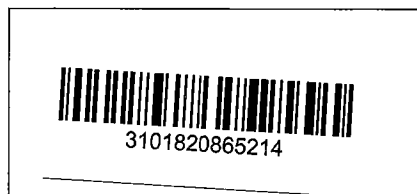




ИЗУМРУД СТУДЕНТ
ЛИ ИАДА АЛ ФЕДЕ АЛ ОГО УНИ Р ИТ



Титульный лист

Направление Естественные науки Инженерные науки
 Математика и информатика Социальные и
 Экономика и управление гуманитарные науки

Вариативный блок 1 2 3 4 5

Курс 1 2 3 4 5 отсутствует

Фамилия С О С Н О В С К И Й

Имя К О Н С Т А Н Т И Н

Отчество М А К С И М О В И Ч

Дата рождения 21 11 2003

Город участия Е К А Т Е Р И Н Б У Р Г

Аудитория Д 3

Дата 02 02 2026

Подпись *Клог*

Пример заполнения
 А Б В Г Д Е Ж З И Й К Л М Н О П Р С Т У Ф
 Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я 1 2 3 4 5 6 7 8 9 0



Будем хранить данные о показаниях датчиков в реляционной базе данных

- 1) Будем хранить полную дату и время сбора показаний датчиков, тип DATETIME
- 2) Пронумеруем все зоны размещения и будем хранить её номер INT
- 3) Будем записывать тип датчика VARCHAR и все показатели (температура, влажность, освещенность, влажность почвы) будем хранить в INT формате (т.к. все значения целые)

Пример таблицы DATA

Дата	Время	Номер зоны	Тип	тем-ра	влажк.	освещ	влажк. почвы
02.02.2026	10:00	1	"type1"	20	50	4000	40

Индивидуальные требования из условий задачи будем хранить в отдельной таблице

- 1) Название растения VARCHAR
 - Остальное также INT
- (Будет выглядеть как в условии задачи)

Алгоритм принятия решений по коррекции микроклимата:

Имея индивидуальные требования, мы можем выполнить предварительный расчет показателей, которые удовлетворяют всем растениям одновременно. Все удовлетворит тем-ра 22-24, влажк-ть 70, освещенность для освещенности. Нет такого значения.

Наш алгоритм будет стремиться поддерживать температуру и влажность, которая удовлетворяет всем, а освещенность применим индивидуальной пошкой для каждого.

ЛИНИЯ ОТСЧЕТА

С помощью запроса в таблицу по адресу получаем текущее показание, далее начинаем работать с температурой

Для температуры предварительно в константы запишем значения, которые удовлетворяют все растения (MIN_NORM_TEMP=22, MAX_NORM_TEMP=24)

```

if cur_temperature > MAX_NORM_TEMP
while cur_temperature > MAX_NORM_TEMP
conditioner work() # включаем кондиционер на охлаждение
cur_temperature = get_actual_temperature()
# обновим текущую температуру, чтоб не гнать много
# змрии на охлаждение
conditioner stop()
elif cur_temperature < MIN_NORM_TEMP
while cur_temperature < MIN_NORM_TEMP
conditioner work_on_hot() # включаем на обогрев
cur_temperature = get_actual_temperature()
conditioner stop()

```

Однако блок с ~~тем~~ температурой будем делать вторым, так как влажность почвы для нас наиболее приоритетный показатель, для почвы норма для всех $\geq 60\%$, поэтому будем делать одну проверку для всех

```

if cur_water_in < 60
do_pov()

```

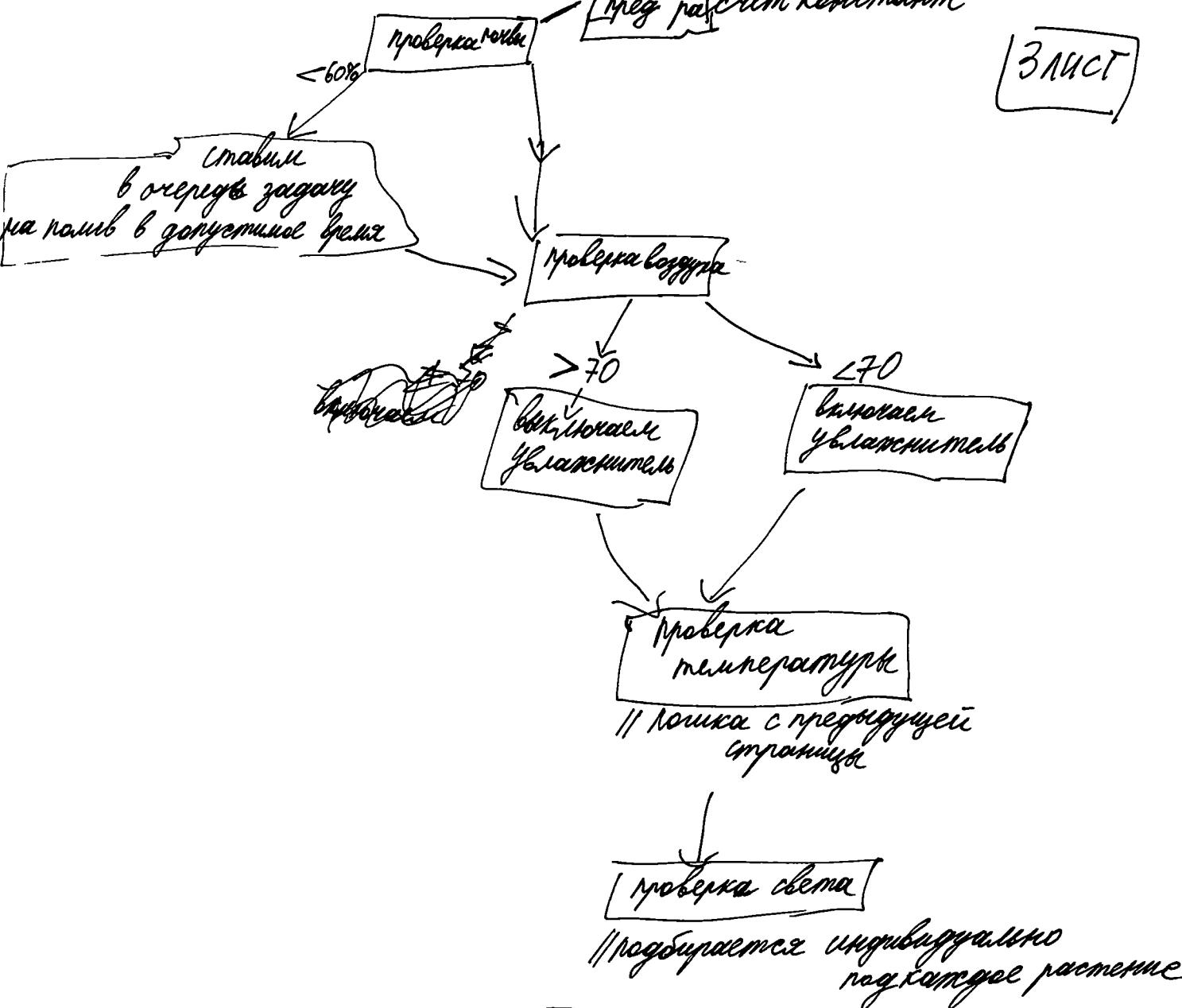
Воздух также устремим к показателю 70

Однако логика do_pov() должна быть сложнее, так как растения можно полить только определенное кол-во раз и в от-ое время суток, то будем ставить задачу на полив в очередь, когда это можно сделать

свет будет иметь наименьший приоритет, поэтому будем проверять в конце

пред расчет констант

3 лист



Тестовые сценарии 15

- 1) Температура < 22 и > 24 не соответствует требованиям хотя бы одного растения
 - 2) Влажность $\neq 70\%$ не соот-ст —
- Для освещения любое значение будет не \neq соот-ть хотя бы одному растению
- Действия системы, как и описывал выше, она в порядке приоритета должна стремиться к референсным значениям
- 3) Мин объем воды утром орхидея + папоротник
вечером драцена + папоротник

ЛИНИИ СЧЕТ

Если вся теплица 30 м², будем считать, что всех растений равное кол-во и у них равная площадь,
тогда: утром $\frac{2}{3} \cdot 30 \cdot 2 = 40 \text{ л}$

вечером: $\frac{2}{3} \cdot 30 \cdot 2 = 40 \text{ л}$

Тогда в день надо минимум 80 л воды 10

Архитектура

Пусть будет class GetAndValidation, который с помощью своих методов будет читать из базы и валидировать полученные данные, в случае их некорректности сразу кидать ошибку и оповещать в систему с требованием проверки датчиков на работоспособность

Основное «приложение» будет с аналитикой и принятием решений, оно будет брать данные через методы класса GetAndValidation так мы будем гарантировать корректные данные на этапе аналитики, и запускать основную логику, которая будет ^{в случае необходимости} вызывать методы ^{внутренних устройств} (нагреватель, кондиционер)

Видимый отдельный сервис для взаимодействия с устройствами, * с помощью которого сможем взаимодействовать с устройствами. Взаимосвязь сервисов между собой могут через Kafka (чтоб в случае отката одного данные сохранялись)

Для визуализации лучше всего будет создать дашборд в Grafana. А для пользователей создать минимальное GUI-приложение

Хорошей практикой будет настроить систему «алертов» (оповещений), чтоб в случае превышения порога определенных значений приходило уведомление с предупреждением

