Giới thiệu về thư viện excelExport,

Đây là một thư viện hỗ trợ cho việc bổ sung dữ liệu vào một file excel có sẵn templete. Giúp việc export-download excel file đơn giản hơn. Thư viện cung cấp các hàm đọc file excel rồi bổ sung data vào file đó. rồi bạn có thể save lại ổ cứng hoặc truyền nó cho client. Tập dữ liệu truyền vào là dạng json, hoặc hashMap. Và JSONArray jsonArray.

( xem thêm nội dung mô tả trong file excel)

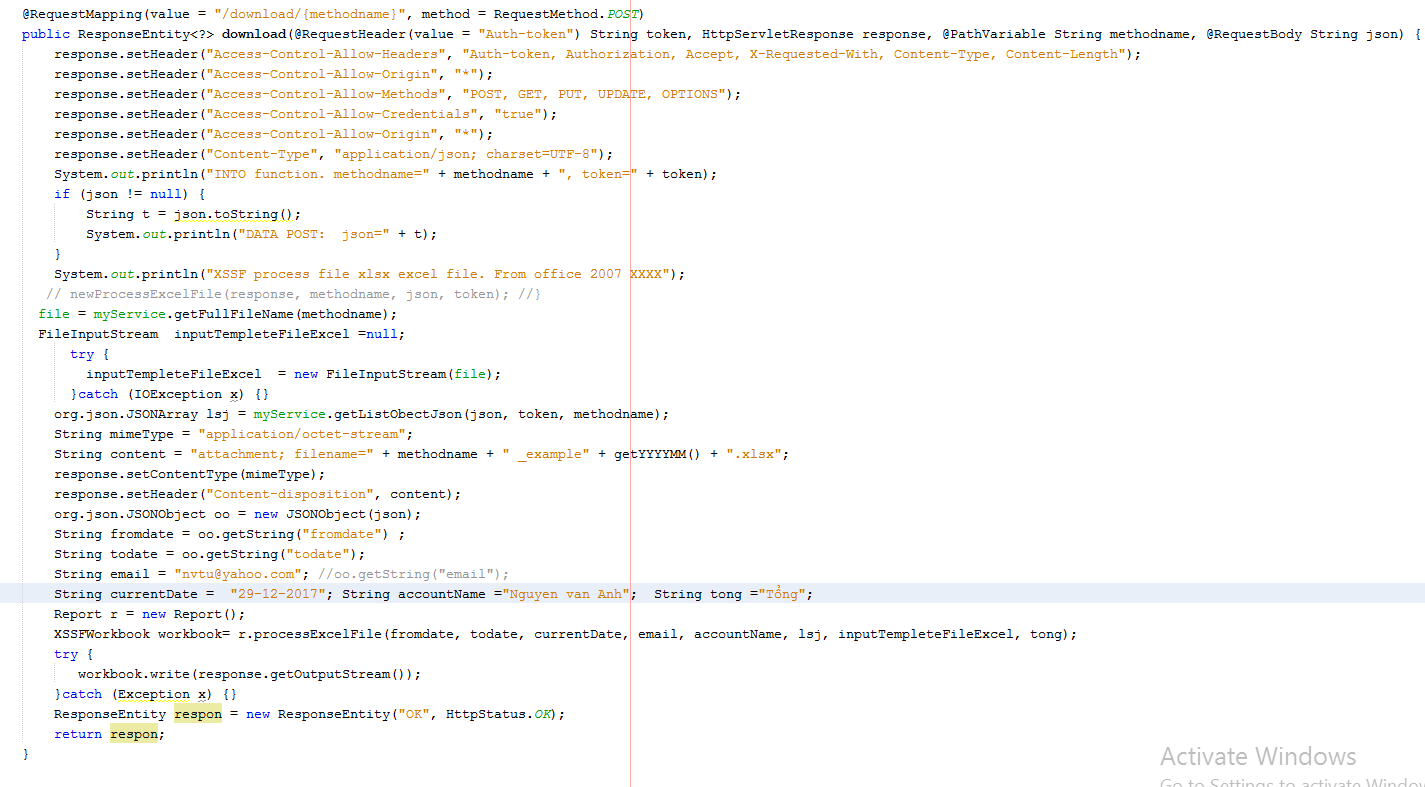
Thư viện excelExport cung cấp 3 hàm xử lý file excel. Trong đó có 1 tham số FileInputStream, Lý do bạn phải truyền vào FileInputStream là vì bạn phải close FileInputStream sau khi gửi file cho client. .

Mục đích là trợ giúp cho việc xuất file excel .

**Khi sử dụng hàm sau thì phải truyền vào chi tiết các tham số: fromDate, toDate …. Tên các tham số này phải đúng với tên các tham số đặt trong file excel.**

(\*) public XSSFWorkbook processExcelFile(String fromDate, String toDate, String curentDate, String email, String accountName, JSONArray jsonArray, FileInputStream inputTempleteFileExcel, String tong)

|  |
| --- |
| **Việc sử dụng 2 hàm (\*\*) , (\*\*\*) là tương đương. Khi bạn muốn truyền fromDate, toDate …. Thì cho nó vào HashMap hoặc JSON object. Tên các keys của json hoặc hashMap phải ứng với các tham số trong file excel.** |
| (\*\*) public XSSFWorkbook processExcelFile( HashMap<String,String> paramaters, JSONArray jsonArray, FileInputStream inputTempleteFileExcel, String tong) |
| (\*\*\*) public XSSFWorkbook processExcelFile( JSONObject paramaters, JSONArray jsonArray, FileInputStream inputTempleteFileExcel, String tong) |
|  |
| Cơ bản thì HashMap và Json thì đều có keys-value. Khi data truyền từ client (trong body ) request, có thể chứa ngày tháng, userName, email của người trích xuất báo cáo. Mà bạn muốn đặt các tham số này vào báo cáo thì sử đặt trong HashMap hoặc Json để thư viện đặt nó vào trong file excel cho bạn |
| Lập trình viên sử dụng thư viện thì liệt kê các tham số ( keys) trong hashmap|Json vào cột parameters. |
|  |
| * Đoạn mã sau đây ví dụ về sự dụng export data into file excel, và response về cho brower.   Dưới đây là một demo của, việc sử dụng thư viện reportExcel với Spring Rest để trả file excel cho client: |



@RequestMapping(value = "/download/{methodname}", method = RequestMethod.POST)

public ResponseEntity<?> download(@RequestHeader(value = "Auth-token") String token, HttpServletResponse response, @PathVariable String methodname, @RequestBody String json) {

response.setHeader("Access-Control-Allow-Headers", "Auth-token, Authorization, Accept, X-Requested-With, Content-Type, Content-Length");

response.setHeader("Access-Control-Allow-Origin", "\*");

response.setHeader("Access-Control-Allow-Methods", "POST, GET, PUT, UPDATE, OPTIONS");

response.setHeader("Access-Control-Allow-Credentials", "true");

response.setHeader("Access-Control-Allow-Origin", "\*");

response.setHeader("Content-Type", "application/json; charset=UTF-8");

System.out.println("INTO function. methodname=" + methodname + ", token=" + token);

if (json != null) {

String t = json.toString();

System.out.println("DATA POST: json=" + t);

}

//process (response, methodname, json , token) ;

//if( fileSuffixes.contains("xslx")) {

System.out.println("XSSF process file xlsx excel file. From office 2007 XXXX");

// newProcessExcelFile(response, methodname, json, token); //}

file = myService.getFullFileName(methodname);

FileInputStream inputTempleteFileExcel =null;

try {

inputTempleteFileExcel = new FileInputStream(file);

}catch (IOException x) {}

org.json.JSONArray lsj = myService.getListObectJson(json, token, methodname);

String mimeType = "application/octet-stream";

String content = "attachment; filename=" + methodname + " \_example" + getYYYYMM() + ".xlsx";

response.setContentType(mimeType);

response.setHeader("Content-disposition", content);

org.json.JSONObject oo = new JSONObject(json);

String fromdate = oo.getString("fromdate") ;

String todate = oo.getString("todate");

String email = "nvtu@yahoo.com"; //oo.getString("email");

String currentDate = "29-12-2017";

String accountName ="Nguyen van Anh";

String tong ="Tổng";

Report r = new Report();

XSSFWorkbook workbook= r.processExcelFile(fromdate, todate, currentDate, email, accountName, lsj, inputTempleteFileExcel, tong);

try {

workbook.write(response.getOutputStream());

}catch (Exception x) {}

ResponseEntity respon = new ResponseEntity("OK", HttpStatus.OK);

return respon;

}

Xem tiếp cấu trúc file excel. Trước khi fill data và sau khi fill data trả về cho client .

Sử dụng dependencies sau để bổ sung các thư viện đính kèm apache poi và json. Hoặc import file đính kèm.

<dependency>

<groupId>org.json</groupId>

<artifactId>json</artifactId>

<version>20160810</version>

</dependency>

<dependency>

<groupId>org.apache.poi</groupId>

<artifactId>poi-ooxml</artifactId>

<version>3.15</version>

<type>jar</type>

</dependency>

<dependency>