

**Mini ERP**

## **4조 팀 보고서**

# 목차



프로젝트 개요

팀원 소개

사용 언어 및 TOOL

소프트웨어 운영환경

프로그램 설계과정

사용자 매뉴얼

# 프로젝트 개요



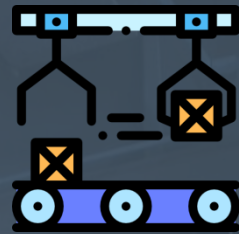
## 업무 관리

영업,물류,재고,거래  
생산 업무  
조회,수정,삭제



## 자동 계획수립

재고,건적,상품 관리  
및 계획 자동 수립



## 기계 관리

기계 명령 전달  
상태 조정



## 실시간 모니터링

실시간 자재 투입 현황  
및 기계 에러추적



## 그룹 메시지

공장 내부 인원들의  
소통을

# 팀원소개



이상권,김태우(도중하차)

업무 관리 담당



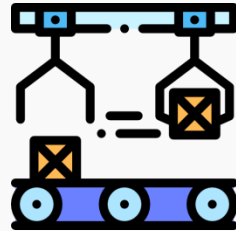
조성호

자동 계획 수립 담당



손종완

그룹 메시지 담당



지성욱

팀장 및 기계 관리



실시간 모니터링 담당

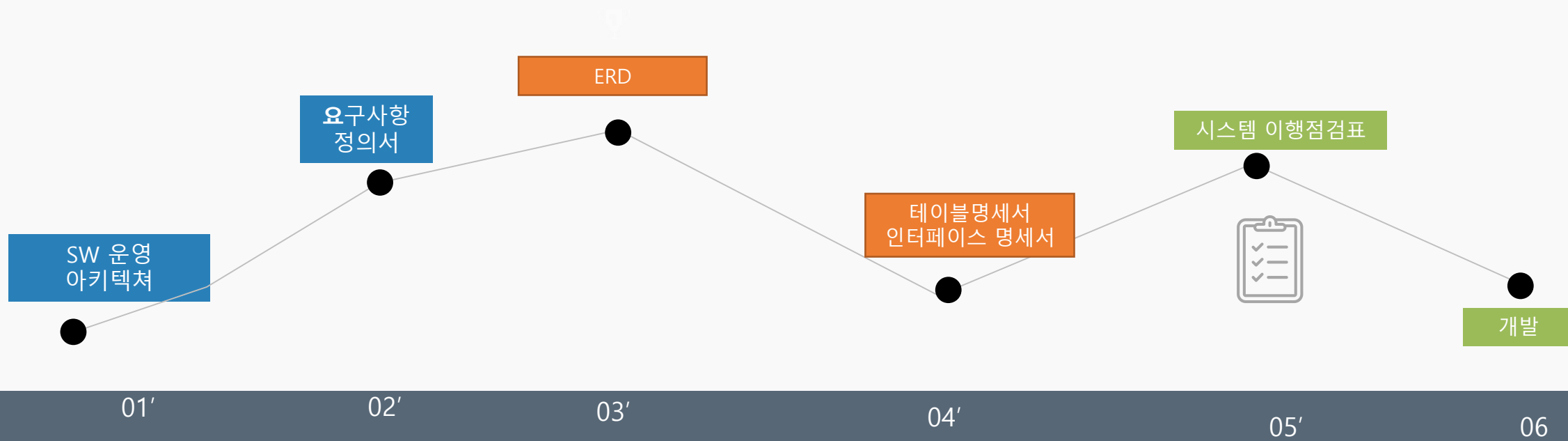
# 사용 언어 및 도구

라이선스 및 프레임워크 명		
IDE	Visual Studio	
Framework	.NET Framework 4.6.1	MS-EULA, BCL 하의 MS라이선스
	BarcodeLib	Apache-2.0 license
	<a href="https://github.com/barnhill/barcodelib">https://github.com/barnhill/barcodelib</a>	
	.NET Core Libraries <System.Drawing.Common>	The MIT License (MIT)
	<a href="https://github.com/dotnet/corefx/blob/master/LICENSE.TXT">https://github.com/dotnet/corefx/blob/master/LICENSE.TXT</a>	
	Tulpep.NotificationWindow	The Code Project Open License
	<a href="https://github.com/Tulpep/Notification-Popup-Window">https://github.com/Tulpep/Notification-Popup-Window</a>	
	Tulpep.NotificationWindow	
	<a href="https://github.com/Tulpep/Notification-Popup-Window">https://github.com/Tulpep/Notification-Popup-Window</a>	
	file-transfer-p2p	GPLv3
	<a href="https://github.com/crossdeck/file-transfer-p2p">https://github.com/crossdeck/file-transfer-p2p</a>	
DBMS	MS-SQL	
Server	Microsoft Azure SQL DataBase	

# 소프트웨어 운영 환경

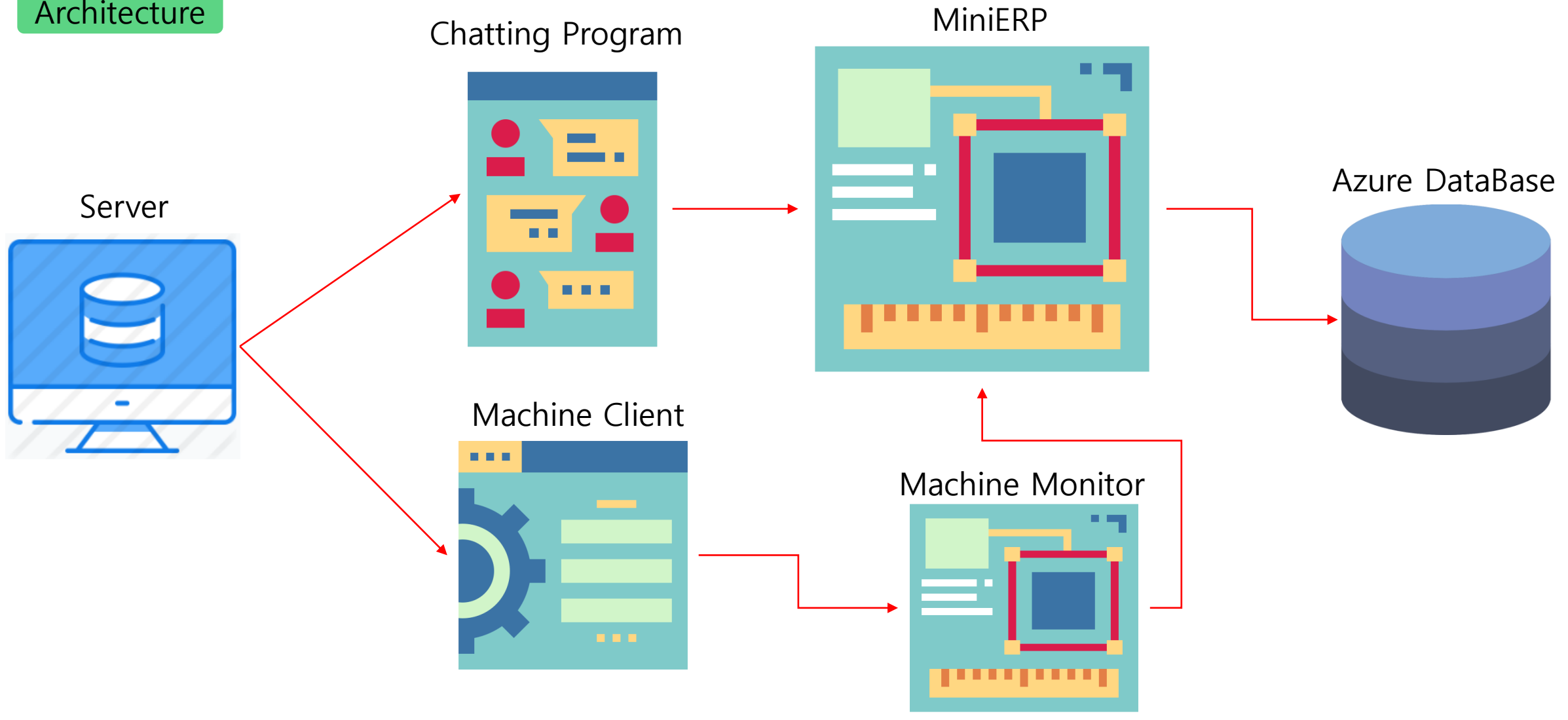
클라이언트 프로그램			서버 프로그램		머신 클라이언트	
구분	권장사양	최소사양	권장사양	최소사양	권장사양	최소사양
Windows	Windows 10	Windows 7	Windows 10	Windows 7	Windows 10	Windows 7
.NetFramework	4.6.1	4.6	4.6.1	4.6	4.6.1	4.6
CPU	I5 6200U	I5 6200U이상	I5 3700이상	I5 6200U이상	I5 6200U	I5 6200U이상
RAM	2기가	1기가	4기가	2기가	2기가	1기가
HDD	300G이상	100G이상	300G이상	100G이상	300G이상	100G이상

# 소프트웨어 개발 과정



# <소프트웨어 아키텍처>

## Architecture

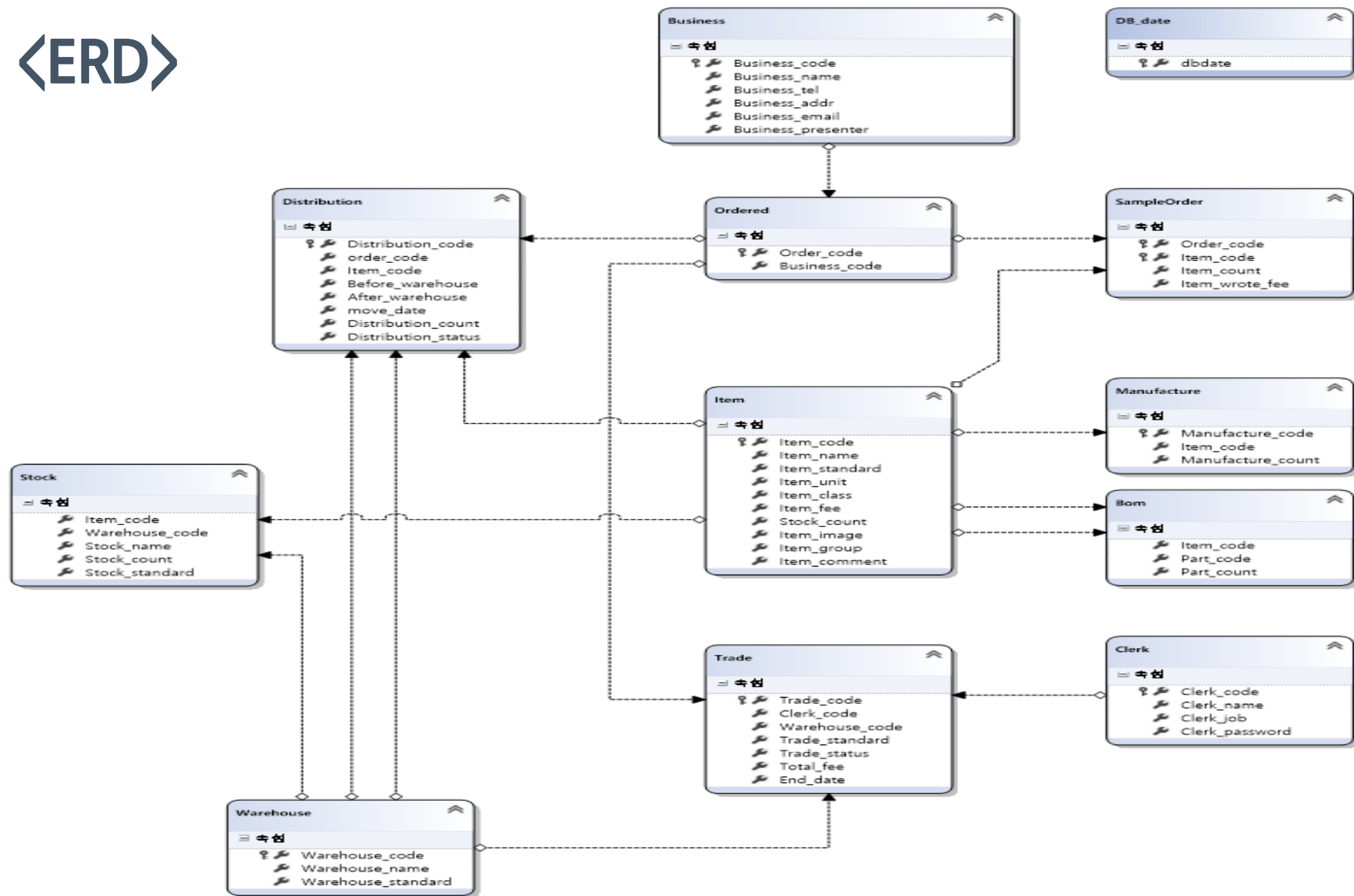




# 요구 사항 정의서

요구사항	설명
창고 관리	새 창고를 등록하는 경우 이미 등록되어 있는 창고코드는 중복하여 사용할 수 없습니다.
	등록되어 있는 창고의 이름, 구분을 수정할 수 있고 삭제할 수 있습니다.
주문 관리	거래처와 거래하기로 한 품목을 조회할 수 있습니다.
	주문번호로 해당 주문 건에 대해 품목과 수량을 확인할 수 있습니다.
	주문 받은 품목에 대해 재고수량을 확인하여 재고수량을 제외하고 추가로 생산 해야 할 수량을 계산하여 생산계획서를 만들고 파일로 출력할 수 있습니다.
사원 관리	새 사원을 등록하는 경우 이미 등록되어 있는 사원코드는 중복하여 사용할 수 없습니다.
	등록되어 있는 사원의 이름, 직급을 수정할 수 있고 삭제할 수 있습니다.
재고 관리	새 품목을 등록하는 경우 이미 등록되어 있는 품목코드는 중복하여 사용할 수 없습니다.
물류 관리	정해진 기간내의 물류이동을 조회할 수 있습니다.
	특정 날짜에 특정 창고에서 출고되거나 출고된 물품목록을 조회하고 파일로 출력할 수 있습니다.
	특정 날짜에 특정 창고로 입고되거나 입고된 물품목록을 조회하고 파일로 출력할 수 있습니다.
거래 관리	특정 기간에 이루어질 거래, 이루어진 거래에 대한 정보를 조회할 수 있고, 현재 진행중인 거래만 따로 모아서 조회할 수 있습니다.
	특정 거래에 대해 담당자와 거래상태를 변경할 수 있습니다.
소통	로그인한 사용자들에 한하여 서로 메시지와 파일을 주고 받을 수 있습니다.
기계 제어	기계들이 서버에 접속했는지 확인할 수 있고, 해당 기계에 대해 자재투입현황을 알 수 있습니다.
	기계의 상태를 감지하여 동작이 올바르게 않을 경우 원격으로 재 부팅할 수 있습니다.

**<ERD>**



# <테이블 명세서>

테이블정의서		Database	ERP_INFO					
		스키마	dbo					
테이블명		Item						
코멘트		품목 테이블						
Col	Name	Type	Length	Key	Null	Identity	Default	Comments
1	item_code	nvarchar	20	PK				해당 필드로 바번호 생성할 것
2	item_name	nvarchar	50					
3	item_standard	nvarchar	20		O			규격(박스, 롤, 컨테이너 등 )
4	item_unit	nvarchar	20		O			단위 ( 미터, 리터, 킬로그램 )
5	item_class	nvarchar	20					구분(완제품,자재,반제품...)
6	item_fee	int						품목 현재가격 (지속적으로 업데이트)
7	stock_count	int						품목 재고 갯수
8	item_image	Image			O			이미지
9	item_group	nvarchar	20		O			해당품목그룹 (식음료,컴퓨터셋트)
10	item_comment	nvarchar	Max		O			비고

# <테이블 명세서>

테이블정의서		Database	ERP_INFO						
		스키마	dbo						
테이블명		Business							
코멘트		거래처 테이블							
Col	Name	Type	Length	Key	Null	Identity	Default	Comments	
1	Business_code	nvarchar	20	PK				거래처 번호 (사용자 입력)	
2	Business_name	nvarchar	50					거래처 이름	
3	Business_tel	nvarchar	13					거래처 전화번호	
4	Business_addr	nvarchar	Max					거래처 주소	
5	Business_email	nvarchar	50		NULL				
6	Business_presenter	nvarchar	20					대표자명	

# <테이블 명세서>

테이블정의서		Database	ERP_INFO						
		스키마	dbo						
테이블명		Warehouse							
코멘트		창고 테이블							
Col	Name	Type	Length	Key	Null	Identity	Default	Comments	
1	WareHouse_code	nvarchar	20	PK				거래처창고는 거래처코드로 입력(그외 사용자입력)	
2	WareHouse_name	nvarchar	50					거래처창고는 거래처이름으로 입력	
3	WareHouse_standard	nvarchar	20					구분 ( 창고 / 공장 / 거래처)	

# <테이블 명세서>

테이블정의서		Database	ERP_INFO						
		스키마	dbo						
테이블명		Clerk							
코멘트		사원 테이블							
Col	Name	Type	Length	Key	Null	Identity	Default	Comments	
1	Clerk_code	nvarchar	20	PK				사번 (사용자 입력?)	
2	Clerk_name	nvarchar	50					사원명 ( 외노자 포함 )	
3	Clerk_job	nvarchar	30					직급	
4	Clerk_password	varbinary	32					비밀번호(SHA2_256)	

# <테이블 명세서>

테이블정의서		Database	ERP_INFO						
		스키마	dbo						
테이블명		Ordered							
코멘트		주문 테이블							
Col	Name	Type	Length	Key	Null	Identity	Default	Comments	
1	Order_code	nvarchar	20	PK				날짜_시퀀스번호(ex: 20170102_0001) (자동생성)	
2	Business_code	nvarchar	20	FK				거래처 번호	

# 〈테이블 명세서〉

테이블정의서		Database	ERP_INFO						
		스키마	dbo						
테이블명		SampleOrder							
코멘트		견적 정보 테이블							
Col	Name	Type	Length	Key	Null	Identity	Default	Comments	
1	Order_code	nvarchar	20	PK,FK				Order 테이블	
2	Item_code	nvarchar	20	PK,FK				Item 테이블	
3	Item_count	int						음수 불가 추후 기능구현시 설계시 참고	
4	Item_wrote_fee	int						주문시점 당시의 물품가격	



테이블정의서	Database	ERP_INFO						
	스키마	dbo						
테이블명	Trade							
코멘트	거래 테이블							
Col	Name	Type	Length	Key	Null	Identity	Default	Comments
1	Trade_code	nvarchar	20	PK,FK				Order 테이블
2	Clerk_code	nvarchar	20	FK				Clerk 테이블
3	Warehouse_code	nvarchar	20	FK	O			출고 or 입고 창고 / Warehouse 테이블
4	Trade_standard	nvarchar	20					거래구분 (구매/판매)
5	Trade_status	nvarchar	20				승인 전	진행사항 ( 승인전, 진행완료 등 ) ,
6	Total_fee	int						
7	End_Date	DateTime			O			거래종료일
비고								
1. 구분이 “판매” 일때 거래상태가 “승인,생산->출하대기 로 수정되면 거래의 건적들이 물류테이블에 등록된다. 2. 구분이 “판매”일때 거래상태가 “출하대기->출하완료 ” 로 수정되면 해당 거래와 관련된 물류의 상태가 “대기 -> 출고 ” 로 변경된다. 3. 구분이 “판매”일때 거래상태가 “출하완료->완료 ” 로 수정되면 해당 거래와 관련된 물류의 상태가 “완료 ” 로 변경된다. 4. 구분이 “판매”일때 거래상태가 “출하완료->반품 ” 로 수정되면 해당 거래와 관련된 물류의 상태가 “반품 ” 으로 변경된다. 5. 구분이 “구매 ” 일때 거래상태가 “승인전 -> 주문완료 ” 로 수정되면 해당 거래의 건적들이 물류테이블에 등록된다. 6. 구분이 “구매 ” 일때 거래상태가 “주문완료->입고대기 ” 로 수정되면 해당 거래와 관련된 물류의 상태가 “출고 ” 로 변경된다. 7. 구분이 “구매 ” 일때 거래상태가 “입고대기->완료 ” 로 수정되면 해당 거래와 관련된 물류의 상태가 “완료 ” 로 변경된다. 8. 구분이 “구매 ” 일때 거래상태가 “취소 ” 로 수정되면 해당 거래와 관련된 물류의 상태가 “취소 ” 로 변경된다.								

테이블정의서		Database	ERP_INFO					
		스키마	dbo					
테이블명		Distribution						
코멘트		물류 테이블						
Col	Name	Type	Length	Key	Null	Identity	Default	Comments
1	Distribution_code	nvarchar	20	PK		O		날짜_시퀀스번호(ex: 20170102_0001) (자동생성)
2	trade_code	nvarchar	20	FK	O			관련 거래코드
3	Item_code	nvarchar	20	FK				Item 테이블
4	Before_wareHouse	nvarchar	20	FK				이전 위치 / warehouse 테이블
5	After_wareHouse	nvarchar	20	FK				이후 위치 / warehouse 테이블
6	move_date	datetime			O			이동날짜
7	Distribution_Count	int						이동될 물류의 개수
8	distribution_status	nvarchar	20				대기	물류 상태(출고중, 완료) 등
비고								
1. 출고창고가 거래처가 아닐 때 "대기 " 상태로 등록되면 재고테이블에서 해당 창고의 품목재고를 물류갯수만큼 빼준다. 2. 출고창고가 거래처가 아닐 때 "출고 -> 완료 " 상태로 수정되면 입고 재고테이블에서 창고의 품목재고를 물류갯수만큼 더해준다. 3. 출고창고가 거래처가 아닐 때 "대기,출고 -> 취소,반품 " 상태로 수정되면 재고테이블에서 출고창고의 품목재고를 물류갯수만큼 더해준다.								

테이블정의서		Database		ERP_INFO				
		스키마		dbo				
테이블명		Stock						
코멘트		재고 테이블						
Col	Name	Type	Length	Key	Null	Identity	Default	Comments
1	item_code	nvarchar	20	FK				Item 테이블
2	Warehouse_code	nvarchar	20	FK				Warehouse 테이블
3	Stock_name	nvarchar	50					
4	Stock_count	int						재고 수량
5	Stock_standard	nvarchar	20		O			규격(박스,롤,컨테이너....)
비고								
1. 재고에 데이터가 추가/수정 되면 품목테이블에서 해당 품목코드의 재고갯수 변동, 2. 변동 이후 재고가 0개이면 해당 재고데이터 삭제								

# <테이블 명세서>

테이블정의서		Database	ERP_INFO					
		스키마	dbo					
테이블명		Bom						
코멘트		자재 청구 테이블						
Col	Name	Type	Length	Key	Null	Identity	Default	Comments
1	Item_code	nvarchar	20	FK				Item 테이블
2	part_code	nvarchar	20	Fk				Item 테이블
3	part_count	int						필요 개수

# <테이블 명세서>

테이블정의서		Database	ERP_INFO						
		스키마	dbo						
테이블명		Manufacture							
코멘트		생산 테이블							
Col	Name	Type	Length	Key	Null	Identity	Default	Comments	
1	Manufactor_code	nvarchar	20	PK		O		날짜_시퀀스번호(ex: 20170102_0001) (자동생성)	
2	Item_code	nvarchar	20	FK				Item 테이블	
3	Manufactor_count	int						생산 갯수	

# <테이블 명세서>

테이블정의서		Database	ERP_INFO					
		스키마	dbo					
테이블명		DB_date						
코멘트		시퀀스초기화를 위한 테이블						
Col	Name	Type	Length	Key	Null	Identity	Default	Comments
1	dbdate	date		PK		O		날짜

# <인터페이스 명세서>

INFO				견적 출력				
S/P 번호		1		S/P 명		GET_ORDER		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
품목번호	item_code	nvarchar	20	주문번호	order_code	nvarchar	20	not null
품목이름	item_name	nvarchar	50					
수량	item_count	int						
단가	item_written_fee	int						
처리내용	주문번호를 받아 견적을 출력하는 프로시저							
비고								
				완료일	2019-01-31	담당자	손종완	

# <인터페이스 명세서>

INFO		자재계획서 출력						
S/P 번호		2		S/P 명		GET_RESOURCES_PLAN		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
품목번호	part_code	nvarchar	20	주문번호	order_code	nvarchar	20	not null
품목이름	item_name	nvarchar	50					
수량	part_count	int						
규격	item_standard	nvarchar	20					
처리내용	각 견적의 품목번호들과 재고테이블의 품목번호를 조회하여 재고테이블의 재고수량만큼 견적서의 수량을 뺀 후 BOM을 조회하여 필요한 자재를 출력하는 프로시저							
비고								
				완료일	2019-01-31	담당자	지성욱	



# <인터페이스 명세서>

INFO		구매계획서 출력						
S/P 번호		3		S/P 명		GET_PURCHASE_PLAN		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
품목번호	part_code	nvarchar	20	주문 번호	order_code	nvarchar	20	not null
품목이름	item_name	nvarchar	50					
수량	part_count	int						
규격	item_standard	nvarchar	20					
처리내용	GET_RESOURCES_PLAN에서 반환된 데이터를 사용, stock_count만큼 part_count를 빼준 후 출력한다. 단, 뺀값이 1보다 작을 경우 출력값에서 제외							
비고								
				완료일	2019-02-07	담당자	지성욱	

# <인터페이스 명세서>

INFO		생산계획서 출력						
S/P 번호		4		S/P 명		GET_MANUFACTURE_PLAN		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
품목번호	item_code	nvarchar	20	주문번호	order_code	nvarchar	20	not null
품목이름	item_name	nvarchar	50					
수량	item_count	int						
규격	item_standard	nvarchar	20					
단가	item_wrote_fee	int						
처리내용	견적서 품목번호들로 재고테이블을 조회하여 견적서 item_count에 재고 stock_count를 빼준 값을 출력한다.							
비고								
				완료일	2019-02-01	담당자	이상권	

# <인터페이스 명세서>

INFO				제품생산서 출력				
S/P 번호		5		S/P 명		GET_MANUFACTURE_DOC		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
품목번호	item_code	nvarchar	20	생산날짜	manufacture_date	date		not null
품목이름	item_name	nvarchar	50	공장번호	warehouse_code	nvarchar	20	not null
수량	item_count	int						
규격	item_standard	nvarchar	20					
처리내용								
비고								
				완료일	2019-01-31	담당자	손종완	

# <인터페이스 명세서>

INFO		거래 조회 (주문조회)						
S/P 번호		6-1		S/P 명		GET_TRADE		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
주문번호	trade_code	nvarchar	20	거래날짜(LOW)	low_date	DATE		not null
거래처번호	business_code	nvarchar	20	거래날짜(MAX)	max_date	DATE		not null
거래처명	business_name	nvarchar	50	거래처번호	business_code	nvarchar	20	
품목요약	item_summary	nvarchar	50	품목번호	item_code	nvarchar	20	
총액	total_fee	int		페이지번호	PageNumber	int		
거래상태	trade_status	nvarchar	20	페이지당ROW	RowsPerPage	int		
거래구분	trade_standard	nvarchar	20					
종료일	end_date	datetime						
처리내용	거래날짜 범위내에 있는 거래들을 조회하여 출력한다.							
비고								
				완료일	2019-02-15	담당자	조성호	

# <인터페이스 명세서>

INFO		거래 조회 (주문조회)						
S/P 번호		6-1		S/P 명		GET_TRADE		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
주문번호	trade_code	nvarchar	20	거래날짜(LOW)	low_date	DATE		not null
거래처번호	business_code	nvarchar	20	거래날짜(MAX)	max_date	DATE		not null
거래처명	business_name	nvarchar	50	거래처번호	business_code	nvarchar	20	
품목요약	item_summary	nvarchar	50	품목번호	item_code	nvarchar	20	
총액	total_fee	int		페이지번호	PageNumber	int		
거래상태	trade_status	nvarchar	20	페이지당ROW	RowsPerPage	int		
거래구분	trade_standard	nvarchar	20					
종료일	end_date	datetime						
처리내용	거래날짜 범위내에 있는 거래들을 조회하여 출력한다.							
비고								
				완료일	2019-02-15	담당자	조성호	

# <인터페이스 명세서>

INFO				진행중인 구매/판매 조회				
S/P 번호		6-2		S/P 명		GET_PROG_TRADE		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
거래번호	trade_code	nvarchar	20	거래구분	trade_standard	nvarchar	20	not null
거래처번호	business_code	nvarchar	20	페이지번호	PageNumber	int		
거래처명	business_name	nvarchar	50	페이지당ROW	RowsPerPage	int		
품목요약	item_summary	nvarchar	50					
금액	total_fee	int						
창고번호	warehouse_code	nvarchar	20					
입고(출고)창고 이름	warehouse_name	nvarchar	50					
처리내용	구분과 거래상태를 확인하여 현재 진행중인 구매/판매를 조회한다.							
비고								
				완료일	2019-02-15	담당자	조성호	

# <인터페이스 명세서>

INFO				거래 수정					
S/P 번호				6-3		S/P 명		MODIFY_TRADE	
송신				수신					
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL	
				거래번호	trade_code	nvarchar	20	not null	
				사원번호	clerk_code	nvarchar	20		
				입/출고창고번호	warehouse_code	nvarchar	20		
				거래상태	trade_status	nvarchar	20		
처리내용	거래 테이블을 수정한다. 또한 거래상태가 변경되면 그에 맞게 유기적으로 수정되는건 트리거로 처리								
비고	거래 테이블 트리거 잘만들자!								
				완료일	2019-02-25	담당자	조성호		

# <인터페이스 명세서>

INFO		물류 조회						
S/P 번호		7-1		S/P 명		GET_DISTRIBUTION		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
물류번호	distribution_code	nvarchar	20	날짜범위(LOW)	low_date	DATE		not null
출고창고	before_warehouse	nvarchar	20	날짜범위(MAX)	max_date	DATE		not null
입고창고	after_warehouse	nvarchar	20	출고창고검색어	before_warehouse	nvarchar	20	
품목번호	item_code	nvarchar	20	입고창고검색어	after_warehouse	nvarchar	20	
품목명	item_name	nvarchar	50	품목번호검색	item_code	nvarchar	20	
개수	distribution_count	int		물류상태	distribution_status	nvarchar	20	
물류상태	distribution_status	nvarchar	20					
이동날짜	move_date	date						
처리내용	물류이동날짜(move_date) 범위내에 있는 거래들을 조회하여 출력한다. 입/출/품목번호 검색은 사용자에게 따로 검색목록을 보여주어 선택하도록 하게 한다.							
비고	수신의 distribution_status는 기본값 '전체'로 설정 추후 관련 테이블이나 함수 나올경우 수정							
					완료일	2019-02-11	담당자	이상권



# <인터페이스 명세서>

INFO				물류 등록				
S/P 번호		7-2		S/P 명		SET_DISTRIBUTION		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
				관련주문 번호	order_code	nvarchar	20	
				출고창고	before_warehouse	nvarchar	20	not null
				입고창고 문자열	after_warehouse	nvarcahr	MAX	not null
				품목 문자열	item_code	nvarchar	MAX	not null
				이동날짜	move_date	nvarchar	MAX	not null
				이동 갯수	distribution_count	nvarchar	MAX	not null
처리내용	문자열을 잘라서 물류테이블에 등록, 구분자   (insert트리거를 통해 재고관리)							
비고								
				완료일	2019-02-26	담당자	조성호	

# <인터페이스 명세서>

INFO				물류 수정				
S/P 번호		7-3		S/P 명		MODIFY_DISTRIBUTION		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
				물류번호	distribution_code	nvarchar	20	
				출고창고	before_warehouse	nvarchar	20	
				입고창고	after_warehouse	nvarchar	20	
				품목번호	item_code	nvarchar	20	
				개수	distribution_count	int		
				이동날짜	move_date	date		
				물류상태	distribution_status	nvarchar	20	
처리내용	물류번호로 조회하여 데이터를 수정한다. 상태가 변경되면 그에 따라 재고테이블이 수정된다. (대기 -> 출고) 출고창고가 거래처창고가 아니라면 해당 창고의 물류품목재고가 줄어듦. (출고 -> 완료) 입고창고가 거래처창고가 아니라면 해당 창고의 물류품목재고가 증가함. (대기,출고->취소) 출고창고가 거래처창고가 아니라면 해당 창고의 물류품목재고가 증가함							
비고	재고 관련부분 트리거로 처리 (update트리거)							
				완료일	2019-02-25	담당자	조성호	

# <인터페이스 명세서>

INFO				주문 작업				
S/P 번호		8-1		S/P 명		SET_ORDER		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
				거래처번호	business_code	nvarchar	20	
				사원번호	clerk_code	nvarchar	20	
				창고번호	warehouse_code	nvarchar	20	
				거래구분	trade_standard	nvarchar	20	
				품목번호	item_code	nvarchar	MAX	
				품목개수	item_count	nvarchar	MAX	
처리내용	거래처번호를 통해 주문테이블 데이터 저장, 해당 order_code와 받아온 품목번호,개수를 건적저장프로시저 매개변수로 보냄. 마지막으로 order_code와 사원번호,창고번호를 거래등록프로시저로 보냄							
비고	품목번호와 품목개수를 각각  로 구분한 문자열로 보낸다. (예) item_code: a001 a002 b003 item_count: 1 7 33							
				완료일	2019-01-31	담당자	조성호	

# <인터페이스 명세서>

INFO		주문작업 - 건적등록						
S/P 번호		8-2-1		S/P 명		SET_ORDERLIST		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
				주문번호	order_code	nvarchar	20	
				품목번호	item_code	nvarchar	MAX	
				품목개수	item_count	nvarchar	MAX	
처리내용	품목번호와 품목개수는 문자열로 받아 구분자를 통해 나눠서 하나하나 테이블에 저장한다							
비고	나누는건 따로 정의한다. (SPLIT_SAMPLEORDER)							
				완료일	2019-01-31	담당자	조성호	

# <인터페이스 명세서>

INFO		견적정보 나누기					
FUNC 번호		8-2-2		FUNC 명		SPLIT_SAMPLEORDER	
송신				수신			
한글명	영문명	Type	길이	한글명	영문명	Type	길이
주문번호	order_code	nvarchar	20	주문번호	order_code	nvarchar	20
품목번호	item_code	nvarchar	20	품목번호	item_code	nvarchar	MAX
품목개수	item_count	int		품목개수	item_count	nvarchar	MAX
품목가격	item_wrote_fee	int					
처리내용	품목번호와 품목개수는 문자열로 받아 구분자(,)를 통해 나누고 품목번호를 통해 품목가격도 반환받는다. 결과로 송신에 써있는 컬럼을 가지고있는 테이블이 출력된다.						
비고							
				완료일	2019-01-31	담당자	조성호

# <인터페이스 명세서>

INFO				주문작업 - 거래등록					
S/P 번호				8-3		S/P 명		SET_TRADE	
송신				수신					
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL	
				주문번호	trade_code	nvarchar	20		
				사원번호	clerk_code	nvarchar	20		
				창고번호	warehouse_code	nvarchar	20		
				거래구분	trade_standard	nvarchar	20		
처리내용	거래테이블에 데이터를 저장한다.								
비고	그 외의 값들은 defalut값으로 처리								
				완료일	2019-01-31	담당자	조성호		

# <인터페이스 명세서>

INFO				견적수정				
S/P 번호		9		S/P 명		MODIFY_ORDERLIST		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
				주문번호	order_code	nvarchar	20	
				품목번호	item_code	nvarchar	MAX	
				품목개수	item_count	nvarchar	MAX	
처리내용	품목번호와 품목개수는 문자열로 받고 견적 정보 나누기를 통해 구분자를 기준으로 나누고 품목번호를 통해 품목가격도 반환받는다. 그 후 주문번호를 이용해 견적테이블에서 해당 견적데이터를 지우고 데이터를 다시 추가한다. 마지막으로 거래테이블의 total_fee도 수정한다.							
비고	나누는건 함수로 처리한다. (SPLIT_ITEM)							
					완료일	2019-02-01	담당자	조성호

# <인터페이스 명세서>

INFO				BOM 등록				
S/P 번호		10		S/P 명		INSERT_BOM		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
				품목번호	item_code	nvarchar	20	not null
				파트번호	part_code	nvarchar	MAX	not null
				파트개수	part_count	nvarchar	MAX	not null
처리내용	BOM 테이블에 데이터를 입력하는 프로시저							
비고								
				완료일	2019-02-26	담당자	이상권	



# <인터페이스 명세서>

INFO				재고 변동(저장,수정 등 포함)				
S/P 번호			11	S/P 명		SET_STOCK_ITEM		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
				품목번호	item_code	nvarchar	20	not null
				창고번호	warehouse_code	nvarchar	20	not null
				재고개수	stock_count	int		not null
처리내용	품목번호,창고번호 같은게 있다면 재고개수 + / 재고테이블에 Insert							
비고	ITEM테이블과의 무결성은 트리거로 처리							
				완료일	2019-02-26	담당자	조성호	

# <인터페이스 명세서>

INFO				거래처 등록				
S/P 번호		0-1-1		S/P 명		InsertBusiness		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
				거래처번호	Business_code	nvarchar	20	not null
				거래처이름	Business_name	nvarchar	50	not null
				거래처연락처	Business_tel	nvarchar	13	not null
				거래처주소	Business_addr	nvarchar	MAX	not null
				거래처이메일	Business_email	nvarchar	50	
				거래처대표	Business_presenter	nvarchar	20	not null
처리내용	입력된 데이터를 거래처 테이블에 저장한다. 또한, 거래처번호와 거래처이름을 이용해 참고테이블에 데이터를 추가한다.							
비고								
				완료일	2019-02-01	담당자	지성욱	

# <인터페이스 명세서>

INFO				거래처 수정				
S/P 번호		0-1-2		S/P 명		UpdateBusiness		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
				거래처번호	Business_code	nvarchar	20	not null
				거래처이름	Business_name	nvarchar	50	
				거래처연락처	Business_tel	nvarchar	13	
				거래처주소	Business_addr	nvarchar	MAX	
				거래처이메일	Business_email	nvarchar	50	
				거래처대표	Business_presenter	nvarchar	20	
처리내용	입력된 데이터의 거래처번호를 통해 해당 데이터를 찾아 수정한다.							
비고								
				완료일	2019-01-29	담당자	지성욱	

# <인터페이스 명세서>

INFO				거래처 삭제				
S/P 번호		0-1-3		S/P 명		DeleteBusiness		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
				거래처번호	Business_code	nvarchar	20	not null
처리내용	거래처번호를 통해 해당 데이터를 찾아 삭제한다.							
비고								
				완료일	2019-01-29	담당자	지성욱	

# <인터페이스 명세서>

INFO				품목 등록				
S/P 번호		0-2-1		S/P 명		InsertItem		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
				품목번호	item_code	nvarcahr	20	not null
				품목이름	item_name	nvarchar	50	not null
				규격	item_standard	nvarchar	20	
				단위	item_unit	nvarchar	20	
				구분	item_class	nvarchar	20	not null
				품목가격	item_fee	int		not null
				품목사진	item_image	image		
				품목그룹	item_group	nvarcahr	20	
				비고	item_comment	nvarchar	MAX	
처리내용	입력된 데이터를 품목 테이블에 저장한다.							
비고								
				완료일	2019-01-29	담당자	손종완	

# <인터페이스 명세서>

INFO		품목 수정						
S/P 번호		0-2-2		S/P 명		UpdateItem		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
				품목번호	item_code	nvarcahr	20	not null
				품목이름	item_name	nvarchar	50	
				규격	item_standard	nvarchar	20	
				단위	item_unit	nvarchar	20	
				구분	item_class	nvarchar	20	
				품목가격	item_fee	int		
				품목사진	item_image	image		
				품목그룹	item_group	nvarcahr	20	
				비고	item_comment	nvarchar	MAX	
처리내용	입력된 데이터의 품목번호를 통해 해당 데이터를 찾아 수정한다. 추가로 수정된 데이터를 재고테이블과 견적테이블,견적BOM테이블도 수정한다.							
비고	가격 수정시 견적서의 가격, 거래테이블의 총가격 수정							
				완료일	2019-02-01	담당자	손종완	

# <인터페이스 명세서>

INFO				품목 삭제				
S/P 번호		0-2-3		S/P 명		DeleteItem		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
				품목번호	item_code	nvarchar	20	not null
처리내용	품목번호를 통해 해당 데이터를 찾아 삭제한다.							
비고								
				완료일	2019-01-29	담당자	손종완	

# <인터페이스 명세서>

INFO		사원 등록						
S/P 번호		0-3-1		S/P 명			InsertClerk	
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
				사원번호	clerk_code	nvarchar	20	not null
				사원이름	clerk_name	nvarchar	50	not null
				사원직급	clerk_job	nvarchar	30	not null
				비밀번호	clerk_password	nvarchar	20	not null
처리내용	입력된 데이터를 사원 테이블에 저장한다. 비밀번호는 암호화하여 저장							
비고								
				완료일	2019-02-19	담당자	이상권	



# <인터페이스 명세서>

INFO				사원 수정					
S/P 번호				0-3-2		S/P 명		UpdateClerk	
송신				수신					
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL	
				사원번호	clerk_code	nvarchar	20	not null	
				사원이름	clerk_name	nvarchar	50	not null	
				사원직급	clerk_job	nvarchar	30	not null	
처리내용	입력된 데이터의 사원번호를 통해 해당 데이터를 찾아 수정한다.								
비고									
				완료일	2019-02-27	담당자	이상권		

# <인터페이스 명세서>

INFO				사원 삭제				
S/P 번호		0-3-3		S/P 명		DeleteClerk		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
				사원번호	clerk_code	nvarchar	20	not null
처리내용	사원번호를 통해 해당 데이터를 찾아 삭제한다.							
비고								
				완료일	2019-01-29	담당자	손종완	

# <인터페이스 명세서>

INFO				창고 등록				
S/P 번호		0-4-1		S/P 명		InsertWarehouse		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
				창고번호	Warehouse_code	nvarchar	20	not null
				창고이름	Warehouse_name	nvarchar	50	not null
				창고구분	Warehouse_standard	nvarchar	20	not null
처리내용	입력된 데이터를 창고 테이블에 저장한다.							
비고								
				완료일	2019-01-29	담당자	이상권	

# <인터페이스 명세서>

INFO				창고 수정				
S/P 번호		0-4-2		S/P 명		UpdateWarehouse		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
				창고번호	Warehouse_code	nvarchar	20	not null
				창고이름	Warehouse_name	nvarchar	50	not null
				창고구분	Warehouse_standard	nvarchar	20	not null
처리내용	입력된 데이터의 창고번호를 통해 해당 데이터를 찾아 수정한다.							
비고								
				완료일	2019-01-29	담당자	이상권	

# <인터페이스 명세서>

INFO				창고 삭제				
S/P 번호		0-4-3		S/P 명		DeleteWarehouse		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
				창고번호	Warehouse_code	nvarchar	20	not null
처리내용	창고번호를 통해 해당 데이터를 찾아 삭제한다.							
비고								
				완료일	2019-01-29	담당자	이상권	

# <인터페이스 명세서>

INFO				입고 확인서				
S/P 번호		12		S/P 명		GET_Inport		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
입고창고	after_warehouse	nvarchar	20	입고날짜	move_date	date	20	
보낸창고	before_warehouse	nvarchar	20	보낸창고번호	before_warehouse	nvarchar	20	
품목명	Item_name	nvarchar	50					
수량	Distribution_count	int						
규격	Item_standard	nvarchar	20					
처리내용	입고날짜에 보낸창고/품목이름 검색							
비고								
				완료일		담당자	손종완	

# <인터페이스 명세서>

INFO				출하 증명서				
S/P 번호		13		S/P 명		GET_Export		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
출하창고	Before_warehouse	nvarchar	20	출고날짜	move_date	date	20	
도착창고	After_warehouse	nvarchar	20	도착창고번호	after_warehouse	nvarchar	20	
품목명	Item_name	nvarchar	50					
수량	Distribution_count	int						
규격	Item_standard	nvarchar	20					
처리내용	출고날짜와 도착창고번호를 입력받아 해당날짜에 예정된 해당							
비고	NO 처리?							
				완료일		담당자	손종완	

# <인터페이스 명세서>

INFO		시퀀스 초기화						
S/P 번호		14		S/P 명		RESET_SEQUENCE		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	
처리내용	현재 날짜와 기준날짜를 비교하여 날짜가 다르다면 시퀀스들을 초기화 한다. 또한, 이 경우 기준날짜를 현재날짜로 수정한다.							
비고								
				완료일	2019-02-07	담당자	조성호	



# <인터페이스 명세서>

INFO				거래처 목록 출력				
S/P 번호		15		S/P 명		GET_BUSINESS		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
거래처코드	Business_code	nvarchar	20	거래처코드	Business_code	nvarchar	20	O
거래처명	Business_name	nvarchar	50	거래처명	Business_name	nvarchar	50	O
거래처 연락처	Business_tel	nvarchar	13	거래처 연락처	Business_tel	nvarchar	13	O
거래처 주소	Business_addr	nvarchar	MAX	거래처 이메일	Business_email	nvarchar	50	O
거래처 이메일	Business_email	nvarchar	50	대표자명	Business_presenter	nvarchar	20	O
대표자명	Business_presenter	nvarchar	20					
처리내용	조건에 맞는 거래처코드, 거래처명, 거래처연락처, 거래처주소, 거래처이메일, 대표자명 출력							
비고	조건은 모두 Nullable							
				완료일	2019-02-14	담당자	이상권	

# <인터페이스 명세서>

INFO				창고(공장)목록 전체 출력				
S/P 번호		16		S/P 명		GET_WAREHOUSE		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
창고(공장)코드	Warehouse_code	nvarchar	20	창고(공장)코드	Warehouse_code	nvarchar	20	O
창고(공장)명	Warehouse_name	nvarchar	50	창고(공장)명	Warehouse_name	nvarchar	50	O
구분(창고 or 공장)	Warehouse_standard	nvarchar	20	구분(창고 or 공장)	Warehouse_standard	nvarchar	20	O
처리내용	조건에 맞는 창고(공장)코드, 창고(공장)명, 구분(창고 or 공장) 출력							
비고	조건은 모두 Nullable							
				완료일	2019-02-20	담당자	이상권	

# <인터페이스 명세서>

INFO		품목 테이블 출력						
S/P 번호		17		S/P 명		GET_ITEM		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
품목번호	Item_code	nvarchar	20	페이지번호	PageNumber	int		
품목이름	Item_name	nvarchar	50	페이지당ROW	RowsPerPage	int		
규격	Item_standard	nvarchar	20	검색어	search	nvarchar	50	
단위	Item_unit	nvarchar	20					
구분	Item_class	nvarchar	20					
가격	Item_fee	int						
재고갯수	Stock_count	int						
이미지	Item_image	image						
그룹명	Item_group	nvarchar	20					
비고	Item_comment	nvarchar	MAX					
처리내용	품목테이블 데이터 전체 출력 혹은 검색							
비고								
				완료일	2019-02-20	담당자	조성호	

# <인터페이스 명세서>

INFO				사원코드 중복 체크				
S/P 번호		19		S/P 명		Check_ClerkData		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
				사원코드	clerk_code	nvarchar	20	not null
처리내용	입력받은 사원코드로 Clerk테이블을 조회합니다.							
비고								
				완료일	2019-02-26	담당자	이상권	

# <인터페이스 명세서>

INFO				창고(공장)코드 중복 체크				
S/P 번호		20		S/P 명		Check_WarehouseData		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
				창고(공장)코드	warehouse_code	nvarchar	20	not null
처리내용	입력받은 창고코드로 Warehouse테이블을 조회합니다.							
비고								
				완료일	2019-02-26	담당자	이상권	

# <인터페이스 명세서>

INFO				거래처코드 중복 체크				
S/P 번호		21		S/P 명		Check_BusinessData		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
				거래처코드	business_code	nvarchar	20	not null
처리내용	입력받은 거래처코드로 Business테이블을 조회합니다.							
비고								
				완료일	2019-02-26	담당자	이상권	

# <인터페이스 명세서>

INFO				Bom테이블 내용 출력				
S/P 번호		22		S/P 명		GET_BOM		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
				품목코드	item_code	nvarchar	20	O
				품목명	item_name	nvarchar	50	O
처리내용	입력받은 아이템코드와 아이템이름으로 Item테이블과 Bom테이블을 조회하여 품목코드와 품목명을 출력합니다.							
비고	중복된 데이터는 출력되지 않습니다.							
				완료일	2019-02-27	담당자	이상권	

# <인터페이스 명세서>

INFO				선택한 품목에 대한 파트코드, 이름, 소요량 출력				
S/P 번호		23		S/P 명		GET_BomDetail		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
				품목코드	item_code	nvarchar	20	not null
처리내용	입력받은 아이템코드를 이용해 해당 아이템을 만드는데 필요한 파트의 코드, 이름, 소요량을 출력합니다.							
비고								
				완료일	2019-02-27	담당자	이상권	



# <인터페이스 명세서>

INFO				Bom테이블 데이터 삭제					
S/P 번호				24	S/P 명		DeleteBom		
송신				수신					
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL	
				품목코드	item_code	nvarchar	20	not null	
처리내용		입력받은 아이템코드에 해당하는 Bom테이블의 데이터를 삭제합니다.							
비고									
					완료일	2019-02-27	담당자	이상권	

# <인터페이스 명세서>

INFO				Bom테이블 조회				
S/P 번호		25		S/P 명		GET_BOM_SIMPLE		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
				품목코드 또는 품목명	code_name	nvarchar	20	O
처리내용		Bom테이블과 Item테이블을 조인한 뷰에서 입력받은 code_name을 이용하여 해당하는 데이터를 출력합니다.						
비고								
					완료일	2019-02-27	담당자	이상권

# <인터페이스 명세서>

INFO				재고 조회				
S/P 번호		26		S/P 명		GET_STOCK		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
				창고번호	warehouse_code	nvarchar	20	not null
				품목번호	item_code	nvarchar	20	
처리내용	특정 창고의 재고를 조회한다. 품목번호로 검색가능							
비고								
					완료일	2019-02-27	담당자	조성호

# <인터페이스 명세서>

INFO				아이템(1개) 조회				
S/P 번호			27		S/P 명		GET_ItemByCode	
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
품목코드	nvarchar		20	품목코드	item_code	nvarchar	20	not null
품목명	nvarchar		50					
규격	nvarchar		20					
단위	nvarchar		20					
구분	nvarchar		20					
품목가격	int							
이미지	Image							
품목그룹	nvarchar		20					
비고	nvarchar		Max					
처리내용	품목코드를 입력받아 그 품목코드와 정확하게 일치하는 아이템(1개)의 정보를 얻어옵니다.							
비고								
					완료일	2019-02-27	담당자	조성호

# <인터페이스 명세서>

INFO				매출/매입전표 출력				
S/P 번호		28		S/P 명		PROC_STATEMENT		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
품목번호	item_code	nvarchar	20	주문번호(거래번호)	order_code	nvarchar	20	
품목이름	item_name	nvarchar	50					
품목규격	item_standard	nvarchar	20					
품목갯수	item_count	int						
품목단가	item_wrote_fee	int						
처리내용		주문번호를 통해 매입/매출전표 데이터를 출력						
비고		그외 데이터는 프로그램에서 얻기						
					완료일	2019-03-01	담당자	조성호

# <인터페이스 명세서>

INFO				특정 사원 조회				
S/P 번호		0-3-4		S/P 명		Clerk_password_check		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
사원번호	clerk_code	nvarchar	20		clerk_code	nvarchar	20	not null
사원이름	clerk_name	nvarchar	50		clerk_password	nvarchar	20	
사원직급	clerk_job	nvarchar	30					
사원비밀번호	clerk_password	nvarchar	20					
처리내용		사원이 입력한 사원번호와 사원비밀번호와 일치하는 사원의 칼럼들을 갖고옴						
비고								
					완료일	2019-01-29	담당자	손종완

# <인터페이스 명세서>

INFO		사원 조회						
S/P 번호		0-3-5		S/P 명		Get_Clerk		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
사원번호	clerk_code	nvarchar	20	사원이름	clerk_name	nvarchar	50	O
사원이름	clerk_name	nvarchar	50	사원직급	clerk_job	nvarchar	30	O
사원직급	clerk_job	nvarchar	30					
사원비밀번호	clerk_password	nvarchar	20					
처리내용	조건에 맞는 사원목록 출력							
비고	조건은 모두 Nullable							
				완료일	2019-02-19	담당자	이상권	

# <인터페이스 명세서>

INFO				물류테이블 트리거(INSERT, UPDATE)				
TRG 번호		1		S/P 명		TRG_DISTRIBUTION		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
처리내용	1. 출고창고가 거래처가 아닐 때 “대기 ” 상태로 등록되면 재고테이블에서 해당 창고의 품목재고를 물류갯수만큼 빼준다. 2. 출고창고가 거래처가 아닐 때 “출고 -> 완료 ” 상태로 수정되면 입고 재고테이블에서 창고의 품목재고를 물류갯수만큼 더해준다. 3. 출고창고가 거래처가 아닐 때 “대기,출고 -> 취소,반품 ” 상태로 수정되면 재고테이블에서 출고창고의 품목재고를 물류갯수만큼 더해준다.							
비고								
				완료일	2019-02-20	담당자	조성호	



# <인터페이스 명세서>

INFO				거래테이블 트리거(UPDATE)				
TRG 번호		2		S/P 명		TRG_TRADE		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
처리내용	1. 구분이 “판매”일때 거래상태가 “승인,생산->출하대기 ” 로 수정되면 거래의 건적들이 물류테이블에 등록된다. 2. 구분이 “판매”일때 거래상태가 “출하대기->출하완료 ” 로 수정되면 해당 거래와 관련된 물류의 상태가 “대기 -> 출고 ” 로 변경된다. 3. 구분이 “판매”일때 거래상태가 “출하완료->완료 ” 로 수정되면 해당 거래와 관련된 물류의 상태가 “완료 ” 로 변경된다. 4. 구분이 “판매”일때 거래상태가 “출하완료->반품 ” 로 수정되면 해당 거래와 관련된 물류의 상태가 “반품 ” 으로 변경된다. 5. 구분이 “구매 ” 일때 거래상태가 “승인전 -> 주문완료 ” 로 수정되면 해당 거래의 건적들이 물류테이블에 등록된다. 6. 구분이 “구매 ” 일때 거래상태가 “주문완료->입고대기 ” 로 수정되면 해당 거래와 관련된 물류의 상태가 “출고 ” 로 변경된다. 7. 구분이 “구매 ” 일때 거래상태가 “입고대기->완료 ” 로 수정되면 해당 거래와 관련된 물류의 상태가 “완료 ” 로 변경된다. 8. 구분이 “구매 ” 일때 거래상태가 “취소 ” 로 수정되면 해당 거래와 관련된 물류의 상태가 “취소 ” 로 변경된다.							
비고								
				완료일	2019-02-20	담당자	조성호	

# <인터페이스 명세서>

INFO				재고 테이블 트리거 (INSERT , UPDATE)				
TRG 번호		3		S/P 명		TRG_Stock_Item_stock_count		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
처리내용	재고에 데이터가 추가/수정 되면 품목테이블에서 해당 품목코드의 재고갯수 변동, 변동 이후 재고가 0개이면 해당 재고데이터 삭제							
비고								
				완료일	2019-02-20	담당자	조성호	

# <인터페이스 명세서>

INFO		DATE_SEQUENCE 형식의 문자열을 DATE로 반환하는 함수					
FUNC 번호		0		FUNC 명		SPLIT_SAMPLEORDER	
송신				수신			
한글명	영문명	Type	길이	한글명	영문명	Type	길이
반환되는날짜	return_date	date		date_code	date_code	nvarchar	20
처리내용	매개변수로 받은 date_sequence 문자열을 date 타입으로 반환해준다.						
비고	매개변수는 반드시 'yyyymmdd_sequence'형식이어야 한다.						
					완료일	2019-01-31	담당자
							조성호

# <인터페이스 명세서>

INFO				상태구분 출력				
FUNC 번호		1		FUNC 명		GET_STANDARD		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
상태번호	no	int		상태구분이름	name	nvarchar	30	not null
상태명	standard	nvarchar	20					
처리내용		상태구분이름을 통해 해당 상태의 목록 테이블을 반환하는 함수						
비고		상태구분이름의 매개변수로 아래의 것을 주어야한다.						
		판매: sell						
		구매: buy						
		물류: distribution						
		창고: warehouse						
					완료일	2019-02-15	담당자	조성호

# <인터페이스 명세서>

INFO		테이블 로우 갯수 출력						
FUNC 번호		2		FUNC 명		GET_TABLEROW		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
로우 갯수	row_count	int		테이블이름	table_name	nvarchar	30	not null
처리내용	테이블 이름을 받아 해당 테이블의 로우 갯수를 반환하는 함수							
비고	SYSINDEXES와 SYSOBJECTS를 조인해서 만들면 좋다. 단순히 COUNT같은것을 쓰는것보다 훨씬 빠르고 DB에 부하가 덜걸린다. 참고: 땅쿠 네이버 블로그, ms-sql 테이블 레코드수 빨리 취하기, <a href="https://m.blog.naver.com/tank51882/60144213744">https://m.blog.naver.com/tank51882/60144213744</a>							
				완료일	2019-02-20	담당자	조성호	

# <인터페이스 명세서>

INFO				품목 요약 출력				
FUNC 번호		3		FUNC 명		GET_SAMPLE_SIMPLE		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
품목요약	item_simple	nvarchar	MAX	주문번호	order_code	nvarchar	20	not null
처리내용	해당 주문의 견적정보를 요약한 문자열을 반환한다. (예) 배터리 외 3종							
비고								
				완료일	2019-02-25	담당자	조성호	

# <인터페이스 명세서>

INFO		Bom 테이블에 저장할 데이터를 테이블 형태로 반환하는 함수						
FUNC 번호		4		FUNC 명		SPLIT_BOM		
송신				수신				
한글명	영문명	Type	길이	한글명	영문명	Type	길이	NULL
Bom리스트	BOMLIST	table	가변길이	품목번호	item_code	nvarchar	20	not null
				파트번호	part_code	nvarchar	MAX	not null
				파트개수	part_count	nvarchar	MAX	not null
처리내용	품목코드와 그 품목을 구성하는 파트번호, 파트개수를 입력받아서 Bom테이블에 저장할 데이터를 테이블형태로 반환합니다.							
비고	BOMLIST는 Bom테이블과 구조가 같습니다.(item_code, part_code, part_count)							
				완료일	2019-02-26	담당자	이상권	

# 시스템 이행점검표

## 서버 프로그램

	점검항목	여부
채팅서버	채팅 서버를 제대로 만들어 주는지?	
	해당 서버에 클라이언트가 메시지를 주고받는지?	
	클라이언트가 보낸 메시지를 제대로 저장하는지?	
	메시지들을 저장 후 파일로 제대로 저장하는지?	
FTP서버	FTP 서버를 제대로 만들어 주는지?	
	FTP 파일들이 제대로 업로드 다운로드 되는지?	
머신 서버	머신 서버를 제대로 만들어 주는지?	
	머신 클라이언트가 보낸 메시지를 제대로 받는지	
	머신 클라이언트의 작업 내역을 제대로 저장하는지	
	작업내역을 저장후 파일로 제대로 저장하는지?	



# 시스템 이행점검표

## 통합 클라이언트

	점검항목	여부
메시지	채팅 서버에 제대로 접속되는지?	
	FTP서버 제대로 접속되는지?	
	FTP 서버에서 파일을 제대로 주고 받는지?	
	메시지를 제대로 주고 받는지?	
ERP관리	데이터를 입력했을 때 해당 데이터 출력 폼에 제대로 입력되는지?	
	데이터가 창에 알맞게 표시 되는지?	
	출력물을 Excel로 내보냈을 때 값이 알맞게 출력되는지?	
	DB연결이 원활하게 되는지?	
	품목의 바코드 입력이 제대로 되는지	
머신 클라이언트	머신 서버에 제대로 접속되는지	
	바코드 클라이언트에 명령이 전송 되는지?	
	바코드 클라이언트의 접속이 감지 되는지?	
로그인	유효한 정보에만 로그인 되는지	
	자동 로그인을 선택했을 때 사용 가능한지?	
	개인 정보가 암호화 되어 저장되는지	
대시보드	거래 내용이 수정,추가,삭제 되었을 때 반영되는지?	

## 기계 클라이언트

	점검항목	여부
바코드	머신 서버에 제대로 접속되는지?	
	바코드를 찍고 투입자재 개수가 확인 되는지?	

# 화면 설계(지성욱)

## 머신 클라이언트

[pc1]

화면고정

종료

재시작

자재투입현황

설정

바코드 인식

Status

☐ server

☐ err\_1 ☐ err\_2

Log

설정

Name ▶

Ip ▶ 192.168.0.2

1.머신 클라이언트의 조작 메뉴.

-화면 고정: 해당 프로그램이 항상 화면에 최상위로 고정됨.

-종료: 해당 프로그램이 완전히 종료됨.

-재 시작: 해당 프로그램이 종료 후 다시 시작됨.

-자재 투입 현황:

해당 프로그램에서 인식된 자재의 이름과 개수를 보여줌.

-설정: 해당 프로그램이 동작하는 '머신'의 정보를 수정함.

-1) Name: 해당 머신 이름을 변경합니다.

-2) IP: 머신 서버에 접속 하기위한 IP를 변경함.

2.해당 프로그램과 연결된 바코드 스캐너가  
인식한 바코드를 입력할 때 사용됨.

3.해당 프로그램의 상태를 나타냄.

-server: 머신 서버와 연결이 되어있는지를 확인.

-err: 해당 에러가 발생할 경우 체크함.

해당 프로그램에서 일어나는 모든 상황에 대한  
이력이 나타남.

# 화면 설계(지성욱)

## 바코드 인쇄

품목바코드인쇄

1  2 검색

원하는 품목을 더블클릭

아이템 코드	아이템 명	규격	단위
20190128...	아이폰XS	휴대폰	스마트폰
battr_1	배터리부...	부품	부품1
battr_2	배터리부...	부품	부품2
Chicken	꼬꼬인형		
Cola	코카콜라	음료수	탄산음료
gtx960	nvidia gpu	부품	컴퓨터부품
mipad4	미패드4	태블릿	태블릿pc
samsun...	배터리	부품	배터리
samsun...	삼성케이스	케이스완...	케이스
samsun...	삼성케이...	부품	케이스
samsun...	삼성케이...	케이스부품	케이스
samsun...	갤럭시cpu	cpu	cpu
samsun...	갤럭시액정	액정	부품
Samsun...	갤럭시10	갤럭시	스마트폰

4 출력할 개수를 선택후 버튼클릭

5 84. 최대 42개 1 출력

3

- 1.검색의 키워드를 입력하기 위한 텍스트 박스.
- 2.검색 버튼 (단축키 Enter로 검색도 가능)
- 3.검색된 품목들의 내역이 표시됨.
4. 3의 품목을 더블클릭시 '아이템코드'의 바코드의 이미지가 나타냄.
5. 4의 바코드 이미지로 출력할 만큼의 개수를 선택합니다.

# 화면 설계(지성욱)

## 주문 조회

MainPage 주문 조회

거래처명으로 검색 1      검색 2      6

주문번호	거래처코드	거래처
20190303_0002	ddd2019	조성호그룹
20190303_0001	sjw9433	SJW그룹
20190301_0003	sss2018	지성욱그룹
20190301_0002	sss2018	지성욱그룹
20190301_0001	asd1234	미농
20190228_0001	ddd2019	조성호그룹
20190227_0001	asd1234	미농
20190215_0002	lsk5555	이상권그룹
20190215_0001	sjw9433	SJW그룹

주문번호	품목	품목코드	갯수	금액	단위(unit)	규격(standard)
20190228_0001	코카콜라	Cola	150	1000		
20190228_0001	nvidia gpu	gtx960	170	500000		
20190228_0001	미패드4	mipad4	150	200000		
20190228_0001	배터리	samsung_battr	200	20000		

3

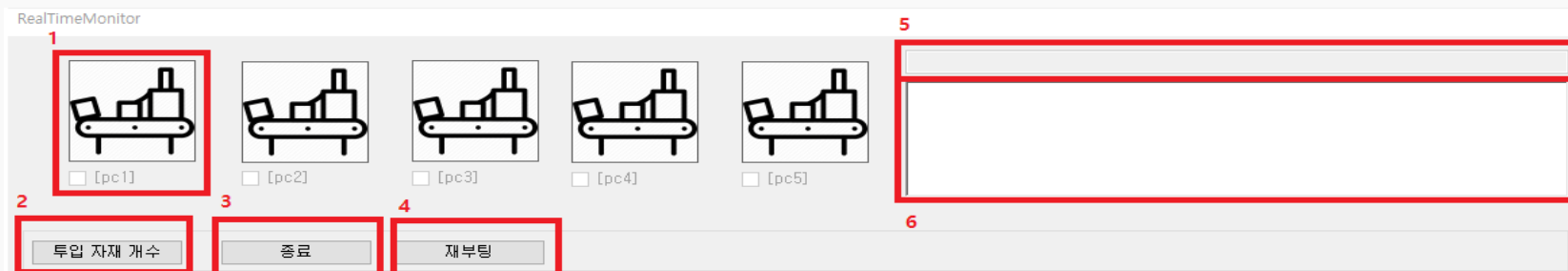
4

5      총 금액 : 721000

1. 검색 키워드를 입력하는 텍스트 박스
2. 검색 버튼 (단축키 Enter로 검색도 가능)
3. 검색된 주문의 정보가 나타납니다.
4. 3의 주문을 클릭할 경우 해당 주문의 주문내역이 나타납니다.
5. 해당 주문의 총 금액을 나타냅니다.
6. 해당 주문 조회를 종료하는 버튼 입니다.

# 화면 설계(지성욱)

## 머신 모니터링



1. 해당 머신 클라이언트가 머신 서버에 접속하게 되면 체크박스에 체크가 됩니다.
2. 특정 머신 에 투입 자재 개수를 요청하기 위한 버튼입니다.  
-접속된 머신 을 선택 후 해당 버튼을 클릭하면 자재 개수가 호출됩니다.
3. 특정 머신 클라이언트를 종료 하기 위한 버튼입니다.  
-접속된 머신 을 선택 후 해당 버튼을 클릭 하면 종료됩니다.
4. 특정 머신 클라이언트를 재 부팅 하기 위한 버튼입니다.  
접속된 머신 을 선택 후 해당 버튼을 클릭해 재 부팅 합니다.
5. 현 시점에 서버에서 보내온 메시지를 나타냅니다.
6. 서버에서 보내온 메시지의 모든 이력을 나타냅니다.

# 화면 설계(손종완)

## 견적서 조회

MainPage 견적서 조회

주문번호  1.주문코드 텍스트박스 2.주문코드 선택 버튼 3.검색 버튼 4.검색 결과 창

Excel로 내보내기 5.엑셀 출력 버튼

품목코드	품목명	수량	규격	단가	단위
------	-----	----	----	----	----

1.견적서를 찾아주는 주문번호

2.주문코드를 선택해서 갖고오는 버튼

3.결과를 찾는 버튼

4.해당 검색 결과를 보여주는 창

5.출력된 검색 결과 창의 내용을 엑셀로 출력해주는 버튼

# 화면 설계(손종완)

## 물류 조회

MainPage 견적서 조회 생산계획서 조회 입고확인서 조회 출고확인서 조회 물류 조회

기간

2019년 3월 3일 일 ~ 2019년 3월 3일 일

1. 기간을 선택할수 있는 메뉴

품목코드

2. 품목 코드 입력 텍스트 박스

3. 품목코드 선택 버튼

4. 상태 선택 라디오 버튼

상태  
☐ 대기  
☐ 출고  
☐ 취소  
☐ 완료

창고

5. 창고코드 텍스트

6. 창고코드를 선택하는 버튼

8. 도착 창고코드

7. 도착 창고코드 텍스트

9. 결과를 검색해주는 버튼

10. 검색 결과 창

검색

물류코드	출발창고	도착창고	품목명	물류갯수	물류상태
------	------	------	-----	------	------

신규

# 화면 설계(손종완)

## 생산 계획서 조회

MainPage

견적서 조회

생산계획서 조회

주문코드

1.주문코드를 입력 받는 텍스트

2.주문코드 선택

3.검색결과 출력 버튼

4.검색결과를 엑셀

5.검색 결과창

Excel로 내보내기

파일로 출력하는 버튼

Q 버튼

검색

품목코드	품목명	품목규격	수량	단가
------	-----	------	----	----



# 화면 설계(손종완)

## 입고확인서 조회

MainPage

견적서 조회

생산계획서 조회

입고확인서 조회

창고코드

1.창고코드를

입력받는 텍스트

2.창고코드

3.검색 결과 출력

버튼

4.검색 결과를 엑셀로

출력하는 버튼

Excel로 내보내기

X

이동날짜

2019년 3월 3일 일

선택 버튼

보낸창고	품목명	규격	수량
5.검색 결과를 보여주는 창			

# 화면 설계(손종완)

## 출고확인서 조회

MainPage

견적서 조회

생산계획서 조회

입고확인서 조회

출고확인서 조회

창고코드

1.창고코드 입력 텍스트

2.창고코드

3.검색 결과

4.이동날짜 선택 메뉴

5.엑셀로 검색결과를 보내주는 창

6.현재 창을 닫는 버튼

7.검색결과를 보여주는 창

트

2019년 3월 3일 일

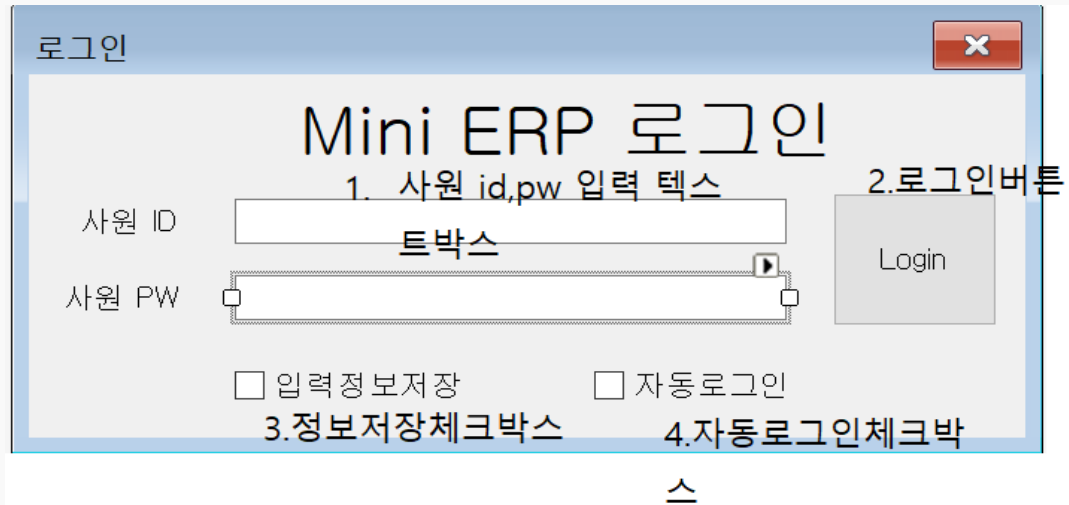
Excel로 내보내기

X

도착창고	품목명	수량	규격
------	-----	----	----

# 화면 설계(손종완)

## 로그인 화면



로그인

### Mini ERP 로그인

1. 사원 id,pw 입력 텍스트

2.로그인버튼

사원 ID

사원 PW

3.정보저장체크박스

4.자동로그인체크박스

☐ 입력정보저장 ☐ 자동로그인

스

트박스

Login

2.클릭시 텍스트 박스의 내용을 보내서 DB에서  
검사함

3.해당 내용을 종료시 파일로 저장할지 판단

4.해당 내용을 저장하고 다음에 실행시 로그인  
버튼을 클릭하게 만듦

## 메시지 화면

### O.현재 메시지 창 이름

- 폼 크기:가로 891,1234  
자동크기:false

# 화면 설계(이상권)

## 거래처조회

MainPage | 영업관리 | 물류관리 | 재고관리 | 판매/구매관리 | 출력

MainPage | 거래처 조회

1. 거래처명을 입력하세요.

거래처코드	거래처명	거래처연락처	거래처주소	거래처이메일	거래처대표
sss2018	지성옥그룹	010-7456-8847	은평구	total_rook@naver.com	지성옥
sjw9433	SJW그룹	010-8488-8799	원주시	g2023sjw@naver.com	손종완
lsk5555	이상권그룹	010-7766-8844	인천광역시	lsk7877@daum.net	이상권
ddd2019	조성호그룹	010-8877-9984	이천시	soyou@gmail.com	조성호
asd1234	미봉	010-1234-5678	금천구		미봉

2. 신규  3. 선택삭제

거래처 코드

거래처 연락처

거래처 대표

거래처 이름

거래처 이메일

거래처 등록

거래처 코드  전화번호

거래처명  이메일

대표  주소

1. 검색버튼을 누르면 검색조건을 선택할 수 있습니다.

2. 신규버튼을 누르면 등록할 수 있는 품이 띄워집니다.

3. 선택삭제버튼으로 목록에서 선택된 항목을 삭제할 수 있습니다.(하나씩 선택 가능합니다.)

4. 거래처명을 입력 후 엔터를 눌러야 검색이 실행됩니다.

# 화면 설계(이상권)

## 사원조회

The screenshot shows a web application interface for employee search. The main window has a menu bar with 'MainPage', '영업관리', '물류관리', '재고관리', '판매/구매관리', and '출력'. Below the menu, there's a 'MainPage' tab and a '사원 조회' sub-tab. The interface includes a search bar with the placeholder text '사원이름을 입력하세요.' and a '검색' button. Below the search bar is a table of employee data. At the bottom, there are two buttons: '신규' and '선택삭제'. A '사원정보 수정' dialog box is open on the right, showing fields for '사원코드', '사원명', and '직급'. A '사원등록' dialog box is open at the bottom left, showing fields for '사원 코드', '사원 이름', '직책', and '비밀번호'. Red arrows and numbers 1 through 5 indicate specific actions: 1. Enter employee name and press Enter. 2. Click the search button. 3. Double-click a row in the table. 4. Click the '신규' button. 5. Click the '선택삭제' button.

사원코드	사원명	직급
12345	홍완	대리
159	김저격	과장
5555	이상권	사장
7777	조성호	과장
8888	지성욱	PM
a001	조씨	사원
a1155	지성욱	과장
asd123	지성욱	회장
qwe123	누구지	졸병

사원정보 수정

사원코드 5555

사원명 이상권

직급 사원

수정 닫기

1. 사원의 이름을 입력하고 엔터를 눌러 검색할 수 있습니다.

2. 검색버튼을 누르면 직급으로도 검색이 가능합니다.

3. 더블클릭하면 해당 사원의 정보를 수정할 수 있습니다.

4. 신규버튼으로 사원을 등록할 수 있습니다.

5. 선택삭제 버튼을 이용해 사원을 삭제할 수 있습니다.

사원등록

사원 코드

사원 이름

직책 사원

비밀번호

저장

# 화면 설계(이상권)

## 창고조회

The screenshot shows a web application interface for warehouse management. The main window has a menu bar with 'MainPage', '영업관리', '물류관리', '재고관리', '판매/구매관리', and '출력'. Below the menu is a sub-header 'MainPage 창고 조회'. The interface includes a search bar at the top with a placeholder '창고명을 입력하세요.' and a '검색' button. Below the search bar is a table with columns '창고코드', '창고명', and '창고구분'. The table contains three rows: A001 (구디창고1, 창고), A002 (구디창고2, 창고), and W001 (구디공장1, 공장). Below the table is a search filter section with '검색 기능', '창고코드' input, a search icon, and radio buttons for '구분' (전체, 창고, 공장). At the bottom left are '신규' and '선택삭제' buttons. Two modal windows are shown: '창고 등록' (Warehouse Registration) and '창고 수정' (Warehouse Modification). The '창고 등록' window has fields for '창고코드', '창고명', and '구분' (radio buttons for 창고 and 공장), with buttons for '저장', '다시작성', and '닫기'. The '창고 수정' window has fields for '창고(공장)코드' (A002), '창고(공장)명' (구디창고2), and '구분' (radio buttons for 창고 and 공장), with buttons for '수정' and '닫기'. Red arrows and numbers 1 through 5 indicate the workflow: 1. Enter warehouse name and press Enter. 2. Press the search button. 3. Double-click a row in the table. 4. Press the '신규' button. 5. Press the '선택삭제' button.

창고코드	창고명	창고구분
A001	구디창고1	창고
A002	구디창고2	창고
W001	구디공장1	공장

창고 등록

창고코드:

창고명:

구분: ☒ 창고 ☐ 공장

저장 다시작성 닫기

창고 수정

창고(공장)코드: A002

창고(공장)명: 구디창고2

구분: ☒ 창고 ☐ 공장

수정 닫기

1. 창고명을 입력한 후 엔터를 눌러 검색할 수 있습니다.

2. 검색버튼을 누르면 다른 정보로도 검색할 수 있습니다.

3. 더블클릭하면 해당 정보를 수정할 수 있습니다.

4. 신규버튼을 눌러 창고를 등록할 수 있습니다.

5. 선택버튼으로 창고를 삭제할 수 있습니다.

# 화면 설계(이상권)

## BOM조회

1. BOM 번호 및 품목명을 입력하세요.

2. 검색

3. 품목코드

4. 조회

5. 신규

6. 선택삭제

Bom 조회

자재 코드	자재 이름	필요수량
samsung_lcd	lcd 시액판	1
samsung_cpu	갤럭시cpu	4
samsung_case_master	삼성케이스	1

Bom 상세정보

생산품목: SamsungPhone(갤럭시10)

기본생산수량: 1

생산량: [ ] 적용

품목코드	품목명	소요량
samsung_lcd	갤럭시액정	1
samsung_cpu	갤럭시cpu	4
samsung_case_master	삼성케이스	1

1. 품목의 코드 또는 이름을 입력한 후 엔터를 눌러 검색할 수 있습니다.
2. 검색버튼을 이용해서도 검색이 가능합니다.
3. 더블클릭하면 등록되어있는 BOM을 수정할 수 있습니다.
4. 조회를 누르면 등록되어있는 BOM을 확인할 수 있습니다.
5. 신규를 눌러 새 BOM을 등록할 수 있습니다.
6. 선택삭제버튼을 이용해 BOM을 삭제할 수 있습니다.



# 화면 설계(이상권)

## 사원 검색

1. **사원 이름을 입력하세요** 검색

사원코드	사원명	직급
12345	종완	대리
159	김저격	과장
5555	이상권	사장
7777	조성호	과장
8888	지성욱	PM
a001	조씨	사원
a1155	지성욱	과장
asd123	짚성욱	회장
qwe123	누구지	졸병

2. 적용 닫기

1. 사원이름을 입력하고 엔터를 누르거나 검색버튼을 클릭하여 목록을 간추릴 수 있습니다.

2. 적용버튼을 누르면 선택된 사원의 정보를 객체에 저장하고 폼이 닫힙니다.

이 폼을 호출한 폼에서 프로퍼티를 이용해 값을 사용할 수 있습니다.

# 화면 설계(이상권)

## 거래처 조회

거래처코드	거래처명
sss2018	지성옥그룹
sjw9433	SJW그룹
lsk5555	이상권그룹
ddd2019	조성호그룹
asd1234	미농

1. 거래처명을 입력하고 엔터를 누르거나 검색버튼을 클릭하여 목록을 간추릴 수 있습니다.

2. 적용버튼을 누르면 선택된 거래처의 정보를 객체에 저장하고 품이 달립니다.  
이 품을 호출한 품에서 프로퍼티를 이용해 값을 사용할 수 있습니다.

# 화면 설계(이상권)

## 창고 검색

창고 검색

1. 창고명을 입력하세요 검색

구분	창고코드	창고명
창고	A001	구디창고1
창고	A002	구디창고2
공장	W001	구디공장1

2.

적용 닫기

1. 창고명으로 검색하여 목록을 간추릴 수 있습니다.

2. 적용버튼을 누르면 선택된 창고의 정보를 객체에 저장하고 품이 닫힙니다.  
이 품을 호출한 품에서 프로퍼티를 이용해 값을 사용할 수 있습니다.

# 화면 설계(이상권)

## 품목 검색

품목 검색

1. 품목번호 또는 이름을 검색

2. 품목 검색

아이템코드	아이템명	규격	단위
20190128TT01	아이폰XS	휴대폰	스마트폰
battery_1	배터리부품1	부품	부품1
battery_2	배터리부품2	부품	부품2
Chicken	꼬꼬인형		
Cola	코카콜라	음료수	탄산음료
gtx960	nvidia gpu	부품	컴퓨터부품
mipad4	미패드4	태블릿	태블릿pc
samsung_battery	배터리	부품	배터리
samsung_case_...	삼성케이스	케이스완제품	케이스
samsung_case_p...	삼성케이스하판	부품	케이스
samsung_case_p...	삼성케이스상판	케이스부품	케이스
samsung_cpu	갤럭시cpu	cpu	cpu
samsung_lcd	갤럭시액정	액정	부품
SamsungPhone	갤럭시10	갤럭시	스마트폰

3. 적용

4. 삭제

닫기

1. 품목코드나 이름으로 검색하여 목록을 간추릴 수 있습니다.

2. 더블클릭하면 해당 품목의 정보를 볼 수 있습니다.

3. 적용버튼을 클릭하면 선택된 품목의 정보를 객체에 저장하고 품이 닫힙니다.

이 품을 호출한 품에서 프로퍼티를 이용해 값을 사용할 수 있습니다.

4. 선택한 품목을 삭제할 수 있습니다.

# 화면 설계(조성호)

## 주문 등록

주문 등록

담당자 2012218052 이상권  거래처 Goode\_B01 구디아카데미

출고창고 B01 1-B창고  거래구분 ☒ 판매 ☐ 구매

선택	품목코드	품목명	단위	규격	수량	단가	총액
<input type="checkbox"/>	Chicken	꼬꼬인형			5	15000	75000
<input type="checkbox"/>	Cola	코카콜라	탄산음료	음료수	10	2000	20000
<input type="checkbox"/>	samsung_case...	삼성케이스	케이스	케이스완제품	15	50000	750000

총액합 : 845,000

⑤ ⑥

창고 검색

창고명을 입력하세요 검색

구분	창고코드	창고명
창고	A_WhareHouses	A라인창고
창고	A00101	A창고
창고	B01	1-B창고
창고	B02	2-B창고

1. 해당 데이터를 검색하여 선택할 수 있는 창을 띄워줍니다.
2. 주문품목을 보여줍니다.
3. 품목을 검색하여 선택할 수 있는 창을 띄워줍니다.
4. ②의 선택부분에 체크 되어있는 품목 삭제합니다.
5. 주문서를 저장합니다.
6. 주문서를 다시 작성합니다.

# 화면 설계(조성호)

## 거래조회

주문 등록

담당자

2012218052

이상권

거래처

Goode\_B01

구디아카데미

출고창고

B01

1-B창고

거래구분

☒ 판매 ☐ 구매

②

선택	품목코드	품목명	단위	규격	수량	단가	총액
<input type="checkbox"/>	Chicken	꼬꼬인형			5	15000	75000
<input type="checkbox"/>	Cola	코카콜라	탄산음료	음료수	10	2000	20000
<input type="checkbox"/>	samsung_case...	삼성케이스	케이스	케이스완제품	15	50000	750000

⑤

⑥

저장

다시작성

총액합 : 845,000

③

④

품목추가

품목삭제

창고 검색

창고명을 입력하세요

검색

구분	창고코드	창고명
창고	A_WhareHouses	A라인창고
창고	A00101	A창고
창고	B01	1-B창고
창고	B02	2-B창고

적용

닫기

1. 해당 데이터를 검색하여 선택할 수 있는 창을 띄워줍니다.
2. 주문품목을 보여줍니다.
3. 품목을 검색하여 선택할 수 있는 창을 띄워줍니다.
4. ②의 선택부분에 체크되어 있는 품목 삭제합니다.
5. 주문서를 저장합니다.
6. 주문서를 다시 작성합니다.

# 화면 설계(조성호)

## 거래수정

거래수정

담당자 2012218080 지성욱 ①

창고 ② A\_WhareHoues A라인창고 ④

거래상태 ③ 승인

주문 품목

품목코드	갯수	가격
Chicken_Set 치킨세트	30	50000

창고 검색 창고명을 입력하세요 검색

구분	창고코드	창고명
창고	A_WhareHoues	A라인창고
창고	A00101	A창고
창고	B01	1-B창고
창고	B02	2-B창고

< >

1. 해당 정보를 검색하여 선택할 수 있는 창을 띄워줍니다
2. 현 거래상태에 따라 선택할 수 있는 범위의 거래상태를 보여주고 선택할 수 있습니다.
3. 해당 거래의 견적내용을 보여줍니다.
4. 변경된 정보로 해당 거래를 수정합니다.

# 화면 설계(조성호)

## 물류등록

창고 이동

현재 창고 A라인창고

② 현재 창고 품목

품목번호	품목이름	갯수
battry_1	배터리부품1	13
mipad4	미패드4	30

④ >>

⑤ <<

⑥ 개수: 5

확인 취소

① 목표 창고

③ 이동 품목

품목번호	품목이름	갯수
mipad4	미패드4	20

⑦ 비우기

이동날짜: 2019년 3월 4일 월요일

이동 ⑧

1. 이동가능한 창고를 선택할 수 있습니다.
2. 현재창고의 재고내역을 보여줍니다.
3. 이동할 품목내역을 보여줍니다.
4. 선택한 품목을 이동품목으로 옮깁니다.
5. 선택한 이동품목을 취소합니다.
6. ④에서 이동할 품목의 개수를 설정합니다.
7. 이동품목목록을 비웁니다.
8. 입력된 정보를 토대로 물류를 등록합니다.



# 화면 설계(조성호)

## 재고추가

재고 추가

저장창고 A00101 A창고

품목

개수 1

추가

창고 검색

창고명을 입력하세요 검색

구분	창고코드	창고명
창고	A_WhareHouses	A라인창고
창고	A00101	A창고
창고	B01	1-B창고
창고	B02	2-B창고

적용 닫기

1. 해당 정보를 검색하여 선택할 수 있는 창을 띄워줍니다.
  2. 추가할 재고의 개수를 입력합니다.
  3. 입력된 정보를 토대로 재고를 추가합니다.
- 재고에 데이터가 추가/수정 되면  
품목테이블에서 해당 품목코드의 재고 개수 변동
  - 변동 이후 재고가 0개이면 해당 재고데이터 삭제

## 단위테스트-1

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

## 단위테스트-2

GET_PROG_T RADE_01	거래 구분에 따라 진행중인 거래에 대한 데이터가 제대로 조회되는지 확인	거래 구분에 따라 진행중인 조회결과 출력	입력된 거래상태에 맞게 진행중인 거래들이 제대로 조회되어 출력됨.	PASS
MODIFY_TR ADE_01	잘못된 데이터(등록되지않은 주문번호)를 입력했을 때 예외처리가 제대로 되는지 확인	등록되지않은 주문번호라고 에러메시지 출력 후 트랜잭션 ROLLBACK	오류메시지 출력된 후 트랜잭션 ROLLBACK 수행	PASS
SET_DISTRIB UTION_01	입력데이터의 구분자를 " "가 아닌 다른 구분자를 사용할 경우 오류메시지 출력하는지 확인	구분자를  로 입력해달라는 오류메시지 출력 후 트랜잭션 ROLLBACK	오류메시지가 출력되지 않고 트랜잭션 ROLLBACK이 되지않음	FAIL
MODIFY_DI STRIBUTION	물류데이터가 입력데이터대로 제대로 수정되는지 확인	물류번호를 통해 데이터 수정	해당 물류번호의 데이터가 입력데이터에 맞게 수정됨.	PASS
SET_ORDER_ 01	입력데이터(주문데이터)가 업무프로세스에 따라 제대로 입력되는지 확인	업무프로세스에 따라 데이터가 DB에 저장	업무프로세스의 순서대로 입력데이터가 주문,견적,거래 테이블에 알맞게 입력됨	PASS
SET_ORDER_ 02	입력데이터가 견적테이블에 제대로 저장되는지 확인	견적테이블에 데이터가 저장	입력데이터가 제대로 견적테이블에 저장됨.	PASS
SPLIT_SAMP LEORDER_0 1	구분자 " "를 통해 제대로 가상테이블이 만들어져 반환되는지 확인		order_code item_code count	PASS
		order_code: "20121212_0001"	20121212_0001 a01 5	
		item_code: "a01 a02 a03"	20121212_0001 a02 6	
		item_count: "5 6 7"	20121212_0001 a03 7	
			데이터를 가지고있는 가상테이블 반환	
SPLIT_DATE CODE_01	날짜_시퀀스 타입의 데이터를 입력하였을 때 제대로 DATE타입의 변수를 반환하는지 확인	"20121212_0001"을 매개변수로 넘겨주면 "2012-12-12"의 DATE타입의 변수 반환	"2012-12-12"의 DATE타입의 변수가 반환됨	PASS
SET_ORDER_ 03	입력 데이터가 제대로 거래테이블에 입력되는지 확인	입력데이터가 거래테이블에 등록	입력데이터가 제대로 거래테이블에 입력됨.	PASS
MODIFY_OR DERLIST_01	주문번호를 통해 견적데이터가 제대로 수정되는지 확인	기존 견적 데이터를 삭제후 입력데이터를 등록	기존 견적데이터가 삭제되고 입력데이터가 추가됨.	PASS
SET_STOCK_ ITEM_01	존재하지않은 품목번호를 입력하거나 창고번호를 입력했을 때 에러메시지를 출력하는지 확인	에러메시지 출력	에러메시지 출력됨	PASS
RESET_SEQU ENCE_01	DB날짜테이블의 최상단레코드와 현재 시스템날짜를 비교하여 다르다면 DB내에 있는 모든 시퀀스가 제대로 초기화되고 DB날짜테이블이 갱신되는지 확인	DB날짜"2019-03-03"	모든 시퀀스가 초기화되고 DB날짜를 "2019-03-04"로 갱신	PASS
		시스템날짜"2019-03-04" 모든 시퀀스가 초기화되고 DB날짜를 "2019-03-04"로 갱신		
GET_ITEM_0 1	검색어를 통해 품목데이터가 제대로 조회되는지 확인	검색어에 맞는 품목데이터 조회	검색어에 맞는 품목데이터가 조회됨	PASS
GET_STOCK_ 01	검색데이터가 입력되지않을경우 모든 데이터가 조회되는지 확인	재고테이블의 모든 데이터 조회	제고테이블의 모든 데이터가 조회됨.	PASS
PROC_STAT EMENT_01	매개변수로 입력된 주문번호의 전표의 견적데이터 제대로 출력되는지 확인	해당 주문번호의 견적데이터가 출력된다.	해당 주문번호의 전표에 필요한 견적데이터가 출력	PASS

## 주문등록

			작성자	작성일			작성자	수행일			
			조성호	2019. 2. 15			조성호	19.2.15~19.3.4			
테스트 대상 상세 :											
주문등록과 관련된 처리들											
테스트 시나리오 목적 :											
주문등록와 관련된 처리들이 제대로 동작 되는지 확인											
가정 및 필요사항 :											
프로그램을 통해 주문등록 업무프로세스에 맞게 수행											
STEP (ID)	테스트 항목		기대결과			실제결과			비고		
ORDER_01	주문등록시 사원,거래처,입/출고창고, 품목을 관련 선택품을 통해 선택하여 입력		각각의 텍스트박스에 선택한 데이터가 입력된다.			선택한 데이터가 제대로 입력 되었다.			PASS		
ORDER_02	입력된 품목데이터의 개수를 수정할 때 숫자가 아닌 문자를 입력하여 예외 처리가 제대로 동작되는지 확인		숫자만 입력할수있다는 에러메시지를 출력			문자도 입력되었다.			FAIL		
ORDER_03	체크박스에 선택된 품목들이 품목삭제 버튼을 클릭하였을 때 제대로 삭제되는지 확인		체크박스에 선택된 품목이 데이터그리드 뷰에서 삭제된다.			체크된것만 제대로 삭제되었다.			PASS		
ORDER_04	필요한 데이터가 전부 입력된 후 주문 등록을 수행할 때 유효성검사를 통해 오류검출		잘못된 데이터에 대한 오류메시지를 보여 주고 주문등록 작업이 취소된다.			제대로 유효성검사에서 오류메시지가 출력되었고 수행중이던 작업도 취소가 되었다.			PASS		

물류등록/수정

작성자

조성호

작성일

2019. 2. 15

작성자

조성호

수행일

19.2.15~19.3.4

테스트 대상 상세 :

물류등록/수정과 관련된 처리들

테스트 시나리오 목적 :

물류등록 업무프로세스와 관련된 처리들이 제대로 동작 되는지 확인

가정 및 필요사항 :

프로그램을 통해 업무프로세스에 맞게 수행

STEP (ID)	테스트 항목	기대결과	실제결과	비고
DISTRIBUTION_01	물류등록시 현재창고 콤보박스에 DB에 등록된 전체 창고목록이 출력되는지 확인	콤보박스에 DB에 등록된 창고목록이 전부 출력	DB에 등록된 창고목록이 전부 출력되었다.	PASS
DISTRIBUTION_02	현재창고를 선택하지 않았을 때 목표창고목록이 표시 되는지 확인	목표창고의 창고목록이 표시되지않는다.	목표창고의 창고목록이 표시되지않았다.	PASS
DISTRIBUTION_03	현재창고가 선택되었을 때 목표창고에는 현재창고를 제외한 창고목록이 표시 되는지 확인	현재창고를 제외한 창고목록이 출력	현재창고로 포함된 창고목록이 출력되었다.	FAIL
DISTRIBUTION_04	현재창고를 선택하였을 때 현재창고품목 ListView에 있는 모든 재고목록을 보여주는지 확인	현재창고의 모든 재고목록을 보여준다.	해당 창고에 있는 재고목록을 보여주었다.	PASS
DISTRIBUTION_05	현재창고품목에서 선택한 품목이 이동품목으로 옮겨졌을 때 현재창고품목의 개수가 이동품목만큼 줄어드는지 확인	이동된 품목의 개수가 이동된 만큼 줄어든다. 또한 남은갯수가 0개라면 목록에서 삭제된다.	제대로 개수가 줄어들었고, 남은갯수가 0개일경우 목록에서 삭제되었다.	PASS
DISTRIBUTION_06	이동갯수를 입력할 때 이동가능한 최소/최대치가 제대로 적용 되고있는지 확인.	최소치 1, 최대치 재고갯수로 설정되어있는 걸 확인.	제대로 적용되어있는걸 확인	PASS
DISTRIBUTION_07	이동품목의 리스트를 지울 때 해당 데이터가 현재창고품목에 추가되거나 더해지는지 확인	지워진 개수만큼 현재창고품목에 추가되거나 더해진다.	동일객체참조오류 발생	FAIL
DISTRIBUTION_08	이동날짜를 현재날짜의 이전은 선택안 되도록 하였는지 확인	현재날짜와 현재날짜 이후만 선택가능하도록 처리되어있다.	현재날짜와 현재날짜 이후만 선택가능하도록 처리되어있다.	PASS
DISTRIBUTION_09	이동버튼 클릭시 현재창고, 목표창고, 이동품목, 이동날짜에 대한 유효성검사가 제대로 처리되는지 확인	유효성검사를 통과하지못할경우 예외처리 출력	예외처리가 출력되지않는다.	FAIL
DISTRIBUTION_10	모든데이터가 입력되었을 때 물류테이블에 제대로 입력되는지 확인	입력데이터에 맞게 물류테이블에 데이터가 입력된다.	문제없이 입력데이터가 전부 물류테이블에 입력되었다.	PASS

# 통합 테스트(지성욱)

테스트 대상 상세 :

대쉬보드

테스트 시나리오 목적 :

대쉬보드가 당초의 목적에 부합하게 동작하는지

가정 및 필요사항 :

-

STEP (ID)	테스트 항목	기대결과	실제결과	비고
ID01	처음 프로그램 구동시 대쉬보드의 거래 내역이 정상적으로 보이는가	거래테이블에 들어있는 결과가 보일것이라고 예상함	의도한 결과와 동일하게, 거래 테이블의 결과 값을 출력함	PASS
ID02	만들어진 거래내역의 상세정보를 조회할 수 있는가	거래 컨트롤의 버튼클릭시 해당 거래의 상세정보를 볼 수 있을 것 이라고 생각함	의도한 결과와 일치함, 상세정보를 확인할 수 있음	PASS
ID03	상세정보에서 진행단계를 변경 할 수 있는가	상세정보의 진행단계 클릭시 진행단계 변경이 가능할 것이라고 생각함	의도한 결과와 일치함, 실제로 승인, 취소 등이 가능함	PASS
ID04	상세정보에서 담당자를 변경 할 수 있는가	상세정보의 진행단계 클릭시 담당자 변경이 가능할 것이라고 생각함	의도한 결과와 일치함, 실제로 담당자 변경이 가능함	PASS
ID05	상세정보에서 출고창고를 변경 할 수 있는가	상세정보의 진행단계 클릭시 출고창고 변경이 가능할 것이라고 생각함	의도한 결과와 일치함, 실제로 출고 창고의 변경이 가능함	PASS
ID06	대쉬보드에서 실제로 거래에대한 처리가 가능한가.	해당 거래를 클릭하고 상세정보의 진행상태를 변경해 가며 거래의 모든 하나의 처리가 가능할 것이라고 생각함	의도한 결과와 일치함, 등록된 거래에 대해 모든 처리가 가능했으며, 하나의 거래의 완료가 가능했음	PASS
ID07	대쉬보드에서 거래의 취소가 가능한가.	거래의 취소가 가능 할 것이라고 생각함	상세 정보에서 거래상태를 취소로 변경할 경우 거래의 취소가 가능했음.	PASS
ID07	대쉬보드에서 거래상태 변경시 UI로 확인이 가능한가	UI단 변경이 가능할 것이라고 생각함	의도한 결과와 일치함, UI의 거래 상태가 변경됨	PASS

# 통합 테스트(지성욱)

테스트 대상 상세 :				
Machine_Server 과의 Machine_Client, Machine_Monitoring 과의 상호작용 테스트				
테스트 시나리오 목적 :				
해당 서버를 경유해 클라이언트 들 간 명령어를 전하고 그에대한 반환값을 올바른 결과로 전달하는지에 대한 확인.				
가정 및 필요사항 :				
해당서버에 각각의 클라이언트가 접속을 해야 함, 또한 Machine_Client의 Name을 이용해 명령어를 전달하기에 동일한 이름을 가진 클라이언트가 있다면 안됨.				

STEP (ID)	테스트 항목	기대결과	실제결과	비고
ID01	서버에 클라이언트가 접속했을때 반응	서버에 해당 클라이언트가 접속되었음을 알려주며, 모니터링 클라이언트에 머신이 접속됨을 알려줌.	서버에 머신 클라이언트가 접속되었음을 알리고, 기대결과에 부합한 결과를 보여줌.	PASS
ID02	머신클라이언트가 접속했음을 모니터링 클라이언트가 감지	머신 클라이언트가 최초 접속할 경우 서버에 접속 메시지를 날려 모니터링 클라이언트가 이를 감지해 접속 확인을 체크함.	머신 클라이언트가 서버에 접속 메시지를 보낸것을 감지해 원하는 결과를 보여줌.	PASS
ID03	서버가 닫혀있을때 클라이언트 들이 접속할 경우	TCP Client의 connected 프로퍼티가 false일 경우 retrun 하도록 해 의도대로 접속종료 될 것이라고 생각함.	TCP Client의 객체 초기화 부분전에 connected를 참조해 null참조 오류 출력됨.	FAIL
ID04	모니터링 클라이언트가 접속종료 명령어를 클라이언트에게 보낼 경우	서버에 접속이 되었음으로 모니터링 클라이언트가 명령어를 보내면 머신 클라이언트가 접속종료 할 것이라 생각함.	머신클라이언트의 이름을 가진 명령어를 확인하고 원하는 결과를 나타냄	PASS
ID05	모니터링 클라이언트가 재접속 명령어를 클라이언트에게 보낼 경우	머신클라이언트가 재접속 되면서 서버와 다시 연결될때 연결 메시지를 보낼 것이라고 생각함.	기대결과와 같은 결과를 나타냄	PASS
ID06	모니터링 클라이언트가 투입자재 확인에 대한 명령어를 클라이언트에게 보낼 경우	머신클라이언트가 서버에서 받은 투입자재에대한 명령어를 획득하고 현재까지 머신클라이언트에서 저장된 바코드의 정보와 개수를 반환받을 것이라고 생각함	메시지를 획득 하고, 바코드의 정보와 개수를 가진 메시지를 모니터링클라이언트에서 확인할 수 있다.	PASS



# 통합 테스트(지성욱)

테스트 대상 상세 :				
품목 바코드 인쇄품 동작				
테스트 시나리오 목적 :				
품목 바코드 인쇄 품이 올바르게 동작하는지				
가정 및 필요사항 :				
-				

STEP (ID)	테스트 항목	기대결과	실제결과	비고
ID01	품목의 이름을 빈값으로 두고 검색	전체결과가 나올것이라 생각함	칸이 비어있을경우, 모든 아이템에 대해 결과가 나옴 의도한 결과와 일치함	PASS
ID02	품목의 이름으로 검색	입력한 검색키워드가 포함된 모든 품목이 검색될 것이라고 생 각함	검색키워드를 가지고있는 모든 품목이 검색됨, 의도한 결과와 일치함	PASS
ID03	검색된 품목을 더블클릭해 바코드를 생성함	클릭한 품목에 맞는 바코드가 생성될 것이라고 생각함	의도한 결과대로 해당 품목을 가진 바코드가 출력됨	PASS
ID04	A4용지에 인쇄할수 있는 바코드의 개수보다 더 크게 입력함	A4 한장에 42개 인쇄가 가능하지만 그 이상으로 출력할 경우 입력되지않을것이라 생각함.	의도대로 42개 이상입력시 불가능하다고 메시지를 출력함	PASS
ID05	인쇄할 바코드의 개수를 0 으로 입력	0개는 출력되지않을것이라고 생각함.	0개에 대한 예외처리를 해 두지않아 아무것도 인쇄되지않은 이미지파일이 출력됨	FAIL
ID06	인쇄한 바코드가 실제로 스캐너로 인식이 되지는지	결국똑같은 바코드이기에 스캐너로 스캔한다면, 품목의 코드가 나올것으로 생각함.	의도한 결과와 일치함. 품목의 코드를 확인 할 수 있음	PASS
ID07	만들어진 이미지가 실제로 저장이 되는지	saveFileDailog에서 지정한 경로에 만들어진 바코드이미지가 저장되어있을것이라고 생각함	의도한 결과대로 해당위치에 선택한 바코드가 입력한 개수만큼 저장된 이미지파일이 생성됨	PASS



# 통합 테스트(지성욱)

테스트 대상 상세 :	
주문조회 품 동작	
테스트 시나리오 목적 :	
주문조회 품이 원하는 의도대로 동작하는지	
가정 및 필요사항 :	
-	

STEP (ID)	테스트 항목	기대결과	실제결과	비고
ID01	거래처의 이름을 빈값으로 두고 검색	전체결과가 나올것이라 생각함	칸이 비어있을경우, 모든 주문에 대해 결과가 나오 의도한 결과와 일치함	PASS
ID02	거래처의 이름으로 검색 그룹을 키워드로 잡고 검색	입력한 검색키워드가 포함된 모든 거래처가 검색될 것이라고 생각함 결과로 지성욱 그룹, 조성호 그룹이 나올것이라고 생각함.	검색키워드를 가지고있는 모든 품목이 검색됨, 의도한 결과와 일치함 지성욱 그룹, 조성호 그룹의 주문내역이 출력됨	PASS
ID03	검색된 주문조회를 클릭해 상세 내역을 호출함	해당 주문에 대한 모든 품목의 내역이 나올것이라고 생각함	의도한 결과대로 이루어짐, 품목과 개수, 금액등이 호출됨	PASS
ID04	상세정보를 호출할때 모든 품목에 대한 총액이 계산이 되는지	아이폰, 배터리부품1, 배터리부품2의 모든 개수에대한 금액을 계산해 알려줄것이라고 생각함	기대결과와 일치함, 총액이 원하는 결과대로 출력됨	PASS

# 통합 테스트(지성욱)

테스트 대상 상세 :				
Machine_Client의 올바른 동작을 확인				
테스트 시나리오 목적 :				
머신클라이언트가 사용자가 원하는대로 동작하는지를 테스트함.				
가정 및 필요사항 :				
머신 서버에 접속을 했다고 가정함				

STEP (ID)	테스트 항목	기대결과	실제결과	비고
ID01	서버에 클라이언트가 접속했을때 반응	서버에 해당 머신이 접속했다고 알림을 보낼것이라 생각함.	서버에 접속했다고 메시지를 보내며, 의도한 결과와 일치함	PASS
ID02	서버가 닫혀있을경우 클라이언트가 접속했을경우.	서버에 접속할때 일정시간(1초) 이상이 지나도록 연결이되지 않으면 일정시간 안에 접속이되지않아 지정해놓은 예외가 접속하지못했다고 가정해 에러를 출력하지 않을것이라 생각함.	호출되어 의도한 결과와 일치함	PASS
ID03	Appconfig에 저장되어있는 IP를 변경합니다.	메뉴스트립의 '설정'메뉴에서 Ip를 변경한다면 바뀌게 됨.	기대결과와 같은 결과를 나타냄.	PASS
ID04	Appconfig에 저장되어있는 Name을 변경합니다.	메뉴스트립의 '설정'메뉴에서 Name를 변경한다면 바뀌게 됨.	기대결과와 같은 결과를 나타냄.	PASS
ID05	Appconfig에 저장되어있는 IP를 변경합니다. 올바른 아이피가 아닌 값을 입력할 경우	Ipaddr 클래스의 parser 메소드를 이용해 올바른 아이피라면 등록되도록 예외처리하여, 올바르지않다면 등록되지 않을것.	쓰레기 결과를 입력했을경우 입력값이 IP형식을 취하지 못한다면 등록되지않음. 의도한 결과와 일치함.	PASS
ID06	화면고정옵션이 동작하는지	메뉴의 '화면고정' 클릭시 머신클라이언트가 최상단에 고정됨.	의도한 결과와 일치함.	PASS
ID07	자재투입현황 기능이 동작하는지	자재투입현황 메뉴를 호출할 경우, 현재 이 머신을 지나간 모든 자재에 대한 정보와 개수가 서버로 보낼것이라고 생각함	모든 정보가 담긴 바코드리스트의 결과와 만들어진 메시지의 결과가 동일함	PASS
ID08	자재에대한 바코드를 잘 읽어 저장하는지	바코드스캐너가 지나간 자재의 바코드를 읽고 컬렉션에 저장할 것 이라 생각함.	스캐너가 동작할때마다 읽은 바코드의 정보가 찍혀 리스트에 추가되고, 동일한 코드일경우 개수가 증가함.	PASS
ID09	"종료" 메뉴가 동작하는지.	종료할 경우 서버와의 통신을 올바르게 끊기위해 서버종료 메소드를 호출 후 프로그램종료	의도한대로 서버종료 메소드를 호출 후 서버에는 올바르게 접속종료되었다고 처리됨.	PASS
ID10	"재접속" 메뉴가 동작하는지.	재접속할 경우 서버와의 통신을 올바르게 종료한 후 머신클라이언트를 다시 실행할것이라 생각함	의도한대로 올바르게 서버와 통신을 종료 후, 재시작되며 서버와 다시 통신을 성공함.	PASS

# 통합 테스트(손종완)

테스트 대상 상세 :				
서버 프로그램중에서 chatserver을 작동시켰을때 클라이언트가 여러 작동을 한후의 서버가 하는 반응				
테스트 시나리오 목적 :				
채팅 프로그램을 사용하기 위해서 해당 프로그램이 하드웨어 성능이 낮은 컴퓨터에서도 돌아가는지 확인하며 해당 프로그램이 정상적으로 작동하는지 확인 채팅 서버 프로그램이 설계때 의도대로 클라이언트에 반응해주는지 확인				
가정 및 필요사항 :				
해당 채팅서버가 사용하는 ip와 포트를 다른 프로그램이 사용하고 있지 않아야함 해당 서버에 접속할때 닉네임이 서로 중복되지 않아야한다.				

STEP (ID)	테스트 항목	기대결과	실제결과	비고
ID01	채팅 클라이언트가 접속시 제대로 로그가 뜨는지	접속자 로그가 뜨면서 해당 접속자가 사용자에게 추가	접속 로그 및 접속자가 사용자에게 추가됨	PASS
ID02	제대로 채팅 클라이언트의 종료를 끊는지	해당 클라이언트와 프로그램과의 연결이 끊어졌습니다 출력후 해당 접속을 종료	해당 접속이 끊어진것을 인지 하고 접속이 끊김을 로그에 추가	PASS
ID03	채팅클라이언트에서 방을 만드는 메시지 보냈을때	해당 방을 만들고 원하는 접속자들을 추가 하고 방에 메시지 저장	원하는 대로 서버에 방을 만들고 해당 방에 메시지 내용들을 저장	PASS
ID04	채팅 클라이언트에서 전체 메시지를 보낼때	해당 내용이 서버에 접속한 인원들에게 전송	전체 사용자에게 메시지가 전달	PASS
ID05	채팅 클라이언트에서 특정 방에 메시지 보낼때	해당 내용이 방 접속인원으로 등록된 인원들에게만 전송	해당 방의 인원들에게만 메시지 내용이 전달	PASS
ID06	채팅 클라이언트에서 이미 있는 방을 만들려 할때	해당 방 이름은 있음을 전달	해당 방은 이미 있다는 메시지 전달	PASS
ID07	채팅 클라이언트가 채팅서버에 접속했을때	해당 클라이언트에게 채팅에 접속한 인원 현재 개설된 방 내용을 전 달	방 정보와 접속 인원을 전달	PASS
ID08	채팅 클라이언트에서 방을 삭제	해당 방이 채팅서버에서 삭제되고 모든 클라이언트에게 방이 삭제되었다는 메시지를 전달	방이 삭제 되어 해당 방을 삭제한다는 메시지를 전달	PASS
ID09	채팅 클라이언트에서 방에 참가한다는 메시지를 보낼때	해당 방에 해당 접속자가 추가됨	해당 방의 접속자에 클라이언트의 닉네임이 추가됨	PASS
ID10	채팅 클라이언트에서 이미 들어와있는 닉네임으로 접근	이미 있는 닉네임이라고 출력되어 해당 접속을 차단	해당 접속을 비허용하고 차단됨	PASS
ID11	채팅 클라이언트가 접속종료 없이 연결이 끊김을 로그에 추가하는지	해당 클라이언트와 프로그램과의 연결이 끊어졌습니다 출력후 해당 접속을 종료	해당 접속이 끊어졌을때 채팅 접속이 예외처리 되 어 해당 처리가 끝남	PASS

# 통합 테스트(손종완)

테스트 대상 상세 :				
채팅서버및 FTP서버에 접속한 클라이언트				
테스트 시나리오 목적 :				
채팅 프로그램을 사용하기 위해서 해당 프로그램이 하드웨어 성능이 낮은 컴퓨터에서도 돌아가는지 확인하며 해당 프로그램이 정상적으로 작동하는지 확인 채팅 클라이언트 프로그램이 설계때 의도대로 서버와 통신하면서 제대로 작동하는지				
가정 및 필요사항 :				
접속시 로그인 창 의 ID들은 이미 접속하지 않는 아이디로 접속하여야한다				

STEP (ID)	테스트 항목	기대결과	실제결과	비고
ID01	채팅 클라이언트가 서버 접속시 방목록과 전체 접속 인원을 받아오는지	화면에 채팅 접속인원과 방 목록이 출력됨	채팅 접속인원과 방 목록이 보여짐	PASS
ID02	메시지를 전송시 제대로 서버에 전송되고 화면에 출력되는지	화면에 자신이 보낸 메시지가 출력됨	화면에 자신이 보낸 메시지 내용이 추가됨	PASS
ID03	방을 만들었을때 제대로 서버에도 해당 이름의 방이 만들어지는지	방목록에 자신이 만든 방의 이름이 추가됨	방목록에 자신이 만든 방이 추가됨	PASS
ID04	서버에서 메시지가 도착했을때 알림 메시지가 제대로 뜨는지	PC 하단 오른쪽에 채팅 알림이 나옴	PC 하단쪽에 메시지 알림이 나타남	PASS
ID05	자신에게 온 알림 메시지를 클릭했을때 해당 방의 채팅창으로 이동하는지	알림이 온 메시지의 방으로 채팅방이 이동	해당 채팅방으로 이동	PASS
ID06	FTP서버에 200MB이상 파일을 올리려고 했을때 올려지는지	해당 용량이 크므로 더 작은 파일을 선택하라는 메시지박스 출력	경고 문구가 나오면서 해당 파일은 업로드 안됨	PASS
ID07	FTP서버에서 파일을 다운로드 제대로 받을수있는지	자신이 원하는 위치를 지정하여 저장할수 있는 창이 열림	다운로드 창이 열림	PASS
ID08	해당 프로그램을 종료시 완전히 종료가 되는지	해당 프로그램이 종료시 CMD로 해당 프로그램과 같은 이름의 프로세서들을 모두 강제로 종료시키는 명령출력	해당 클라이언트와 같은 모든 이름의 프로그램은 종료됨	PASS
ID09	FTP서버에서 파일을 다운받을때 해당 프로그램이 멈추지 않는지	해당 프로그램이 멈추지 않고 제대로 작동	프로그램이 멈추지 않고 제대로 작동됨	PASS
ID10	해당 프로그램을 최소화했을때 다른 폼 또한 같이 최소화되는지	메시지 폼과 함께 다른 폼들도 최소화시킴	띄워져있는 폼 창들이 같이 움직임	PASS

# 통합 테스트(손종완)

테스트 대상 상세 :

서버 프로그램중에서 ftpserver을 작동시켰을때 클라이언트가 여러 작동을 한후의 서버가 하는 반응

테스트 시나리오 목적 :

채팅 프로그램을 사용하기 위해서 해당 프로그램이 하드웨어 성능이 낮은 컴퓨터에서도 돌아가는지 확인하며 해당 프로그램이 정상적으로 작동하는지 확인  
FTP서버를 작동시켰을때 해당 파일들을 업로드 시키고 다운로드 시켜줄수 있는지

가정 및 필요사항 :

해당 FTP서버가 사용하는 ip와 포트를 다른 프로그램이 사용하고 있지 않아야함

STEP (ID)	테스트 항목	기대결과	실제결과	비고
ID01	FTP클라이언트가 파일을 업로드 시킴	해당 클라이언트가 선택한 방의 이름과 같은 폴더에 파일을 업로드함 해당 폴더가 없을때는 폴더 생성후 업로드	해당 폴더를 생성후 업로드 성공	PASS
ID02	FTP클라이언트가 파일을 선택하고 다운로드해도 프로그램이 다운안되고 작동하는지	FTP 클라이언트가 선택한 파일을 제대로 클라이언트의 로컬 컴퓨터에 저장	해당 파일을 다운받아 저장 성공	PASS
ID03	FTP클라이언트가 접속종료했을때 제대로 접속 종료 를 감지하는지	FTP 클라이언트가 접속종료 했음을 텍스트 박스에 로그로 출력	접속을 종료했음을 텍스트박스에 출력	PASS
ID04	FTP클라이언트가 강제 종료되었을때도 해당 클라이언트의 접속을 종료하는지	FTP 클라이언트에서 접속이 끊겼음을 출력	접속이 끊겼음을 텍스트박스에 출력	PASS

# 통합 테스트(손종완)

테스트 대상 상세 :	
서버 프로그램중에서 machineserver을 작동시켰을때 클라이언트가 여러 작동을 한후의 서버가 하는 반응	
테스트 시나리오 목적 :	
채팅 프로그램을 사용하기 위해서 해당 프로그램이 하드웨어 성능이 낮은 컴퓨터에서도 돌아가는지 확인하며 해당 프로그램이 정상적으로 작동하는지 확인 machine서버를 작동시켰을때 해당 클라이언트에 메시지를 전달하는지	
가정 및 필요사항 :	
해당 머신서버가 사용하는 ip와 포트를 다른 프로그램이 사용하고 있지 않아야함	

STEP (ID)	테스트 항목	기대결과	실제결과	비고
ID01	머신 클라이언트가 접속 후 제대로 해당 클라이언트 의 메시지를 모든 접속 클라이언트에 전달	서버가 받은 메시지들이 모두 클라이언트에 전달됨	메시지들이 모두 접속한 클라이언트에 전달	PASS
ID02	머신 클라이언트가 접속 종료 메시지를 전달했을때 제대로 접속 클라이언트에서 제거되는지	해당 클라이언트가 접속한 클라이언트 중에서 제거됨	해당 클라이언트가 종료 했음을 모든 클라이언 트에 메시지로 전달	PASS
ID03	머신 클라이언트가 접속 종료한다는 메시지를 전달 않고 강제 종료 되었을때	해당 클라이언트와 접속이 끊겼음을 로그로 텍스트박스에 출력	접속이 끊겼음을 텍스트박스에 출력	PASS

# 단위 테스트(이상권)

## 테스트 대상 상세 :

DataBase 의 Item 테이블에서 Item\_image 컬럼의 데이터를 읽어서 Image 객체를 반환할 수 있는지 테스트합니다.

## 테스트 시나리오 목적 :

사용자가 선택한 아이템의 정보를 Frm\_ItemInfo 품의 PictureBox 의 Image 속성값으로 넘겨주기 위함입니다.

## 가정 및 필요사항 :

DataBase에는 바이트배열 형태로 저장되어 있습니다.  
바이트배열 타입(읽은 값)을 이미지 타입으로 변경해야 합니다.

STEP (ID)	테스트 항목	기대결과	실제결과	비고
ID01	SqlDataReader 객체로 DB 의 Item 테이블에서 Item_image 필드를 읽습니다.	바이트 배열을 반환합니다.	Item_image 가 null 인 경우 읽은 값을 바이트배열로 받을 수 없었습니다.	FAIL
ID02	Item_image 필드의 값이 null 인 경우에도 오류를 출력하지 않는지 테스트합니다.	에러가 나지 않습니다.	에러가 나지 않습니다.	PASS
ID03	얻어온 값을 받은 매개변수로 받은 메서드에서 에러가 나지 않는지 테스트합니다.	바이트배열 타입을 이미지 타입으로 변환하여 반환합니다.	바이트배열 타입을 이미지 타입으로 변환하여 반환합니다.	PASS

# 단위 테스트(지성욱)

테스트 대상 상세 :				
머신 서버와 머신 클라이언트 간 상태변화				
테스트 시나리오 목적 :				
머신 서버와 클라이언트가 실제로 프로그래머 의도대로 동작하는지에 대한 확인				
가정 및 필요사항 :				
머신 서버와 클라이언트 연결하며, 커맨드에 대한 정보는 머신 모니터링이 담당				
STEP (ID)	테스트 항목	기대결과	실제결과	비고
ID01	서버에 접속	서버에 접속이 잘 될 것이라 생각	첫 접속 시 이름을 보내지 않아, 연결이 되지 않음.	FAIL
ID02	서버의 메시지를 획득	잘 획득 될 것 이라고 예상	예상과 부합함	PASS
ID03	프로그램 종료 시 서버와 통신이 올바르게 종료되는지	올바르게 종료 될 것이라고 생각함	올바르게 종료되지 못하여 예외 발생함	FAIL
ID04	서버에서 보낸 커맨드가 실행되는지	서버에서 보낸 커맨드가 실행 될 것이라고 생각함	예상과 부합함	PASS
ID05	종료 커맨드가 실행 되는지	종료 될 것이라고 예상함	통신이 올바르게 종료되지 못해 예외가 발생함	FAIL
ID06	재 시작 커맨드가 실행 되는지	프로그램이 재 시작 될 것이라고 예상함	통신이 올바르게 종료되지 못해 예외가 발생함	FAIL
ID07	자재 투입 현황 커맨드가 실행되는지	올바르게 실행 될 것이라고 예상함	예상과 부합함	PASS
ID08	서버로 메시지가 보내 지는지	서버로 메시지가 보내질 것이라고 예상함	예상과 부합함	PASS



# 단위 테스트(지성욱)

테스트 대상 상세 :

머신 서버와 머신 모니터링 클라이언트간 상태변화

테스트 시나리오 목적 :

머신 서버와 머신 모니터링 클라이언트가 실제 프로그래머 의도대로 움직이는지에 대한 확인

가정 및 필요사항 :

머신 서버와 클라이언트 연결합니다.

STEP (ID)	테스트 항목	기대결과	실제결과	비고
ID01	서버에 접속	서버에 접속이 잘 될 것이라 생각	서버와 연결 성공	PASS
ID02	서버의 메시지를 획득	잘 획득 될 것 이라고 예상	예상과 부합함	PASS
ID03	프로그램 종료 시 서버와 통신이 올바르게 종료되는지	올바르게 종료 될 것이라고 생각함	올바르게 종료되지 못하여 예외 발생함	FAIL
ID04	서버로 메시지가 보내 지는지	서버로 메시지가 보내질 것이라고 예상함	예상과 부합함	PASS
ID05	클라이언트에서 만들어진 명령어가 서버로 보내 지는지	만들어진 커맨드가 서버로 보내 질 것이라고 예상함	예상과 부합함	PASS

# 단위 테스트(지성욱)

테스트 대상 상세 :				
주문 조회의 상태 변화				
테스트 시나리오 목적 :				
주문 조회가 정확히 이루어 지는지에 대한 시나리오				
가정 및 필요사항 :				
특정 거래처를 검색해 해당 거래를 조회하는 것을 가정함				
STEP (ID)	테스트 항목	기대결과	실제결과	비고
ID01	주문 조회 실행 시 모든 주문에 대한 결과 출력	모든 결과가 출력 될 것 이라고 예상함	예상과 부합함	PASS
ID02	조성호 그룹을 검색	'조성' 을 키워드로 검색해 조성호 그룹의 주문 내역이 보일 것이라고 예상함	예상과 부합함	PASS
ID03	검색된 주문의 상세정보를 조회함	해당 주문의 상세 정보가 보여질 것이라고 예상함	예상과 부합함	PASS
ID04	상세 주문의 총액이 계산 되는지	모든 품목의 총액을 계산 해 보여 줄 것이라고 예상함	예상과 부합함	PASS

# 단위 테스트(지성욱)

테스트 대상 상세 :				
품목 바코드 출력에 대한 상태변화				
테스트 시나리오 목적 :				
실제로 품목의 코드를 가져와 올바른 바코드를 생성하고, 바코드 출력물을 생성하는지를 확인함				
가정 및 필요사항 :				
코카콜라를 검색해 바코드를 생성하고, 42 개의 출력물을 뽑는다고 업무를 가정함.				
STEP (ID)	테스트 항목	기대결과	실제결과	비고
ID01	프로그램 구동 시 품목의 항목을 가져오는가	DB 의 품목 테이블과 일치하게 가져 올 것이라고 예상함	예상과 부합함	PASS
ID02	코카콜라로 검색해 결과가 도출되는가	검색이 되어 올바른 품목을 나타낼 것이라고 예상함	예상과 부합함	PASS
ID03	해당 품목으로 바코드 생성이 되는가	올바른 바코드의 생성이 가능 할 것이라고 예상함	예상과 부합함	PASS
ID04	개수 입력 칸에 숫자가 아닌 문자 입력	숫자만 입력 해 달라고 메시지 호출 후 리턴	예상과 부합함, 예외처리 되어 있음	PASS
ID05	개수 입력 칸에 42개 이상 입력	A4 용지 한 장당 최대 개수 42 개라고 메시지 호출 후 42 개 고정	예상과 부합함	FAIL
ID06	개수 입력 칸에 0 입력	바코드 리스트가 만들어지지 않을 것이라고 예상함	예상과 다름, 비어있는 이미지 파일이 생성됨	PASS
ID07	이미지 출력	올바른 이미지가 출력됨	예상과 부합함	PASS
ID08	실제 출력된 이미지를 스캐너로 인식	스캐너로 인식되어 만들어진 바코드의 값이 나올 것이라고 예상함	예상과 부합함 스캐너 스캔 결과 코카콜라의 코드가 나타남	PASS

# 단위 테스트(지성욱)

테스트 대상 상세 :				
머신 서버와 머신 클라이언트 간 상태변화				
테스트 시나리오 목적 :				
머신 서버와 클라이언트가 실제로 프로그래머 의도대로 동작하는지에 대한 확인				
가정 및 필요사항 :				
머신 서버와 클라이언트 연결하며, 커맨드에 대한 정보는 머신 모니터링이 담당				
STEP (ID)	테스트 항목	기대결과	실제결과	비고
ID01	서버에 접속	서버에 접속이 잘 될 것이라 생각	첫 접속 시 이름을 보내지 않아, 연결이 되지 않음.	FAIL
ID02	서버의 메시지를 획득	잘 획득 될 것 이라고 예상	예상과 부합함	PASS
ID03	프로그램 종료 시 서버와 통신이 올바르게 종료되는지	올바르게 종료 될 것이라고 생각함	올바르게 종료되지 못하여 예외 발생함	FAIL
ID04	서버에서 보낸 커맨드가 실행되는지	서버에서 보낸 커맨드가 실행 될 것이라고 생각함	예상과 부합함	PASS
ID05	종료 커맨드가 실행 되는지	종료 될 것이라고 예상함	통신이 올바르게 종료되지 못해 예외가 발생함	FAIL
ID06	재 시작 커맨드가 실행 되는지	프로그램이 재 시작 될 것이라고 예상함	통신이 올바르게 종료되지 못해 예외가 발생함	FAIL
ID07	자재 투입 현황 커맨드가 실행되는지	올바르게 실행 될 것이라고 예상함	예상과 부합함	PASS
ID08	서버로 메시지가 보내 지는지	서버로 메시지가 보내질 것이라고 예상함	예상과 부합함	PASS

# 단위 테스트(손종완)

테스트 대상 상세 :				
해당 클래스에서 메서드를 호출해서 사용했을때 반응				
테스트 시나리오 목적 :				
해당 클래스의 메서드를 사용하여 제대로 기능을 하는지를 살펴봄				
가정 및 필요사항 :				
Excel 창은 이미 닫겨 있어야 하며 제대로 된 기능을 사용하기 위해 Excel이 설치되어 있어야함				

STEP (ID)	테스트 항목	기대결과	실제결과	비고
ID01	생산 계획서를 출력 선택하였을때 제대로 출력되는지	생산견적서를 원하는 경로에 출력	생산견적서가 제대로 출력도미	PASS
ID02	견적서를 출력 선택 하였을때 제대로 출력되는지	견적서가 경로에 출력됨	견적서가 제대로 출력됨	PASS
ID03	원하는 경로에 해당 문서가 출력되는지	원하는 경로에 문서가 출력됨	선택 경로에 출력됨	PASS
ID04	입고증명서가 원하는 경로에 값들이 제대로 출력되는지	입고증명서가 값이 넣어져 경로에 출력됨	입고증명서가 제대로 출력됨	PASS
ID05	이미 열려져있는 엑셀파일에 값이 출력되는지	해당 파일을 사용할 수 없음이 뜸	오류가 발생	FAIL
ID06	출고 증명서가 제대로 값이 넣어져 출력되는지	출고 증명서가 값이 넣어져 출력됨	출고증명서가 선택 경로에 출력됨	PASS
ID07	입고 증명서가 제대로 값이 넣어져 출력되는지	입고 증명서가 값이 넣어져 경로에 출력됨	입고 증명서가 선택 경로에 출력됨	PASS
ID08	매출 전표가 제대로 값이 넣어져 출력되는지	매출 전표가 경로에 출력됨	매출 전표가 선택 경로에 출력됨	PASS
ID09	매입 전표가 제대로 값이 넣어져 출력되는지	매입 전표가 경로에 출력됨	매입 전표가 경로에 출력됨	PASS

# 단위 테스트(손종완)

테스트 대상 상세 :	
해당 클래스에서 메서드를 호출해서 사용했을때 반응	
테스트 시나리오 목적 :	
해당 클래스의 메서드를 사용하여 제대로 기능을 하는지를 살펴봄	
가정 및 필요사항 :	
해당 DB와 연결되어있어야하며 인터넷 또한 원활한 속도가 되어야함	

STEP (ID)	테스트 항목	기대결과	실제결과	비고
ID01	물류데이터를 조회 클릭하였을때 검색이 제대로 되는지	검색 결과가 DB내용과 일치하게 출력	제대로 된 값이 출력됨	PASS
ID02	주문 코드를 선택하여 넣었을때 제대로 주문 코드입력 텍스트 박스에 입력되는지	해당값과 동일한 값이 입력	제대로 된 값이 주문 코드 입력 텍스트 박스에 입력	PASS
ID03	창고 코드 검색을 눌렀을때 제대로 된 값이 내보내 지는지	해당값과 동일한 값이 입력	제대로 된 값이 주문 코드 입력 텍스트 박스에 입력	PASS
ID04	검색을 눌렀을대 제대로 된 검색 결과가 나오는 지	해당 값들로 일치 되는 검색결과가 출력	검색결과가 검색 조건과 일치하는 값이 출력됨	PASS

# 단위 테스트(손종완)

테스트 대상 상세 :	
해당 클래스에서 메서드를 호출해서 사용했을때 반응	
테스트 시나리오 목적 :	
해당 클래스의 메서드를 사용하여 제대로 기능을 하는지를 살펴봄	
가정 및 필요사항 :	
해당 DB와 연결되어있어야하며 인터넷 또한 원활한 속도가 되어야함	

STEP (ID)	테스트 항목	기대결과	실제결과	비고
ID01	견적서를 조회 클릭하였을때 검색이 제대로 되는지	검색 결과가 DB내용과 일치하게 출력	제대로 된 값이 출력됨	
ID02	주문 코드를 선택하여 넣었을때 제대로 주문 코드입력 텍스트 박스에 입력되는지	해당값과 동일한 값이 입력	제대로 된 값이 주문 코드 입력 텍스트 박스에 입력	
ID03	엑셀 내보내기를 눌렀을때 제대로 된 값이 내보내지는지	데이터그리드뷰에 있는 내용과 동일한 내용이 출력됨	검색 결과로 나온 데이터그리드뷰의 내용이 엑셀 파일로 출력됨	

# 단위 테스트(손종완)

테스트 대상 상세 :	
해당 클래스에서 메서드를 호출해서 사용했을때 반응	
테스트 시나리오 목적 :	
해당 클래스의 메서드를 사용하여 제대로 기능을 하는지를 살펴봄	
가정 및 필요사항 :	
해당 DB와 연결되어있어야하며 인터넷 또한 원활한 속도가 되어야함	

STEP (ID)	테스트 항목	기대결과	실제결과	비고
ID01	출고 증명서를 검색 하였을때 제대로 된 검색 결과가 나오는지	검색 결과가 DB내용과 일치하게 출력	제대로 된 값이 출력됨	
ID02	참고 코드를 선택하여 넣었을때 제대로 참고 코드입력 텍스트 박스에 입력되는지	해당값과 동일한 값이 입력	제대로 된 값이 참고 코드 입력 텍스트 박스에 입력	
ID03	엑셀 내보내기를 눌렀을때 제대로 된 값이 내 보내 지는지	데이터그리드뷰에 있는 내용과 동일한 내용이 출력됨	검색 결과로 나온 데이터그리드뷰의 내용이 엑셀 파일로 출력됨	



# 단위 테스트(손종완)

테스트 대상 상세 :	
해당 클래스에서 메서드를 호출해서 사용했을때 반응	
테스트 시나리오 목적 :	
해당 클래스의 메서드를 사용하여 제대로 기능을 하는지를 살펴봄	
가정 및 필요사항 :	
해당 DB와 연결되어있어야하며 인터넷 또한 원활한 속도가 되어야함	

STEP (ID)	테스트 항목	기대결과	실제결과	비고
ID01	입고 증명서를 검색 하였을때 제대로 된 검색 결과가 나오는지	검색 결과가 DB내용과 일치하게 출력	제대로 된 값이 출력됨	
ID02	창고 코드를 선택하여 넣었을때 제대로 창고 코드입력 텍스트 박스에 입력되는지	해당값과 동일한 값이 입력	제대로 된 값이 창고 코드 입력 텍스트 박스에 입력	
ID03	엑셀 내보내기를 눌렀을때 제대로 된 값이 내 보내 지는지	데이터그리드뷰에 있는 내용과 동일한 내용이 출력됨	검색 결과로 나온 데이터그리드뷰의 내용이 엑셀 파일로 출력됨	

# 단위 테스트(손종완)

테스트 대상 상세 :	
해당 클래스에서 메서드를 호출해서 사용했을때 반응	
테스트 시나리오 목적 :	
해당 클래스의 메서드를 사용하여 제대로 기능을 하는지를 살펴봄	
가정 및 필요사항 :	
해당 DB와 연결되어있어야하며 인터넷 또한 원활한 속도가 되어야함	

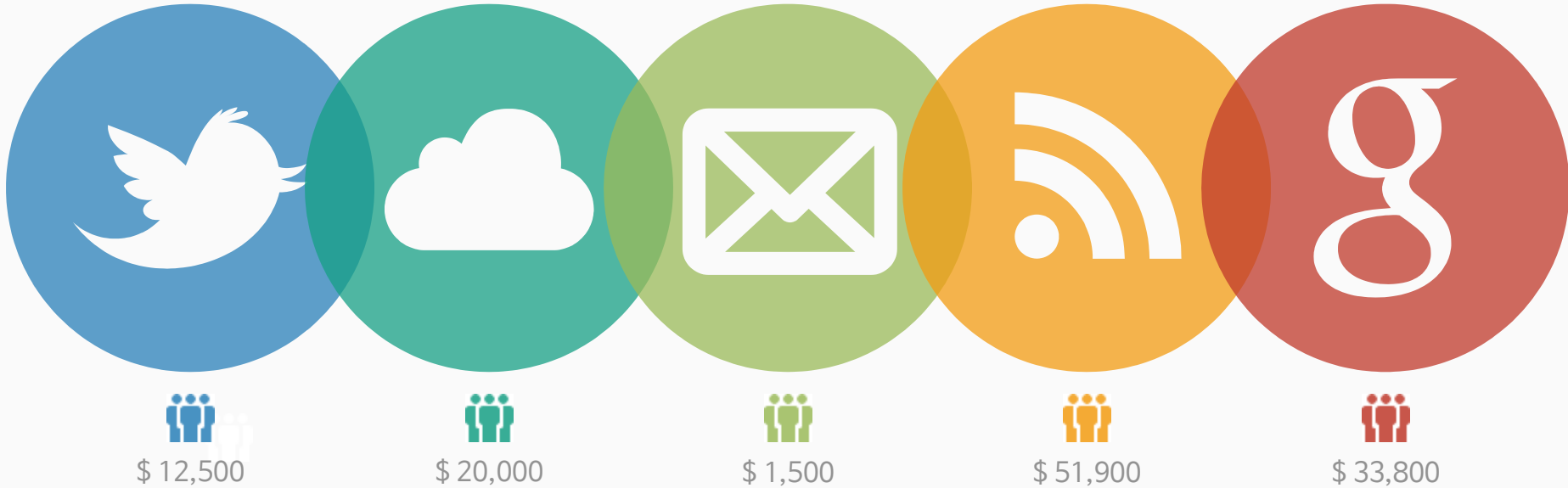
STEP (ID)	테스트 항목	기대결과	실제결과	비고
ID01	생산 계획서를 조회 클릭하였을때 검색이 제대로 되는지	검색 결과가 DB내용과 일치하게 출력	제대로 된 값이 출력됨	
ID02	주문 코드를 선택하여 넣었을때 제대로 주문 코드 입력 텍스트 박스에 입력되는지	해당값과 동일한 값이 입력	제대로 된 값이 주문 코드 입력 텍스트 박스에 입력	
ID03	엑셀 내보내기를 눌렀을때 제대로 된 값이 내보내지는지	데이터그리드뷰에 있는 내용과 동일한 내용이 출력됨	검색 결과로 나온 데이터그리드뷰의 내용이 엑셀 파일로 출력됨	

## 결함/오류조치

ID	결함/오류	조치
TRG_TRADE_02	거래테이블 UPDATE 트리거에서 잘못된 데이터가 입력되었을 때 에러메시지가 제대로 출력되지않음.	에러가 날수있는 부분을 좀 더 세분화하여 예외처리를 하여 해결함.
GET_SAMPLE_SIMPLE_01	GET_SAMPLE_SIMPLE 프로시저에 잘못된 데이터(존재하지않는 주문번호)를 입력하였을 때 에러메시지가 출력되지않음.	주문테이블에 해당 데이터가 있는지 여부를 우선적으로 판단하도록 수정함. 만약, 없을경우 오류메시지 출력하도록 처리함
SET_DISTRIBUTION_01	SET_DISTRIBUTION 프로시저의 매개변수 입력데이터에 "."가 아닌 다른 구분자를 사용하여 입력하였는데도 오류메시지가 출력되지않고 트랜잭션 ROLLBACK이 수행되지않음.	예외처리구문이 누락되어있어 해당 부분에 예외처리구문을 추가하여 해결함.
ORDER_02	데이터에 숫자만 입력되어야 하는데 문자가 입력되었는데도 예외처리가 발생하지않음.	INT32.TryParse 함수를 이용해 예외처리구문을 추가하여 해결함
DISTRIBUTION_03	선택된 현재창고를 제외한 창고목록이 출력되어야 하는데 현재창고도 출력됨.	창고목록객체에서 현재창고에 선택된 창고를 삭제하여 해결함.
DISTRIBUTION_07	ListViewItem객체를 서로다른 ListView에서 참조하게 되어 동일객체참조오류 발생	ListViewItem의 Clone 함수를 통해 객체를 복사하여 해결
DISTRIBUTION_09	유효성검사를 통과하지않았는데도 에러메시지가 출력되지 않음.	예외처리구문에 오류가 있어 해당부분을 수정

# 감사합니다

This sub-title is example slide



If a picture paints a thousand words, then why can't I paint you  
The words will never show the you I've come to know  
if a face could launch a thousand ships, then where am I to go  
There's no one home but you  
You're all that's left me too  
And when my love for life is running dry,  
you come and pour yourself on me  
If a man could be two places at one time  
I'd be with you  
Tomorrow and today, beside you all the way  
If the world should stop revolving  
spinning slowly down to die, I'd spend the end with you  
And when the world was through, Then one by one  
the stars would all go out and you and I would simply fly away