University of British Columbia, Vancouver
Department of Computer Science

CPSC 304 Project Cover Page

Milestone #: ___4____

Date: __November 28, 2025_____

Group Number: ____68_____

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Kaitlin Gill	28374149	g0u9b	gkaitlin@student.ubc.ca
Natalia Blanco	16174898	a5z0k	talia22@student.ubc.ca
Kieran Freitag	19405778	t9j0l	kfreitag@kieran.ca

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

University of British Columbia, Vancouver
Department of Computer Science

Repository Link : https://github.students.cs.ubc.ca/CPSC304-2025W-T1/team_68.git

SQL Initialization Script:
https://github.students.cs.ubc.ca/CPSC304-2025W-T1/team_68/blob/main/database_initialization.sql (database_initialization.sql in root of team_68 folder)

University of British Columbia, Vancouver
Department of Computer Science

10 Queries:

1. Insert - appService.js (line 127)
INSERT INTO GARDEN (garden_id, name, postal_code, street_name, house_number, owner_id)
VALUES (:garden_id, :name, :postal_code, :street_name, :house_number, :owner_id)

This query inserts a set of attributes into the Garden table. It ensures that the owner_id and postal_code already exist (references the Person table and Postal Code table respectively). If the attributes referencing the Location table (postal_code, street_name, house_number) do not exist, these attributes are also inserted together into the Location before the Garden insert.

2. Update - appService.js (line 234)
UPDATE PLANT SET is_ready = :0, section_id = :1 WHERE plant_id = :2
[1, 3, '3']
*Note query above contains example input

This query allows the user to update any number of non-primary key attributes in the Plant table.

3. Delete - appService.js (line 507)
DELETE FROM TOOLTYPE WHERE name = :toolTypeName
(toolTypeName is set via dropdown menu)

This query allows the user to select a tool type and delete it from the database (removes from relations Tool type and Tool).

4. Selection - appService.js (line 200)
SELECT * FROM PLANT WHERE plant_id = :0 AND LOWER(type_name) = :1 OR is_ready = :2
['2', 'tomato', '0']
*Note query above contains example input

This query takes selected attributes, logic (AND/OR) and input values to retrieve from the Plant table. The output is a copy of the Plant table with the chosen characteristics. Conditions are based on equality.

5. Projection - appService.js (line 332)

6. Join - appService.js (line 468)
SELECT p.plant_id, p.latitude, p.longitude, p.radius, p.is_ready, p.section_id, pt.name as type_name, pt.requirements, pt.description
FROM Plant p
JOIN PlantType pt ON p.type_name = pt.name

```
WHERE pt.name = :plantTypeName
ORDER BY p.plant_id`
```

7. Aggregation with GROUP BY - appService.js (line 407)

```
SELECT type_name, count(*) as plant_count
FROM PLANT
GROUP BY type_name
ORDER BY type_name
```

This query groups all plants from all gardens by their plant type and counts how many plants belong to each type. It returns one row per plant type, in alphabetical order.

8. Aggregation with HAVING - appService.js (line 441)

```
SELECT s.section_id, s.garden_id, g.name as garden_name, SUM(w.volume_litres) as
total_water
FROM Section s
JOIN Garden g ON s.garden_id = g.garden_id
JOIN Water w ON s.section_id = w.section_id
GROUP BY s.section_id, s.garden_id, g.name
HAVING SUM(w.volume_litres) > 50
ORDER BY total_water DESC`
```

This query joins the Section, Garden, and Water tables to calculate the total amount of water used each by each section. It returns the sections whose total usage of water is greater than 50 litres, sorted from highest to lowest water consumption.

9. Nested Aggregation with GROUP BY - appService.js (line 407)

```
SELECT s.section_id, s.garden_id, g.name as garden_name, COUNT(DISTINCT p.type_name) as
diversity
FROM Section s
JOIN Garden g ON s.garden_id = g.garden_id
JOIN Plant p ON s.section_id = p.section_id
GROUP BY s.section_id, s.garden_id, g.name
HAVING COUNT(DISTINCT p.type_name) > (SELECT AVG(diversity_count)
FROM (SELECT COUNT(DISTINCT p2.type_name) as diversity_count
FROM Plant p2
GROUP BY p2.section_id)
)
ORDER BY diversity DESC, s.section_id
```

This query finds all sections whose plant-type diversity is higher than average across all sections. It counts the distinct plant types per section and returns only sections (and their garden name and id) whose diversity is above the average.

10. Division - appService.js (line 375)

```
SELECT s.section_id, s.garden_id, g.name as garden_name
FROM Section s
JOIN Garden g ON s.garden_id = g.garden_id
WHERE NOT EXISTS (SELECT pt.name
                  FROM PlantType pt
                  WHERE NOT EXISTS (SELECT p.plant_id
                                    FROM Plant p
                                    WHERE p.section_id = s.section_id
                                    AND p.type_name = pt.name)
                   )
ORDER BY s.section_id
```

This query finds all sections which contain every plant type in the database. It returns the section along with its garden name and id only if there is no plant type missing from the section.

Description and Schema:

Our project is a smart-garden monitoring system that allows for a garden to be broken up into sections, each of which may be allocated resources, plants, and tools as required. Each section has a monitoring system which tracks various environmental endpoints and resource usage, as well as maintains a maintenance log. Every garden has an owner and others may be granted access to it.

This is represented in a web application built on top of an Oracle database schema that allows users to interact with several connected tables that represent real smart-gardening information. Users can view all the tables and run meaningful queries that show details like gardens with high water consumption, find gardens with higher plant-type diversity, and more. Users can also interact to insert into our garden table, update our plant table, and delete from our tools table.

University of British Columbia, Vancouver
Department of Computer Science

Summary of Changes from Previous Schema:

1) Our original location table (country, city, postal_code, street_address) was broken into 2 separate tables: Location which contains postal_code, house_number, street_name and PostalCode which contains postal_code, province, and city. This change was made to ensure our relations were all in BCNF.

2) Our original sections table(id, latitude, longitude, length, width) was broken into 2 separate tables: SectionDimensions (length, width, area) and Sections (section_id, garden_id, latitude, longitude, length, width). This change was made to ensure our relations were all in BCNF.

3) Garden table: The Garden table now contains a location foreign key (postal_code, house_number, street_name) (it also includes a mandatory owner_id foreign key to Person). This ensures that every garden has both a valid location and an owner, preventing any orphaned gardens from existing in our database.

4) Tool table: The Tool table now includes type_name as a foreign key to our separate ToolType table (which stores name and function), instead of having tool type information directly as attributes. This was a normalization done in a previous milestone so that we don't duplicate tool information in our database.

5) Section table: The Section table now stores length and width as attributes, but these also are a foreign key to the SectionDimensions (see number 2 on this list for that change) table (which stores the area for each length and width combination). This ensures our dimensions are valid and our relations are all in BCNF.

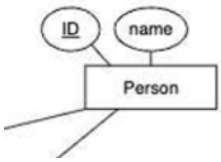
6) Plant table: a column for plant_type has been added to be a foreign constraint to the plant_type table. This ensures every plant has a type with their description stored in the planttype table. This reduces any duplication of information in our database.

7) Maintenance Log table: this was previously a weak entity with only timestamp and entry as attributes. In our implementation, it is not a weak entity, (maintenance_log_id as primary key) making it a strong entity instead, with section_id as just a foreign key to connect each log to a garden section.

University of British Columbia, Vancouver
Department of Computer Science

A copy of the original schema and screenshots that show what data is present in each relation after the SQL initialization script is run:

1) **Person**:

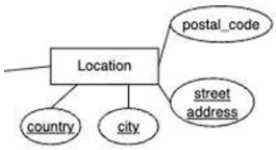


Person(ID, name)

Person

Person ID	Name
1	Sarah Johnson
2	Michael Chen
3	Emily Rodriguez
4	David Thompson
5	Jessica Williams
6	Robert Martinez
7	Amanda Lee

2) PostalCode



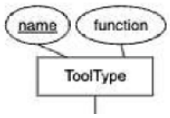
PostalCode(<u>postal_code</u>, province, city)

PostalCode

Postal Code	Province	City
V6T1Z4	British Columbia	Vancouver
V5K0A1	British Columbia	Vancouver
V7M2E3	British Columbia	North Vancouver
V3H4K6	British Columbia	Burnaby
V6B1A1	British Columbia	Vancouver
V4N3M2	British Columbia	Richmond

- This table was modified from the original schema (see above for explanation).

3) ToolType



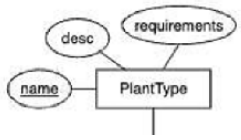
ToolType(<u>name</u>, function)

ToolType

Name	Function
Shovel	Digging into soil
Rake	Gathering leaves or leveling soil
Pruning Shears	Trimming and shaping plants
Watering Can	Manual watering of plants
Hoe	Breaking up soil
Trowel	Small-scale digging and planting
Garden Fork	Turning and aerating compost

University of British Columbia, Vancouver
Department of Computer Science

4) PlantType

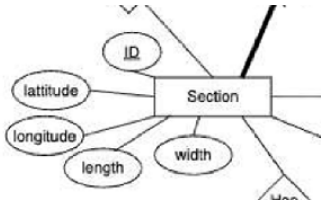


PlantType(<u>name</u>, requirements, description)

PlantType

Name	Requirements	Descriptions
Tomato	Full sun, regular watering, pH 6.0-6.8	make into sauces for pasta
Lettuce	Partial shade, consistent moisture, pH 6.0-7.0	used for salads
Carrot	Full sun, loose soil, pH 6.0-6.8	root vegetable good for soups
Basil	Full sun, warm temperatures, pH 6.0-7.0	herb plant
Cucumber	Full sun, consistent watering, pH 6.0-7.0	great veggie to dip in hummus
Bell Pepper	Full sun, warm soil, pH 6.0-6.8	not very spicy pepper, comes in red, green, orange, and yellow
Strawberry	Full sun, well-drained soil, pH 5.5-6.5	yummy berry good in cakes

5) SectionDimensions



SectionDimensions(<u>length, width</u>, area)

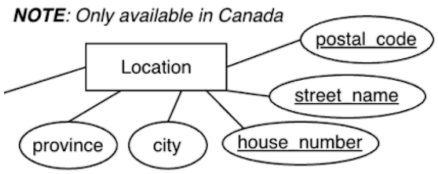
SectionDimensions

Length	Width	Area
10.500	9.300	97.650
5.500	12.920	71.060
7.300	8.120	59.276
10.900	11.400	124.260
12.500	11.700	146.250

- This table was modified from the original schema (see above for explanation).

University of British Columbia, Vancouver
Department of Computer Science

6) Location



Location (<u>postal_code</u>, <u>house_number</u>, <u>street_name</u>)

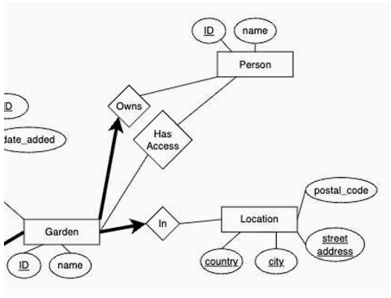
Location

Postal Code	House Number	Street Name
V3H4K6	8900	Eastlake Drive
V4N3M2	7890	Garden City Road
V5K0A1	1234	Commercial Drive
V6B1A1	456	Granville Street
V6T1Z4	2329	West Mall
V7M2E3	567	Lonsdale Avenue

- This table was modified from the original schema (see above for explanation).

University of British Columbia, Vancouver
Department of Computer Science

7) Garden



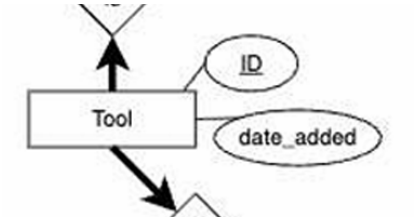
Garden(<u>garden_id,</u> name, *postal_code, street_name, house_number*, *owner_id*)

Garden

Garden ID	Name	Postal Code	Street Name	House Number	Owner ID
1	UBC Community Garden	V6T1Z4	West Mall	2329	1
2	Sunrise Urban Farm	V5K0A1	Commercial Drive	1234	2
3	North Shore Garden Haven	V7M2E3	Lonsdale Avenue	567	3
4	Burnaby Heights Garden	V3H4K6	Eastlake Drive	8900	4
5	Downtown Rooftop Garden	V6B1A1	Granville Street	456	5
6	Richmond Family Garden	V4N3M2	Garden City Road	7890	6

- This table was modified from the original schema (see above for explanation).

8) Tool



Tool(tool_id, date_added, garden_id, type_name)

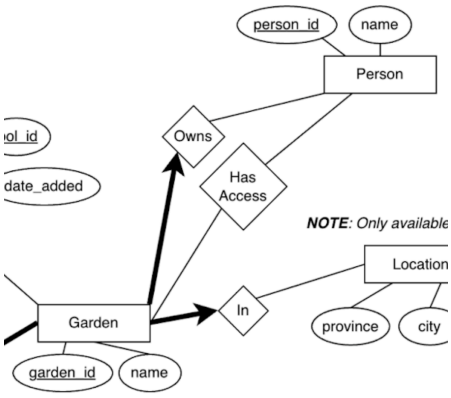
Tool

Tool ID	Date Added	Garden ID	Type Name
1	2024-03-15T07:00:00.000Z	1	Shovel
2	2024-03-15T07:00:00.000Z	1	Rake
3	2024-04-01T07:00:00.000Z	1	Watering Can
4	2024-02-10T08:00:00.000Z	2	Pruning Shears
5	2024-03-20T07:00:00.000Z	2	Hoe
6	2024-01-05T08:00:00.000Z	3	Trowel
7	2024-03-10T08:00:00.000Z	3	Garden Fork
8	2024-04-15T07:00:00.000Z	4	Shovel
9	2024-05-01T07:00:00.000Z	5	Watering Can

- This table was modified from the original schema (see above for explanation).

University of British Columbia, Vancouver
Department of Computer Science

9) HasAccess

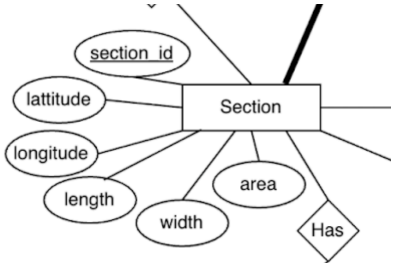


HasAccess(garden_id, person_id)

HasAccess

Garden ID	Person ID
1	1
1	2
1	7
2	2
2	3
3	3
3	4
4	4
5	5
6	6

10) **Section**



Section(<u>section_id</u>, garden_id, latitude, longitude, length, width)

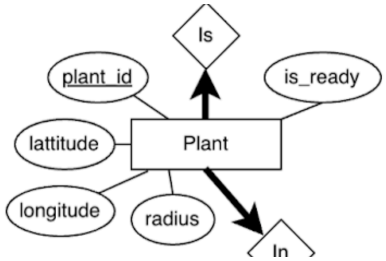
Section

Section ID	Garden ID	Latitude	Longitude	Length	Width
1	1	49.261	-123.246	10.500	9.300
2	1	49.261	-123.246	5.500	12.920
3	2	49.270	-123.069	7.300	8.120
4	2	49.270	-123.070	10.900	11.400
5	3	49.316	-123.076	12.500	11.700
6	4	49.249	-122.981	5.500	12.920
7	5	49.283	-123.121	7.300	8.120

- This table was modified from the original schema (see above for explanation).

University of British Columbia, Vancouver
Department of Computer Science

11) **Plant**



Plant(<u>plant_id</u>, latitude, longitude, radius, is_ready, type, section_id)

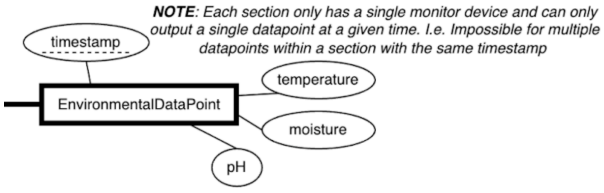
Plant

Plant ID	Latitude	Longitude	Radius	Status (1=Ready, 0=Not Ready)	Type	Section ID
1	49.261	-123.246	0.300	1	Tomato	1
2	49.261	-123.246	0.250	1	Tomato	1
3	49.261	-123.246	0.200	0	Basil	1
4	49.261	-123.246	0.150	1	Lettuce	2
5	49.261	-123.246	0.150	1	Lettuce	2
6	49.270	-123.069	0.250	0	Carrot	3
7	49.270	-123.069	0.300	1	Cucumber	3
8	49.270	-123.070	0.250	0	Bell Pepper	4
9	49.316	-123.076	0.200	1	Strawberry	5
10	49.249	-122.981	0.300	1	Tomato	6
11	49.261	-123.246	0.180	1	Lettuce	1
12	49.261	-123.246	0.220	0	Carrot	1
13	49.261	-123.246	0.280	1	Cucumber	1
14	49.261	-123.246	0.240	1	Bell Pepper	1
15	49.261	-123.246	0.190	0	Strawberry	1
16	49.261	-123.246	0.270	1	Tomato	2
17	49.261	-123.246	0.160	1	Basil	2
18	49.261	-123.246	0.210	0	Carrot	2
19	49.261	-123.246	0.290	1	Cucumber	2
20	49.261	-123.246	0.230	0	Bell Pepper	2

- This table was modified from the original schema (see above for explanation).

University of British Columbia, Vancouver
Department of Computer Science

12) **EnvironmentalDataPoint**

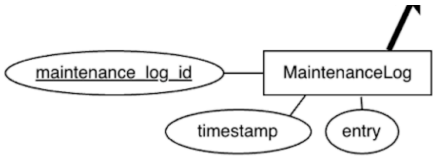


EnvironmentalDataPoint(<u>timestamp</u>, <u>section_id</u>, temperature, moisture, ph)

EnvironmentalDataPoint

Timestamp	Section ID	Temperature	Moisture	pH
2024-10-01T15:00:00.000Z	1	22.500	65	6.500
2024-10-01T21:00:00.000Z	1	25.800	58	6.400
2024-10-02T15:00:00.000Z	1	21.300	70	6.600
2024-10-01T15:00:00.000Z	2	23.100	62	6.800
2024-10-01T15:00:00.000Z	3	24	55	6.200
2024-10-01T15:00:00.000Z	4	22.800	60	6.700
2024-10-01T15:00:00.000Z	5	20.500	68	5.800

13) MaintenanceLog



MaintenanceLog(<u>maintenance_log_id</u>, timestamp, entry, section_id)

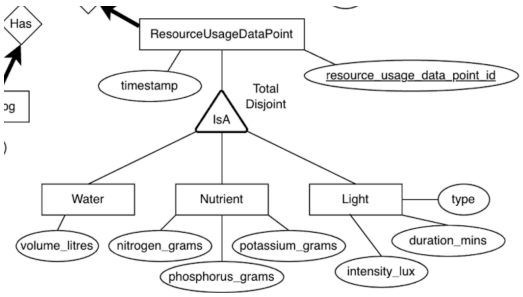
MaintenanceLog

Maintenance Log ID	Timestamp	Entry	Section ID
1	2024-09-15T17:30:00.000Z	weeded entire section and added nutrients to tomato plants	1
2	2024-09-20T21:15:00.000Z	pruned tomato plants and removed diseased leaves	1
3	2024-09-22T21:15:00.000Z	harvested lettuce heads, replanted new seedlings	2
4	2024-09-25T18:45:00.000Z	applied organic fertilizer to cucumber vines	3
5	2024-09-28T23:20:00.000Z	installed trellis system for cucumber plants	3
6	2024-10-01T15:30:00.000Z	treated bell peppers for aphid infestation using neem oil	4
7	2024-10-03T15:30:00.000Z	removed strawberry runners and composted old leaves	5

- This table was modified from the original schema (see above for explanation).

University of British Columbia, Vancouver
Department of Computer Science

14) Water



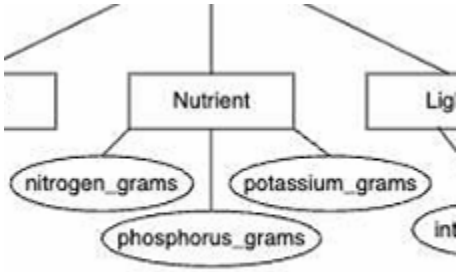
Water(volume_litres, timestamp, resource_usage_data_point_id, section_id)

Water

Resource Usage ID	Timestamp	Volume (l)	Section ID
1	2024-10-01T14:00:00.000Z	45.500	1
2	2024-10-02T02:00:00.000Z	38.200	1
3	2024-10-02T14:00:00.000Z	42	1
4	2024-10-01T14:00:00.000Z	25.500	2
5	2024-10-01T14:00:00.000Z	65	3
6	2024-10-01T14:00:00.000Z	80.500	4
7	2024-10-01T14:00:00.000Z	15	5

University of British Columbia, Vancouver
Department of Computer Science

15) **Nutrient**



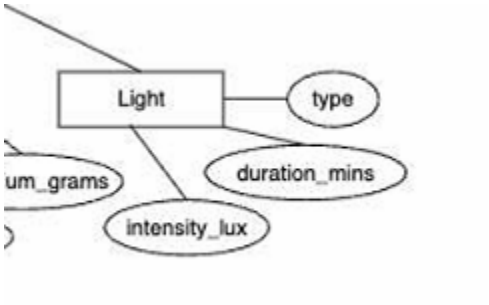
Nutrient(<u>resource_usage_data_point_id,</u> timestamp, nitrogen_grams, potassium_grams, phosphorus, grams, *section_id*)

Nutrient

Resource Usage ID	Timestamp	Nitrogen (g)	Potassium (g)	Phosphorus (g)	Section ID
1	2024-09-15T17:00:00.000Z	150			1
2	2024-09-22T17:00:00.000Z	120		70	2
3	2024-09-25T18:00:00.000Z		120	95	3
4	2024-09-28T17:00:00.000Z	200	140	110	4
5	2024-09-30T16:00:00.000Z			50	5
6	2024-10-02T17:00:00.000Z	160	110		6

University of British Columbia, Vancouver
Department of Computer Science

16) **Light**



Light(<u>resource_usage_data_point_id</u>, timestamp, intensity_lux, duration_mins, type, *section_id*)

Light

Resource Usage ID	Timestamp	Intensity (lux)	Duration (min)	Type	Section ID
1	2024-10-01T19:00:00.000Z	85000	480	Natural Sunlight	1
2	2024-10-01T19:00:00.000Z	78000	420	Natural Sunlight	2
3	2024-10-01T19:00:00.000Z	92000	510	Natural Sunlight	3
4	2024-10-02T03:00:00.000Z	15000	240	LED Grow Light	5
5	2024-10-01T19:00:00.000Z	88000	465	Natural Sunlight	6
6	2024-10-02T01:00:00.000Z	25000	360	LED Grow Light	7