.
24. תדירות יציאות קו שיכולה להשתנות עבור זמנים שונים ביום.
25. אפשרות הפעלת כל התחנות עם לוח אלקטרוני.
26. יתבצע עדכון זמן הגעה שבמעקב לתחנה לא רק בהגעה לאחת התחנות הקודמות אלא לפחות פעם בשניה (האמיתית) תוך הערכת זמן ההגעה ע"פ התקדמות בין שתי תחנות נוכחיות + זמן ממוצע ביתר התחנות עד לתחנה שבמעקב. **נ.ב. - הבונוס הזה הוא משמעותי מבחינת חווית המשתמש**.
27. שימוש בממשק Api של Microsoft עבור המפות שמוצגות בכל תחנה ובכל קו (כולל מסלול הנסיעה של הקו), תוך מתן אפשרות לבחור מחדש את מיקום התחנה ע"י לחיצה כפולה במפה על המיקום הרצוי.
28. שימוש ב regex עבור בדיקת תקינות קלט במספר מקומות.
29. שימוש ב TimePicker עבור בחירת שעת תחילת הסימולציה.