



دانشگاه صنعتی شریف  
دانشکده مهندسی کامپیوتر

# معماری نرم افزاری سامانه‌ی آجاره

پروژه درس تحلیل و طراحی سیستم‌ها

مدرس: دکتر طارمی‌راد

گروه ۲: کیارش بنی‌هاشم      سید محمدصادق مهدوی      بنیامین قاسمی‌نیا      آریو لطفی  
۹۶۱۰۹۹۶۳      ۹۶۱۱۰۲۹۴      ۹۶۱۰۹۸۳۳      ۹۶۱۱۰۲۸۳

## مقدمه

در این مستند به شرح معماری نرم‌افزاری سامانه‌ی آجاره می‌پردازیم. سامانه‌ی آجاره یک سامانه‌ی اجاره‌ی کالا است که در آن کاربران می‌توانند با ایجاد حساب کاربری، کالاهای مختلفی مانند کنسول‌های بازی را اجاره کرده و نظرات خود را در مورد کالاهای مختلف ثبت کنند.

در این مستند از روش 4 + 1 view برای بررسی معماری نرم‌افزار استفاده می‌کنیم. با توجه به این که نرم‌افزار در مرحله‌ی اولیه از پیچیدگی سخت‌افزاری کمی برخوردار است و نیز پیچیدگی‌های مربوط به استفاده‌ی همروند از سیستم، توسط Framework‌هایی مانند Django قابل حل است، مطابق پیشنهاد؟؟ دید Physical و Process در این مستند بررسی نمی‌شوند زیرا در این پروژه مطرح نیستند و تنها دیدهای Logical، Development و Scenario مورد بررسی قرار می‌گیرند.

در طراحی این معماری از سبک Model View Controller یا MVC استفاده شده است که با توجه به ماهیت مبنای وب پروژه، با استفاده از Framework‌هایی مانند Django، باعث تسریع توسعه‌ی نرم‌افزار و سادگی طراحی می‌شود.

## نگاه منطقی<sup>۱</sup>

سیستم آجاره شامل ۶ کلاس اصلی است که هر کدام به صورت زیر هستند:

- **User** نمایانگر هر کاربر در سیستم است. فیلدهای اصلی آن نام کاربری و آدرس ایمیل و رمز عبور هستند.
- **Profile** هر کاربر یک پروفایل دارد که مابقی اطلاعات اضافی آن از جمله عکس پروفایل و آدرس و غیره داخل آن هستند.
- **Product** نمایانگر محصول در سیستم است. اجزای اصلی هر محصول عکس، قیمت و توضیحات آن هستند.
- **Order** با هر سفارش کاربر برای محصولات یک مدل سفارش ساخته می‌شود که تاریخ سفارش و وضعیت سفارش در آن نگه‌داری می‌شوند.
- **Comment** هر کاربر می‌تواند روی پست‌ها کامنت‌گذاری کند که در این مدل ذخیره می‌شوند.
- **Rating** هر کاربر می‌تواند به محصولات امتیاز بدهد که در این مدل ذخیره می‌شود.

دیاگرام یوامال<sup>۲</sup> مدل‌های سیستم در شکل ۱ آمده‌است.

## نگاه توسعه<sup>۳</sup>

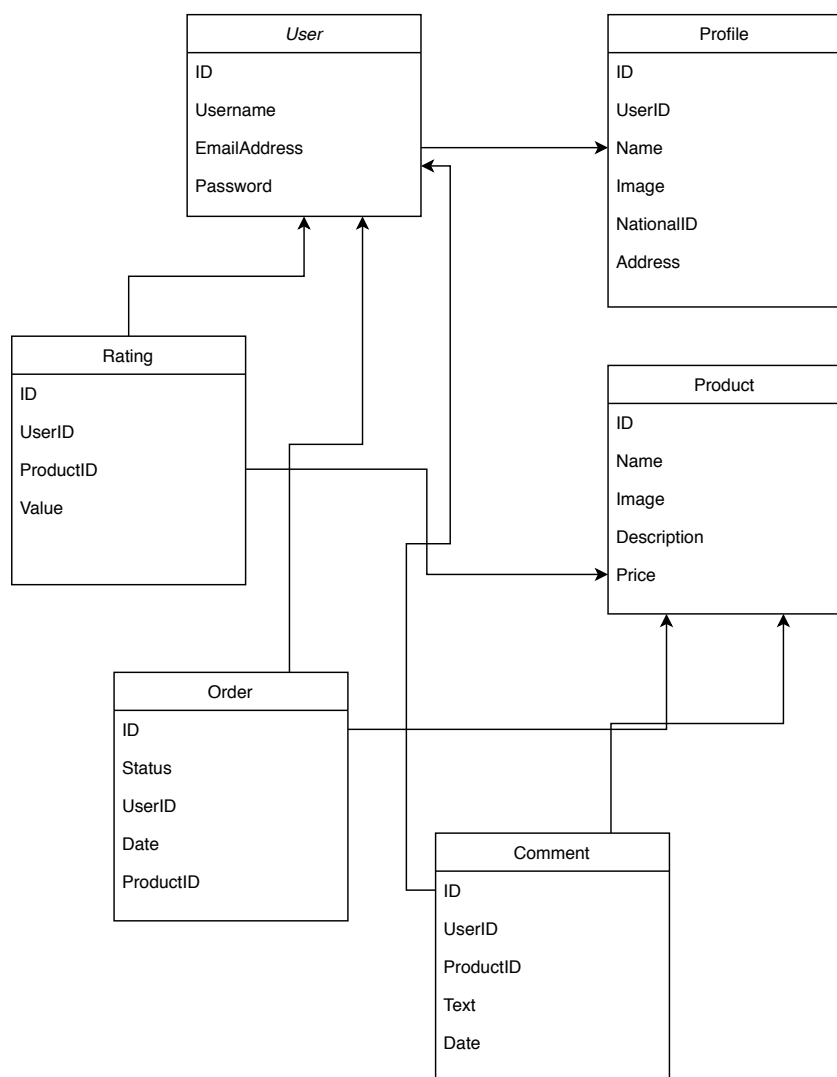
نگاه توسعه تمرکزش روی سازمان دادن به نرم‌افزار و محیط توسعه آن است. در سیستم‌های بزرگ معمولاً بین ۴ تا ۶ لایه توسعه وجود دارد. برای سیستم آجاره ۴ لایه در نظر می‌گیریم:

---

<sup>۱</sup> Logical View

<sup>۲</sup> UML

<sup>۳</sup> Development View



شکل ۱: دیاگرام مدل‌های سیستم

- **سخت‌افزار و سیستم‌عامل** این لایه مربوط به مدیریت سیستم‌عامل و سخت‌افزار سرورهای موجود برای سیستم آجاره هست. مدیریت میزان منابع و نسخه و کانفیگ سیستم‌عامل مربوط به این لایه است.
  - **زیرساخت** این لایه مربوط به زیرساخت‌های لازم برای پروژه از جمله پایگاه‌داده<sup>۱</sup> است که انتخاب و نصب و نگهداری آن مربوط به این لایه است.
  - **کاربران و محصولات** این لایه مربوط به ساخت حساب کاربری برای کاربران و ایجاد محصولات است. کلاس‌ها و عملکردهای مربوط به ثبت‌نام و ورود کاربران همچنین ایجاد محصولات در این لایه قرار دارد.
  - **اجاره و پس‌دادن** این لایه مربوط به عملیات اجاره دادن و پس‌گرفتن محصولات اجاره داده‌شده است. این لایه‌که به اصطلاح Domain Specific است در آخرین لایه قرار دارد.
- توجه کنید که در این سیستم محتویات هر لایه فقط به لایه‌های قبل از آن وابسته‌اند. همچنین مسئولیت هر لایه با یک نفر است (لایه اول تا چهارم به ترتیب صادق، کیارش، بنیامین و آریو).

## نگاه سناریو<sup>۲</sup>

در این بخش به دید سناریوی پروژه می‌پردازیم. به این منظور، دو سناریو که کارکرد سیستم را نشان دهند در نظر می‌گیریم. یک نکته‌ی مهم در مورد سناریوها این است که اگرچه طراحی ما مبنی بر استایل MVC است، در بعضی از بخش‌ها به علت سادگی مفهومی کنترلر، از اشاره به آن صرف‌نظر کرده‌ایم و مستقیماً View را به Model وصل کرده‌ایم.

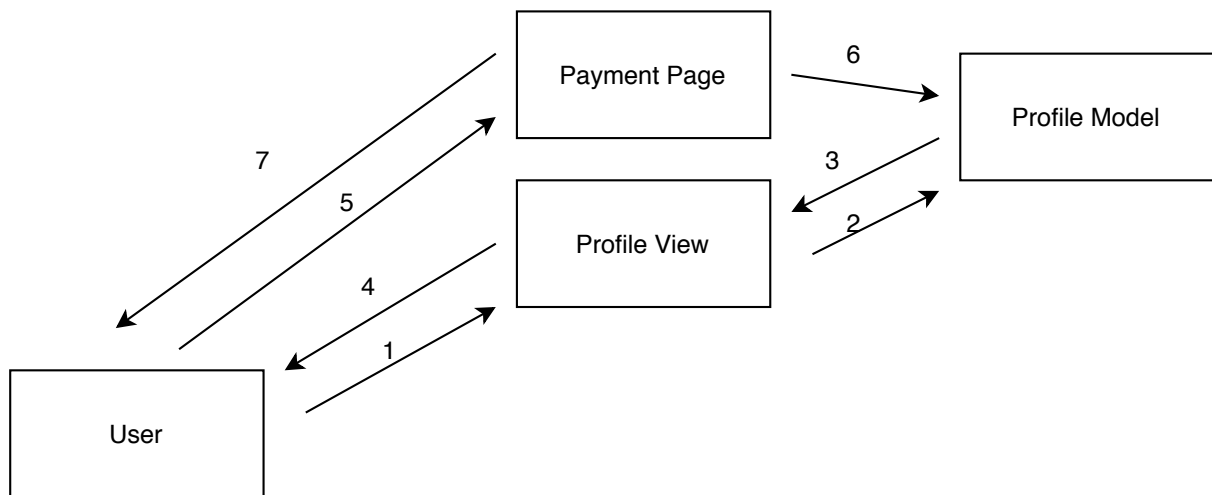
### سناریوی پرداخت

در این سناریو، کاربر به صفحه‌ی کاربری خود مراجعه کرده و با اطلاع از میزان اعتبارش، تصمیم به افزایش اعتبار می‌کند. شکل این سناریو در ۲ آمده است. مراحل سناریو به شرح زیر هستند.

۱. ابتدا کاربر درخواست دسترسی به صفحه‌ی پروفایل خود را می‌دهد که توسط Profile View، گرفته می‌شود.
۲. Profile View با دسترسی به مدل کاربر، درخواست کسب اطلاعات مورد نیاز را می‌دهد.
۳. Profile model اطلاعات مورد نیاز را به Profile view می‌دهد.
۴. Profile view، اطلاعات کاربر را به وی نشان می‌دهد.
۵. کاربر با دسترسی به صفحه‌ی پرداخت، اقدام به افزایش اعتبار خود می‌کند.
۶. در صورت موفقیت پرداخت، اعتبار کاربر افزایش می‌یابد.
۷. نتیجه‌ی تراکنش به اطلاع کاربر می‌رسد.

<sup>۱</sup> Database

<sup>۲</sup> Scenario View



شکل ۲: سناریوی پرداخت

## سناریوی اجاره

در این سناریو، کاربر با دسترسی به صفحه‌ی جست‌وجو، ابتدا کالای مورد نظر خود را می‌یابد و سپس با بررسی مشخصات آن، اقدام به اجاره می‌کند. شکل این سناریو در ۳ آمده است.

۱. ابتدا کاربر با دسترسی به صفحه‌ی جست‌وجو، لیست کالاهای مورد نظرش را می‌خواهد.

۲. صفحه‌ی جست‌وجو لیست کالاهای مناسب را با درخواست دادن به product model به دست می‌آورد.

۳. لیست مورد نظر به کاربر داده می‌شود.

۴. کاربر درخواست مشاهده‌ی مشخصات کالای مورد نظر را می‌دهد.

۵. اطلاعات کالای مورد نظر تهیه از Product model گرفته می‌شود.

۶. اطلاعات کالا به کاربر نشان داده می‌شود.

۷. کاربر اقدام به ثبت سفارش می‌کند.

۸. درخواست ثبت سفارش به Order controller داده می‌شود.

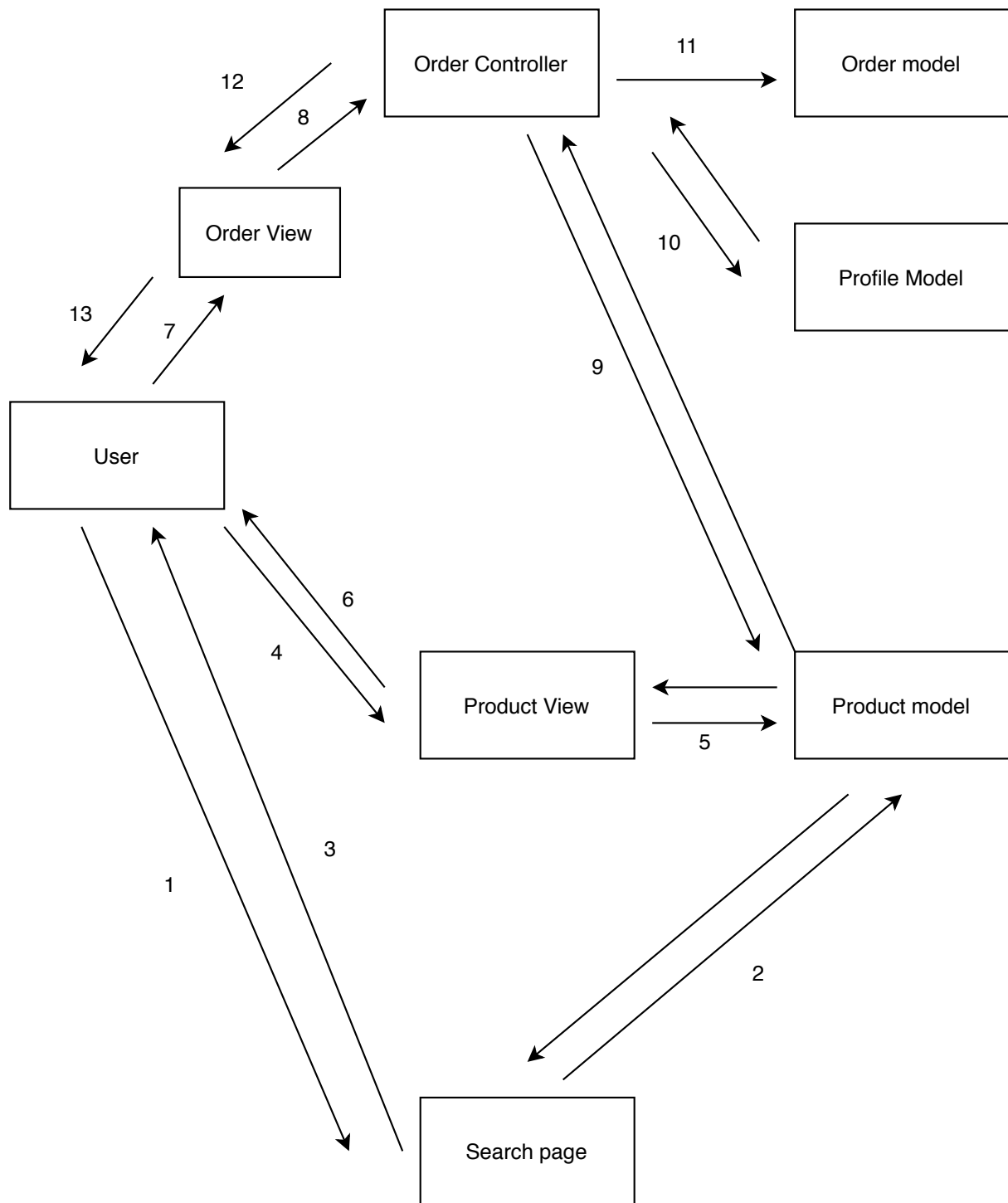
۹. اطلاعات کالای مورد نظر از قبیل قابل اجاره بودن و قیمت تهیه می‌شود.

۱۰. اطلاعات مالی کاربر با درخواست دادن به Profile model تهیه می‌شود.

۱۱. در صورتی که سفارش قابل انجام باشد، سفارش مناسب در Order model ثبت می‌شود.

۱۲. نتیجه‌ی سفارش به Order view داده می‌شود.

۱۳. نتیجه‌ی سفارش به کاربر نشان داده می‌شود.



شکل ۳: سناریوی اجاره