

CINS 465: Assignment 6

This is an individual assignment. **Do your own work. Do not copy / paste anything from any outside source** You may brainstorm with or get help on technical problems from others, but all submitted work must be your own.

Objectives

- Modify your Django-generated Chess site to store the board state via a backend Django model, and perform movement of pieces on the backend via Django form submission, rather than on the front-end via javascript DOM manipulation.
- Starting with your solution for Assignment 5:
 1. Make a copy of your current **assignment5** project directory named **assignment6**.
 - For example, under Linux
 - Navigate to the directory containing your outer Assignment 5 directory, e.g.
 - `cd ~/bherring/cins465/django/`
 - `cp -R assignment5 assignment6`
 2. Now open the assignment6 project directory in your favorite editor or IDE (Atom highly recommended).
 3. Modify your existing chess site so that it uses Django models to store the chessboard state on the backend, and Django form submission to perform the chess piece movement, as we did for my Sudoku example site during week 7's Tuesday & Thursday lectures (my notes are posed under Course Content > Lectures > Week 7 > Lecture 13 & Lecture 14).

Requirements

- You must maintain all functionality and layout from the Assignment 5 requirements.
- You must migrate your chess board data into a new Django model as shown during week 7's Tuesday lecture (my notes for this are posed under Course Content > Lectures > Week 7 > Lecture 13), and use this model in views.py to generate the chessboard data sent to the render function.
- You must replace your statically generated HTML form with at least one Django generated form defined within a new forms.py file, as I showed in class, including any needed form validation.
 - Validation must include checking that all form controls contain valid data and displaying the appropriate errors(s) to the user if not.
- **Note: In order for you to get good practice with Django form validation, if you used drop-down controls for piece movement in Assignment 5, switch over to using a pair of text input controls instead, one for the source location and one for the destination location. You must use standard chess notation, for example e2 (for white king's pawn).**
- You must modify your chessboard html template to render your Django form using Django form tags as demonstrated in lecture.
- You must modify your views.py file to handle the submission of your new piece movement form such that it handles form validation, piece movement (by manipulating the Django model you created) and sends the correct form content to the render function
- You must implement a separate navigation item for resetting the chessboard to the start state (simply refreshing the page will no longer do this as you must always render the current state of the chessboard from the data stored in your backend model).
- **You must submit a single zip or tar file** containing your entire Django project such that when I open it I can navigate into your assignment6 project directory and start your web server using manage.py.