

TALLINNA TEHNIKAÜLIKOOL

School of Information Technologies

Aleksei Hüpin 214242IADB

**Medical Web Application**

Distributed system project

Supervisor: Andres Käver

Tallinn 2024

## **Author's declaration of originality**

I hereby certify that I am the sole author of this report. All the used materials, references to the literature and the work of others have been referred to. This thesis has not been presented for examination anywhere else.

Author: Aleksei Hüpin  
25.02.2024

## Table of Contents

Author's declaration of originality.....	2
Introduction.....	4
Database overview of entities.....	5
Scope of work.....	6
Entity Relationship Diagram.....	7
Design Template.....	8

## **Introduction**

The goal of this project is to create a medication application for training management. The idea for this application came from the author's personal work experience in medication software development. This project is aimed at facilitating the process of ordering medication equipment, remedies, and other request types. The main goal is to develop a Minimum Viable Product (MVP) with basic functionality to have the ability to manage patient registration and order medicine for them. Furthermore, these applications can be improved in different ways. Features such as permission policies, signing the order, co-signing the order, health terminology services integration, and much more can be added later.

This application can be a valuable tool for small clinics that want to organize their inner ordering processes and patient treatment.

## Database overview of entities

**Patient** – this entity represent an individual receiving medical treatment.

**Practitioner** – this entity represent a healthcare professional or practitioner involved in patient treatment.

**Contact** – this entity represent a person contact data.

**Order** – this entity represent an order for medication or medical equipment placed by a practitioner for a patient.

**Activity** – this entity represent activities or procedures that practitioners can associate with orders.

**Template** – this entity represents template of order, having ordering data within fields.

**Field** – this entity represent specific fields of data points associated with an activity. These fields may include information such as frequency, quantity or dosage for a particular medical activity.

**Location** – this entity represents location where order can be performed.

**Unit** – this entity represent unit of measurement.

**Status** – this entity represent status of other entity.

Additional tables for many-to-many relationships will be:

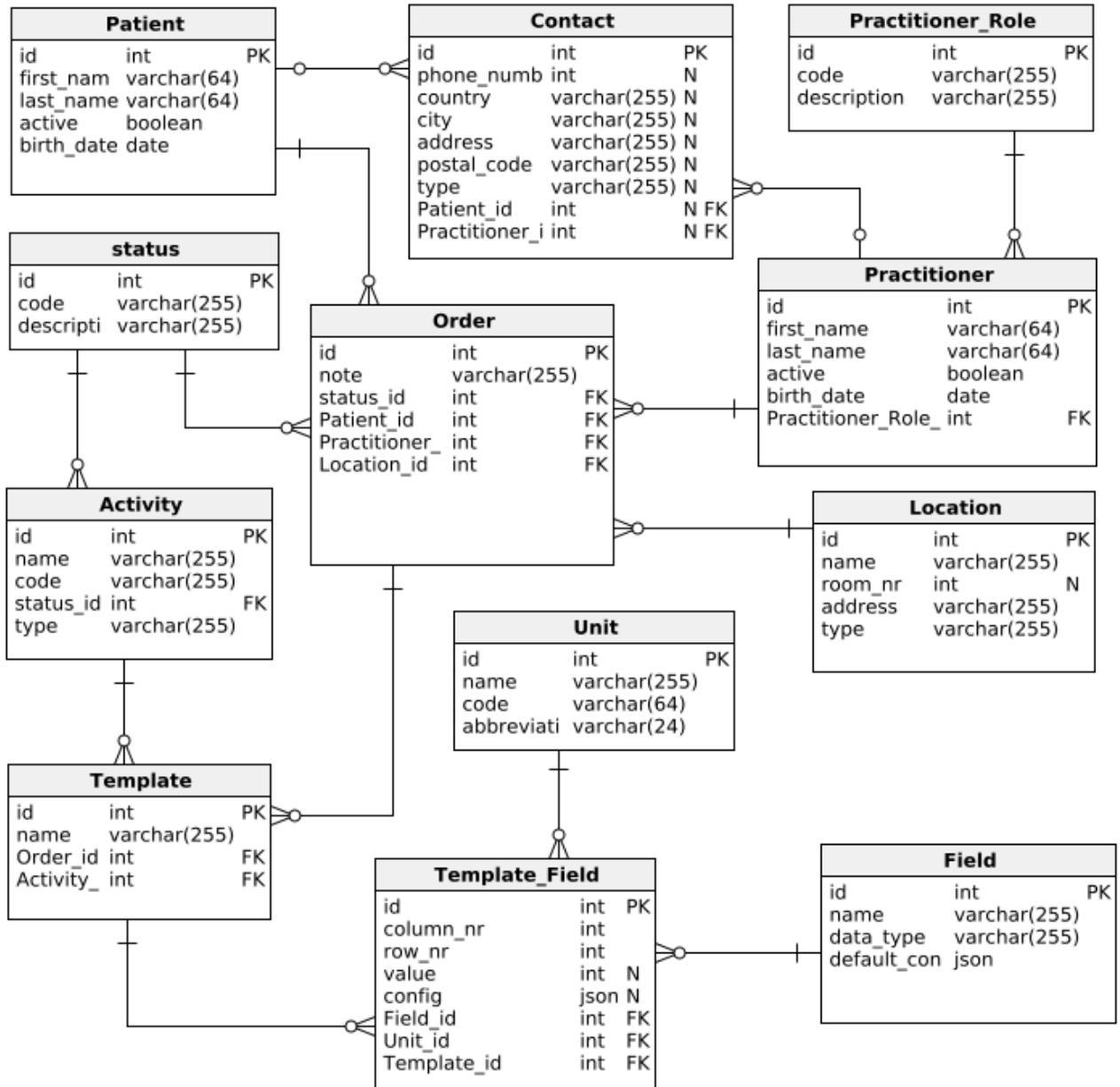
**template** between **order** and **activity\_field**

**template\_field** between **template** and **field**

## Scope of work

1. The Frontend application will be built using the React framework, based on the TypeScript language. React adopts a modular design philosophy, promoting a clear separation of concerns among different parts of the application. In React, components are the building blocks, allowing for a modular and reusable structure.
2. Backend application will be build using .NET framework based on C# language. .NET has a rich class library, and a wide range of tools and resources for building applications. .NET has a strong security model that includes features like code access security, encryption, and role-based security. This makes it easier to build applications that are secure and protect user data.
3. Database PostgreSQL will be used for storing the user authentication data, application entities and their relationship.

# Entity Relationship Diagram



# Design Template

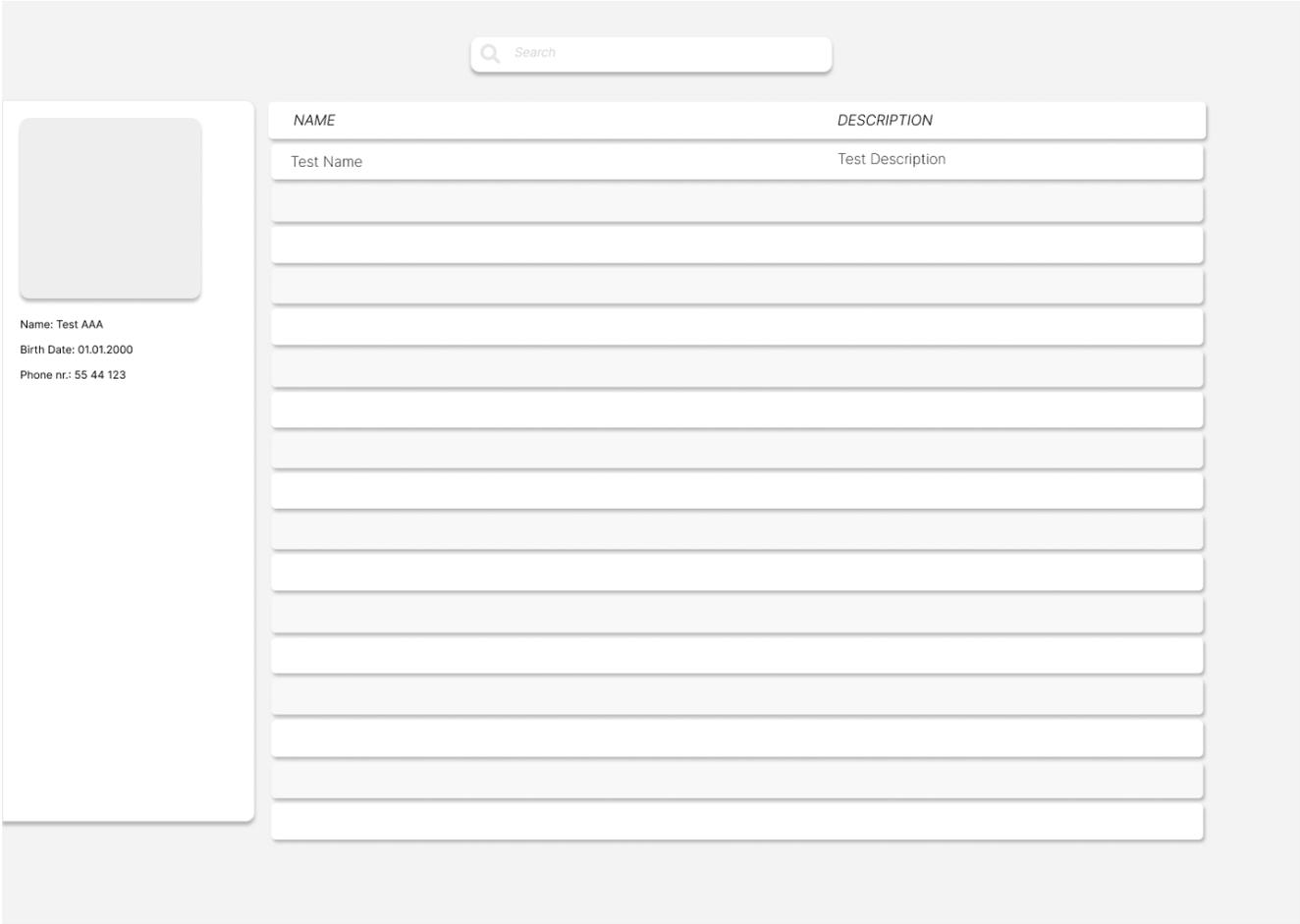


Table component design used to represent the order list.

Left side bar design. Populated with patient data content.

## ORDER

Name: Test AAA  
Birth Date: 01.01.2000  
Phone nr.: 55 44 123

### Acetaminophen Save

Code

Description

#### Configuration

<input type="text" value="Frequency"/>	▼	<input type="text" value="2 times a day"/>	▼	
<input type="text" value="Dosage"/>	▼	<input type="text" value="500"/>	▼	Units
<input type="text" value="Select Field"/>	▼	<input type="text" value="Placeholder"/>		
<input type="text" value="Select Field"/>	▼	<input type="text" value="Placeholder"/>		
<input type="text" value="Select Field"/>	▼	<input type="text" value="Placeholder"/>		

+

Order form component design. Representation of order template with added fields.