

# گزارش گام یک فاز سوم

## انتخاب سنگین ترین صفحه شبکه با توجه

### انتخاب: صفحه Dashboard

#### توجه:

صفحه‌ی داشبورد (Dashboard) به عنوان سنگین ترین صفحه در شبکه اجتماعی HumHub انتخاب شده است، زیرا این صفحه بیشترین تراکم داده و تعاملات کاربری را در بر دارد. توجه دقیق این انتخاب شامل موارد زیر است:

#### ۱. بارگذاری حجم زیادی از داده‌ها

- با ورود کاربر به داشبورد، محتوای متعددی شامل پست‌های کاربران، نظرات، لایک‌ها، تصاویر، ویجت‌ها و نوتیفیکیشن‌ها به صورت هم‌زمان بارگذاری می‌شوند.
- هر پست شامل اطلاعات نویسنده، زمان، محتوای متنی، فایل‌های پیوست، دکمه‌های تعامل (لایک، نظر) و غیره است.

#### ۲. استفاده سنگین از درخواست‌های AJAX

- صفحه داشبورد از قابلیت بارگذاری پویا (Infinite Scroll / AJAX loading) استفاده می‌کند.
- برای هر پیمایش (اسکرول)، درخواست جدیدی برای پست‌های بیشتر فرستاده می‌شود.
- این رفتار باعث افزایش حجم درخواست‌های HTTP در زمان استفاده می‌شود.

#### ۳. تعاملات کاربری بالا

- کاربران در این صفحه بیشترین تعامل را دارند: پست گذاشتن، لایک، نظر دادن، باز کردن نوتیفیکیشن‌ها، آپلود فایل و...
- این تعاملات باعث ارسال درخواست‌های متعدد از نوع GET و POST به سرور می‌شود.

#### ۴. پرتکرارترین صفحه بین کاربران

- پس از ورود، داشبورد اولین صفحه‌ای است که کاربران به آن هدایت می‌شوند.
- در نتیجه بالاترین میزان بازدید و بار سرور مربوط به همین صفحه است.

## ۵. نقاط اتصال به ماژول‌های مختلف

- داشبورد به چندین ماژول متصل است Post, Comment, Notification, UserActivity, File Upload و....
- بنابراین می‌توان عملکرد کلی سیستم را با تست این صفحه ارزیابی کرد.

### نتیجه‌گیری:

با توجه به موارد بالا، انتخاب صفحه داشبورد به عنوان سنگین‌ترین و مهم‌ترین صفحه برای تست عملکرد (Load/Stress/Soak) کاملاً موجه و فنی است. این صفحه نماینده‌ای مناسب از حجم ترافیک واقعی در سیستم است.

## تعریف سناریوهای تست (حداقل ۵ سناریو)

### سناریو ۱: مشاهده داشبورد (Dashboard View)

#### شرح:

در این سناریو کاربر پس از ورود به سامانه، تنها صفحه داشبورد را مشاهده می‌کند. این عمل یکی از متداول‌ترین درخواست‌ها در شبکه‌های اجتماعی است و در اکثر نشست‌ها به صورت پیش‌فرض انجام می‌شود.

#### درخواست‌های HTTP:

- GET /index.php?r=dashboard%2Fdashboard

### سناریو ۲: مشاهده پیام‌ها (View Messages)

#### شرح:

کاربر ابتدا داشبورد را مشاهده می‌کند و سپس وارد بخش پیام‌ها (inbox) می‌شود. این رفتار یکی از تعاملات متداول پس از ورود به سیستم است.

#### درخواست‌های HTTP:

- GET /index.php?r=dashboard%2Fdashboard
- GET /index.php?r=mail%2Fmail%2Findex

### سناریو ۳: مشاهده و ویرایش پروفایل (View & Edit Profile)

شرح:

کاربر پس از ورود، وارد صفحه پروفایل خود می‌شود و سپس سعی در ویرایش اطلاعات خود دارد. این سناریو شامل هم GET و هم POST است.

#### درخواست‌های HTTP:

- GET /index.php?r=dashboard%2Fdashboard
- GET /index.php?r=user%2Fprofile%2Fhome&cguid=...
- POST /index.php?r=admin%2Fuser%2Fedit&id=1

### سناریو ۴: تعامل اجتماعی با کاربران (Follow & Send Message)

شرح:

کاربر وارد پروفایل یکی از کاربران دیگر شده و اقدام به فالو کردن و ارسال پیام خصوصی می‌کند. این سناریو رفتار اجتماعی کاربران را شبیه‌سازی می‌کند.

#### درخواست‌های HTTP:

- GET /index.php?r=dashboard%2Fdashboard
- GET /index.php?r=user%2Fprofile%2Fhome&cguid=...
- POST /index.php?r=user%2Fprofile%2Ffollow...
- POST /index.php?r=mail%2Fmail%2Fcreate...

### سناریو ۵: مشاهده اعلان‌ها و یک اعلان خاص (Notifications Interaction)

کاربر ابتدا وارد داشبورد می‌شود، سپس لیست اعلان‌های خود را مشاهده کرده و در نهایت یک اعلان خاص را باز می‌کند.

#### درخواست‌های HTTP:

- GET /index.php?r=dashboard%2Fdashboard
- GET /index.php?r=notification%2Foverview
- GET /index.php?r=notification%2Fentry&id=6&cld=6

هدف اصلی تست	نوع درخواست‌ها	تعداد درخواست	نام سناریو	سناریو
تست بار صفحه اصلی	GET	1	مشاهده داشبورد	۱
بررسی پیام‌ها و تأثیر آن بر سرعت سیستم	GET	2	مشاهده پیام‌ها	۲
عملکرد ماژول پروفایل	GET + POST	3	ویرایش پروفایل	۳
تست تعامل اجتماعی + follow پیام خصوصی	GET + POST	4	فالو و پیام به کاربر دیگر	۴
بررسی سرعت اعلان‌ها و بارگذاری اعلان خاص	GET	3	بررسی اعلان‌ها	۵

### آزمون‌های عملکرد Stress Testing و Load Testing

تست‌های عملکرد (Performance Testing) دسته‌ای از تست‌های نرم‌افزاری هستند که رفتار سیستم را تحت شرایط مختلف بار ارزیابی می‌کنند. دو آزمون مهم در این دسته عبارتند از: **آزمون بار (Load Testing)** و **آزمون فشار (Stress Testing)**.

#### 1. آزمون بار (Load Testing)

آزمون بار به فرآیندی گفته می‌شود که در آن سیستم تحت بار معمول یا پیش‌بینی‌شده‌ای از سوی کاربران قرار می‌گیرد تا عملکرد آن در شرایط واقعی استفاده سنجیده شود.

- بررسی پاسخ‌گویی سیستم در حالت بار طبیعی یا حداکثر پیش‌بینی‌شده.
- اطمینان از اینکه سیستم می‌تواند به طور پایدار، درست و بدون کاهش محسوس در کارایی به کاربران سرویس دهد.

#### ویژگی‌ها:

- بار به تدریج یا ناگهانی تا سطح مشخصی (مثلاً ۱۰۰۰ کاربر فعال) افزایش می‌یابد.
- تمرکز بر پایداری سیستم در شرایط عادی سنگین است.
- شاخص‌هایی مانند زمان پاسخ، نرخ درخواست، مصرف منابع و درصد خطا بررسی می‌شوند.

#### مثال:

در پروژه‌ی ما، شبیه‌سازی حضور **1000 کاربر هم‌زمان با نرخ ورود ۱۰ کاربر در ثانیه** برای چند دقیقه، نمونه‌ای از Load Testing است.

## ۲. آزمون فشار (Stress Testing)

آزمون فشار حالتی از تست عملکرد است که در آن سیستم به صورت افزایشی و پله‌ای تحت بار بیشتر از ظرفیت پیش‌بینی شده قرار می‌گیرد تا آستانه‌ی توانایی سیستم و رفتار آن در حالت شکست بررسی شود.

- شناسایی نقطه شکست سیستم نقطه‌ای که دیگر قادر به پاسخگویی نیست یا به شدت کند می‌شود.

- بررسی نحوه واکنش سیستم در شرایط بحرانی و بازیابی بعد از فشار بالا.

### ویژگی‌ها:

- بار به صورت مرحله‌ای افزایش می‌یابد (مثلاً از ۱۰۰ به ۱۵۰۰ کاربر).

- شاخص‌هایی مانند زمان پاسخ بسیار بالا، نرخ خطا، مصرف منابع بالا بررسی می‌شوند.

- تعریف آستانه اهمیت دارد. مثلاً:

اگر زمان بارگذاری داشبورد از ۶۰ ثانیه بیشتر شود، سیستم در فشار بحرانی قرار دارد.

### مثال:

در پروژه‌ی ما، آزمایش با افزایش تدریجی کاربران از ۱۰۰ تا ۱۵۰۰ و بررسی کند شدن سیستم یا خطاهای HTTP، نمونه‌ای از Stress Testing است.

### تفاوت‌های آزمون Stress و Load

ویژگی	Load Testing	Stress Testing
هدف اصلی	بررسی عملکرد سیستم در بار معمول	بررسی حد نهایی توان سیستم و نقطه شکست
محدوده بار	تا سطح پیش‌بینی شده	بالتر از حد پیش‌بینی شده یا ظرفیت واقعی
تحلیل نتایج	تمرکز بر پایداری و پاسخ‌دهی	تمرکز بر رفتار سیستم در بحران و بازیابی
تمرکز اصلی	عملکرد در بار پایدار	مقاومت در شرایط غیرعادی و بحرانی
خروجی مهم	میانگین زمان پاسخ، خطاهای جزئی	زمان‌های بالا، قطع سرویس، خطاهای جدی
اهمیت در پروژه‌ها	بررسی آمادگی برای استفاده کاربران واقعی	آمادگی در برابر حملات، رفتار ناگهانی، اشکالات زیرساختی

## جمع‌بندی

- هر دو تست برای ارزیابی ظرفیت سیستم ضروری هستند.
- **Load Testing** کمک می‌کند بفهمیم آیا سیستم برای استفاده‌ی روزمره کاربران پایدار است یا نه.
- **Stress Testing** کمک می‌کند بفهمیم سیستم تا چه اندازه تحمل دارد، و اگر از حد بگذرد چه رفتارهایی از خود نشان می‌دهد.
- در پروژه‌ی HumHub، اجرای هر دو تست به ما کمک کرد هم عملکرد طبیعی و هم نقاط بحرانی سیستم را شناسایی کنیم.

## تحلیل نتایج (گراف و جداول)

بعد از اجرای تست‌ها در: Locust UI:

- نمودار زمان پاسخ‌دهی
- تعداد درخواست‌ها در ثانیه
- درصد خطا
- پایداری سیستم تحت بار

نتیجه تست‌ها به طور کامل روی گیت‌هاب قرار داده شده است.