



دانشگاه اصفهان  
دانشکده مهندسی کامپیوتر



# AMAZON ANALYTICS

S I M P L E , E A S Y A N D E L E G A N T

مهدی حق‌وردی  
سید محمدحسین هاشمی

استاد راهنما: دکتر محمد رضا شعر باف  
دستیار استاد: آقای رضا پورمحمدی

مهر ۱۴۰۲

# فهرست مطالب

۱	ساختار شکست کار	۱
۱	۱.۱ وب سرویس جمع آوری داده	۱.۱
۲	۱.۱.۱ Stock	۲.۱.۱
۲	۱.۱.۱ Site	۳.۱.۱
۲	۱.۱.۱ Shipment	۴.۱.۱
۲	۱.۱.۱ ساختار شکست کار	۲.۱
۶	۲.۱ وب سرویس تحلیل داده و ارائه گزارش	۱.۲.۱
۶	۱.۲.۱ Stock	۲.۲.۱
۷	۱.۲.۱ Site	۳.۲.۱
۷	۱.۲.۱ Shipment	۴.۲.۱
۷	۱.۲.۱ ساختار شکست کار	۳.۱
۱۰	۳.۱ وب اپلیکیشن	۱.۳.۱
۱۰	۱.۳.۱ ساختار شکست کار	

## فصل ۱

# ساختار شکست کار

در این فصل، ساختار شکست کار پروژه‌ی Amazon Analytics نوشته شده است. این ساختار بر اساس فصل ششم اسلایدهای استاد نوشته شده‌اند. در این فصل سعی شده است ساختار شکست کار، برای قسمت‌های کد نویسی پروژه نوشته بشود، و به کارهایی که در گام اول و دوم پروژه مربوط هستند پرداخته نشده است.

موارد تحویل دادنی که در فاز دوم تحت عنوان موارد تحویل دادنی نوشته شده‌اند، نقطه سطح بالای شروع نوشتن ساختار شرکت کار برای پروژه Amazon Analytics هستند.

سپس این نقاط سطح بالا بر اساس ۳ قسمت اصلی فرض شده برای شرکت آمازون (Site، Stock، Shipment) شکسته می‌شوند، و برای هر قسمت بسته‌های کاری کوچک‌تری نوشته می‌شوند. یک سری از بسته‌های کاری نرم‌افزاری نبوده و نیاز به ماژول‌های سخت‌افزاری و نصب فیزیکی هستند، که از بسته‌ی کاری آنها صرفاً یک اسم آورده شده و نقطه‌ی عطف آنها نصب و تست ماژول‌های سخت‌افزاری خواهد بود.

## ۱.۱ وب سرویس جمع‌آوری داده

وظیفه‌ی این وب سرویس ۱. جمع‌آوری داده، ۲. ذخیره‌ی داده و ۳. ارائه داده<sup>۱</sup> است. این سرویس تمامی داده‌های مورد نیاز را از قسمت‌های مختلف و اصلی شرکت جمع‌آوری کرده، آنها به صورت بهینه ذخیره کرده و امکانات قدرتمند برای بازیابی اطلاعات را در اختیار استفاده‌کنندگان سرویس، قرار می‌دهد. در ادامه راجع به قسمت‌های مختلف شرکت و ارتباط آنها با وب سرویس توضیح داده خواهد شد و در نهایت بسته شکست کاری این قسمت نشان داده می‌شود.

---

<sup>۱</sup> ارائه داده یعنی از ابزارات قوی برای جستجو و برگرداندن داده از دیتابیس استفاده کند، برای مثال GraphQL

### ۱.۱.۱ Stock

قسمت Stock تمامی کارهای انبارداری شرکت را انجام می‌دهد. مهم‌ترین کارهایی که انجام می‌دهد عبارتند از ۱. وقتی کسی محصولات را می‌بیند، از موجود بودن یا نبودن آن اطمینان حاصل کند، ۲. اگر سفارشی ثبت شد، پردازش آن را انجام دهد، ۳. محصول را به حلقه‌ی بعدی زنجیره پردازش و تحویل کالا بدهد.

قسمت اعظمی از کارهایی که این قسمت انجام می‌دهد، به صورت سیستمی و کامپیوتری انجام می‌شوند؛ و نکته‌ی مهم اینست که برای رصد کردن عملکرد این قسمت باید قسمت‌های نرم‌افزاری برای monitoring به کدهای این قسمت اضافه شوند.

### ۲.۱.۱ Site

بیشتر اطلاعاتی که ما در قسمت Site از شرکت جمع‌آوری می‌کنیم، اطلاعات بازخوردی از کاربران و مشتریان شرکت هستند. در کنار این اطلاعات ما اطلاعات سیستمی و نرم‌افزاری که بیشتر به صورت آماری هستند هم برای ارزیابی عملکرد این قسمت جمع‌آوری می‌کنیم.

موارد بازخوردی شامل ۱. گرفتن بازخورد در مورد صفحات توضیح ویژگی محصولات، ۲. بازخورد راجع به فروشنده، ۳. بازخورد در مورد محصول خریداری شده (بعد از خرید) و ۴. بازخورد در مورد طراحی و تجربه‌ی کاربری سایت هستند. و اطلاعات آماری که شامل ۱. تست سرعت بارگذاری سایت هست.

### ۳.۱.۱ Shipment

قسمت اعظم اطلاعاتی که از قسمت Shipment نیاز به دریافت داریم، اطلاعاتی از قبیل سری‌های زمانی<sup>۲</sup>، اطلاعات جغرافیایی<sup>۳</sup> و همچنین وسایل نقلیه استفاده شده برای تحویل مرسوله هستند. این اطلاعات، چون ماهیت فیزیکی دارند، باید به صورت فیزیکی هم جمع‌آوری شوند و پس از دریافت اطلاعات از سنسورها به صورت مناسب و efficient<sup>۴</sup> ذخیره و بازیابی شوند.

### ۴.۱.۱ ساختار شکست کار

در این قسمت ساختار شکست کار را نوشته‌ایم. دقت کنید که این جعبه‌ی بزرگ خود شامل ساختارهای شکست کار کوچک‌تر است که هر کدام نقطه‌ی عطف خودش را دارد.

<sup>۲</sup> یعنی این کالا در این زمان از قسمت stock خارج شد و در این زمان وارد کامیون حمل شد و این زمان به مشتری تحویل داده شد.

<sup>۳</sup> یعنی این کالا از این انبار واقع در فلان شهر تحویل داده شد و از طریق این مسیر و آن مسیر به مقصد رسید.

<sup>۴</sup> برای مثال برای سری‌های زمانی، دیتابیس‌های مخصوصی وجود دارد.

## ۱ وب سرویس جمع‌آوری داده

### ۱.۱ طراحی کلی وب سرویس

- ۱.۱.۱ بررسی و استخراج نیازمندی‌ها
- ۲.۱.۱ نوشتن نیازمندی‌ها
- ۳.۱.۱ استخراج کلی endpointها<sup>a</sup>
- ۴.۱.۱ انتخاب نام برای endpointها

← نقطه عطف: نوشته شدن OpenAPI Specification از روی موارد و تصمیم‌های بالا

### ۲.۱ پیاده‌سازی وب سرویس

- ۱.۲.۱ بررسی معماری نرم‌افزاری وب سرویس
- ۲.۲.۱ تعیین و طراحی معماری
- ۳.۲.۱ انتخاب زبان برنامه‌نویسی
- ۴.۲.۱ انتخاب دیتابیس و toolchain عه دیتابیس<sup>b</sup>
- ۵.۲.۱ پیاده‌سازی
- ۶.۲.۱ تست سرویس
- ۷.۲.۱ داکرایز کردن سرویس
- ۸.۲.۱ دیپلوی کردن سرویس

← نقطه عطف: تست و دیپلوی شدن سرویس

## قسمت ۱.۱.۱ Stock

### ۳.۱ نوشتن کدهای monitoring برای موجود بودن یا نبودن محصول

- ۱.۳.۱ پیدا کردن توابع و کلاس‌هایی که اطلاعات لازم برای سرویس را تولید یا مصرف می‌کنند
- ۲.۳.۱ اضافه کردن کد، برای ارسال اطلاعات به وب سرویس
- ۳.۳.۱ تست کردن ارسال اطلاعات
- ۴.۳.۱ تست کردن دریافت صحیح اطلاعات

← نقطه عطف: پاس شدن تمامی تست‌های لازم

### ۴.۱ Integrate کردن توابعی برای ردیابی و گرفتن و ارسال کردن اطلاعات در قسمت‌های ثبت و پردازش سفارش

- ۱.۴.۱ پیدا کردن توابع و کلاس‌هایی که اطلاعات لازم برای سرویس را تولید یا مصرف می‌کنند
- ۲.۴.۱ اضافه کردن کد، برای ارسال اطلاعات به وب سرویس

- ۳.۴.۱ تست کردن ارسال اطلاعات
- ۴.۴.۱ تست کردن دریافت صحیح اطلاعات
- ← **نقطه عطف:** پاس شدن تمامی تست‌های لازم

## ۵.۱ نوشتن توابعی برای ارسال اطلاعات ردیابی محصول در خروج از Stock

- پیدا کردن توابع و کلاس‌هایی که اطلاعات لازم برای سرویس را تولید یا مصرف می‌کنند
- ۱.۵.۱ اضافه کردن کد، برای ارسال اطلاعات به وب سرویس
- ۲.۵.۱ تست کردن ارسال اطلاعات
- ۳.۵.۱ تست کردن دریافت صحیح اطلاعات
- ← **نقطه عطف:** پاس شدن تمامی تست‌های لازم

## قسمت ۲.۱.۱ Site

## ۶.۱ تحلیل و طراحی صفحات گرفتن بازخورد

- ۱.۶.۱ لیست کردن تمامی مواردی که باید از آنها بازخورد گرفته شود
- ۲.۶.۱ طوفان فکری، تحقیق در مورد صفحات
- ۳.۶.۱ تحلیل و طراحی
- ۴.۶.۱ گرفتن بازخورد
- ۵.۶.۱ نهایی کردن طراحی در فیگما
- ← **نقطه عطف:** تحویل طراحی‌های فیگما

## ۷.۱ نوشتن کدهای front-end برای صفحات طراحی شده

- ۱.۷.۱ طراحی صفحات از روی طرح‌های فیگما
- ۲.۷.۱ نوشتن کدها
- ۳.۷.۱ انجام integration test ها
- ۴.۷.۱ [داکرایز کردن آنها]<sup>c</sup>
- ← **نقطه عطف:** تست و تحویل کدها برای به تیم front-end آمازون

## ۸.۱ نوشتن کد و استفاده از Prometheus instrumentation library ها

- ۱.۸.۱ تحقیق برای پیدا کردن ابزار مناسب
- ۲.۸.۱ انتخاب و یادگیری ابزار مناسب
- ۳.۸.۱ integrate کردن آن
- ۴.۸.۱ تست کردن

۵.۸.۱ نوشتن کد برای ارسال موارد مورد نیاز به وب سرویس  
← **نقطه عطف:** مستقر کردن و تست دریافت اطلاعات آماری

### قسمت ۳.۱.۱ Shipment

#### ۹.۱ تحقیق در مورد انواع سنسورها برای داده‌های سری‌های زمانی

۱.۹.۱ تحقیق در مورد سنسورها  
۲.۹.۱ بررسی سنسورها برای نصب  
← **نقطه عطف:** انتخاب سنسور مناسب

#### ۱۰.۱ تست کردن اطلاعات دریافتی

۱.۱۰.۱ تست کردن سنسورها در شرایط آزمایشگاهی  
۲.۱۰.۱ تست کردن سنسورها در شرایط طبیعی  
۳.۱۰.۱ بررسی نتایج تست‌ها  
۴.۱۰.۱ نوشتن مازول‌های سخت‌افزاری برای ارسال اطلاعات به وب سرویس  
← **نقطه عطف:** verify شدن اطلاعات دریافتی

#### ۱۱.۱ تحقیق در مورد انواع سنسورها برای داده‌های جغرافیایی

۱.۱۱.۱ تحقیق در مورد سنسورها  
۲.۱۱.۱ بررسی سنسورها برای نصب  
← **نقطه عطف:** انتخاب سنسور مناسب

#### ۱۲.۱ تست کردن اطلاعات دریافتی

۱.۱۲.۱ تست کردن سنسورها در شرایط آزمایشگاهی  
۲.۱۲.۱ تست کردن سنسورها در شرایط طبیعی  
۳.۱۲.۱ بررسی نتایج تست‌ها  
۴.۱۲.۱ نوشتن مازول‌های سخت‌افزاری برای ارسال اطلاعات به وب سرویس  
← **نقطه عطف:** verify شدن اطلاعات دریافتی

#### ۱۳.۱ انتخاب دیتابیس مناسب برای داده‌های سری زمانی

۱.۱۳.۱ تحقیق در مورد دیتابیس‌های مخصوص سری زمانی موجود  
۲.۱۳.۱ بررسی نیازمندی‌های Amazon Analytics  
۳.۱۳.۱ بررسی عمیق و تطبیق دادن قابلیت‌ها و نیازمندی‌ها

← **نقطه عطف:** انتخاب شدن یک دیتابیس مناسب

## ۱۴.۱ انتخاب دیتابیس مناسب برای داده‌های جغرافیایی

۱.۱۴.۱ تحقیق در مورد دیتابیس‌های مخصوص سری زمانی موجود

۲.۱۴.۱ بررسی نیازمندی‌های Amazon Analytics

۳.۱۴.۱ بررسی عمیق و تطبیق دادن قابلیت‌ها و نیازمندی‌ها

← **نقطه عطف:** انتخاب شدن یک دیتابیس مناسب

<sup>a</sup> یعنی بگویم برای گرفتن داده‌های فلان قسمت stock به یک endpoint با این پارامترها نیاز داریم.

<sup>b</sup> یعنی انتخاب ORM یا ODM، و همچنین سیستم نگهداری و مدیریت migrationها.

<sup>c</sup> اینکه این مورد در یک قلاب نوشته شده، یعنی ممکن است الزامی نباشد.

## ۲.۱ وب سرویس تحلیل داده و ارائه گزارش

وظیفه‌ی این وب سرویس، تحلیل داده‌ها و ارائه‌ی تحلیل‌هاست. ارائه شکل‌های مختلفی از گزارشات از قبیل نمودارها و گزارش‌های time series و ... هم به عهده‌ی این وب سرویس است. داده‌های مورد نیاز این سرویس، به صورت کاملاً ساختاربندی شده و تمیز از ۱.۱ تامین می‌شوند. یعنی این سرویس هیچ وابستگی به قسمت‌های دریافت اطلاعات<sup>۵</sup> ندارد و فقط به endpointهای این وب سرویس وابستگی دارد.<sup>۶</sup> در ادامه راجع به قسمت‌های مختلف شرکت و ارتباط آنها با وب سرویس توضیح داده خواهد شد و در نهایت بسته شکست کاری این قسمت نشان داده می‌شود.

### ۱.۲.۱ Stock

تحلیل داده‌های دریافتی از ۱.۱ در قسمت Stock شامل تحلیل‌ها در مورد:

۱. بررسی مدت‌دار موجود بودن کالاهای خریداری شده<sup>۷</sup>

۲. بررسی زمان پردازش یک محصول از لحظه‌ی ثبت شدن توسط مشتری تا اتمام پردازش در قسمت انبار<sup>۸</sup>

۳. بررسی موقعیت مکانی سفارش دهنده‌ها و موقعیت مکانی انبار انتخاب شده برای پردازش محصول<sup>۹</sup>

<sup>۵</sup> آنهایی که از سنسورها و سیستم‌های آمازون جمع می‌شوند.

<sup>۶</sup> یعنی سرویس‌ها کاملاً decouple شده‌اند.

<sup>۷</sup> این بررسی برای تحلیل عملکرد انبارداری آمازون استفاده می‌شود.

<sup>۸</sup> این بررسی برای تحلیل عملکرد پردازنده‌های کالا در انبارها استفاده می‌شود.

<sup>۹</sup> این بررسی برای تحلیل اینکه آیا نزدیک‌ترین یا خلوت‌ترین انبار برای پردازش انتخاب می‌شود یا خیر استفاده می‌شود.



نرم افزارهای تحلیل گر این قسمت نیازمند دریافت داده ها به صورت لحظه ای یا با دوره های کوتاه هستند تا بتوانند تحلیل های دقیق زمانی و جغرافیایی ارائه دهند.

### ۲.۲.۱ Site

تحلیل هایی که این قسمت ارائه می دهد بسیار وابسته به میزان اطلاعاتی که از ۱.۱ دریافت می کند دارد<sup>۱۰</sup>. بسته های کاری این قسمت نسبتاً سبک هستند و چیزی که برای آنها حائز اهمیت است، میزان اطلاعات دریافتی آنهاست.

### ۳.۲.۱ Shipment

تحلیل هایی که این قسمت ارائه می دهد، تلفیقی از داده های زمانی و جغرافیایی هستند که داده هاشان از سنسورهای سخت افزاری و آماری که در سیستم های آمازون ثبت می شوند (یعنی ثبت سفارش ها و...) و از طریق ۱.۱ تامین می شوند. تحلیل ها شامل

۱. زمان تحویل کالا از انبار تا رسیدن به وسیله ای انتقال دهنده ی بزرگ<sup>۱۱</sup>
۲. زمان تحویل کالا از وسیله ای انتقال دهنده ی بزرگ به وسیله ای تحویل دهنده به مشتری
۳. زمان تحویل کالا به مشتری

### ۴.۲.۱ ساختار شکست کار

#### ۲ وب سرویس تحلیل داده و ارائه گزارش

##### ۱.۲ طراحی کلی وب سرویس

بررسی و استخراج نیازمندی ها

۱.۱.۲ نوشتن نیازمندی ها

۲.۱.۲ استخراج کلی endpoint ها<sup>a</sup>

۳.۱.۲ انتخاب نام برای endpoint ها

← نقطه عطف: نوشته شدن OpenAPI Specification از روی موارد و تصمیم های بالا

##### ۲.۲ پیاده سازی وب سرویس

۱.۲.۲ بررسی معماری نرم افزاری وب سرویس

۲.۲.۲ تعیین و طراحی معماری

<sup>۱۰</sup> چون اطلاعات وب سرویس جمع آوری اطلاعات، در این قسمت، از بازخوردهای مخاطبین بدست می آید  
<sup>۱۱</sup> برای مثال تریلی یا کشتی برای جابجایی اولیه کالا

- ۳.۲.۲ انتخاب زبان برنامه‌نویسی
- ۴.۲.۲ انتخاب دیتابیس و toolchain عه دیتابیس<sup>b</sup>
- ۵.۲.۲ پیاده‌سازی
- ۶.۲.۲ تست سرویس
- ۷.۲.۲ داکرایز کردن سرویس
- ۸.۲.۲ دیپلوی کردن سرویس
- ← **نقطه عطف:** تست و دیپلوی شدن سرویس

## قسمت ۱.۲.۱ Stock

## ۳.۲ نوشتن endpointها و منطق لازم برای ۱

- ۱.۳.۲ بررسی شفافیت و کتبی<sup>c</sup> مسئله
- ۲.۳.۲ پیدا کردن راه‌حل درست برای تحلیل زمانی
- ۳.۳.۲ توافق روی نتایجی که راه‌حل انتخاب شده ارائه
- ۴.۳.۲ پیاده‌سازی
- ۵.۳.۲ تست کدهای نوشته شده
- ← **نقطه عطف:** تایید صحت عملکرد

## ۴.۲ نوشتن endpointها و منطق لازم برای ۲

- ۱.۴.۲ بررسی شفافیت و کتبی<sup>d</sup> مسئله
- ۲.۴.۲ پیدا کردن راه‌حل درست برای تحلیل زمانی
- ۳.۴.۲ توافق روی نتایجی که راه‌حل انتخاب شده ارائه
- ۴.۴.۲ پیاده‌سازی
- ۵.۴.۲ تست کدهای نوشته شده
- ← **نقطه عطف:** تایید صحت عملکرد

## ۵.۲ نوشتن endpointها و منطق لازم برای ۳

- ۱.۵.۲ بررسی شفافیت و کتبی<sup>e</sup> مسئله
- ۲.۵.۲ پیدا کردن راه‌حل درست برای تحلیل زمانی
- ۳.۵.۲ توافق روی نتایجی که راه‌حل انتخاب شده ارائه
- ۴.۵.۲ پیاده‌سازی
- ۵.۵.۲ تست کدهای نوشته شده
- ← **نقطه عطف:** تایید صحت عملکرد

## ۶.۲ بررسی انواع خروجی ممکن که میتوان برای ۳ مورد بالا، ارائه داد

- ۱.۶.۲ بررسی مورد اول، دوم و سوم و خروجی‌های آن‌ها
- ۲.۶.۲ تحقیق در مورد مفید بودن خروجی‌های ممکن
- ۳.۶.۲ تست خروجی‌های انتخاب شده
- ← **نقطه عطف:** انتخاب خروجی‌های تایید شده و ارائه

#### قسمت ۲.۲.۱ Site

### ۷.۲ پیاده‌سازی منطق و کدهای لازم برای fulfill کردن وظایف تحلیلی این قسمت

- ۱.۷.۲ تحقیق در مورد چگونگی تحلیل
- ۲.۷.۲ پیاده‌سازی
- ۳.۷.۲ تست
- ← **نقطه عطف:** ارائه کدها و تایید صحت عملکرد صحیح

#### قسمت ۳.۲.۱ Shipment

### ۸.۲ نوشتن کدهای monitoring برای ۱

- ۱.۸.۲ بررسی کتبی و ریاضی داده‌ها دریافتی
- ۲.۸.۲ بحث و بررسی چگونگی تحلیل داده‌ها برای گرفتن نتیجه‌ی درست
- ۳.۸.۲ انتخاب منطق تحلیل
- ۴.۸.۲ پیاده‌سازی راه‌حل انتخاب شده
- ۵.۸.۲ تست کدها
- ← **نقطه عطف:** پاس شدن تست‌ها و تحویل کدها

### ۹.۲ نوشتن کدهای monitoring برای ۲

- ۱.۹.۲ بررسی کتبی و ریاضی داده‌ها دریافتی
- ۲.۹.۲ بحث و بررسی چگونگی تحلیل داده‌ها برای گرفتن نتیجه‌ی درست
- ۳.۹.۲ انتخاب منطق تحلیل
- ۴.۹.۲ پیاده‌سازی راه‌حل انتخاب شده
- ۵.۹.۲ تست کدها
- ← **نقطه عطف:** پاس شدن تست‌ها و تحویل کدها

### ۱۰.۲ نوشتن کدهای monitoring برای ۳

- ۱.۱۰.۲ بررسی کتبی و ریاضی داده‌ها دریافتی
- ۲.۱۰.۲ بحث و بررسی چگونگی تحلیل داده‌ها برای گرفتن نتیجه‌ی درست
- ۳.۱۰.۲ انتخاب منطق تحلیل
- ۴.۱۰.۲ پیاده‌سازی راه‌حل انتخاب شده
- ۵.۱۰.۲ تست کدها
- ← **نقطه عطف:** پاس شدن تست‌ها و تحویل کدها

<sup>a</sup> یعنی مثلاً بگویم برای بررسی مکان سفارش‌ها و مکان انبار انتخابی به چه داده‌ها و چگونه به آنها نیاز داریم  
<sup>b</sup> یعنی انتخاب ORM یا ODM، و همچنین سیستم نگهداری و مدیریت migrationها.  
<sup>c</sup> بررسی ریاضی‌وار  
<sup>d</sup> بررسی ریاضی‌وار  
<sup>e</sup> بررسی ریاضی‌وار

### ۳.۱ وب اپلیکیشن

می‌توان گفت چیزی که به عنوان Amazon Analytics به شرکت آمازون ارائه می‌شود، همین وب اپلیکیشن است.<sup>۱۲</sup> در واقع ۱.۱ و ۲.۱ دو قسمت internal این پلتفرم هستند و UI یی ندارند. چیزی که UI دارد و ورودی اصلی آن، تحلیل‌های ۲.۱ است، و آن‌ها را به روش‌های مختلفی همچون ۱.۱ گزارشات کتبی، ۲.۲ گزارشات آماری، ۳.۲ نمودارها، ۴.۲ گراف‌ها و... نمایش می‌دهد، همین وب اپلیکیشن است.

### ۱.۳.۱ ساختار شکست کار

#### ۳ وب اپلیکیشن

#### ۱.۳ تحلیل و تحقیق نیازمندی‌های استفاده‌کنندگان Amazon Analytics برای UX

- ۱.۱.۳ جمع‌آوری اطلاعات
- ۲.۱.۳ طوفان فکری
- ۳.۱.۳ نوشتن نیازمندی‌ها
- ۴.۱.۳ بررسی دوباره‌ی نیازمندی‌ها
- ۵.۱.۳ ارزیابی نیازمندی‌ها
- ← **نقطه عطف:** سند نیازمندی‌ها

#### ۲.۳ طراحی فیگمایی UI بر اساس تحلیل و تحقیق UX

- ۱.۲.۳ بررسی سند نیازمندی‌ها

۲.۲.۳ طراحی با استفاده از فیگما  
۳.۲.۳ بازبینی طراحی‌ها  
← نقطه عطف: طراحی‌های فیگما

---

۳.۳ نوشتن کدهای UI بر اساس طراحی‌های فیگما

۱.۳.۳ بررسی طراحی‌ها  
۲.۳.۳ بررسی فریم‌ورک‌ها  
۳.۳.۳ انتخاب فریم‌ورک  
۴.۳.۳ تحقیق و انتخاب معماری کدها  
۵.۳.۳ پیاده‌سازی  
← نقطه عطف: تحویل کدهای front-end

---

۴.۳ تست کردن کدهای front-end

← نقطه عطف: پاس شدن تمامی تست‌های کدها

---

۵.۳ طراحی یک API با سطح انتزاع بالاتر از ۲.۱ برای گرفتن اطلاعات و decouple کردن

← نقطه عطف: نوشته شدن API

---

۶.۳ تست back-end

← نقطه عطف: پاس شدن تست‌ها

---

۷.۳ وصل کردن کدهای front-end به back-end

← نقطه عطف: اجرای integration test روی web app

---