

فاز اول پروژه تست

یاسمن گودرزی ۹۹۳۱۱۰۰ - محمد حسین زینال زاده ۹۹۳۱۰۲۴

مقدمه و هدف از پروژه

در عصر حاضر، با گسترش استفاده از سامانه‌های نرم‌افزاری در حوزه‌های مختلف زندگی در ابعاد مختلف، اطمینان از عملکرد صحیح و بدون خطای این سامانه‌ها اهمیت فراوانی دارد. نرم‌افزارها نقش حیاتی در زندگی روزمره ما ایفا می‌کنند و کیفیت و کارایی آنها تاثیر مستقیم بر تجربه کاربران دارد. تست نرم‌افزار نه تنها به بهبود کیفیت محصول نهایی کمک می‌کند، بلکه می‌تواند از بروز مشکلات جدی در محیط عملیاتی جلوگیری کند. یکی از روش‌های موثر برای دستیابی به این اطمینان، انجام تست‌های جامع و سیستماتیک بر روی سامانه‌ها است. علاوه بر این انجام تست‌ها به صورت خودکار باعث کاهش هزینه و صرفه جویی در زمان می‌شود. بنابراین تست‌ها را به صورت خودکار انجام خواهیم داد. در این گزارش، به بررسی و تست سایت آپارات که یک سایت اشتراک‌گذاری فیلم و ویدیو است، می‌پردازیم.

هدف از این پروژه، بررسی و ارزیابی طراحی رابط کاربری این وبسایت می‌باشد. واسط کاربری به عنوان لن اولین و مهم ترین جز سامانه که در دید کاربر قرار می‌گیرد، تاثیر مستقیمی بر تجربه و حس رضایت کاربر از وب سایت ما می‌گذارد. شناسایی خطاها و اشکالات موجود در طراحی رابط کاربری این وبسایت و بررسی آن‌ها از اهداف این پروژه است. شناسایی خطاها این امکان را به ما می‌دهد تا با بهبود آن در ارتقا کیفیت سامانه و افزایش قابلیت اطمینان سامانه موثر باشیم.

معرفی کامل برنامه مورد آزمون

وبسایت «آپارات» یک پلتفرم اشتراک ویدیو ایرانی است که در سال ۱۳۹۰ تأسیس شد. این سرویس امکان آپلود، مشاهده و به اشتراک‌گذاری ویدیوها را به کاربران می‌دهد. آپارات به عنوان یکی از محبوب‌ترین سایت‌های ویدیویی در ایران شناخته می‌شود و بستری است برای تولید و انتشار محتوای ویدیویی در موضوعات مختلف مانند خبری، آموزشی، سرگرمی و غیره. این سایت به کاربران اجازه می‌دهد که ویدیوها را به صورت عمومی یا خصوصی به اشتراک بگذارند و امکان دارد که کانال‌های شخصی خود را ایجاد کنند تا محتوای ویدیویی خود را مدیریت کنند.

حوزه مورد آزمون

حوزه انتخابی، آزمون محیط یا رابط کاربری است. این آزمون به منظور بررسی و تایید عملکرد و اجزای مرتبط با رابط کاربری (UI) نرم افزار یا سیستم انجام می شود .

این آزمون شامل بررسی کلیک کردن بر روی دکمه ها، وارد کردن اطلاعات در فیلدها، بررسی تغییرات در صفحات و نمایشگرها و سایر عملیات مرتبط با رابط کاربری می شود.

ارتباط میان آزمون دود (smoke testing) و آزمون رابط کاربری در این است که، در روند انجام آزمون رابط کاربری، آزمون دود نیز یکی از جنبه های مهم مورد بررسی هدف گیری می باشد. به طوری که در این مرحله از آزمون، عملکرد عمومی نرم افزار یا سیستم به صورت کلی بررسی می شود و جزئیات رابط کاربری نیز دربرگیرنده این بررسی می باشد. به عبارت دیگر، آزمون رابط کاربری به عنوان یک زیر مجموعه از آزمون دود در نظر گرفته می شود، زیرا بررسی رابط کاربری، بخشی از بررسی های عمومی آزمون دود می باشد و هر دو، هدف مشترکی را دنبال می کنند؛ که آن هم تأیید صحت عملکرد نرم افزار یا سیستم در مراحل ابتدایی می باشد.

ابزار مورد استفاده

ابزارهای مختلفی برای تست طراحی رابط کاربری وجود دارد که در ادامه به صورت تیتروار به همراه ویژگی های آنها به طور خلاصه می پردازیم:

• Selenium

- پشتیبانی از زبان های برنامه نویسی متعددی مانند: Python، Java، C# و ...

- قابلیت تعامل با مرورگرهای مختلف از جمله: Chrome، Firefox، Safari و ...

- Appium: به منظور انجام تست های Automation بر روی اپلیکیشن های موبایل (iOS و Android) استفاده می شود.

- امکان تست کردن همزمان بر روی دو پلتفرم iOS و Android.

- پشتیبانی از زبان های برنامه نویسی مانند Python، Java، Javascript و ...

- Cypress: متمرکز بر وب است.

- قابلیت اجرای تست ها در محیطی کنترل شده و بدون نیاز به مرورگر.

- امکان اجرای تست ها به صورت موازی و سریع.

- TestComplete: ابزاری کامل برای تست UI برای برنامه های ویندوز، وب و موبایل.

- دارای ویژگی هایی مانند: ضبط و ایجاد تست به صورت خودکار، پشتیبانی از برنامه های

NET، JAVA و

- Robot Framework: یک Framework تست خودکار که قابل توسعه بوده و قابل استفاده برای تست های UI و API می باشد.

- پشتیبانی از انواع مختلف پلاگین ها و کتابخانه ها برای افزایش قابلیت ها.

- Espresso: برای تست های UI بر روی برنامه های اندروید استفاده می شود.

- اجرای تست ها با سرعت بالا و در محیطی موازی.

- امکان انجام تست های ترکیبی (UI و واحد) با استفاده از چارچوب JUnit.

مزایای Selenium Grid

با استفاده از Selenium Grid، می‌توان تست‌ها را به صورت موازی و همزمان بر روی مرورگرها و سیستم‌های عامل مختلف اجرا کرد، که این امر باعث می‌شود تا زمان اجرای تست‌ها به طور قابل توجهی کاهش یابد و کارایی تست‌ها افزایش یابد. استفاده از Selenium Grid بهبود زمان بازگشت نتایج تست را نیز فراهم می‌کند، به ویژه زمانی که مجموعه تست بزرگی وجود دارد و زمان بیشتری برای اجرا نیاز است. این ابزار، انعطاف‌پذیری را فراهم می‌کند و اطمینان می‌دهد که تعداد حداکثری از تست‌ها در زمان محدودی انجام شوند. همچنین، با توجه به استفاده از زیرساخت مجازی، نگهداری آن آسان است. ما برای انجام تست از ابزار selenium استفاده خواهیم کرد.

Selenium

Selenium یک Framework بر پایه Javascript می‌باشد. این ابزار به تست‌کنندگان و توسعه‌دهندگان امکان می‌دهد تا تست‌های خودکار برای وب‌سایت‌ها و برنامه‌های وب ایجاد کنند و آن‌ها را در مرورگرهای مختلف اجرا کنند. selenium شامل چهار دسته زیر است :

Selenium IDE

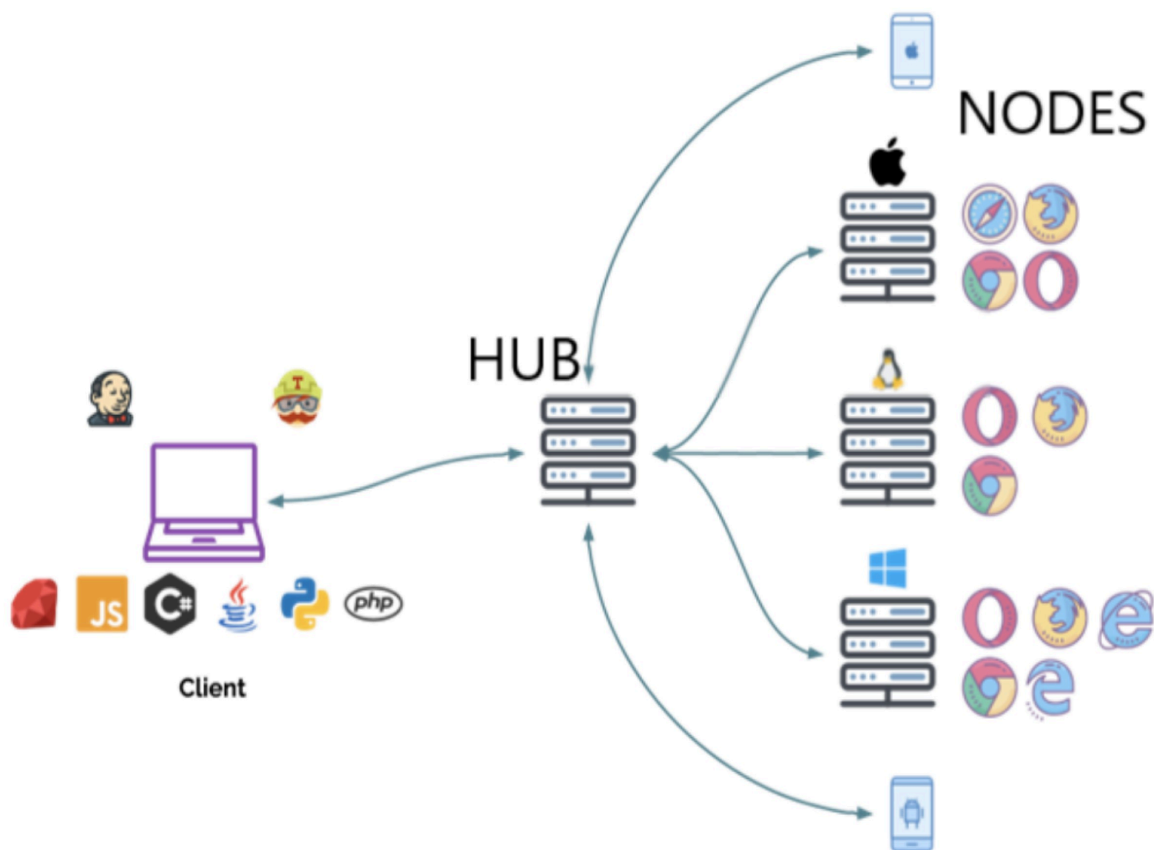
Selenium RC

Selenium Webdriver

selenium grid

Selenium Grid

Selenium Grid یک ابزار قدرتمند است که به تست‌های خودکار امکان اجرا به صورت موازی بر روی چندین مرورگر و سیستم عامل مختلف را فراهم می‌کند. بررسی معماری Selenium Grid و روند اجرای آن این مفهوم به ساختار و نحوه عملکرد Selenium Grid اشاره دارد. معماری در اینجا، به طراحی کلی سیستم، تعامل بین مؤلفه‌ها، و جریان کارهایی که در آن اجرا و انجام می‌شوند، اشاره دارد. به عبارت دیگر، معماری، مشخص‌کننده چگونگی ارتباط مؤلفه‌های Selenium Grid با یکدیگر می‌باشد. به عنوان مثال: تعیین چگونگی انجام کارهایی از جمله: مدیریت جلسات، توزیع تست‌ها و ... به طور کلی، معماری به عملکرد کلی سیستم توجه دارد.



شکل ۱

با توجه به شکل ۱، به توضیح مولفه‌های مختلف در این ساختار می‌پردازیم.

1. **Hub**: سروری است که درخواست‌های دسترسی از سمت WebDriver یا client را پذیرفته و تست‌کیس‌ها را به درایورهای ریموت در نودهای مختلف مسیردهی می‌کند. Hub تست‌ها را از Client دریافت کرده و آنها را، به صورت از راه دور، بر روی گونه‌های مختلف در نودها، به طور موازی، اجرا می‌کند. Hub به عنوان یک نقطه مرکزی برای مدیریت و مسیردهی اطلاعات و درخواست‌ها عمل می‌کند. به طور مشابه که در شبکه‌های کامپیوتری Gateway به عنوان واسطه‌ای میان شبکه داخلی و خارجی عمل می‌کند و وظیفه مسیردهی را برعهده دارد.
2. **Node**: منظور دستگاه‌های ریموت هستند (منظور از دستگاه‌های ریموت، دستگاه‌هایی با فواصل فیزیکی مختلف و از راه دور که نسبت به Hub یا دیگر node ها قرار گرفته اند). این دستگاه‌ها می‌توانند شامل دستگاه‌های مجازی نیز باشند (که شامل یک سیستم‌عامل اصلی و یک WebDriver ریموت می‌باشند). آنها درخواست‌ها را از Hub به صورت دستورات تست JSON دریافت کرده و آنها را با استفاده از WebDriver اجرا می‌کنند. به طور کلی، hub مرکز

مسیردهی تست‌ها و انجام آنها را بر روی نودها بر عهده دارد. در حالی که نودها، اجرای تست‌ها را بر روی محیط‌های مختلف واقعی انجام می‌دهند.

بنابراین، مطابق توضیحات مرتبط به ساختار و معماری کلی، برای استفاده از Selenium Grid، نیاز به دو عنصر اصلی داریم:

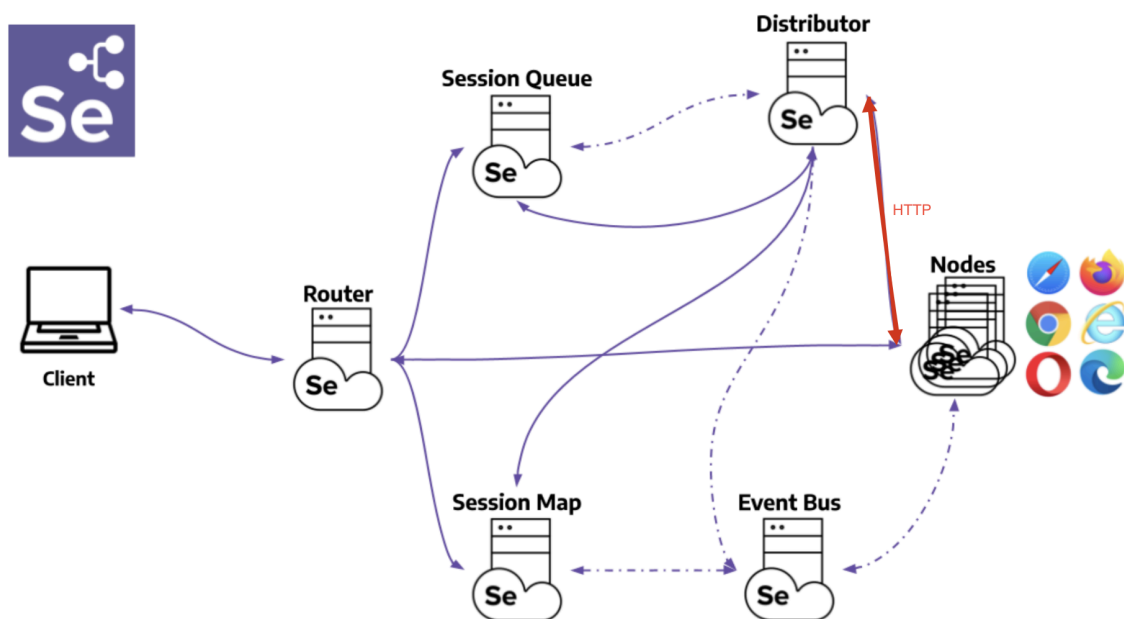
- **Hub** به عنوان نقطه مرکزی برای مدیریت تست‌ها و تخصیص آنها به Node استفاده می‌شود. Hub مسئول ایجاد و مدیریت صف تست‌ها و توزیع آنها به Node های موجود است. همچنین، Hub هماهنگی بین تست‌ها و Node های مختلف را نیز بر عهده دارد.
- **Node** ها میزبان تست‌های واقعی هستند که در محیط‌های مختلفی نصب شده‌اند. این محیط‌ها می‌توانند شامل: مرورگرها، سیستم‌های عامل و پیکربندی‌های مختلف باشند. هر Node می‌تواند تست‌ها را بر روی یکی از مرورگرها یا سیستم‌های عامل مختلف اجرا کند.

بررسی اجزای مختلف و سازماندهی شده Selenium Grid

مطابق با شکل 2، ساختار کلی selenium grid دارای اجزا مختلف و جزییات بیشتری در عملکرد کلی آن می‌باشد که هر یک وظایف مختلفی را بر عهده دارند. از جمله از این اجزا می‌توان به موارد زیر اشاره کرد که هر یک با یکدیگر تعامل و ارتباط نیز دارند:

Router: به عنوان نقطه ورودی شبکه محسوب می‌شود و درخواست‌های خارجی را دریافت کرده و آنها را به مولفه صحیح و مناسب ارسال می‌کند. اگر درخواست session جدیدی دریافت شود، آن را به صف session ارسال می‌کند. اگر درخواست مربوط به یک جلسه موجود باشد، Router، با توجه به نقشه جلسه جستجو می‌کند تا شناسه نودی که جلسه در آن اجرا می‌شود را دریافت کند و سپس درخواست را مستقیماً به نود ارسال می‌کند.

Session Map: Session Map، در واقع یک Data Mining است که رابطه بین شناسه جلسه و نودی که جلسه در آن اجرا می‌شود را، نگهداری می‌کند. این نقشه به Router در فرآیند ارسال درخواست به نود کمک می‌کند. به عبارت دیگر، Router از نقشه جلسه، برای یافتن نود مرتبط با یک شناسه جلسه استفاده می‌کند.



شکل ۲

Distributor : سعی می کند با استفاده از HTTP با Node ارتباط بگیرد تا وجود آن را تأیید کند. اگر درخواست موفقیت آمیز باشد، Distributer نود را ثبت می کند و تمام قابلیت های نود را از طریق Model Grid پیگیری می کند.

زمانی که یک درخواست Session جدید به Router ارسال می شود، به صف Session جدید منتقل می شود و در آنجا در صف قرار می گیرد. Distributor برای درخواست های جلسه جدید معلق به صف جلسه جدید پرس و جو می کند و سپس یک نود مناسب را پیدا می کند که Session در آنجا ایجاد شود. پس از ایجاد Session، توزیع کننده رابطه بین شناسه جلسه و نودی که جلسه در آنجا اجرا می شود را در Session Map ذخیره می کند.

عملکرد کلی Selenium Grid به صورت زیر می باشد:

1. مدیر سیستم Selenium Grid (HUB) را شروع می کند.
2. Node ها راه اندازی شده و به Hub متصل می شوند.
3. تست ها توسط client مانند Selenium WebDriver به Hub ارسال می شوند.
4. Hub تست ها را بین Node ها توزیع می کند و Node ها آنها را بر روی مرورگرها و سیستم های عامل مورد نظر، اجرا می کنند.
5. نتایج تست ها به سیستم تست ها گزارش داده می شوند.

اجرای تست کیس

با وجود ویژگی های مهم selenium این ابزار قابلیت نوشتن تست کیس را ندارد. برای این کار از فریم ورک robot استفاده خواهیم کرد.

از آنجا که Robot Framework از Selenium Library برای ارتباط با WebDriver استفاده می کند، امکان استفاده از Selenium Grid برای اجرای تست های خودکار با Robot Framework وجود دارد. با این حال، نیاز است که تنظیمات و پیکربندی های لازم برای اتصال Robot Framework به Selenium Grid انجام شود.

به طور کلی، سازگاری Selenium Grid با Robot Framework وجود دارد، اما برای استفاده موثر از آن، نیاز به تنظیمات و پیکربندی مناسب است تا این دو ابزار با یکدیگر هماهنگ شوند و تست ها به درستی بر روی محیط های مختلفی از Selenium Grid اجرا شوند.

برای ایجاد هماهنگی بین Selenium Grid و Robotframework و اجرای تست های خودکار در محیط های متعدد، برخی از تنظیمات و اقدامات زیر لازم است:

1. نوشتن تست ها با استفاده از Robot Framework: تست ها باید با استفاده از سینتکس Robot Framework نوشته شوند و از کتابخانه Selenium Library برای اجرای تست ها بر روی Selenium Grid استفاده کرد.

2. اجرای تست ها بر روی Selenium Grid: پس از نوشتن تست ها، باید با استفاده از Robot Framework بر روی Selenium Grid اجرا شوند. Robot Framework به صورت خودکار تست ها را به Node های Selenium Grid ارسال و نتایج آنها را گزارش می دهد.

مراحل کلی پیاده‌سازی Selenium Grid

1. Installation
2. Start Hub
3. Start Nodes
4. Configure Nodes
5. define test case in robot
6. Using Selenium Grid to run tests

لینک دانلود و نصب

https://www.selenium.dev/documentation/grid/getting_started/

مراحل نصب:

```
sudo apt install python3 python3-pip
```

```
pip3 install selenium
```

```
pip3 install robotframework
```

```
pip3 install robotframework-seleniumlibrary
```

```
pip install selenium webdriver-manager
```

نیازمندی‌های آزمون (test requirement)

مورد آزمون ۱: باز کردن سایت آپارات و هماهنگ بودن عنوان آن با اسم آپارات

عنوان مورد آزمون: بررسی هماهنگی عنوان صفحه اصلی سایت آپارات با نام "آپارات"

مراحل آزمون:

مرورگر وب را باز کنید.

آدرس URL سایت آپارات (<https://www.aparat.com>) را وارد کنید.

پس از بارگذاری کامل صفحه، عنوان صفحه را بررسی کنید.

نتیجه مورد انتظار:

عنوان صفحه شامل کلمه "آپارات" باشد.

مورد آزمون ۲: لاگین کردن در سایت آپارات

عنوان مورد آزمون: لاگین به سایت آپارات

نیازمندی‌ها:

داشتن حساب کاربری معتبر در آپارات.

دسترسی به اینترنت.

مراحل آزمون:

مرورگر وب را باز کنید.

به آدرس سایت آپارات (<https://www.aparat.com>) بروید.

روی دکمه "ثبت نام و ورود" در بالای صفحه کلیک کنید.

نام کاربری (یا ایمیل) و رمز عبور خود را وارد کنید.

روی دکمه "ورود" کلیک کنید.

نتیجه مورد انتظار:

کاربر با موفقیت وارد حساب کاربری خود شده و صفحه اصلی با نام کاربری نمایش داده شود.

مورد آزمون ۳: بررسی عناصر اصلی در صفحه اصلی آپارات

عنوان مورد آزمون: بررسی عناصر اصلی در صفحه اصلی سایت آپارات

نیازمندی‌ها:

دسترسی به اینترنت.

مراحل آزمون:

مرورگر وب را باز کنید.

به آدرس سایت آپارات (<https://www.aparat.com>) بروید.

بررسی کنید که عناصر زیر در صفحه اصلی وجود دارند:

- لوگوی آپارات
- نوار جستجو
- منوی اصلی
- ویدیوهای پیشنهادی
- دسته‌بندی‌های مختلف ویدیوها

نتیجه مورد انتظار:

تمامی عناصر ذکر شده در مراحل آزمون به درستی در صفحه اصلی نمایش داده شوند.

مورد آزمون ۴: جستجوی نام ویدیو و مشاهده نتایج جستجو

عنوان مورد آزمون: جستجوی ویدیو در سایت آپارات و بررسی نتایج جستجو

نیازمندی‌ها:

دسترسی به اینترنت.

مراحل آزمون:

مرورگر وب را باز کنید.

به آدرس سایت آپارات (<https://www.aparat.com>) بروید.

در نوار جستجو، نام ویدیوی مورد نظر را وارد کنید.

روی دکمه جستجو کلیک کنید.

نتایج جستجو را بررسی کنید و اطمینان حاصل کنید که ویدیوهای مرتبط نمایش داده شوند.

نتیجه مورد انتظار:

ویدیوهای مرتبط با عبارت جستجو شده در نتایج جستجو نمایش داده شوند.

طرح آزمون (test plan)

TP-Aparat-001 :

شناسه برنامه تست

مقدمه

این سند به منظور برنامه‌ریزی و توصیف فعالیت‌های تست نرم‌افزار برای سایت آپارات تهیه شده است. در این برنامه تست، موارد و ویژگی‌های مورد آزمون، منابع مورد نیاز، روش‌های تست، و زمان‌بندی فعالیت‌های تست مشخص می‌شوند.

اقلام تست:

سایت آپارات (<https://www.aparat.com>)

نسخه: جدیدترین نسخه تولیدی

ویژگی‌های مورد آزمون:

هماهنگی تایتل صفحه اصلی با نام "آپارات"

فرآیند لاگین کردن

عناصر اصلی در صفحه اصلی

جستجوی ویدیو و مشاهده نتایج

ویژگی‌های خارج از محدوده تست

عملکردهای جانبی سایت نظیر بارگذاری ویدیو، سیستم نظرات و ... در این تست بررسی نمی‌شود.

رویکرد کلی تست:

تست خودکار برای اطمینان از عملکرد صحیح ویژگی‌های ذکر شده.

استفاده از مرورگرهای مختلف برای بررسی سازگاری سایت.

فعالیت‌ها و تکنیک‌های اصلی:

مرور سایت و بررسی دستی تایتل

تست لاگین با استفاده از حساب کاربری

بررسی عناصر اصلی صفحه اصلی

جستجوی ویدیو و مشاهده نتایج

معیارهای پذیرش/رد تست:

موفقیت در هماهنگی تایتل صفحه با نام آپارات.

ورود موفقیت‌آمیز به حساب کاربری با اطلاعات صحیح.

نمایش صحیح تمامی عناصر اصلی صفحه اصلی.

نمایش ویدیوهای مرتبط با عبارت جستجو شده.

معیارهای توقف و از سرگیری تست

توقف تست در صورت بروز خطاهای بحرانی یا ناتوانی در دسترسی به سایت.
از سرگیری تست پس از رفع خطاها و بررسی مجدد عملکرد صحیح.
تحویل دهی‌ها:

گزارش تست (Test Report)

مشخصات طراحی تست (Test Design Specifications)

موارد تست (Test Cases)

گزارش لاگ‌های تست (Test Logs)

گزارش رویدادهای تست (Test Incident Reports)

گزارش خلاصه تست (Test Summary Report)

وظایف تست:

مدیریت تست: تعیین منابع و زمان‌بندی.

طراحی تست: تهیه و تنظیم موارد تست.

اجرای تست: اجرای تست‌ها و ثبت نتایج.

بررسی و تحلیل: بررسی نتایج تست و تحلیل رویدادها.

نیازهای محیطی:

دسترسی به اینترنت با سرعت مناسب.

سیستم‌های تست شامل مرورگرهای مختلف (Chrome, Firefox, Safari, etc).

سطح امنیتی مناسب برای حفاظت از اطلاعات حساب‌های کاربری.

مسئولیت‌ها:

مدیر تست: مسئول مدیریت کلی تست و زمان‌بندی.

تیم طراحی تست: مسئول تهیه موارد تست و سناریوهای تست.

تیم اجرای تست: مسئول اجرای تست‌ها و ثبت نتایج.

تیم بررسی و تحلیل: مسئول تحلیل نتایج و گزارش‌دهی.

نیازهای نیروی انسانی و آموزشی

نیاز به تسترهای ماهر در زمینه تست وب‌سایت.

آموزش‌های لازم برای آشنایی با سایت آپارات و ابزارهای مورد استفاده.

زمان‌بندی:

تعیین نقاط عطف تست.

برآورد زمان مورد نیاز برای هر فعالیت تست.

زمان‌بندی دقیق برای تمامی وظایف تست و نقاط عطف.

ریسک‌ها و برنامه‌های پشتیبانی

ریسک‌ها:

عدم دسترسی به سایت آپارات.

بروز خطاهای غیرقابل پیش‌بینی.

برنامه‌های پشتیبانی:
تهیه نسخه‌های پشتیبان از اطلاعات.
برنامه‌ریزی برای رفع سریع خطاها.

سند موارد آزمون

برای نوشتن تست کیس‌ها ابتدا محتوا آنها را با استفاده از فریم ورک روبات می‌نویسیم و سپس در فایل
Apparat-Steps.resource متغیرهای آن را تعریف میکنیم.
برای اجرا از دستور زیر استفاده کنید

robot Aparat-Test-Case_Design.robot
robot Aparat-Test-Case_Compatibility.robot
robot Aparat-Test-Case_Functionality.robot

Aparat-Test-Case_Design

در این مجموعه تست کیس به دنبال تست کردن دیزاین و محتوای سایت هستیم

test case 1:

```
*** Test Cases ***  
Check Main Elements on Aparat Home Page  
    Open Browser On Chrome  
    Check Aparat WebSite Logo  
    Check Aparat WebSite SearchBar  
    Closed Browser On Chrome
```

```

Open Browser On Chrome
    Open Browser  ${TARGET_URL}  ${BROWSER}

Check Aparat WebSite Logo
    Element Should Be Visible  css=.header-item > a > svg.icon-logo-fa

Check Apparat WebSite SearchBar
    Element Should Be Visible  id=search

Closed Browser On Chrome
    Close Browser

```

بررسی میکنیم که آیا المنت اصلی سایت که شامل لوگوی اصلی سایت است بالا می آید . و همچنین سرچ بار که مهم ترین و بخش اصلی سایت است بالا بیاید.

test case 2:

```

Check Login Button Color and Font
    Open Browser On Chrome
    Wait Until Page Contains Element  ${LOGIN_BUTTON_CSS}  timeout=10s
    ${button_color}    Get Element Attribute  ${LOGIN_BUTTON_CSS}  ${LOGIN_BUTTON_XPATH}  background-color
    ${button_font}     Get Element Attribute  ${LOGIN_BUTTON_CSS}  ${LOGIN_BUTTON_XPATH}  font-family
    ${button_font_size} Get Element Attribute  ${LOGIN_BUTTON_CSS}  ${LOGIN_BUTTON_XPATH}  font-size

    Log    Button Color: ${button_color}
    Log    Button Font: ${button_font}
    Log    Button Font Size: ${button_font_size}

    Close Browser

```

```

Get Element Attribute
[Arguments]  ${css}  ${xpath}  ${attribute}
${value}    Execute Javascript
...    var css = arguments[0];
...    var xpath = arguments[1];
...    var attribute = arguments[2];
...    var element = null;
...    try {
...        if (css && document.querySelector(css)) {
...            element = document.querySelector(css);
...        } else if (xpath) {
...            element = document.evaluate(xpath, document, null, XPathResult.FIRST_ORDERED_NODE_TYPE, null).singleNodeValue;
...        }
...    } catch (e) {
...        return 'Invalid selector or element not found';
...    }
...    if (element === null) {
...        return 'Element not found';
...    }
...    return window.getComputedStyle(element).getPropertyValue(attribute);
...    ARGUMENTS  ${css}  ${xpath}  ${attribute}
[Return]  ${value}

```

در تست کیس دوم هدف بررسی رنگ و فونت پس زمینه المنت ها است

Aparat-Test-Case_Compatibility

در این مجموعه تست کیس هدف آزمون محیط کاربر است. برای این کار یک فعالیت یکسان را در دو تست کیس مجزا یکبار برای کروم و بار دیگر برای فایرفاکس تست میکنیم.

test case 3 and 4:

| |
|----------------------------------|
| Browser Compatibility On Firefox |
| Run Test On Browser firefox |
| Closed Browser On Chrome |
| Browser Compatibility On Chrome |
| 💡 Run Test On Browser Chrome |
| Closed Browser On Chrome |
| |

| |
|---|
| Run Test On Browser firefox |
| Open Browser \${TARGET_URL} firefox |
| Wait Until Element Is Visible css=#header 10s |
| Element Should Be Visible css=#header |
| Run Test On Browser chrome |
| Open Browser \${TARGET_URL} chrome |
| Wait Until Element Is Visible css=#header 10s |
| Element Should Be Visible css=#header |

test case 5:

| |
|-------------------------------------|
| Open Aparat WebSite and Check Title |
| Open Browser On Chrome |
| Check WebSite Title |
| Closed Browser On Chrome |
| |

در این مورد آزمون، هدف بررسی این است که آیا سایت آپارات به درستی در مرورگر گوگل نشان داده می شود یا خیر. برای این کار تست کیس زیر را می نویسیم که سایت آپارات را در مرورگر باز میکند و عنوان آن را بررسی میکند.

Aparat-Test-Case_Functionality

در این مجموعه تست کیس ها دنبال فانکشنالیتی فعالیت ها هستیم

test case 6:

```
Search Video's Name and Check Search's Result
```

```
Open Browser On Chrome
```

```
Entered Videos Name To Search Bar
```

```
Click On Search Button
```

```
Check Aparat WebSite Result
```

```
Closed Browser On Chrome
```

```
Entered Videos Name To Search Bar
```

```
Input Text id=search امیرکبیر
```

```
Click On Search Button
```

```
Press Key id=search \\13
```

```
Check Aparat WebSite Result
```

```
Wait Until Page Contains "نتایج جستجوی" امیرکبیر
```

در این تست کیس هدفمون بررسی قابلیت سرچ در وبسایت آپارات بوده است.

test case 7:

```
Verify Skip Ad Button on Aparat
```

```
Open Browser On Movie
```

```
Sleep 10s
```

```
Visible Skip Ad Button
```

```
Click Skip Ad Button
```

```
Sleep 5s
```

```
Close Browser
```



```

Visible Skip Ad Button
    Wait Until Element Is Visible    xpath://div[@class='vast-skip-button']    10s
    Wait Until Element Is Enabled    xpath://div[@class='vast-skip-button']    10s

Click Skip Ad Button
    Click Element    xpath://div[@class='vast-skip-button']

```

در این تست کیس هدفمون رد کردن تبلیغ های موجود در اول فیلم است. ابتدا بررسی میکنیم این دکمه وجود داشته باشد سپس روی آن کلیک میکنیم.

test case 8:

```

Verify Jump Back Button on Aparat
    Open Browser On Movie
    Sleep                10s
    Visible Skip Ad Button
    Click Skip Ad Button
    Sleep                40s
    Click Jump Back Button
    Sleep                5s
    Close Browser

```

```

Click Jump Back Button
    Execute JavaScript    document.querySelector("button.romeo-button.jump-back").click()

```

در این تست کیس بررسی میکنیم که دکمه بازگشت پنج ثانیه در صفحه قابل کلیک باشد.

test case 9:

```

Verify Play Pause Toggle on Aparat
  Open Browser On Movie
  Sleep          5s
  Visible Skip Ad Button
  Click Skip Ad Button
  Sleep          5s
  Visible Play Button
  Click Play Button
  Sleep          2
  Check Pause
  Click Play Button
  Sleep          2
  Check Play
  Close Browser

```

```

Visible Play Button
  Wait Until Element Is Visible  //button[@class='romeo-button romeo-play-toggle']
Click Play Button
  Execute JavaScript  document.querySelector('button.romeo-button.romeo-play-toggle').click()
Check Pause
  ${pause_status}= Get Element Attribute  xpath://button[@class='romeo-button romeo-play-toggle']  aria-label
  Should Be Equal As Strings  ${pause_status}  پخش  K

Check Play
  ${play_status}= Get Element Attribute  xpath://button[@class='romeo-button romeo-play-toggle']  aria-label
  Should Be Equal As Strings  ${play_status}  توقف  K

```

در این تست کیس بررسی میکنیم که دکمه پلی قابل کلیک باشد. ابتدا تبلیغ را اسکیپ میکنیم و سپس روی دکمه توقف کلیک میکنیم و بعد روی دکمه پخش و وضعیت را بررسی میکنیم.

test case 10:

در این تست کیس بررسی میکنیم که بتوان بر روی دکمه لاگین کلیک کرد. سپس بررسی میکنیم که کاربر بتواند شماره موبایل خود را در کادر وارد کند

```

Login to Aparat
  Open Browser On Chrome
  Click Login And Sign Up Button
  Sleep  10
  Close Browser

```

```
Click Login And Sign Up Button
Click Element  css=button[aria-label="user"]
```

سند نتایج اجرای آزمون

test case 1:

```
(.venv) yasaman@yasaman-VivoBook-ASUSLaptop-X509JP-R521JP:~/PycharmProjects/TestProject$ robot Aparat-Test-Case_Design.robot
=====
Aparat-Test-Case Design
=====
Check Main Elements on Aparat Home Page | PASS |
-----
Aparat-Test-Case Design | PASS |
1 test, 1 passed, 0 failed
=====
```

Aparat-Test-Case Design Log

Generated
20240630 23:29:33 UTC+03:30
26 seconds ago

Test Statistics

| Total Statistics | Total | Pass | Fail | Skip | Elapsed | Pass / Fail / Skip |
|------------------|-------|------|------|------|----------|--------------------|
| All Tests | 1 | 1 | 0 | 0 | 00:00:22 | <div></div> |

| Statistics by Tag | Total | Pass | Fail | Skip | Elapsed | Pass / Fail / Skip |
|-------------------|-------|------|------|------|---------|--------------------|
| No Tags | | | | | | <div></div> |

| Statistics by Suite | Total | Pass | Fail | Skip | Elapsed | Pass / Fail / Skip |
|-------------------------|-------|------|------|------|----------|--------------------|
| Aparat-Test-Case Design | 1 | 1 | 0 | 0 | 00:00:22 | <div></div> |

Test Execution Log

SUITE

Aparat-Test-Case Design

Full Name:

Aparat-Test-Case Design

Source:

/home/yasaman/PycharmProjects/TestProject/Aparat-Test-Case_Design.robot

Start / End / Elapsed:

20240630 23:29:11.819 / 20240630 23:29:33.966 / 00:00:22.147

Status:

1 test total, 1 passed, 0 failed, 0 skipped

TEST

Check Main Elements on Aparat Home Page

Full Name:

Aparat-Test-Case Design.Check Main Elements on Aparat Home Page

Start / End / Elapsed:

20240630 23:29:11.928 / 20240630 23:29:33.965 / 00:00:22.037

Status:

PASS

+ KEYWORD

Aparat-Steps . Open Browser On Chrome

+ KEYWORD

Aparat-Steps . Check Aparat WebSite Logo

+ KEYWORD

Aparat-Steps . Check Aparat WebSite SearchBar

+ KEYWORD

Aparat-Steps . Closed Browser On Chrome

test case 2:

```
ssaman@godardzi-pouyan:~/PycharmProjects/testproject/final$ robot Aparat-Test-Case_Design.robot
[WARN ] Error in file '/home/yasaman/PycharmProjects/testproject/final/Apparat-Steps.resource' on line 44: The '[Return]' setting is deprecated. Use the 'RETURN' statement instead.
=====
Aparat-Test-Case Design
=====
Check Login Button Color and Font | PASS |
=====
Aparat-Test-Case Design | PASS |
. test, 1 passed, 0 failed
=====
Output: /home/yasaman/PycharmProjects/testproject/final/output.xml
Log: /home/yasaman/PycharmProjects/testproject/final/log.html
```

Aparat-Test-Case Design Log

Generated
20240701 10:26:04 UTC+03:30
29 seconds ago

Test Statistics

| Total Statistics | Total | Pass | Fail | Skip | Elapsed | Pass / Fail / Skip |
|-------------------------|-------|------|------|------|----------|--------------------|
| All Tests | 2 | 2 | 0 | 0 | 00:00:14 | <div></div> |
| Statistics by Tag | Total | Pass | Fail | Skip | Elapsed | Pass / Fail / Skip |
| No Tags | | | | | | <div></div> |
| Statistics by Suite | Total | Pass | Fail | Skip | Elapsed | Pass / Fail / Skip |
| Aparat-Test-Case Design | 2 | 2 | 0 | 0 | 00:00:14 | <div></div> |

Test Execution Errors

20240701 10:25:50.066 [WARN] Error in file '/home/yasaman/PycharmProjects/testproject/final/Apparat-Steps.resource' on line 44: The '[Return]' setting is deprecated. Use the 'RETURN' statement instead.

Test Execution Log

SUITE

Aparat-Test-Case Design

Full Name:

Aparat-Test-Case Design

Source:

/home/yasaman/PycharmProjects/testproject/final/Aparat-Test-Case_Design.robot

Start / End / Elapsed:

20240701 10:25:49.949 / 20240701 10:26:04.228 / 00:00:14.279

Status:

2 tests total, 2 passed, 0 failed, 0 skipped

TEST

Check Main Elements on Aparat Home Page

TEST

Check Login Button Color and Font

Full Name:

Aparat-Test-Case Design.Check Login Button Color and Font

Start / End / Elapsed:

20240701 10:26:01.878 / 20240701 10:26:04.228 / 00:00:02.350

Status:

PASS

KEYWORD

Aparat-Steps.Open Browser On Chrome

KEYWORD

SeleniumLibrary.Wait Until Page Contains Element \${LOGIN_BUTTON_CSS} timeout=10s

KEYWORD

\${button_color} = Aparat-Steps.Get Element Attribute \${LOGIN_BUTTON_CSS} \${LOGIN_BUTTON_XPATH} background-color

KEYWORD

\${button_font} = Aparat-Steps.Get Element Attribute \${LOGIN_BUTTON_CSS} \${LOGIN_BUTTON_XPATH} font-family

KEYWORD

\${button_font_size} = Aparat-Steps.Get Element Attribute \${LOGIN_BUTTON_CSS} \${LOGIN_BUTTON_XPATH} font-size

KEYWORD

Buildin.Log Button Color: \${button_color}

KEYWORD

Buildin.Log Button Font: \${button_font}

KEYWORD

Buildin.Log Button Font Size: \${button_font_size}

KEYWORD

SeleniumLibrary.Close Browser

test case 3 and 4:

```
Terminal: Local x + v
Output: /home/yasaman/PycharmProjects/testproject/final/output.xml
Log: /home/yasaman/PycharmProjects/testproject/final/log.html
Report: /home/yasaman/PycharmProjects/testproject/final/report.html
yasaman@godarzi-pouyan:~/PycharmProjects/testproject/final$ robot Aparat-Test-Case_Compatibility.robot
=====
Aparat-Test-Case Compatibility
=====
Browser Compatibility On Firefox | PASS |
-----
Browser Compatibility On Chrome | PASS |
-----
Aparat-Test-Case Compatibility | PASS |
2 tests, 2 passed, 0 failed
=====
Output: /home/yasaman/PycharmProjects/testproject/final/output.xml
Log: /home/yasaman/PycharmProjects/testproject/final/log.html
Report: /home/yasaman/PycharmProjects/testproject/final/report.html
yasaman@godarzi-pouyan:~/PycharmProjects/testproject/final$
```

Testcase2 Log

Generated
20240701 07:48:05 UTC+03:30
1 hour 29 minutes ago

Test Statistics

| Total Statistics | Total | Pass | Fail | Skip | Elapsed | Pass / Fail / Skip |
|------------------|-------|------|------|------|----------|--------------------|
| All Tests | 1 | 1 | 0 | 0 | 00:00:24 | <div></div> |

| Statistics by Tag | Total | Pass | Fail | Skip | Elapsed | Pass / Fail / Skip |
|-------------------|-------|------|------|------|---------|--------------------|
| No Tags | | | | | | <div></div> |

| Statistics by Suite | Total | Pass | Fail | Skip | Elapsed | Pass / Fail / Skip |
|---------------------|-------|------|------|------|----------|--------------------|
| Testcase2 | 1 | 1 | 0 | 0 | 00:00:24 | <div></div> |

Test Execution Log

SUITE Testcase2

Full Name: Testcase2

Source: /home/yasaman/PycharmProjects/testproject/testcase2.robot

Start / End / Elapsed: 20240701 07:47:40.868 / 20240701 07:48:05.049 / 00:00:24.181

Status: 1 test total, 1 passed, 0 failed, 0 skipped

TEST Verify Skip Ad Button on Aparat

Full Name: Testcase2.Verify Skip Ad Button on Aparat

Start / End / Elapsed: 20240701 07:47:40.992 / 20240701 07:48:05.048 / 00:00:24.056

Status: PASS

KEYWORD SeleniumLibrary.Open Browser \${URL} \${BROWSER}

KEYWORD SeleniumLibrary.Maximize Browser Window

KEYWORD BuiltIn.Sleep 1s

KEYWORD SeleniumLibrary.Wait Until Element Is Visible xpath://div[@class='vast-skip-button'] 10s

KEYWORD SeleniumLibrary.Wait Until Element Is Enabled xpath://div[@class='vast-skip-button'] 10s

KEYWORD SeleniumLibrary.Click Element xpath://div[@class='vast-skip-button']

KEYWORD BuiltIn.Sleep 5s

KEYWORD SeleniumLibrary.Close Browser

test case 5:

```
(.venv) yasaman@yasaman-VivoBook-ASUSLaptop-X509JP-R521JP:~/PycharmProjects/TestProject$ robot Aparat-Test-Case_Compatibility.robot
=====
Aparat-Test-Case Compatibility
=====
Open Aparat WebSite and Check Title | PASS |
```

Aparat-Test-Case Compatibility Log

Generated
20240630 23:47:20 UTC+03:30
4 seconds ago

Test Statistics

| Total Statistics | Total | Pass | Fail | Skip | Elapsed | Pass / Fail / Skip |
|------------------|-------|------|------|------|----------|--------------------|
| All Tests | 1 | 1 | 0 | 0 | 00:00:32 | <div></div> |

| Statistics by Tag | Total | Pass | Fail | Skip | Elapsed | Pass / Fail / Skip |
|-------------------|-------|------|------|------|---------|--------------------|
| No Tags | | | | | | <div></div> |

| Statistics by Suite | Total | Pass | Fail | Skip | Elapsed | Pass / Fail / Skip |
|--------------------------------|-------|------|------|------|----------|--------------------|
| Aparat-Test-Case Compatibility | 1 | 1 | 0 | 0 | 00:00:32 | <div></div> |

Test Execution Log

SUITE

Aparat-Test-Case Compatibility

Full Name:

Aparat-Test-Case Compatibility

Source:

/home/yasaman/PycharmProjects/TestProject/Aparat-Test-Case_Compatibility.robot

Start / End / Elapsed:

20240630 23:46:48.890 / 20240630 23:47:20.632 / 00:00:31.742

Status:

1 test total, 1 passed, 0 failed, 0 skipped

TEST

Open Aparat WebSite and Check Title

Full Name:

Aparat-Test-Case Compatibility.Open Aparat WebSite and Check Title

Start / End / Elapsed:

20240630 23:46:49.003 / 20240630 23:47:20.631 / 00:00:31.628

Status:

PASS

KEYWORD

Aparat-Steps.Open Browser On Chrome

KEYWORD

Aparat-Steps.Check WebSite Title

KEYWORD

Aparat-Steps.Closed Browser On Chrome

test case 6:

```
Aparat-Test-Case Functionality
=====
Search Video's Name and Check Search's Result | PASS |
-----

Aparat-Test-Case Functionality | PASS |
1 test, 1 passed, 0 failed
=====

Output: /home/yasaman/PycharmProjects/TestProject/output.xml
Log: /home/yasaman/PycharmProjects/TestProject/log.html
Report: /home/yasaman/PycharmProjects/TestProject/report.html
(.venv) yasaman@yasaman-VivoBook-ASUSLaptop-X509JP-R521JP:~/PycharmProjects/TestProject$
```

Aparat-Test-Case Functionality Log

Generated
20240630 23:59:26 UTC+03:30
1 minute 13 seconds ago

Test Statistics

| Total Statistics | Total | Pass | Fail | Skip | Elapsed | Pass / Fail / Skip |
|--------------------------------|-------|------|------|------|----------|--------------------|
| All Tests | 1 | 1 | 0 | 0 | 00:00:46 | |
| Statistics by Tag | Total | Pass | Fail | Skip | Elapsed | Pass / Fail / Skip |
| No Tags | | | | | | |
| Statistics by Suite | Total | Pass | Fail | Skip | Elapsed | Pass / Fail / Skip |
| Aparat-Test-Case Functionality | 1 | 1 | 0 | 0 | 00:00:46 | |

Test Execution Log

SUITE

Aparat-Test-Case Functionality

Full Name:

Aparat-Test-Case Functionality

Source:

/home/yasaman/PycharmProjects/TestProject/Aparat-Test-Case_Functionality.robot

Start / End / Elapsed:

20240630 23:58:40.511 / 20240630 23:59:26.900 / 00:00:46.389

Status:

1 test total, 1 passed, 0 failed, 0 skipped

TEST

Search Video's Name and Check Search's Result

Full Name:

Aparat-Test-Case Functionality.Search Video's Name and Check Search's Result

Start / End / Elapsed:

20240630 23:58:40.676 / 20240630 23:59:26.899 / 00:00:46.223

Status:

PASS

KEYWORD

Apparat-Steps.Open Browser On Chrome

KEYWORD

Apparat-Steps.Entered Videos Name To Search Bar

KEYWORD

Apparat-Steps.Click On Search Button

KEYWORD

Apparat-Steps.Check Aparat WebSite Result

KEYWORD

Apparat-Steps.Closed Browser On Chrome

test case 7:

```
(.venv) yasaman@yasaman-VivoBook-ASUSLaptop-X509JP-R521JP:~/PycharmProjects/TestProject$ robot Aparat-Test-Case_Functionality.robot
=====
Aparat-Test-Case Functionality
=====
Verify Skip Ad Button on Aparat | PASS |
-----
Aparat-Test-Case Functionality | PASS |
1 test, 1 passed, 0 failed
=====
Output: /home/yasaman/PycharmProjects/TestProject/output.xml
Log: /home/yasaman/PycharmProjects/TestProject/log.html
Report: /home/yasaman/PycharmProjects/TestProject/report.html
```

Aparat-Test-Case Functionality Log

Generated
20240701 00:03:13 UTC+03:30
59 seconds ago

Test Statistics

| Total Statistics | Total | Pass | Fail | Skip | Elapsed | Pass / Fail / Skip |
|--------------------------------|-------|------|------|------|----------|--------------------|
| All Tests | 1 | 1 | 0 | 0 | 00:00:33 | <div></div> |
| Statistics by Tag | Total | Pass | Fail | Skip | Elapsed | Pass / Fail / Skip |
| No Tags | | | | | | <div></div> |
| Statistics by Suite | Total | Pass | Fail | Skip | Elapsed | Pass / Fail / Skip |
| Aparat-Test-Case Functionality | 1 | 1 | 0 | 0 | 00:00:33 | <div></div> |

Test Execution Log

SUITE

Aparat-Test-Case Functionality

Full Name: Aparat-Test-Case Functionality

Source: /home/yasaman/PycharmProjects/TestProject/Aparat-Test-Case_Functionality.robot

Start / End / Elapsed: 20240701 00:02:40.291 / 20240701 00:03:13.011 / 00:00:32.720

Status: 1 test total, 1 passed, 0 failed, 0 skipped

TEST

Verify Skip Ad Button on Aparat

Full Name: Aparat-Test-Case Functionality.Verify Skip Ad Button on Aparat

Start / End / Elapsed: 20240701 00:02:40.448 / 20240701 00:03:13.011 / 00:00:32.563

Status: PASS

KEYWORD

Aparat-Steps.Open Browser On Movie

KEYWORD

BuiltIn.Sleep 20s

KEYWORD

Aparat-Steps.Visible Skip Ad Button

KEYWORD

Aparat-Steps.Click Skip Ad Button

KEYWORD

BuiltIn.Sleep 5s

KEYWORD

SeleniumLibrary.Close Browser

test case 8:

```
yasaman@godarzi-pouyan:~/PycharmProjects/testproject/final$ robot Aparat-Test-Case_Functionality.robot
=====
Aparat-Test-Case Functionality
=====
Verify Jump Back Button on Aparat | PASS |
-----
Aparat-Test-Case Functionality | PASS |
1 test, 1 passed, 0 failed
=====
Output: /home/yasaman/PycharmProjects/testproject/final/output.xml
Log: /home/yasaman/PycharmProjects/testproject/final/log.html
Report: /home/yasaman/PycharmProjects/testproject/final/report.html
yasaman@godarzi-pouyan:~/PycharmProjects/testproject/final$
```


Testcase2 Log

Generated
20240701 07:48:05 UTC+03:30
1 hour 55 minutes ago

Test Statistics

| Total Statistics | Total | Pass | Fail | Skip | Elapsed | Pass / Fail / Skip |
|------------------|-------|------|------|------|----------|--------------------|
| All Tests | 1 | 1 | 0 | 0 | 00:00:24 | <div></div> |

| Statistics by Tag | Total | Pass | Fail | Skip | Elapsed | Pass / Fail / Skip |
|-------------------|-------|------|------|------|---------|--------------------|
| No Tags | | | | | | <div></div> |

| Statistics by Suite | Total | Pass | Fail | Skip | Elapsed | Pass / Fail / Skip |
|---------------------|-------|------|------|------|----------|--------------------|
| Testcase2 | 1 | 1 | 0 | 0 | 00:00:24 | <div></div> |

Test Execution Log

SUITE

Testcase2

Full Name:

Testcase2

Source:

/home/yasaman/PycharmProjects/testproject/testcase2.robot

Start / End / Elapsed:

20240701 07:47:40.868 / 20240701 07:48:05.049 / 00:00:24.181

Status:

1 test total, 1 passed, 0 failed, 0 skipped

TEST

Verify Skip Ad Button on Aparat

test case 9:

```
yasaman@godarzi-pouyan:~/PycharmProjects/testproject/final$ robot Aparat-Test-Case_Functionality.robot
=====
Aparat-Test-Case Functionality
=====
Verify Play Pause Toggle on Aparat | PASS |
-----
Aparat-Test-Case Functionality | PASS |
1 test, 1 passed, 0 failed
=====
Output: /home/yasaman/PycharmProjects/testproject/final/output.xml
Log: /home/yasaman/PycharmProjects/testproject/final/log.html
Report: /home/yasaman/PycharmProjects/testproject/final/report.html
yasaman@godarzi-pouyan:~/PycharmProjects/testproject/final$
```

Aparat-Test-Case Functionality Log

Generated
20240701 10:11:20 UTC+03:30
22 seconds ago

Test Statistics

| Total Statistics | Total | Pass | Fail | Skip | Elapsed | Pass / Fail / Skip |
|--------------------------------|-------|------|------|------|----------|--------------------|
| All Tests | 1 | 1 | 0 | 0 | 00:00:18 | <div></div> |
| Statistics by Tag | Total | Pass | Fail | Skip | Elapsed | Pass / Fail / Skip |
| No Tags | | | | | | <div></div> |
| Statistics by Suite | Total | Pass | Fail | Skip | Elapsed | Pass / Fail / Skip |
| Aparat-Test-Case Functionality | 1 | 1 | 0 | 0 | 00:00:18 | <div></div> |

Test Execution Log

SUITE

Aparat-Test-Case Functionality

Full Name:

Aparat-Test-Case Functionality

Source:

/home/yasaman/PycharmProjects/testproject/final/Aparat-Test-Case_Functionality.robot

Start / End / Elapsed:

20240701 10:11:01.924 / 20240701 10:11:20.233 / 00:00:18.309

Status:

1 test total, 1 passed, 0 failed, 0 skipped

TEST

Verify Play Pause Toggle on Aparat

Full Name:

Aparat-Test-Case Functionality.Verify Play Pause Toggle on Aparat

Start / End / Elapsed:

20240701 10:11:02.042 / 20240701 10:11:20.232 / 00:00:18.190

Status:

PASS

+ KEYWORD

Aparat-Steps.Open Browser On Movie

+ KEYWORD

Builtin.Sleep 5s

+ KEYWORD

Aparat-Steps.Visible Skip Ad Button

+ KEYWORD

Aparat-Steps.Click Skip Ad Button

+ KEYWORD

Builtin.Sleep 5s

+ KEYWORD

Aparat-Steps.Visible Play Button

+ KEYWORD

Aparat-Steps.Click Play Button

+ KEYWORD

Builtin.Sleep 2

+ KEYWORD

Aparat-Steps.Check Pause

+ KEYWORD

Aparat-Steps.Click Play Button

+ KEYWORD

Builtin.Sleep 2

+ KEYWORD

Aparat-Steps.Check Play

+ KEYWORD

SeleniumLibrary.Close Browser

test case 10:

```
yasaman@godarzi-pouyan:~/PycharmProjects/testproject/final$ robot Aparat-Test-Case_Functionality.robot
=====
Aparat-Test-Case Functionality
=====
Login to Aparat | PASS |
-----
Aparat-Test-Case Functionality | PASS |
1 test, 1 passed, 0 failed
=====
Output: /home/yasaman/PycharmProjects/testproject/final/output.xml
Log: /home/yasaman/PycharmProjects/testproject/final/log.html
Report: /home/yasaman/PycharmProjects/testproject/final/report.html
```

Aparat-Test-Case Functionality Log

Generated
20240701 10:09:38 UTC+03:30
34 seconds ago

Test Statistics

| Total Statistics | Total | Pass | Fail | Skip | Elapsed | Pass / Fail / Skip |
|--------------------------------|-------|------|------|------|----------|--------------------|
| All Tests | 1 | 1 | 0 | 0 | 00:00:35 | <div></div> |
| Statistics by Tag | Total | Pass | Fail | Skip | Elapsed | Pass / Fail / Skip |
| No Tags | | | | | | <div></div> |
| Statistics by Suite | Total | Pass | Fail | Skip | Elapsed | Pass / Fail / Skip |
| Aparat-Test-Case Functionality | 1 | 1 | 0 | 0 | 00:00:35 | <div></div> |

Test Execution Log

| | |
|------------------------|--|
| SUITE | Aparat-Test-Case Functionality |
| Full Name: | Aparat-Test-Case Functionality |
| Source: | /home/yasaman/PycharmProjects/testproject/final/Aparat-Test-Case_Functionality.robot |
| Start / End / Elapsed: | 20240701 10:09:03.701 / 20240701 10:09:38.367 / 00:00:34.666 |
| Status: | 1 test total, 1 passed, 0 failed, 0 skipped |
| TEST | Login to Aparat |
| Full Name: | Aparat-Test-Case Functionality.Login to Aparat |
| Start / End / Elapsed: | 20240701 10:09:03.818 / 20240701 10:09:38.367 / 00:00:34.549 |
| Status: | PASS |
| KEYWORD | Aparat-Steps . Open Browser On Chrome |
| KEYWORD | Aparat-Steps . Click Login And Sign Up Button |
| KEYWORD | builtin . Sleep 10 |
| KEYWORD | SeleniumLibrary . Close Browser |

جمع‌بندی

در نهایت ما تست رابط کاربری را از سه منظر بر روی وبسایت آپارات انجام دادیم. ابتدا از منظر محتوا و دیزاین که شامل فونت و رنگ و نشان دادن محتوا بود بررسی کردیم. در این مرحله به علت نداشتن استاندارد های مشخص نوشتن تست کیس سخت بود و معیار مشخص و متناسبی نداشتیم. در قسمت دسترسی پذیری نیز بررسی می کنیم که آیا یک فعالیت در مرورگر های مختلف برقرار است یا خیر. در این قسمت به علت تنوع مرورگر ها هم در نوع و هم در ورژن کار سخت است. در مرحله آخر هم از لحاظ فانکشن و عملی ui را بررسی کردیم. در این مرحله چالش های ما برقراری ارتباط با دکمه ها و المنت های ui بود .