

# Casual Use Case Specification: UC1- Hire Trade Worker

**Use Case Name:** Hire Trade Worker

**Brief Description: This Use Case enables Manger and Co-Owner add new employees’ account to the system when the company hires new employees.**

# Section 1: Business Rule(s):

BR 01: Only Manager or Co-Owner can create and delete new employees’ account

BR 02: Each Staff must field a name in the employee account

BR 03: Each Staff must field at least one valid phone number

BR 04: The system must create unique employee ID

# Section 2: Scenarios (HD):

**Scenario 1: Add new employee account**

# Preconditions:

When the use case begins, the system is displaying the admin menu and admin must log in to the system.

| **Step#** | **Actor (****StoreManger or Co-Owner)** | **System** | **Data Used** |
| --- | --- | --- | --- |
| 1 | Chooses to add a new employee account | Displays adminPassword prompt and requests adminPassword |  |
| 2 | Enters adminPassword | Accepts enter | adminPassword |
| 3 | Chooses to continue | Validates adminPassword,  displays a new employeeID and temporary employeePassword, and  displays employee information form, retrieve, and display the list of provinces | provinceList  employeeID  employeePassword |
| 4 | Enters employee’s information | Accepts enter | employeeFirstName  employeeLastName  streetAddress  city  province  homePhoneNumber  cellPhoneNumber |
| 5 | Chooses to confirm | Validates employee name and phone numbers, saves the employee’s information, displays confirm message, and offers to add another new employee account |  |
| 6 | Repeats above 3 steps until user chooses to exit |  |  |
| 7 | Choose to exit | Display the main menu |  |

**Successful Post-Conditions: Added a new employee account to the Database.**

# Casual Use Case Specification: UC2

**Use Case Name: Schedule Project Details**

**Brief Description: This Use Case enables StoreManger and Co-Owner to add new schedules at least before a day, remove schedules which are already added, update schedules which are already added, and post the schedule to employees.**

# Section 1: Business Rule(s):

BR 01: Only StoreManeger or Co-Owner can create, update, delete, and post schedules

BR 02: StoreManeger or Co-Owner can only add schedules of date which do not have a schedule.

BR 03: StoreManeger or Co-Owner can only update schedules of date which already have a schedule.

BR 04: StoreManeger or Co-Owner must add new schedules at least before a day

BR 05: A staff cannot be added in the same part-time.

# Section 2: Scenarios (HD):

**Scenario 1: Add a new schedule**

# Preconditions:

When the use case begins, the system is displaying the admin menu and admin must log in to the system.

| **Step#** | **Actor (StoreManger or Co-Owner)** | **System** | **Data Used** |
| --- | --- | --- | --- |
| 1 | Chooses to add a new schedule | Displays adminPassword prompt and requests adminPassword |  |
| 2 | Enters adminPassword | Accepts enter | adminPassword |
| 3 | Choose to continue | Validates adminPassword,  retrieves and displays the list of scheduleDates which do not exist a schedule and is before current days | schdueDateList |
| 4 | Choose schdueDate | Retrieves and displays the list of partTimes | ScheduleDate  partTimesList |
| 5 | Choose partTime | Displays all employees’ full name, and requests to select employee’s full name | partTime |
| 6 | Select employees name | Accepts selections, and validates that the employee was not added into the same partTime | employeeFitstName  employeeLastName |
| 7 | Chooses to confirm | Adds employees into the schedule, displays |  |
| 8 | Chooses to save | Adds the schedule, displays confirm message, and offers to add another new schedule |  |
| 9 | Chooses to exit | Displays the main menu |  |

**Successful Post-Conditions: Added a new employee schedule to the Database**

**Scenario 2: Update a schedule**

# Preconditions

When the use case begins, the system is displaying the admin menu and admin must log in to the system.

| **Step#** | **Actor (StoreManger or Co-Owner)** | **System** | **Data Used** |
| --- | --- | --- | --- |
| 1 | Chooses to update schedules | Displays adminPassword prompt and requests adminPassword |  |
| 2 | Enters adminPassword | Accepts enter | adminPassword |
| 3 | Chooses to continue | Validates adminPassword,  retrieves and displays the list of scheduleDates which exist a schedule | schdueDatesList |
| 4 | Chooses to schdueDate | Retrieves and displays the list of partTimes | ScheduleDate  partTimesList |
| 5 | Chooses to partTime | Displays employees’ full name within the schedule | partTime |
| 6 | Chooses to add | Displays employees’ full name who are not in the schedule |  |
| 7 | Selects employee name | Accepts selection, and validates that the employee was not added into the same partTime | employeeFitstName  employeeLastName |
| 8 | Repeats above 1 step until user chooses to confirm |  |  |
| 9 | Chooses to confirm | Adds employees into the schedule, and displays employees’ full name within the schedule |  |
| 10 | Choose to remove | Displays employees’ full name within the schedule |  |
| 11 | Select employees name | Accepts selections |  |
| 12 | Repeats above 1 step until user chooses to confirm |  |  |
| 13 | Choose to confirm | Removes employees into the schedule, and displays employees’ full name within the schedule |  |
| 14 | Choose to save | Updates the schedule, displays confirm message, and offers to update other schedules |  |
| 15 | Choose to exit | Displays the main menu |  |

**Successful Post-Conditions: the schedule in the database are modified and up to date**