**RPS**

**Rock, Paper & Scissors**

נעם חדד 311240774

עופר מונטאנו 209048693

דור בר שי 205661739

אוהד בן עזרא 208567727

תוכן עניינים:

[תקציר מנהלים 4](#_Toc46943266)

[צילומי מסך 5](#_Toc46943267)

[התחברות 5](#_Toc46943268)

[הרשמה 6](#_Toc46943269)

[ראשי 7](#_Toc46943270)

[הגדרות 8](#_Toc46943271)

[פרופיל 9](#_Toc46943272)

[רשימת חברים 10](#_Toc46943273)

[הוספת חבר 11](#_Toc46943274)

[המתנה למשחק 12](#_Toc46943275)

[קרדיט 13](#_Toc46943276)

[חדר משחק 14](#_Toc46943277)

[ניצחון 15](#_Toc46943278)

[תיקו 16](#_Toc46943279)

[טבלת שחקנים 17](#_Toc46943280)

[תרשים Use Case 18](#_Toc46943281)

[תיאור ה Use Cases: 19](#_Toc46943282)

[View Statistics UC description: 19](#_Toc46943283)

[Register UC description: 20](#_Toc46943284)

[Login UC description: 21](#_Toc46943285)

[Edit Profile UC description (TBA): 22](#_Toc46943286)

[Confirm Friends Request UC description (TBA): 22](#_Toc46943287)

[View Friends List UC description: 23](#_Toc46943288)

[Invite Friend To Play UC description (TBA): 23](#_Toc46943289)

[Remove Friend UC description (TBA): 24](#_Toc46943290)

[Add Friend UC description: 24](#_Toc46943291)

[Open Room UC description: 25](#_Toc46943292)

[Place Flag UC description (TBA): 25](#_Toc46943293)

[Place Trap UC description (TBA): 26](#_Toc46943294)

[Place Soldiers UC description (TBA): 27](#_Toc46943295)

[Send Message UC description (TBA): 28](#_Toc46943296)

[Exit Room UC description: 28](#_Toc46943297)

[Move Pawn UC description: 29](#_Toc46943298)

[Battle UC description: 29](#_Toc46943299)

[Choose RPS UC description: 30](#_Toc46943300)

[תרשים Class Diagram 31](#_Toc46943301)

[ארכיטקטורה פיזית 32](#_Toc46943302)

[ארכיטקטורה לוגית 33](#_Toc46943303)

[תהליכים מרכזיים 34](#_Toc46943304)

[התחברות למשחק: 34](#_Toc46943305)

[צעד בחדר משחק: 35](#_Toc46943306)

[בדיקות 36](#_Toc46943307)

[בדיקות שבוצעו ב Unity: 36](#_Toc46943308)

[בדיקות שבוצעו באמצעות סימולטור של מכשיר אנדרואיד: 37](#_Toc46943309)

[תהליך העבודה 38](#_Toc46943310)

[סיכום 40](#_Toc46943311)

תקציר מנהלים

הצוות שלנו פיתח משחק Mobile רשתי ל Android המבוסס על משחק רשת ישן שהיה קיים ב- ICQ.

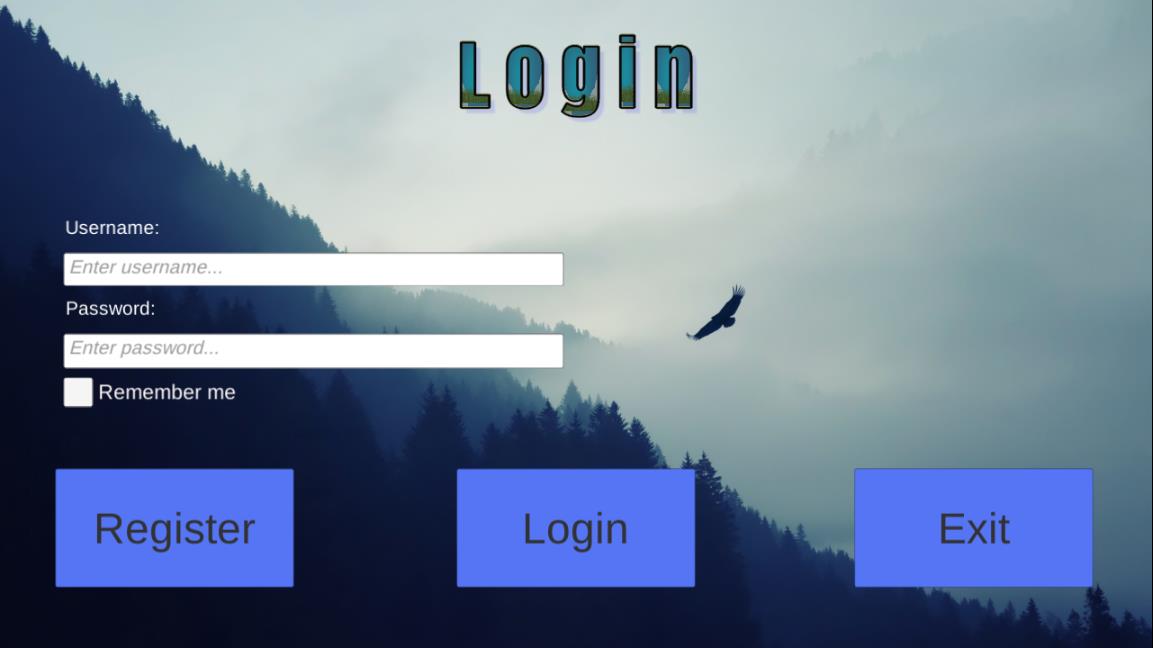
בעת כניסה למשחק המשתמש נדרש לבצע הרשמה חד פעמית, ולאחר מכן התחברות למערכת. לאחר ההתחברות המשתמש יכול להתחיל לשחק עם משתמש אחר על גבי רשת האינטרנט.

המשחק משוחק על לוח משובץ בגודל 6x7, כך שכל משתמש מתחיל עם 14 חיילים הממוקמים בתחתית הלוח. החיילים מחולקים לכמה טיפוסים: אבן (4), נייר (4), מספריים (4), מלכודת (1), ודגל (1). החיילים האקטיביים (אבן, נייר ומספריים), יכולים לנוע משבצת אחת קדימה, אחורה, ימינה או שמאלה, והם יכולים להתקיף חייל יריב כאשר הם נתקלים בו. החיילים הפסיביים (מלכודת ודגל) לא יכולים לנוע במרחב. מטרת המשחק הינה לתפוס את דגל היריב, כאשר לוגיקת היחסים בין כוחות החיילים היא בהתאם לחוקי הבסיס[[1]](#footnote-1) של משחק האבן, נייר ומספריים, ובנוסף מלכודת אשר מנצחת את שלושתם. כאשר מתקיים תיקו בין חיילים בעלי אותו כוח, שני השחקנים בוחרים כוח חדש לחיילם ומתקיים דו-קרב חוזר.

המשתמש הראשון שהצליח לתפוס את דלגו של היריב מנצח במשחקון. כאשר זה קורה שני השחקנים מקבלים הודעה על המנצח במשחק ומוחזרים לתפריט הראשי.

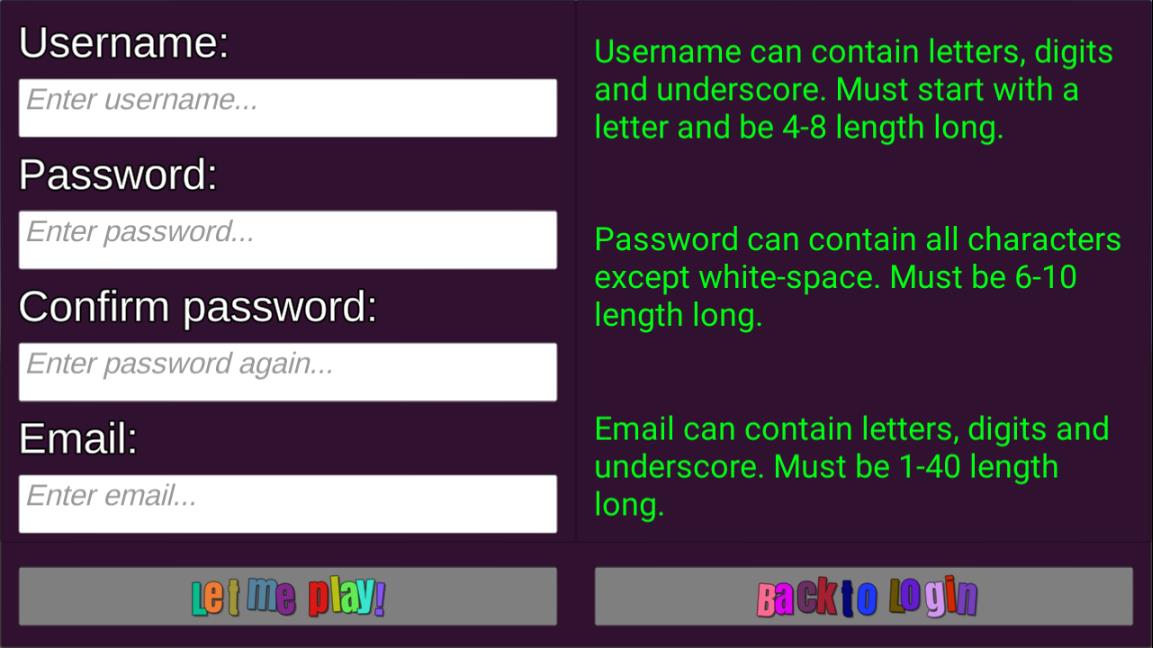
צילומי מסך

התחברות



מסך ההתחברות נועד לעשות ווריפיקציה של המשתמשים הנכנסים למשחק, רק שחקנים שנרשמו למערכת יכולים להתחבר. על המשתמשים להכניס שם משתמש וסיסמא על מנת להתחבר.

הרשמה



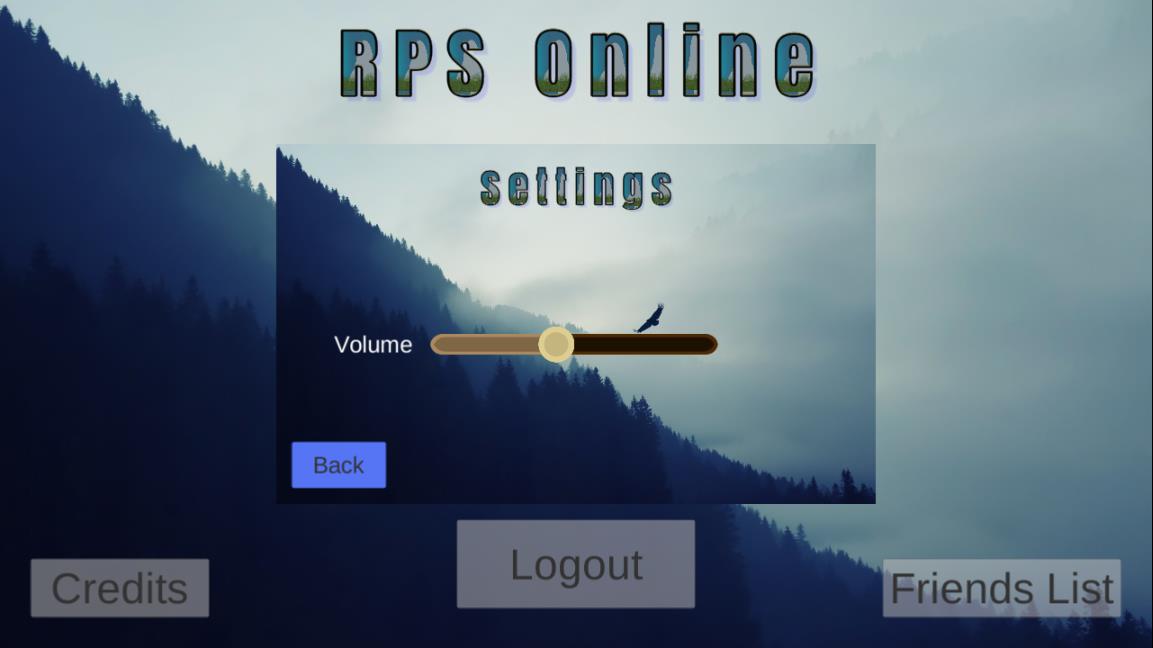
מסך ההרשמה נועד על מנת לאפשר למשתמשים להרשם למערכת. משתמשים המעוניינים להירשם נדרשים להכניס שם משתמש, סיסמא ואימייל העומדים בקריטריונים מסויימים.

ראשי



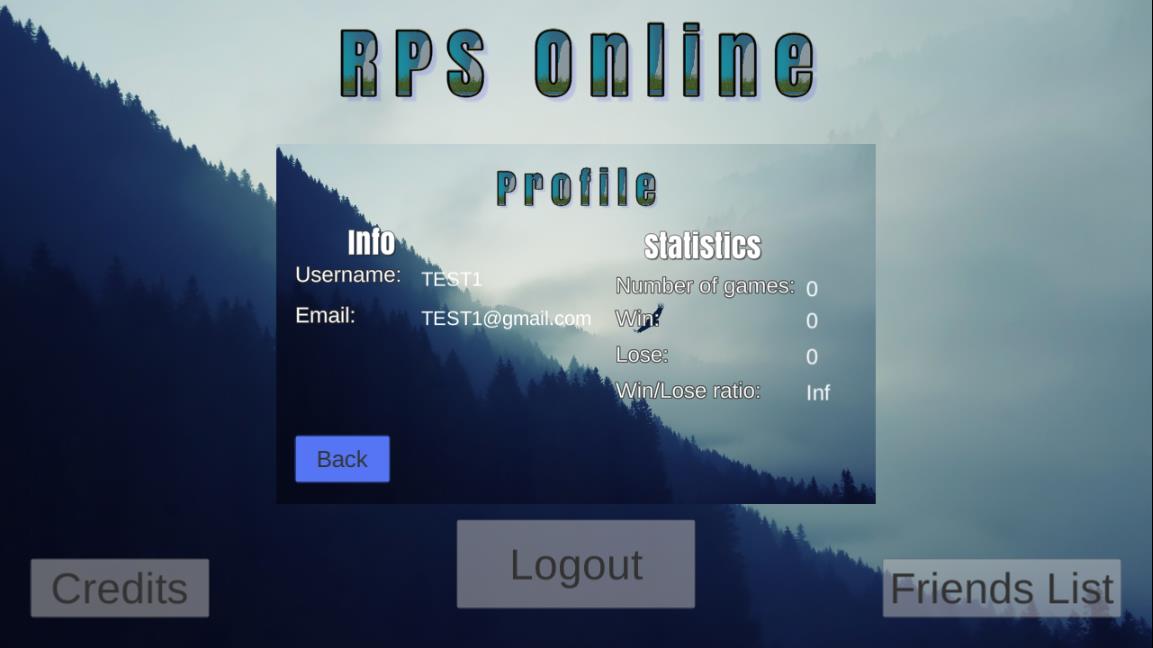
המסך הראשי נועד לניווט בין המסכים השונים.

הגדרות



מסך ההגדרות נועד לשינוי הגדרות המשחק. ניתן לשנות את עצמת מוזיקת הרקע.

פרופיל



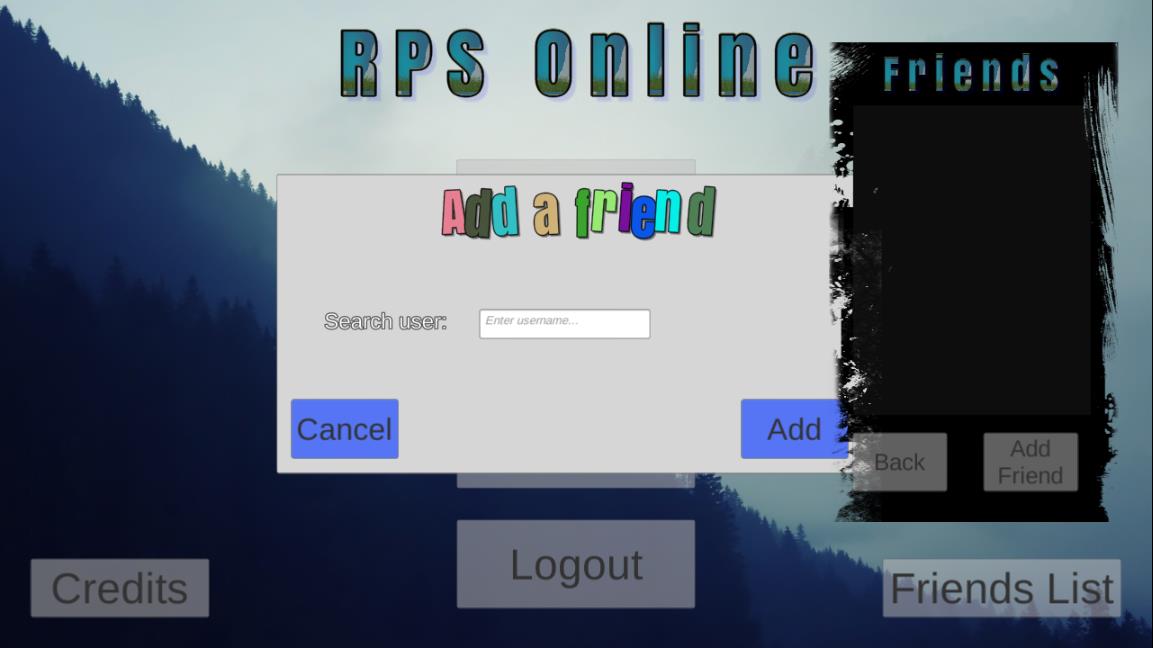
במסך הפרופיל השחקן יכול לראות נתונים של המשחקים שעשה עד כה: מספר הניצחונות, מספר ההפסדים, סך מספר המשחקים ששיחק ופרטים נוספים בחשבונו.

רשימת חברים



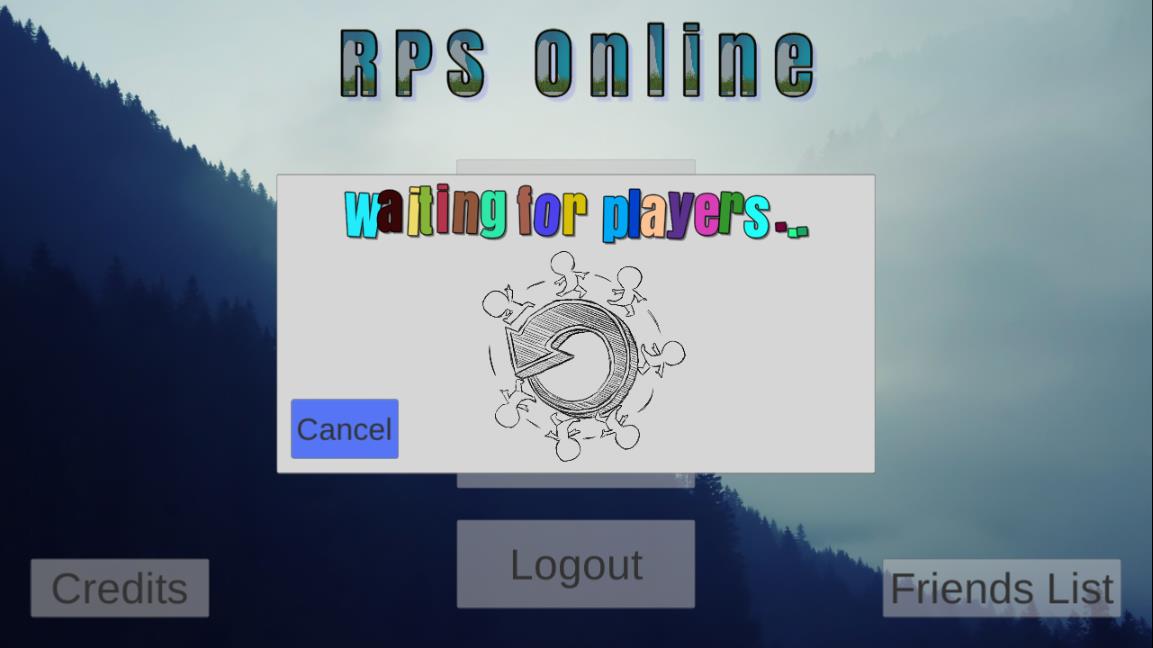
במסך רשימת החברים השחקן יכול לראות שחקנים אחרים המוגדרים כ"חברים" שלו. על מנת ששחקן אחר יופיע כחבר יש להוסיף אותו באופן ידני במסך "הוספת חבר".

הוספת חבר



במסך הוספת חבר ניתן להקליד את שם המשתמש של שחקן אחר שברצונך להוסיף לרשימת החברים שלך. במידה והוקלד שם משתמש שלא קיים במערכת תופיע הודעה האומרת זאת, במידה וקיים הוא יתווסף לרשימת החברים שלך.

המתנה למשחק



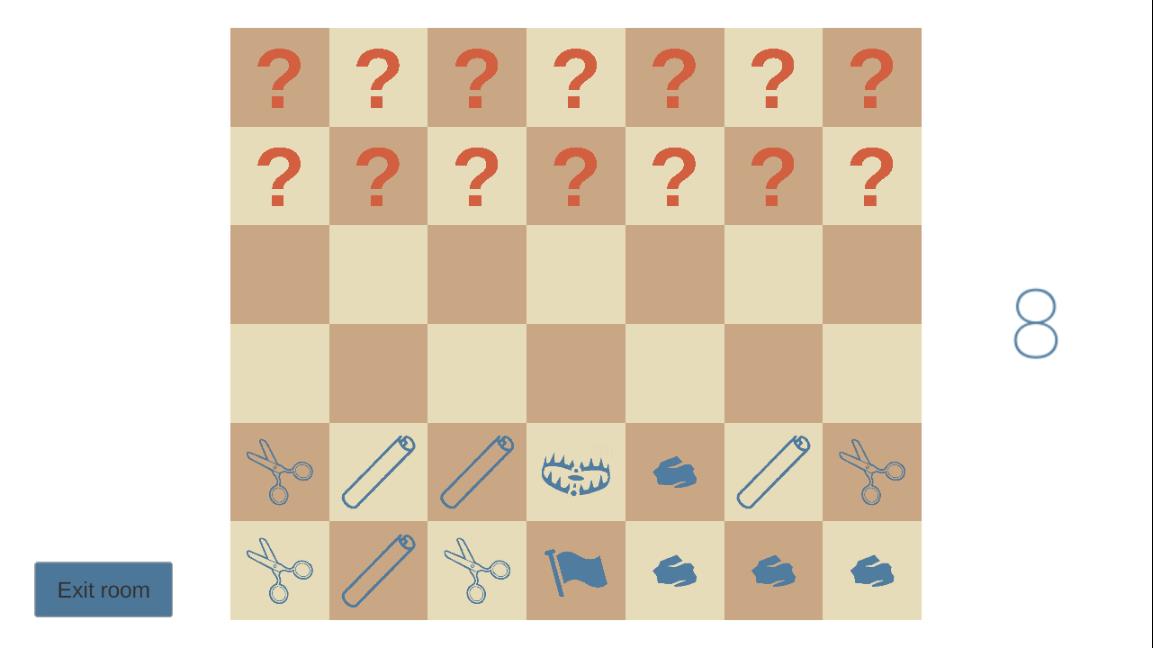
במסך המתנה למשחק השחקן מחכה ליצירת חיבור בינו לבין שחקנים האחרים שגם ממתינים למשחק, ברגע שיש 2 שחקנים שמחפשים משחק הם יכנסו לחדר פרטי וישחקו ראש בראש.

קרדיט



מסך הקרדיט נועד כדי לצפות ברשימת היוצרים של המשחק.

חדר משחק



מסך "חדר משחק" הינו מסך ששני שחקנים עוברים אליו אחרי שהמתינו במסך "המתנה למשחק". זהו החדר הפרטי שבו הם משחקים ראש בראש.

ניצחון



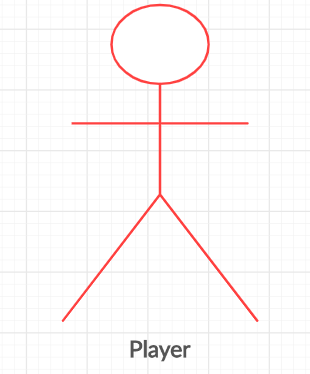
מסך הניצחון הינו מסך המופיע כאשר המשחק נגמר ב"חדר משחק" ואחד מהשחקנים ניצח. שמו מופיע במסך זה ויש אפשרות לחזור למסך הראשי.

תיקו

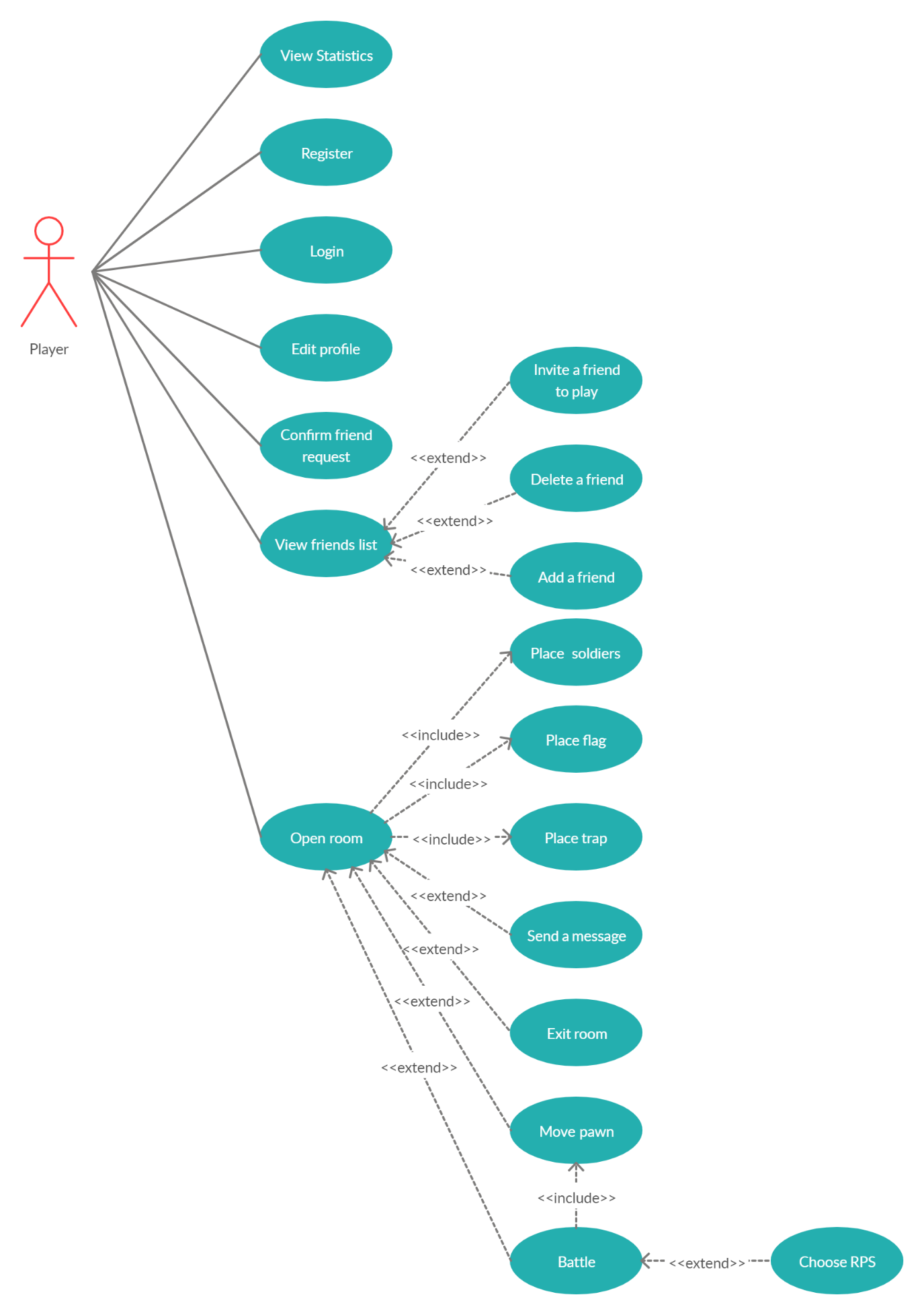


מסך התיקו נועד לבחירת סוג חייל חדש במידה ויש תיקו בין שני חיילי משחק.

טבלת שחקנים



תרשים Use Case



תיאור ה Use Cases:

### View Statistics UC description:

|  |  |
| --- | --- |
| **תרחיש שימוש (use case)** | ראיית סטטיסטיקה |
| **שחקן** | Player |
| **תיאור** | השחקן יוכל לראות סטטיסטיקות לגביו |
| **מסלול עיקרי** | בעת לחיצה על כפתור "סטטיסטיקות" בתפריט הראשי השחקן יוכל לראות את הפרטים הבאים:   * מספר המשחקים הכולל בהם שיחק. * מספר המשחקים בהם ניצח. * מספר המשחקים בהם הפסיד. * מספר המשחקים בהם פרש. * זמן התחברות למערכת כולל. * זמן התחברות למערכת ב session הזה. * זמן חדר משחק ממוצע. * מספר מהלכים ממוצע לחדר משחק. * מספר מהלכים כולל. * משבצת עדיפה למלכודת. * משבצת עדיפה לדגל. * מספר כולל של "שובר שוויון". |
| **תנאי התחלה** | על השחקן הרשום להיות מחובר למערכת |
| **תנאי סוף** | הצגת חלון עם הנתונים (מעודכנים מרגע פתיחת החלונית) |

### Register UC description:

|  |  |
| --- | --- |
| **תרחיש שימוש (use case)** | הרשמה למערכת |
| **שחקן** | Player |
| **תיאור** | הרשמות של השחקן למערכת. |
| **מסלול עיקרי** | 1. בקשת קלט מהשחקן עם הנושאים הבאים:  * דואר מייל: פורמט \_\_.\_\_@\_\_\_\_ * שם משתמש: * סיסמה: * אימות סיסמה: * פרטים אישיים:  1. עיבוד נתונים במערכת:  * אם המייל או שם המשתמש נמצא כבר במערכת, נשלחת הודעת שגיאה. * אם לא, הנתונים נשמרים במערכת  1. עדכון השחקן על רישומו למערכת בהצלחה או לא |
| **תנאי התחלה** | המערכת מקוונת |
| **תנאי סוף** | רישום נתוני השחקן למערכת בעת סיום התהליך במידה וכל הפרטים נכונים. |

### Login UC description:

|  |  |
| --- | --- |
| **תרחיש שימוש (use case)** | התחברות למערכת |
| **שחקן** | Player |
| **תיאור** | התחברות למערכת. |
| **מסלול עיקרי** | 1. השחקן נכנס למשחק. 2. המערכת תקפיץ לשחקן חלון התחברות. 3. השחקן יזין את שם המשתמש והסיסמא שלו. 4. אופציה: השחקן יוכל לבחור האם לשמור את שם המשתמש לפעמים הבאות. 5. אופציה: השחקן יוכל לבחור האם לשמור את הסיסמא לפעמים הבאות. 6. השחקן ילחץ על כפתור "התחבר". 7. המערכת תאמת את הנתונים שהשחקן יכניס. 8. במידה והנתונים נכונים- השחקן יועבר לתפריט הראשי. 9. במידה והנתונים אינם נכונים- המערכת תקפיץ לשחקן הודעה וחזרה לסעיף 3. |
| **תנאי התחלה** | על השחקן להיות רשום במערכת. |
| **תנאי סוף** | השחקן יוכל לגשת לכל התרחישים האחרים. |

### Edit Profile UC description (TBA):

|  |  |
| --- | --- |
| **תרחיש שימוש (use case)** | עריכת פרופיל |
| **שחקן** | Player |
| **תיאור** | עדכון פרטים אישיים של השחקן. |
| **מסלול עיקרי** | 1. עיבוד הנתונים של הפרטים האישיים של השחקן במערכת והצגתן בחלון עם תאי קלט שבתוכן פרטי הנתונים, כפתור לביצוע עדכון הפרטים וכפתור לסגירת החלון. 2. שחקן יכול לשנות כרצונו את הפרטים (בהתאם לפורמט של כל נושא) ובעת סיום השינוי עליו ללחוץ על כפתור ביצוע עדכון הפרטים 3. המערכת שומרת את הנתונים (בהתאם לפורמט) ונותנת התראה בעת אי עמידה בפורמט הנושא. 4. השחקן לוחץ כפתור סגירת החלון כאשר ברצונו לסגור את חלון עריכת הפרופיל 5. מערכת נותנת התראה במידה וקיים שינוי באחד והפרטים ושואלת האם הוא מעוניין לצאת מבלי לעדכן. 6. במידה והשחקן לוחץ ביטול, המערכת נשארת בחלון זה. במידה והשחקן לוחץ אישור, המערכת סוגרת חלון. |
| **תנאי התחלה** | השחקן מחובר למערכת והמערכת מקוונת. |
| **תנאי סוף** | נשמרו הנתונים בהצלחה ונסגר החלון. |

### Confirm Friends Request UC description (TBA):

|  |  |
| --- | --- |
| **תרחיש שימוש (use case)** | אישור/דחיית בקשת חברות |
| **שחקן** | Player |
| **תיאור** | אם שחקן אחר מבקש חברות, מקבלים התראה ולוחצים על כפתור אישור/דחייה. |
| **מסלול עיקרי** | 1. המערכת נותנת התראה לשחקן שקיבל בקשת חברות. 2. השחקן מאשר/דוחה את הבקשה. 3. התשובה נשלחת למערכת ולשחקן שביקש. |
| **תנאי התחלה** | השחקן רשום במערכת וקיבל הודעה לבקשת חברות. |
| **תנאי סוף** | אם אישר, השחקן השני נוסף לרשימת החברים. אם לא, הבקשה נמחקת. |

### View Friends List UC description:

|  |  |
| --- | --- |
| **תרחיש שימוש (use case)** | צפייה ברשימת חברים |
| **שחקן** | Player |
| **תיאור** | צפייה ברשימה המכילה את מי שאישר בקשת חברות מהשחקן או חברים שביקשו מהשחקן חברות והשחקן אישר זאת. |
| **מסלול עיקרי** | 1. השחקן לוחץ על צפייה ברשימה. 2. המערכת טוענת מבסיס הנתונים את רשימת החברים ומציגה אותה. |
| **תנאי התחלה** | השחקן מחובר למערכת. |
| **תנאי סוף** | סגירת חלונית הרשימה. |

### Invite Friend To Play UC description (TBA):

|  |  |
| --- | --- |
| **תרחיש שימוש (use case)** | הזמנת חבר למשחק |
| **שחקן** | Player |
| **תיאור** | שליחת הזמנה לחבר כדי לשחקן ביחד |
| **מסלול עיקרי** | 1. השחקן יפתח את רשימת החברים שלו מהתפריט הראשי. 2. השחקן יבחר חבר. 3. השחקן ישלח הזמנה לחבר לשחק איתו. |
| **תנאי התחלה** | 1. שני השחקנים צריכים להיות רשומים במערכת ומחוברים למערכת. 2. כל שחקן צריך להיות ברשימת החברים של השחקן השני. |
| **תנאי סוף** | במידה והשחקן שקיבל את ההזמנה אישר- יפתח חדר משחק שיכיל את שניהם.  במידה ולא יאשר- השחקן השולח יקבל הודעת סירוב. |

### Remove Friend UC description (TBA):

|  |  |
| --- | --- |
| **תרחיש שימוש (use case)** | מחיקת חבר מרשימה |
| **שחקן** | Player |
| **תיאור** | שחקן יוכל להסיר שחקן אחר מרשימת החברות כרצונו. |
| **מסלול עיקרי** | 1. השחקן לוחץ על כפתור המחיקה אשר צמוד לשם משתמש השחקן אשר מעוניין להסירו. 2. המערכת מעדכנת בנתונים כי שני השחקנים אינם חברים. 3. המערכת מעדכנת את השחקן שהמחיקה התבצע בהצלחה. |
| **תנאי התחלה** | השחקן מחובר במערכת והמערכת מקוונת |
| **תנאי סוף** | החבר הוסר מהרשימה |

### Add Friend UC description:

|  |  |
| --- | --- |
| **תרחיש שימוש (use case)** | הוספת חבר לרשימה |
| **שחקן** | Player |
| **תיאור** | השחקן יוכל להוסיף חבר (שחקן אחר) לרשימת החברים. |
| **מסלול עיקרי** | על השחקן להיכנס לרשימת החברים שלו, ללחוץ על כפתור "הוספת חבר חדש", ולרשום את פרטי השחקן אותו הוא מעוניין לצרף. בקשת חברות תישלח לשחקן הנ"ל, והוא יוסף לרשימת החברים רק אם יאשר אותה. עד שהבקשה אושרה, החבר החדש ישאר בסטטוס "ממתין לאישור". |
| **תנאי התחלה** | השחקן מחובר למערכת ונמצא בתוך רשימת החברים. |
| **תנאי סוף** | אפשרות א'- בקשת החברות טרם אושרה: יתווסף חבר חדש לרשימת החברים בסטטוס "ממתין לאישור".  אפשרות ב'- בקשת החברות אושרה: החבר החדש יוסף מהסטטוס "ממתין לאישור".  אפשרות ג'- בקשת החברות נדחתה: החבר החדש יוסר מרשימת החברים. |

### Open Room UC description:

|  |  |
| --- | --- |
| **תרחיש שימוש (use case)** | פתיחת חדר (משחק) |
| **שחקן** | Player |
| **תיאור** | יצירת חדר ליישום ביצוע משחק בין שני שחקנים. |
| **מסלול עיקרי** | 1. במידה והמערכת מזהה אישור בין שני חברים, המערכת פותחת חלון של המשחק 2. באותה צורה, שחקן יכול ללחוץ על כפתור "התחלת משחק" וכך יוכל לשחק עם שחקן רנדומלי שגם לחת על הכפתור. 3. המערכת מזהה שני שחקנים שלחצו על הכפתור ובדומה למקרה הראשון פותחת חלון של המשחק. 4. מיישמת התרחשות של כל פעולה במשחק אשר מתחילה מהשמת דגל. 5. החדר ממשיך לתפקד עד ליציאה של שחקן מהחדר. 6. בסוף המשחק, המערכת שומרת את נתוני המשחק ומציגה חלונית ליציאה מהחדר. 7. כאשר שני השחקנים יוצאים מהחדר, החדר נסגר. |
| **תנאי התחלה** | שני שחקנים מחוברים למערכת והמערכת מקוונת. |
| **תנאי סוף** | התרחשה יציאה של שני השחקנים מהחדר. |

### Place Flag UC description (TBA):

|  |  |
| --- | --- |
| **תרחיש שימוש (use case)** | השמת דגל |
| **שחקן** | Player |
| **תיאור** | השחקן בוחר בתחילת המשחק מי החייל שמחזיק את הדגל. |
| **מסלול עיקרי** | 1. המשחק מבקש מהשחקן לבחור חייל עם דגל. 2. השחקן בוחר חייל עם דגל, וזה נשמר לאורך המשחק. |
| **תנאי התחלה** | השחקן מחובר למערכת ונמצא בתחילת משחק. |
| **תנאי סוף** | ניתן לבחור מיקום למלכודת. |

### Place Trap UC description (TBA):

|  |  |
| --- | --- |
| **תרחיש שימוש (use case)** | השמת מלכודת |
| **שחקן** | Player |
| **תיאור** | בחירת מיקום המלכודת בלוח. |
| **מסלול עיקרי** | 1. השחקן בוחר משבצת על הלוח (בגדר משבצות שנבחרו מראש ע"י המערכת) 2. השחקן ילחץ על כפתור "אישור" על מנת לאשר שזהו אכן המקום שבחר. 3. המערכת תציב שם מלכודת. |
| **תנאי התחלה** | על השחקן להיות בתוך חדר משחק, או ע"י יצירת חדר או הצטרפות לחדר שנוצר ע"י אחד מחבריו. |
| **תנאי סוף** | המלכודת תתווסף ללוח. |

### Place Soldiers UC description (TBA):

|  |  |
| --- | --- |
| **תרחיש שימוש (use case)** | השמת חיילים |
| **שחקן** | Player |
| **תיאור** | הגדרה של כל סוג חייל בלוח מבין אבן נייר ומספריים (מלבד חיילי דגל ומלכודת), מספר החיילים מכל סוג מתחלק באופן שווה. |
| **מסלול עיקרי** | 1. השחקן בוחר חייל על הלוח שלא נחשב מלכודת או דגל. 2. המערכת מקבלת את מיקום החייל בלוח ושולח הודעה באיזה סוג השחקן מעוניין שהוא יהיה, בנוסף לכך מציגה בחלון את כפתורי הסוגים ואת כמות מגבלת החיילים מאותו הסוג. 3. השחקן בוחר בסוג. 4. במידה וכמות המגבלה הינה 0, המערכת מודיעה על שגיאה. במידה ולא המערכת מעדכנת את סוג החייל הממוקם במיקום שקיבלה בהתחלה. 5. תרחיש הבחירה מוגבל בזמן של (נחליט בהמשך) 6. בחירה זו מסתיימת כאשר מגבלות כל הסוגים מגיעות ל0, נגמר הזמן או השחקן ביקש לעשות השמה אקראית מהמערכת. 7. במידה ונגמר הזמן, המערכת עושה השמה אקראית. 8. כאשר שני השחקנים סיימו השמה, המערכת בוחרת אקראית שחקן שיתחיל את המשחק. |
| **תנאי התחלה** | מערכת מקוונת, שחקנים מחוברים למערכת ונמצאים בחדר. |
| **תנאי סוף** | כל סוגי השחקנים נבחרו ומוצגים בחלון המשחק, ומתחיל המשחק. |

### Send Message UC description (TBA):

|  |  |
| --- | --- |
| **תרחיש שימוש (use case)** | שליחת הודעה |
| **שחקן** | Player |
| **תיאור** | השחקן יוכל לשלוח הודעה לשחקן היריב איתו הוא משחק בזה הרגע. |
| **מסלול עיקרי** | במהלך המשחק, יופיע כפתור צאט שעל-ידי לחיצה עליו המשתמש יוכל להקליד הודעה. בסיום כתיבת ההודעה, ההודעה תופיע בחלון המשחק ליד דמות השחקן שכתב אותה. |
| **תנאי התחלה** | השחקן נמצא במהלך משחק. |
| **תנאי סוף** | ההודעה אשר נכתבה מופיע בחלון המשחק ליד דמות השחקן שכתב אותה. |

### Exit Room UC description:

|  |  |
| --- | --- |
| **תרחיש שימוש (use case)** | יציאה מחדר משחק |
| **שחקן** | Player |
| **תיאור** | השחקן יכול לצאת מחדר המשחק בכל רגע נתון. |
| **מסלול עיקרי** | 1. על השחקן ללחוץ על כפתור "אני נכנע". 2. המערכת תעדכן את פרטי השחקן. 3. המערכת תציג אותם לשחקן. 4. המערכת תחזיר את השחקן לתפריט הראשי. |
| **תנאי התחלה** | על השחקן להיות בתוך חדר משחק, או ע"י יצירת חדר או הצטרפות לחדר שנוצר ע"י אחד מחבריו. |
| **תנאי סוף** | השחקן יועבר לתפריט הראשי ויוצגו לו סטטיסטיקות של המשחק שיצא ממנו הרגע. |

### Move Pawn UC description:

|  |  |
| --- | --- |
| **תרחיש שימוש (use case)** | הזזת חייל |
| **שחקן** | Player |
| **תיאור** | השחקן יוכל לבצע צעד משחק עם אחד מחייליו. צעד מוגדר כהתקדמות משבצת אחת לפנים, לאחור, לצד ימין או לצד שמאל. שחקן יוכל לבצע צעד רק למשבצת שאין בה חייל אחר מחייליו. |
| **מסלול עיקרי** | 1. יגיע תורו של השחקן. 2. השחקן יבחר על-ידי לחיצה את אחד מחייליו (יוכל לשנות את הבחירה על ידי לחיצה על חייל אחר) 3. השחקן יבחר משבצת אליה ירצה להזיז את השחקן (קדימה, אחורה, ימינה, או שמאלה) 4. החייל יזוז למשבצת אותה בחר השחקן. אם במשבצת שנבחרה יש חייל של היריב, יתבצע קרב RPS. |
| **תנאי התחלה** | התרחשו השמת הדגל והשמת המלכודת, השחקנים בחדר וכרגע תורו של אותו השחקן. |
| **תנאי סוף** | במקרה של משבצת ריקה, החייל זז והתור הועבר ליריב. במקרה של הזזה למשבצת עליה חייל יריב, מתרחש ביצוע דו-קרב, במקרה של ניצחון מוזז החייל למשבצתו של החייל היריב והתור מועבר ליריב. |

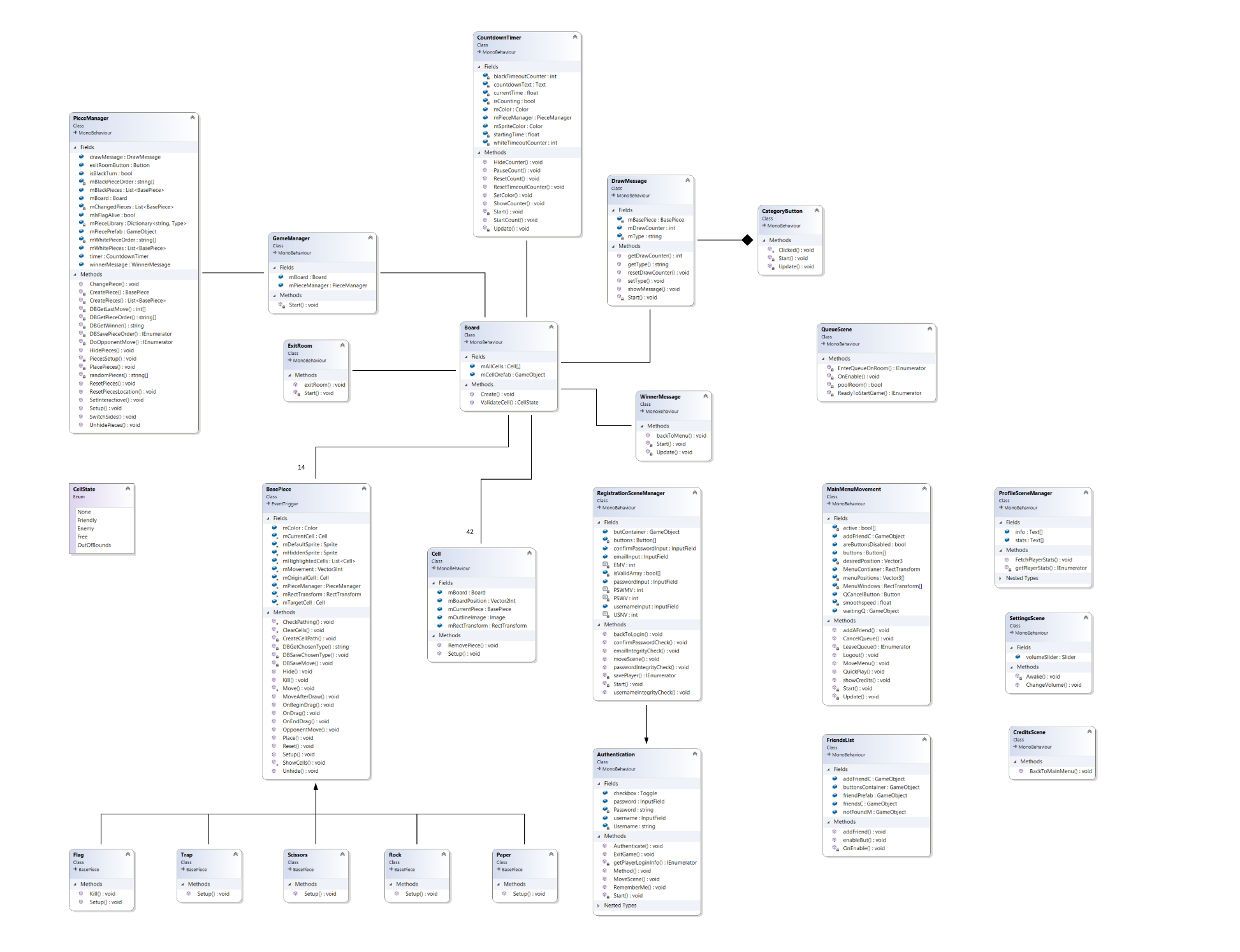
### Battle UC description:

|  |  |
| --- | --- |
| **תרחיש שימוש (use case)** | ביצוע דו-קרב |
| **שחקן** | Player |
| **תיאור** | התרחשות דו-קרב בין חייל לחייל יריב כאשר הם באותה המשבצת. |
| **מסלול עיקרי** | 1. החייל נכנס למשבצת בה עומד חייל יריב. 2. החייל תוקף את היריב במה שהוא מחזיק. 3. מתקיים ניצחון/הפסד/תיקו בהתאם לכללי RPS. |
| **תנאי התחלה** | חייל הוזז ע"י שחקן למשבצת של חייל יריב או חייל יריב הוזז למשבצת של החייל שלך. |
| **תנאי סוף** | בעת ניצחון, הזזת החייל למשבצת של החייל היריב, בעת הפסד, החייל יוסר מהמשחק. במקרה של שוויון, יתרחש בחירת RPS. |

### Choose RPS UC description:

|  |  |
| --- | --- |
| **תרחיש שימוש (use case)** | בחירת RPS |
| **שחקן** | Player |
| **תיאור** | המשתמש יכול לבחור אלמנט חדש (אבן, נייר או מספריים) לאחד מחייליו. |
| **מסלול עיקרי** | בעת שיוויון במהלך דו קרב, המשתמש נדרש לשנות את האלמנט של החייל לביצוע דו-קרב חוזר. לשחקן יפתח מסך שבו הוא צריך לבצע בחירה בין אבן, נייר, ומספריים. לאחר מכן יתקיים סיבוב דו קרב נוסף. בסיום בדו קרב, אם החייל ניצח, הוא ישאר עם האלמנט החדש שנבחר לו באופן חשוף למשתמש היריב. |
| **תנאי התחלה** | יתרחש ביצוע דו-קרב והוכרזה תוצאה של שוויון. |
| **תנאי סוף** | החייל ישאר חשוף עם האלמנט החדש שנבחר לו. |

תרשים Class Diagram

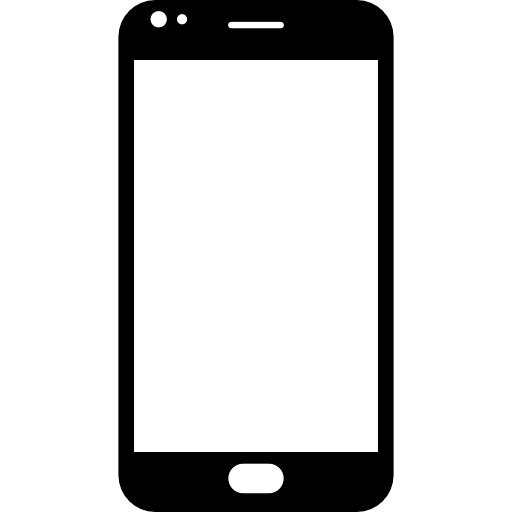
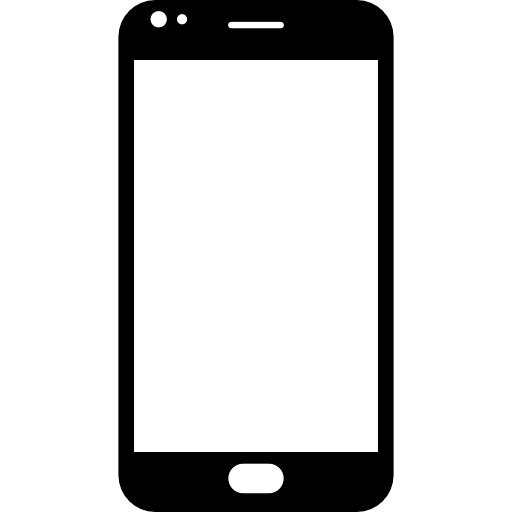


ארכיטקטורה פיזית

המשחק ירוץ על מכשירי טלפון ניידים עם מערכת הפעלה של android, התקנה של המשחק תהיה באמצעות קובץ apk.

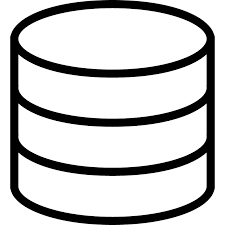
את קובץ ה apk יהיה ניתן להשיג בחנות google play.

בסיס הנתונים יאוחסן ב MS SQL Server והקישוריות תתבצע בעזרת הענן של Azure DevOps.



Azure

DevOps



Android

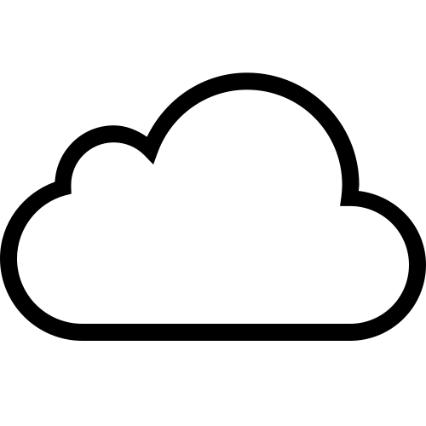
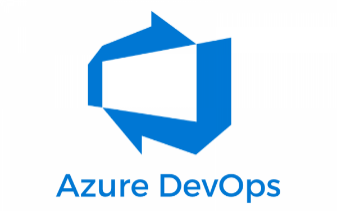
device

Android

device

Sql Server

Database



ארכיטקטורה לוגית

המערכת פותחה בסביבת העבודה של Unity engine לעיצוב המשחק והרצתה על מערכת הפעלה של android

ובסביבת Microsoft Visual studio תוך שימוש בטכנולוגיית Microsoft .net framework למימוש אלגוריתם המשחק.

**צד השרת** ייכתב בטכנולוגיית .NET בשפת C# תוך שימוש ב Web-API השמור בענן והנתונים יישמרו בשרת SQL תוך שימוש במודל entity framework של בסיס הנתונים.

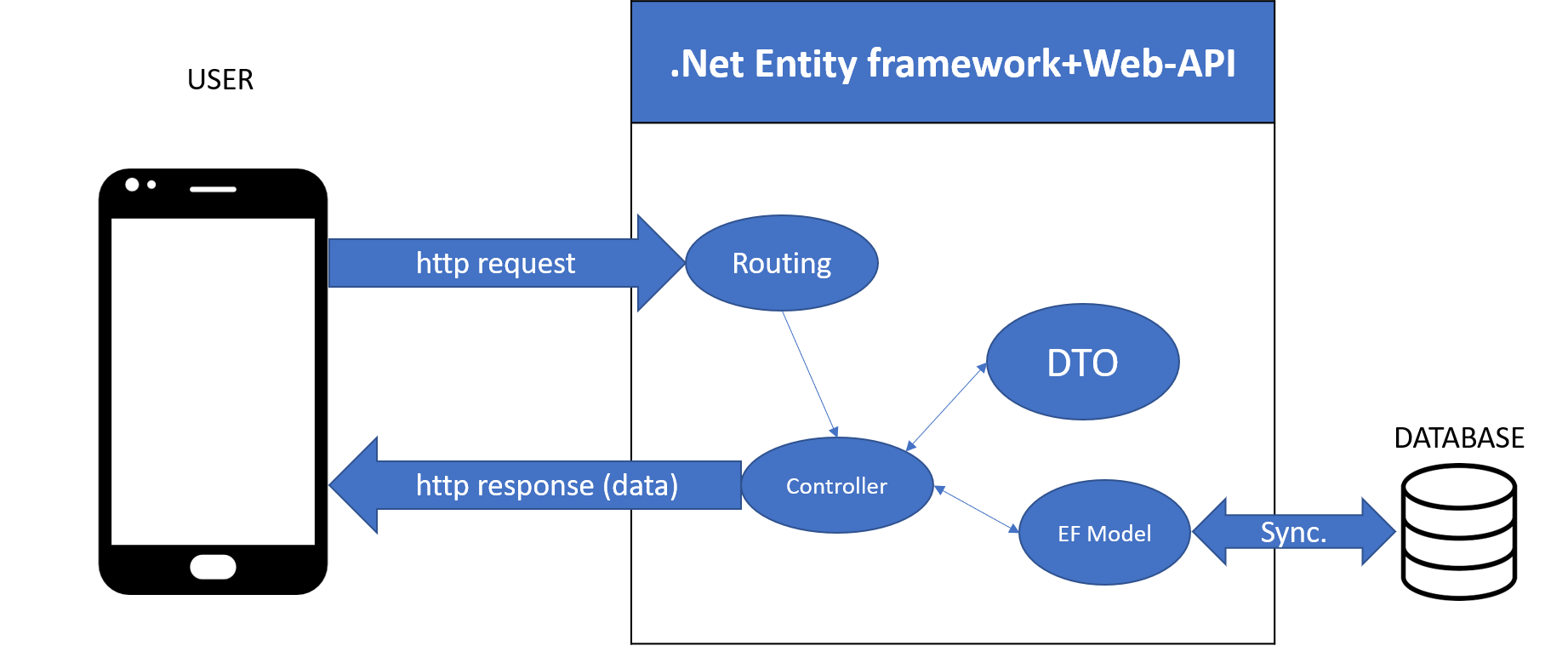
בעקבות כך, ארכיטקטורה הלוגית של המערכת בצד שרת מורכבת מהחלקים הנ"ל:

Model – אשר מכיל בתוכו את כל המידע של מסד הנתונים.

Controller – אשר מכיל בתוכו טיפול בשאילות ובקשות למול מסד הנתונים.

DTO – המידע המוחזר ללקוח.

**תבנית עיצוב:**

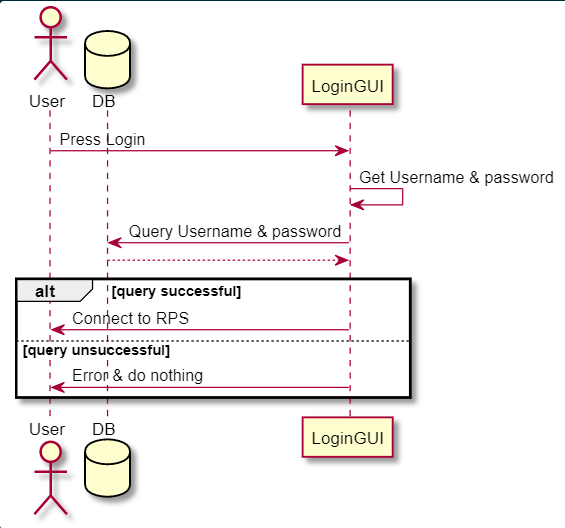


**צד הלקוח** ייכתב בשפת C# תוך שימוש בספריות ייחודיות לUnityEngine. פקודות http request וקבלת הנתונים מhttp response מתבצעות בעזרת אובייקטים הכלולים בספרייה זו אשר מסייעות בשליחה וקבלה של נתונים במנוע של unity.

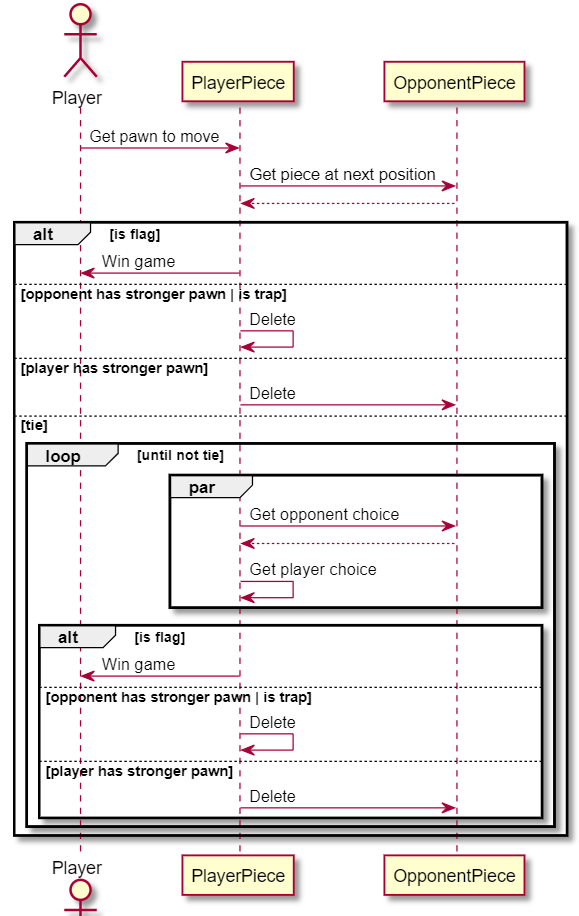
הכלים שבחרנו אפשרו לנו לפתח את המשחק למערכת הפעלה של android, השימוש ב Unity גרם לנו לעצב את המשחק בצורה הנוחה ביותר ומותאמת היטב לכל מכשיר android.

תהליכים מרכזיים

התחברות למשחק:



צעד בחדר משחק:



בדיקות

בדיקות שבוצעו ב Unity:

* + - 1. בדיקות תצוגה:
* התאמה למכשירים בעלי ממדי מסך שונים. (רזולוציה)
* בדיקה שהמשחק רץ רק בצורה landscape.
  + - 1. בדיקות לוגיות:
* הרצנו את המשחק ובדקנו שלוגיקת הקרבות בין החיילים פועלת כנדרש.
* שהמשחק נגמר כאשר אחד הדגלים נכבש.
* שהמשתמש יכול לבצע צעד עם חייל רק למקום הגיוני בלוח.
* שהתור עובר בסיום הטיימר.
* שהשחקנים נכנסים לחדרים לפי סדר כניסתם לתור.
* ווידאנו שהמשחק לא יכול להגיע למבוי סתום, לדוגמא: במצב שאף שחקן לא יכול להזיז חייל או ששני השחקנים מעבירים את תורם ללא תזוזה. (ע"י המתנה שהטיימר ייגמר)
* ווידאנו שכאשר שחקן נוטש חדר, השחקן השני ינצח כעבור זמן מה.
* במידה ויש מספר פעמים תיקו המערכת יודעת להתמודד עם זה.

1. בדיקות וולידציה:

* המשתמש יכול להירשם רק עם שם משתמש פנוי.
* כאשר המשתמש מתנתק מהמשחק הוא יכול להתחבר שוב עם אותו שם המשתמש.
* אם המשתמש לוחץ על "זכור אותי" זה באמת זוכר את הסיסמא שלו להתחברות הבאה.
* לאורך המשחק יש זיהוי של אותו המשתמש בפעולות השונות שהוא מבצע.

בדיקות שבוצעו באמצעות סימולטור של מכשיר אנדרואיד:

**הבדיקות נעשו במספר סימולטורים במקביל.**

בדיקות סנכרון:

* לאחר צעד שמשתמש עושה, המשתמש השני בחדר רואה את הצעד.
* כאשר הטיימר נגמר, התור עובר למשתמש השני והחיילים שלו ניתנים להזזה. (כאשר זה לא תורך אי אפשר להזיז את החיילים)
* הסנכרון בין המכשירים נעשה בזמן הגיוני ללא השהיות.

1. בדיקות חומרה:

* שינוי הווליום של בהגדרות המשחק אכן משפיע על ווליום המכשיר.
* המשחק לא קורס.
* המשחק מגיב למגע על המסך.

תהליך העבודה

בתחילת הפרויקט עשינו סיור מוחין, חשבנו באיזה תוכנה לבנות את המשחק ובאיזו שפה, בסוף נסגרנו על Unity מאחר והוא מספק גם עיצוב בסיסי.

לאחר מכן חשבנו על הסצנות (המסכים) השונות שיהיו במשחק וחילקנו אותם בצורה הבאה:

עופר- לוח, תיקו ונצחון.

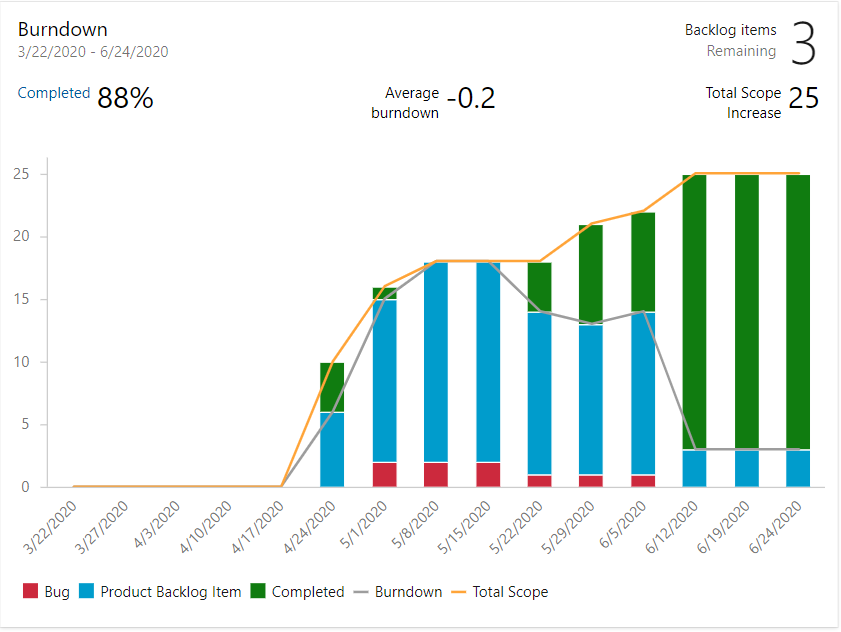
נעם- הרשמה, רשימת חברים, הוספת חבר, פרופיל ותור המתנה.

דור- התחברות, תיקו וראשי.

אוהד- הגדרות וקרדיט.

(החלוקה הנ"ל היוותה את תצורת העבודה ב Unity מבחינת העיצוב, בניית ה UX של המשחק והלוגיקה של הסצנה. מבחינת הלוגיקה והראייה הכללית של המשחק ישבנו ודיברנו ביחד על גישות ותצורות משחק שונות)

כל אחד עבד במקביל לשני, ניהלנו את הגרסאות השונות בעזרת ה Azure וכל חלוקת העבודה היתה דרכו. פעלנו לפי גישת Agile עם Sprints באורך של שבוע, חילקנו את העבודה כך שכל חבר צוות יקבל משימות מגוונות כדי שיתנסה בטכנולוגיות חדשות.



דור התחיל לעבוד על סידור הטבלאות ב Sql Server המקומי.

לאחר שהמשחק התחיל לקבל צורה, נעם איחד את כל הסצנות לפרויקט אחד כך שכל סצנה תדבר עם סצנה אחרת.

בשלב זה דור ונעם התחילו לעבוד על צד השרת, לכתוב את ה Web Api לפי הדרישות של המשחק, מאחר ו Unity הוא One-Threaded זה הקשה מעט על בניית ה Api כי היינו צריכים לעשות כל מיני מעקפים במקומות מסוימים. (גם מהבחינה שלא הכרנו בכלל את Unity ולמדנו אותו מאפס)

לאחר מכן עשינו בדיקות מקומיות לראות שצד השרת עובד טוב עם השרת, עופר ישב על הטמעת צד השרת בלוח המשחק למטרת סנכרון שני השחקנים בחדר ובנוסף, תיקנו באגים שהיו קיימים ודברים שעבדו לא בסנכרון מלא.

לאחר שהכל עבד העלאנו את ה DB וה Web Api לענן של Azure Devops וסיימנו.

סיכום

בראשית הסמסטר חיפשנו נושא לפרויקט אשר יכול לעניין כל פרט בצוות. עלו כמה רעיונות אך בסופו של דבר הרעיון הבולט היה לשחזר משחק ישן, אהוב, שכולנו מכירים, אשר כבר שנים אינו קיים יותר. המשחק המקורי רץ על PC והיה חלק מגרסאות ה- ICQ הראשונות.

לאחר שהחלטנו על רעיון, חיפשנו את הפלטפורמה המתאימה ביותר לעבודה על מנת להפיק את המשחק על הצד הטוב ביותר. נבחנו שתי אופציות: Android Studio (Java) או Unity (C#). בסופו של דבר החלטנו לפתח ב Unity -משום שרובנו לא התנסנו בעבר עם הטכנולוגיה הספציפית הזאת והחלטנו לנצל את ההזדמנות ללמידה עצמית ולהרחבת הידע.

מכאן השלב הבא היה ללמוד את הסביבה והשפה ולהתחיל בעבודה. בתור התחלה חילקנו את העבודה בינינו והתחלנו לעבוד במרץ על העיצוב והלוגיקה של לוח המשחק עצמו. לאחר מכן התקדמנו לעבוד על תפריטי הניווט ועל המעבר בין החלונות השונים ולבסוף הרמנו את כל התקשורת עם השרת והתחלנו בבדיקות איכות.

בשלב בחירת השרת התעכבנו קצת משום שהבחירה הראשונית הייתה להשתמש ב- AWS אך נוכחנו לגלות שהשילוב עם הUnity - הוא מסורבל ואינו מוטמע ביעילות בסביבת העבודה. לכן לאחר חשיבה מחדש החלטנו לעבור ל- AZURE שהיווה פתרון פשוט ונוח יותר.

במהלך הפרויקט נחשפנו לשפות וכלים חדשים, למדנו לעבוד כצוות על פרויקט משותף ובעצם התנסנו בפועל בגישות ה Agile ובתהליך ההנדסה הכללי אותם למדנו בקורס הנדסת תוכנה. לאחר שהתמודדנו עם קשיים שונים בתהליך, אנו מרגישים שהגענו לתוצאה מספקת ומשביעת רצון ביחס לזמן שניתן לנו. כמו כן, נהנו מהעבודה ומהתהליך והשכלנו ללמוד דברים חדשים בדרך.

1. [לחץ כאן](https://he.wikipedia.org/wiki/%D7%90%D7%91%D7%9F_%D7%A0%D7%99%D7%99%D7%A8_%D7%95%D7%9E%D7%A1%D7%A4%D7%A8%D7%99%D7%99%D7%9D) להסבר מתוך ויקיפדיה. [↑](#footnote-ref-1)