**Authentication APIs (Public Access)**

**1. Register User (Staff)**

* **Method:** POST
* **URL:** /api/v1/auth/register
* **Description:** Creates a new staff user account.
* **Roles:** Public
* **Example Body:**

JSON

{

"username": "newstaff03",

"password": "password123",

"fullName": "Nhân Viên Ba",

"email": "staff03@example.com",

"phone": "0905123123"

}

**2. Login User**

* **Method:** POST
* **URL:** /api/v1/auth/login
* **Description:** Authenticates a user and returns a JWT token.
* **Roles:** Public
* **Example Body:**

JSON

{

"username": "manager01",

"password": "password123"

}

**Category APIs (Requires Authentication)**

**3. Create Category**

* **Method:** POST
* **URL:** /api/v1/categories
* **Description:** Creates a new product category.
* **Roles:** MANAGER, ADMIN
* **Example Body:**

JSON

{

"name": "Specialty Coffee",

"description": "Unique single-origin coffees"

}

**4. Get All Categories**

* **Method:** GET
* **URL:** /api/v1/categories
* **Description:** Retrieves a list of all categories.
* **Roles:** STAFF, MANAGER, ADMIN
* **Example Body:** None

**5. Update Category**

* **Method:** PUT
* **URL:** /api/v1/categories/{id} (e.g., /api/v1/categories/1)
* **Description:** Updates an existing category.
* **Roles:** MANAGER, ADMIN
* **Example Body:**

JSON

{

"name": "Cà Phê Việt Nam",

"description": "Cập nhật mô tả"

}

**6. Delete Category**

* **Method:** DELETE
* **URL:** /api/v1/categories/{id} (e.g., /api/v1/categories/10)
* **Description:** Deletes a category (fails if products exist in it).
* **Roles:** MANAGER, ADMIN
* **Example Body:** None

**Product APIs (Requires Authentication)**

**7. Create Product**

* **Method:** POST
* **URL:** /api/v1/products
* **Description:** Creates a new product.
* **Roles:** MANAGER, ADMIN
* **Example Body:** (Make sure categoryId exists)

JSON

{

"name": "Trà Sen Vàng",

"code": "TRA-SENVANG",

"price": 52000.00,

"cost": 18000.00,

"description": "Trà sen kết hợp kem cheese",

"categoryId": 4

}

**8. Get All Products (Paginated)**

* **Method:** GET
* **URL:** /api/v1/products
* **Description:** Retrieves a paginated list of products.
* **Roles:** STAFF, MANAGER, ADMIN
* **Example URL Parameters:** /api/v1/products?page=0&size=5&sort=price,desc
* **Example Body:** None

**9. Get Product by ID**

* **Method:** GET
* **URL:** /api/v1/products/{id} (e.g., /api/v1/products/1)
* **Description:** Retrieves details of a specific product.
* **Roles:** STAFF, MANAGER, ADMIN
* **Example Body:** None

**10. Update Product**

* **Method:** PUT
* **URL:** /api/v1/products/{id} (e.g., /api/v1/products/1)
* **Description:** Updates an existing product (doesn't update availability).
* **Roles:** MANAGER, ADMIN
* **Example Body:** (Make sure categoryId exists)

JSON

{

"name": "Đen Đá Phin (Update)",

"code": "VN-DENDA-NEW",

"price": 26000.00,

"cost": 8500.00,

"description": "Cập nhật mô tả mới.",

"categoryId": 1

}

**11. Toggle Product Availability**

* **Method:** PATCH
* **URL:** /api/v1/products/{id}/toggle-availability (e.g., /api/v1/products/1)
* **Description:** Toggles the available status of a product.
* **Roles:** MANAGER, ADMIN
* **Example Body:** None

**12. Delete Product**

* **Method:** DELETE
* **URL:** /api/v1/products/{id} (e.g., /api/v1/products/50)
* **Description:** Deletes a product.
* **Roles:** MANAGER, ADMIN
* **Example Body:** None

**Product Recipe APIs (Requires Authentication)**

**13. Get Product Recipe**

* **Method:** GET
* **URL:** /api/v1/products/{productId}/recipe (e.g., /api/v1/products/2/recipe)
* **Description:** Gets the ingredient list (recipe) for a product.
* **Roles:** MANAGER, ADMIN
* **Example Body:** None

**14. Set/Update Product Recipe**

* **Method:** PUT
* **URL:** /api/v1/products/{productId}/recipe (e.g., /api/v1/products/2/recipe)
* **Description:** Replaces the entire recipe for a product.
* **Roles:** MANAGER, ADMIN
* **Example Body:** (Make sure ingredientIds exist)

JSON

{

"ingredients": [

{

"ingredientId": 3, // Bột Cà Phê Phin Blend

"quantityNeeded": 0.028 // 28g

},

{

"ingredientId": 4, // Sữa Đặc Ông Thọ Lon

"quantityNeeded": 0.03 // 30ml (Giả định 1 lon = 1 lít?)

},

{

"ingredientId": 9, // Đường Cát Trắng

"quantityNeeded": 0.005 // 5g

}

]

}

**Table APIs (Requires Authentication)**

**15. Get All Tables**

* **Method:** GET
* **URL:** /api/v1/tables
* **Description:** Retrieves a list of all tables and their status.
* **Roles:** STAFF, MANAGER, ADMIN
* **Example Body:** None

**16. Get Table by ID**

* **Method:** GET
* **URL:** /api/v1/tables/{id} (e.g., /api/v1/tables/1)
* **Description:** Retrieves details of a specific table.
* **Roles:** STAFF, MANAGER, ADMIN
* **Example Body:** None

**17. Create Table**

* **Method:** POST
* **URL:** /api/v1/tables
* **Description:** Creates a new table (default status is EMPTY).
* **Roles:** MANAGER, ADMIN
* **Example Body:**

JSON

{

"name": "Sân Thượng VIP",

"capacity": 8

}

**18. Update Table Info (Name, Capacity)**

* **Method:** PUT
* **URL:** /api/v1/tables/{id} (e.g., /api/v1/tables/1)
* **Description:** Updates the name and capacity of a table.
* **Roles:** MANAGER, ADMIN
* **Example Body:**

JSON

{

"name": "A1 (Gần cửa sổ)",

"capacity": 3

}

**19. Update Table Status**

* **Method:** PATCH
* **URL:** /api/v1/tables/{id}/status (e.g., /api/v1/tables/1/status)
* **Description:** Updates the status of a table (EMPTY, SERVING, RESERVED).
* **Roles:** STAFF, MANAGER, ADMIN
* **Example Body:**

JSON

{

"status": "SERVING"

}

**20. Delete Table**

* **Method:** DELETE
* **URL:** /api/v1/tables/{id} (e.g., /api/v1/tables/24)
* **Description:** Deletes a table.
* **Roles:** MANAGER, ADMIN
* **Example Body:** None

**Order APIs (Requires Authentication)**

**21. Create Order**

* **Method:** POST
* **URL:** /api/v1/orders
* **Description:** Creates a new order (status PENDING). Updates table status if AT\_TABLE.
* **Roles:** STAFF, MANAGER, ADMIN
* **Example Body (At Table):** (Make sure tableId=1 is EMPTY, productIds exist)

JSON

{

"tableId": 1,

"type": "AT\_TABLE",

"items": [

{ "productId": 11, "quantity": 1, "notes": "Ít đá"},

{ "productId": 39, "quantity": 1, "notes": null }

]

}

* **Example Body (Take Away):**

JSON

{

"tableId": null,

"type": "TAKE\_AWAY",

"items": [

{ "productId": 18, "quantity": 2, "notes": "Ly lớn"}

]

}

**22. Get All Orders (Paginated)**

* **Method:** GET
* **URL:** /api/v1/orders
* **Description:** Retrieves a paginated list of all orders.
* **Roles:** MANAGER, ADMIN
* **Example URL Parameters:** /api/v1/orders?page=0&size=10&sort=createdAt,desc
* **Example Body:** None

**23. Get Order by ID**

* **Method:** GET
* **URL:** /api/v1/orders/{id} (e.g., /api/v1/orders/1)
* **Description:** Retrieves details of a specific order, including its items.
* **Roles:** STAFF, MANAGER, ADMIN
* **Example Body:** None

**24. Get Pending Order by Table ID**

* **Method:** GET
* **URL:** /api/v1/orders/table/{tableId}/pending (e.g., /api/v1/orders/table/3/pending)
* **Description:** Finds the currently open (PENDING) order for a specific table.
* **Roles:** STAFF, MANAGER, ADMIN
* **Example Body:** None

**25. Add Item to Order**

* **Method:** POST
* **URL:** /api/v1/orders/{orderId}/items (e.g., /api/v1/orders/1/items)
* **Description:** Adds a new item or increases the quantity of an existing item in a PENDING order. Recalculates totals.
* **Roles:** STAFF, MANAGER, ADMIN
* **Example Body:** (Make sure productId exists)

JSON

{

"productId": 46,

"quantity": 2,

"notes": "Thêm 2 trân châu"

}

**26. Update Item in Order**

* **Method:** PUT
* **URL:** /api/v1/orders/{orderId}/items/{orderDetailId} (e.g., /api/v1/orders/1/items/1)
* **Description:** Updates the quantity and notes of an item in a PENDING order. Recalculates totals.
* **Roles:** STAFF, MANAGER, ADMIN
* **Example Body:**

JSON

{

"quantity": 3,

"notes": "Update ghi chú"

}

**27. Remove Item from Order**

* **Method:** DELETE
* **URL:** /api/v1/orders/{orderId}/items/{orderDetailId} (e.g., /api/v1/orders/1/items/2)
* **Description:** Removes an item from a PENDING order. Recalculates totals.
* **Roles:** STAFF, MANAGER, ADMIN
* **Example Body:** None

**28. Pay Order**

* **Method:** POST
* **URL:** /api/v1/orders/{orderId}/pay (e.g., /api/v1/orders/1/pay)
* **Description:** Marks a PENDING order as PAID, records payment method and time. Sets table to EMPTY if applicable.
* **Roles:** STAFF, MANAGER, ADMIN
* **Example Body:**

JSON

{

"paymentMethod": "CASH"

}

(Other options: "TRANSFER", "CARD")

**29. Cancel Order**

* **Method:** POST
* **URL:** /api/v1/orders/{orderId}/cancel (e.g., /api/v1/orders/4/cancel)
* **Description:** Marks a PENDING order as CANCELLED. Sets table to EMPTY if applicable.
* **Roles:** MANAGER, ADMIN
* **Example Body:** None

**Supplier APIs (Requires Authentication)**

**30. Create Supplier**

* **Method:** POST
* **URL:** /api/v1/suppliers
* **Description:** Creates a new supplier.
* **Roles:** MANAGER, ADMIN
* **Example Body:**

JSON

{

"name": "Nhà Cung Cấp Trà Phúc Long",

"contactPerson": "Anh Phúc",

"phone": "0987654321",

"email": "phuclong@example.com",

"address": "Quận 7, TP. HCM"

}

**31. Get All Suppliers**

* **Method:** GET
* **URL:** /api/v1/suppliers
* **Description:** Retrieves a list of all suppliers.
* **Roles:** MANAGER, ADMIN
* **Example Body:** None

**32. Get Supplier by ID**

* **Method:** GET
* **URL:** /api/v1/suppliers/{id} (e.g., /api/v1/suppliers/1)
* **Description:** Retrieves details of a specific supplier.
* **Roles:** MANAGER, ADMIN
* **Example Body:** None

**33. Update Supplier**

* **Method:** PUT
* **URL:** /api/v1/suppliers/{id} (e.g., /api/v1/suppliers/1)
* **Description:** Updates an existing supplier.
* **Roles:** MANAGER, ADMIN
* **Example Body:**

JSON

{

"name": "Nông trại Cà phê Ban Mê (Updated)",

"contactPerson": "Anh Tuấn Mới",

"phone": "0911111119",

"email": "banme.coffee.new@email.com",

"address": "Cập nhật địa chỉ"

}

**34. Delete Supplier**

* **Method:** DELETE
* **URL:** /api/v1/suppliers/{id} (e.g., /api/v1/suppliers/15)
* **Description:** Deletes a supplier.
* **Roles:** ADMIN
* **Example Body:** None

**Ingredient APIs (Requires Authentication)**

**35. Create Ingredient**

* **Method:** POST
* **URL:** /api/v1/ingredients
* **Description:** Creates a new ingredient (initial quantity is 0).
* **Roles:** MANAGER, ADMIN
* **Example Body:**

JSON

{

"name": "Hạt Chia Úc",

"unit": "kg",

"reorderLevel": 0.5

}

**36. Get All Ingredients (Paginated, Searchable)**

* **Method:** GET
* **URL:** /api/v1/ingredients
* **Description:** Retrieves a paginated list of ingredients, optionally filtered by name.
* **Roles:** MANAGER, ADMIN
* **Example URL Parameters:**
  + /api/v1/ingredients?page=0&size=10&sort=name,asc
  + /api/v1/ingredients?name=Sữa (Search for "Sữa")
* **Example Body:** None

**37. Get Ingredient by ID**

* **Method:** GET
* **URL:** /api/v1/ingredients/{id} (e.g., /api/v1/ingredients/1)
* **Description:** Retrieves details of a specific ingredient.
* **Roles:** MANAGER, ADMIN
* **Example Body:** None

**38. Update Ingredient Info**

* **Method:** PUT
* **URL:** /api/v1/ingredients/{id} (e.g., /api/v1/ingredients/1)
* **Description:** Updates the name, unit, and reorder level (does not update quantity).
* **Roles:** MANAGER, ADMIN
* **Example Body:**

JSON

{

"name": "Hạt Robusta Honey Rang Vừa",

"unit": "kg",

"reorderLevel": 6.0

}

**39. Delete Ingredient**

* **Method:** DELETE
* **URL:** /api/v1/ingredients/{id} (e.g., /api/v1/ingredients/32)
* **Description:** Deletes an ingredient.
* **Roles:** ADMIN
* **Example Body:** None

**Purchase Order APIs (Requires Authentication)**

**40. Create Purchase Order**

* **Method:** POST
* **URL:** /api/v1/purchase-orders
* **Description:** Creates a new purchase order (status PENDING).
* **Roles:** MANAGER, ADMIN
* **Example Body:** (Make sure supplierId and ingredientIds exist)

JSON

{

"supplierId": 2,

"items": [

{

"ingredientId": 5,

"quantity": 30,

"unitPrice": 24000

},

{

"ingredientId": 6,

"quantity": 10,

"unitPrice": 48000

}

]

}

**41. Get All Purchase Orders (Paginated)**

* **Method:** GET
* **URL:** /api/v1/purchase-orders
* **Description:** Retrieves a paginated list of purchase orders.
* **Roles:** MANAGER, ADMIN
* **Example URL Parameters:** /api/v1/purchase-orders?page=0&size=5&sort=orderDate,desc
* **Example Body:** None

**42. Get Purchase Order by ID**

* **Method:** GET
* **URL:** /api/v1/purchase-orders/{id} (e.g., /api/v1/purchase-orders/1)
* **Description:** Retrieves details of a specific purchase order.
* **Roles:** MANAGER, ADMIN
* **Example Body:** None

**43. Mark Purchase Order as Completed**

* **Method:** POST
* **URL:** /api/v1/purchase-orders/{id}/complete (e.g., /api/v1/purchase-orders/4/complete)
* **Description:** Marks a PENDING purchase order as COMPLETED and **updates ingredient stock**.
* **Roles:** MANAGER, ADMIN
* **Example Body:** None

**44. Cancel Purchase Order**

* **Method:** POST
* **URL:** /api/v1/purchase-orders/{id}/cancel (e.g., /api/v1/purchase-orders/5/cancel)
* **Description:** Marks a PENDING purchase order as CANCELLED. Does not affect stock.
* **Roles:** MANAGER, ADMIN
* **Example Body:** None