**🧾 Remote Memo CONNECT – הגדרות פרויקט**

**🎯 מטרת האפליקציה**

אפליקציה לשליחת תזכורות קוליות וטקסטואליות בין מכשירים, המיועדת לאנשים עם בעיות זיכרון. ההודעות מתוזמנות להפעלה אוטומטית בזמן שנקבע מראש, ונשמרות במערכת מאובטחת המבוססת על סנכרון peer-to-peer ו־Blockchain לוקאלי.

**📱 פלטפורמות נתמכות**

* Android 9.0 ומעלה
* iOS 12.0 ומעלה
* Web (לצורכי הדמיה עתידית)

**👥 קהל יעד**

* אנשים עם לקויות זיכרון
* בני משפחה ומטפלים המעוניינים לשלוח תזכורות

**🧩 תכונות מרכזיות**

**יצירת הודעה**

1. הקלטת קול ו/או כתיבת טקסט
2. בחירת תאריך ושעה להפעלת התזכורת
3. שמירה מקומית (AsyncStorage), הצגה ביומן ודף הבית
4. שליחה למספר המחובר (בעתיד – לפי receiverId דרך Relay Server)

**קבלת הודעה**

* רישום ביומן
* שמירה מקומית
* הצגה בדף הבית
* הפעלה אוטומטית בזמן שנקבע  
  (בעתיד – תמיכה מלאה ב־receiverId דרך Relay Server)

**מחיקת הודעות**

* מתבצעת אוטומטית לפי פרק זמן שנקבע ב־SettingsScreen

**🧾 סטטוסי הודעות**

סטטוסים נתמכים:  
unread, pending, delivered, received, played, read, alert triggered, alert receipt confirmed, deleted\_by\_peer

* כל עדכון סטטוס מחושב מחדש עם hash
* כל סטטוס נחתם ב־Blockchain רק לאחר אימות על ידי שני הצדדים
* רענון סטטוסים מתבצע כל 5 שניות
* נורית סנכרון:  
  🔴 לא מסונכרן | 🟡 בתהליך | 🟢 מסונכרן

**🧠 ניהול הודעות**

* עריכה/מחיקה לפני ואחרי שליחה
* הצגה בדף הבית:
  + הודעות נכנסות – שמאל, תכלת בהיר
  + הודעות יוצאות – ימין, קרם בהיר
* הודעות שנמחקו ע"י peer מוצגות באפור (50% שקיפות)
* הודעות שנמחקו על ידי שני הצדדים אינן מוצגות – אך ממשיכות להופיע ב־Blockchain
* מסך יומן TRUST מציג את כל ההודעות שנשמרו במכשיר

**🌀 Application Sync Layer – שכבת סנכרון בין מכשירים**

שכבה זו אחראית לעדכון וסנכרון מידע בין שני מכשירים, דרך שני מנועים נפרדים:

**1. useSyncEngine – סנכרון סטטוסים**

* רץ כל 4 שניות
* שולח ל־peer את כל ההודעות היוצאות מהמכשיר (id, status)
* מקבל רשימת statusUpdates מה-peer ומעדכן את ההודעות המקומיות
* אם יש עדכונים → syncStatus = 'idle', אחרת → syncStatus = 'ok'
* במקרה של שגיאה או peer לא זמין → syncStatus = 'idle'
* הודעות unread נשלחות מחדש אוטומטית

**2. useLedgerSync – סנכרון Blockchain (היסטוריית הודעות)**

* רץ כל 5 שניות
* בונה Ledger מקומי (id, status, hash)
* יוצר בלוק חדש עם createBlock (כולל previousHash)
* שולח את ה־Ledger ל־peer באמצעות sendLedgerQuery
* השוואת Ledgers:
  + הודעות חסרות → requestMissingMessage
  + סטטוסים שונים → עדכון הודעות
  + הודעות שנמחקו ב־peer → סימון כ־deleted\_by\_peer (שקופות)
* כל בלוק חדש נשמר רק אם קיימת **התאמה מלאה**
* verifyBlockchainMatch מבצע אימות בין היסטוריות הבלוקים
* אם קיימת התאמה מלאה → syncStatus = 'ok' (ירוק)
* אם אין התאמה → syncStatus = 'idle' (צהוב/אדום)

**🔐 TRUST LEDGER – שכבת אימות בלוקצ'יין**

* כל הודעה נחתמת ב־hashMessage הכולל id, status, text, audioBase64
* כל סנכרון יוצר בלוק חדש עם createBlock
* מבנה בלוק:
  + timestamp
  + רשימת הודעות (ledger)
  + previousHash
  + hash כולל
* לאחר כל סנכרון Ledger, מתבצע verifyBlockchainMatch
* **הבלוק נחתם רק אם שני הצדדים מאשרים התאמה מלאה**
* במקרה של חוסר התאמה – הסנכרון נכשל (syncStatus = 'idle')נורית העדכון אדומה.
* רק בהתאמה מלאה → syncStatus = 'ok' והנורית הופכת ירוקה.

