



Front-End Development

Web Technologies - Module B

Submitted by:

- Hamed Kargarzadeh
- Amirabbas Ghasemi
- Mohammad Ghasemi Nasab

2.5 Hours



Junior national skills competition 2025

هرمزگان اسفند ۱۴۰۳

فهرست مطالب

1. مقدمه
2. توضیحات پروژه و وظایف
3. الزامات غیرعملکردی
4. ساختار بازی
 - a. صفحه بازی
 - b. منطق بازی
 - c. سیستم امتیازدهی
 - d. کار با API
5. صفحات
6. دستورالعمل‌های شرکت‌کننده
7. ارزیابی

۱. مقدمه



یک بازی تعاملی جدید به نام **Tower Builder** باید برای یک محیط مبتنی بر وب توسعه یابد. هدف این بازی، روی هم قرار دادن بلوک‌های متحرک برای ساخت بلندترین برج ممکن است. اگر یک بلوک روی بلوک قبلی قرار نگیرد، یک قلب از سه قلب بازیکن کم میشود تا قلب‌های کاربر تمام شود.

این پروژه بر تعامل مبتنی بر **JavaScript**، دستکاری **DOM**، انیمیشن‌ها و ارتباط با **API** برای ذخیره امتیازات بالا تمرکز دارد.

۲. توضیحات پروژه و وظایف

این آزمون از شما می‌خواهد که یک بازی تعاملی **Tower Builder** با ویژگی‌های زیر توسعه دهید:

در صفحه اول کاربر نام خودش رو وارد می‌کند و با کلیک بر روی دکمه **Start** بازی شروع میشود.

بلوک‌ها به صورت افقی حرکت می‌کنند و با کلیک یا فشار دادن کلید **Spacebar** متوقف می‌شوند.

بلوک‌ها باید روی یکدیگر قرار گیرند، اگر یک بلوک اشتباه پرتاپ شود از سه قلب بازیکن یک قلب کم میشود.

بازی زمانی به پایان می‌رسد که قلب‌های بازیکن تمام شود.

امتیاز هر بازی در انتها باید در یک **API** خارجی ذخیره و بازیابی شوند و در نهایت باید صفحه تاریخچه بازی‌ها از **API** دریافت و نمایش داده شود.

۳. الزامات غیرعملکردی

- شما می‌توانید از فریمورک های **Vue** و **React** استفاده کنید (استفاده از فریمورک اجباری نیست و می‌توانید پروژه را بدون استفاده از فریمورک هم پیاده سازی کنید).
- برنامه باید به درستی در مرورگر **Google Chrome** اجرا شود.

الزامات مربوط به دسترسی پذیری (Accessibility):

- دکمه‌ها باید متن قابل تشخیص داشته باشند.
- بازی باید هم با کلیک ماوس و هم با ورودی صفحه‌کلید قابل اجرا باشد.
- امتیازات بازی باید به صورت پویا (Dynamic) به روزرسانی شوند، بدون نیاز به رفرش صفحه.

۴. ساختار بازی

صفحه بازی (Game Screen)

- یک محیط بازی که در آن بلوک‌ها روی هم قرار می‌گیرند.
- یک نمایشگر امتیاز برای نمایش امتیاز فعلی بازیکن.
- یک دکمه شروع مجدد برای اجرای دوباره بازی.
- نمایش سه قلب کاربر
- پس از پایان بازی، یک جدول رتبه‌بندی نمایش داده شود که امتیازات برتر را از API دریافت می‌کند.

منطق بازی (Game Logic)

- بلوک‌ها به صورت خودکار حرکت را آغاز می‌کنند.
- بازیکن می‌تواند با کلیک ماوس یا فشردن کلید **Spacebar** بلوک را متوقف کند و اگر بلوک بر روی بلوک قبل قرار بگیرد ۱ امتیاز محاسبه می‌شود.
- هر ۲۰ امتیاز، سرعت حرکت بلوک‌ها ۲ برابر می‌شود و بازی سخت‌تر می‌شود.

سیستم امتیازدهی (Scoring System)

- بازیکنان با قراردادن هر بلوک یک امتیاز می‌گیرند.
- بازی زمانی تمام می‌شود که سه قلب بازیکن تمام شود.

کار با API

- **POST /scores** → ارسال امتیاز نهایی بازیکن با **Body** زیر.

```
{  
  "score": 22,  
  "name": "test"  
}
```

- **GET /scores** → دریافت اطلاعات جدول رتبه‌بندی.
- جدول رتبه‌بندی به صورت پویا بعد از هر بازی نمایش داده شود.
- یک فایل **Postman** در اختیار شما قرار می‌گیرد تا جزییات **API** را در اختیار داشته باشید.

۵. صفحات

در این مازول، رقابت‌کنندگان باید سه صفحه‌ی **شروع بازی**، **خود بازی و پایان بازی** را پیاده‌سازی کنند. برای هر صفحه، یک **Wireframe** ارائه شده است که باید طراحی مطابق با آن انجام شود. همچنین، مجموعه‌ای از **Assets** گرافیکی متنوع در اختیار شما قرار گرفته که می‌توانید از آن‌ها استفاده کنید. توجه داشته باشید که پیاده‌سازی زیبا و منظم امتیاز دارد.

صفحه شروع (Start Game)



Enter your name ...

Start

صفحه بازی (Game Screen)

20^{Floor}   





۶. دستورالعمل‌های شرکت‌کننده

- تمامی فایل‌های پروژه را در یک فایل فشرده با نام **XX_B.zip** بسته‌بندی کنید، که **XX** شماره میز کار شما است.
- فایل‌های سرور **API** را داخل فایل **ZIP** قرار ندهید.
- به کیفیت کد، ساختار، و قابلیت نگهداری (**Maintainability**) توجه داشته باشید.
- نسخه نهایی پروژه باید دارای کنترل نسخه مناسب باشد (شامل **Commit**های معنادار در **Git**).
- بر دسترس‌پذیری (**Accessibility**) و عملکرد (**Performance**) تمرکز کنید.
- میتوانید در این پروژه از فریمورک‌های **React** و **Vue** استفاده کنید.
- برای ارتباط با **API** کتابخانه **Axios** در اختیار شما قرار گرفته است.
- ارزیابی بر روی مرورگر **Google Chrome** انجام میشود.
- ارزیابی دسترس‌پذیری (**Accessibility**) با ابزار **Lighthouse** بر روی مرورگر کروم انجام میشود.
- فایل‌های مورد نیاز برای پیاده سازی پروژه در اختیار شما قرار میگیرد.
- چنانچه از فریمورک استفاده کرده باشید ارزیابی بر روی **Build** پروژه انجام میشود.

۷. ارزیابی

امتیاز	معیار
9.5	سازمان دهی و کیفیت کد
15	عملکرد بازی
10	پیاده سازی بصری
8	هماهنگ سازی با API
6	قابلیت های دسترس پذیری
48.5	مجموع: