March 18, 2015

A01952442

Adam Tew

# Final Proposal

### Basic Description:

This application will be an eCommerce website that sells kittens using Django. There will be an SQL database that dynamically feeds data to each page. Although there will also be a lot of HTML, CSS, and JavaScript to help with the front-end functionality, Django will handle all of the backend and dynamics. There will also be session scope components that have to be taken care of, namely login, keeping track of items selected to purchase, and checking out.

This website will consist of eight web pages: index, about, home, contact us, item categories, item details, login, and checkout. All of the data will be stored inside a database which will then be queried by Django to display on the pages. Django will also handle the session scope.

Primarily, the use of Django will handle the backend dynamics, including database queries, handling logging current customer information, validating purchases, and then pushing appropriate information back to the database.

### Target User:

The target audience for this application will be anyone who can browse internet pages and has a desire to purchase or look into purchasing cats. Because this application will be web-based, the idea is that it this application will be build able to run on any machine and any operating system.

### Target Operating System:

Because of the nature of this project being web-based, it will be accessible via any operating system that can handle a web browser and web surfing - which is pretty much every operating system.

### 3rd Part Libraries, API's, data sources:

Django - "a high-level Python Web framework that encourages rapid development and clean, pragmatic design. " <https://www.djangoproject.com/> Within Django, the db module will handle queries and inserts to the database. The SessionMiddleware class will handle the session scope.

MySQL - "a freely available open source Relational Database Management System." <http://www.mysql.com/> MySQL will be the database that holds all of the data for dynamic content. It will be stored locally and the content will queried or inserted using Django.

### Risks:

I've never used a framework for web development before. The primary concerns are that I'm not extremely good with the session scope and getting Django talking to the database.

### Skills:

I've used enough HTML, CSS, and JavaScript to feel pretty comfortable with a lot of the individual pages front-end. I have experience using SQL to select and insert data. I've worked with the session scope, so it's not that intimidating, although I'm not amazing, which is why I put it as a risk.

### Schedule:

1. Setup Django and the database.

The first step will be getting the database and Django setup and talking to each other. In my experience it's reasonable to expect this part to take 15 hours, between installing the software and their dependencies and then getting them to start communicating.

1. Build the HTML, CSS, and JavaScript for the 8 pages.

The week will include building the canvas to put all of the data on. Because each page will be unique and will have to include a place for dynamic content to be included, it can expected that each page will be about an hour and a half to two hours, which will come out to close to 15 hours. This is expecting minimal fun graphics and flair for each page.

1. Build the dynamics for each page.

Week 3 will be all about the database portion. After we've got the database talking to Django and a place to put the data we can then start querying stuff. Because each page is expected to have dynamic content and there will be a learning curve, it might be a little risky to expect this part to be done in less then 15-20 hours. However, if needs be then some of the less important pages may become static due to lack of time.

1. Handle login and session scope for purchases.

Week 4 will be tying it all together. After all of the information is on the pages it will be time to build in the user experience of logging into the website, making purchases, and logging out. Once again, because of the learning curve this part may take upwards of 15 hours, but hopefully after working in the environment for the previous 3 weeks, hopefully the curve will be a little less steep.