

פרויקט איקס-עיגול

צרו אתר המאפשר למשתמש לשחק במשחק איקס-עיגול נגד המחשב.

חוקים:

- קיים לוח בגודל 3X3 המאפשר בכל תא להציג X או O
- ברגע שיש שלושה איקסים בשורה או בטור או באלכסון – האיקס מנצח
- ברגע שיש שלושה עיגולים בשורה או בטור או באלכסון – העיגול מנצח
- המשתמש תמיד מתחיל והוא תמיד איקס
- כאשר המשתמש מקליק עם העכבר על תא – התא הזה מציג תמונה של איקס
- לאחר Delay של שנייה אחת, המחשב מוצא תא מתאים ומציג בו תמונה של עיגול
- אם המשתמש לוחץ על תא שכבר יש בו תמונה – לא קורה כלום
- אם המשתמש לוחץ מחוץ ללוח – לא קורה כלום
- אם המשתמש מנצח – לאחר Delay של שנייה אחת – מוצגת ההודעה: "ניצחת את המחשב!"
- אם המחשב מנצח – לאחר Delay של שנייה אחת – מוצגת ההודעה: "המחשב ניצח!"
- אם יש תיקו (כאשר רק אם אין יותר תאים פנויים) – לאחר Delay של שנייה אחת – מוצגת ההודעה: "תיקו"

ה-Components שיש לבנות:

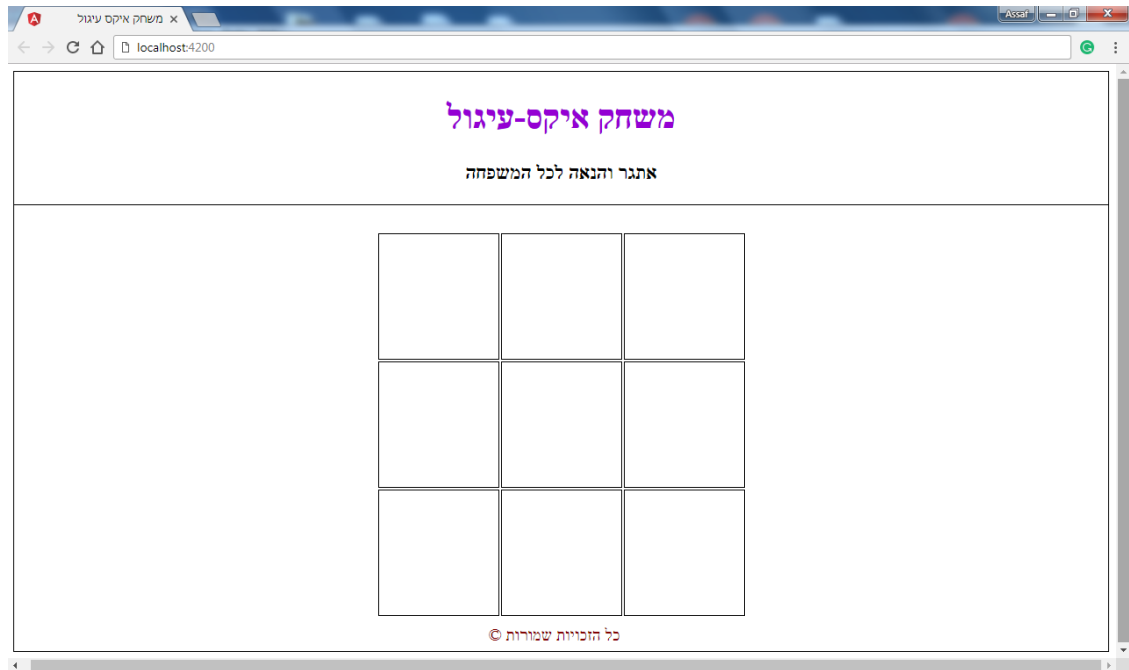
- Layout – מכיל את תוכן הדף כולו – כותרת עליונה, לוח, כותרת תחתונה
- Header – מכיל את הכותרת העליונה
- Footer – מכיל את הכותרת התחתונה
- Board – מכיל את הלוח
- Cell – מכיל תא אחד בלבד שקיים בלוח
- כל Component נוסף שאתם חושבים שצריך לבנות
- אין צורך לבנות מנגנון Routing כי קיים כאן רק "דף" אחד.

הנוהל שבו המחשב מוצא את הצעד שלו:

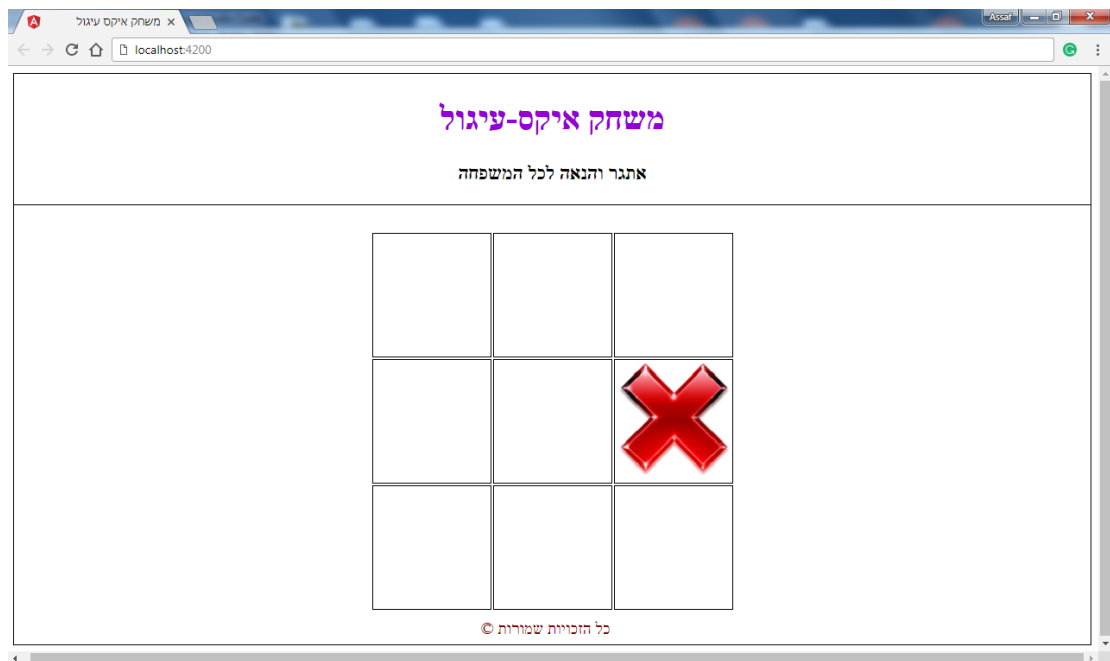
- א. על המחשב לבדוק ראשית אם הוא יכול לנצח. אם כן – עליו לבצע זאת ולנצח.
- ב. אם אין אפשרות לנצח בצעד הנוכחי – על המחשב לבדוק האם הוא יכול לחסום את המשתמש מלנצח בצעד הבא, כי אם המשתמש יכול לנצח בצעד הבא והמחשב לא יחסום אותו – אז כמובן שהמשתמש יוכל לנצח בצעד הבא. לכן אם יש מצב כזה – על המחשב לחסום את המשתמש מלנצח בצעד הבא.
- ג. אם אין אפשרות לנצח או לחסום את המשתמש מלנצח – על המחשב להגריל את מיקום הצעד הנוכחי, כמובן מהתאים הריקים.

להלן סדרה של תמונות הממחישות את ה-UI של המשחק:

בתחילת המשחק – כאשר האתר מורץ – מוצג לוח ריק:



המשתמש תמיד מתחיל, ותמיד הוא X. להלן תמונה של צעד ראשון שהמשתמש יכול לבחור:



כעת תור המחשב. לאחר שנייה אחת, המחשב מוצא צעד מתאים. בגלל שהמחשב לא יכול לנצח כעת ובגלל שאין למחשב אפשרות לחסום את המשתמש מלנצח, המחשב מגריל מיקום כלשהו מבין המקומות הפנויים. לדוגמה:



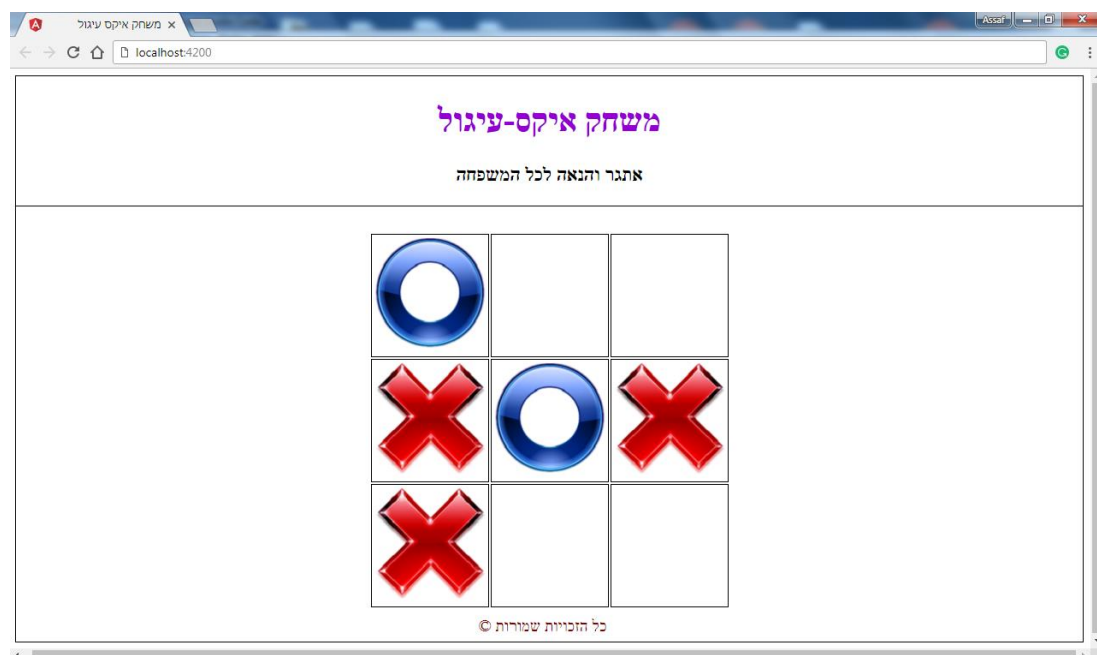
כעת תור המשתמש, שיכול לדוגמה לבחור ב:



בשלב הזה תור המחשב, והוא בודק האם הוא יכול לנצח – והוא לא, ולכן הוא בודק האם הוא יכול לחסום את המשתמש מלנצח בשלב הבא – והוא כן! לכן המחשב בשלב הזה חייב לחסום את המשתמש מלנצח, אחרת המשתמש ינצח בשלב הבא. לפיכך, המחשב יהיה חייב לבחור בדוגמה הזו את הצעד הבא:



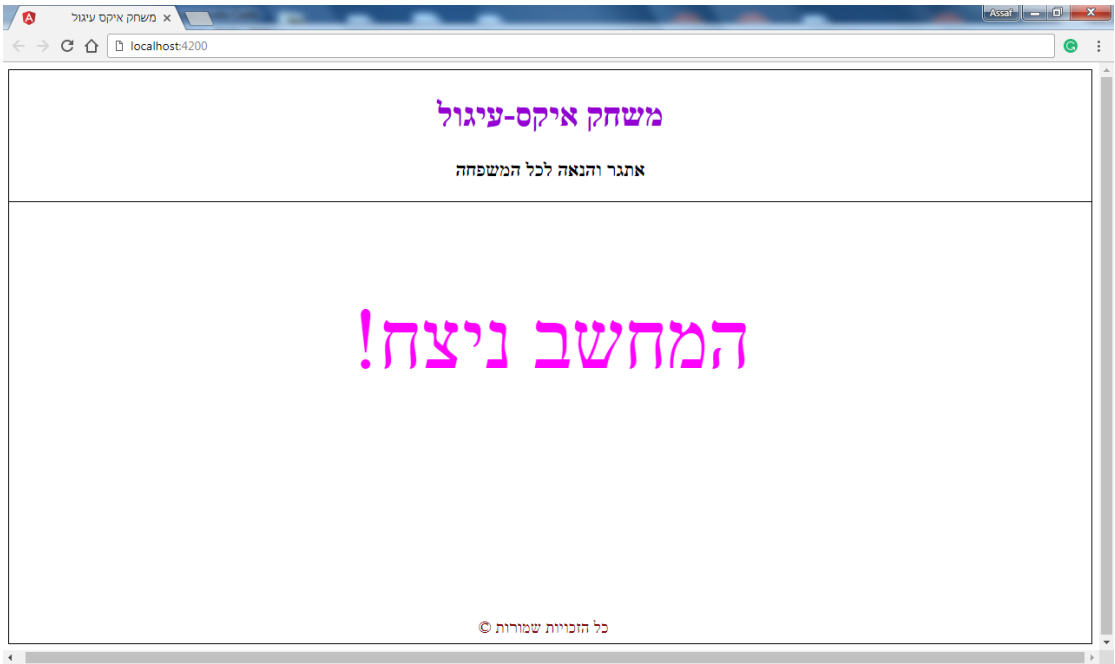
כעת תור המשתמש. נניח והמשתמש בוחר את הצעד הבא:



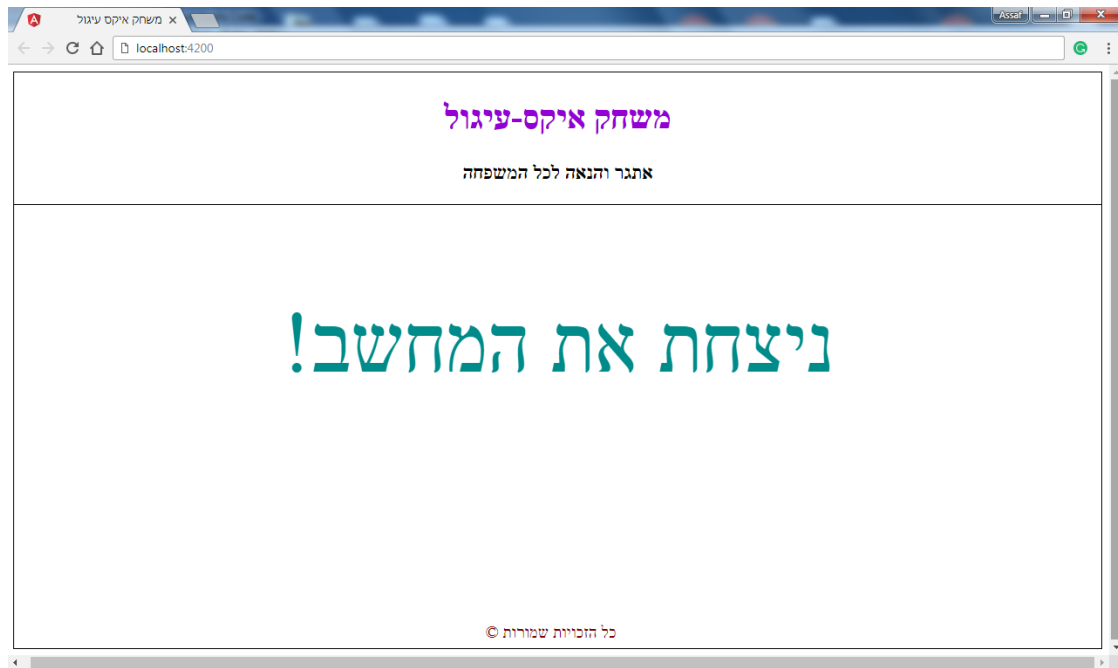
כעת תור המחשב, והוא כמו תמיד בודק ראשית אם הוא יכול לנצח – והוא אכן יכול לנצח! לכן הוא חייב לבחור במיקום שיביא לו את הניצחון, שזה המיקום הבא:



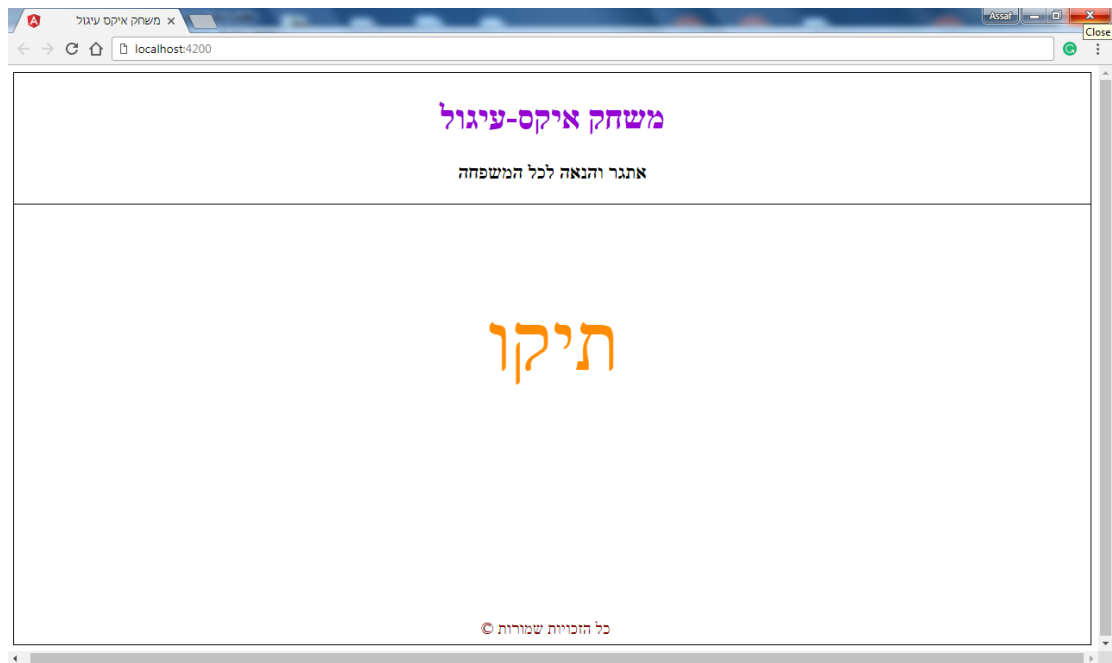
כאן, לאחר Delay של שנייה אחת, על האתר להציג את ההודעה הבאה:



במידה והמשתמש מנצח, יש להציג את ההודעה הבאה:



במידה ויש תיקון – יש להציג את ההודעה הבאה:



שימו לב:

בסיום המשחק ניתן להציג את ההודעה הסיום מבלי להעלים את הלוח כך שהמשתמש יראה גם את הלוח וגם את הודעת הסיום.

אפשר להשתמש בתמונות הבאות עבור האיקס והעיגול:



דגשים נוספים:

- במידה והגיע מצב שבו חייב להיות תיקו, למרות שהמשחק לא נגמר – אין להפסיק את המשחק אלא יש לאפשר למשתמש להמשיך ולשחק (אפילו שזה לא יוביל לשום ניצחון או הפסד) ורק בסיום כל הצעדים של המשתמש והמחשב, כאשר אין יותר תאים ריקים, להציג את הודעת ה-"תיקו".
- על הפרויקט לעבור קומפילציה ולרוץ ללא שגיאות.
- אסור לפרויקט לקרוס בשום מצב.
- כדאי לתעד את הקוד.
- יש לכתוב קוד אסתטי הכולל ריווח, הזחות (Indentations), רווח אחד בין פונקציה לפונקציה וכו'.

בהצלחה ☺