

# Building a Web App with Django

Catherine Badart

# Experience

- Summer internship at Microsoft on the SQL Server on Linux team
- Built a web app for customer engagement
- Django's built in libraries and applications made it easy to make a sophisticated app quickly

# REST

- Use [Django REST Framework](#) to build an API
- Useful for Authentication
- Made development easier for us and people that work on our app down the line

# MVT

- Model: database model/schema, defines structure of your data
- View (same as Controller in MVC): logic behind rendering a page/processing a user's request
- Template (same as View in MVC): UI layer that the user sees. Use `{% data %}` to display user specific information

# Features

- Good Documentation
- Django Applications: don't reinvent the wheel every time
  - Do a small task to make development easier
  - Just pip install them
  - Used the following in my project:
    - django-threadedcomments (threaded comments on posts)
    - django-widget-tweaks (makes forms pretty)
    - django-notifications (email notifications)
    - django-filter (filter query results)
- Django Classes: built in and can define your own
  - Form class makes it easy to put form data in your database
  - Define a User class with attributes you want to save for each user
- Built in user authentication