

Государственное бюджетное профессиональное образовательное учреждение
«Нижегородский Губернский колледж»

РАЗРАБОТКА СЕРВЕРНОЙ ЧАСТИ ПРИЛОЖЕНИЯ

Руководитель _____ Мамшева Ю.С.

28.10.2024

Студент _____ Астапчик Д.А.

28.10.2024

Специальность, группа: 09.02.07, 41П

Нижний Новгород
2024 г

СОДЕРЖАНИЕ

1. ПРОЕКТИРОВАНИЕ СЕРВЕРНОЙ ЧАСТИ	3
1.1 Список сущностей	4
1.2 Описание атрибутов сущностей.....	6
1.3 Список связей между сущностями	12
1.4 ER-диаграмма.....	13
1.5 Даталогическая модель базы данных “FlyV”	14
2. РАЗРАБОТКА СЕРВЕРНОЙ ЧАСТИ	15
2.1 Создание сущностей	16
2.2 Создание процедур и планировщиков заданий	20

1. ПРОЕКТИРОВАНИЕ СЕРВЕРНОЙ ЧАСТИ

На данном этапе было необходимо произвести процесс, целью которого является проектирование серверной части мобильного программного продукта. Для того, чтобы реализовать поставленную задачу, было необходимо выполнить следующий перечень задач:

1. Составить список сущностей необходимых для разработки программного обеспечения;
2. Составить описание атрибутов сущностей;
3. Составить список связей между сущностями;
4. Произвести создание ER-диаграммы “FlyV”;
5. Произвести создание даталогической модели программного продукта.

1.1 Список сущностей

На основании произведённого анализа технического задания, был выявлен следующий перечень информации, необходимый для практического функционирования мобильного программного продукта “FlyV”:

1. Должна храниться информация о сотрудниках предприятия;
2. Должна храниться информация о различных официальных причинах отсутствия сотрудника на рабочем месте;
3. Должна храниться информация о городах, в которых работают сотрудники предприятия;
4. Должна храниться информация о заменах сотрудников другими сотрудниками предприятия на различных проектах, в период отсутствия определённых работников по определённой причине;
5. Должна храниться информация о сотрудниках предприятия и причине их отсутствия на определённый период времени;
6. Должна храниться некоторая информация о модулях, что должны будут выполнены за определённый период времени;
7. Должна храниться информация о сотрудниках, что реализовывают совместно определённые модули.

Рассмотрим список необходимых сущностей (табл 1).

Таблица 1. Список сущностей

Наименование	Назначение
Сотрудники	Хранение информации о сотрудниках
Причины отсутствия	Хранение информации о всех официальных причинах отсутствия сотрудников на предприятии в рабочее время
Города	Хранение информации о городах, в которых производят работают сотрудники
Замены	Хранение информации о сотрудниках, что будет отсутствовать по определённой причине, и заменять отсутствующего сотрудника в определённый период времени
Отсутствия и сотрудники	Хранение информации о сотрудниках и их причинах отсутствия в рабочее время на предприятии
Модули	Хранение информации о модулях
Модули и сотрудники	Хранение информации о модулях и сотрудников, которые реализовывают данные модули

1.2 Описание атрибутов сущностей

На основании информации о сущностях были составлены атрибуты таблиц, выражающие определённое свойство. Рассмотрим результат произведённой деятельности (табл 2-8).

Таблица 2. Атрибуты сущности “Сотрудники”

Ключевое поле	Название	Назначение
ПК (Первичный ключ)	Идентификатор сотрудника	Ключевое поле, предназначенное для однозначной идентификации каждой записи в таблице. Представляет собой первичный ключ. Это уникальное значение, соответствующее каждому сотруднику
ВК (Внешний ключ)	Идентификатор города	Внешний ключ – это атрибут отношения, ключ) который является первичным ключом другого отношения. В данном случае это атрибут сущности города. С помощью внешнего ключа будет определено какой город относится к конкретному пользователю
	Полное имя	
	Дата устройства	
	Количество отгулов	
	Баланс отпускных дней	
	Номер телефона	
	Работает/не работает	

Таблица 3. Атрибуты сущности “Причины отсутствия”

Ключевое поле	Название	Назначение
ПК (Первичный ключ)	Идентификатор причины отсутствия	Ключевое поле, предназначенное для однозначной идентификации каждой записи в таблице. Представляет собой первичный ключ. Это уникальное значение, соответствующее каждой причине отсутствия
	Причина	

Таблица 4. Атрибуты сущности “Города”

Ключевое поле	Название	Назначение
ПК (Первичный ключ)	Идентификатор города	Ключевое поле, предназначенное для однозначной идентификации каждой записи в таблице. Представляет собой первичный ключ. Это уникальное значение, соответствующее городу, в котором работают сотрудники компании
	Город	

Таблица 5. Атрибуты сущности “Замены”

Ключевое поле	Название	Назначение
ПК (Первичный ключ)	Идентификатор замены	Ключевое поле, предназначенное для однозначной идентификации каждой записи в таблице. Представляет собой первичный ключ. Это уникальное значение, соответствующее замене
ВК (Внешний ключ)	Идентификатор первого заменяемого сотрудника	Внешний ключ – это атрибут отношения, ключ) который является первичным ключом другого отношения. В данном случае это атрибут сущности сотрудники. С помощью внешнего ключа будет определено какой сотрудник является заменяющим
ВК (Внешний ключ)	Идентификатор второго заменяемого сотрудника	Внешний ключ – это атрибут отношения, ключ) который является первичным ключом другого отношения. В данном случае это атрибут сущности сотрудники. С помощью внешнего ключа будет определено какой сотрудник является заменяемым
	Дата начала замены	
	Дата конца замены	

Таблица 6. Атрибуты сущности “Отсутствия и сотрудники”

Ключевое поле	Название	Назначение
ПК (Первичный ключ)	Идентификатор отсутствия сотрудника	Ключевое поле, предназначенное для однозначной идентификации каждой записи в таблице. Представляет собой первичный ключ. Это уникальное значение, соответствующее отсутствию сотрудника
ВК (Внешний ключ)	Идентификатор причины	Внешний ключ – это атрибут отношения, ключ) который является первичным ключом другого отношения. В данном случае это атрибут сущности причины отсутствия. С помощью внешнего ключа будет определена причина отсутствия сотрудника
ВК (Внешний ключ)	Идентификатор сотрудника	Внешний ключ – это атрибут отношения, ключ) который является первичным ключом другого отношения. В данном случае это атрибут сущности сотрудники. С помощью внешнего ключа будет определен сотрудник, который отсутствует по определённой причине
	Дата начала отсутствия	
	Количество дней отсутствия	

Таблица 7. Атрибуты сущности “Модули”

Ключевое поле	Название	Назначение
ПК (Первичный ключ)	Идентификатор модуля	Ключевое поле, предназначенное для однозначной идентификации каждой записи в таблице. Представляет собой первичный ключ. Это уникальное значение, соответствующее модулю
	Модуль	
	Дата начала выполнения	
	Дата окончания выполнения	
	Нужное количество сотрудников	

Таблица 8. Атрибуты сущности “Модули и сотрудники”

Ключевое поле	Название	Назначение
ПК (Первичный ключ)	Идентификатор модуля и сотрудника	Ключевое поле, предназначенное для однозначной идентификации каждой записи в таблице. Представляет собой первичный ключ. Это уникальное значение, соответствующее модулю и сотруднику
ВК (Внешний ключ)	Идентификатор модуля	Внешний ключ – это атрибут отношения, ключ) который является первичным ключом другого отношения. В данном случае это атрибут сущности модули. С помощью внешнего ключа будет определен модуль, над которым работает сотрудник
ВК (Внешний ключ)	Идентификатор сотрудника	Внешний ключ – это атрибут отношения, ключ) который является первичным ключом другого отношения. В данном случае это атрибут сущности сотрудники. С помощью внешнего ключа будет определен сотрудник, который работает над определённым модулем

1.3 Список связей между сущностями

На данном этапе был составлен перечень, информации, отражающий все связи между сущностями, необходимыми, для эффективной разработки программного обеспечения “FlyV” (табл 9).

Таблица 9. Список связей

№	Название связи	Сущности, участвующие в связи	Назначение
1	1: M	Города – Сотрудники	В одном городе могут проживать различные сотрудники
2	1: M	Причины отсутствия – Отсутствия и сотрудники	Различные сотрудники могут использовать одну причину отсутствия
3	1: M	Сотрудники – Отсутствия и сотрудники	Один сотрудник может использовать различные причины отсутствия
4	1:1	Сотрудники – Замены	Один сотрудник может заменить другого сотрудника
5	1:1	Сотрудники – Замены	Один сотрудник может быть заменён одним сотрудником
6	1: M	Модули – Модули и сотрудники	Над одним модулем может работать множество сотрудников
7	1: M	Сотрудники – Модули и сотрудники	Один сотрудник может работать над множеством модулей

1.4 ER-диаграмма

На данном этапе, исходя из произведённого анализа всех необходимых данных, был реализованы процессы, целью которых является разработка сущностей и установка специализированных связей (рис 1).

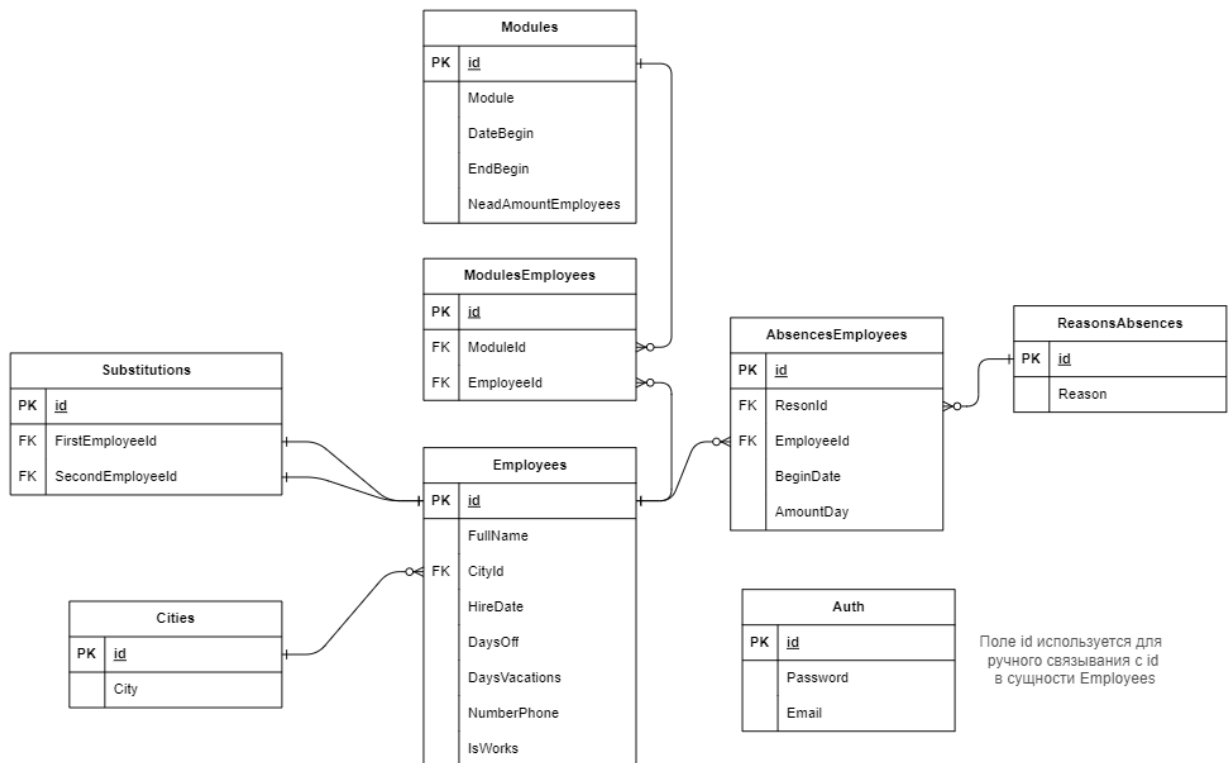


Рисунок 1 – ER-диаграмма для ИП «FlyV»

Аналогично для того, чтобы повысить уровень понимания взаимосвязей сущностей на разных уровнях, было необходимо представить сущность Auth, что является индивидуальной системой, предоставляющей различный функционал. Данная система может в себе содержать почту и зашифрованный пароль пользователя, а также предоставляет функционал авторизации пользователя. Специализированный id из сущности Auth, вручную устанавливается конкретному пользователю системы, в сущности, Employees в поле id.

1.5 Даталогическая модель базы данных “FlyV”

В последствии был произведён процесс, целью которого являлась разработка даталогической модели базы данных посредством эксплуатации средств платформы Supabase. Рассмотрим результат произведённой деятельности (рис 2).

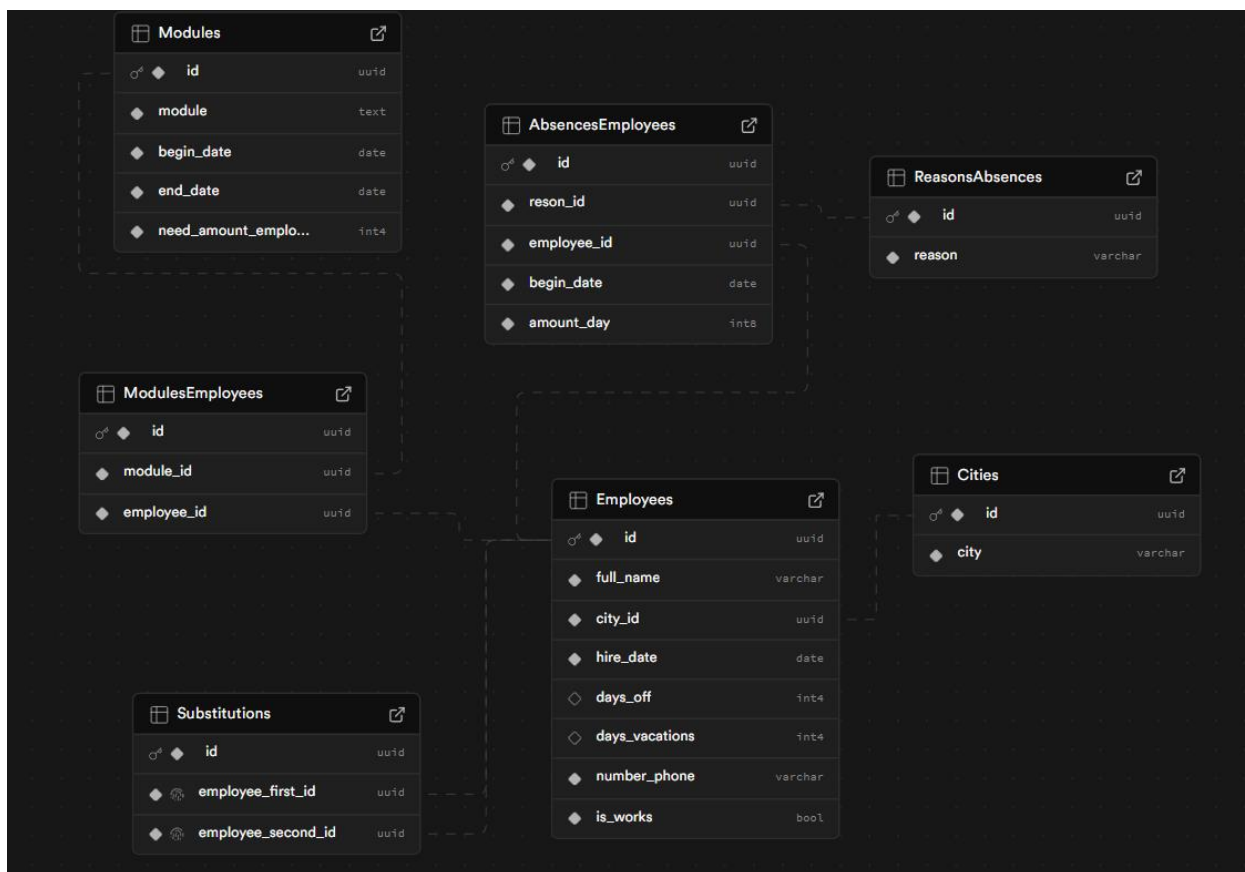


Рисунок 2 – Даталогическая модель базы данных “FlyV”

2. РАЗРАБОТКА СЕРВЕРНОЙ ЧАСТИ

На данном этапе было необходимо произвести процесс разработки серверной части, посредством использования всех необходимых средств специализированной платформы Supabase. Для того, чтобы реализовать поставленную задачу, было необходимо произвести следующий перечень заданий:

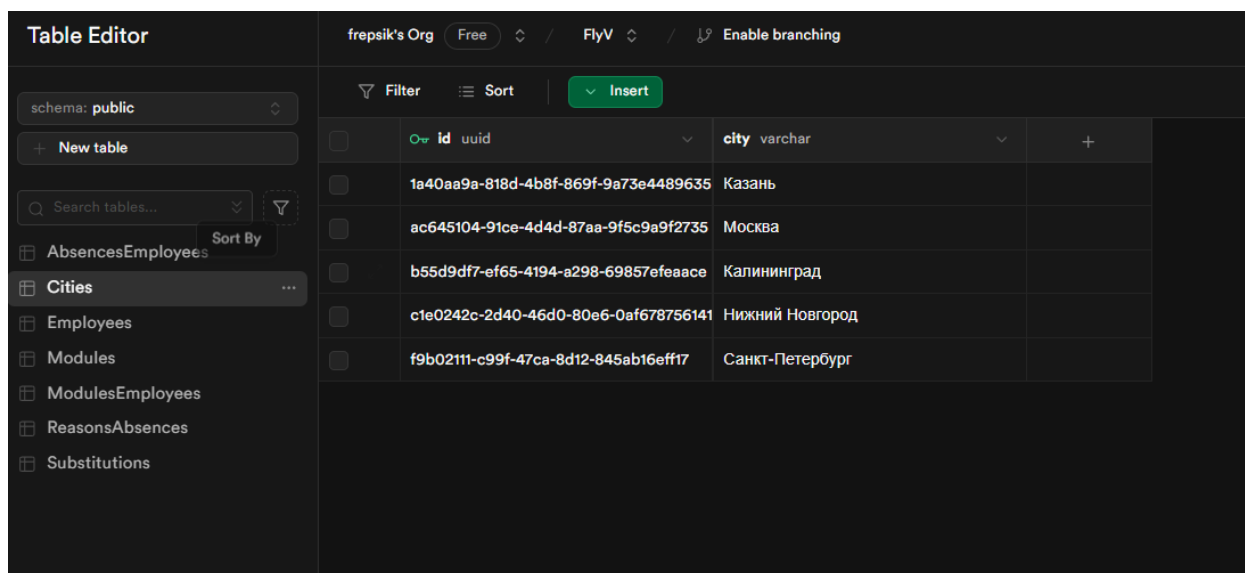
1. Разработка и создание всех сущностей, отношений между ними и заполнение данными;
2. Разработка и создание специализированных процедур и планировщиков заданий.

Рассмотрим результат произведённой деятельности.

2.1 Создание сущностей

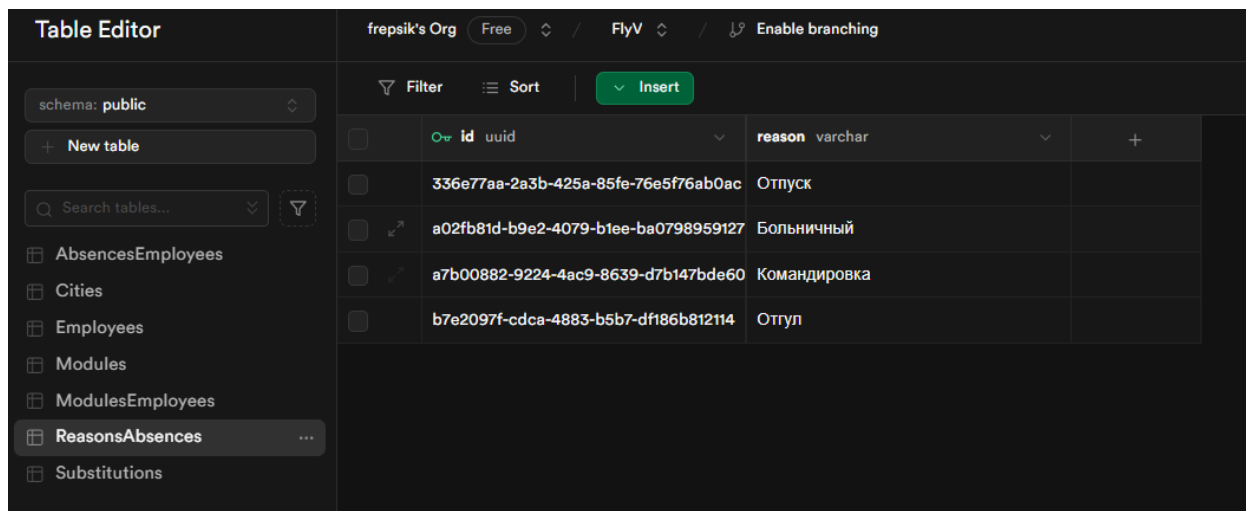
На данном этапе был произведён процесс, целью которого являлось создание всех, ранее рассмотренных сущностей, связей между ними. Аналогично был произведён процесс, что заключался в добавлении данных.

Рассмотрим результат деятельности (рис 3-9)



	id	city	
	1a40aa9a-818d-4b8f-869f-9a73e4489635	Казань	
	ac645104-91ce-4d4d-87aa-9f5c9a9f2735	Москва	
	b55d9df7-ef65-4194-a298-69857efeaace	Калининград	
	c1e0242c-2d40-46d0-80e6-0af678756141	Нижний Новгород	
	f9b02111-c99f-47ca-8d12-845ab16eff17	Санкт-Петербург	

Рисунок 3 – Результат создания сущности “Города”



	id	reason	
	336e77aa-2a3b-425a-85fe-76e5f76ab0ac	Отпуск	
	a02fb81d-b9e2-4079-b1ee-ba0798959127	Больничный	
	a7b00882-9224-4ac9-8639-d7b147bde60	Командировка	
	b7e2097f-cdca-4883-b5b7-df186b812114	Отгул	

Рисунок 4 – Результат создания сущности “Причины отсутствия”

id	module	begin_date	end_date	need_amount_employees
00eb9bff-a96e-4c4d-b6b3-4052bd52a18	Графика и визуальные эффекты	2024-10-01	2024-10-30	4
55be76da-76e2-461e-aadd-7b683a126ac6	Игровой движок и физика	2024-11-01	2024-12-01	5
79c92fd5-8f2f-4c37-950f-fd1b1afb552c	Звуки и музыка	2024-12-05	2024-12-30	2
c4345560-1cdc-4519-8b76-20224735ef20	Анимация и кинематика	2024-11-01	2024-11-18	2

Рисунок 5 – Результат создания сущности “Модули”

id	full_name	city_id	hire_date	days_off	days_vacations	number_phone	is_works
32438ab-8014-420b-822f-ed9a10e637d1	Александр Валера Андреевич	1e40aa94-818d-4b8f-869f-9a73e4...	2024-10-27	3	28	+79890052654	TRUE
369ef26a-8ca7-4bd7-b0b0-3758b48f855a	Астапкин Дмитрий Алексеевич	c1e0242c-2d40-46d0-80e6-0af67...	2022-10-10	3	28	+79200378434	TRUE
402cf10c-7c62-46f1-b6f4-187a2773c31	Солошко Даниил Алексеевич	c1e0242c-2d40-46d0-80e6-0af67...	2024-10-27	3	28	+79200374888	TRUE
547da52d-02fd-42a8-9da5-76f7ae5082f	Козар Виктор Русланович	c1e0242c-2d40-46d0-80e6-0af67...	2024-10-30	3	0	+79200734774	TRUE
9bb7606d-6e02-4ae1-a5da-7e95c86a8f8c	Жуков Сергей Валерьевич	f9b02111-c99f-47ca-8d12-845ab16...	2024-10-27	3	28	+79200568323	TRUE
b2cdea26-4794-477a-b3aa-d20598b8f471	Буланов Кирилл Александрович	c1e0242c-2d40-46d0-80e6-0af67...	2023-09-13	3	28	+79200734488	TRUE
e0a6dc16-d4d3-45e2-9ff5-6ad33ed0421a	Ефимов Дмитрий Александрович	ac645104-91ce-4d4d-87ae-9f5c9a...	2024-10-29	3	28	+793238568425	TRUE
efb54877-8421-40ff-8a49-79282caad789	Артёмов Артём Артёмович	b56d9df7-af65-4194-a298-69857e...	2024-10-27	3	28	+79200568775	TRUE

Рисунок 6 – Результат создания сущности “Сотрудники”

id	module_id	employee_id	
22ea8a95-3ecb-44a6-a8ad-80ce61ff0fc	00eb9bff-a96e-4c4d-b6b3-4052b...	b2cdea26-4794-477a-b3aa-d2059...	
30ee0314-162d-4f93-a9a3-16ca8ad1b8b1	00eb9bff-a96e-4c4d-b6b3-4052b...	402cf10c-7c62-46f1-b6f4-187a277...	
36d02ff3-4882-4b1d-a896-4200c654ae4f	00eb9bff-a96e-4c4d-b6b3-4052b...	efb54877-8421-40ff-8a49-79282c...	
3f2bf3fe-2131-4bb8-845f-8a3ef87fe26e	00eb9bff-a96e-4c4d-b6b3-4052b...	369ef26a-8ca7-4bd7-b0b0-3758b...	

Рисунок 7 – Результат создания сущности “Модули и сотрудники”

id	reason_id	employee_id	begin_date	amount_employees
4617203d-6fbf-40dd-924e-ace056a902f5	a7b00882-9224-4ac9-8639-d7b14...	9bb7606d-6e02-4ae1-a5da-7e95c...	2024-11-12	15
b5be1576-942d-4c3d-ac8f-88794b21965a	a7b00882-9224-4ac9-8639-d7b14...	e0a6dc16-d4d3-45e2-9ff5-6ad33...	2024-10-30	10

Рисунок 8 – Результат создания сущности “Отсутствия и сотрудники”

id	employee_first_id	employee_second_id
2807be1b-e2a0-49eb-bec5-475cf45be53c	402cf10c-7c62-46f1-b6f4-187a277...	32438aeb-8014-420b-822f-ed9a1c...
416b2fe1-5097-48a8-9c17-f5355d7db6e1	9bb7606d-6e02-4aet-a5da-7e95c...	b2cdea26-4794-477a-b3aa-d2059...
6b840a93-1646-4833-8621-957064f3d333	369ef26a-8ca7-4bd7-b0b0-3758b...	e0a6dc16-d4d3-45e2-9ff5-6ad35...
b127e9be-8c29-46a9-a838-cfa3fd5effe8	547da52d-02fd-42e8-9da5-75f74e...	efb54877-8421-40ff-8a49-79282c...

Рисунок 9 – Результат создания сущности “Замены”

В последствии был реализован процесс, что заключался в добавлении специализированных политик, что предоставят возможность вести определённую работу с данными на клиенте мобильного программного продукта. Рассмотрим результат произведённой деятельности (рис 10-16).

Operation	Policy Name	Applied to
DELETE	Enable delete data all users	public role
INSERT	Enable insert for authenticated users only	authenticated role
SELECT	Enable read access for all users	public role
UPDATE	Enable update data all users	public role

Рисунок 10 – Результат добавления политик для сущности “Отсутствия и сотрудники”

Operation	Policy Name	Applied to
DELETE	Enable delete data all users	public role
INSERT	Enable insert for authenticated users only	authenticated role
SELECT	Enable read access for all users	public role
UPDATE	Enable update data all users	public role

Рисунок 11 – Результат добавления политик для сущности “Сотрудники”

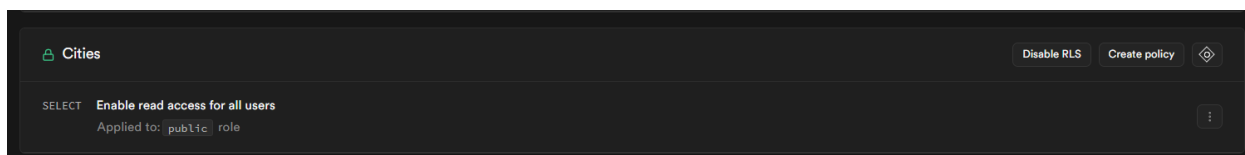


Рисунок 12 – Результат добавления политик для сущности “Города”

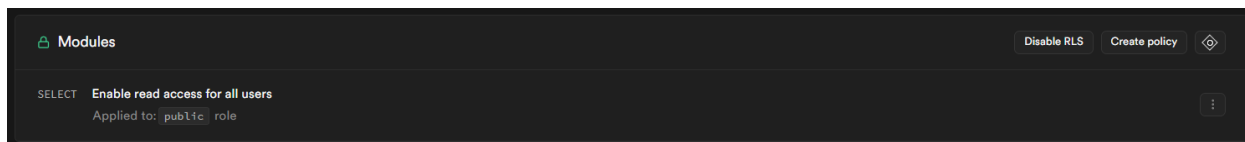


Рисунок 13 – Результат добавления политик для сущности “Модули”

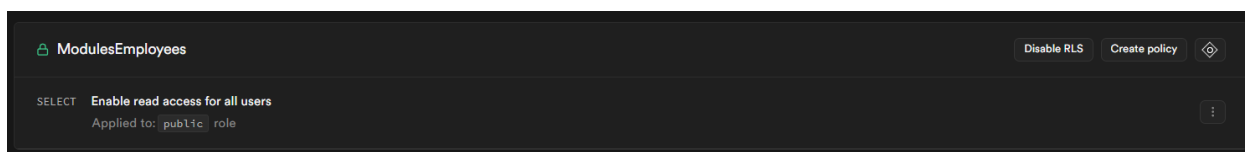


Рисунок 14 – Результат добавления политик для сущности
“Модули и сотрудники”

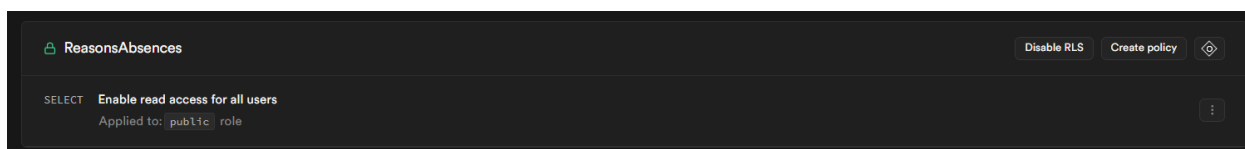


Рисунок 15 – Результат добавления политик для сущности
“Причины отсутствия”

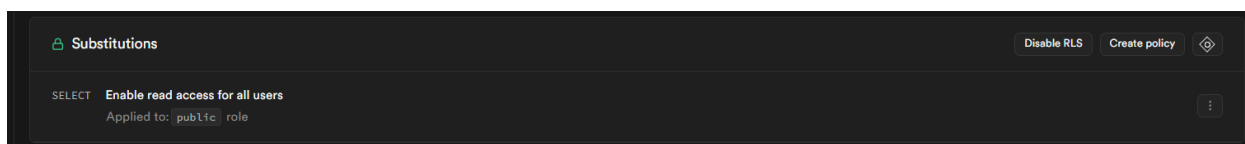


Рисунок 16 – Результат добавления политик для сущности “Замены”

В результате осуществлённой деятельности был составлена даталогическая модель базы данных, что была ранее представлена в качестве иллюстрирующего материала.

2.2 Создание процедур и планировщиков заданий

На данном этапе было необходимо реализовать задачу, что заключалась в необходимости производить автоматизированный перерасчёт баланса отпускных дней каждого сотрудника предприятия, спустя 1 рабочий год. Аналогично, было необходимо произвести алгоритм, цель которого заключалась в автоматизированном перерасчёте количества отгулов в начале рабочего месяца.

Данная задача была реализована посредством создания специализированных процедур и планировщиков заданий. Рассмотрим результат произведённой деятельности (рис 17-21).

```
1  create
2  or replace procedure UpdateDaysOffOnceAMonth()
3  language plpgsql as $$
4  begin
5      update "Employees"
6      set days_off = 3
7      where days_off < 3;
8  end;
9  $$;
```

Рисунок 17 – Процедура обновления отгулов, в начале рабочего месяца

```
11 create
12 or replace procedure UpdateDaysVacationsUserAfterOneYearWorks()
13 language plpgsql as $$
14 begin
15     update "Employees"
16     set days_vacations = 28
17     where extract(month from hire_date) = extract(month from now())
18     and extract(day from hire_date) = extract(day from now() at time zone 'Europe/Moscow')
19     and days_vacations < 28;
20 end;
21 $$;
```

Рисунок 18 – Процедура обновления баланса отпускных дней, спустя 1
рабочий год

Для того, чтобы реализовать автоматизированный запуск ранее представленных процедур, необходимо использовать специализированное расширение pg_cron – планировщик заданий. Изначально необходимо включить данное расширение посредством команды, что указана на рисунке 19.

```
41 create extension if not exists pg_cron;
```

Рисунок 19 – Команда добавления расширения pg_cron

Далее были созданы следующие планировщики (рис 20, 21), заданий, что осуществляют свой запуск по следующим правилам:

1. TaskUpdateDaysOffOnceAMonth – запуск в начале каждого месяца 1 числа;
2. TaskUpdateDaysVacationsUserAfterOneYearWorks – запуск спустя один рабочий год сотрудника.

```
24 select
25 cron.schedule (
26     'TaskUpdateDaysOffOnceAMonth',
27     '0 0 1 * *', /*в 0 минут, в 12 часов, в 1 день месяца, каждый месяц, в любой день недели*/
28     'CALL UpdateDaysOffOnceAMonth()'
29 );
```

Рисунок 20 – cron - TaskUpdateDaysOffOnceAMonth

```
32 select
33 cron.schedule (
34     'TaskUpdateDaysVacationsUserAfterOneYearWorks',
35     '0 0 * * *', /*в 0 минут, в 12 часов, каждый день месяца, каждый месяц, в любой день недели*/
36     'CALL UpdateDaysVacationsUserAfterOneYearWorks()'
37 );
```

Рисунок 21 – cron - TaskUpdateDaysVacationsUserAfterOneYearWorks