

**"טטריס"**

**מסמך עיצוב**

**גרסה: 1.0**

**שם הסטודנטיות: עדי עובדיה וחן דהן**

**ת.ז:** **307934729, 203191630**

**מועד הגשה: 3.5.17**

**תוכן עניינים**

[הקדמה 3](#_Toc481606522)

[הגדרות 4](#_Toc481606523)

[ניתוח איומים 5](#_Toc481606524)

[ארכיטקטורת המערכת 7](#_Toc481606525)

[עיצוב נתונים ופרוטוקולים 8](#_Toc481606526)

[User Stories 9](#_Toc481606527)

[Security User Stories 10](#_Toc481606528)

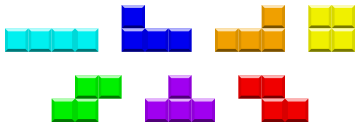
[Test Units 10](#_Toc481606529)

# **הקדמה**

הפרויקט אותו אנו בחרנו לבנות הוא משחק הטטריס הנקרא גם פאזל הבלוק הנופל.

במשחק קיימות 7 צורות טריגונומטריות שונות, כאשר כל צורה מורכבת מ-4 ריבועים.

הצורות הן:



במהלך המשחק על השחקן לכוון את הצורות הנופלות (במהירות קבועה) מחלקו העליון של המסך כאשר המטרה היא לסדר את הצורות בתחתית המסך ללא רווחים ברצף הקוביות. ניתן גם לסובב את הצורות. כאשר נוצרת שורה מסודרת ללא רווחים היא נמחקת ונצברת נקודה לטובת השחקן.

כאשר נוצרת ערימת קוביות עד לקצה העליון של המסך המשחק נגמר.

התוצאה היא מספר השורות המלאות שנמחקו.

צורות הטטריס אינן תמונות, אלא הן ציורים הנוצרים באמצעות Swing drawing API.

הרעיון הפונקציונלי מאחורי המשחק:

* משתמש ב-Timer Class על מנת ליצור משחק חדש
* צורות הטטריס נמשכות מטה
* הצורות נעות על בסיס ריבועי
* מתמטית לוח הוא רשימה פשוטה של מספרים

# **הגדרות**

* שרת – תוכנה המבצעת את השירות הנדרש. תוכנת השרת מבוססת על האזנה לבקשה (request) כלשהי ומענה בתגובה (response) לבקשה שהתקבלה. הבקשה יכולה להגיע בכל זמן ועל השרת להיות מוכן אליה תמיד.
* לקוח – ממשק המשתמש הפועל על מחשב המשתמש. מקבל את המידע הדרוש מהשרת.
* בסיס נתונים – נקרא גם מסדר נתונים. ומשמש לאחסון מסודר של נתונים במחשב, לשם אחזור ועיבוד. האחסון הוא על ידי אמצעי אחסון נתונים המאפשר גישה ישירה לנתונים באמצעות תוכנה ייעודית. יחידת הנתונים הקטנה ביותר היא השדה (field), ואוסף שדות המתארים משהו מסוים נקראים רשומה (record). לכל רשומה יש מפתח ראשי (primary key) המשמש לזיהוי חד משמעי שלה. קיים גם מפתח משני, ומפתח המורכב מכמה שדות נקרא מפתח מורכב.
* Class Timer - ישות לתזמון תהליכים במערכת, באופן חד פעמי או לביצוע חוזר של משימות במרווחי זמן קבועים. מספר תהליכונים יכולים "לחלוק" טיימר אחד ללא צורך בסנכרון חיצוני.

# **ניתוח** **איומים**

הצגת ניתוח האיומים למערכת בהתאם למודל

STRIDE.

STRIDE – סכמת סיווג על מנת לאפיין

איומים לפי סוגי ה- exploit.STRIDE עובד לפי

ראשי התיבות של הקטגוריות הבאות:

* Spoofing Identity: קבלת גישה למערכת ע"י התחזות למשתמש מורשה. התוקף יכול להתחזות לשחקן וכך לקבל גישה למשחק ולבסיס נתונים, לגלות את החולשות ולנצל אותם לפי ההתקפות שלו.
  + חומרת האיום: 3, אמנם התוקף יכול לקבל גישה כשחקן בקלות אך הוא לא יכול לחשוף הרבה חולשות מצד השחקן.
* Tampering with data: הרס מכוון או מניפולציה של נתונים בזמן העברתם.

ישנה אפשרות שהתוקף ישלח תוצאות שונות של המשחק אל בסיס הנתונים דרך השרת או ישנה את פרטי השחקן.

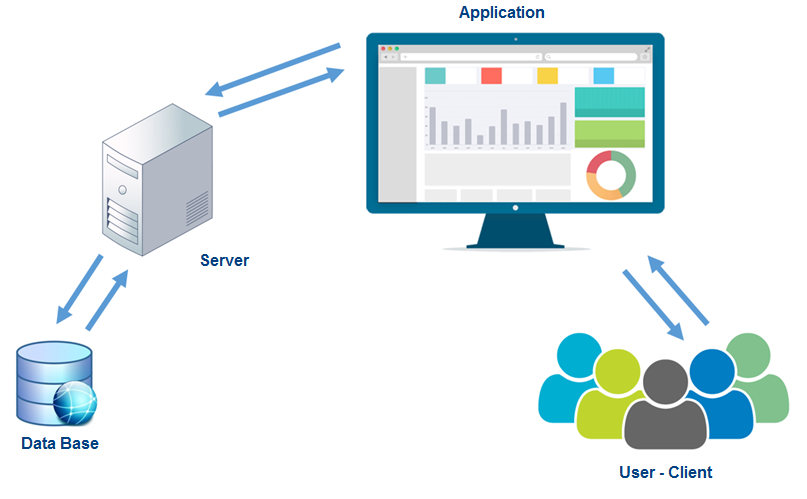
* + חומרת האיום: 4, החומרה גבוהה מכיוון שישנה אפשרות גדולה שהתוקף יצליח להשתלט על העברת הנתונים ולהזיק לנתונים בבסיס הנתונים.
* Repudiation: הכחשה של קבלת נתונים ושליחת בקשה חדשה עבור הנתונים. התוקף יכול לשלוח מס' בקשות לקבל את טבלת המובילים בטענה שהוא לא קיבל את הנתונים בעבר וכך לקבל גישה לבסיס הנתונים ע"י העמסת השרת.
  + חומרת האיום: 3, לתוקף יהיה קל לשלוח כמות של בקשות אך הנזק לא יהיה גדול.
* Information disclosure: משתמש חשוף למידע שהוא לא מורשה אליו. התוקף הצליח לגשת לקוד ולצפות בו וכך הוא יודע איפה החסרונות של המשחק ואיך הניקוד מחושב וכך לנצל את חולשות אלו לפריצה או למשחק מוטעה.
  + חומרת האיום: 4, אמנם יהיה קשה לתוקף לגשת לקוד של המערכת אך ברגע שהוא יקבל אתת הגישה הוא יכול להרוס את המערכת ברגע.
* Denial of service: התוקף מונע מהמשתמש את השירות שהוא אמור לקבל ע"י הצפה של המערכת.

ישנה אפשרות שהתוקף ינסה להפעיל כמות רבה של משחקים עד להגעה למצב שהשרת עמוס וכך יצליח להשתלט על המערכת.

* + חומרת האיום: 5, קל מאוד לתוקף לשלוח הרבה בקשות למשחק חדש ולגרום להצפה של המערכת וכך להשביתה.
* Elevation of privilege: התוקף מקבל הרשאות שלא אמורות להיות לו. התוקף הצליח לקבל הראשות של מנהל ולגשת לקוד של המשחק ולשנות אותו לפי צרכיו. למשל: להגדיל את המהירות בה הצורה יורדת, להמציא צורות חדשות וכו' או לחילופין להשיג לו יתרון במשחק ע"י העלאת נקודות בבסיס הנתונים, האטת ירידת הצורה כך שיהיה לו יותר זמן לשחק ועוד.
  + חומרת האיום: 4, לתוקף יהיה קשה לקבל הרשאות של מנהל אך ברגע שהוא יקבל הוא יוכל להרוס את המערכת.
* סיווג חומרת איום: 5-האיום גבוה במיוחד, 1-האיום נמוך.

# **ארכיטקטורת המערכת**

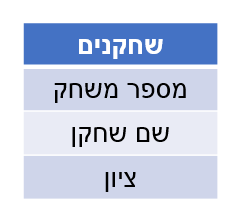
מודל ארכיטקטורת המערכת



# **עיצוב נתונים ופרוטוקולים**

מסד הנתונים בפרויקט זה הוא עבור שמירת נתוני השחקנים (משתמשים).

ישנה טבלה אחת בה עבור כל משחק נשמרים שם השחקן וציונו וכך תהיה אפשרות להציג טבלת מובילים במשחק.



# **User** **Stories**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| EPIC | Feature | As a <role>  תפקיד | I Want <goal/desire>  מטרה | So that <benefit> |
| תנועת קובייה | Client | שחקן | להפוך את הקובייה | השחקן יכול להכניס את הקובייה בצורה המתאימה למקום הפנוי בשורה |
| תנועת קובייה | Client | שחקן | להזיז את הקובייה ימינה או שמאלה | השחקן יכול להכניס את הקובייה למקום המתאים הפנוי |
| טבלת מובילים | Client, DB, Server | שחקן | הצגת המתחרים המובילים | השחקן יכול לראות איזה אף הוא צריך לעבור על מנת להיות בטבלת האלופים |
| משחק | Client | שחקן | להתחיל משחק | השחקן יכול להפעיל משחק חדש להתחיל לשחק |
| משחק | Server | מערכת | הפעלת משחק | הפעלת משחק חדש עבור השחקן |
| שמירה בבסיס נתונים | DB,Server | מערכת | שמירת שחקן בבסיס הנתונים | שמירת נתוני השחקן בבסיס הנתונים והכנסה לטבלה המתאימה |
| תנועת קובייה | Server | מערכת | הורדת קובייה חדשה | כאשר הקובייה הקודמת ירדה למטה עד הסוף תצא קובייה חדשה עבור השחקן |
| מחיקת שורה ברגע שהושלמה | Server | מערכת | מחיקת שורה ברגע שהושלמה | ברגע ששורה מושלמת ייצברו נקודות עבור השחקן |
| פתיחת משחק | Server | מערכת | פתיחת משחק | פתיחת משחק עבור משתמש חדש |
| משחק | Server | מערכת | פסילה וסיום המשחק | ברגע שהקוביות מגיעות לקצה המסך העליון תתרחש פסילה ויסתיים המשחק |

# **Security User Stories**

1. רק לשרת יש אפשרות להוסיף, לעדכן, למחוק רשומות מהבסיס נתונים. כך למשתמש אין גישה לבסיס נתונים חוץ מאשר צפייה בטבלת המובילים.
2. הנתונים מועברים מהלקוח לשרת כך שלא נאבד מידע חשוב לגבי בקשה למשחק חדש, פעולות בבסיס נתונים וכו'.

# **Test** **Units**

1. **תוכנת המשחק**
   1. תנועת הצורה ימינה – על ידי לחצני המקלדת
   2. תנועת הצורה שמאלה – על ידי לחצני המקלדת
   3. הפיכת הצורה היוצאת – על ידי לחצני המקלדת
   4. מחיקת שורה שהושלמה במלואה – כאשר שורה מושלמת באמצעות הצורות השורות יש לבדוק שהיא נמחקה
   5. עצירת המשחק במקרה והקוביות הגיעו לקצה העליון של המסך
   6. הצגת צורה חדשה כאשר הצורה הקודמת הגיעה לתחתית
   7. הפלת צורה מטה על ידי לחיצה על רווח (space)
   8. הפסקת המשחק על ידי לחיצה על p
   9. הצגת טבלת שחקנים מובילים במשחק
2. **שרת**
   1. התחברות לבסיס הנתונים
   2. קבלת בקשות מהלקוח
3. **לקוח**
   1. שליחת בקשות לשרת
   2. הצגת התוכן המבוקש
   3. פתיחת משחק חדש
4. **בסיס נתונים**
   1. יצירת בסיס הנתונים
   2. עדכון בסיס הנתונים
   3. שמירת הנתונים בבסיס הנתונים