

<똑배기>

<MrGreat>

구현 및 시험 단계 산출물

문서버전	2.0
문서 ID	SE-2020-003
최종변경일	2020-12-02
문서상태	릴리즈

요 약

미스터 대박 디너 서비스(Mr.Great)의 구현 및 시험 단계 산출물을 기술.

주요 산출물

- 컴포넌트도
- 시험 보고서
- 오류 보고서

표 1 문서 변경 기록

문서 이름	<톡배기> <Mr.Great> 구현 및 시험 단계 산출물		
문서 ID	SE-2020-003		
버전		변경일	설명
1	0	2020-11-29	컴포넌트도 추가
	1	2020-11-30	오류 보고서 및 시험 보고서 추가
2	0	2020-12-02	릴리즈

1 개 요

1.1 목 적

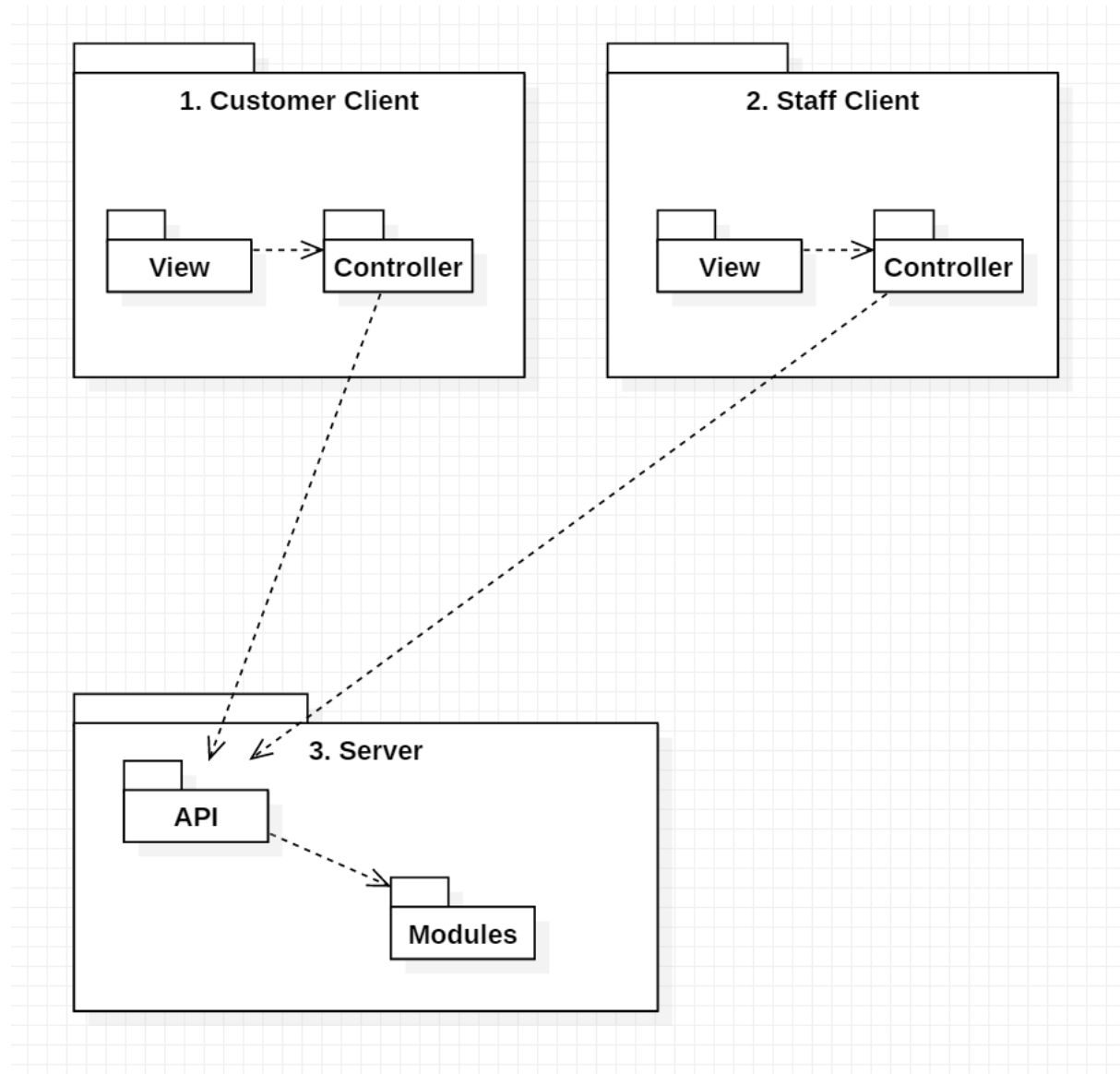
본 문서는 “미스터 대박 디너 서비스”의 구현 및 시험 단계 산출물을 기술한다.

- ✓ 전체 시스템이 어떠한 파일들이나 자원으로 구성되는지를 “컴포넌트도”를 통해서 기술한다
- ✓ 분석 단계에서 기술한 기능적 요구사항이 만족되는지 시험한 결과를 “시험 보고서”를 통해서 보여준다
- ✓ “오류 보고서”를 통해서 구현 단계에서 발생한 주요 오류를 기록하고 향후에 동일한 오류가 발생할 때 대처할 수 있도록 준비한다.

1.2 참고 문헌

없음.

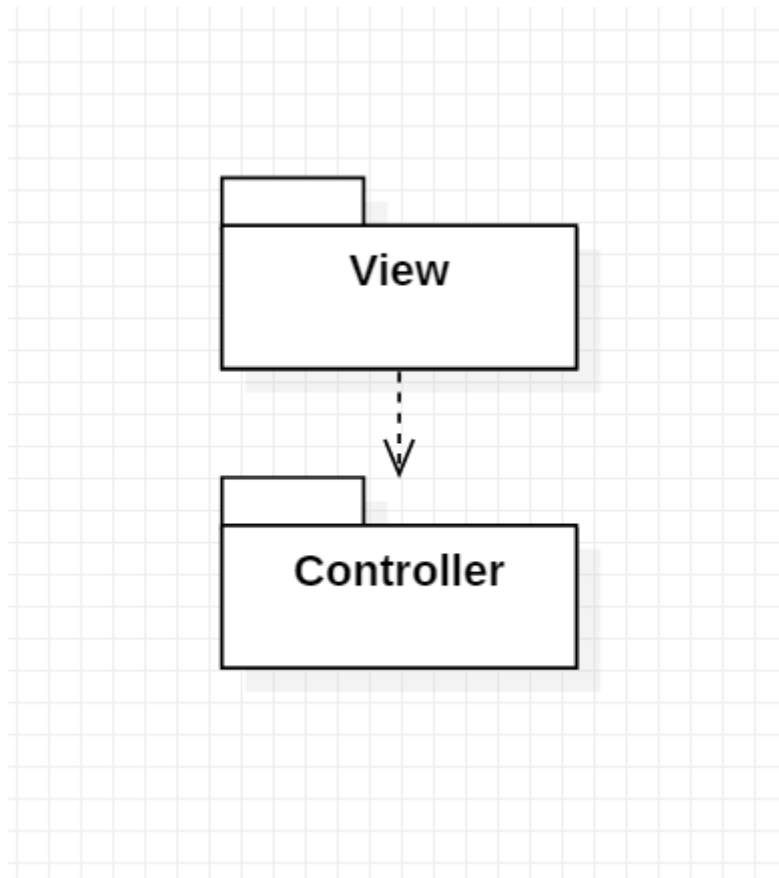
2. 컴포넌트도



다음과 같은 서브시스템을 구현하였다.

- ✓ Customer Client 시스템
- ✓ Staff Client 시스템
- ✓ Server 시스템

2.1 Customer Client 서브시스템



Customer Client 서브시스템은 두 개의 컴포넌트 패키지로 구성된다.

- ✓ View 컴포넌트 패키지
- ✓ Controller 컴포넌트 패키지

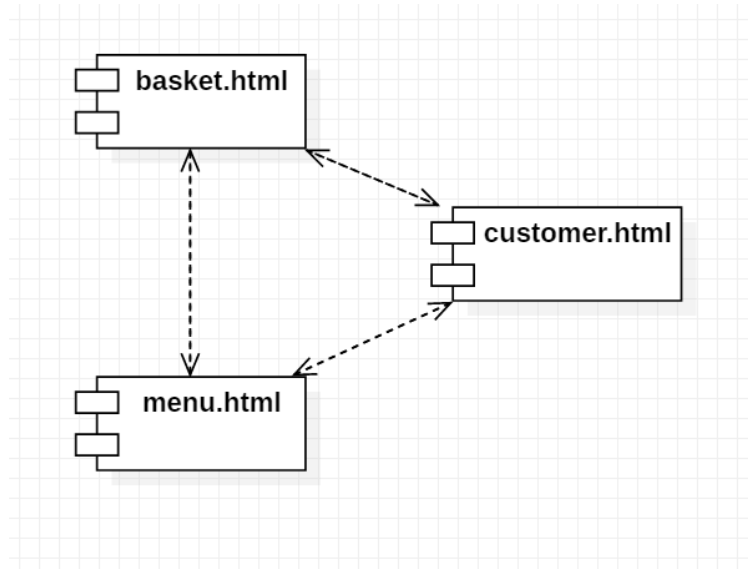
View 컴포넌트 패키지

- ✓ Customer client 의 화면 구성을 출력한다.
- ✓ Controller 컴포넌트 패키지를 사용하여 고객의 요청에 따른 적절한 처리를 하고, 응답을 전송하는 작업을 수행한다.

Controller 컴포넌트 패키지

- ✓ 화면에 동적으로 출력할 것이 있다면 Controller 에서 필요한 컴포넌트를 호출하여 출력한다.
- ✓ Server 에 있는 API 컴포넌트 패키지를 사용하여 고객에게 필요한 처리나 데이터를 요청한다.

2.1.1 View 컴포넌트 패키지 구성



customer.html 컴포넌트

- ✓ 고객에게 메인 화면을 출력한다.
- ✓ “매뉴” 또는 “장바구니”를 클릭하면 화면을 이동할 수 있다.

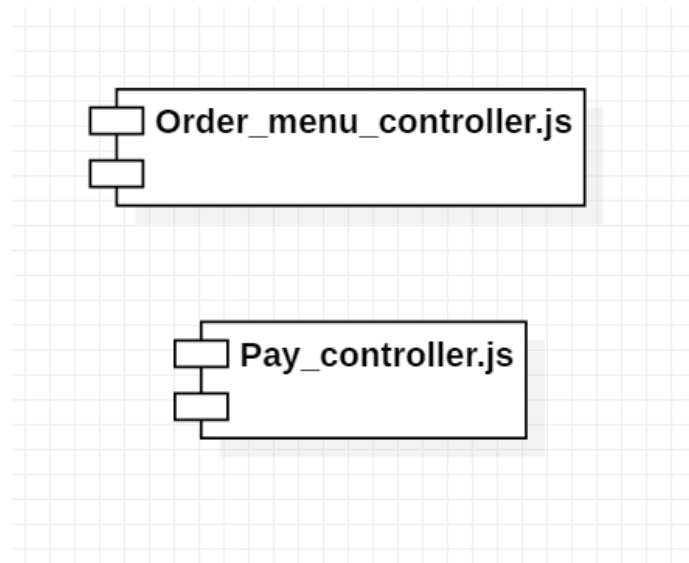
menu.html 컴포넌트

- ✓ 고객에게 메뉴 화면을 출력한다.
- ✓ Controller 컴포넌트 패키지를 이용하여 고객이 주문하기를 누르면 원하는 주문을 수행할 수 있도록 한다.

basket.html 컴포넌트

- ✓ 고객에게 장바구니 화면을 출력한다.
- ✓ Controller 컴포넌트 패키지를 이용하여 고객이 제외하고 싶은 주문을 제외할 수 있도록 한다.
- ✓ Controller 컴포넌트 패키지를 이용하여 고객이 결제를 누르면 결제를 수행할 수 있도록 한다.

2.1.2 Controller 컴포넌트 패키지 구성



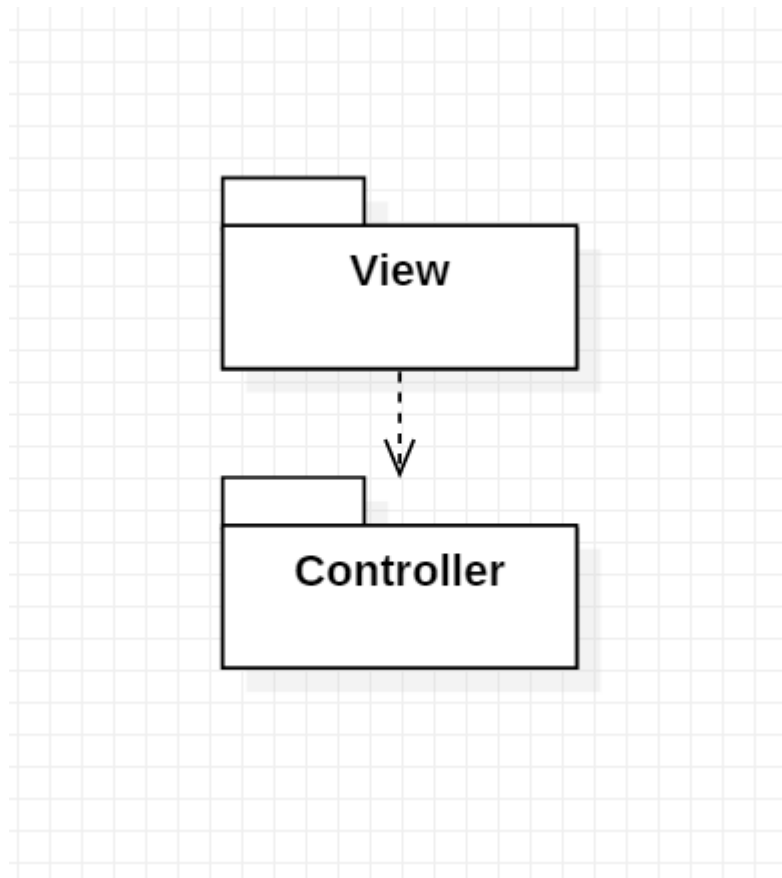
Order_menu_controller.js 컴포넌트

- ✓ display_recent_menu 를 호출하면 API 에 요청을 하여 고객의 최근 주문을 받고 View 에 출력한다.
- ✓ add_order 를 호출하면 View 에서 받은 주문 정보를 API 요청을 통해 서버에 전달하여 고객의 장바구니에 주문을 추가한다.

Pay_controller.js 컴포넌트

- ✓ display_basket 을 호출하면 API 에 요청을 하여 고객의 장바구니 목록을 받고 View 에 출력한다.
- ✓ remove_order 를 호출하면 API 에 요청을 하여 고객의 장바구니에서 선택한 주문을 제외한다.
- ✓ pay 를 호출하면 View 에서 받은 결제 정보를 API 요청을 통해서 전달하여 결제를 수행한다.

2.2 Staff Client 서비스 시스템



Staff Client 서비스 시스템은 두 개의 컴포넌트 패키지로 구성된다.

- ✓ View 컴포넌트 패키지
- ✓ Controller 컴포넌트 패키지

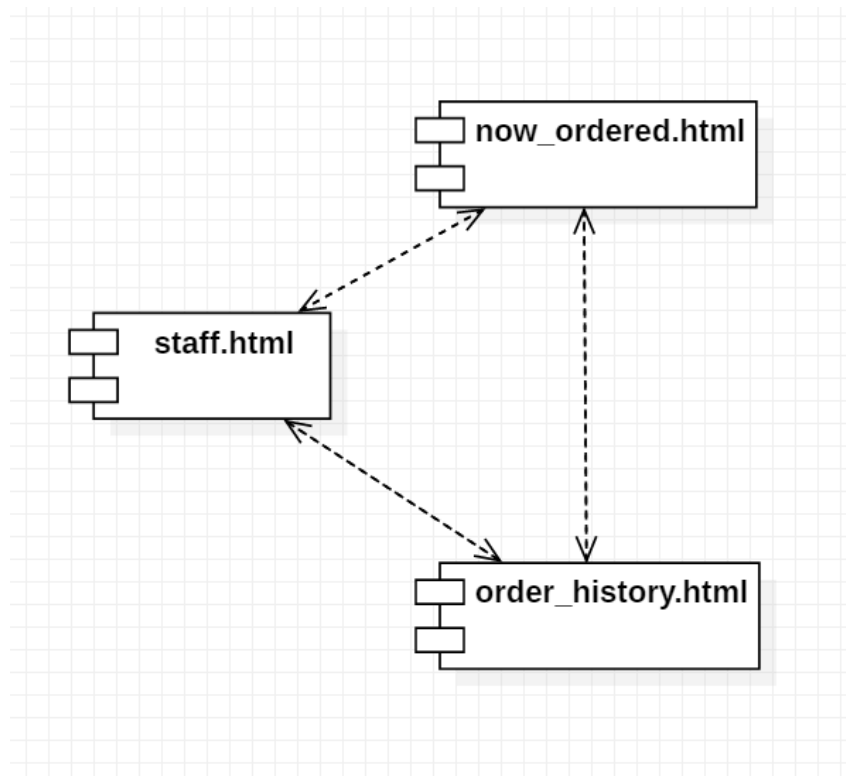
View 컴포넌트 패키지

- ✓ Staff client 의 화면 구성을 출력한다.
- ✓ Controller 컴포넌트 패키지를 사용하여 직원의 요청에 따른 적절한 처리를 하고, 응답을 전송하는 작업을 수행한다.

Controller 컴포넌트 패키지

- ✓ 화면에 동적으로 출력할 것이 있다면 Controller 에서 필요한 컴포넌트를 호출하여 출력한다.
- ✓ Server 에 있는 API 컴포넌트 패키지를 사용하여 직원에게 필요한 처리나 데이터를 요청한다.

2.2.1 View 컴포넌트 패키지 구성



staff.html 컴포넌트

- ✓ 직원에게 메인 화면을 출력한다.
- ✓ “Now Ordered” 또는 “Order History”를 클릭하면 화면을 이동할 수 있다.

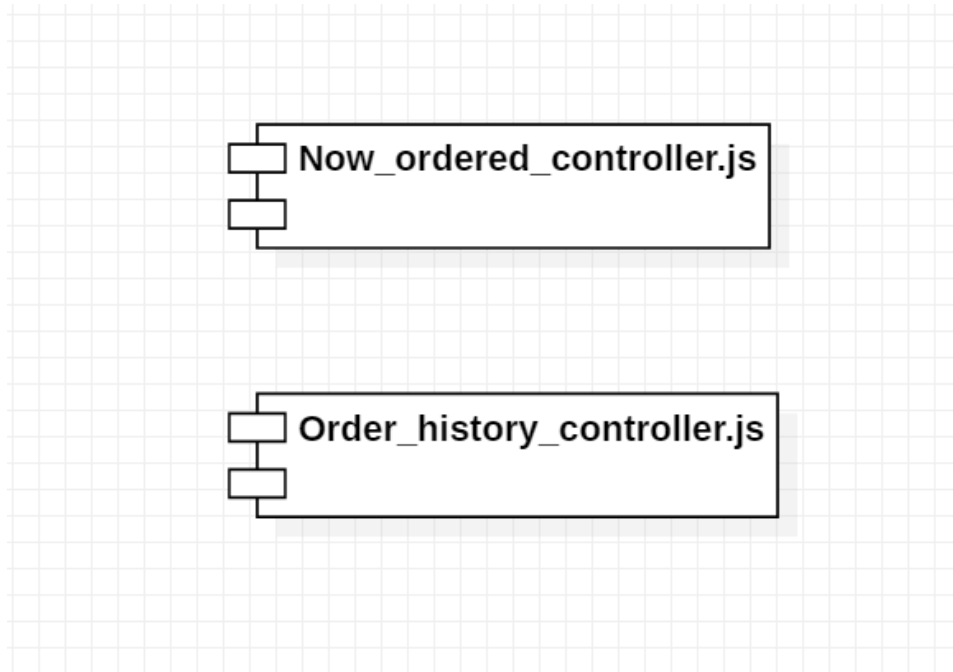
now_ordered.html 컴포넌트

- ✓ 직원에게 현재 주문 목록 화면을 출력한다.
- ✓ Controller 컴포넌트 패키지를 이용하여 직원이 주문의 상태를 변환하려 하면 선택한 주문의 상태 전환을 수행할 수 있도록 한다.

order_history.html 컴포넌트

- ✓ 직원에게 완료된 주문 목록 화면을 출력한다.

2.2.2 Controller 컴포넌트 패키지 구성



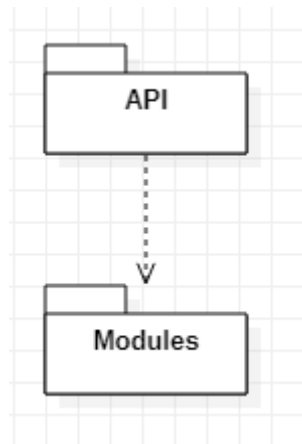
Now_ordered_controller.js 컴포넌트

- ✓ `display_order_lists` 를 호출하면 API 에 요청을 하여 현재 주문 목록을 받고 View 에 출력한다.
- ✓ `change_order_status` 를 호출하면 View 에서 선택한 주문과 전환할 상태를 API 요청을 통해 서버에 전달하여 주문의 상태를 전환한다.

Order_history_controller.js 컴포넌트

- ✓ `display_order_history` 을 호출하면 API 에 요청을 하여 완료된 주문 목록을 받고 View 에 출력한다.

2.3 Server 서브시스템



Server 서브시스템은 두 개의 컴포넌트 패키지로 구성된다.

- ✓ API 컴포넌트 패키지
- ✓ Modules 컴포넌트 패키지

API 컴포넌트 패키지

- ✓ Customer client 와 Staff Client 시스템으로부터 오는 API 요청을 처리한다.
- ✓ Modules 컴포넌트 패키지를 사용하여 요청에 따른 적절한 처리를 하고, 응답을 전송하는 작업을 수행한다.

Modules 컴포넌트 패키지

- ✓ 미스터 대박 서비스에서 사용되는 클래스를 정의하고 구체적인 메소드 구현을 정의한다.
- ✓ API 컴포넌트 패키지에 의해 호출된다.

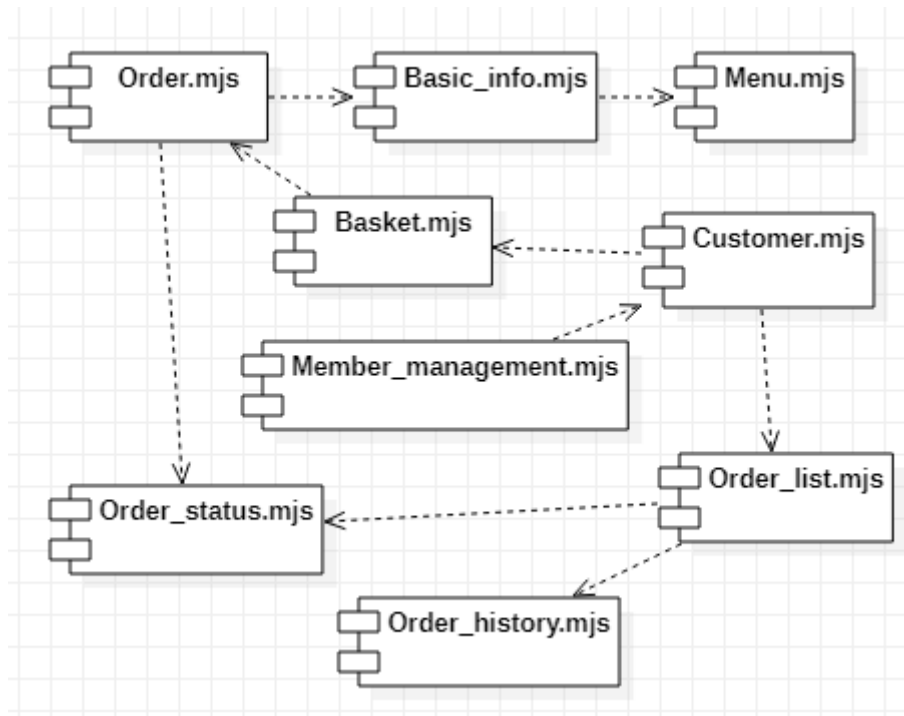
2.3.1 API 컴포넌트 패키지 구성



app.mjs 컴포넌트

- ✓ app.js 컴포넌트는 Customer client 와 Staff client 로부터 오는 REST API 요청을 처리한다.
- ✓ 호출되는 API 에 따라 Modules 컴포넌트 패키지에서 적절한 객체를 생성하고 동작을 처리하며 최종적으로 Client 에게 응답하는 동작을 한다.

2.3.2 Modules 컴포넌트 패키지 구성



Menu.mjs 컴포넌트

- ✓ Menu 클래스를 구현하는 컴포넌트이다.

Basic_info.mjs 컴포넌트

- ✓ Basic_info 클래스를 구현하는 컴포넌트이다.

Order.mjs 컴포넌트

- ✓ Order 클래스를 구현하는 컴포넌트이다.

Basket.mjs 컴포넌트

- ✓ Basket 클래스를 구현하는 컴포넌트이다.

Customer.mjs 컴포넌트

- ✓ Customer 클래스를 구현하는 컴포넌트이다.

Member_management.mjs 컴포넌트

- ✓ Member_management 클래스를 구현하는 컴포넌트이다.

Order_list.mjs 컴포넌트

- ✓ Order_list 클래스를 구현하는 컴포넌트이다.

Order_history.mjs 컴포넌트

- ✓ Order_history 클래스를 구현하는 컴포넌트이다.

Order_status.mjs 컴포넌트

- ✓ Order_status 클래스를 구현하는 컴포넌트이다.

3. 시험 보고서

일련번호	테스트 케이스
TR1	고객 회원가입 테스트
TR2	직원 로그인 테스트
TR3	고객 로그인 테스트
TR4	고객 주문 테스트
TR5	고객 결제 테스트
TR6	직원 주문 처리 테스트
TR7	직원 주문 내역 확인 테스트

일련번호	TR1	테스트케이스	고객 회원가입 테스트
상세설명	고객이 회원가입하는 상황을 가정하여 고객의 정보가 정상적으로 서버에 저장되는지 검사한다.		
테스트 방법	“회원 가입”을 선택하여 회원가입 화면으로 이동한다. 아이디, 비밀번호, 추가 정보를 입력하고 정상적으로 가입되는지 검사한다. 올바르지 않은 입력에 대해서 실패 메시지를 잘 출력하는지 검사한다. 최종적으로 서버 측에 입력받은 정보가 정상적으로 저장되는지 검사한다.		
성공 조건	올바르지 않은 입력에 대해선 실패 메시지를 출력하고, 올바른 입력에 대해 서버 측에 고객의 정보가 정상적으로 저장된다.		
유즈케이스	회원가입 (MG-UC1)		
결과	성공		

일련번호	TR2	테스트케이스	직원 로그인 테스트
상세설명	직원이 로그인하는 상황을 가정하여 시스템이 직원을 식별하고 직원의 메인 페이지로 잘 이동시켜주는지 검사한다.		
테스트 방법	로그인 화면에서 아이디, 비밀번호를 입력하고 “로그인”을 선택한다. 아이디, 비밀번호가 일치한다면 직원 정보를 정상적으로 로드하는지 검사한다. 아이디, 비밀번호가 일치하지 않는다면 실패 메시지를 잘 출력하는지 검사한다. 최종적으로 서버 측에 로그인 된 Customer 가 activated 되었는지, 직원 메인 페이지로 이동했는지 검사한다.		
성공 조건	서버 측에 로그인된 직원이 activated 되고 직원 메인 페이지로 이동한다.		
유즈케이스	로그인 (MG-UC2)		
결과	성공		

일련번호	TR3	테스트케이스	고객 로그인 테스트
상세설명	고객이 로그인하는 상황을 가정하여 시스템이 고객을 잘 식별하고 고객의 메인 페이지로 잘 이동시켜주는지 검사한다.		
테스트 방법	로그인 화면에서 아이디, 비밀번호를 입력하고 “로그인”을 선택한다. 아이디, 비밀번호가 일치한다면 Customer 정보를 정상적으로 로드하는지 검사한다. 아이디, 비밀번호가 일치하지 않는다면 실패 메시지를 잘 출력하는지 검사한다. 최종적으로 서버 측에 로그인 된 Customer 가 activated 되었는지, 고객 메인 페이지로 이동했는지 검사한다.		
성공 조건	서버 측에 로그인된 Customer 가 activated 되고 고객 메인 페이지로 이동한다.		
유즈케이스	로그인 (MG-UC2)		
결과	성공		

일련번호	TR4	테스트케이스	고객 주문 테스트
상세설명	고객이 주문하는 상황을 가정하여 시스템이 입력받은 대로 정확한 정보를 올바른 Customer 객체에 저장하는지 검사한다.		
테스트 방법	“메뉴”를 선택하여 메뉴 목록 페이지로 이동한다. 메뉴 중 하나를 선택하고 세부 주문 내역을 입력한다. “주문하기”를 선택한다. 서버 측에 주문 정보가 정상적으로 저장되었는지 검사한다. 서버에서 주문이 들어올 시 로그를 남기도록 하여 로그를 검사한다.		
성공 조건	주문 정보가 서버에 정상적으로 저장되고 고객의 “장바구니”에서 확인할 수 있다.		
유즈케이스	주문 (MG-UC3)		
결과	성공		

일련번호	TR5	테스트케이스	고객 결제 테스트
상세설명	고객이 결제하는 상황을 가정하여 고객의 결제 정보가 정상적으로 서버에 저장되는지 검사한다.		
테스트 방법	“장바구니”를 선택하여 장바구니 화면으로 이동한다. 고객이 주문한 목록이 정상적으로 보이는지 검사한다. 그리고 주문한 목록이 없다면 결제 버튼이 비활성화 되는지 확인한다. “제외하기”를 누르면 선택한 주문이 정상적으로 제외되는지 확인한다. “결제 수단 선택”을 누르고 결제 수단을 선택한 후 “결제”를 누른다. 최종적으로 서버 측에 결제 정보가 정상적으로 저장되는지 검사한다.		
성공 조건	서버 측에 결제 정보가 정상적으로 저장된다.		
유즈케이스	결제 (MG-UC4)		
결과	성공		

일련번호	TR6	테스트케이스	직원 주문 처리 테스트
상세설명	직원이 주문을 처리하는 상황을 가정하여 현재까지 주문한 목록이 직원에게 잘 보여지고 주문의 상태가 잘 전환되는지 검사한다.		
테스트 방법	“Now ordered”를 선택하여 현재 주문 목록 페이지로 이동한다. 현재까지 주문한 목록이 “조리 전” 목록에 출력되는지 검사한다. 상태를 전환할 주문을 선택한다. “▼” 버튼을 누르면 주문의 상태가 잘 전환되는지 검사한다. 또한 “Complete” 버튼을 누르면 주문이 서버 측에 정상적으로 저장되는지 검사한다.		
성공 조건	“조리 전”인 주문은 “조리 중”으로 전환된다. “조리 중”인 주문은 “배달 중”으로 전환된다. “배달 중”인 주문은 서버에 정상적으로 저장되고 “Order history”에서 확인할 수 있다.		
유즈케이스	주문 처리 (MG-UC5)		
결과	성공		

일련번호	TR7	테스트케이스	직원 주문 내역 확인 테스트
상세설명	직원이 완료된 주문을 확인하는 상황을 가정하여 지금까지 완료된 주문이 정상적으로 로드되는지 검사한다.		
테스트 방법	“Order history”를 선택하여 완료된 주문 목록 페이지로 이동한다. 현재까지 완료된 주문 목록을 정상적으로 로드되는지 검사한다.		
성공 조건	서버 측에 저장된 현재까지 완료된 주문 목록이 정상적으로 로드된다.		
유즈케이스	주문 내역 확인 (MG-UC6)		
결과	성공		

4. 오류 보고서

일련번호	설명	해결여부
BR1	최근 주문 목록이 최신 순으로 출력되지 않음	해결
BR2	결제 금액이 0 원인데 결제가 수행됨	해결
BR3	최근 주문 내역과 단골 혜택이 즉시 적용되지 않음	해결
BR4	모든 메뉴를 0 개씩 선택하면 결제 금액이 음수가 됨	해결
BR5	메뉴 주문 시 수량에 비정상 입력이 들어와도 주문이 가능함	해결
BR6	샴페인 축제 디너의 가격이 너무 과하게 책정됨	해결

일련번호	BR1	설명	최근 주문 목록이 최신 순으로 출력되지 않음	
요구사항	주문(MG-UC3)			
발생환경	고객이 메뉴를 확인하고 있으며 2 회 이상의 주문을 하여 최근 주문 목록이 있을 경우 발생			
발생원인	최근 4 개의 주문을 불러올 때 인덱스를 앞에서 뒤로 출력하지 않고 뒤에서 앞으로 출력함			
위험도	보통		해결여부	해결

일련번호	BR2	설명	결제 금액이 0 원인데 결제가 수행됨	
요구사항	결제(MG-UC4)			
발생환경	고객이 아무 메뉴도 주문을 하지 않고 장바구니에서 결제 버튼을 누르는 경우 발생			
발생원인	결제할 주문이 없을 경우 예외 처리를 하지 않음			
위험도	높음		해결여부	해결

일련번호	BR3	설명	최근 주문 내역과 단골 혜택이 즉시 적용되지 않음	
요구사항	주문(MG-UC3), 결제(MG-UC4)			
발생환경	고객이 주문을 완료한 직후 최근 주문 내역이 갱신되지 않음			
	고객이 3 회 이상 주문을 완료한 직후 단골 혜택이 즉시 적용되지 않음			
발생원인	로그인 시점에서 “Client” 시스템이 “Server” 시스템에서 고객 객체를 받아 그 객체를 갱신하지 않으며 참조함			
위험도	보통		해결여부	해결

일련번호	BR4	설명	모든 메뉴를 0 개씩 선택하면 결제 금액이 음수가 됨	
요구사항	주문(MG-UC3), 결제(MG-UC4)			
발생환경	고객이 메뉴를 선택하고 세부 옵션에서 모두 0 개를 선택함			
발생원인	메뉴와 그 메뉴의 세부 옵션에 대한 가격 설정을 잘못함			
위험도	매우 높음		해결여부	해결

일련번호	BR5	설명	메뉴 주문 시 수량에 비정상 입력이 들어와도 주문이 가능함	
요구사항	주문(MG-UC3)			
발생환경	고객이 주문 상세 내용을 작성할 때 수량에 숫자 외 다른 값을 입력한 뒤, “장바구니에 담기”를 선택하면 발생			
발생원인	‘수량’ 항목이 숫자가 아닌 경우 예외 처리를 해주지 않음			
위험도	높음		해결여부	해결

일련번호	BR6	설명	삼페인 축제 디너의 가격이 너무 과하게 책정됨	
요구사항	주문(MG-UC3)			
발생환경	고객이 바게트빵의 개수를 수정하면 발생			
발생원인	바게트빵의 가격 계산을 올바르게 수행하지 않음			
위험도	높음		해결여부	해결