
Biffy: The Batteries-Included Flask-based Framework Documentation

Release 0.0.1

Andrew Sledge

December 16, 2013

Contents

The Batteries-Included Flask-based Framework.

Biffy is a web framework based on Flask, good intentions, but mostly laziness. The desire here is to create a boilerplate-esque application that will handle pretty much the base line requirements of any modern day web application. This includes, but is not limited to:

- User authentication with social services (Google, Facebook, Twitter)
- Posting of application updates to social accounts
- Clean look and feel (courtesy of Twitter Bootstrap)
- Easy to develop and extend
- Mobile ready (responsive design)
- An easy to use platform for even the novice developer

Powered by beef.

Contents:

Design Philosophy

The idea is to provide a way for quick deployment for Flask applications, and to provide out-of-the-box functionality for many use cases. With the same code base you can deploy many different applications.

Every developer wishes that they could have that magic bullet framework that would solve most of their problems. Unfortunately that magic bullet doesn't and probably will never exist, but we can try to get close. The spirit of Biffy is that frameworks should be modular, come with tons of features, and (most important) easy to use.

1.1 Modular

Flask implements modularity in the form of [Blueprints](#). Biffy takes this concept and bolts on the following capabilities:

- Plug and play according to what you need at deployment
- Blueprints can advertise and offer services to other blueprints
- Consistent layout for ease of development

1.2 Tons of Features

This is the “Batteries-Included” part of Biffy. A good web framework should be able to get you up and running with some sort of base functionality right out of the box. Planned blueprints include a blog, a contact form, a twitter client, a facebook client, a calendar, and maybe a few more.

1.3 Easy to Use: Extending

It should be simple to add to the Biffy base without a lot of work and it should be able to be done very quickly. Biffy comes with blueprint creator.

```
$ biffy-create-blueprint.py my_custom_blueprint
```

1.4 Easy to Use: Deployment

Deployment can be a frustrating task. Biffy wants to make that easy for you, and is best demonstrated by an example:

```
$ biffy-create-app.py blog --blueprints blog,contact
```

This command tells biffy to create an application called `blog` with the blueprints `blog` and `contact` form. Now let's take a look at something a little more complex.

```
$ biffy-create-app.py blog --blueprints blog,contact --database-config  
database.ini
```

So suppose you want to use a boilerplate configuration because you have several sites that will look exactly the same. That is made easy by

```
$ biffy-create-app.py blog --biffy-recipe recipe.ini
```

So right now you're probably thinking that the goal is to reinvent [Django](#). Well, maybe a little bit. The Flask internals are so much more lightweight and if you know anything about Django's history it's that the ORM hasn't always been as portable or flexible (though, it is getting better). This project seeks to fulfill those needs and more.

Included Modules

This is the “Batteries-Included” part of Biffy. A good web framework should be able to get you up and running with some sort of base functionality right out of the box. Planned blueprints include a blog, a contact form, a twitter client, a facebook client, a calendar, and maybe a few more.

2.1 Users

More information coming soon.

2.2 Admin

More information coming soon.

2.3 Blog

More information coming soon.

2.4 Contact

More information coming soon.

2.5 Calendar

More information coming soon.

Indices and tables

- *genindex*
- *modindex*
- *search*