



Quiz  
of  
Kings



دانشگاه صنعتی امیرکبیر  
(پلی تکنیک تهران)

## دانشکده مهندسی کامپیوتر

### گزارش کارآموزی

محل کارآموزی: شرکت کویز آو کینگز

نام استاد کارآموزی: مهندس جعفر عبدی

نام دانشجو: محمدرضا اخگری زیری

شماره دانشجویی: ۹۶۳۱۰۰۱

## تقدیر و شکر

بدینوسیله مراتب قدردانی و امتنان خود را خدمت،

جناب آقای امیرطاعی، سرپرست تیم توسعه دهنده سمت مشتری شرکت کوییز آو کینگز، بابت آموزش‌ها و زحمات بی‌دریغشان،

جناب آقای علی نظری، دبیر انجمن علمی دانشکده مهندسی کامپیوتر، بابت معرفی این دوره کارآموزی

و تمامی کارکنان و همراهان بنده در شرکت کوییز آو کینگز،

ابراز و از تمامی زحمات آنان تشکر می‌نمایم.

محمدرضا اعلی‌زیری

شهریور ۹۹

## چکیده

امروزه یکی از رسانه‌هایی که میزان قابل توجهی از افراد جامعه را متوجه خود ساخته، بازی‌های رایانه‌ای است. از نیمهٔ دوم قرن بیستم، همراه و همگام با تحولات و پیشرفت‌ها در صنایع الکترونیکی، بازی‌ها و اسباب بازی‌هایی جدید و متفاوت با گذشته، پا به عرصه وجود گذاشته که اصطلاحاً «بازی‌های رایانه‌ای» نامیده می‌شوند. صنعت ساخت بازی‌های ویدیویی و رایانه‌ای، در حال حاضر، به گسترده‌ترین و سودآورترین حرفه‌ها در صنعت سرگرمی تبدیل شده است.

در این گزارش، با توجه پس از نیازسنجی شرکت کوپیزا و کینگز، متوجه شدیم که موتور بازی‌سازی قبلی به نسخه متن‌باز<sup>۱</sup> تبدیل شده و دیگر از نسخه ۱۰ سیستم‌عامل آندروید<sup>۲</sup> پشتیبانی نمی‌کند. بنابراین مسائل، ما اقدام به تغییر پلتفرم بازی از موتور بازی‌سازی کرونا<sup>۳</sup>، به موتور یونیتی<sup>۴</sup> کردیم.

## واژه‌های کلیدی:

بازی‌های رایانه‌ای، موتور بازی‌سازی، موتور کرونا، موتور یونیتی

---

<sup>۱</sup> Open source

<sup>۲</sup> Android

<sup>۳</sup> Corona Engine

<sup>۴</sup> Unity

فصل اول مقدمه.....	۱
۱-۱- مقدمه.....	۲
۱-۲- برنامه نویسی سمت مشتری و سرویس دهنده.....	۳
۱-۲-۱- سرویس دهنده.....	۳
۱-۲-۲- وظایف سرویس دهنده ها.....	۴
۱-۲-۳- مشتری.....	۴
۱-۲-۴- وظایف مشتری ها.....	۴
۱-۳- نتیجه گیری.....	۵
فصل دوم معرفی محل کارآموزی.....	۶
۲-۱- تاریخچه.....	۷
۲-۲- معرفی کلی شرکت.....	۸
۲-۲-۱- بازی انفرادی.....	۸
۲-۲-۲- لیگ ستاره.....	۹
۲-۲-۳- لیگ دوستانه.....	۹
۲-۲-۴- بازی گروهی.....	۹
۲-۲-۵- رکورد.....	۹
۲-۳- افتخارات.....	۹
۲-۴- هدف.....	۱۰
۲-۵- جمع بندی.....	۱۰
فصل سوم فعالیت ها و تجارب کارآموزی.....	۱۱
۳-۱- انتخاب پلتفرم.....	۱۲
۳-۱-۱- موتور کرونا.....	۱۲
۳-۱-۱-۱- فواید.....	۱۲
۳-۱-۱-۲- معایب.....	۱۳
۳-۱-۲- موتور یونیتی.....	۱۴
۳-۲- تسلط بر کد.....	۱۵
۳-۲- مستند سازی.....	۱۶
۳-۲-۱- API.....	۱۶
۳-۲-۲- Rest.....	۱۷
۳-۲-۳- RestfulAPI.....	۱۷
۳-۳- زبان YAML.....	۱۷

۱۹	۳-۴- یادگیری یونیتی.....
۱۹	۳-۵- طراحی.....
۱۹	۳-۵-۱- معماری MVC.....
۲۰	۳-۵-۲- Model.....
۲۱	۳-۵-۳- View.....
۲۱	۳-۵-۴- Controller.....
۲۱	۳-۶- آکادمی.....
۲۱	۳-۷- متدولوژی اسکرام.....
۲۳	<b>فصل چهارم نتیجه گیری.....</b>
۲۴	۴-۱- جمع بندی.....
۲۴	۴-۲- کارهای آینده.....
۲۵	<b>منابع و مراجع.....</b>

صفحه

فهرست شکل‌ها

شکل ۱-۲- نمای از نسخه‌ی اولیه بازی کوییزآو کینگز.....	۷
شکل ۲-۲- نمای از نسخه فعلی بازی کوییزآو کینگز.....	۸
شکل ۱-۳- نمای یک API.....	۱۷
شکل ۲-۳- معماری یک RestAPI.....	۱۸
شکل ۳-۳- نمایش معماری MVC.....	۲۰

## فصل اول

### مقدمه

## ۱-۱- مقدمه

به باور روان‌شناسان و جامعه‌شناسان، بازی که عمیقاً با زندگی پیوند خورده است، در روحیه و تکوین شخصیت افراد، اثر عمیقی بر جای می‌گذارد و باعث می‌شود که فرد از طریق آشنایی با الگوها و هنجارهای جامعه خویش و درونی‌سازی ارزش‌های آن در راستای اجتماعی شدن، گام بردارد.

پدیده‌ی بازی بر حسب علل، زمینه‌ها و شرایط، طی زمان‌ها و مکان‌های مختلف، به لحاظ ساختار، محتوا و کارکرد (اثر)، شکل‌های گوناگونی به خود گرفته است.

اگرچه تاریخچه‌ی بازی‌های رایانه‌ای به دهه‌ی ۱۹۷۰م، برمی‌گردد؛ اما در مدت کوتاهی پس از پیدایش، نه تنها تحول عمیق و اساسی در نظام تفریحات و سرگرمی کودکان به‌وجود آوردند، بلکه در حال حاضر، گستره‌ی بازی‌های رایانه‌ای به عنوان یک وسیله‌ی گذران اوقات فراغت، به عرصه‌ی بزرگ‌سالان نیز کشیده شده است.

بازی‌های الکترونیکی در عین آن که خود دارای شبکه، سایت‌های مخصوص و گروه‌های گفت و گو و صحبت (چت) در اینترنت هستند، به صورت مجموعه‌ای صوتی و تصویری (برنامه‌های ویدیویی و تلویزیونی) به دیگر رسانه‌ها منتقل شده‌اند.

در این بین، رقابتی که بین شرکت‌ها و تولیدکنندگان مختلف در تصاحب هرچه بیشتر بازار فروش به وجود آمده، آن‌ها را بر آن داشته که با ارائه‌ی بازی‌های هرچه جالب‌تر و مهیج‌تر، سود بیشتری را به خود اختصاص دهند.

شرکت دانش‌پروان امیرکبیر نیز به عنوان یکی از فعالان این زمینه، بازی تحت شبکه‌ای به نام کویزآو کینگز<sup>۵</sup> ارائه داده است. هدف این بازی افزایش سطح علمی بازیکنان، علاوه بر سرگرمی آن‌ها است.

---

<sup>۵</sup> Quiz of Kings



## ۲-۱- برنامه نویسی سمت مشتری و سرویس دهنده

در هر برنامه تحت شبکه‌ای، ما مفاهیمی به نام مشتری<sup>۶</sup> و سرویس دهنده<sup>۷</sup> را شنیده‌ایم، لازم است تا در ابتدای کار توضیحی از این مفاهیم بدهیم تا در ادامه بتوانیم درباره‌ی آن‌ها صحبت کنیم.

واژه‌های سرویس دهنده و مشتری، در دهه‌ی ۸۰ میلادی همزمان با عرضه تجهیزات شبکه به بازار، مطرح شد. تصور افراد از مشتری و سرویس دهنده صرفاً به تجهیزات سخت‌افزاری کامپیوترها مربوط می‌شد. سرویس دهنده به کامپیوتری اطلاق می‌گردید که در حقیقت مانند مرکز ثقل ارائه خدمات در شبکه بود و مشتری نیز کامپیوتری بود که این امکانات را مورد استفاده قرار می‌داد. این تعاریف اگرچه درست بودند اما ناقص بودند. یعنی این تعریف که مشتری صرفاً یک کامپیوتر فیزیکی و سرور کامپیوتر فیزیکی متصل به آن، که وظیفه ارائه سرویس را برعهده دارد، باشد، تعریف کاملی نیست. در ادامه به بررسی این دو مفهوم به طور کامل می‌پردازیم.

### ۱-۲-۱- سرویس دهنده

سرویس دهنده، یک کامپیوتر همواره روشن است که در تمامی مواقع در دسترس است و به کامپیوترها و تجهیزات دیجیتالی که به شبکه متصل هستند، سرویس می‌دهد. و به همین دلیل به آن، سرویس دهنده می‌گویند. اساساً راه‌اندازی سرویس دهنده‌ها برای ارائه خدمت به تجهیزات متصل در شبکه انجام می‌شود تا بتوانند به درخواست‌های مشتری‌ها پاسخ دهند. به عنوان نمونه در زمان اتصال شما به یک وب‌سایت و استفاده از خدمات آن، سرویس دهنده آن سایت در حال خدمت‌رسانی به شما است. سرویس دهنده‌ها خدمات متنوعی در اختیار مشتری‌ها قرار می‌دهند که از آن جمله می‌توان به انتقال فایل<sup>۸</sup>، سرویس وب، سرویس بازی<sup>۹</sup> و به سرویس دهنده‌هایی مانند آن‌ها اشاره کرد.

---

<sup>۶</sup> Client

<sup>۷</sup> Server

<sup>۸</sup> Server

<sup>۹</sup> Game Server

## ۲-۲-۱- وظایف سرویس دهنده‌ها

از آنجایی که سرویس دهنده‌ها وظیفه ارائه خدمت به مشتری‌ها را دارند، بدیهی است که به لحاظ سخت‌افزاری، کامپیوترهای قدرتمندی محسوب می‌شوند. اشتراک منابع برای استفاده مشتری‌ها در شبکه توسط سرویس دهنده صورت می‌گیرد. اختصاص حافظه، حافظه اصلی، پردازشگر به مشتری‌های شبکه، میزان استفاده از اینترنت از جمله منابعی است که توسط سرویس دهنده به سرویس گیرندگان در شبکه اختصاص می‌یابد. سرویس دهنده‌ها تعیین کننده میزان استفاده از منابع شبکه توسط مشتری‌ها هستند. یک سرویس دهنده می‌بایست به درخواست مشتری‌ها و هر سرویس گیرنده‌ای در شبکه پاسخ دهد. زمانی که دو سرویس گیرنده که کاربر محسوب می‌شوند قصد تبادل پیغام و اطلاعات دارند، سرویس دهنده ابتدا داده را ذخیره و سپس به سرویس گیرنده دیگر ارسال می‌کند.

## ۳-۲-۱- مشتری

در مقابل سرویس دهنده، تجهیزات و کامپیوترهایی که از سرویس دهنده، خدمات دریافت می‌کنند مشتری نامیده می‌شوند. تمامی تجهیزات، کامپیوترها و یا نرم‌افزارهای متصل به سرویس دهنده که به سرویس دهنده به آن‌ها خدمت ارائه می‌دهد، سرویس گیرنده محسوب می‌گردند. برای مثال، کامپیوترهای موجود در شبکه که به چاپگر اتصال دارند، از آنجایی که از چاپگر شبکه، درخواست خدمات چاپ اطلاعات دارند، سرویس گیرنده شبکه محسوب می‌گردند.

## ۴-۲-۱- وظایف مشتری‌ها

مشتری‌ها وظیفه اصلی که برعهده دارند، ارسال درخواست‌های خود به سرویس دهنده است. باتوجه به نوع نیاز هر مشتری، درخواست‌ها توسط آن از طریق برنامه‌های کاربردی به سرویس دهنده ارسال می‌گردد. می‌توانیم در نظر بگیریم که مشتری از سرویس دهنده تقاضای کار<sup>۱۰</sup> می‌نماید.

---

<sup>10</sup> Task

ارسال و به اشتراک گذاشتن اطلاعات در شبکه از دیگر وظایف مهم مشتری‌ها محسوب می‌گردد. نمونه آن زمانی است که کاربران در یک سرویس‌دهنده گفت‌وگو<sup>۱۱</sup>، اطلاعات خود را به سمت سرویس‌دهنده ارسال می‌نمایند و سپس ارسال اطلاعات از کاربر به کاربر دیگر از سوی سرویس‌دهنده انجام می‌شود.

### ۳-۱- نتیجه‌گیری

حال که به تفاوت این دو مفهوم در یک برنامه پی بردیم، لازم به ذکر است که ما در دوره‌ی کارآموزی، توسعه‌ی این برنامه در سمت مشتری را آغاز کردیم. اما به علت مشکلاتی که در ادامه ذکر خواهیم کرد، اقدام به تغییر پلتفرم و زبان برنامه‌نویسی سمت مشتری کردیم.

برای این اقدام لازم به تسلط کافی از کد قبلی و محیط بازی داشتیم و همچنین باید به تمام API های مورد نیاز برای بازنویسی برنامه آشنا بودیم. که شرح این مسائل در فصل سوم به صورت کاملاً واضح شرح داده شده است.

---

<sup>11</sup> Chat

## فصل دوم

### معرفی محل کارآموزی

## ۱-۲- تاریخچه

در سال‌های ۱۳۹۴ تا ۱۳۹۶ جمعی از دانشجویان دانشگاه امیرکبیر، با ایده‌ای مبنی بر پیاده‌سازی یک بازی موبایل همراه با موضوع اطلاعات عمومی دور هم جمع شدند. این جمع نسخه‌ی اولیه‌ای از بازی را ایجاد کردند و بر روی فروشگاه‌های برنامه‌ها از جمله بازار و گوگل استور<sup>۱</sup> قرار دادند.

بازی با استقبال بسیار خوبی از سمت کاربران مواجه شد و همین باعث شد تا شرکت دانش‌پروران امیرکبیر (کوییزآو کینگز) در سال ۱۳۹۶ با مالکیت خصوصی تأسیس شود. در شکل ۱-۲ نمایی از نسخه اولیه این بازی نمایش داده می‌شود.



شکل ۱-۲- نمایی از نسخه‌ی اولیه بازی کوییزآو کینگز

کوییز آو کینگز، یک بازی برخط، در سبک کلمات و دانستنی ساخت شرکت دانش پروران امیرکبیر است. این بازی در لیگ بازی‌های رایانه‌ای ایران که توسط بنیاد ملی بازیهای رایانه‌ای در سال ۱۳۹۶ و ۱۳۹۷ برگزار شد، پرمخاطب‌ترین بازی در مسابقات انتخاب شد و جایزه ویژه این مسابقات به نفرات برتر این بازی تعلق گرفت.

<sup>۱</sup> Google Store



شکل ۲-۲- نمایشی از نسخه فعلی بازی کوئیز آو کینگز

## ۲-۲- معرفی کلی شرکت

کوئیز آو کینگز بزرگ‌ترین استارت‌آپ حوزه‌ی بازی‌های ویدئویی در کشور با بیش از ۶۰ نیروی فعال است. میانگین سنی این نیروهای حرفه‌ای ۲۶ سال بوده و همواره سعی شده است تا کوئیز آو کینگز روحیه‌ی جوان خود را حفظ کند.

محور اصلی بازی کوئیز آو کینگز بر روی رقابت برخط استوار است و رقابت در این بازی بر سر دانش و اطلاعات عمومی بازیکن‌ها صورت می‌پذیرد. این رقابت در پنج حالت انفرادی، لیگ ستاره، لیگ دوستانه، بازی گروهی و رکورد با ۱۹ موضوع مختلف قابل انجام است. این تغییرات به مرور زمان در بازی ایجاد شد که در شکل ۲-۲ آنها را مشاهده می‌کنیم. بخش‌های مختلف بازی در ادامه شرح داده می‌شود:

### ۲-۲-۱- بازی انفرادی

در هر مسابقه انفرادی کوئیز آو کینگز دو بازیکن به عنوان رقیب یکدیگر باید به ۱۸ سؤال چهارگزینه‌ای پاسخ دهند. این سوالات در قالب ۶ موضوع مختلف پرسیده می‌شود و هر بازیکن به صورت یک در میان موضوع سوالات را انتخاب می‌کند.

### ۲-۲-۲- لیگ ستاره

در هر رقابت لیگ ستاره، ده بازیکن باید به ۱۲ سؤال چهارگزینه‌ای در موضوعات مختلف پاسخ دهند و در پایان بازیکنان بر اساس تعداد پاسخ صحیح و زمان پاسخگویی، رتبه‌بندی می‌شوند.

### ۲-۲-۳- لیگ دوستانه

لیگ دوستانه نیز مانند لیگ ستاره است ولی با این تفاوت که بازیکن حریفان خود را از بین دوستان خود انتخاب می‌کند.

### ۲-۲-۴- بازی گروهی

هر گروه شامل تعدادی بازیکن است. در بازی گروهی افراد یک گروه در قالب یک تیم به رقابت با گروهی دیگر می‌پردازند. تعداد سوالات در این حالت از بازی ۱۰ سؤال از تمامی موضوعهای بازی است.

### ۲-۲-۵- رکورد

در این بخش هر بازیکن به صورت شخصی به جواب دادن سوالات موضوعات مختلف به صورت متوالی پرداخته و فقط سه بار امکان پاسخ اشتباه دادن وجود دارد. بعد از پایان بازی، رکورد او در جدول ثبت شده و برترین رکوردهای روزانه و هفتگی در جدول رده‌بندی مشخص می‌شوند.

### ۲-۳- افتخارات

- برترین بازی سال ۹۵ و ۹۶ (کلمات و دانستنی) از دید داوران و از دید مردم و برترین بازی سال ۹۷ (کلمات و دانستنی) از دید مردم در جشنواره وب و موبایل ایران.
- بهترین بازی مردمی سال ۹۵ در ششمین جشنواره بازی‌های رایانه‌ای.
- بازی سال ۹۷ در بخش بهترین به‌روزرسانی از دید مردم در هشتمین جشنواره بازی‌های رایانه‌ای.

## ۴-۲- هدف

محتوای بازی ها به سرعت در حال تغییر است و با کاربردهای گرافیکی و به دلیل جاذبه ای که در آنها نهفته است، روز به روز واقعی تر به نظر رسیده، نظر افراد بیشتری را به خود جلب می نمایند. و متأسفانه موضوعی که در این آشفتگی بازار مورد توجه نیست، همان جنبه های اخلاقی و تربیتی کاربران این بازی ها است. از این رو شرکت کوییزآو کینگز با ایجاد این بازی سعی دربالا بردن معلومات عمومی بازیکنان در کنار سرگرمی دارد.

## ۵-۲- جمع بندی

من در این شرکت در دوره کارآموزی وظیفه توسعه بخش مشتری این برنامه بودم. دوره کارآموزی بنده به دلیل بیماری کرونا، دورکاری بود و محیط کار به صورت کار در منزل بود. برای ساعت کاری سیستم، وظیفه محور پیاده شده بود، که در شرح کار و متولوژی اسکران به آن خواهیم پرداخت.



## فصل سوم

## فعالیت‌ها و تجارب کارآموزی

بنیانگذاران شرکت کویزآو کینگز، برای پیاده‌سازی سریع نسخه ابتدایی خود از موتور بازی‌سازی کرونا که از زبان لوا پشتیبانی می‌کرد، استفاده کرده بودند. اما این موتور بازی‌سازی علی‌رغم فوایدی که داشت، مشکلاتی نیز به همراه داشت که تصمیم به تغییر پلتفرم گرفته شد، در طول کارآموزی اقدام به تغییر پلتفرم کردیم.

### ۱-۳- انتخاب پلتفرم

موتور بازی‌سازی کرونا با مشکلاتی که به همراه داشت، باعث شد تا شرکت، تصمیم به تغییر پلتفرم بگیرد. اولین چالش ما، انتخاب موتور جایگزین بود. با بررسی موتورهای موجود و مقایسه آن‌ها، موتور یونیتی را انتخاب کردیم. برای درک بهتر این وضعیت، در ادامه به مقایسه این دو موتور بازی‌سازی می‌پردازیم، برای این منظور، ابتدا نیاز داریم تا فواید و معایب موتور بازی‌سازی کرونا را بیان کنیم و سپس علت انتخاب موتور یونیتی را به تفصیل شرح دهیم.

#### ۱-۱-۳- موتور کرونا

توسعه بازی‌های تلفن همراه به طور خاص و انحصاری بسیار محبوب شده‌است. اکثر برنامه‌های کاربردی<sup>۱</sup> و بازی‌های جدید بر اساس موتور کرونا ساخته شده‌اند.

#### ۱-۱-۱-۳- فواید

در این بخش قصد داریم، با مزایای استفاده از موتور کرونا در بازی‌سازی آشنا شویم. این مزایا عبارتند از:

- پشتیبانی از چندین پلتفرم<sup>۲</sup>: یکی از مزایای اصلی موتور کرونا این است که برنامه‌نویس، فقط کافیست یک مرتبه کد خود را بنویسد و برای اجرای آن بر روی سیستم‌عامل‌های مختلف نیازی به تغییر و اصلاح کد نیست.

---

<sup>۱</sup> Applications

<sup>۲</sup> Multi platform

- **فریم بر ثانیه<sup>۱</sup>:** موتور کرونا به دلیل سرعت بالا و حداقل تاخیر در مقایسه با سایر SDK<sup>۲</sup>های موجود، در بازار امروز بسیار شناخته شده است. با استفاده از این SDK، شما می‌توانید به راحتی تمام ابعاد صفحه‌ای که نیاز دارید را تنظیم کنید. عموماً با استفاده از SDKهایی مانند PhoneGap و ImpactJS مقدار FPS بسیار کم است و توسعه بازی‌های تلفن همراه، یک تجربه عقب‌مانده محسوب می‌شود ولی با موتور کرونا این طور نیست.
- **شبیه‌ساز:** شبیه‌ساز استفاده شده در موتور کرونا کاملاً مشابه تلفن همراه شما است. در صورت استفاده از این SDK، شبیه‌ساز، پروژه شما را به عنوان یک فایل ذخیره می‌کند و فرایند توسعه را بسیار سریع‌تر می‌سازد.
- **زبان برنامه‌نویسی لوا:** موتور کرونا از زبان برنامه‌نویسی لوا استفاده می‌کند و یکی از ویژگی‌های این زبان یادگیری آسان و استفاده راحت آن است.

## ۲-۱-۱-۳- معایب

- همه‌ی موتورهای بازی‌سازی در کنار مزایا، معایبی نیز دارند. این موتور نیز دارای معایبی می‌باشد که ادامه راه توسعه برنامه را با مشکل مواجه می‌کند. در ادامه به شرح این معایب می‌پردازیم:
- **قیمت:** رایج ترین مسئله‌ای که کاربران موتور کرونا با آن مواجه هستند، قیمت بالای آن است که باید برای استفاده از آن پرداخت شود.
  - **عدم پشتیبانی از افزونه<sup>۳</sup>های غیربومی:** در صورتی که شما در پروژه خود نیاز به استفاده از یک افزونه دیگر داشته باشید، هیچ راهی برای ادغام افزونه‌های جدید نخواهید داشت. به این ترتیب ممکن است استفاده از موتور کرونا برای تمام برنامه‌های کاربردی شما امکان پذیر نباشد.

<sup>۱</sup> Frame per Second (FPS)

<sup>۲</sup> Software development kit

<sup>۳</sup> Plugin

- **عدم پشتیبانی نسخه‌های جدید سیستم‌عامل:** سیستم‌عامل‌های تلفن همراه به سرعت در حال بروزرسانی هستند، اما بروزرسانی کرونا، به مراتب آهسته‌تر است و همین باعث شده است تا اجرای بازی بر روی نسخه ۱۰ سیستم‌عامل اندروید دارای مشکل باشد.
- **متن باز شدن موتور:** موتور بازی‌سازی کرونا به تازگی اعلام کرده است که قصد دارد به صورت متن باز به کاربران ارائه شود. این مسأله، سرنوشت این موتور را نامعلوم کرده و باعث می‌شود تا آینده شرکت با خطر روبه رو شود، زیرا مشخص نیست که در آینده، آیا توسعه‌دهندگان، از این مسأله استقبال خوبی خواهند کرد یا خیر.
- **زبان برنامه‌نویسی لوا:** زبان برنامه‌نویسی لوا علاوه بر فایده، معایبی نیز دارد، که می‌توان به عدم پشتیبانی از شی گرایشی اشاره کرد که در برنامه‌های توسعه یافته و عظیم، شی گرایشی رویکرد مناسبی برای برنامه‌نویسی است.
- **انجمن<sup>۱</sup> ضعیف:** این موتور، انجمن ضعیفی دارد و برنامه‌نویسان شرکت برای رفع مشکلات و جست‌وجو با مشکل مواجه می‌شدند.

## ۲-۱-۳- موتور یونیتی

با مقایسه مزایا و معایب موتور بازی‌سازی کرونا، متوجه شدیم که باید پلتفرمی انتخاب شود که با همراه داشتن مزایای موتور کرونا، مشکلات آن موتور را نیز رفع کند. و در بین گزینه‌های موجود، موتور بازی‌سازی یونیتی انتخاب ما بود. دلایل انتخاب این موتور را در ادامه ذکر می‌کنیم:

- **پشتیبانی از چند پلتفرم:** یکی از دلایل موفقیت یونیتی، اجرا و پیاده‌سازی آن بر روی پلتفرم‌های مختلف است. برنامه‌ای که بر روی یک پلتفرم نوشته فقط با چند کلیک برای استفاده در یک پلتفرم دیگر آماده خواهد شد. یونیتی هم‌اکنون پلتفرم‌های تلفن همراه، کامپیوتر، وب و کنسول‌های بازی را پشتیبانی می‌کند. برای نمونه:
  - بر روی تلفن همراه، سیستم‌های اندروید، بلک بری، ویندوز و iOS
  - بر روی کامپیوترها، سیستم‌عامل‌های لینوکس، ویندوز و مک

<sup>1</sup> Community

## - برای اهداف وب، Flash، Web Player و Google Client

پشتیبانی می‌شوند.

- **فریم بر ثانیه:** این موتور نیز مانند کرونا تمامی ابعاد صفحه را پشتیبانی می‌کند.
- **شبیه‌سازی:** یونیتی شبیه‌سازی برای تمامی پلتفرم‌ها دارد و استفاده از آن بسیار آسان است.
- **بروز بودن:** یونیتی اکثراً افزونه‌ها را دارد و به سرعت به‌روزرسانی می‌شود که مشکلات موتور کرونا را برطرف می‌کند.
- **زبان برنامه‌نویسی سی‌شارپ:** زبان برنامه‌نویسی استفاده شده در این موتور، سی‌شارپ است که شباهت نحوی<sup>۱</sup> به زبان جاوا دارد. همچنین این زبان از ساختار شی‌گرایی پشتیبانی می‌کند.
- **انجمن قوی:** یونیتی یکی از بهترین موتورهای موجود است که امکانات و ابزارهای دو بعدی و سه بعدی زیادی را در اختیار کاربران قرار می‌دهد. در حال حاضر ۳۴٪ از بهترین بازی‌های تلفن همراه و ۹۰٪ از بازی‌های واقعیت مجازی توسط یونیتی ساخته شده‌اند.

## ۲-۳- تسلط بر کد

قدم بعدی یا به عبارتی مهم‌ترین قدم پس از انتخاب پلتفرم، تسلط بر کد سابق است. تسلط بر کد سابق و همچنین محیط بازی برای پیاده‌سازی نسخه جدید لازم است، زیرا هم باعث میشود تا با راه‌کارهای توسعه‌دهندگان سابق برای حل مشکل آشنا شویم و هم از قسمت‌های پنهان بازی اطلاع یابیم. برای این منظور در ابتدای امر شروع به یادگیری زبان برنامه‌نویسی لوا کردیم. این زبان، زبانی اسکریپتی است و آشنایی بنده با زبان پایتون، کمک شایانی برای یادگیری زبان کرد.

برای تمرین این زبان، اقدام به پیاده‌سازی صفحه‌ای از بازی گروهی کردیم. این وظیفه جامع و کامل بود و تقریباً تمام قسمت‌های view را پوشش می‌داد.

پس از تسلط نسبی بر زبان لوا، به باقی مراحل پرداختیم.

---

<sup>۱</sup> Syntactically

## ۲-۳- مستند سازی

همانطور که در قسمت مقدمه گزارش کارآموزی ذکر کردیم، بازی تحت شبکه و به صورت مشتری/سرویس‌دهنده است. مشکلی که وجود داشت، نبودن مستندی برای کد سمت مشتری و همچنین برای درخواست‌ها<sup>۱</sup> و پاسخ‌ها<sup>۲</sup> بود. چالش بعدی ما پیدا کردن ابزار مستندسازی بود.

برای مستندسازی کد سمت مشتری ابتدا نیاز به نوشتن نیازمندی‌ها بود، برای این نیازمندی‌ها در ابتدا برای هر قسمت از بازی داستانی<sup>۳</sup> نوشتیم و در هر قسمت تمامی درخواست‌ها و همچنین پیش‌شرایط و اثرات بعدی آن قسمت را نوشتیم. تا در مرحله بعدی برای طراحی بازی به دید کلی‌ای از بازی رسیده باشیم.

در قسمت درخواست‌ها، برای اتصال به سرویس‌دهنده از RestfulApi استفاده شده بود. که در ابتدا به توضیح آن می‌پردازیم.

### ۱-۲-۳- API

API<sup>۴</sup> که با رابط برنامه‌نویسی کاربردی ترجمه می‌شود یک مجموعه از قواعد و مکانیزم‌ها است که از طریق آن برنامه‌های کاربردی و یا بخش‌های مختلف یک برنامه با همدیگر ارتباط برقرار می‌کنند. نام خود این مکانیزم بیانگر همه چیز است. منظور از رابط چیزی است که دو شیء یا دو موجودیت مختلف را به همدیگر ربط می‌دهد. API می‌تواند داده‌هایی که شما برای برنامه کاربردی خود نیاز دارید را از طریق یک فرمت مناسب به خروجی بفرستد و یا آن را برگشت دهد. فرمت JSON و XML از این دست فرمت‌ها هستند. در این مطلب ما قصد داریم روی JSON تمرکز بکنیم. شکل ۱-۳ نمای یک API را نشان می‌دهد.

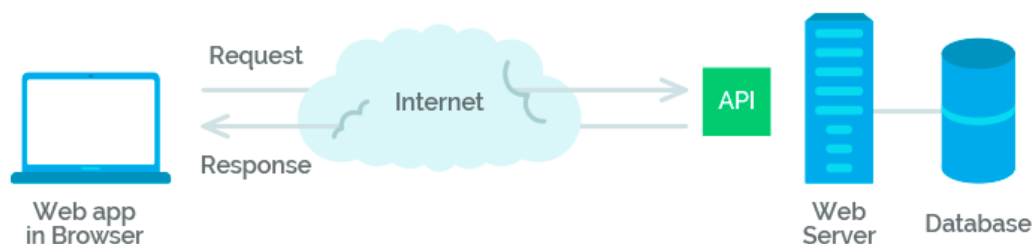
---

<sup>۱</sup> Requests

<sup>۲</sup> Responses

<sup>۳</sup> User story

<sup>۴</sup> Application Programming Interface



شکل ۱-۳- نمای یک API

## Rest - ۳-۲-۲

این کلمه اختصاری برای representational state transfer است. تعریف REST را به صورت ساده می‌توان اینطور بیان کرد: نمایش اطلاعات برای کاربران از راهی که خوانایی بالایی داشته باشد. یکی از مفاهیم اصلی که باید در ارتباط با REST بدانید این است که REST یک پروتکل یا استاندارد نیست، این تنها یک راه‌حل و یا یک سبک معماری برای نوشتن API ها است.

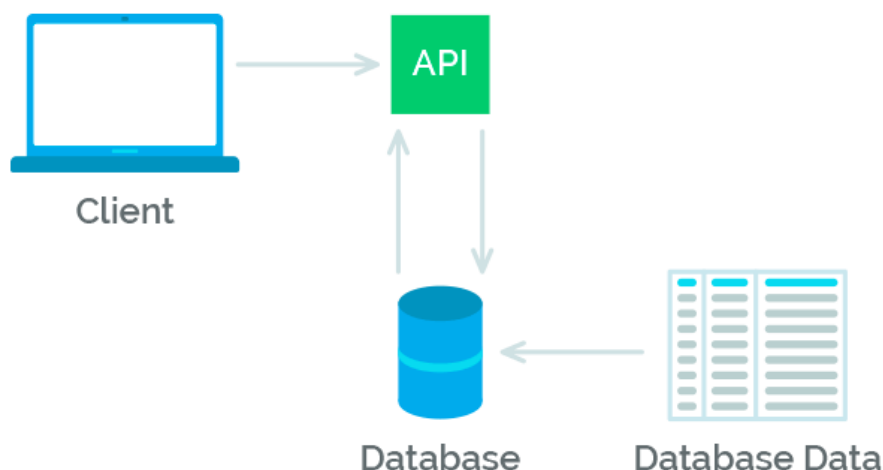
## RestfulAPI - ۳-۲-۳

حال که فهمیدیم REST چیست، درک کردن RESTful API بسیار ساده‌تر است. همانطور که گفته شد REST یک روش معماری و چیدمان است و حال RESTful را می‌توان مفسری برای REST دانست. برای مثال اگر شما یک سرور دارید و قسمت Back-End یک REST API دارد، اگر یک کاربر از قسمت Client-Side یک درخواست برای استفاده از API بکند، کاربر شما Restful خواهد بود. شکل ۲-۳ معماری یک RestAPI را نشان می‌دهد.

## YAML - ۳-۳

پس از فهمیدن مفهوم RestfulAPI، ما نیاز به مستندسازی درخواست‌ها و پاسخ‌ها داشتیم. چالش بعدی ما پیدا کردن ابزاری ساده و قابل استفاده به سرعت بود. ابزاری که ما برای مستندسازی انتخاب کردیم، زبان برنامه‌نویسی YAML بود.

## REST API Design



شکل ۲-۳- معماری یک RestAPI

YAML یک زبان برای طبقه‌بندی داده قابل خواندن-برای-انسان است. معمولاً برای فایل‌های پیکربندی استفاده می‌شود، اما می‌تواند در کاربردهای دیگری هم مورد استفاده قرار گیرد. YAML بسیاری از برنامه‌های ارتباطات مشابهی را به عنوان XML هدف قرار می‌دهد، اما یک نحوی کمینه هم دارد که سازگاری با SGML تجزیه و تحلیل می‌کند. به دو روش نوشته می‌شود اولی سبک زبان پایتون که با پیش‌آمدی در متن تودرتویی را پشتیبانی می‌کند و دومی روش فشرده‌تری است که مانند JSON برای لیست‌ها از [] و برای مپ‌ها از {} استفاده می‌کند که YAML 1.2 را زیر مجموعه‌ی JSON قرار می‌دهد.

انواع داده‌های شخصی‌سازی شده در YAML مجاز هستند، اما YAML به صورت طبیعی اعداد (مانند رشته‌ها، اعداد صحیح و شناور)، لیست‌ها و آرایه‌ها (که رمزنگاری شده یا مپ یا دیکشنری شناخته می‌شود) را رمز گذاری می‌کند.

پسوند نام پرونده برای YAML، از سال ۲۰۰۶ تا به اکنون، yml است.



## ۴-۳- یادگیری یونیتی

برای یادگیری یونیتی، با دو بخش مهم مواجه بودیم، یک یادگیری زبان سی‌شارپ و دیگری یادگیری یونیتی. به علت تسلط بر زبان جاوا، در یادگیری زبان سی‌شارپ مشکلی نداشتیم. یونیتی شامل سه قسمت می‌شود، یونیتی UI، 2D و 3D. برای این بازی، ما احتیاج به یادگیری Unity UI داشتیم. برای یادگیری این مفاهیم از کورس‌های آنلاین یودمی استفاده کردیم.

## ۵-۳- طراحی

به علت تفاوت ساختار زبان‌های برنامه‌نویسی دو موتور و همچنین کمبود وقت در طراحی نسخه اولیه کوییزآو کینگز، مجبور به طراحی دوباره کلاس‌ها شدیم. معماری انتخابی ما، معماری <sup>۱</sup>MVC بود، البته یونیتی معماری <sup>۲</sup>ECS را نیز برای بازی‌ها پیشنهاد می‌کند، که با توجه به شباهت بازی به برنامه‌کاربردی تلفن همراه از آن معماری استفاده نکردیم و به همین جهت به توضیح آن نمی‌پردازیم. در طراحی کلاس‌ها سعی شد از مفاهیم شی‌گرایی و الگوهای طراحی <sup>۳</sup> استفاده شود.

### ۱-۵-۳- معماری MVC

MVC نوعی روش معماری نرم‌افزار است که در توسعه‌ی وب اپلیکیشن‌ها بسیار پرکاربرد است و ورود آن به صنعت توسعه‌ی نرم‌افزار به دهه‌ی ۱۹۷۰ بازمی‌گردد. امروزه فریمورک‌های مطرحی که در توسعه‌ی نرم‌افزارهای کوچک و بزرگ مورد استفاده قرار می‌گیرند مبتنی بر این معماری‌اند.

---

<sup>۱</sup> Model View Controller

<sup>۲</sup> Entity component system

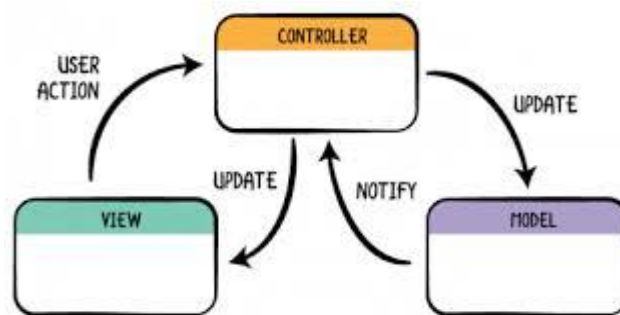
<sup>۳</sup> Design patterns

Trygve Reenskaug یک دانشمند علوم کامپیوتری نروژی است که به عنوان خالق اصلی معماری MVC شناخته می‌شود به طوری که وی در سال ۱۹۷۹ به منظور توسعه‌ی GUI با استفاده از زبان Smalltalk از این معماری استفاده کرد و در سال‌های بعد این معماری به مرور در میان توسعه‌دهندگان مختلف رواج پیدا کرد تا جایی که الگوهای دیگری من جمله HMVC، MVA، MVP و MVVM بر این اساس شکل گرفتند. شکل ۳-۳ نمایی از یک معماری mvc را نشان می‌دهد.

### Model - ۳-۵-۲

Model را به نوعی می‌توان به منزله‌ی مغز اپلیکیشن در نظر گرفت به طوری که اصطلاحاً Business Logic یا به عبارتی «آنچه اپلیکیشن به خاطرش توسعه یافته است» در این لایه طرح‌ریزی می‌شود. مسلماً نیاز به توضیح نیست که مثلاً در یک وب اپلیکیشن بخشی از منطق نرم‌افزار مرتبط با ارتباط با پایگاه داده<sup>۱</sup> به منظور انجام عملیات CRUD<sup>۲</sup> است که وظایفی از این دست در مدل عملی می‌گردند.

کنترلر که در ادامه بیشتر با مفهوم‌اش آشنا خواهیم شد، همواره ارتباط تنگاتنگی با مدل دارد است به طوری که می‌توان گفت ارتباط مدل با ویو از طریق کنترلر امکان‌پذیر خواهد بود.



شکل ۳-۳- نمایش معماری MVC

1  
Database

2 سرواژه‌ی CRUD برگرفته از کلمات Create، Read، Update و Delete است که به ترتیب به منظور «ثبت»، «فراخوانی»، «به‌روزرسانی» و «حذف» داده‌ها از دیتابیس مورد استفاده قرار می‌گیرند.

### View - ۳-۵-۳

View، همان‌طور که از نام آن مشخص است، این وظیفه را دارا است تا داده‌هایی که در مدل ساخته و پرداخته شده را در معرض دید کاربران وب‌اپلیکیشن قرار دهد و به عبارتی می‌توان گفت که همان User Interface یا به اختصار UI است.

### Controller - ۳-۵-۴

Controller در این معماری سه‌لایه نقش واسط را دارا است به طوری که درخواست‌ها را از بخش View گرفته و در اختیار Model قرار می‌دهد و پس از آنکه مدل پردازش‌هایش را روی درخواست ورودی انجام داد، پاسخ را مجدد در اختیار کنترلر قرار داده و کنترلر هم پاسخ نهایی را در اختیار View می‌گذارد تا در معرض دید کاربران قرار دهد.

### ۳-۶ - آکادمی

در طی کارآموزی جلساتی با عنوان آکادمی نیز برگزار می‌شد، هدف این جلسات افزایش سطح دانش فنی و تیمی بود. از مواردی که در جلسات آکادمی بررسی شد، می‌توان به موارد زیر اشاره کرد:

- الگو طراحی singleton و command
- چگونه یک merge request خوب بدهیم
- Clean code
- UML

جلسات به صورت ارائه‌ی شخصی در مورد موضوعی و گفت‌وگو و بحث درباره‌ی آن بحث بود.

### ۳-۷ - متدولوژی اسکرام

برای چلبک شدن و مدیریت چلبک پروژه‌ها روش‌های مختلفی وجود دارد، برخی از روش‌های مدیریت پروژه‌های چلبک به نسبت سایر آنها معروف تر هستند و بیشتر مورد توجه و استفاده قرار گرفته‌اند.

آشنایی کامل و بی نقص و همزمان با همه متدهای مدیریت چابک پروژه مانند اسکرام، کانبان و XP امری ضروری نیست اما داشتن آشنایی با آنها در اجرای چابکی به شما این امکان را می‌دهد تا در انتخاب روش‌های متناسب با هر نیاز و شرایط دست بازی داشته باشید.

اسکرام پر استفاده ترین روش در مدیریت چابک پروژه‌ها است که سه رکن اصلی دارد:

۱. شفافیت<sup>۱</sup>، ایجاد دید روشن برای افرادی که در به وجود آمدن نتیجه مسئول هستند.

۲. بازبینی<sup>۲</sup>، بررسی مستمر اینکه در حال انجام چه کاری با توجه به اهداف هستید.

۳. تطبیق<sup>۳</sup>، تغییر فرآیندهای تیم به منظور کاهش خطاها.

تیم‌های اسکرام در بسته‌های کوتاه مدت تکراری به نام اسپرینت کار می‌کنند. در عمل اسکرام از تعدادی از فعالیت‌ها و رسم‌های تکرارشونده در هر اسپرینت پیروی می‌کند. با این کار در واقع اسکرام گارد ریل‌های محافظی ایجاد میکنند تا افراد به طور موثرتری کار کنند.

کوئیز آو کینگز نیز برای حل مشکل ساعت کاری و حضور در شرکت، بر اساس اسکرام، ساعت کاری وظیفه‌محور را بیان کرد که در این مورد، ساعت کاری برای حضور مهم نیست و در هر اسپرینت تعدادی تسک با توجه به زمانی که باید شخص در آن اسپرینت کار کند مشخص می‌شود و شخص می‌تواند این تسک‌ها را در زمان کمتری انجام دهد.

....

---

<sup>1</sup> Transparency

<sup>2</sup> Inspection

<sup>3</sup> Adaptation

## فصل چهارم

## نتیجه گیری

## ۴-۱- جمع بندی

هدف این کارآموزی شروع تغییر پلتفرم یک برنامه توسعه یافته و عظیم بود. این کار سختی‌های بسیاری داشت و همچنین خطرات شکست بسیار. اما تیم توسعه دهنده کوییز این خطر را به جان خریدند تا به محصولی بهتر دست پیدا کنند. مواجه شدن با موتور بازی سازی جدید و بدون دانش قبلی کاری پر از خطر بود که تیم ما توانست با تلاش بی‌وقفه گام بزرگی را در این زمینه بردارد.

در هنگامی که این گزارش نوشته می‌شود، نسخه اولیه بازی به زبان انگلیسی در کشور فیلیپین در حال انتشار و اجرا است.

## ۴-۲- کارهای آینده

کار تغییر پلتفرم به علت حجم بزرگ بازی هنوز به صورت کامل انجام نشده است، قصد ما ادامه این کار تا رسیدن به نسخه کاملی از بازی است. و در این راه، روش‌های متفاوتی را برای بهتر شدن بازدهی بازی به کار خواهیم برد.

## منابع و مراجع