API 명세서

1. 인증 (Authentication)

1.1 회원가입

```
POST /api/v1/auth/signup
```

Request Body:

```
{ "email": "user@example.com", "name": "홍길동", "password": "password123" }
```

Response:

```
{ "id": 1, "email": "user@example.com", "name": "홍길동", "createdAt": "2024-01-01T00:00:00" }
```

1.2 로그인

```
POST /api/v1/auth/login
```

Request Body:

```
{ "email": "user@example.com", "password": "password123" }
```

Response:

```
{ "accessToken": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...", "refreshToken": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...", "user": { "id": 1, "email": "user@example.com", "name": "홍길동" } }
```

---

2. 매장 관리 (Stores)

2.1 매장 생성

```
POST /api/v1/stores
```

Request Body:

```
{ "name": "Downtown Coffee Co.", "location": "1200 Main St, Seattle", "brn": "123-45-67890", "alias": "Downtown Branch", "openTime": "09:00", "closeTime": "21:00", "nShifts": 3 }
```

Response:

```
{ "id": 1, "name": "Downtown Coffee Co.", "location": "1200 Main St, Seattle", "brn": "123-45-67890", "alias": "Downtown Branch", "openTime": "09:00", "closeTime": "21:00", "nShifts": 3, "userId": 1, "createdAt": "2024-01-01T00:00:00" }
```

2.2 매장 목록 조회

```
GET /api/v1/stores
```

Query Parameters:

- `page` (optional): 페이지 번호 (default: 0)
- `size` (optional): 페이지 크기 (default: 10)

Response:

```
{ "content": [ { "id": 1, "name": "Downtown Coffee Co.", "location": "1200 Main St, Seattle", "alias": "Downtown Branch", "openTime": "09:00", "closeTime": "21:00", "nShifts": 3 } ], "totalElements": 1, "totalPages": 1, "size": 10, "number": 0 }
```

2.3 매장 상세 조회

```
GET /api/v1/stores/{storeId}
```

Response:

```
{ "id": 1, "name": "Downtown Coffee Co.", "location": "1200 Main St, Seattle", "brn": "123-45-67890", "alias": "Downtown Branch", "openTime": "09:00", "closeTime": "21:00", "nShifts": 3, "userId": 1, "createdAt": "2024-01-01T00:00:00", "updatedAt": "2024-01-01T00:00:00" }
```

2.4 매장 정보 수정

```
PUT /api/v1/stores/{storeId}
```

Request Body:

```
{ "name": "Downtown Coffee Co.", "location": "1200 Main St, Seattle", "openTime": "08:00", "closeTime": "22:00", "nShifts": 4 }
```

2.5 매장 삭제

```
DELETE /api/v1/stores/{storeId}
```

3. 매장 멤버 관리 (Store Members)

3.1 직원 초대

```
POST /api/v1/stores/{storeId}/members/invite
```

Request Body:

```
{ "email": "staff@example.com", "role": "STAFF", "department": "HALL", "minHoursPerWeek": 20, "hourlyWage": 15000 }
```

Response:

```
{ "inviteCode": "8479-1234", "expiresAt": "2024-01-08T00:00:00" }
```

3.2 초대 코드로 매장 가입

```
POST /api/v1/stores/join
```

Request Body:

```
{ "inviteCode": "8479-1234" }
```

Response:

```
{ "id": 1, "storeId": 1, "userId": 2, "role": "STAFF", "department": "HALL", "status": "ACTIVE", "minHoursPerWeek": 20, "hourlyWage": 15000, "pinCode": "0000" }
```

3.3 매장 멤버 목록 조회

```
GET /api/v1/stores/{storeId}/members
```

Query Parameters:

- `role` (optional): MANAGER, STAFF
- `status` (optional): INVITED, ACTIVE
- `department` (optional): HALL, KITCHEN

Response:

```
{ "content": [ { "id": 1, "userId": 2, "name": "김직원", "email": "staff@example.com", "role": "STAFF", "department": "HALL", "rank": null, "hourlyWage": 15000, "minHoursPerWeek": 20, "status": "ACTIVE", "pinCode": "1234", "createdAt": "2024-01-01T00:00:00" } ], "totalElements": 1 }
```

3.4 멤버 정보 수정

```
PUT /api/v1/stores/{storeId}/members/{memberId}
```

Request Body:

```
{ "role": "STAFF", "department": "KITCHEN", "hourlyWage": 16000, "minHoursPerWeek": 30 }
```

3.5 멤버 삭제 (퇴사)

```
DELETE /api/v1/stores/{storeId}/members/{memberId}
```

3.6 핀번호 설정/변경

```
PUT /api/v1/stores/{storeId}/members/{memberId}/pin
```

Request Body:

```
{ "pinCode": "1234" }
```

---

4. 근무 템플릿 관리 (Shift Templates)

4.1 근무 템플릿 생성

```
POST /api/v1/stores/{storeId}/shift-templates
```

Request Body:

```
{ "name": "Opening Shift", "startTime": "06:00", "endTime": "11:00",
"requiredStaff": 2, "shiftType": "NORMAL", "dayType": "WEEKDAY" }
```

Response:

```
{ "id": 1, "storeId": 1, "name": "Opening Shift", "startTime": "06:0
0", "endTime": "11:00", "requiredStaff": 2, "shiftType": "NORMAL",
"dayType": "WEEKDAY" }
```

4.2 근무 템플릿 목록 조회

```
GET /api/v1/stores/{storeId}/shift-templates
```

Query Parameters:

- `shiftType` (optional): NORMAL, PEAK
- `dayType` (optional): WEEKDAY, HOLIDAY

Response:

```
{ "content": [ { "id": 1, "name": "Opening Shift", "startTime": "06:00", "endTime": "11:00", "requiredStaff": 2, "shiftType": "NORMAL", "dayType": "WEEKDAY" }, { "id": 2, "name": "Peak Shift", "startTime": "11:00", "endTime": "15:00", "requiredStaff": 6, "shiftType": "PEAK", "dayType": "WEEKDAY" } ] }
```

4.3 근무 템플릿 수정

```
PUT /api/v1/stores/{storeId}/shift-templates/{templateId}
```

Request Body:

```
{ "name": "Opening Shift", "startTime": "08:00", "endTime": "12:00", "requiredStaff": 3 }
```

4.4 근무 템플릿 삭제

```
DELETE /api/v1/stores/{storeId}/shift-templates/{templateId}
```

---

5. 직원 선호도 관리 (Employee Preferences)

5.1 선호도 일괄 등록/수정

```
POST /api/v1/stores/{storeId}/members/{memberId}/preferences
```

Request Body:

```
{ "preferences": [ { "shiftTemplateId": 1, "dayOfWeek": 1, "type": "PREFERRED" }, { "shiftTemplateId": 1, "dayOfWeek": 2, "type": "NEUTRAL" }, { "shiftTemplateId": 2, "dayOfWeek": 1, "type": "UNAVAILABLE" } ] }
```

Response:

```
{ "message": "Preferences saved successfully", "count": 3 }
```

5.2 선호도 조회

```
GET /api/v1/stores/{storeId}/members/{memberId}/preferences
```

Query Parameters:

- `weekStartDate` (optional): 주간 시작일 (YYYY-MM-DD)

Response:

```
{ "memberId": 1, "memberName": "김직원", "preferences": [ { "id": 1, "shiftTemplateId": 1, "shiftName": "Opening Shift", "dayOfWeek": 1, "type": "PREFERRED", "createdAt": "2024-01-01T00:00:00" } ] }
```

5.3 개별 선호도 수정

```
PUT /api/v1/stores/{storeId}/members/{memberId}/preferences/{preferenceId}
```

Request Body:

```
{ "type": "PREFERRED" }
```

6. 자동 스케줄링 (Auto Scheduling)

6.1 주간 스케줄 자동 생성

```
POST /api/v1/stores/{storeId}/schedules/auto-generate
```

Request Body:

```
{ "weekStartDate": "2024-10-21", "considerPreferences": true }
```

Response:

```
{ "weekStartDate": "2024-10-21", "weekEndDate": "2024-10-27", "totalAssignments": 42, "assignments": [ { "id": 1, "memberId": 1, "memberName": "김직원", "shiftTemplateId": 1, "shiftName": "Opening Shift", "workDate": "2024-10-21", "startTime": "06:00", "endTime": "11:00" } ], "conflicts": [] }
```

6.2 스케줄 미리보기 (Draft)

```
POST /api/v1/stores/{storeId}/schedules/preview
```

Request Body:

```
{ "weekStartDate": "2024-10-21", "considerPreferences": true }
```

Response: (동일한 형식이지만 실제 저장되지 않음)

6.3 스케줄 확정

```
POST /api/v1/stores/{storeId}/schedules/{scheduleId}/confirm
```

Response:

```
{ "message": "Schedule confirmed successfully", "scheduleId": 1, "notificationsSent": 10 }
```

7. 스케줄 조회 및 관리 (Schedule Management)

7.1 주간 스케줄 조회

```
GET /api/v1/stores/{storeId}/schedules
```

Query Parameters:

- `weekStartDate`: 주간 시작일 (YYYY-MM-DD) - required
- `memberId` (optional): 특정 직원 필터

Response:

```
{ "weekStartDate": "2024-10-21", "weekEndDate": "2024-10-27", "assignments": [ { "id": 1, "memberId": 1, "memberName": "김직원", "department": "HALL", "shiftTemplateId": 1, "shiftName": "Opening Shift", "workDate": "2024-10-21", "startTime": "06:00", "endTime": "11:00", "duration": 5.0, "status": "ASSIGNED" } ], "summary": { "totalHours": 42.0, "totalAssignments": 8, "memberStats": [ { "memberId": 1, "memberName": "김직원", "totalHours": 35.0, "shiftsCount": 7 } ] } }
```

7.2 개별 배정 수정

```
PUT /api/v1/stores/{storeId}/schedules/assignments/{assignmentId}
```

Request Body:

```
{ "updatedStartTime": "2024-10-21T08:00:00", "updatedEndTime": "2024
-10-21T12:00:00" }
```

7.3 배정 삭제

```
DELETE /api/v1/stores/{storeId}/schedules/assignments/{assignmentId}
```

7.4 수동 배정 추가

```
POST /api/v1/stores/{storeId}/schedules/assignments
```

Request Body:

```
{ "memberId": 1, "shiftTemplateId": 1, "workDate": "2024-10-24", "st
artTime": "08:00", "endTime": "16:00" }
```

---

8. 대타 요청 (Substitution Requests)

8.1 대타 요청 생성

```
POST /api/v1/stores/{storeId}/substitute-requests
```

Request Body:

```
{ "assignmentId": 1, "reason": "Feeling unwell, need cover asap." }
```

Response:

```
{ "id": 1, "originalAssignmentId": 1, "requesterId": 1, "requesterName": "김직원", "shiftInfo": { "shiftName": "Opening Shift", "workDate": "2024-10-24", "startTime": "08:00", "endTime": "16:00" }, "reason": "Feeling unwell, need cover asap.", "status": "PENDING", "createdAt": "2024-10-20T10:00:00" }
```

8.2 대타 요청 목록 조회 (Marketplace)

```
GET /api/v1/stores/{storeId}/substitute-requests
```

Query Parameters:

- `status` (optional): PENDING, APPLIED, APPROVED, REJECTED
- `startDate` (optional): 필터링 시작일
- `endDate` (optional): 필터링 종료일

Response:

```
{ "content": [ { "id": 1, "requesterId": 1, "requesterName": "Sarah Jenkins", "department": "CASHIER", "shiftInfo": { "shiftName": "Evening Shift", "workDate": "2024-10-29", "startTime": "16:00", "endTime": "21:00", "duration": 5.0 }, "reason": "Family emergency", "status": "PENDING", "applicationCount": 0, "createdAt": "2024-10-20T10:00:00" } ] }
```

8.3 대타 지원

```
POST /api/v1/stores/{storeId}/substitute-requests/{requestId}/apply
```

Response:

```
{ "applicationId": 1, "requestId": 1, "applicantId": 2, "applicantName": "이대타", "status": "PENDING", "createdAt": "2024-10-20T11:00:00" }
```

8.4 대타 지원 목록 조회

```
GET /api/v1/stores/{storeId}/substitute-requests/{requestId}/applications
```

Response:

```
{ "requestId": 1, "applications": [ { "applicationId": 1, "applicantId": 2, "applicantName": "이대타", "department": "HALL", "status": "PENDING", "createdAt": "2024-10-20T11:00:00" } ] }
```

8.5 대타 지원 승인/거절 (관리자)

```
PUT /api/v1/stores/{storeId}/substitute-requests/{requestId}/applications/{applicationId}
```

Request Body:

```
{ "action": "APPROVE" }
```

- action: APPROVE, REJECT

Response:

```
{ "message": "Application approved successfully", "requestId": 1, "applicationId": 1, "newAssignment": { "id": 42, "memberId": 2, "memberName": "이대타", "workDate": "2024-10-29", "shiftName": "Evening Shift" } }
```

8.6 내 대타 요청 목록

```
GET /api/v1/stores/{storeId}/members/me/substitute-requests
```

Response:

```
{ "myRequests": [ { "id": 1, "shiftInfo": { "workDate": "2024-10-29", "shiftName": "Evening Shift", "startTime": "16:00", "endTime": "21:00" }, "reason": "Family emergency", "status": "PENDING", "applicationCount": 2, "createdAt": "2024-10-20T10:00:00" } ] }
```

---

9. 출퇴근 관리 (Attendance)

9.1 출근 기록

```
POST /api/v1/stores/{storeId}/attendance/clock-in
```

Request Body:

```
{ "pinCode": "1234", "assignmentId": 1 }
```

Response:

```
{ "attendanceId": 1, "memberId": 1, "memberName": "김직원", "assignmentId": 1, "clockInTime": "2024-10-24T08:55:00", "scheduledStartTime": "09:00", "status": "EARLY" }
```

9.2 퇴근 기록

```
POST /api/v1/stores/{storeId}/attendance/clock-out
```

Request Body:

```
{ "pinCode": "1234", "attendanceId": 1 }
```

Response:

```
{ "attendanceId": 1, "memberId": 1, "memberName": "김직원", "clockInTime": "2024-10-24T08:55:00", "clockOutTime": "2024-10-24T17:05:00", "scheduledEndTime": "17:00", "totalHours": 8.17, "scheduledHours": 8.0, "overtime": 0.17, "status": "COMPLETED" }
```

9.3 일일 출퇴근 현황

```
GET /api/v1/stores/{storeId}/attendance/daily
```

Query Parameters:
- `date` : 조회 날짜 (YYYY-MM-DD) - required

Response:

```
{ "date": "2024-10-24", "storeId": 1, "attendances": [ { "attendanceId": 1, "memberId": 1, "memberName": "김직원", "department": "HALL", "assignmentId": 1, "shiftName": "Opening Shift", "scheduledStartTime": "09:00", "scheduledEndTime": "17:00", "clockInTime": "2024-10-24T08:55:00", "clockOutTime": "2024-10-24T17:05:00", "status": "COMPLETED", "totalHours": 8.17, "overtime": 0.17 }, { "attendanceId": 2, "memberId": 2, "memberName": "이직원", "department": "KITCHEN", "assignmentId": 2, "shiftName": "Peak Shift", "scheduledStartTime": "11:00", "scheduledEndTime": "15:00", "clockInTime": "2024-10-24T11:02:00", "clockOutTime": null, "status": "IN_PROGRESS", "totalHours": null, "overtime": null } ], "summary": { "totalScheduled": 5, "clockedIn": 3, "clockedOut": 1, "absent": 1 } }
```

9.4 출퇴근 기록 조회 (기간별)

```
GET /api/v1/stores/{storeId}/attendance
```

Query Parameters:

- `startDate` : 시작일 (YYYY-MM-DD) - required
- `endDate` : 종료일 (YYYY-MM-DD) - required
- `memberId` (optional): 특정 직원 필터

Response:

```
{ "startDate": "2024-10-21", "endDate": "2024-10-27", "attendances": [ { "attendanceId": 1, "memberId": 1, "memberName": "김직원", "workDate": "2024-10-21", "shiftName": "Opening Shift", "clockInTime": "2024-10-21T08:55:00", "clockOutTime": "2024-10-21T17:05:00", "scheduledHours": 8.0, "actualHours": 8.17, "overtime": 0.17, "status": "COMPLETED" } ], "summary": { "totalScheduledHours": 160.0, "totalActualHours": 165.5, "totalOvertime": 5.5, "attendanceRate": 95.5 } }
```

9.5 출퇴근 수정 (관리자)

```
PUT /api/v1/stores/{storeId}/attendance/{attendanceId}
```

Request Body:

```
{ "clockInTime": "2024-10-24T09:00:00", "clockOutTime": "2024-10-24T17:00:00", "note": "Adjusted by manager" }
```

Response:

```
{ "attendanceId": 1, "memberId": 1, "clockInTime": "2024-10-24T09:00:00", "clockOutTime": "2024-10-24T17:00:00", "actualHours": 8.0, "updatedBy": 1, "updatedAt": "2024-10-24T18:00:00", "note": "Adjusted by manager" }
```

9.6 내 출퇴근 기록 조회

```
GET /api/v1/stores/{storeId}/members/me/attendance
```

Query Parameters:

- `startDate` (optional): 시작일 (YYYY-MM-DD)
- `endDate` (optional): 종료일 (YYYY-MM-DD)
- `month` (optional): 월별 조회 (YYYY-MM)

Response:

```
{ "memberId": 1, "memberName": "김직원", "period": { "startDate": "2024-10-01", "endDate": "2024-10-31" }, "attendances": [ { "attendanceId": 1, "workDate": "2024-10-01", "shiftName": "Opening Shift", "clockInTime": "2024-10-01T08:55:00", "clockOutTime": "2024-10-01T17:05:00", "actualHours": 8.17, "status": "COMPLETED" } ], "summary": { "totalDays": 20, "totalHours": 160.5, "totalOvertime": 5.5, "averageHoursPerDay": 8.025 } }
```

10. 급여 계산 (Payroll - 2차 고려)

10.1 월별 급여 계산

```
GET /api/v1/stores/{storeId}/payroll
```

Query Parameters:

- `year` : 년도 (YYYY) - required
- `month` : 월 (MM) - required
- `memberId` (optional): 특정 직원 필터

Response:

```
{ "period": "2024-10", "payrolls": [ { "memberId": 1, "memberName": "김직원", "hourlyWage": 15000, "regularHours": 160.0, "overtimeHours": 5.5, "totalHours": 165.5, "regularPay": 2400000, "overtimePay": 110000, "totalPay": 2510000, "deductions": 0, "netPay": 2510000 } ], "storeTotalPay": 10500000 }
```

10.2 개별 급여 명세서

```
GET /api/v1/stores/{storeId}/members/{memberId}/payroll/{year}/{month}
```

Response:

```
{ "memberId": 1, "memberName": "김직원", "period": "2024-10", "hourlyWage": 15000, "workDays": 20, "regularHours": 160.0, "overtimeHours": 5.5, "totalHours": 165.5, "regularPay": 2400000, "overtimePay": 110000, "grossPay": 2510000, "deductions": { "tax": 0, "insurance": 0, "other": 0, "total": 0 }, "netPay": 2510000, "attendanceDetails": [ { "date": "2024-10-01", "hours": 8.17, "pay": 122550 } ] }
```

11. 통계 및 리포트 (Statistics & Reports)

11.1 매장 대시보드

```
GET /api/v1/stores/{storeId}/dashboard
```

Query Parameters:

- `period` (optional): WEEK, MONTH, YEAR (default: WEEK)

Response:

```
{ "storeId": 1, "storeName": "Downtown Coffee Co.", "period": "WEEK", "dateRange": { "startDate": "2024-10-21", "endDate": "2024-10-27" }, "staffing": { "totalMembers": 15, "activeMembers": 14, "averageHoursPerMember": 35.5 }, "attendance": { "totalScheduledShifts": 70, "completedShifts": 68, "absentShifts": 2, "attendanceRate": 97.14 }, "labor": { "totalScheduledHours": 560.0, "totalActualHours": 572.5, "totalOvertimeHours": 12.5, "estimatedPayrollCost": 8587500 }, "substitutions": { "pendingRequests": 3, "approvedRequests": 5, "rejectedRequests": 1 } }
```

11.2 직원별 근무 통계

```
GET /api/v1/stores/{storeId}/members/{memberId}/stats
```

Query Parameters:

- `startDate` : 시작일 (YYYY-MM-DD) - required
- `endDate` : 종료일 (YYYY-MM-DD) - required

Response:

```
{ "memberId": 1, "memberName": "김직원", "period": { "startDate": "2024-10-01", "endDate": "2024-10-31" }, "workStats": { "totalDays": 20, "totalHours": 160.5, "averageHoursPerDay": 8.025, "overtimeHours": 5.5, "attendanceRate": 95.2 }, "shiftDistribution": [ { "shiftName": "Opening Shift", "count": 10, "totalHours": 80.0 }, { "shiftName": "Peak Shift", "count": 10, "totalHours": 80.5 } ], "substitutions": { "requestedCount": 2, "coveredCount": 3 }, "earnings": { "estimatedPay": 2407500 } }
```

11.3 시간대별 인력 분석

```
GET /api/v1/stores/{storeId}/analytics/hourly-staffing
```

Query Parameters:

- `date` : 분석 날짜 (YYYY-MM-DD) - required

Response:

```
{ "date": "2024-10-24", "hourlyData": [ { "hour": 9, "scheduledStaff": 3, "actualStaff": 3, "departments": { "HALL": 2, "KITCHEN": 1 } }, { "hour": 12, "scheduledStaff": 6, "actualStaff": 5, "departments": { "HALL": 3, "KITCHEN": 2 } } ], "peakHours": [12, 13, 18, 19], "understaffedHours": [13] }
```

12. 알림 (Notifications - 2차 고려)

12.1 알림 목록 조회

```
GET /api/v1/notifications
```

Query Parameters:

- `isRead` (optional): true/false
- `type` (optional): SCHEDULE, SUBSTITUTE, ATTENDANCE, SYSTEM

Response:

```
{ "content": [ { "id": 1, "type": "SCHEDULE", "title": "새로운 주간 스케줄이 배정되었습니다", "message": "2024년 10월 21일 주간 스케줄을 확인해주세요", "isRead": false, "createdAt": "2024-10-20T10:00:00", "relatedData": { "weekStartDate": "2024-10-21" } }, { "id": 2, "type": "SUBSTITUTE", "title": "대타 요청이 승인되었습니다", "message": "10월 29일 저녁 근무 대타가 승인되었습니다", "isRead": false, "createdAt": "2024-10-20T15:30:00", "relatedData": { "requestId": 1, "workDate": "2024-10-29" } } ], "unreadCount": 5 }
```

12.2 알림 읽음 처리

```
PUT /api/v1/notifications/{notificationId}/read
```

Response:

```
{ "id": 1, "isRead": true, "readAt": "2024-10-20T16:00:00" }
```

12.3 전체 알림 읽음 처리

```
PUT /api/v1/notifications/read-all
```

Response:

```
{ "message": "All notifications marked as read", "count": 5 }
```

공통 응답 형식

성공 응답

```
{ "success": true, "data": { ... }, "message": "Success message" }
```

에러 응답