[ 기본 프로세스 순서 ]

0. 가정

- 인원

> 24시간 가동되는 공장

> 3개의 근무조(주간: 07시~15시, 주야간: 15시~23시, 야간: 23시~07시) 형태로 1개 조가 8시간씩 근무 (3조 3교대)

> 라인 1개당 생산 공정에 1명, 품질검사 공정에 1명 배치 (1개 조) → 라인 1개 기준 하루에 총 6명 근무

> 실제로는 1개 조가 5일간 같은 시간대에 근무 후 1~2일 휴무를 가지고 그 다음 시간대로 이동하여 근무 (4조 3교대)

> 하지만 시뮬레이션을 위해 3개 조가 휴무 없이 계속 동일 시간대에 근무하도록 함 (3조 3교대)

> 라인 1개당 총 3개 조 6명이 필요하고, 공장 1개당 총 18명, 회사 전체로는 총 36명의 근무 인원 필요

- 창고

> 공장별로 1개의 자재 창고와 1개의 제품 창고 존재

\* 자재 창고는 각 자재(3개)를 구분해 보관하기 위해 3개의 section으로 구분

\* 제품 창고는 각 제품(3개)을 구분해 보관하기 위해 3개의 section으르 구분

> 각 라인별로 1개의 자재 임시 창고 존재

\* 생산을 위해 자재 창고에서 이동된 자재를 구분해 보관하기 위해 3개의 section으로 구분

- 자재 발주

> 제품 주문 수량의 110%를 만들 수 있는 자재 재고량을 유지하기 위해 각 자재를 필요한 수량만큼만 발주

- 자재 입고 및 반출

> 자재의 반출 (자재 창고 → 라인)은 생산 시작 순간 동시에 한 번에 이뤄짐

> 자재의 입고 (납품 업체 → 자재 창고)는 자재 발주 순간 동시에 한 번에 이뤄짐

> 자재 입고 전 수입 검사(납품된 자재 품질 검사)는 하지 않음

- 생산 지시

> 계획 수량은 불량률을 고려하여 주문 수량의 105%

- 생산 / 품질검사

> '생산 정보' 데이터와 '품질 검사 정보' 데이터는 같은 타임 스탬프(제품일련번호)를 가짐

> '품질 검사 정보'에서 2개의 검사(수치 검사, 홀 검사)가 독립적으로 이뤄지지만 그 데이터는 한 번에 같은 파일로 생성

- 제품 입고 / 반출 / 납품

> 정상 판정된 제품은 하나씩 제품 창고로 바로 입고됨

> 별도의 납품 절차는 없음

> 생산이 종료되고 계획 수량 이상이 제품 창고로 입고되면 바로 제품 창고에서 반출해 납품 완료됨

> 납품 후 남은 제품은 제품 창고에 보관해서 다음에 동일 제품 주문이 들어오면 해당 제품 재고만큼 빼고 생산

--------------------------------------------- [ 프로세스 시작 ] -------------------------------------------------------------

1. 제품 발주 - 고객사

- 제품 발주서 생성 (ERP)

2. 생산 의뢰 - 영업팀

- 생산 의뢰서 생성 (ERP)

--------------------------------------------- [ 프로젝트 시작 ] -------------------------------------------------------------

3. ERP에서 생성된 생산 의뢰서를 MES에서 자동으로 넘겨받는 작업을 주문서를 작성하는 것으로 대체

- 주문 테이블에 직접 값을 생성 → JAVA/JSP

- 주문서: 공장코드 / 주문번호 / 주문일자 / 품목코드 / 주문수량 / 납기일 / 납기완료일

4. 자재 발주 (원재료 구매) - 구매팀

- 자재량이 충분하면 자재 발주 없이 다음 단계로 진행

- 자재 현재 재고량을 고려하여 필요 수량만 발주

- 자재 발주서 생성 → JAVA/JSP

- 자재 발주서: 공장코드 / 발주번호 / 발주일자 / 품목코드 / 발주수량 / 납기일(???) / 입고완료일(???)

- 자재 발주서 생성시 자동으로 자재 창고에 발주 수량 입고 → JAVA/JSP

5. 생산 계획 수립 후 생산 지시 - 생산 관리팀

- 생산 지시서 (생산 테이블에 자료 추가) 생성 → JAVA/JSP

- 생산 지시서: 공장코드 / 라인코드 / 주문번호 / 생산지시번호 / 작업시작일 / 작업종료일 / 작업상태 / 계획수량

6. 생산 - 생산팀

- 생산 시작 버튼 클릭시 제품 생산에 필요한 자재가 자재 창고에서 한 번에 반출되어 라인으로 이동

- 생산 시작 버튼 클릭시 '생산 정보' 데이터를 백그라운드에서 주기적 (30초 or 1분?)으로 자동 생성 (PYTHON)

- 제품이 하나씩 생산될 때마다 라인에 보관 중인 자재도 필요 수량만큼 동시에 감소

- 마지막 제품이 생산되면 5번에서 생성된 생산 지시서(생산 테이블)의 상태를 완료로 바꿈

- 생성된 데이터를 csv 파일에 쓰기 (DB 서버 전송 오류에 대비한 local backup 개념)

- 생성된 데이터를 DB에 저장

7. 품질 검사 - 품질 보증팀

- 생산 시작 버튼 클릭시 '품질 검사 정보' 데이터를 백그라운드에서 주기적 (30초 or 1분?)으로 자동 생성 (PYTHON)

- 생산 제품 정보 = 품질 검사 정보

> 공장코드 / 라인코드 / 제품일련번호 / 수치측정값들 / 검사시간 / 검사결과 / 홀측정값들 / 검사시간 / 검사결과 /

불량유형(수치 / 홀) / 불량현상 (가로길이 / 세로길이 / 가로직진도 / 세로직진도 / 홀관련?) / 검사자

8. 제품 창고 입고 (재고) - 생산 관리팀

- 생산된 제품의 품질 검사 결과가 G(Good)일 경우 제품 창고로 하나씩 바로 입고됨

- 생산이 종료되고 계획 수량 이상이 제품 창고로 입고되면 그 시간을 3번 주문서의 납기 완료일로 정함

-------------------------------------------- [ 프로젝트 종료 ] -------------------------------------------------------------

9. 납품 - 영업팀

- 8번이 완료되면 자동으로 납품이 완료된 것으로 간주

-------------------------------------------- [ 프로세스 종료 ] -------------------------------------------------------------

[ 구현할 내용들]

'생산 실적 보고서' / '불량 보고서' - 일일 / 주간 / 월간 / 연간