**1. 비밀번호 변경**

마이페이지에서 기존비밀번호, 새비밀번호, 새비밀번호확인 후 비밀번호를 변경한다.

**1.1 Request :: POST방식**

HTTP URL = " /ChangePassword"

- Parameter 형식 (Request Method = POST 방식)

|  |  |  |  |
| --- | --- | --- | --- |
| 파라미터명 | 타입 | 필수여부 | 설명 |
| login\_id | String | Y | 로그인중인 아이디 |
| password | String | Y | 현재 패스워드 |
| new\_pwd1 | String | Y | 변경할 패스워드 |
| new\_pwd2 | String | Y | 변경할 패스워드 확인 |

**1.2 Response Format :: JOSN 형태로 반환**

- 반환값 형식

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 엘리먼트 명 | Depth | 배열  구분 | 설명 | 값 구분 |
| status | 1 |  | 성공여부 |  |
| message | 1 |  | 메시지 |  |

**1.3 샘플 JSON예제**

|  |
| --- |
| 실패할 경우  {  “status”:”error”,  “message”:”비밀번호가 일치하지 않습니다.”  }  성공할 경우  {  “status”:”success”,  “message”:”비밀번호가 변경되었습니다.”  } |

**2. 회원정보 조회**

마이페이지에서 회원(abc123)정보를 수정하기위해 로그인 아이디로 기존 회원정보를 가져온다.

**2.1. Request :: GET 방식**

HTTP URL = " /UserInfo?id=abc123"

- Parameter 형식 (Request Method = GET 방식)

|  |  |  |  |
| --- | --- | --- | --- |
| 파라미터명 | 타입 | 필수여부 | 설명 |
| id | String | 필수 | 로그인 아이디 |

**2.2. Response Format :: JOSN 형태로 반환**

- 반환값 형식

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 엘리먼트 명 | Depth | 배열  구분 | 설명 | 값 구분 |
| status | 1 |  | 성공여부 |  |
| data | 1 |  | 회원 정보 |  |
| id | 2 |  | 회원 아이디 |  |
| name | 2 |  | 회원 패스워드 |  |
| email | 2 |  | 회원 이메일 |  |
| phone | 2 |  | 회원 전화번호 |  |
| company | 2 |  | 회원 기업명 |  |

**2.3. 샘플 JSON예제**

|  |
| --- |
| 실패할 경우  {  “status”:“error”,  “data”:”error”  }  성공할 경우  {  “status”:”success”,  “data”:  {  “id”:”abc123”,  “name”:”홍길동”,  “email”:”[abc@naver.com](mailto:abc@naver.com)”,  “phone”:”010-1234-1234”,  “company”:”캐드서브”  }  } |

**3. 회원정보 수정**

마이페이지에서 회원정보(email, phone, company)를 수정한다.

**3.1. Request :: POST 방식**

HTTP URL = " /UserInfo\_update"

- Parameter 형식 (Request Method = POST 방식)

|  |  |  |  |
| --- | --- | --- | --- |
| 파라미터명 | 타입 | 필수여부 | 설명 |
| id | String | Y | 로그인 아이디 |
| name | String | Y | 회원 이름 |
| email | String | Y | 수정된 이메일 |
| phone | String | Y | 수정된 전화번호 |
| company | String | Y | 수정된 회사명 |

**3.2. Response Format :: JOSN 형태로 반환**

- 반환값 형식

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 엘리먼트 명 | Depth | 배열  구분 | 설명 | 값 구분 |
| status | 1 |  | 성공여부 |  |
| message | 1 |  | 메시지 |  |

**3.3. 샘플 JSON예제**

|  |
| --- |
| 실패할 경우  {  “status”:”error”,  “message”:”error”  }  성공할 경우  {  “status”:”success”,  “message”:”회원정보 수정이 완료되었습니다.”  } |

**4. 회원가입**

회원가입

**4.1. Request :: POST 방식**

HTTP URL = "/signup"

- Parameter 형식 (Request Method = POST 방식)

|  |  |  |  |
| --- | --- | --- | --- |
| 파라미터명 | 타입 | 필수여부 | 설명 |
| id | String | Y | 아이디 |
| pwd | String | Y | 비밀번호 |
| name | String | Y | 이름 |
| email | String | Y | 이메일 |
| phone | String | Y | 전화번호 |
| company | String | Y | 기업명 |

**4.2. Response Format :: JOSN 형태로 반환**

- 반환값 형식

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 엘리먼트 명 | Depth | 배열  구분 | 설명 | 값 구분 |
| status | 1 | N | 성공여부 | success : 성공(정상)  error : 실패 |
| data | 1 | N | 정보 |  |
| id | 2 | N | 아이디 |  |
| name | 2 | N | 이름 |  |
| email | 2 | N | 이메일 |  |
| phone | 2 | N | 전화번호 |  |
| company | 2 | N | 기업명 |  |

**4.3. 샘플 JSON예제**

|  |
| --- |
| - 실패할 경우 샘플  {"status":"error"," data ":"error”}  -성공할 경우 샘플  {" status ":"success"," data ":{“id”:”abc123”, “name”:”Kim”, “email”:[abc@naver.com](mailto:abc@naver.com), “phone”:”010-123-1234”, “company”:”AA”}} |

**5. 아이디 중복체크**

아이디 중복체크

**5.1. Request :: POST 방식**

HTTP URL = "/check/id"

- Parameter 형식 (Request Method = POST 방식)

|  |  |  |  |
| --- | --- | --- | --- |
| 파라미터명 | 타입 | 필수여부 | 설명 |
| id | String | Y | 아이디 |

**5.2. Response Format :: JOSN 형태로 반환**

- 반환값 형식

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 엘리먼트 명 | Depth | 배열  구분 | 설명 | 값 구분 |
| status | 1 | N | 성공여부 | success : 성공(정상)  error : 실패 |
| data | 1 | N | 중복여부 | ok : 사용가능  no : 사용불가능 |

**5.3. 샘플 JSON예제**

|  |
| --- |
| - 실패할 경우 샘플  {"status":"error"," data ":"error”}  -성공할 경우 샘플  {"status" : "success", "data“ : ”ok”} |

**6. 로그인**

로그인

**6.1. Request :: POST 방식**

HTTP URL = "/login"

- Parameter 형식 (Request Method = POST 방식)

|  |  |  |  |
| --- | --- | --- | --- |
| 파라미터명 | 타입 | 필수여부 | 설명 |
| id | String | Y | 아이디 |
| pwd | String | Y | 비밀번호 |

**6.2. Response Format :: JOSN 형태로 반환**

- 반환값 형식

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 엘리먼트 명 | Depth | 배열  구분 | 설명 | 값 구분 |
| status | 1 | N | 성공여부 | success : 성공(정상)  error : 실패 |
| data | 1 | N | 정보 |  |
| id | 2 | N | 아이디 |  |
| name | 2 | N | 이름 |  |
| email | 2 | N | 이메일 |  |
| phone | 2 | N | 전화번호 |  |
| company | 2 | N | 기업명 |  |

**6.3. 샘플 JSON예제**

|  |
| --- |
| - 실패할 경우 샘플  {"status":"error"," data ":"error”}  -성공할 경우 샘플  {" status ":"success"," data ":{“id”:”abc123”, “name”:”Kim”, “email”:[abc@naver.com](mailto:abc@naver.com), “phone”:”010-123-1234”, “company”:”AA”}} |

**7. 개인 라이센스 현황 조회**

개인 라이센스 요청 현황 전체보기

**7.1. Request :: POST 방식**

HTTP URL = ""

- Parameter 형식(Request Method = POST 방식)

|  |  |  |  |
| --- | --- | --- | --- |
| 파라미터명 | 타입 | 필수여부 | 설명 |
| user\_id | String | Y | 사용자 아이디 |

**7.2. Response Format :: JSON 형태로 반환**

- 반환값 형식

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 엘리먼트 명 | Depth | 배열  구분 | 설명 | 값 구분 |
| status | 1 | N | 성공여부 | success : 성공(정상)  error : 실패 |
| data | 1 | N | 정보 |  |
| sw\_name | 2 | Y | 소프트웨어명 |  |
| device\_name | 2 | Y | 디바이스명 |  |
| corp\_name | 2 | Y | 회사명 |  |
| start\_date | 2 | Y | 라이센스 시작날짜 |  |
| end\_date | 2 | Y | 라이센스 만료날짜 |  |
| state | 2 | Y | 상태 |  |

**7.3 샘플JSON예제**

|  |
| --- |
| - 실패할 경우 샘플  {"status":"error"," data ":"error“}  -성공할 경우 샘플  {"status":"success","data":“{“sw\_name”:“Microsoft Excel”, “device\_name”:“HOME\_PC”, “corp\_name”:“Microsoft”, “start\_date”:“2017-01-01”, “end\_date”:“2017-02-01”, “state”:“2”}“}  state: 0(발급대기), 1(정상발급), 2(데모버젼), 3(발급거절), 5(연장요청) |

**8. 개인 라이센스 현황 소프트웨어별 조회**

소프트웨어별 개인 라이센스 조회

**8.1. Request :: POST 방식**

HTTP URL = ""

- Parameter 형식(Request Method = POST 방식)

|  |  |  |  |
| --- | --- | --- | --- |
| 파라미터명 | 타입 | 필수여부 | 설명 |
| user\_id | String | Y | 사용자 아이디 |
| sw\_name | String | Y | 소프트웨어명 |
|  |  |  |  |

**8.2. Response Format :: JSON 형태로 반환**

- 반환값 형식

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 엘리먼트 명 | Depth | 배열  구분 | 설명 | 값 구분 |
| status | 1 | N | 성공여부 | success : 성공(정상)  error : 실패 |
| data | 1 | N | 정보 |  |
| sw\_name | 2 | Y | 소프트웨어명 |  |
| device\_name | 2 | Y | 디바이스명 |  |
| corp\_name | 2 | Y | 회사명 |  |
| start\_date | 2 | Y | 라이센스 시작날짜 |  |
| end\_date | 2 | Y | 라이센스 만료날짜 |  |
| state | 2 | Y | 상태 |  |

**8.3. 샘플JSON예제**

|  |
| --- |
| - 실패할 경우 샘플  {"status":"error"," data ":"error“}  -성공할 경우 샘플  {"status ":"success"," data":”{“sw\_name”:“Microsoft Excel”, “device\_name”:“HOME\_PC”, “corp\_name”:“Microsoft”, “start\_date”:“2017-01-01”, “end\_date”:“2017-02-01”, “state”:“2”}”}  state: 0(발급대기), 1(정상발급), 2(데모버젼), 3(발급거절), 5(연장요청) |

**9. 라이센스 요청 현황 조회**

관리자 계정으로 라이센스 요청 현황 전체 조회

**9.1. Request :: POST 방식**

HTTP URL = ""

- Parameter 형식(Request Method = POST 방식)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 파라미터명 | 타입 | 필수여부 | 설명 | |
| user\_id | String | Y | 사용자 아이디 |
| corp\_id | String | Y | 회사명 |
| authority | int | Y | 권한(0:개인, 1:관리자) |
|  |  |  |  |

**9.2. Response Format :: JSON 형태로 반환**

- 반환값 형식

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 엘리먼트 명 | Depth | 배열  구분 | 설명 | 값 구분 |
| status | 1 | N | 성공여부 | success : 성공(정상)  error : 실패 |
| data | 1 | N | 정보 |  |
| sw\_name | 2 | Y | 소프트웨어명 |  |
| user\_name | 2 | Y | 사용자명 |  |
| request\_date | 2 | Y | 요청 날짜 |  |
| state | 2 | Y | 상태 |  |
|  |  |  |  |  |
|  |  |  |  |  |

**9.3. 샘플JSON예제**

|  |
| --- |
| - 실패할 경우 샘플  {"status":"error"," data ":"error“}  -성공할 경우 샘플  {"status":"success","data":”{“sw\_name”:“Microsoft Excel”, “user\_name”:“박지성”, “request\_date”:”2017-01-01”,  “state”:“0”}”, “{“sw\_name”:”Football Manager”,”user\_name”:”김남일”, “request\_date”:”2017-02-02”, “state”:”1”}”}  state: 0(데모신청 대기), 1(일반신청 대기), 2(연장신청 대기) |

**10. 라이센스 요청 현황 소프트웨어명 별로 조회**

관리자 계정으로 라이센스 요청 현황 소프트웨어명 별로 조회

**10.1. Request :: POST 방식**

HTTP URL = ""

- Parameter 형식(Request Method = POST 방식)

|  |  |  |  |
| --- | --- | --- | --- |
| 파라미터명 | 타입 | 필수여부 | 설명 |
| user\_id | String | Y | 사용자 아이디 |
| corp\_id | String | Y | 회사명 |
| authority | int | Y | 권한(0:개인, 1:관리자) |
| sw\_name | String | Y | 소프트웨어명 |

**10.2. Response Format :: JSON 형태로 반환**

- 반환값 형식

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 엘리먼트 명 | Depth | 배열  구분 | 설명 | 값 구분 |
| status | 1 | N | 성공여부 | success : 성공(정상)  error : 실패 |
| data | 1 | N | 정보 |  |
| sw\_name | 2 | Y | 소프트웨어명 |  |
| user\_name | 2 | Y | 사용자명 |  |
| request\_date | 2 | Y | 요청 날짜 |  |
| state | 2 | Y | 상태 |  |
|  |  |  |  |  |

**10.3. 샘플JSON예제**

|  |
| --- |
| - 실패할 경우 샘플  {"status":"error"," data ":"error“}  -성공할 경우 샘플  {"status":"success","data":”{“sw\_name”:“Microsoft Excel”, “user\_name”:“박지성”, “request\_date”:”2017-01-01”,  “state”:“0”}”, “{“sw\_name”:”Microsoft Excelr”,”user\_name”:”김남일”, “request\_date”:”2017-02-02”, “state”:”1”}”}  state: 0(데모신청 대기), 1(일반신청 대기), 2(연장신청 대기) |

**11. 라이센스 요청 현황 신청 분류 별 조회**

관리가 계정으로 라이센스 요청 현황 분류 별로 조회(일반신청, 데모신청)

**11.1. Request :: POST 방식**

HTTP URL = ""

- Parameter 형식(Request Method = POST 방식)

|  |  |  |  |
| --- | --- | --- | --- |
| 파라미터명 | 타입 | 필수여부 | 설명 |
| user\_id | String | Y | 사용자 아이디 |
| corp\_id | String | Y | 회사명 |
| authority | int | Y | 권한(0:개인, 1:관리자) |
| state | int | Y | 신청 상태(0: |

**11.2. Response Format :: JSON 형태로 반환**

- 반환값 형식

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 엘리먼트 명 | Depth | 배열  구분 | 설명 | 값 구분 |
| status | 1 | N | 성공여부 | success : 성공(정상)  error : 실패 |
| data | 1 | N | 정보 |  |
| sw\_name | 2 | Y | 소프트웨어명 |  |
| user\_name | 2 | Y | 사용자명 |  |
| request\_date | 2 | Y | 요청 날짜 |  |
| state | 2 | Y | 상태 |  |
|  |  |  |  |  |
|  |  |  |  |  |

**11.3. 샘플JSON예제**

|  |
| --- |
| - 실패할 경우 샘플  {"status":"error"," data ":"error“}  -성공할 경우 샘플  {"status":"success","data":”{“sw\_name”:“Microsoft Excel”, “user\_name”:“박지성”, “request\_date”:”2017-01-01”,  “state”:“0”}”, “{“sw\_name”:”Microsoft Excelr”,”user\_name”:”김남일”, “request\_date”:”2017-02-02”, “state”:”0”}”}  state: 0(데모신청 대기), 1(일반신청 대기), 2(연장신청 대기) |

**12. 라이센스 요청 현황 신청 분류 별, 소프트웨어명 별 조회**

관리자 계정으로 라이센스 요청 현황(일반신청, 데모신청) + 소프트웨어명 별로 조회

**12.1. Request :: POST 방식**

HTTP URL = ""

- Parameter 형식(Request Method = POST 방식)

|  |  |  |  |
| --- | --- | --- | --- |
| 파라미터명 | 타입 | 필수여부 | 설명 |
| user\_id | String | Y | 사용자 아이디 |
| corp\_id | String | Y | 회사명 |
| sw\_name | String | Y | 소프트웨어명 |
| authority | int | Y | 권한(0:개인, 1:관리자) |
| state | int | Y | 신청 상태 |

**12.2. Response Format :: JSON 형태로 반환**

- 반환값 형식

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 엘리먼트 명 | Depth | 배열  구분 | 설명 | 값 구분 |
| status | 1 | N | 성공여부 | success : 성공(정상)  error : 실패 |
| data | 1 | N | 정보 |  |
| sw\_name | 2 | Y | 소프트웨어명 |  |
| user\_name | 2 | Y | 사용자명 |  |
| request\_date | 2 | Y | 요청 날짜 |  |
| state | 2 | Y | 상태 |  |
|  |  |  |  |  |

**12.3. 샘플JSON예제**

|  |
| --- |
| - 실패할 경우 샘플  {"status":"error"," data ":"error“}  -성공할 경우 샘플  {"status":"success","data":”{“sw\_name”:“Microsoft Excel”, “user\_name”:“박지성”, “request\_date”:”2017-01-01”,  “state”:“0”}”, “{“sw\_name”:”Microsoft Excelr”,”user\_name”:”김남일”, “request\_date”:”2017-02-02”, “state”:”0”}”}  state: 0(데모신청 대기), 1(일반신청 대기), 2(연장신청 대기) |

**13. 라이센스 발급 완료 현황 전체 조회**

관리자 계정으로 라이센스 발급 완료 현황 전체 조회

**13.1. Request :: POST 방식**

HTTP URL = ""

- Parameter 형식(Request Method = POST 방식)

|  |  |  |  |
| --- | --- | --- | --- |
| 파라미터명 | 타입 | 필수여부 | 설명 |
| user\_id | String | Y | 사용자 아이디 |
| corp\_id | String | Y | 회사명 |
| authority | int | Y | 권한(0:개인, 1:관리자) |
|  |  |  |  |

**13.2. Response Format :: JSON 형태로 반환**

- 반환값 형식

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 엘리먼트 명 | Depth | 배열  구분 | 설명 | 값 구분 |
| status | 1 | N | 성공여부 | success : 성공(정상)  error : 실패 |
| data | 1 | N | 정보 |  |
| sw\_name | 2 | Y | 소프트웨어명 |  |
| user\_name | 2 | Y | 사용자명 |  |
| start\_date | 2 | Y | 시작 날짜 |  |
| end\_date | 2 | Y | 만료 날짜 |  |
| state | 2 | Y | 상태 |  |
|  |  |  |  |  |

**13.3. 샘플JSON예제**

|  |
| --- |
| - 실패할 경우 샘플  {"status":"error"," data ":"error“}  -성공할 경우 샘플  {"status":"success","data":”{“sw\_name”:“Microsoft Excel”, “user\_name”:“박지성”, “request\_date”:”2017-01-01”,  “state”:“0”}”, “{“sw\_name”:”Microsoft Excelr”,”user\_name”:”김남일”, “request\_date”:”2017-02-02”, “state”:”0”}”}  state: 0(데모신청), 1(일반신청) |

**14. 내 디바이스 조회**

등록된 디바이스를 모두 조회한다

**14.1. Request :: POST 방식**

HTTP URL = " /device/all "

- Parameter 형식 (Request Method = POST 방식)

|  |  |  |  |
| --- | --- | --- | --- |
| 파라미터명 | 타입 | 필수여부 | 설명 |
| user\_id | String | 필수 | 조회 할 사용자의 id |
|  |  |  |  |
|  |  |  |  |

**14.2. Response Format :: JSON 형태로 반환**

- 반환값 형식

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 엘리먼트 명 | Depth | 배열  구분 | 설명 | 값 구분 |
| status | 1 | N |  | success : 성공  error : 실패 |
| data | 1 | N | 정보 |  |
| nickname | 2 | Y | 디바이스 닉네임 |  |
| type | 2 | Y | 디바이스 타입 |  |
| mac | 2 | Y | PC |  |
| usim | 2 | Y | Mobile |  |
| pin | 2 | Y | Moblie |  |
| serial | 2 | Y | PC |  |
| udid | 2 | Y | PC/Mobile |  |

+ 굳이 다 보여줘야 할 필요?

**14.3. 샘플 JSON예제**

|  |
| --- |
| - 실패할 경우 샘플  {"status":"error", "data":"error"}  -성공할 경우 샘플  {"status":"sucess", "data":"{“nickname”:”HOME-PC”, “type”:”0”, “mac”:” dc:a9:04:90:60:98”, “usim”:”null”, “pin”:”null”, “serial”:” C02V10CKHV2H”, “udid”:” 671B6464-DBA2-5D10-BBCB-FB395080B984”}"} |

**15. 소프트웨어별 내 디바이스 조회**

특정 소프트웨어의 라이센스를 가진 디바이스를 조회한다

**15.1. Request :: POST 방식**

HTTP URL = " /device/soft "

- Parameter 형식 (Request Method = POST 방식)

|  |  |  |  |
| --- | --- | --- | --- |
| 파라미터명 | 타입 | 필수여부 | 설명 |
| user\_id | String | 필수 | 조회 할 사용자의 id |
| sw\_name | String | 필수 | 선택한 소프트웨어명 |
|  |  |  |  |

**15.2. Response Format :: JSON 형태로 반환**

- 반환값 형식

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 엘리먼트 명 | Depth | 배열  구분 | 설명 | 값 구분 |
| status | 1 | N |  | success : 성공  error : 실패 |
| data | 1 | N | 정보 |  |
| nickname | 2 | Y | 디바이스 닉네임 |  |
| type | 2 | Y | 디바이스 타입 |  |
| mac | 2 | Y | PC |  |
| usim | 2 | Y | Mobile |  |
| pin | 2 | Y | Moblie |  |
| serial | 2 | Y | PC |  |
| udid | 2 | Y | PC/Mobile |  |

**15.3. 샘플 JSON예제**

|  |
| --- |
| - 실패할 경우 샘플  {"status":"error", "data":"error"}  -성공할 경우 샘플  {"status":"sucess", "data":"{“nickname”:”HOME-PC”, “type”:”0”, “mac”:” dc:a9:04:90:60:98”, “usim”:”null”, “pin”:”null”, “serial”:” C02V10CKHV2H”, “udid”:” 671B6464-DBA2-5D10-BBCB-FB395080B984”}"} |

**16. 디바이스 추가**

새로운 디바이스를 추가한다

**16.1. Request :: POST 방식**

HTTP URL = " /addnewdevice "

- Parameter 형식 (Request Method = POST 방식)

|  |  |  |  |
| --- | --- | --- | --- |
| 파라미터명 | 타입 | 필수여부 | 설명 |
| user\_id | String | 필수 | 조회 할 사용자의 id |
|  |  |  |  |
|  |  |  |  |

**16.2. Response Format :: JSON 형태로 반환**

- 반환값 형식

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 엘리먼트 명 | Depth | 배열  구분 | 설명 | 값 구분 |
| status | 1 | N |  | success : 성공  error : 실패 |
| data | 1 | N | 정보 |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**16.3. 샘플 JSON예제**

|  |
| --- |
| - 실패할 경우 샘플  {"status":"error", "data":"error"}  -성공할 경우 샘플  {"status":"sucess", "data":} |

**17. 통계 전체 조회**

전체 소프트웨어에 대한 라이센스 사용 통계를 조회한다

(총 다운로드/ 라이센스 요청/ 발급된 라이센스/ 데모버전)

**17.1. Request :: POST 방식**

HTTP URL = " /statistics/all "

- Parameter 형식 (Request Method = POST 방식)

|  |  |  |  |
| --- | --- | --- | --- |
| 파라미터명 | 타입 | 필수여부 | 설명 |
| user\_id | String | 필수 | 조회 할 사용자의 id |
| corp\_id | int | 필수 | 조회 할 사용자의 회사 id |
| authority | int | 필수 | 조회 할 사용자의 권한 |

**17.2. Response Format :: JSON 형태로 반환**

- 반환값 형식

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 엘리먼트 명 | Depth | 배열  구분 | 설명 | 값 구분 |
| status | 1 | N |  | success : 성공  error : 실패 |
| data | 1 | N | 정보 |  |
| software\_list | 2 | Y | 회사가 보유한 소프트웨어 목록 |  |
| total\_count | 2 | N | 총 다운로드 수 |  |
| req\_count | 2 | N | 라이센스 요청 수 |  |
| complete\_count | 2 | N | 발급된 라이센스 수 |  |
| demo\_count | 2 | N | 데모 라이센스 발급 수 |  |

**17.3. 샘플 JSON예제**

|  |
| --- |
| - 실패할 경우 샘플  {"status":"error", "data":"error"}  -성공할 경우 샘플  {"status":"sucess", "data":”{“software\_list”:”{“Microsoft Office Excel”, “Microsoft Office Word”}”, “total\_count”:”10”, “req\_count”:”5”, “complete\_count”:”7”, demo\_count:”3”}”} |

**18. 소프트웨어별 통계 조회**

선택한 소프트웨어에 대한 라이센스 사용 통계를 조회한다

(총 다운로드/ 라이센스 요청/ 발급된 라이센스/ 데모버전)

**18.1. Request :: POST 방식**

HTTP URL = " /statistics/software "

- Parameter 형식 (Request Method = POST 방식)

|  |  |  |  |
| --- | --- | --- | --- |
| 파라미터명 | 타입 | 필수여부 | 설명 |
| user\_id | String | 필수 | 조회 할 사용자의 id |
| corp\_id | int | 필수 | 조회 할 사용자의 회사 id |
| authority | int | 필수 | 조회 할 사용자의 권한 |
| sw\_name | String | 필수 | 조회 할 소프트웨어명 |

**18.2. Response Format :: JSON 형태로 반환**

- 반환값 형식

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 엘리먼트 명 | Depth | 배열  구분 | 설명 | 값 구분 |
| status | 1 | N |  | success : 성공  error : 실패 |
| data | 1 | N | 정보 |  |
| software\_list | 2 | Y | 회사가 보유한 소프트웨어 목록 |  |
| total\_count | 2 | N | 총 다운로드 수 |  |
| req\_count | 2 | N | 라이센스 요청 수 |  |
| complete\_count | 2 | N | 발급된 라이센스 수 |  |
| demo\_count | 2 | N | 데모 라이센스 발급 수 |  |

**18.3. 샘플 JSON예제**

|  |
| --- |
| - 실패할 경우 샘플  {"status":"error", "data":"error"}  -성공할 경우 샘플  {"status":"sucess", "data":”{“software\_list”:”{“Microsoft Office Excel”, “Microsoft Office Word”}”, “total\_count”:”10”, “req\_count”:”5”, “complete\_count”:”7”, demo\_count:”3”}”} |

**19. 월별 라이센스 발급 현황 조회**

전체 소프트웨어에 대한 월별 라이센스 발급 현황을 조회한다

(총 다운로드/ 라이센스 요청/ 발급된 라이센스/ 데모버전)

**19.1. Request :: POST 방식**

HTTP URL = " /statistics/monthly/all"

- Parameter 형식 (Request Method = POST 방식)

|  |  |  |  |
| --- | --- | --- | --- |
| 파라미터명 | 타입 | 필수여부 | 설명 |
| user\_id | String | 필수 | 조회 할 사용자의 id |
| corp\_id | int | 필수 | 조회 할 사용자의 회사 id |
| authority | int | 필수 | 조회 할 사용자의 권한 |

**19.2. Response Format :: JSON 형태로 반환**

- 반환값 형식

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 엘리먼트 명 | Depth | 배열  구분 | 설명 | 값 구분 |
| status | 1 | N |  | success : 성공  error : 실패 |
| data | 1 | N | 정보 |  |
| software\_list | 2 | Y | 회사가 보유한 소프트웨어 목록 |  |
| total\_count | 2 | Y | 월별 라이센스 발급 현황 |  |

**19.3. 샘플 JSON예제**

|  |
| --- |
| - 실패할 경우 샘플  {"status":"error", "data":"error"}  -성공할 경우 샘플  {"status":"sucess", "data":”{“software\_list”:”{“Microsoft Office Excel”, “Microsoft Office Word”}”, “total\_count”:”{“1월”:”100”, ”2월”:”80”, “3월”:”50”, “4월”:”78” … }”}”} |

**20. 소프트웨어 목록 조회**

다운로드 가능한 소프트웨어 목록을 조회한다.

**20.1. Request :: POST 방식**

HTTP URL = "/software"

- Parameter 형식 (Request Method = POST 방식)

|  |  |  |  |
| --- | --- | --- | --- |
| 파라미터명 | 타입 | 필수여부 | 설명 |
| 없음 |  |  |  |
|  |  |  |  |
|  |  |  |  |

**20.2. Response Format :: JSON 형태로 반환**

- 반환값 형식

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 엘리먼트 명 | Depth | 배열  구분 | 설명 | 값 구분 |
| status | 1 | N |  | success : 성공  error : 실패 |
| data | 1 | N | 전체 소프트웨어 목록 |  |
| sw\_name | 2 | Y | 소프트웨어 이름 |  |
| sw\_img | 2 | Y | 소프트웨어 이미지 경로 |  |
| sw\_detail | 2 | Y | 소프트웨어에 대한 설명 |  |
| sw\_path | 2 | Y | 소프트웨어 위치 경로 |  |

(버전, 회사명은 DB에 있고 화면에 없음/ 설명, 이미지는 디비에 있고 화면에 있음->디비 수정?)

**20.3. 샘플 JSON예제**

|  |
| --- |
| - 실패할 경우 샘플  { "status":"error", "data":"error" }  -성공할 경우 샘플  { "status":"success", "data": {  [“sw\_name”:“Microsoft Office Excel”,”sw\_img”:”img1”,”sw\_detail”:”엑셀”,”sw\_path”=”path1”],  [“sw\_name”:“Microsoft Office Word”,”sw\_img”:”img2”,”sw\_detail”:”워드”,”sw\_path”=”path2”],  [“sw\_name”:“Eclipse”,”sw\_img”:”img3”,”sw\_detail”:”이클립스”,”sw\_path”=”path3”]  }  } |

**21. 소프트웨어 다운로드**

소프트웨어를 다운로드한다.

**21.1. Request :: GET방식**

HTTP URL = ""

- Parameter 형식 (Request Method = GET 방식)

|  |  |  |  |
| --- | --- | --- | --- |
| 파라미터명 | 타입 | 필수여부 | 설명 |
| user\_id | String | 필수 | 소프트웨어를 다운로드 할 사용자의 id |
| sw\_name | String | 필수 | 다운로드 할 소프트웨어 명 |
|  |  |  |  |

**21.2. Response Format :: JSON 형태로 반환**

- 반환값 형식

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 엘리먼트 명 | Depth | 배열  구분 | 설명 | 값 구분 |
| status | 1 | N |  | success : 성공  error : 실패 |
| data | 1 | N | 정보 |  |

**19.3. 샘플 JSON예제**

|  |
| --- |
| - 실패할 경우 샘플  {"status":"error", "data":"error"}  -성공할 경우 샘플  {"status":"success", "data":””} |