**InActive Tool business logic**

**In Active Controller:**

@RequestMapping(value = "/ChangeState", method = RequestMethod.***GET***)

@ResponseBody

**public** String changeState(HttpServletRequest request, HttpServletResponse response)

This function helps to change the state of user accounts (from inactive to active or from active to inactive). At the same time, log file is recorded.

String tj = request.getParameter("tj");

String state = request.getParameter("state");

String comment = request.getParameter("comment");

System.***out***.println(tj + ", " + state + ", " + comment);

String[] emails = tj.split(",");

**for** (**int** i = 0; i < emails.length; i++) {

**if** (ConstantUtil.*ACTIVE*.equals(state)) {

activeService.setIsActive(emails[i], **true**);

} **else** **if** (ConstantUtil.*INACTIVE*.equals(state)) {

activeService.setIsActive(emails[i], **false**);

……

“tj” is a string which contains all user ids split with “;”, “state” is the string format of current state (“active” or “inactive”), “comment” is the comment which operators expected to submit. When this code is conducted, states of user accounts will be modified in DB of TestRail.

@RequestMapping(value = "/GettingOverDate", method = RequestMethod.***GET***)

**public** Map<String, Object> gettingOverDate(HttpServletRequest request, HttpServletResponse response)

This function helps get the current setting of auto-inactive users.

@RequestMapping(value = "/Setting", method = RequestMethod.***GET***)

@ResponseBody

**public** Boolean setting(HttpServletRequest request, HttpServletResponse response)

This function will reset “overDays”, “swh” and “interval” parameters of this system for auto-inactive. Background thread of auto-inactive will be updated.

@RequestMapping(value = "/ViewData", method = RequestMethod.***GET***)

@ResponseBody

**public** Object viewData(HttpServletRequest request, HttpServletResponse response)

The function achieves information of each user’s logging in from DB of TestRail, including “name”, “email”, “isActive”, “updated”.

@RequestMapping(value = "/GetSetting", method = RequestMethod.***GET***)

@ResponseBody

**public** TestRailActiveSetting getSetting()

This function achieves the state of the background auto-inactive thread, including “overDays”, “swh” and “interval” parameters.

@RequestMapping(value = "/ViewLogInfo", method = RequestMethod.***GET***)

@ResponseBody

**public** List<TestRailActiveRecord> viewLogInfo() **throws** IOException

This function allows us to achieve the log file of operation history.

@RequestMapping(value = "/DeleteLogFile", method = RequestMethod.***GET***)

@ResponseBody

**public** **boolean** deleteLogFile()

This function allows us to delete the log file of operation history.