|  |
| --- |
| Edinburgh Napier University |
| Advanced Web Technologies (Set09103) |
| Coursework 2 Report: The breakup app |
|  |
| **40071697 - Amantle Craviolatti** |
| **11/29/2016** |

|  |
| --- |
|  |

Table of Contents

[Introduction 2](#_Toc469460033)

[Design 3](#_Toc469460034)

[Enhancements 6](#_Toc469460035)

[Evaluation 7](#_Toc469460036)

[Appendices 8](#_Toc469460037)

[References 11](#_Toc469460038)

[Additional Resources 11](#_Toc469460039)

# Introduction

The web application developed for the second coursework is The Break-Up App, a chat application and general information toolkit for users undergoing the difficult process of getting over the termination of a romantic relationship. The application is inspired in functionality by websites such as Slack, HipChat, Quora and Twitter- as well as a recent personal breakup. The main feature, the chat room is especially designed to allow users a space to anonymously discuss their emotional state without recording the discussions evidencing a difficult time. Additional features include curated links to online articles for advice, as well as a custom google search, crawling in particular Quora, as well as Google at large; this feature is also designed to benefit users by ensuring no history is kept of searches related to the painful event, aiding online privacy, as well as emotional healing in the manner of the temporary chat space.

The python based application uses Bootstrap, Flask-SocketIO, Jinja2, Html, CSS and Google Custom Search API to provide this functionality.

This report outlines the development process for this web application from design to implementation including evaluations of the development journey and final product as well as recommendations for possible future enhancements

# Design

The design for this application began first with personal and also general observations of how technology, especially the internet, caters to the emotional needs of people undergoing the healing process following a romantic break-up. Through Google searches researching methods to understand and quicken the healing process post break up, it became clear that there were many resources available in the form of advice forums and articles [[1]](#_References). These technologies presented advantages in the way that they allowed for users to discuss their emotional state with other going through similar situations, however these discussions tended to be recorded online for years after the event, granting a degree of permanency to the emotional turmoil being discussed that could be considered unfavourable due to unpleasantness of what is otherwise an impermanent state. Advice articles also, while being useful and easily sourced, lessened their impact due to the many different types of advice available from too many sources, thus there was an element where this method of acquiring support could be improved through curation.

Finally, as with the problem with the forum boards, Google itself proved disadvantageous to promoting healing through the search engine store of searches when logged into a particular account. Although an existing method of going around this functionality in the technology would be to conduct searches linked to the conditions of the breakup via incognito or use of search engines such as duck, duck go; the application was an opportunity to present a median solution that did not force users to alter their normal search behaviours

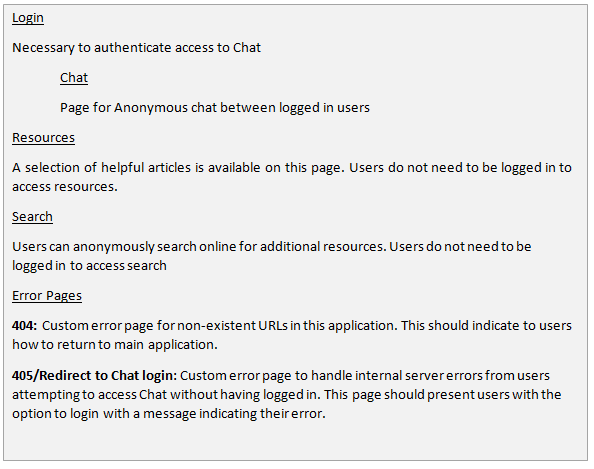
Having established the initial idea for the application the implementation of the functionality needed to be clarified. To do this, research took place on various websites, of which some elements informed the design of components of the application:

Table 1: Design inspiration for application functionality

|  |  |  |
| --- | --- | --- |
| Website/resource investigated | Element of interest | Component emulation within Break up App |
| Quora:  A question and answer site moderated by logged in users. [[2]](#_References) | Ability to have intelligent discourse with like-minded users- even when asking questions anonymously. | Chat with logged in users |
| HipChat:  Group and private instant messaging for business [[3]](#_References) | Free version deletes chat history after a few weeks | Short term save of chat |
| Slack:  Team Collaboration and Chat [[4]](#_References) | Overall “look” within chat rooms | Appearance and login |

Once these elements were identified, work began on creating a suitable URL structure for the application through paper design mock-ups [(see Appendices](#_Appendices)) mimicking the navigation of the application. The following structure was selected during this stage of Design:

Figure 1: URL Hierarchy for the Break-up App



Digital prototyping of the paper mock-ups followed the identification of the chat functionality as the core and the most complex feature of the app, the focus of these was on implementing a robust URL hierarchy within the app in order to support the chat feature. The creation of the hierarchy also included the addition of static content on the “resources” page allowing for early detection of problem links and custom error pages. In particular, cases in which a user could try to access the chat function without having signed in where designed to redirect the user to the sign in page with a JavaScript custom error message.

It was also at this stage that the decision was made to use a customised CSS Bootstrap theme [[5] [6]](#_References) to handle the aesthetic aspects of the application in order to allow more time for development of the application. In line with the purpose of the app, the theme was similarly simple and played on the idea that the user may have been feeling “blue” through the use of an eggshell blue as the central colour.

Following the digital prototyping of the basic elements of the application, research began on how to implement chat within the application and the addition of the custom search.

The use of sessions and temporary cookies integrated with a SQLite database [[7]](#_References) was investigated initially to allow “chatting” however this method risked impacting the performance once multiple users began to use the application. The need for cookies also detracted from the idea of “interacting within a void”, as the evidence of the application would remain on user’s devices.

A search for “Flask Chat API” revealed the Flask SocketIO API as the most suitable for lightweight integration with the application. This was in large part due to the fact that the package interfaces directly between web sockets, handling all data temporarily on the server before being displayed via the client and automatically erasing interactions upon closing the connections.[[8]](#_References)

For the custom search function, initial experimentation took place with using HTTP Post methods to send data to Google. This however was slow and at times returned errors, leading the design to change to include a custom Google search with the API included via JavaScript within the search page.[[9]](#_References)

Upon completion of the research for the chat and custom functionality, findings were then implemented alongside improved iterations of the URL hierarchy established in the digital Prototyping stage.

# Enhancements

The current version of the Break-up App leaves much room to improve on existing features. The most feasible enhancements within the current context of the application – namely as that of a 3 tool toolkit to aid break up recovery- are listed below with notes for how these improvements might be implemented.

Table 2: Possible enhancements for application

|  |  |
| --- | --- |
| Enhancements | Notes |
| Option to download chat conversation |  |
| Temporary “posting” to Chat page if no other users are logged in to the application. These posts would then be automatically deleted once a second user has logged in and had a chance to read the missed conversation. |  |
| Tailoring of resources according to the number of weeks since break-up | Due to time constraints this feature was not implemented during this iteration of the application. However the framework exists for this enhancement and this could easily be implemented using a Jinja2 filter for user input based on how many weeks the user entered in the login form. |
| Logout notification when leaving chat | Attempts were made within the current iteration of the application to implement a simple button triggering a JavaScript alert before returning the user to the main page; however this led the application to crash. In future implementations this could easily be included following work to improve the robustness of the server. |
| Asynchronous threading for all actions relating to Chat |  |

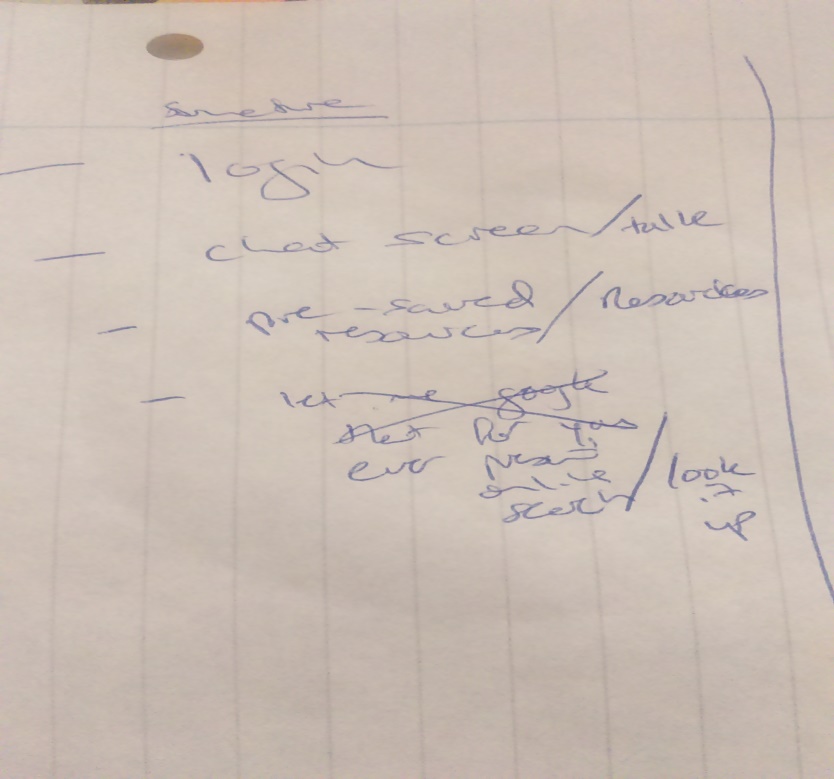
# 

# Evaluation

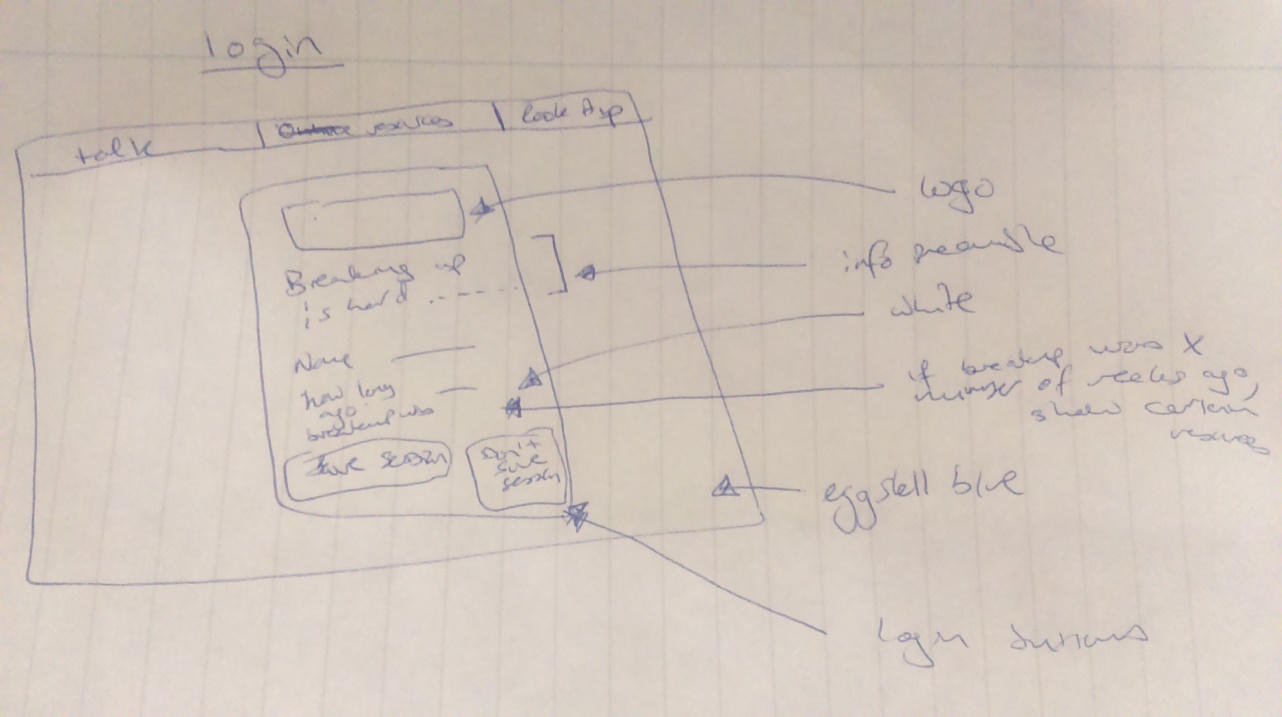
The application generally is fit for purpose having met the requirements of the idea to provide a basic toolkit for users coping with the effects of a relationship breakup. However all the features could benefit from further work in future iterations of the application; in particular the overall performance of the application is a current weakness due to the combination of APIs in use. This is reflected in slow load speeds between pages and occasional crashing of the server when the chat page is being repeatedly refreshed by a user. Such occurrences currently lower the viability of the application being accessed outside of a test environment. However the Installation packages such as Eventlet or Gevent and introduction of mirror servers or load bearers to manage activity on Chat pages to limit users online would potentially eliminate performance issues altogether. Security within the application could also be improved through the change of the current Security Key used for web socket connections to a more complex password.

# Appendices

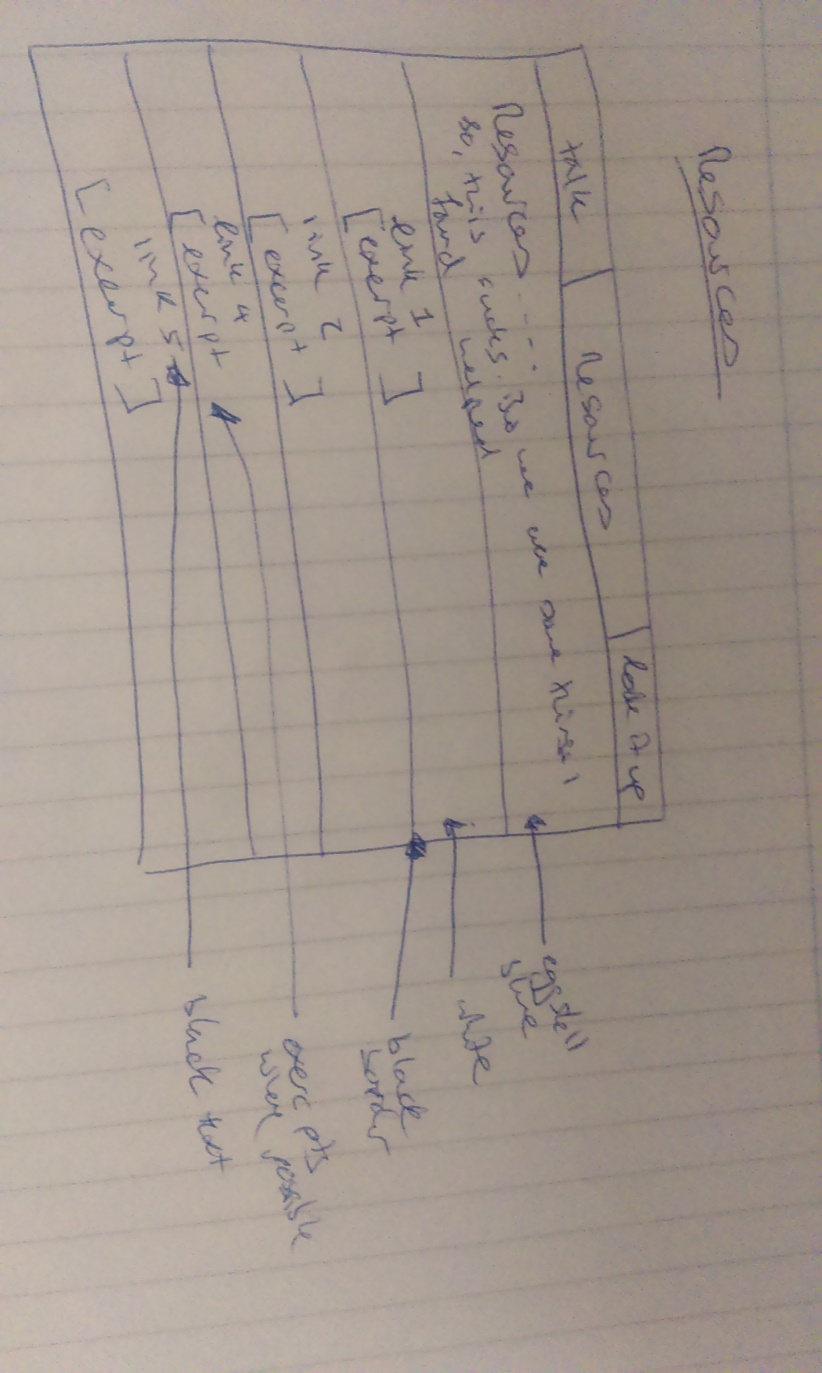
Early Sketch of URL Structure



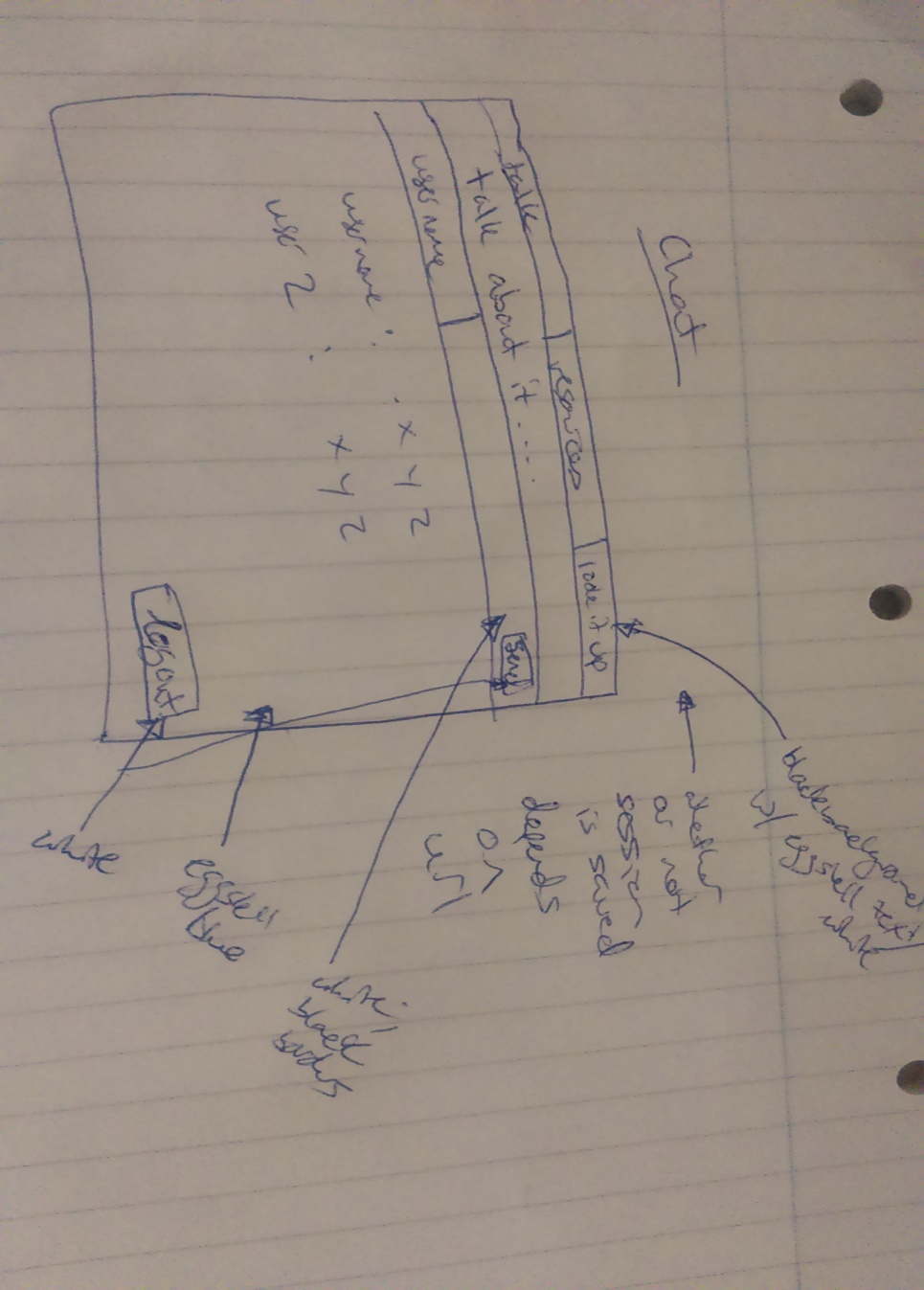
Early Mock-up of Login Page



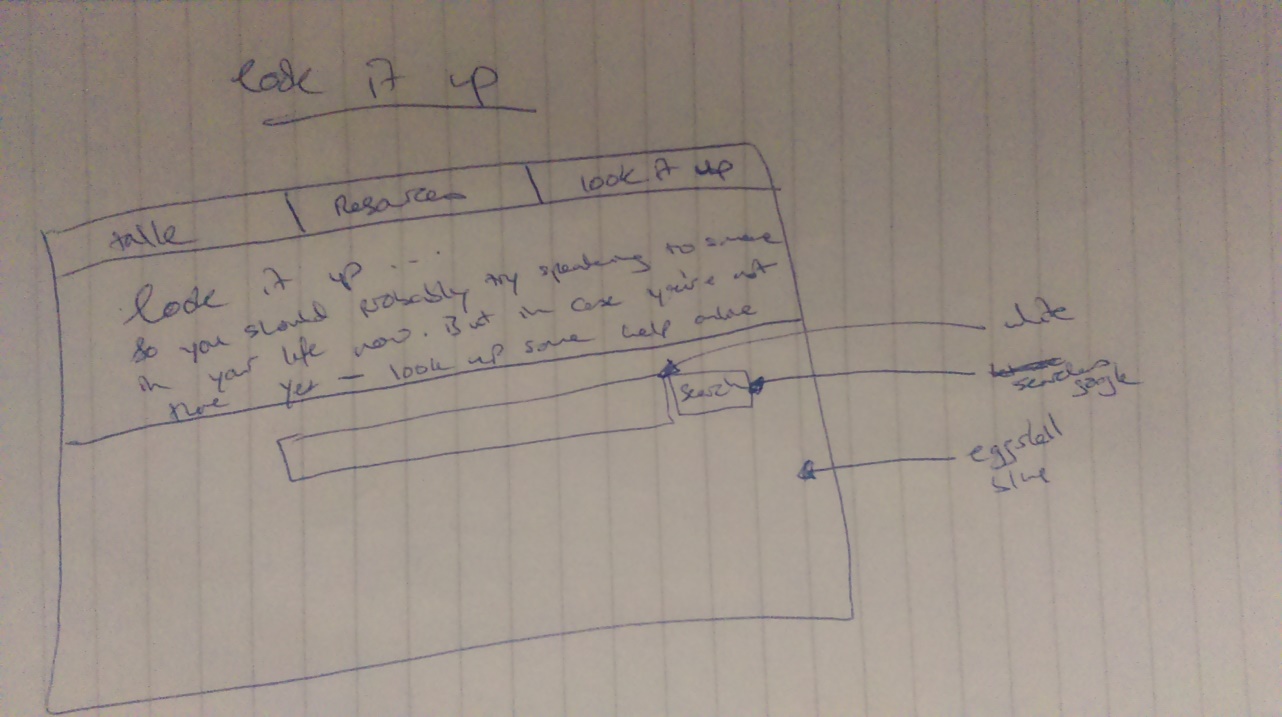
Mock-up of Resources page



Early mock-up of Chat page



Mock-up of search page



# References

[1] *Google*. (2016). *Google.co.uk*. <https://www.google.co.uk/webhp?sourceid=chrome-instant&ion=1&espv=2&ie=UTF-8#q=how%20to%20get%20over%20a%20break%20up>

[2] *About Quora - Quora*. (2016). *Quora*. <https://www.quora.com/about>

[3] *HipChat | Team Group Chat, Video Chat & Screen Sharing*. (2016). *HipChat* <https://www.hipchat.com/>

[4] *Slack: Be less busy*. (2016). *Slack*. <https://slack.com/>

[5] *SET09103 Advanced Web Technologies: Notes & Workbook 2015-2016*. (2016) (1st ed., pp. 65-70). Edinburgh. <https://www.dropbox.com/s/k41vw5a49y64nt7/workbook.pdf?dl=1>

[6] *Bootstrap Magic : Generate your own bootstrap theme quickly and easily*. (2016). *Pikock.github.io*. <https://pikock.github.io/bootstrap-magic/app/index.html#!/editor>

[7] *SET09103 Advanced Web Technologies: Notes & Workbook 2015-2016*. (2016) (1st ed., pp. 71-75). Edinburgh. <https://www.dropbox.com/s/k41vw5a49y64nt7/workbook.pdf?dl=1>

[8] *Welcome to Flask-SocketIO’s documentation! — Flask-SocketIO documentation*. (2016). *Flask-socketio.readthedocs.io*. <http://flask-socketio.readthedocs.io/en/latest/>

[9] *Implementing search box|Custom Search |Google Developers*. (2016). *Google Developers*. <https://developers.google.com/custom-search/docs/tutorial/implementingsearchbox>

# Additional Resources

**https://www.google.co.uk/webhp?sourceid=chrome-instant&ion=1&espv=2&ie=UTF-8#safe=strict&q=when+you+broke+up+with+him+**

**http://www.lovepanky.com/love-couch/your-ex/how-to-survive-the-first-168-hours-after-a-break-up**

**https://www.enotalone.com/relationships/4472.html**

**http://thoughtcatalog.com/kim-quindlen/2015/09/how-to-survive-the-first-7-days-of-a-breakup-even-just-barely/**

**http://lovesagame.com/what-i-wish-someone-had-told-me-right-after-my-breakup/**

**https://www.bustle.com/articles/100753-6-signs-its-time-to-break-up-with-someone-even-youre-still-in-love-with-them**

**https://www.buzzfeed.com/annaborges/dump-them?utm\_term=.mpn0N2bM6#.yuqzlVvyk**

**http://www.thefrisky.com/2008-11-11/how-to-survive-the-first-30-days-of-a-breakup/**

**https://www.psychologytoday.com/blog/valley-girl-brain/201209/7-phrases-will-help-you-get-over-breakup**

**http://www.livestrong.com/article/185635-how-to-get-over-a-guy-you-still-love/**

**https://en.wikipedia.org/wiki/Electron\_(software\_framework)**

**https://en.wikipedia.org/wiki/Bootstrap\_(front-end\_framework)**

**https://webtech-hq.slack.com/messages/@slackbot/**

**https://www.buzzfeed.com/annaborges/dump-them?utm\_term=.rp5voN3L4#.hxdDdwEJ5**

**http://www.w3schools.com/js/tryit.asp?filename=tryjs\_confirm**

**http://www.w3schools.com/js/tryit.asp?filename=tryjs\_alert**

**http://www.w3schools.com/js/js\_popup.asp**

**http://stackoverflow.com/questions/24254945/internal-server-error-flask**

**http://jsfiddle.net/gh/gist/library/pure/6130833/**

**http://eventlet.net/**

**https://github.com/miguelgrinberg/Flask-SocketIO/issues/153**

**http://www.w3schools.com/tags/att\_form\_method.asp**

**https://blog.miguelgrinberg.com/post/easy-websockets-with-flask-and-gevent**

**https://developers.google.com/custom-search/docs/tutorial/implementingsearchbox**

**https://blog.ochronus.com/what-is-a-http-head-request-good-for-some-uses-ffefc7b78024#.dxpyiuxti**

**http://stackoverflow.com/questions/14792605/python-flask-import-error**

**https://www.youtube.com/watch?v=RdSrkkrj3l4**