Bixly Blog Intro Project

The Rules:

- The user may not use any pre-made django app not distributed with the main django bundle.

The Tickets

Complete these in order, with documentation for us to examine.

1. Create a basic blog. Visiting the root URL should display all entries, with a title, an entry body, and a date associated with every post. Entries can be created in the django admin.

2. Give your blog some style. Use HTML5 elements (look up the <article> tag, for example) and CSS. You may find and use HTML boilerplate if you wish.

3. Allow users to filter which entries they're viewing by year or year+month

4. Allow pagination of your blog. Each page should show a "previous" and "next" link which allows the user to view groups of 3 entries at a time. Pagination should apply to the homepage, as well as the pages where the user can filter by dates. You should list which page the user is currently on and also how many total pages there are. If there is no "next" or "previous" page, either don't display the button or grey it out. (hint: look at django's built-in pagination support https://docs.djangoproject.com/en/dev/topics/pagination/)

5. Allow the author to create a blog entry outside the django admin. This means that the django admin should not be necessary to update the blog at all. (bonus: use a javascript WYSIWYG markdown editor).

6. Allow entries to have multiple "tags" associated with them. For example, if I write a post about cooking cajun style chicken, I should be able to tag it with "chicken", "cooking", and "cajun", I should then be able to click on one of the tags, and view all posts that share that tag. (hint: you will probably want to use django's many-to-many model type to accomplish this https://docs.djangoproject.com/en/dev/topics/db/models/#many-to-

many-relationships)

7. Allow users to comment on entries. Users should need to type in their name and their comment. (bonus: require reCAPTCHA to verify the user is not a spambot) (You may use django's built-in comment app, https://docs.djangoproject.com/en/dev/ref/contrib/comments/)

8. Allow multiple authors to create blog posts. This means that the blog author should be able to login to the blog and when he or she creates an entry, that entry will automatically display the author's username in addition to the blog title and date. (look at the built-in django user authentication https://docs.djangoproject.com/en/dev/topics/auth/)

9. Support markdown. The blog author should be able to write their entries in the markdown syntax, and have their entries be formatted according to the markdown spec. (note, you may not use django-markdown to accomplish this, though you may use a python markdown library such as this one https://github.com/trentm/python-markdown2)

10.   Build a search tool. Users should be able to type in a search query on the homepage and the blog should search through the title and entry text of all blog entries, and display the entries that match (Note, you will probably need to use mysql to accomplish this https://docs.djangoproject.com/en/dev/ref/models/querysets/#search).