package com.twitter.visibility.configapi.configs

import com.twitter.timelines.configapi.Config

import com.twitter.visibility.configapi.params.RuleParams.\_

import com.twitter.visibility.configapi.params.VisibilityExperiments.\_

import com.twitter.visibility.models.SafetyLevel

import com.twitter.visibility.models.SafetyLevel.\_

private[visibility] object VisibilityExperimentsConfig {

import ExperimentsHelper.\_

val TestExperimentConfig: Config = mkABExperimentConfig(TestExperiment, TestHoldbackParam)

val NotGraduatedUserLabelRuleHoldbackExperimentConfig: Config =

mkABExperimentConfig(

NotGraduatedUserLabelRuleExperiment,

NotGraduatedUserLabelRuleHoldbackExperimentParam

)

def config(safetyLevel: SafetyLevel): Seq[Config] = {

val experimentConfigs = safetyLevel match {

case Test =>

Seq(TestExperimentConfig)

case \_ => Seq(NotGraduatedUserLabelRuleHoldbackExperimentConfig)

}

experimentConfigs

}

}