

N-in-a-Row

תרגיל 3

מגישים:

עידן גור, ת.ז. 307948836 , אי-מייל: idangoor1@gmail.com

URL למשחק: <http://localhost:8080/NinaRow>

ממשק המשחק:

צד הלקוח נבנה ב-React והמעבר בין שלושת המסכים שהוגדרו בתרגיל הוא על בסיס עיקרון של מחסנית, כלומר בכל מעבר מוצלח למסך הבא אנחנו למעשה מלבישים את המסך החדש על גבי המסך הישן. כמו כן הממשק נבנה כך שיהיה רספונסיבי לגודל חלון הדפדפן, בעזרת שימוש ב-media queries, על מנת שהמשחק יישאר פונקציונלי ונוח בעת ביצוע resize. ברגע שהמשתמש הצליח להתחבר בהצלחה למערכת, הוא יוכל לבחור האם להעלות משחק חדש על ידי בחירה של קובץ XML בפורמט המתאים ולחיצה על כפתור ה-Upload, או שמא לבחור משחק מבין המשחקים שהעלו שחקנים אחרים למערכת. אם בחר להעלות משחק בעצמו יוכל גם למחוק אותו בשלב מאוחר יותר, אך ורק במידה שאין באותו רגע משתמשים אחרים במשחק. ראוי לציין גם שרק המשתמש שיצר את המשחק הוא זה שיוכל למחוק אותו.

כל משחק ברשימה מיוצג על ידי בועה הכוללת בתוכה את כל המידע הרלוונטי (שם המשחק, שם היוצר, גודל הלוח, יעד הרצף, סוג המשחק, כמות שחקנים בפועל מתוך כמות שחקנים נדרשת, וסטטוס המשחק). בנוסף יופיעו שני כפתורים המאפשרים למשתמש להתחבר למשחק בתוך שחקן/צופה מן הצד.

ברגע שממשק מסתיים אז מאחורי הקלעים כל השחקנים שבו "נבעטים" החוצה באופן כפוי, על מנת שהמשחק יהפוך מיידית לזמין מחדש עבור שחקנים אחרים המעוניינים להיכנס. חשוב להדגיש – לשחקן בצד הלקוח לא משוקף שהוא כבר אינו במשחק, אלא עדיין מיוצג לו מצב הלוח, הודעה על תוצאות המשחק, וכל שנותר עליו לעשות הוא לצאת בכוחות עצמו בעזרת לחיצה על הכפתור Leave Game.

במסך המשחק ניתן לראות את הפרטים הבסיסיים של המשחק, רשימת השחקנים, רשימת הצופים, לוח המשחק, וצ'אט (במידה והמשתמש התחבר כשחקן, ולא כצופה). ברגע שממשק התחיל אז סטטוס המשחק שלו ישתנה ל-ACTIVE וכן ניתן יהיה לראות שהשחקן שזהו תורו יצוף באנימציה וצבעו יהיה בולט.

שחקן ממוחשב אינו יכול לפרוש ממשחק פעיל כנדרש, אך בחרתי לתת לו את האפשרות לבצע זאת כל עוד המשחק עוד לא התחיל או לחלופין כשהמשחק נגמר, וזאת על מנת למנוע צימוד הדוק בין המשחק עצמו לשחקני המחשב שהתחברו אליו. כל הודעת שגיאה על פעולה שאינה תקינה תצוץ בצבע אדום בחלקו העליון של הדפדפן.

בונסים

צופה מן הצד – שחקן יכול לבחור להצטרף למשחק כצופה מן הצד. משמע שהוא יוכל לראות מה מצב המשחק, מה מצב הלוח והשחקנים, אך אינו יכול להשתתף בצ'אט.

צ'אט – שחקנים יכולים לנהל שיחה בצ'אט שנמצא מתחת ללוח המשחק. לעומת זאת, צופים מן הצד לא יקבלו גישה לצ'אט כלל כנדרש.

ארכיטקטורה

בתרגיל זה מימשתי מודול NinaRowEngine המכיל את כל הלוגיקה של המשחק, ומודול NinaRowApp האחראי לממשק המשתמש בצד הלקוח ול-servlets של צד השרת.

NinaRowEngine

מחלקה	תפקיד והסבר
GameDescriptor	מייצג קונטיינר המכיל את האלמנטים המרכזיים של המשחק כדוגמת המשחק עצמו, השחקנים, ההיסטוריה וכדומה.
Game	מייצג את המשחק עצמו.
Board	מייצג את לוח המשחק בעזרת מערך דו מימדי של Disc.
DynamicPlayers	מייצג קולקציה של שחקנים וקולקציה של צופים, ואחראי על ניהול התורות שלהם.
Player	מייצג שחקן.
Disc	מייצג דיסקית.
Connect	מייצג רצף של דיסקיות.
GameSerializer	מחלקה המאפשרת לבצע סיריליזציה של קובץ XML ללוגיקה של משחק.
Chat	מייצג צ'אט של משחק.
GameManager	מייצג קולקציה של כל המשחקים במערכת ומאפשר ניהול בסיסי שלהם (הוספה, מחיקה וכו').
PlayerManager	מייצג קולקציה של כל השחקנים במערכת ומאפשר ניהול בסיסי שלהם (הוספה, מחיקה וכו').

NinaRowApp

Servlet	תפקיד והסבר
NinaRowServlet	הבקשה הראשונה שנשלחת לשרת כאשר המשתמש נכנס לראשונה לאתר. השרת יפנה את המשתמש לקובץ index.html שמכיל את התצוגה הרצויה.
LoginServlet	מחולק ל-2: בקשת GET מוצלחת תחזיר את הפרטים של המשתמש במערכת (שם וסוג השחקן). בקשת POST מוצלחת תיצור שחקן חדש במערכת.
LogoutServlet	בקשה לשרת לטובת מחיקת שחקן מהמערכת.
UploadGameServlet	בקשה להעלאת משחק חדש למערכת על ידי קובץ XML.
DeleteGameServlet	בקשה למחיקת משחק מן המערכת.
PlayerListServlet	בקשה לקבלת רשימת כל השחקנים במערכת.
GameListServlet	בקשה לקבלת רשימת כל המשחקים במערכת.
JoinPlayerServlet	בקשה לצירוף המשתמש כשחקן למשחק מסוים.
JoinVisitorServlet	בקשה לצירוף המשתמש כצופה למשחק מסוים.
leaveGameServlet	בקשה לפרישת המשתמש מן המשחק.
GameInfoServlet	בקשה לקבלת פרטים אודות המשחק.
BoardServlet	בקשה לקבלת פרטים אודות לוח המשחק.
ChatServlet	בקשה לקבלת הודעות הצ'אט במשחק.
SendChatServlet	בקשה לשליחת הודעה חדשה לצ'אט.
PopOutServlet	בקשה לביצוע PopOut על ידי השחקן.
PushInServlet	בקשה לביצוע PushIn על ידי השחקן.