reporter.mozilla.org

Prepared for: Mike Morgan
Prepared by: Aakash Desai

February 5, 2010

Summary

Problem

Currently, reporter.mozilla.org only allows users to query for reports sent in by the Broken Website Reporter on Firefox. With the creation of the Website Problem Reporter on Mobile Firefox, the site needs to be updated to allow users to query those reports as well. To add on to that, I also want to take this opportunity to guide the website towards a larger overall goal that is directed towards consolidating all of our feedback outlets into this one search form. The end goal of this project is to make reporter.mozilla.org Mozilla Corporation's primary conduit in consolidating, aggregating and displaying user feedback for any of our products.
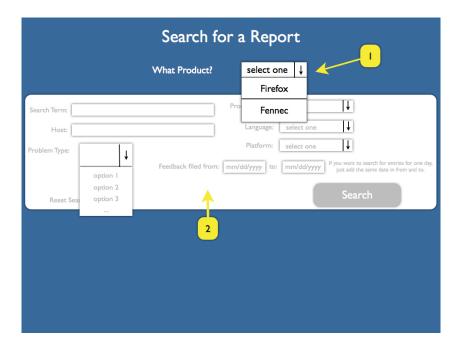
Goals

1. Fennec website problem reports are searchable on reporter.mozilla.org

2. The search form is intuitive and easy to use

3. Improve the aesthetics of the homepage, search results page and detailed report page
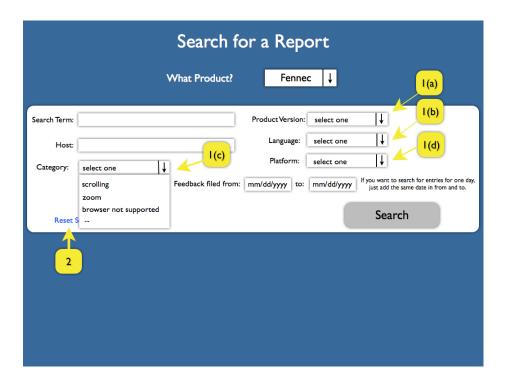
Implementation

Query Form

Pre Product-Selection

New Features

1. **Product selection menu box:** The menu box will open a list of product application names (i.e. Fennec, Firefox) for an entry in the database. The resulting actions should occur depending on the option selected:

   - **Firefox**: menu list options should be populated with choices that are filtered for the product string term, "Firefox" or any other internal project name related to Firefox (i.e. "Namoroka" and "Shiretoko").

   - **Fennec**: menu list options will be populated with choices that are filtered for the product string term, "Fennec".

2. **Grayed-out form area fields:** The input buttons/fields/menu boxes in the query form (i.e. Product Version, Language, Platform, Problem Type, etc.) will be grayed out and unusable.
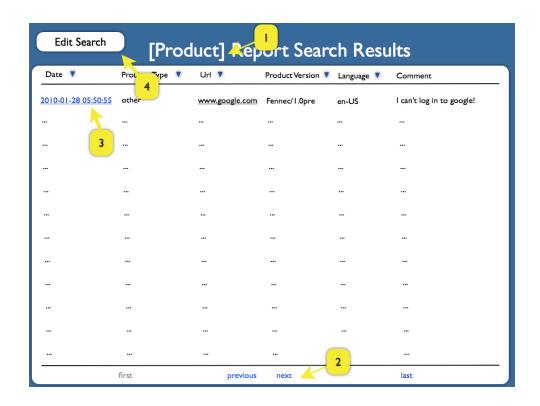
Post Product-Selection

New Features

1. **Product-based options boxes:** The menu list options in each list box should only show choices that are in the database subset that match the product string. If other options are selected, then the remaining un-selected fields should have their options sufficiently filtered further.

   a. **Product Version:** This menu list box will contain the set of product version strings

   b. **Language:** This menu list box will contain choices populated with language strings in the database subset.

   c. **Category:** This menu list box will contain the problem categories available for that product.

   d. **Platform:** This menu list box will contain the platform strings from the user agent that are in the database subset.

2. **Reset search hyperlink:** The link will clear all populated fields within the form when clicked. The product selection menu box should *not* be cleared when this hyperlink is clicked.
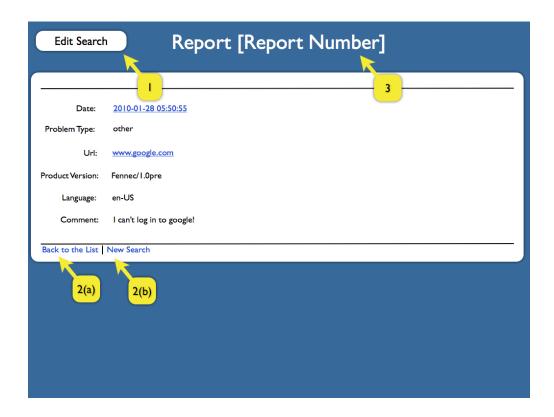
Search Results Page

New Features

1. **Auto-populated product name in title:** The title of the search results page should include the product name selected from the first page.

2. **New page parsing buttons:** The paging system won't include the page numbers, but four pagination hyperlinks named "first", "previous", "next" and "last".

3. **Report links in left-most column:** Hyperlinks to the report page for each entry should be shown in the left-most column selected. In this implementation, it's the date column.

4. **Edit search button on the top left part of the page:** A large button should be added on the top left section of the page. When clicked, it will load a populated search entry form. The populated entries should correspond to the entries submitted to view the search results page.

Report Page

New Features

1. **Edit search button on the top left part of the page:** A large button should be added on the top left section of the page. It will link to back to the search form, but with pre-populated fields shown in the current search results page.

2. **Hyperlinks on the bottom left of the report entry area**

   a. **Back to List:** The hyperlink will send the user back to its appropriate search results page

   b. **New Search:** The hyperlink will send the user back to a clear search form (i.e. product has not been selected).

3. **Auto-populate report number in title:** The title will include the database number associated with the report.

Roadmap

#	Feature	Description	Phase
2.1	Hendrix Database	Move the current Hendrix data from newsgroups to a query-able database	2.0
2.2	Widgets	Add statistic widgets that show pertinent information from aggregated reports in the database	2.0
3.1	Firefox Problem Categories	Change the 9 current problem categories on the Broken Website Reporter to more usable options on Firefox.	3.0
4.1	Litmus 2 Database	Make the database of Litmus 2 comments searchable through the query form.	4.0
4.2	User Accounts	Add an account system to allow repeat users to make and save their own widgets and searches.	4.0

2.1 - Hendrix Database

The feedback collected by Hendrix should be posted to a database and searchable via the reporter.mozilla.org interface. Hendrix newsgroups will need to be removed once all previously entered feedback has been moved over to the new database. All forms that currently send feedback to the newsgroups will need to be changed so as to send to the new location.

The database categories associated with each report will include the following:
- Product
- Feedback

2.2 - Widgets

Aggregated content should be shown as a graph, list and/or table. The initial plan is to include two default widgets onto the site after the product (i.e. Fennec, Firefox) has been selected. They are the following:

Reports vs. Time Graph
- A line graph that shows the search form viewer the number of feedback coming into the database over an undefined period of time.

Top 5 URLs Reported List
- A list of top aggregated domains for the selected product.

3.1 - Firefox Problem Categories

The problem types currently listed in the Report Broken Website option in Firefox will be changed to new categories that are easier to . Changes will need to be made to the strings in the feature in Firefox as well as reporter.mozilla.org. The reports already in the database, and are associated to removed categories, need to be changed to "other".

4.1 - Litmus 2 Database

Once the next revision of Litmus 2 is completed, the search form needs to include an option to search the database of comments in the manual test framework.

4.2 - User Accounts

An account system will need to be created to allow anyone from Project Managers to Community Members to create and save their own widgets and saved searches onto a home-like interface shown below the search form.