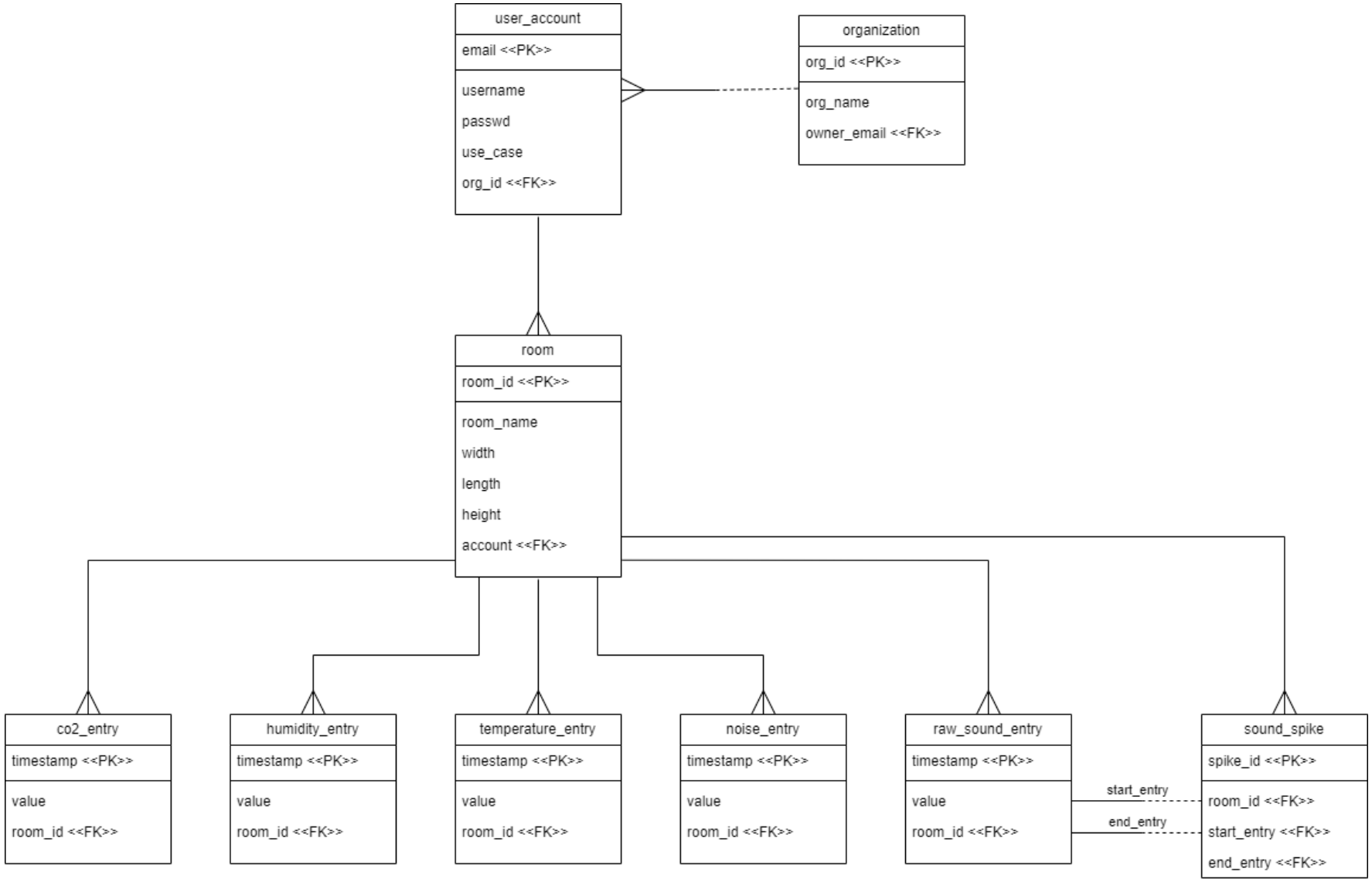
Logical Data Model



List of Tables and Fields

➔ Table: **organization**

This table stores all the organizations that hold accounts with RoomSense

- org_id : integer <<PK>>
  - The ID for the organization, incrementally generated
- org_name : text
  - The name of the organization, can be whatever the user decides
- owner_email : text <<FK>> – Links to table **user_account**
  - The email for the owner of the organization links to the email in the user_account table, all rooms of the organization are linked to this user

Example records:

	org_id	org_name	owner_email
1	1	KdG	pothoek@kdg.be

➔ Table: **user_account**

This table stores all of the user accounts that can use RoomSense, created by signing up

- email : text <<PK>>
  - The email for the user, has to be unique from any other user
- username : varchar(30)
  - The username chosen by the user, up to 30 characters
- passwd : varchar(30)
  - The password chosen by the user, up to 30 characters
- use_case : use_case_t
  - The use case for the user, defined by the object type **use_case_t**
- org_id : integer <<FK>> – Links to table **organization**
  - The *optional* organization ID for the company they are part of if not a home use_case

Example records:

	email	username	passwd	use_case	org_id
1	roman.gordon@student.kdg.be	RomanG	WorldWideRoomSense	home	<null>
2	anna.test@gmail.com	anna.test	annatest	home	<null>
3	hans.vochten@kdg.be	Hans	HansVochten	home	<null>
4	abel@gmail.com	Abel	AbelAbel	home	<null>
5	jasper.marichal@student.kdg.be	Jasper	JasperKdG	home	<null>
6	pothoek@kdg.be	KdG Pothoek	PothoekCampus	school	1

➔ Object type: **use_case_t**

- Values: ‘office’, ‘school’, ‘home’

➔ Table: **room**

This table stores all the rooms that our users have created via the add room page

- room_id : integer <<PK>>
  - The room ID, incrementally generated
- room_name : varchar(60)
  - The room name, chosen by the user up to 60 characters long
- width : real
  - The width of the room, a double set by the user
- height : real
  - The height of the room, a double set by the user
- length : real
  - The length of the room, a double set by the user
- account : text <<FK>> – Links to table **user_account**
  - The email address of the user account linked to owning this room

Example records:

	room_id	room_name	account	width	length	height
1	1	Roman Home	roman.gordon@student.kdg.…	3.7	4.85	2.8
2	2	Roman Kitchen	roman.gordon@student.kdg.…	1.6	3.75	2.3
3	3	annas room	anna.test@gmail.com	5.5	5.5	5.5
4	4	annas room	anna.test@gmail.com	5.5	5.5	5.5
5	5	Anna's kitchen	anna.test@gmail.com	6.4	2.6	2.5
6	6	Anna's cinema	anna.test@gmail.com	15	10	5.5
7	7	Boldi Home	roman.gordon@student.kdg.…	4	7	3

➔ Table: **co2_entry**

This table stores all the CO2 entries recorded by our RoomSense devices

- timestamp : timestamp with timezone <<PK>>
  - The timestamp of the entry, unique
- value : integer
  - The value of the entry
- room_id : integer <<FK>> – Links to table **room**
  - The room that the entry was measured in

Example records:

	room_id	value	timestamp
1	1	50	2023-11-14 20:58:58.962000 +00:00
2	1	51	2023-11-14 20:59:00.626000 +00:00
3	1	50	2023-11-14 20:59:02.289000 +00:00
4	1	51	2023-11-14 20:59:03.947000 +00:00
5	1	51	2023-11-14 20:59:05.610000 +00:00
6	1	51	2023-11-14 20:59:07.273000 +00:00
7	1	52	2023-11-14 20:59:08.931000 +00:00

➔ Table: **humidity_entry**

This table stores all the humidity entries recorded by our RoomSense devices

- ◆ timestamp : timestamp with timezone <<PK>>
  - The timestamp of the entry, unique
- ◆ value : integer
  - The value of the entry
- ◆ room_id : integer <<FK>> – Links to table **room**
  - The room that the entry was measured in

Example records:

	room_id	value	timestamp
1	1	48	2023-11-13 16:04:49.965000 +00:00
2	1	48	2023-11-13 16:04:51.403000 +00:00
3	1	48	2023-11-13 16:04:52.836000 +00:00
4	1	48	2023-11-13 16:04:54.274000 +00:00
5	1	48	2023-11-13 16:04:55.712000 +00:00
6	1	48	2023-11-13 16:04:57.149000 +00:00
7	1	48	2023-11-13 16:04:58.583000 +00:00

➔ Table: **temperature_entry**

This table stores all the temperature entries recorded by our RoomSense devices

- ◆ timestamp : timestamp with timezone <<PK>>
  - The timestamp of the entry, unique
- ◆ value : integer
  - The value of the entry
- ◆ room_id : integer <<FK>> – Links to table **room**
  - The room that the entry was measured in

Example records:

	room_id	value	timestamp
1	1	25	2023-11-13 16:04:49.967000 +00:00
2	1	25	2023-11-13 16:04:51.403000 +00:00
3	1	25	2023-11-13 16:04:52.840000 +00:00
4	1	25	2023-11-13 16:04:54.274000 +00:00
5	1	25	2023-11-13 16:04:55.712000 +00:00
6	1	25	2023-11-13 16:04:57.149000 +00:00
7	1	25	2023-11-13 16:04:58.583000 +00:00

➔ Table: **noise_entry**

This table stores all the noise entries recorded by our RoomSense devices

- ◆ timestamp : timestamp with timezone <<PK>>
  - The timestamp of the entry, unique
- ◆ value : integer
  - The value of the entry
- ◆ room_id : integer <<FK>> – Links to table **room**
  - The room that the entry was measured in

Example records:

	room_id	value	timestamp
1	7	197	2023-11-29 21:11:35.544116 +00:00
2	7	198	2023-11-29 21:19:14.333821 +00:00
3	7	194	2023-11-29 21:23:24.230860 +00:00
4	7	206	2023-11-29 21:23:26.646820 +00:00
5	7	133	2023-11-29 21:23:29.054160 +00:00
6	7	134	2023-11-29 21:23:31.458704 +00:00
7	7	143	2023-11-29 21:23:33.920729 +00:00

➔ Table: **raw_sound_entry**

This table stores all the raw sound entries that are linked to sound spikes recorded by our devices

- ◆ timestamp : timestamp with timezone <<PK>>
  - The timestamp of the entry, unique
- ◆ value : integer
  - The value of the entry
- ◆ room_id : integer <<FK>> – Links to table **room**
  - The room that the entry was measured in

Example records:

	room_id	value	timestamp
1	7	553	2023-11-29 20:52:36.643591 +00:00
2	7	553	2023-11-29 20:52:36.645593 +00:00
3	7	555	2023-11-29 20:52:36.645596 +00:00
4	7	555	2023-11-29 20:52:36.645599 +00:00
5	7	552	2023-11-29 20:52:36.645602 +00:00
6	7	551	2023-11-29 20:52:36.645605 +00:00
7	7	554	2023-11-29 20:52:36.645608 +00:00

➔ Table: **sound_spike**

This table stores all the sound spikes that are recorded by our RoomSense devices

- spike_id : integer <<PK>>
  - The ID of the spike, incrementally generated
- room_id : integer <<FK>> – Links to table **room**
  - The ID of the room that this spike took place in
- start_entry : timestamp with timezone – Links to table **raw_sound_entry**
  - The timestamp of the first entry that is a part of this spike
- end_entry : timestamp with timezone – Links to table **raw_sound_entry**
  - The timestamp of the last entry that is a part of this spike

Example records:

	room_id	spike_id	start_entry	end_entry
1	7	3	2023-11-29 21:25:09.843502 +00:00	2023-11-29 21:25:11.928140 +00:00
2	7	4	2023-11-29 21:25:59.173934 +00:00	2023-11-29 21:26:00.431220 +00:00
3	7	5	2023-11-29 21:26:01.753752 +00:00	2023-11-29 21:26:04.278978 +00:00
4	7	6	2023-11-29 21:26:08.872338 +00:00	2023-11-29 21:26:10.241411 +00:00
5	7	7	2023-11-29 21:48:02.931583 +00:00	2023-11-29 21:48:04.077666 +00:00
6	7	8	2023-11-29 21:48:08.748170 +00:00	2023-11-29 21:48:10.824976 +00:00
7	7	9	2023-11-29 21:48:11.158732 +00:00	2023-11-29 21:48:16.968356 +00:00