UNIVERSITY WASTE MANAGEMENT SYSTEM

OVERVIEW:

The University Waste Management System is a system which manages the waste across the university. This system handles:

1) Inventory tracking
2) Dustbin monitoring
3) Waste collection logging
4) Worker management
5) Maintenance requests
6) Work scheduling
7) Leave approvals
8) Salary processing

ENTITIES:

The database consists of the following entities:

1) University
2) Campus
3) Building
4) Inventory
5) Dustbin
6) Worker
7) Waste collection
8) Maintenance request
9) Work schedule
10) Leave request
11) Salary payment

1) UNIVERSITY

It stores the university's information.

Name	Data type	Constraints	Description
University_id	INT	PK	Unique university identifier
Name	VARCHAR	NOT NULL	University name
Main_campus_location	VARCHAR	NOT NULL	Main campus location
Contact_email	VARCHAR	UNIQUE	Official contact email
Created_at	DATETIME	Current timestamp	Record creation time

- The primary key is the university_id.
- The unique index is on contact_email to avoid duplicates.

2) CAMPUS

It stores all the campuses under the university.

Name	Data Type	Constraints	Description
Campus_id	INT	PK	Unique campus ID
University_id	INT	FK University(university_id)	Parent university
Campus_name	VARCHAR	NOT NULL	Campus name
Address	VARCHAR	NOT NULL	Campus address
Contact_number	VARCHAR		Campus contact
Created_at	Datetime	Current timestamp	Creation timestamp

- University_id is the foreign key here. This improves campus search.

3) **BUILDING:**

It stores the buildings within a campus.

Name	Type	Constraints	Description
Building_id	INT	PK	Unique building ID
Campus_id	INT	FK Campus(campus_id)	Parent campus
Building_name	VARCHAR	NOT NULL	Building name
Building_type	VARCHAR		Type of building
Created_at	Datetime	Current timestamp	Created timestamp

- Index on campus_id.

4) **INVENTORY:**

It manages the cleaning supplies and the equipment needed.

Name	Type	Constraints	Description
Inventory_id	INT	PK	Inventory ID
Campus_id	INT	FK Campus(campus_id)	Campus location
Item_name	VARCHAR	NOT NULL	Item name
Item_category	VARCHAR		Category
quantity	INT	CHECK (quantity >= 0)	Available quantity
Recorder_level	INT	CHECK (>= 0)	Minimum stock level
Unit_price	DECIMAL	CHECK (>= 0)	Price per unit
Supplier_name	VARCHAR		supplier
Last_updated	DATE		Last update
Created_at	DATETIME	Current timestamp	Timestamp

- Inventory_id is the primary key here.
- Index on campus_id.
- Index on item_name for frequent searches.

5) **DUSTBIN:**

It tracks the dustbins in the building.

Name	Type	Constraints
Dustbin_id	INT	PK
Building_id	INT	FK Building(building_id)
Bin_type	VARCHAR	CHECK (‘wet’, ‘dry’, etc)
capacity	INT	CHECK (>=0)
Current_fill_level	INT	CHECK (>=0)
Last_emptied_date	DATE	
status	VARCHAR	CHECK (‘empty’, ‘full’, etc)
Created_at	DATETIME	

- Index on building_id.

6) **WORKER**

It stores the information of the employee.

Name	Type	Constraints
Worker_id	INT	PK
Campus_id	INT	FK Campus(campus_id)
name	VARCHAR	NOT NULL
CNIC	VARCHAR	UNIQUE
phone	VARCHAR	UNIQUE
Hire_date	DATE	NOT NULL
salary	DECIMAL	CHECK (>=0)
Worker_type	VARCHAR	CHECK (‘Cleaner’, ‘Supervisor’, etc)
Employment_status	VARCHAR	CHECK (‘active’, ‘resigned’, etc)
Created_at	DATETIME	Current timestamp

- Index on campus_id.
- Unique index on CNIC.

- Unique index on phone.

7) WASTE COLLECTION

It controls the waste collection activities.

Name	Type	Constraints
Collection_id	INT	PK
Dustbin_id	INT	FK Dustbin(dustbin_id)
Worker_id	INT	FK Worker(worker_id)
Collection_date	DATE	NOT NULL
Collection_time	TIME	NOT NULL
Waste_weight	DECIMAL	CHECK (>=0)
remarks	VARCHAR	
Created_at	DATETIME	Current timestamp

- Composite index on dustbin_id, collection_date.
- Index on worker_id.

8) MAINTENANCE REQUEST

It tracks dustbin maintenance issues.

Name	Type	Constraints
Maintenance_id	INT	PK
Dustbin_id	INT	FK Dustbin(dustbin_id)
Worker_id	INT	FK Worker(worker_id)
Maintenance_date	DATE	
Issue_description	VARCHAR	NOT NULL
status	VARCHAR	CHECK (‘pending’, ‘resolved’)
Created_at	DATETIME	Current timestamp

9) WORK SCHEDULE

It manages work scheduling.

Name	Type	Constraints

Schedule_id	INT	PK
Worker_id	INT	FK Worker(worker_id)
Building_id	INT	FK Building(building_id)
Shift_date	DATE	NOT NULL
Shift_start_date	TIME	NOT NULL
Shift_end_date	TIME	NOT NULL
Created_at	DATETIME	Current timestamp

- Composite index on worker_id, shift_date.

10)LEAVE REQUEST

It manages the worker leave applications.

Name	Type	Constraints
Leave_id	INT	PK
Worker_id	INT	FK Worker(worker_id)
Approved_by	INT	FK Worker(worker_id)
Leave_start_date	DATE	NOT NULL
Leave_end_date	DATE	NOT NULL
Leave_type	VARCHAR	CHECK (‘sick’, ‘casual’, etc)
status	VARCHAR	CHECK (‘pending’, ‘approved’, etc)
reason	VARCHAR	
Requested_at	DATETIME	Current timestamp
Actioned_at	DATETIME	

- Index on worker_id.
- Index on status.

11)SALARY PAYMENT

It records the salary disbursements.

Name	Type	Constraints
Payment_id	INT	PK
Worker_id	INT	FK Worker(worker_id)
month	INT	CHECK (between 1 and 12)
year	INT	CHECK if valid
Basic_salary	DECIMAL	
bonus	DECIMAL	DEFAULT 0
deductions	DECIMAL	DEFAULT 0
Net_salary	DECIMAL	
Payment_date	DATE	
Payment_status	VARCHAR	CHECK (‘paid’, ‘pending’, ‘failed’)
Created_at	DATETIME	Current timestamp

- Composite index on worker_id, month, year.

USER ROLE DEFINITIONS:

1) **System administrator:** Full access is given to the administrator. He manages campuses, workers and the inventory. He can also approve or disapproves leave requests. Also the process of salaries is controlled by the administrator.
2) **Supervisor:** He views and manages assigned workers. He also approves the maintenance requests and monitors dustbins. He has access to the inventory.
3) **Worker:** He views the assigned schedules. He can also submit the leave request. He is assigned to collect the waste. Also, he can view his salary records.

INDEX SPECIFICATIONS WITH RATIONALE:

Table	Index	Reason
Campus	University_id	Frequent lookup by university
Building	Campus_id	Filters buildings per campus
Worker	Campus_id	Worker per campus
Waste Collection	Dustbin_id, collection_date	Reporting of waste
Salary Payment	Worker_id, month, year	To lookup monthly salary
Leave Request	status	Filtering pending approvals

Indexes are added on frequently searched or used foreign keys to approve query performance.

CONSTRAINTS SUMMARY:

- There are primary keys in all tables.
- Foreign keys in every table ensures referential integrity.
- Unique constraints on CNIC, phone, email.
- CHECK constraints on status fields, salary (>= 0), month 1-12, valid values.