Технічні науки

УДК 004.021

Кузнецов Роман Олександрович
студент
Харківського національного університету радіоелектроніки
Кузнецов Роман Александрович
студент
Харьковского национального университета радиоэлектроники
Kuznetsov Roman
Student of the
Kharkiv National University of Radio Electronics

Науковий керівник:
Голян Наталія Вікторівна
кандидат технічний наук, доцент кафедри ПІ
Харківський національний університет радіоелектроніки

РЕАЛІЗАЦІЯ АЛГОРИТМУ ВІДСТЕЖЕННЯ УМОВ ЗБЕРІГАННЯ МЕДИКАМЕНТІВ ТА ПОШУКУ НАЛЕЖНИХ ПРИМІЩЕНЬ В ПРОГРАМНІЙ СИСТЕМІ ДЛЯ ЗБЕРІГАННЯ ТА КОНТРОЛЮ СТАНУ ЛІКАРСЬКИХ ЗАСОБІВ

РЕАЛИЗАЦИЯ АЛГОРИТМА ОТСЛЕЖИВАНИЯ УСЛОВИЙ ХРАНЕНИЯ МЕДИКАМЕНТОВ И ПОИСКА НАДЛЕЖАЩИХ ПОМЕЩЕНИЙ В ПРОГРАММНОЙ СИСТЕМЕ ДЛЯ ХРАНЕНИЯ И КОНТРОЛИРОВАНИЯ СОСТОЯНИЯ ЛЕКАРСТВЕННЫХ ПРЕПАРАТОВ

IMPLEMENTATION OF AN ALGORITHM FOR MONITORING STORAGE CONDITIONS OF MEDICINES AND SEARCHING FOR APPROPRIATE PLACEMENTS IN A SOFTWARE SYSTEM FOR STORAGE AND CONDITION CONTROL OF MEDICINES

Анотація. *Проведено аналіз проблеми та програмно реалізовано алгоритм для відстеження умов зберігань медикаментів та пошуку належних для них приміщень.*

Ключові слова: *медикамент, смарт-пристрій, мікроклімат, умови зберігання, приміщення, сповіщення.*

Аннотация. *Проведен анализ проблемы и программно реализован алгоритм для отслеживания условий хранения медикаментов и поиска надлежащих для них помещений.*

Ключевые слова: *медикамент, смарт-устройство, микроклимат, условия хранения, помещение, оповещение.*

Summary. *Analyzed the problem and programmatically implemented the algorithm to monitor storage conditions of the medicines and searching for the appropriate placements.*

Key words: *medicine, smart-device, microclimate, storage conditions, placement, notification.*

Today, in the days of rapidly growing morbidity in the world, there is an urgent need for free access to existing general-purpose drugs, including vaccines. Such drugs require strict adherence to their storage conditions (temperature, relative humidity), because if they are neglected, they can very quickly lose their medicinal properties or even acquire those that can harm the health of patients.

That is why there is a need to create a software system, one of the functions of which would be to automate the tracking of conditions in which medicines are stored and to quickly obtain a list of placements where such medicines can be transferred in case of climate change to inappropriate. In addition, users should receive email notifications when such an event occurs.

The end users of the system are medicines suppliers. In the system, they can manage warehouses, placements, medicines, and their storage. It is appropriate to use a smart device to monitor the microclimate in the placement. It connects to a specific placement, monitors temperature and humidity at regular intervals, and transmits information to the system server.

Thus, we can describe the algorithm itself. The smart device is connected to the placement and is configured by the system administrator via the appropriate web-form to read indicators in it. The smart device iterates at a given interval, reads the temperature and humidity in the placement and sends it to the server to a special endpoint. The Arduino platform [1] and its programming language are used to implement the logic of the smart device. Part of the program implementation is shown in figure 1. This code snippet creates a JSON data object and populates it with the appropriate fields. After that, an HTTP request is generated and sent to the server. The request consists of the HTTP heading line, which includes method type, URL, and protocol version. Then there goes headers, empty line, and request in the formatted JSON. After that, the request is sent via the network and handled on the server`s side.

```
JsonObject &root = jsonBuffer.createObject();
root["id"] = id;
root["temperature"] = temperature;
root["humidity"] = humidity;

String data;
root.printTo(data);

client.println("POST /device HTTP/1.1");
client.println("Host: localhost");
client.println("User-Agent: Arduino/1.0");
client.println("Connection: close");
client.println("Content-Type: application/json");
client.print("Content-Length: ");
client.println(data.length());
client.println();
client.println(data);
```

Fig. 1. Serialization and sending microclimate data to the server

The server is implemented using the Java programming language [2]. It obtains the placement ID, temperature and humidity from the smart device request and calls the next steps of the algorithm, which are shown in figure 2.

```java
private PlacementDto updateSmartDevice(Placement placement, SmartDeviceDto smartDeviceDto) {
    updateSmartDeviceIndicators(placement, smartDeviceDto);

    Warehouse warehouse = placement.getWarehouse();
    Set<Placement> placementsInCurrentWarehouse = warehouse.getPlacements();

    List<PossibleMoveLocations> possibleMoveLocations = placement.getMedicineStorages().stream()
            .map(MedicineStorage::getMedicine)
            .distinct()
            .flatMap(medicine -> findPossibleMoveLocations(medicine, placement, placementsInCurrentWarehouse))
            .collect(Collectors.toList());

    if (!possibleMoveLocations.isEmpty()) {
        sendEmailNotification(warehouse, placement, possibleMoveLocations);
    }

    return PlacementMapper.toPlacementDto(placementRepository.save(placement));
}
```

Fig. 2. Software call of all steps of the algorithm

First there is the call of updating the indicators of the smart device for the placement to new values. All available storages are then retrieved from the placement attached to the smart device. Each such storage is transformed into its own medicine and the operation of obtaining such medicines is performed without repetition. Thus, we get a list of medicines that are stored in the current placement. The next algorithm step called for each medicine is the logic part shown in figure 3.

```java
private Stream<PossibleMoveLocations> findPossibleMoveLocations(Medicine medicine,
                                                                Placement currentPlacement,
                                                                Set<Placement> placements) {
    Integer minTemperature = medicine.getMinTemperature();
    Integer maxTemperature = medicine.getMaxTemperature();
    double currentTemperature = currentPlacement.getSmartDevice().getTemperature();

    if (currentTemperature < minTemperature || currentTemperature > maxTemperature) {
        List<Placement> possibleMovePlacements = placements.stream()
                .filter(place -> !place.equals(currentPlacement))
                .filter(place -> {
                    Double temperature = place.getSmartDevice().getTemperature();
                    return temperature >= minTemperature && temperature <= maxTemperature;
                }).collect(Collectors.toList());
        return Stream.of(new PossibleMoveLocations(medicine, possibleMovePlacements));
    }
    return Stream.empty();
}
```

Fig. 3. The logic of finding placements with normal conditions to transfer

Here is performed retrieving the actual, minimum and maximum temperatures for the medicine. This is followed by a check for compliance with storage conditions, and if it does not pass - search for placements in the current warehouse is started where the microclimate indicators are satisfactory for this medicine. Then a key-value object is created, where the key is the medicine, and the value is the possible placements list to move it there.

This step is called for each medicine. Next goes the check - if the list of such key-value pairs is not empty (i.e., found medicines for which storage conditions are violated), then the logic of generating and sending an e-mail to notify the user is called. Part of the logic of constructing such a letter is shown in figure 4.

```
return possibleMoveLocations.stream()
        .map(moveLocations -> {
            Medicine medicine = moveLocations.getMedicine();
            List<Placement> possibleMovePlacements = moveLocations.getPossibleMovePlacements();
            StringBuilder builder = new StringBuilder("<h3 align=\"center\">")
                    .append(medicine.getName()).append(" (").append(medicine.getStorageForm()).append(")");
            if (possibleMovePlacements.isEmpty()) {
                return builder.append(" - наразі приміщення з відповідними умовами відсутні").append("</h3>");
            }
            return builder
                    .append("</h3>")
                    .append("<table><thead><tr>")
                    .append("<th>Номер приміщення</th>")
                    .append("<th>Тип</th>")
                    .append("<th>Поточна температура</th>")
                    .append("<th>Поточна вологість</th>")
                    .append("</tr></thead><tbody>")
                    .append(possibleMovePlacements.stream().map(placement ->
                            new StringBuilder("<tr>")
                                    .append("<td>").append(placement.getId()).append("</td>")
                                    .append("<td>").append(placement.getType()).append("</td>")
                                    .append("<td>")
                                    .append(placement.getSmartDevice().getTemperature()).append(" °C")
                                    .append("</td>")
                                    .append("<td>")
                                    .append(placement.getSmartDevice().getHumidity()).append(" %")
                                    .append("</td>")
                    ).collect(Collectors.joining()))
                    .append("</tbody></table>");
        }).collect(Collectors.joining());
```

Fig. 4. Part of the logic of constructing a letter of notification of violated conditions

For each key-value pair a table will be formed with a description of the premises for transfer to storage. If the conditions for the drug are violated, but there are currently no placements with a satisfactory microclimate - this will be reported without a table. In other cases, a table will be formed for each medicine, showing the previously found placements. Thus, the user will be fully informed about changes in conditions and possible actions in such situations.

Therefore, an algorithm was designed and implemented to monitor and update the indicators of the microclimate of medicines storage, as well as to find suitable placements in case of violations and notify users of the software system for storage and conditions control of medicines.

References

1. Arduino Hardware [Web-resource]. URL: https://www.arduino.cc/en/hardware (Accessing date: 26.05.2022)
2. Bogunuva M. RESTful Java Web Services: A pragmatic guide to designing and building RESTful APIs using Java, 3rd Edition / Bogunuva Mohanram Balachandar., 2017. 420 p. (Packt Publishing).