

ДОДАТОК А

Код програми

```
class ApplicationController < ActionController::Base
  # Only allow modern browsers supporting webp images, web push, badges, import
  maps, CSS nesting, and CSS :has.
  allow_browser_versions: :modern
end

class CamerasController < ApplicationController
  before_action :set_sector

  def show
    @camera = @sector.cameras.find(params[:id])
    @events = @camera.events.limit(10)
  end

  def new
    @camera = @sector.cameras.new
  end

  def create
    @camera = @sector.cameras.new(camera_params)
    if @camera.save
      redirect_to sector_path(@sector), notice: 'Камера добавлена'
    else
      render :new, status: :unprocessable_entity
    end
  end

  private

  def set_sector
    @sector = Sector.find(params[:sector_id])
  end

  def camera_params
    params.require(:camera).permit(:name, :stream_file, :status)
  end
end

class DashboardController < ApplicationController
  def index
    @sectors = Sector.includes(:cameras)
  end
end
```

```

end
end

class SectorsController < ApplicationController
  def index
    @sectors = Sector.all
  end

  def show
    @sector = Sector.find(params[:id])
  end

  def new
    @sector = Sector.new
  end

  def create
    @sector = Sector.new(sector_params)
    if @sector.save
      redirect_to root_path, notice: 'Sector added'
    else
      render :new, status: :unprocessable_entity
    end
  end

  private

  def sector_params
    params.require(:sector).permit(:name, :status)
  end
end

<h1 class="form-header">Add camera to <%= @sector.name %></h1>

<%= form_with model: [@sector, @camera], local: true, html: { class: 'camera-
form' } do |f| %>
  <div class="form-group">
    <%= f.label :name, "Camera name" %>
    <%= f.text_field :name, class: 'form-input' %>
  </div>

  <div class="form-group hidden">
    <%= f.label :stream_file, "Stream file" %>
    <%= f.text_field :stream_file, value: 'cam_sector_2.png', class: 'form-input' %>
  </div>

```

```

<div class="form-group">
  <%= f.hidden_field :status, value: 'offline' %>
</div>

<div class="form-group">
  <%= f.label :ip_address, "IP-address" %>
  <%= f.text_field :ip_address, class: 'form-input' %>
</div>

<div class="form-group">
  <%= f.submit 'Add camera', class: 'form-submit' %>
</div>
<% end %>

<%= link_to 'Back', sector_path(@sector), class: 'back-link' %>

<h2 class="camera-header">
  <%= @camera.name %> (Сектор: <%= @sector.name %>)
</h2>

<div class="media-container">
  <% if @camera.stream_file.end_with?(".mp4") %>
    <video class="camera-video" controls autoplay loop muted>
      <source src="<%= asset_path(@camera.stream_file) %>" type="video/mp4">
      Video unsupported.
    </video>
  <% elsif @camera.stream_file.end_with?(".png") %>
    
  <% end %>
</div>

<h3 class="events-title">Last events</h3>
<ul class="events-list">
  <% @events.each do |event| %>
    <li class="event-item">
      <span class="event-date"><%= event.occurred_at.strftime('%d.%m.%Y
      %H:%M') %></span> —
      <span class="event-description"><%= event.description %></span>
      [<span class="event-type"><%= event.event_type %></span>]
    </li>
  <% end %>
</ul>

```

```
<%= link_to 'Back', root_path, class: 'button back-button' %>
```

```
class Camera < ApplicationRecord
  belongs_to :sector
  has_many :events, dependent: :destroy

  validates :stream_file, presence: true

  validates :ip_address, presence: true, format: { with: /\A(?:\d{1,3}\.){3}\d{1,3}\z/,
message: "wrong IP-address format" }, allow_blank: true

  after_initialize :set_default_status, if: :new_record?

  private

  def set_default_status
    self.status ||= 'offline'
  end
end
```

```
class Event < ApplicationRecord
  belongs_to :camera

  default_scope { order(occurred_at: :desc) }
end
```

```
class Sector < ApplicationRecord
  has_many :cameras, dependent: :destroy
end
```

```
class User < ApplicationRecord
  # Include default devise modules. Others available are:
  # :confirmable, :lockable, :timeoutable, :trackable and :omniauthable
  devise :database_authenticatable, :registerable,
         :recoverable, :rememberable, :validatable
end
```

```
<h1 class="monitoring-panel-title">Monitoring Panel</h1>
```

```
<%= link_to '+ Add new sector', new_sector_path, class: 'button add-
sector-button' %>
```

```
<div class="sectors-container">
  <% @sectors.each do |sector| %>
    <div class="sector-card">
```

```

<h2 class="sector-name">
  <%= link_to sector.name, sector_path(sector), class: 'sector-name-link' %>
  <span class="sector-status <%= sector.status.downcase %>">
    [<%= sector.status %>]
  </span>
</h2>

<div class="camera-list">
  <%= sector.cameras.each do |camera| %>
    <div class="camera-item">
      <strong><%= camera.name %>:</strong> <%= camera.status.capitalize
%><br>
      <%= link_to 'View Camera', sector_camera_path(sector, camera), class:
'link-to-camera' %>
    </div>
  <%= end %>
</div>

  <%= link_to '+ Add new camera', new_sector_camera_path(sector),
class: 'button add-camera-button' %>
</div>
<%= end %>
</div>

<!DOCTYPE html>
<html>
<head>
  <title><%= content_for(:title) || "Robo Monitoring" %></title>
  <meta name="viewport" content="width=device-width,initial-scale=1">
  <meta name="apple-mobile-web-app-capable" content="yes">
  <meta name="mobile-web-app-capable" content="yes">
  <%= csrf_meta_tags %>
  <%= csp_meta_tag %>

  <%= yield :head %>

  <%=# Enable PWA manifest for installable apps (make sure to enable in
config/routes.rb too!) %>
  <%=# tag.link rel: "manifest", href: pwa_manifest_path(format: :json) %>

  <link rel="icon" href="/icon.png" type="image/png">
  <link rel="icon" href="/icon.svg" type="image/svg+xml">
  <link rel="apple-touch-icon" href="/icon.png">

  <%=# Includes all stylesheet files in app/assets/stylesheet %>

```

```

<%= stylesheet_link_tag :app, "data-turbo-track": "reload" %>
<%= javascript_importmap_tags %>
</head>

<body>
  <%= yield %>
</body>
</html>

{
  "name": "RoboMonitoring",
  "icons": [
    {
      "src": "/icon.png",
      "type": "image/png",
      "sizes": "512x512"
    },
    {
      "src": "/icon.png",
      "type": "image/png",
      "sizes": "512x512",
      "purpose": "maskable"
    }
  ],
  "start_url": "/",
  "display": "standalone",
  "scope": "/",
  "description": "RoboMonitoring.",
  "theme_color": "red",
  "background_color": "red"
}

<h1 class="sector-header">Sector: <%= @sector.name %></h1>

<p class="sector-status"><strong>Status:</strong> <%= @sector.status.capitalize %></p>

<%= link_to ' + Add camera', new_sector_camera_path(@sector), class:
'button add-camera' %>

<h2 class="cameras-header">Cameras inside this sector:</h2>

<ul class="cameras-list">
  <% @sector.cameras.each do |camera| %>
    <li class="camera-item">

```

```
<%= link_to camera.name, sector_camera_path(@sector, camera), class:
'camera-name' %>
  - <%= camera.status.capitalize %>
</li>
<% end %>
</ul>

<%= link_to 'Back', root_path, class: 'back-link' %>
```

ДОДАТОК Б

Демонстраційний матеріал

				моніторингу роботизованої виробничої ділянки			В
Перевір.	Демська Н.П.				Н	1	1
Н.контр.	Стародубцев М. Г.				Кафедра КІТАР ХНУРЕ		
Затв.	Невлюдов І. Ш.						