

BrowserStack Assignment

**Given the complexities of both Live and App Live, I've chosen to concentrate on App Live for this assignment.*

Table of Contents:

- 1) The Scene (Overview)
- 2) Goal & definition
- 3) Hypothesis of who is facing this problem and why do they need a solution?
- 4) Assumptions
- 5) How do you measure success?
- 6) How do our competitors solve this problem?
- 7) What is the solution? How would you change the existing JIRA integration to increase adoption?
- 8) Annexure

The Scene:

Testing mobile apps on real iOS and Android devices presents numerous challenges. Let's look at some key statistics below to understand why building a robust testing infra for the mobile apps is important.

Key Statistics [Source in Reference section]:

1. As of 2024, the global number of smartphone users stands at a staggering 4.88 billion (60.42% of the world's population) and forecasted to grow to 6 billion by 2028
2. There are 7.21B Smartphone subscriptions with (2.33 B having multiple smartphones).
3. 72.6% of internet users will access the web solely via their smartphones by 2025 (existing > 50%)

Recalling from my days as Test lead for Acrobat Android app – **Device fragmentation** was the biggest challenge due to different Permutations of manufacturers, models, OS versions, customized devices (Chromebook, Foldables, Wearable, etc.), customized OS, screen sizes/resolutions apart from many other critical factors like network, memory, hardware etc.

While **Emulators/ Simulators** are helpful for initial development – real devices are preferred to ensure their websites /applications render as intended so customers have a good user experience, regardless of the device you view it in.

Maintaining such a wide range of physical combinations is a cumbersome and costly process.

This is where BrowserStack comes to rescue. **BrowserStack's vision is to be the testing infrastructure for the internet.** They provides the teams instant access to a cloud platform that allows them to comprehensively test their websites & mobile apps for functionality, performance, and visual appeal so they can release bug-free software faster and at scale.

Problem Statement:

BrowserStack currently provides a feature to file a bug directly onto JIRA after capturing it while testing. Redesign the feature to increase its adoption for Live Product.

Goal: Increase the adoption of the Live Product

Definition of Adoption

For now, I have not covered for activation of the feature to log a bug in Jira as a problem due to limited information and scope of this assignment.

I have assumed Adoption as:

“The user has logged more than 2 bugs across more than 1 Sprint (2 week) cycle”.

Leading indicator: The user is logging bugs in Jira using our feature.

Hypothesis - who is facing this problem and why do they need a solution?

With online research (Reference: User voice section in the end) and interviewing few users, below is the user personas identified:

- **Testers / Quality Assurance** team – require efficient workflows for detailed bug reporting
- **Developers / DevOps** teams – to ensure bug free release
- **Product teams / Business Analysts** – to do validations and user acceptance testing

Target Customers: Medium & Large organization leaders (Engineering /QA managers)

End Goal: The user shall be able to focus on the unique reproducible bugs leading to their closure.

Normally SDLC Testing in today there are multiple scenarios where you expect customer to use the product. (Type of testing: UI Testing, Network Testing, Load testing, Accessibility testing, Localization testing, etc.)

Pain points:

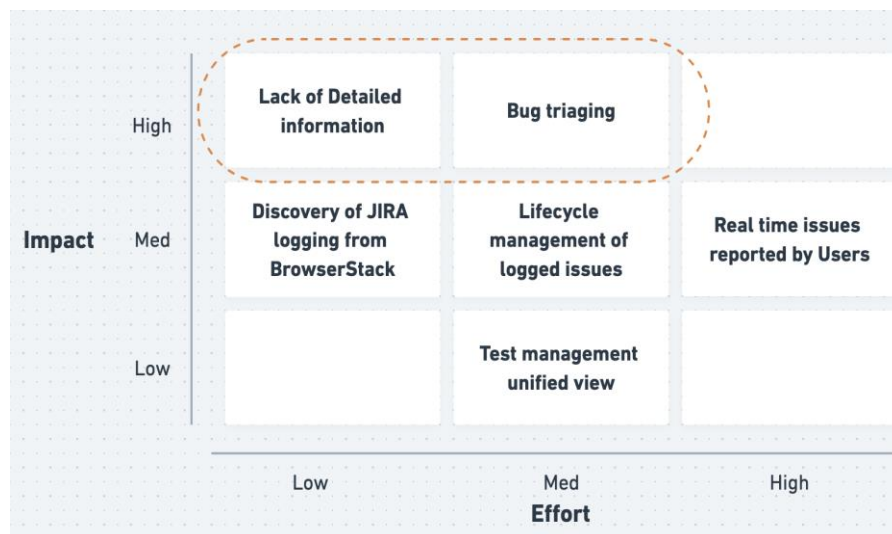
The users want to log issues with relevant info without losing the context and manage everything from the source itself.

These users need a solution because:

- 1) **Lack of Detailed information:** While the existing feature captures high level details like Device, Browser version, etc. with Screenshot – it doesn't support things like Video play, Console logs, Network log (Har file), etc. This can lead to misunderstandings, requiring further communication between QA and development teams.
- 2) **Bug triaging:** The biggest challenge is to triage bugs once it is reported (Both Manual & Automation). Duplicate bugs, Non Reproducible, Bugs re-opened regressions / existing in previous versions, etc. This makes the entire process cumbersome and time-consuming.
- 3) **Lifecycle management of logged issues in Jira:** Lack of Jira bugs view within the BrowserStack leads to Switching to & fro with JIRA which disrupts workflow, making it harder for users to maintain focus on testing and debugging tasks.
**Checkout page example – view my existing bugs
- 4) **Lack of awareness of the Jira functionality:** During the interviews one of the user talked about the unawareness of the feature (Refer Annexure). If our data indicates similar insights – we can work to educate the segment of users. We can validate with existing analytics data – user has already linked Jira account but not logged a bug in two weeks.
- 5) **Real time issues reported by Users:** Reported on feedback and support channels like Zendesk, Freshdesk or via app like Instabug – need to be groomed and fed into Jira.
- 6) **Test management:** A unified view of the test management end to end will be helpful. For example, User may start the testing with Test cases in Jira/ Test rail, then may log bugs linked with those test cases, Etc.

Prioritization:

With the current user insights (qualitative) and research: Lack of detailed information & 2) Bug Triaging is the top problems I would go ahead for prioritization.



Assumptions:

1. Looking at the complexities of the both Live and App Live - I have focused primarily on App Live for now in this assignment.
2. Given the assignment talks about increasing the adoption of the Jira feature, I have assumed this is bundled with our current packages. The increase in stickiness will allow us to serve our customers better.
3. The premium version of Bug Capture (Recent Birdeatsbug merger) still is designed primarily with Browser testing in mind (tried using extension). It needs customization for Mobile specific testing.
4. I have only focused on Manual testing scenarios as Live Product is primarily used for that (Automation is managed via AppAutomate).

Metrics:

- 1) **Number of bugs marked as needs more info/ Duplicate/ cannot be reproduced:**
A reduction in the number of bug resolution with above Resolution indicates better quality of bugs reported.
- 2) **Adoption Rate:** Increase in the number of users utilizing the JIRA integration feature within BrowserStack.
- 3) **Time to Report Bugs:** Reduction in the average time taken to report a bug from BrowserStack to JIRA.
- 4) **User Feedback:** Improvement in user satisfaction scores related to the bug reporting process, gathered through surveys and feedback forms. (CSAT)
- 5) Number of bugs logged by the users in frequency cycle (Traffic)
- 6) New integrations enabled in the BrowserStack Month over Month
- 7) Monthly Active Users to measure Stickiness

How do our competitors solve this problem?

Sauce Labs: It offers similar integration to that of the current by BrowserStack. However it by default captures video session information when you report a bug. Similar pain points were mentioned by the users of Sauce Labs in terms of Jira integration.

Test Sigma: Test Sigma doesn't offer a manual testing platform like Live. While speaking to one of the paid users (who have implanted it for Automation) – they shared that Jira integration exists for the Automation workflows with ability to capture details, etc.

PCloudy: The user experience of the testing tool felt similar to BrowserStack. The online reviews suggest that the real-time audio/video streaming during tests is used by the User. I was unable to test the Jira functionality as there were integration errors with Jira and Slack.

Perfecto: Perfecto doesn't have any Jira integration. It allowed user to mark the test as pass or fail and populate issue in their internal library view with ability to customize views.

LambdaTest: I was not able to access their devices in freemium version. However in their tutorial video – I found the experience exactly same as our current Jira experience.

JAM (Free extension) for Browser is also used by manual testing to record video with logs.

There are other Accelq, Katalon, Testim.io, etc. focussing on Automation instead of Manual.

What is the solution? How would you change the existing JIRA integration to increase adoption?

Solutions:

1) Configurable Bug reporting:

Different Sets – whichever type the users selects (Or auto populate based on the context) Select that information are captured.

- Workflow issues: Console logs capturing different Verbose, Debug, Info, Warning, Error, Assert etc.
- Network issues: List of api calls with Har files containing details of the log.
- Device State: adb or other relevant logs.
- User Telemetry: Verbose logs to make sure right Analytics calls are captured.

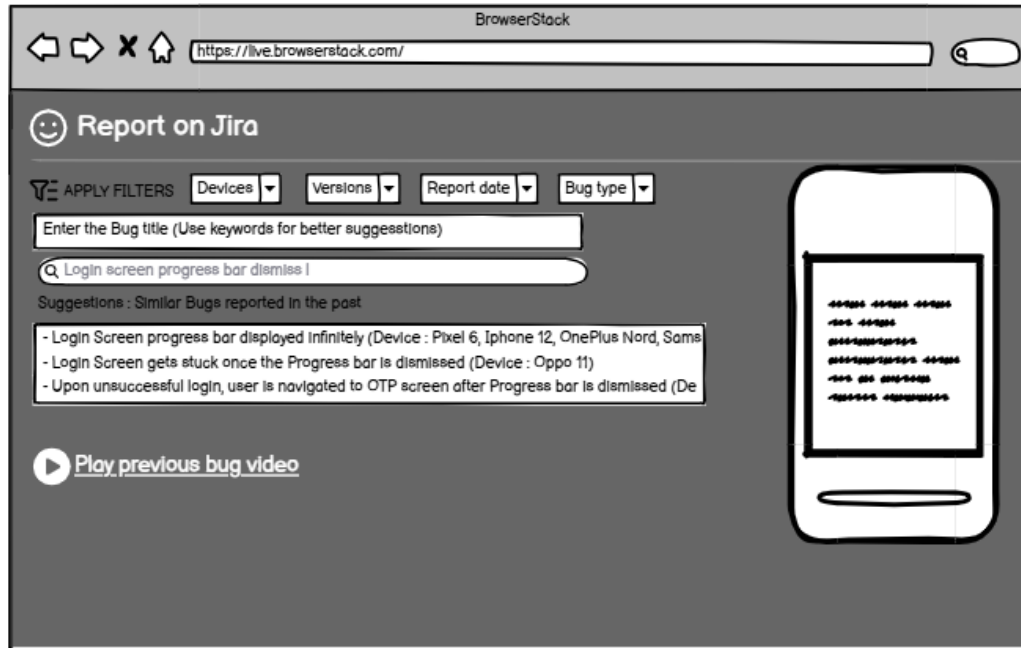
Using the synergies of BugCapture – we can enhance the existing workflows with more detailed information.

2) Smart Suggestions on already reported bug:

Multiple users testing or using the applications leads to lot of overlapping bugs.

When the users' tries to add a bug – suggestions should be provided from the previous reported bugs similar based on logs, title, issue type, device profile or a combination of both.

I have prepared a low fidelity wireframe for the Bug triage feature as suggested in the solutions list.



3) Unified view to show the BrowserStack reported bug with Smart Auto Triage feature

Using the historical information of the bugs reported via the platform – every time a new bug is added – BrowserStack can use generative AI solutions to triage the bugs as Duplicate, Regression, etc.

Annexure

User voice

As part of this exercise - I spoke to few of the users of BrowserStack & its competitors. Here's what they said:

"If we get an ability to triage our bugs – that would be crazy. The bugs logged by my team during the testing face – the biggest challenge for me to see how many of them are unique. Though workflows are different – at times the root cause is same."

- **Senior Engineering Manager, Quality Assurance at Sumologic, 13 years in Testing.**

"Can it provide integration with some user reporting tools like Instabug? The first step we do once such bug is logged by the user is to debug it if the logs are not sufficient. We used BrowserStack for few years before shifting to Saucelabs as Adobe went into B2B partnership with them."

- **Engineering Manager, Acrobat team at Adobe. She has 10+ years into testing.**

"I personally loved BrowserStack as a Product; we used to leverage its Network testing IP specific due to issues observed in different geographies".

- **Product lead, Dukes University. Part of testing team 5 years ago.**

“I used TestSigma and BrowserStack till last year – and was not aware that this JIRA functionality is there! Probably they can educate me about the tools. The number of tools & features has grown in past few years and it's hard to keep track of it”

- Moved to Program management, was Automation lead at Adobe in past role.

“The integration of Sauce labs is easy. Few years ago – when we have to choose for a solution – we went ahead with TestFairy (acq by SL later) due to lower pricing”

- DevOps Engineer, Sumologic.

“I believe Jira is the starting point of all Test journeys. I wish there was a way where I could integrate my BrowserStack into Jira. Whenever I run a Test case on one device – if I just select the list of next devices and BrowserStack does the testing for me and share the result.

Outside Jira, one of the pain points for me if there was a way to clone the test device to keep it ready before testing with samples apps, key accounts, test data etc. “

-Leads Localization testing team of 10 members at Adobe

References:

- <https://www.statista.com/forecasts/1143723/smartphone-users-in-the-world>
- <https://www.bankmycell.com/blog/how-many-phones-are-in-the-world>
- <https://www.cnn.com/2019/01/24/smartphones-72percent-of-people-will-use-only-mobile-for-internet-by-2025.html>

Other details:

1. Account used for BrowserStack: Gaurav.dubeyfb@gmail.com
2. Account used for Atlassian/ PCloudly / Sauce Labs: Gaurav.dubey@iiml.org