



به موارد زیر توجه کنید:

- مهلت ارسال پروژه تا ساعت ۲۳:۵۹ روز اعلام‌شده است.
- اسم‌گذاری کلاس‌ها و متدها باید به گونه‌ای باشد که وظیفه‌ی متد یا کلاس را به درستی بیان کند (به گونه‌ای که احتیاج به کامنت‌گذاری به حداقل برسد).
- برای طرح سوال و پرسش و پاسخ فقط و فقط از پیتزای درس استفاده کنید.
- پیشنهاد می‌شود چند روز صرفاً روی ساختار کلاس‌ها و توابع فکر کنید. اگر ساختار کد‌ها مناسب نباشد، ممکن است به اتمام رساندن پروژه به قدری سخت شود که نتوانید تمام بخش‌ها را پیاده‌سازی کنید.
- پروژه به صورت گروهی و دو نفره خواهد بود.
- همکاری شما از طریق git خواهد بود. با توجه به ثبت پروژه در گیت، میزان تاثیر گذاری هر فرد در پروژه ارزیابی می‌شود و در صورت کم کاری قابل توجه یک نفر، نمره او نصف خواهد شد.
- پروژه خود را بر روی github یا gitlab بسازید و از امکانات بسیار کاربردی این ابزار استفاده کنید. برای آشنایی با گیت می‌توانید از این لینک استفاده کنید.
- تنها و تنها کد روی گیت تحویل گرفته خواهد شد و هر کدی غیر از آن قابل تحویل نیست.
- با توجه به مهلت ارسال نمرات، ددلاین قابل تمدید نیست. لذا به وقت دقت کنید.

خلاصه

در این پروژه، شما ۲ بازی معروف فکری را به همراه گرافیک و بخش شبکه پیدا سازی خواهید کرد. اولین بازی تخته یا Backgammon و دومین بازی شطرنج یا Chess است.

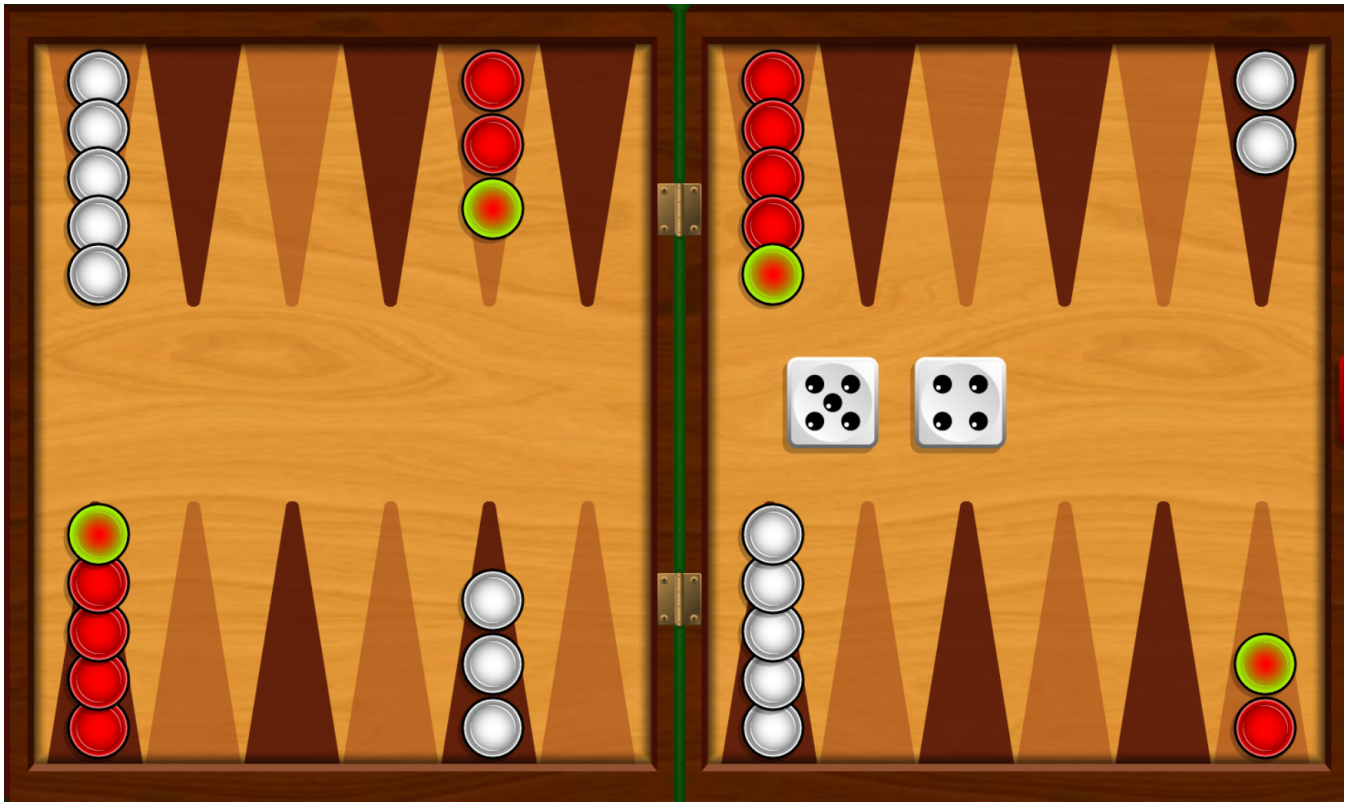
تخته

بازی تخته‌نرد یکی از بازی‌های باستانی ایرانی می‌باشد. دقت کنید شما قرار است با طراحی این بازی به یادگیری java بپردازید. java نه آلت قمار است و نه روی آن شرط بندی می‌شود؛ پس نگران نباشید!



این بازی دو بازیکن دارد که هر بازیکن ۱۵ مهره در اختیار دارد. هدف بازی آوردن مهره‌ها به خانه‌ی خودی و خارج کردن آن‌ها از بازی است. بازی که شما طراحی می‌کنید به صورتی است که دو نفر وارد بازی می‌شوند و طبق مکانیزمی که در قوانین توضیح داده شده؛ نوبت اول مشخص می‌شود و بازیکن‌ها شروع به بازی می‌کنند. در واقع لازم نیست کد شما قابلیت بازی کردن داشته باشد.

زمین بازی و چینش اولیه مهره ها به صورت زیر است:

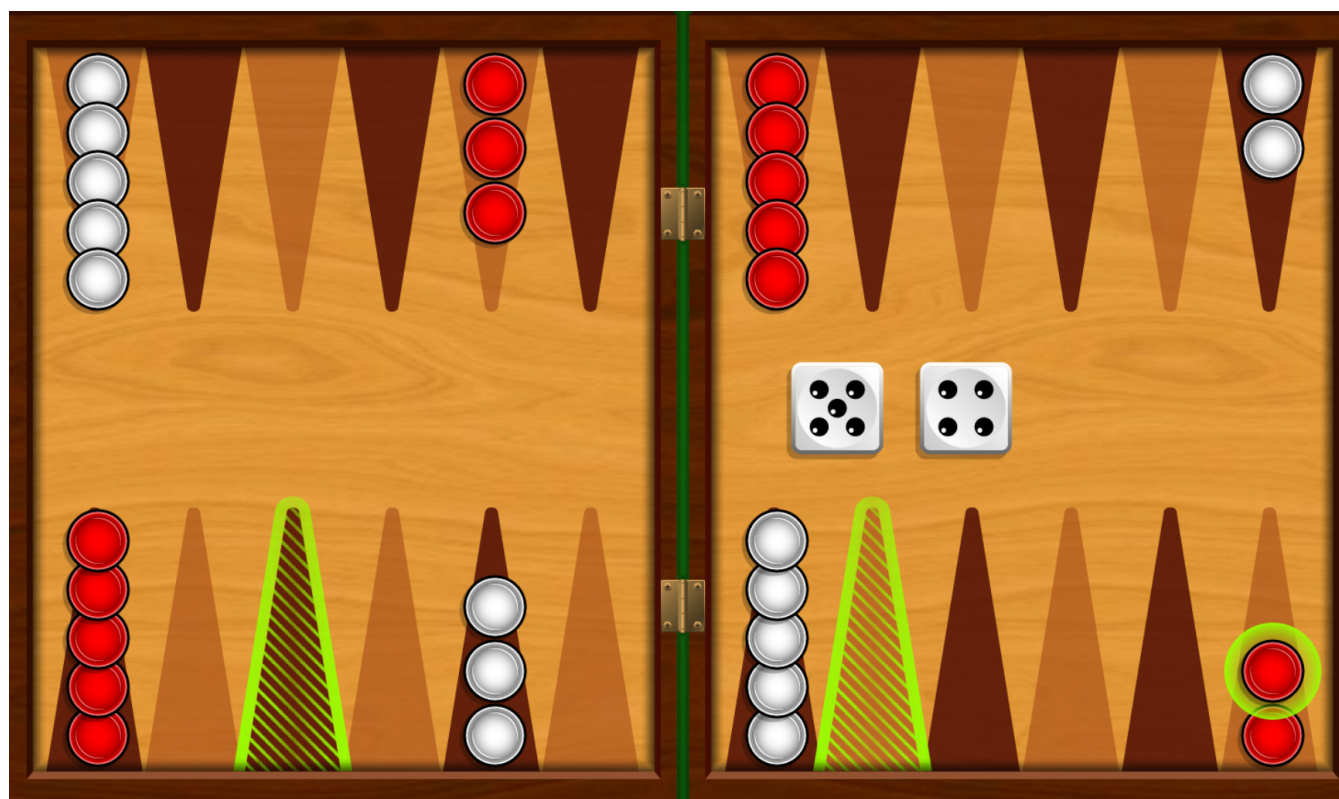


قوانین بازی را با جزئیات می‌توانید در بخش آموزش بازی در این لینک مشاهده کنید.

انتظارات پیاده‌سازی بازی:

- مدل بازی را با دقت و با استفاده از مفاهیم شی‌گرایی طراحی کنید.
 - پیشنهاد می‌شود ابتدا شمایی از اجزای احتمالی و ارتباط این اجزا با یکدیگر طراحی کنید.
 - لازم است قوانین بازی به صورت کاملاً صحیح پیاده سازی شده باشند و امکان بازی اشتباه یا تقلب وجود نداشته باشد.
- برای ارتباط میان مدل بازی و المان‌های گرافیکی api مناسبی طراحی کنید.
 - طراحی api بر عهده‌ی شماست اما دقت کنید که لازم است متدهای ارتباطی طراحی شده بر اساس عملکرد از یکدیگر جدا شده باشند. به عبارت دیگر، متدهایی با عملکردهای چندگانه و نامربوط طراحی نکنید.
 - دقت کنید که هرچه این api را با دقت بیشتری طراحی کرده باشید برای پیاده‌سازی فاز شبکه کار خودتان راحت تر خواهد بود.
- جمع‌آوری ویژگی‌های آماری بازی شامل موارد زیر:
 - تعداد مهره‌های زده شده از هر بازیکن
 - تعداد برد و باخت‌ها قبلی بازیکن
 - تعداد مهره‌های خارج شده از بازی
 - مجموع اعداد تاس‌های هر بازیکن تا کنون (تاس‌های جفت ۴ بار محسوب می‌شوند)
- پیاده‌سازی گرافیک بازی با استفاده از swing یا javafx

- از معماری Model View Controller در پیاده‌سازی خود استفاده کنید.
- زیبایی و روان بودن ui بازی نمره اضافی دارد.
- بازی نباید دچار وقفه شود یا crash کند.
- روان بودن کنترل بازی
- * کنترل می‌تواند به صورت drag & drop یا به صورت کلیک باشد.
- در صورتی که از کلیک برای کنترل استفاده می‌کنید حرکت مهره‌ها به صورت animation باشد. به عبارت دیگر مهره‌ها از مبدا به مقصد حرکت کنند و حرکت ناگهانی نباشد.
- تاس‌ها و نوبت فرد به صورت واضح نمایش داده شود.
- * برای شبیه‌سازی ریختن تاس می‌توانید از sprite animation بهره بگیرید.
- آمار بازی حین بازی و در اتمام بازی نمایش داده شود.
- در حین بازی ممکن است بازیکنی به این نتیجه برسد که باخت را قبول کند. بازی شما باید این قابلیت را داشته باشد که بازیکن باخت خود را قبول کند و امتیاز مناسب به حریف داده شود و بازی خاتمه یابد.
- بازی شما بر اساس تاس ریخته شده باید حرکت‌های ممکن کاربر را به او نشان دهد.
- تا زمانی که مهره‌ای انتخاب نشده است لازم است مهره‌هایی که قابلیت حرکت بر اساس تاس فعلی دارند highlight شوند. هنگامی که یکی از این مهره‌ها انتخاب شد (با کلیک یا drag) باید ستون‌هایی که مهره می‌تواند در آن‌ها فرود بیاید مشخص شوند.



- هنگامی که حرکت یکی از تاس‌ها انجام می‌شود باید تاس کمرنگ شود که بازیکن متوجه باشد که تاس استفاده شده است. تاس‌های جفت ۴ حرکت دارند در نتیجه اگر تاسی جفت آمد تعداد ۴ تاس از آن روی زمین نمایش داده می‌شود.

- هر بازیکن مادامی که در نوبت خودش قرار دارد می‌تواند با فشردن `ctrl+z` حرکت‌های قبلی خود در این نوبت را برگرداند.
 - پس از مصرف شدن آخرین تاس، ۳ ثانیه مهلت بازگشت به بازیکن داده می‌شود و سپس در صورتی که حرکت `undo` نشود؛ نوبت بازی به بازیکن حریف منتقل خواهد شد.
 - در صورتی که با این تاس هیچ مهره‌ای امکان حرکت ندارد؛ این موضوع باید به بازیکن اعلام شود و سپس، نوبت به حریف منتقل شود.
 - بازی پس از اتمام باید قابلیت `replay` و `new game` را داشته باشد. (این دو گزینه در این فاز تفاوتی با هم ندارند اما در بازی به صورت شبکه معنا خواهند داشت.)
 - قبل از شروع بازی، باید صفحه‌ای باشد که بازیکن بتواند تنظیمات این راند را تغییر دهد. برای مثال کل مدت نوبت هر بازیکن، تعداد دست‌های بازی تا اعلام برنده، و تنظیمات دیگری که در بازی نیاز است.
 - امتیازی: یک قابلیت سیو برای بازی در حال انجام طراحی کنید. بازی کنونی روی فایلی ذخیره شود و بعداً با گزینه `load`، همان بازی عیناً ادامه یابد. طبعاً این قابلیت نباید محدودیتی روی تعداد سیوها یا انتخاب فایل سیو بگذارد.
- در پایان نکاتی که برای طراحی خوب است به آن دقت کنید آورده می‌شود:
- شماره‌گذاری هوشمندانه ستون‌ها، که ممکن است برای هر بازیکن متفاوت باشد، به پیاده‌سازی شما کمک خواهد کرد.
 - مکانیزم جمع‌کردن مهره‌ها از خانه نیاز به توجه ویژه‌ای در مدل شما خواهد داشت.
 - ستون‌های دارای بیش از یک مهره بسته هستند.
 - فرآیند زدن مهره‌ها باید به خوبی طراحی شود.
 - هنگامی که یک بازیکن مشغول جمع‌کردن مهره‌ها از خانه است ممکن است مهره‌اش زده شود.
 - نشان‌دن کشته نیاز به عدد مستقیم تاس دارد.
- برای آشنایی با بازی می‌توانید در این سایت، بازی را امتحان کنید.

شطرنج

شطرنج بازی ای است در یک جدول ۸ در ۸ که با دو رنگ سیاه و سفید رنگ آمیزی شده و دارای انواع مهره های شاه (King)، وزیر (Queen)، فیل (Bishop)، اسب (Knight)، قلعه (Rook) و سرباز (Pawn) است. هر کدام از این مهره ها نحوه حرکت خاصی دارند و یک خانه توسط دو مهره اشغال نمی شود. لذا اگر مهره ای به خانه ای برود که قبلاً در آن مهره ای ساکن بوده، مهره قبلی از بازی خارج می شود.

هدف این بازی، کیش و مات کردن است. یعنی حالتی که شاه رقیب با یک حرکت قابل حذف باشد و همچنین راهی برای فرار از این موقعیت نداشته باشد یا به عبارتی، تمام خانه هایی که می تواند به آن برود نیز قابل زدن باشند. به علت شهرت غیر قابل انکار بازی، قوانین بازی نیازی به بحث ندارند اما برای آشنایی کامل میتوانید از این لینک، این لینک و همچنین این لینک بازدید نمایید. دقت کنید که معیار قوانین ۳ لینک بالاست.



انتظارات پیاده‌سازی بازی: بسیاری از موارد پیاده‌سازی شباهت به بخش قبلی دارند اما تفاوت‌هایی نیز وجود دارد. لطفاً با دقت مطالعه کنید.

- مدل بازی را با دقت و با استفاده از مفاهیم شیء‌گرایی طراحی کنید.
- پیشنهاد می‌شود ابتدا شمایی از اجزای احتمالی و ارتباط این اجزا با یکدیگر طراحی کنید.
- لازم است قوانین بازی به صورت کاملاً صحیح پیاده‌سازی شده باشند و امکان بازی اشتباه یا تقلب وجود نداشته باشد.
- برای ارتباط میان مدل بازی و المان‌های گرافیکی api مناسبی طراحی کنید.
- طراحی api بر عهده‌ی شماست اما دقت کنید که لازم است متدهای ارتباطی طراحی شده بر اساس عملکرد از یکدیگر جدا شده باشند. به عبارت دیگر، متدهایی با عملکردهای چندگانه و نامربوط طراحی نکنید.
- دقت کنید که هرچه این api را با دقت بیشتری طراحی کرده باشید برای پیاده‌سازی فاز شبکه کار خودتان راحت‌تر خواهد بود.
- در صورتی که در بخش قبل api مناسبی طراحی کرده باشید، ممکن است بتوانید از آن در این فاز استفاده کنید. دقت کنید که از این مسأله استقبال می‌شود. چرا که نشان‌دهنده راحتی در بودن اضافه کردن یک بازی جدید به این پک است.
- جمع‌آوری ویژگی‌های آماری بازی شامل موارد زیر:
 - تعداد مهره‌های زده شده از هر بازیکن
 - تعداد برد و باخت‌ها قبلی بازیکن
 - تعداد مهره‌های خارج شده از بازی
 - مدت زمان میانگین تصمیم‌گیری هر بازیکن در نوبت خود
 - درصد استفاده از هر نوع مهره
 - امتیازی: heat map از فعالیت مهره‌ها روی برد
- پیاده‌سازی گرافیک بازی با استفاده از swing یا javafx
 - از معماری Model View Controller در پیاده‌سازی خود استفاده کنید.
 - زیبایی و روان بودن ui بازی نمره اضافی دارد.
 - بازی نباید دچار وقفه شود یا crash کند.
 - روان بودن کنترل بازی
 - * کنترل می‌تواند به صورت drag & drop یا به صورت کلیک باشد.
 - در صورتی که از کلیک برای کنترل استفاده می‌کنید حرکت مهره‌ها به صورت animation باشد. به عبارت دیگر مهره‌ها از مبدا به مقصد حرکت کنند و حرکت ناگهانی نباشد.
 - زمان باقی‌مانده و نوبت فرد به صورت واضح نمایش داده شود.
 - انیمیشن مناسب از حذف مهره‌ها یا وزیر شدن سربازها
 - آمار بازی حین بازی و در اتمام بازی نمایش داده شود.
- در حین بازی ممکن است بازیکنی به این نتیجه برسد که باخت را قبول کند. بازی شما باید این قابلیت را داشته باشد که بازیکن باخت خود را قبول کند و امتیاز مناسب به حریف داده شود و بازی خاتمه یابد.
- پس از انتخاب مهره، بازی شما باید حرکت‌های ممکن کاربر را به او نشان دهد.

• بازی باید کیش شدن، مات شدن و مساوی کردن (stalemate) را تشخیص دهد و عملکرد مناسب را نشان دهد

- در صورت کیش شدن، واضحا بازی نباید حرکتی را اجازه دهد که بازیکن را از وضعیت کیش در نمی آورد

• بازی پس از اتمام باید قابلیت replay و new game را داشته باشد. (این دو گزینه در این فاز تفاوتی با هم ندارند اما در بازی به صورت شبکه معنا خواهند داشت.)

• قبل از شروع بازی، باید صفحه ای باشد که بازیکن بتواند تنظیمات این راند را تغییر دهد. برای مثال کل مدت نوبت هر بازیکن، تعداد دست های بازی تا اعلام برنده، و تنظیمات دیگری که در بازی نیاز است.

• امتیازی: یک قابلیت سیو برای بازی در حال انجام طراحی کنید. بازی کنونی روی فایلی ذخیره شود و بعدا با گزینه load، همان بازی عینا ادامه یابد. طبعا این قابلیت نباید محدودیتی روی تعداد سیو ها یا انتخاب فایل سیو بگذارد.

در پایان نکاتی که برای طراحی خوب است به آن دقت کنید آورده می شود:

- تبدیل سرباز به وزیر نیازمند تمرکز ویژه ای است.
- به حذف مهره ها دقت شایانی کنید. بسیاری از باگ های به ظاهر غیر قابل رفع بخاطر اشیائی است که صحیح حذف نشده اند.
- دقت کنید که مساوی کردن در چند حالت مختلف ممکن است. تمامی حالات را بررسی و پیاده سازی کنید.
- اگر تلاش کنید برای تمامی مهره ها یک کلاس بندی کلی در نظر بگیرید و مهره های خاص از این کلاس کلی ارث بگیرند، کار شما بسیار آسانتر می شود.

بازی تحت شبکه

در این بخش لازم است بازی خود را به گونه ای تغییر دهید که بازیکن ها بتوانند تحت شبکه بازی کنند. در واقع نرم افزار شما شامل یک کد سرور خواهد بود که به تعداد دلخواهی کلاینت سرویس می دهد. کلاینت ها نمی توانند به صورت مستقیم با یکدیگر ارتباط بگیرند.

- بخش امتیازدهی و تولید عدد تاس الزاماً باید داخل سرور صورت گیرد.
- تصمیم با شماست که مدل بازی را به سرور منتقل کنید یا بخشی از مدل را در کلاینت ها باقی بگذارید.
- برای وارد شدن به حالت بازی به صورت شبکه، بازیکن ابتدا باید ثبت نام کرده باشد.
- دریافت نام، ایمیل و رمز عبور بازیکن در این مرحله ضروری است.
- همچنین باید ایمیل وی به نحوی **راستی آزمایی** شود تا مطمئن شوید بازیکن به آن دسترسی دارد.
- در صورتی که بازیکن قبلاً در سیستم ثبت نام کرده است، با وارد کردن ایمیل و رمز عبور خود، وارد می شود.
- بازیکن بازی مورد نظر خود را انتخاب می کند و در مرحله ی بعد، حریفی باید برای بازی با وی انتخاب شود.
- با انتخاب **جستجوی حریف**، بازیکنی که کمترین اختلاف امتیاز را با این بازیکن دارد (و در حال بازی نیست) انتخاب کنید (امتیاز در ادامه توضیح داده می شود). درخواستی برای حریف ارسال می شود، در صورتی که وی درخواست را نپذیرد، به سراغ بازیکن بعدی بروید که کمترین اختلاف امتیاز را با این بازیکن دارد.
- با انتخاب **بازی با دوستان**، بازیکن می تواند درخواست بازی را برای یک شخص خاص (که با ایمیلش مشخص می شود) ارسال کند. در صورتی که بازیکن مقابل در حال بازی باشد یا آنلاین نباشد این موضوع باید به اطلاع درخواست دهنده برسد.

نحوه امتیازدهی

امتیاز هر بازیکن همواره عددی نامنفی است که از مقدار ۱۰۰ شروع می شود. پس از اتمام هر بازی، پاداش آن از امتیاز بازنده کسر و به امتیاز برنده افزوده می شود.

- پاداش بازی برابر یک پنجم اختلاف امتیاز دو بازیکن، قبل از شروع بازی است.
- در صورتی که اختلاف امتیاز دو بازیکن صفر باشد، پاداش برابر یک دهم امتیاز یکی از آنها است.
- در صورتی که بازی مساوی شود پاداشی تعلق نمی گیرد.
- در صورتی که بازیکنی در میانه از بازی انصراف دهد، پاداش برابر یک دهم امتیاز انصراف دهنده خواهد بود.

منوهای بازی

- حین بازی در صفحه ی بازی نام و امتیاز حریف نمایش داده می شود. با کلیک بر روی نام بازیکن در هر قسمت بایستی لیست آمارگان بازی های او نمایش داده شود.
- قابلیت ارسال پیام به طرف مقابل را تعبیه کنید. پیام های با حریف، پس از اتمام بازی پاک می شوند.
- ده نفر با بالاترین امتیاز در صفحه ی اول بازی برای همه قابل مشاهده باشند (به ترتیب امتیاز و با ذکر امتیاز)
- هر بازیکن پس از ورود باید بتواند امتیاز و آمار خود را مشاهده کند.

در پایان نکاتی که برای طراحی خوب است به آن دقت کنید آورده می‌شود:

- از آنجایی که دیباگ کردن بازی در حضور امکانات شبکه کار دشواری است پیشنهاد می‌شود ابتدا بازی را پیاده‌سازی کنید و سپس امکانات شبکه را اضافه کنید.
- در طراحی بازی امکانات شبکه را در ذهن داشته باشید.
- اعتبارسنجی ایمیل به راحتی با استفاده از دریافت کد قابل پیاده‌سازی است.

در انتها، ضمن آرزوی موفقیت، امیدواریم تلاش شما در راستای این پروژه به یادگیری هر چه بهتر شما کمک کند.
همه تلاش ما یادگیری شما بوده است.
التماس دعا. تعطیلات قبل امتحانات خوشی را برای شما آرزومندیم. دی