**Analysis**

**2. Create URL configuration and STUB views for a new model**

View :-

**get\_next\_questiongroup**(request, questionnaire\_id, order\_info = None)

IF order\_info == None

order\_info = 1

ELSE order\_info = order\_info

Based on order\_info, get **questiongroup\_id**

**#below is a method to build the form based on questiongroup\_id**

make\_question\_group\_form(**questiongroup\_id**)

IF request.POST #if the for is submitted

#get answers and save it.

Then order\_info = order\_info +1

Return questionnaire\_id, order\_info to the same method

Explanation :-

This view shows that at first instance with URL configuration of

../<questionnaire\_id>/<order\_info>/

If order\_info is not set, set it to 1, else set it on what the user has entered

Eg. question/1/1 or question/1/2

The parameter questionnaire id and order\_info, query will be met to check what questiongroup is associated with the two parameters.

And then the retrieved questiongroup\_id will be used in forms.py method named make\_question\_group\_form()

Skipping forward if the form has been saved or finalized etc, the set order\_info + 1 to retrieve the next set of questiongroup, using this method again

Question is – How do I make sure the method wont return out of range order\_info?

The plan is to include a check\_next\_order\_info is TRUE or FALSE, after incrementing order\_info + 1

Check\_next\_order\_info(questionnaire\_id,order\_info) method will be called

**Check\_next\_order\_info**(questionnaire\_id,order\_info)

FOR all questionnaire, GET the LAST order\_info

If CURRENT order\_info > LAST order\_info

Return FALSE

Else Return TRUE

Then at the view function

If Check\_next\_order\_info(questionnaire\_id,order\_info) == TRUE

Return reverse(‘**get\_next\_questiongroup**’, questionnaire\_id, order\_info)

IF Check\_next\_order\_info(questionnaire\_id,order\_info) == FALSE

Return (‘questionnaire\_finish’)