To create a Django web application with login and signup functionality using Google, Facebook, and GitHub, you can follow these general steps:

1. Install the necessary packages:
   - django-allauth: This package provides easy integration with social authentication providers, including Google, Facebook, and GitHub. Install it using pip:
   ```
   pip install django-allauth
   ```

   - social-auth-app-django: This package provides additional integration with social authentication providers. Install it using pip:
   ```
   pip install social-auth-app-django
   ```

2. Add `allauth` and `social_django` to your `INSTALLED_APPS` in your `settings.py` file:
   ```python
   INSTALLED_APPS = [
       # other apps
       'django.contrib.sites',
       'allauth',
       'allauth.account',
       'allauth.socialaccount',
       'allauth.socialaccount.providers.google',
       'allauth.socialaccount.providers.facebook',
       'allauth.socialaccount.providers.github',
       'social_django',
   ]
   ```

3. Add `sites` framework to your `MIDDLEWARE` in your `settings.py` file:
   ```python
   MIDDLEWARE = [
       # other middleware
       'django.contrib.sites.middleware.CurrentSiteMiddleware',
   ]
   ```

4. Add the social authentication provider details to your `settings.py` file. You will need to obtain client IDs and secrets for each provider from their respective developer portals:
   ```python
   SOCIALACCOUNT_PROVIDERS = {
       'google': {
           'APP': {

```python
                    'client_id': 'YOUR_GOOGLE_CLIENT_ID',
                    'secret': 'YOUR_GOOGLE_CLIENT_SECRET',
                    'key': '',
                },
                'SCOPE': [
                    'profile',
                    'email',
                ],
                'AUTH_PARAMS': {
                    'access_type': 'online',
                }
            },
            'facebook': {
                'APP': {
                    'client_id': 'YOUR_FACEBOOK_CLIENT_ID',
                    'secret': 'YOUR_FACEBOOK_CLIENT_SECRET',
                    'key': '',
                },
                'SCOPE': [
                    'email',
                    'public_profile',
                ],
                'METHOD': 'oauth2',
                'LOCALE_FUNC': lambda request: 'en_US',
                'VERIFIED_EMAIL': False,
                'VERSION': 'v10.0',
            },
            'github': {
                'APP': {
                    'client_id': 'YOUR_GITHUB_CLIENT_ID',
                    'secret': 'YOUR_GITHUB_CLIENT_SECRET',
                    'key': '',
                },
                'SCOPE': [
                    'user',
                    'user:email',
                ],
            }
        }
```

5. Add the social authentication URLs to your project's `urls.py` file:
   ```python
   from django.urls import path, include
```

```
from allauth.account.views import LoginView, SignupView

urlpatterns = [
    # other URLs
    path('accounts/', include('allauth.urls')),
    path('accounts/login/', LoginView.as_view(), name='account_login'),
    path('accounts/signup/', SignupView.as_view(), name='account_signup'),
]
```

6. Customize the templates used for login and signup:
   - Copy the `account` folder from `allauth/templates` to your project's templates directory.
   - Create new files called `login.html` and `signup.html` inside the `account` folder.
   - Customize the templates as needed, including the social authentication buttons.

That's it! Your Django web application now has login and signup functionality with social authentication using Google, Facebook, and GitHub.