My ambition is to build a webpage beyond the list of catalog, but more advanced system such as recommendation system or searching system based on Django framework.

Ideally, I would love to use the dataset from expedia (https://www.kaggle.com/c/expedia-hotel-recommendations/data) for my SI650 class information retrieval final project (which focus on developing the recommendation algorithm), and implement the interface and backend by using Django for this final project. This data set has 24 variables including:date_time,site_name,posa_continent,user_location_country, user_location_region, user_location_city, orig_destination_distance, user_id, is_mobile, is_package, channel, srch_ci, srch_co,srch_adults_cnt,srch_children_cnt, srch_rm_cnt, srch_destination_id, srch_destination_type_id, hotel_continent, hotel_country, hotel_market, is_booking, cnt, hotel_cluster. I assume there could be 7 tables covering user's information, search information, destination information, etc. User can choose the searching options (numbers of adults/children, location, etc.) on the webpage, and get the recommended hotel cluster from expedia.

If the system cannot work out, I will apply searching function of the Django:Full-text search to this program. The dataset mentioned above also can be used for this alternative, but because it is random selection from Expedia, I would also consider about the dataset like https://www.kaggle.com/new-york-city/new-york-city-current-job-postings, an updating job data from New York containing 27 variables which can have many-to-many relationship and various tables. User can search more useful information in this dataset: search the job description, skills requirement, corresponding salaries, etc.