Automating Checking - Phase 1 - TodoMVC Test Plan

Objective

This test plan intends to outline the testing which will be performed to confirm that the TodoMVC website is deployable before it is given to customers later this week.

We have been provided with a feature list to be tested, which has been modified to make it as realistic as possible given the time constraints.

Overview

TodoMVC is a to-do list application doesn't have any persistent storage; it stores its data in the user's local storage. Consequently, the application is not designed to be a real-world to-do list manager, as there's no way to share the data between sessions, browsers or machines.

Scope

In scope

- we will be testing the feature list across the frameworks listed on TodoMVC
- The links on the page
- The tab system that filters JavaScript, Compile-toJS and Labs
- Mobile browsing experience
- General performance

Out of scope

- Security
- Large scale performance (such as high level stress testing)
- Sidebar links (Official Resources, Community, etc) for each template
- Keyboard navigation - not yet implemented

Team

We have 7 testers in our test team, however 4 of them are currently allocated to other projects which are due for release in the coming months. This leaves us with 3 team members for TodoMVC testing.

The testers assigned to this project are:

- `A.Adams` - Test Team Lead, the most experienced member of the team.
- `B.Barker` - Test Engineer who's been the primary tester on TodoMVC for the entire project.
- `C.Chapel` - Test Engineer who joined the company last week and is primarily still completing onboarding.

Targets for team

- Write automated testing for the chosen starter language (JavaScript, React)
- apply these tests to the other frameworks
- to be completed alongside exploratory testing
- submission of all test reports to Product owner stand-in

Day 1/4

- Write the automated tests for JS React in Chrome (Adams and Barker)
- Exploratory tests - free roam and then structured to the feature list (Chapel)
- Run these tests and make sure they are running correctly (ALL)

Day 2/4

- clear up any left over checks from day 1 tests
- modify/create the same tests for the other frameworks and browsers available (ALL staff)
    - Browsers - Chrome, Safari, Firefox, Edge
    - Frameworks - those listed on the MVC website - starting with JavaScript tab, then moving to the Compile-to-JS, then Labs as time allows
- Run all tests on all variants and list any areas of note
    - failed tests
    - bugs
    - etc
- Provide all gathered information and bug reports to the other team members - Product owner stand in, and devs

- This then allows the Product owner stand in to decide what to do with regards to instructing the devs and the deployment

Day 3

- Review tests (ALL)
- review any that came back with errors
- creation of documentation
- Chapel - to do exploratory testing with Mobile and write summary reports of experience

Day 4

- Any essential catchup(Ideally this is not the case)
- Test sign off report and submission (ALL)

Features to be tested

- Add a new ToDo item
    - Can't add an item with an empty value
    - Can add a value with a single character
    - Check that every single accented character and symbol is supported
    - Check that every single emoji character is supported
- Modify a ToDo item (by double-clicking)
- If you modify a ToDo item and click Escape during edit, it should cancel the modification
- A ToDo item can be ticked-off (it will appear with a line through it)
- A completed ToDo item can be unticked again
- A ToDo item can be reordered by dragging it up or down in the list
- Delete an incomplete ToDo item

- Delete a completed ToDo item

- Status bar displays "0 items left" when there are no items left

- Status bar displays "1 item left" when there is 1 item left

- Status bar displays "2 items left" when there are 2 items left

- Status bar displays "3 items left" when there are 3 items left

- Status bar is hidden when there are no ToDo items in the system

- ToDo items have a 248 Character limit

- When there are any completed items, a "Clear completed" link appears in the status bar

- When the "Clear completed" link is clicked, all completed items are deleted

- Status bar is hidden when there are no ToDo items in the system

- Clicking the down arrow symbol next to the "What needs to be done?" box will toggle all items to Completed or Not Completed

- ToDo items from one variant do not "bleed over" into other applications (e.g if you create a ToDo item in the React version, it is *not* visible in the Vue.js version)

- clicking on the links on the pages take you where expected

Assumptions

- The deadline for this project is 'the end of this week' which we are assuming to be 4 days of working time

- Testers will run and give severity and priority ratings to bugs found, but the decisions around development and deploying with or without fixes is not down to the testers
- We are assuming that with the Product owner being on holiday that there is someone in post other than the testers who are available to make decisions. They are fully up to speed and they are invested in the product

Defect management

Bugs should be reported with:

- a brief description
- steps to reproduce
- expectation vs actual
- priority/severity

High - Requires immediate attention
Medium - Ideally to be fixed before deployment (would be nice to be fixed by deployment)
Low - could be fixed after deployment

Exit criteria

Testing will come to an end when the testers have completed the planned activities above, it will then be over to the PO stand-in to determine launch

---

First 60 minute exploratory testing guide:

https://todomvc.com/

Things to look for:

- The features and general feel as listed below
- View and formatting
- general performance

Initial view:

□ Click through the site, providing narrative about how you think it works, what you think its aim is and what you think you should click to view the website

Features to test

- □ Add a new ToDo item
- □ Can't add an item with an empty value
- □ Can add a value with a single character
- □ Check that every single accented character and symbol is supported
- □ Check that every single emoji character is supported
- □ Modify a ToDo item (by double-clicking)
- □ if you modify a ToDo item and click Escape during edit, it should cancel the modification
- □ A ToDo item can be ticked-off (it will appear with a line through it)
- □ A completed ToDo item can be unticked again
- ☑ ~~A ToDo item can be reordered by dragging it up or down in the list~~
- □ Delete an incomplete ToDo item
- □ Delete a completed ToDo item
- □ Status bar displays "0 items left" when there are no items left
- □ Status bar displays "1 item left" when there is 1 item left
- □ Status bar displays "2 items left" when there are 2 items left
- □ Status bar displays "3 items left" when there are 3 items left
- □ Status bar is hidden when there are no ToDo items in the system
- □ Status bar can toggle between Active, All and Completed
- ☑ ~~ToDo items have a **248-character limit~~
- □ When there are any completed items, a "Clear completed" link appears in the status bar
- □ When the "Clear completed" link is clicked, all completed items are deleted
- □ When clicking on All in the status bar it shows all
- □ when clicking on active it shows all incompleted
- □ when clicking on completed it shows all completed
- □ Clicking the down arrow symbol next to the "What needs to be done?" box will toggle all items to Completed or Not Completed
- □ ToDo items from one variant do not "bleed over" into other applications (e.g if you create a ToDo item in the React version, it is *not* visible in the Vue.js version)
- □ clicking on the links on the front page take you where you would reasonably expect

Steps taken:

- discussion around the the site in general
- used 'mythril' (in JavaScript) as an example framework to run through the feature list (firefox)
- briefly looked at mobile version on firefox and safari
- Ran through the feature list on chrome with Elm (in Compile-to-JS)
- wrote up bugs and progress report for the exploratory testing

Bugs

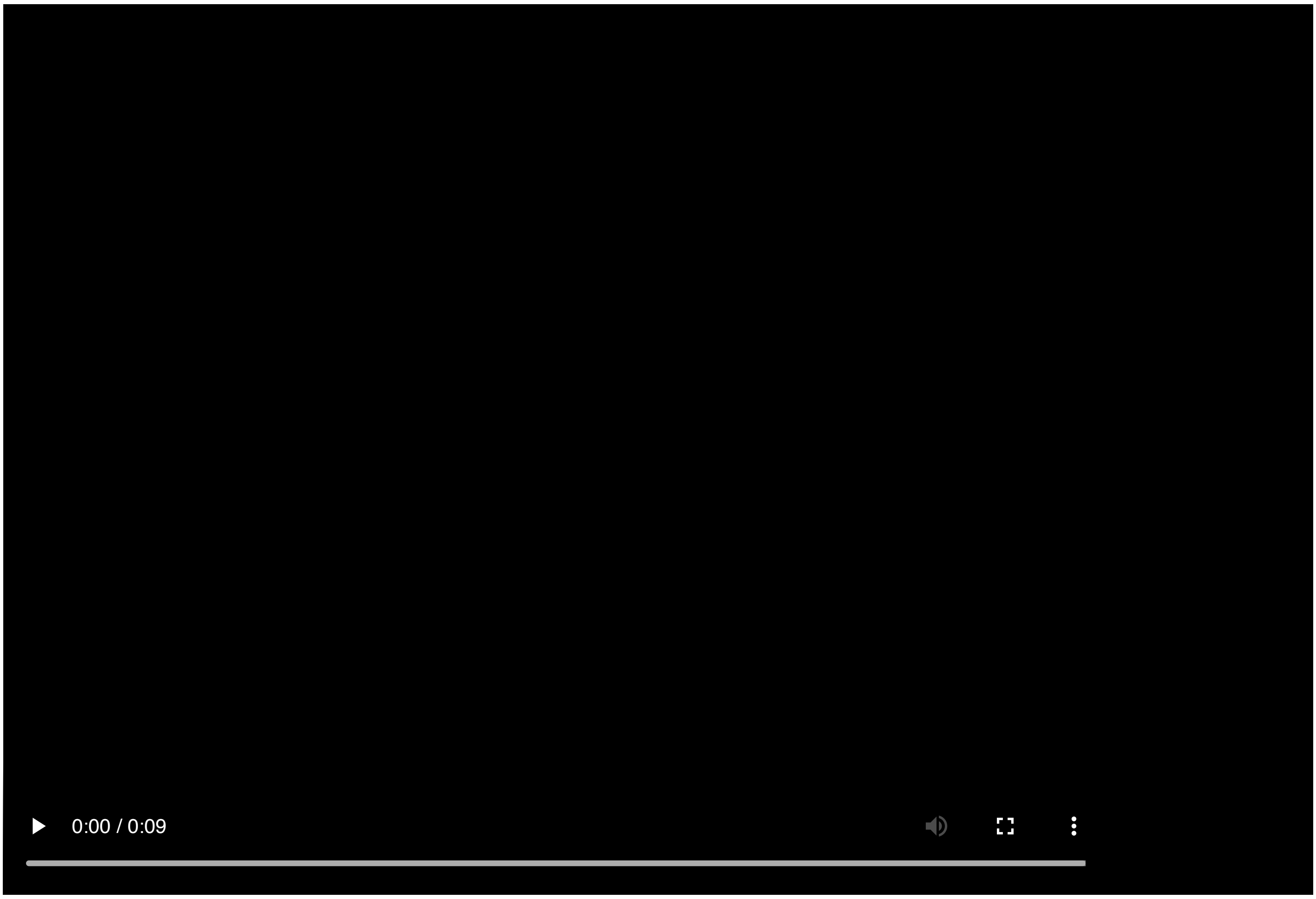
Description

Unable to 'reorder by dragging it up or down in the list'.

We were unable to re-order items

Steps to reproduce

1. In this case we were using Firefox and 'Mythril'
2. entered in the to do items as shown in the video
3. tried 'dragging to move' but was unable to re-order the items



Risk/priority

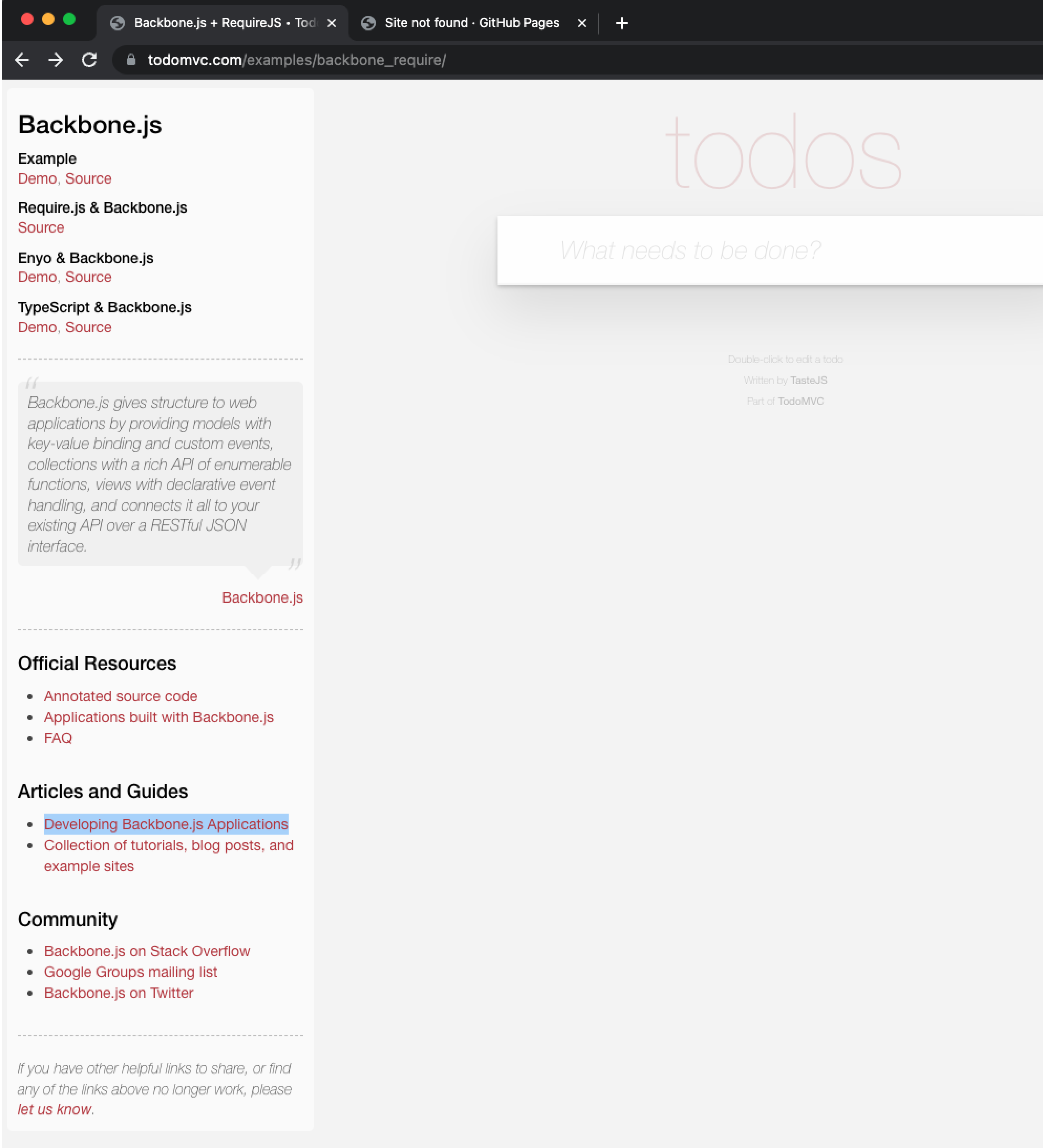
High - This is due to it being one of the specified features of the testing list - that is not working, as far as our exploratory test showed.
Would need time to retest before deployment

Description

Broken link is leading to a 404 page on the Backbone.js page

Steps to reproduce

1. go to start page
2. click on 'Backbone.js'
3. in the 'articles and guides' section on the left - click on 'Developing Backbone.js Applications'
4. This leads to a 404 page



Backbone.js + RequireJS • Tod
Site not found · GitHub Pages
todomvc.com/examples/backbone_require/

Backbone.js

Example
Demo, Source

Require.js & Backbone.js
Source

Enyo & Backbone.js
Demo, Source

TypeScript & Backbone.js
Demo, Source

> *Backbone.js gives structure to web applications by providing models with key-value binding and custom events, collections with a rich API of enumerable functions, views with declarative event handling, and connects it all to your existing API over a RESTful JSON interface.*

Backbone.js

Official Resources

- Annotated source code
- Applications built with Backbone.js
- FAQ

Articles and Guides

- Developing Backbone.js Applications
- Collection of tutorials, blog posts, and example sites

Community

- Backbone.js on Stack Overflow
- Google Groups mailing list
- Backbone.js on Twitter

If you have other helpful links to share, or find any of the links above no longer work, please let us know.

todos

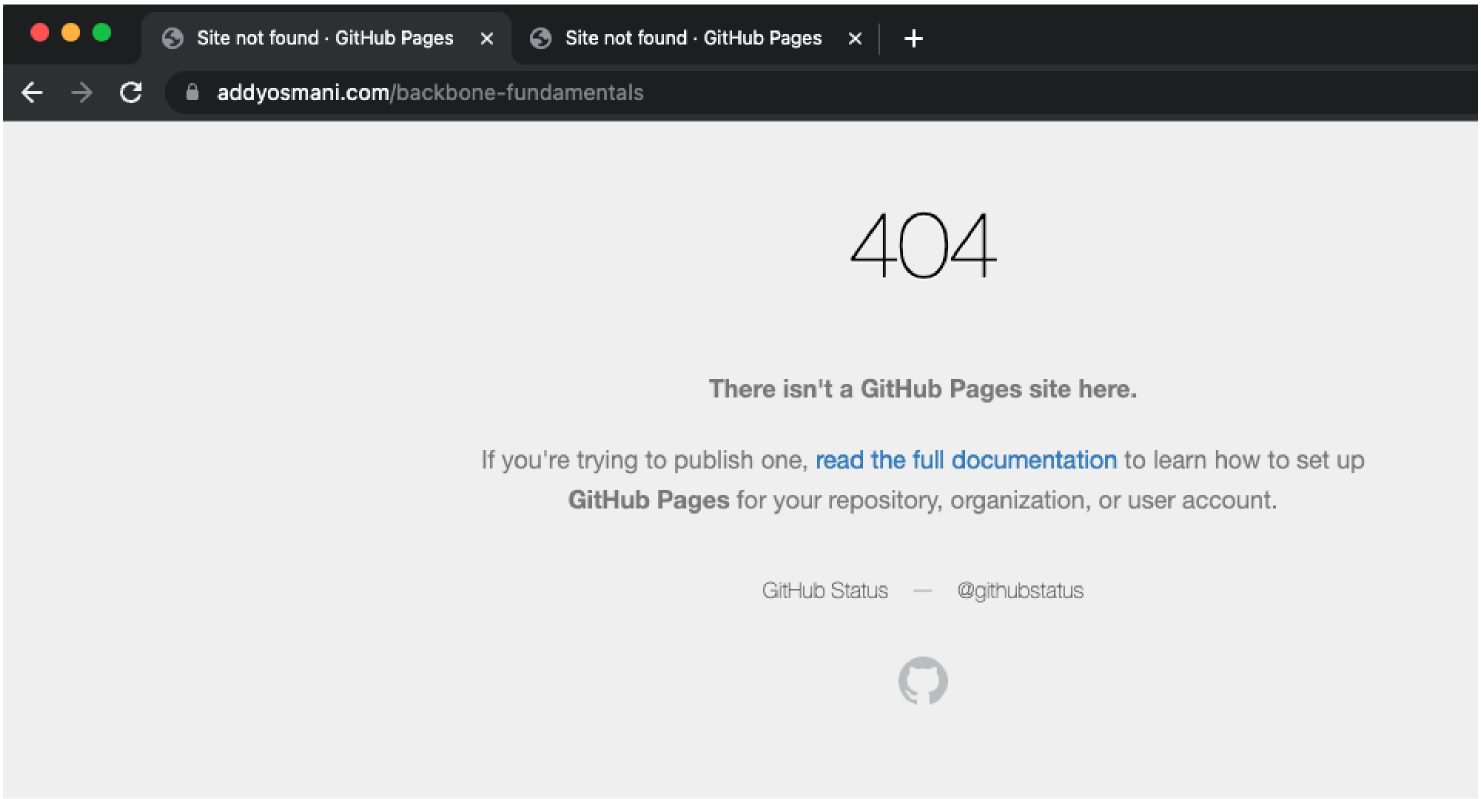
What needs to be done?

Double-click to edit a todo

Written by TasteJS

Part of TodoMVC

After clicked:



Site not found · GitHub Pages

addyosmani.com/backbone-fundamentals

404

There isn't a GitHub Pages site here.

If you're trying to publish one, read the full documentation to learn how to set up GitHub Pages for your repository, organization, or user account.

Risk/priority

Low - These links' destinations are not under our control. Ideally, we would check this at some point, but we do not believe this would be within scope given our deadline.

Extra Observations

- No way to add an item with mouse
- Mobile page loads as a 'desktop' version of the site (using safari and Firefox tested)
    - can still add items using keyboard
- No clear 'navigate back' button when within a framework - had to use browser buttons
- Note that the site allowed entries of well over 248 characters. for example, we copy pasted the whole of the wiki article on WW2 and it accepted it as one item
- In general the format and the view is good - all clear, no obvious out of place formatting
- The general performance was good - pages loaded fine, not lagging or delays in using the frameworks we tested on the site
- the features we tested worked as described - other than those listed and/or highlighted in the 'Bug' section.