

# Design Document

## 1. Strona Startowa



## 2. Zakładka „Zadania” ---> Task, Jest to Lista Zadań z wyróżnionymi kategoriami jak:

- Nazwa Kategorii
- Opis Zadania
- Priorytet
- Data ważności

Oraz możliwościami edytowania i usuwania zadania

Lista zadań

Utwórz Kategorie

Nazwa Kategorii	Opis Zadania	Priorytet	Data ważności	
Np. Urodziny	Kupic tort, talerzyki i szampana	Bardzo ważne	28.11.2023	<input type="button" value="Edit"/> <input type="button" value="Delete"/>
				<input type="button" value="Edit"/> <input type="button" value="Delete"/>
				<input type="button" value="Edit"/> <input type="button" value="Delete"/>

3. Po Wciśnięciu Utwórz Kategorie ukaże nam się okno Dodawania Zadania.

Dodaj Zadanie

Nazwa

Opis Zadania

Priorytet

Termin Ważności

dd . mm . rrrr

Create

Cofnij

```
//get - create
Odwolania: 0 | thedjextreme, 3 dni temu | 1 autor, 1 zmiana
public IActionResult Create()
{
    return View();
}
[HttpPost]
[ValidateAntiForgeryToken]
//post -create
Odwolania: 0 | thedjextreme, 3 dni temu | 1 autor, 1 zmiana
public IActionResult Create(zadanie obj)
{
    _db.zadanie.Add(obj);
    _db.SaveChanges();
    return RedirectToAction("Index");
}
```

Tworzenie zadania

- Po wciśnięciu w utworzonym Zadaniu „Edit” będziemy mogli edytować okno zadania

Google ToDo
Zadania
Historia
Logowanie

### Edytuj Zadanie

Nazwa	<input type="text"/>
Opis Zadania	<input type="text"/>
Priorytet	<input type="text"/>
Termin Ważności	<input type="text" value="dd . mm . rrrr"/> <input type="button" value="📅"/>

Edycja zadania

```
//get - edit
Odwołania: 0 | thedjextreme, 3 dni temu | 1 autor, 1 zmiana
public IActionResult Edit(int? id)
{
    if (id == null || id == 0)
    {
        return NotFound();
    }
    var obj = _db.zadanie.Find(id);
    if (obj == null)
    {
        return NotFound();
    }

    return View(obj);
}

//POST - EDIT
[HttpPost]
[ValidateAntiForgeryToken]
Odwołania: 0 | thedjextreme, 3 dni temu | 1 autor, 1 zmiana
public IActionResult Edit(zadanie obj)
{
    if (ModelState.IsValid)
    {
        _db.zadanie.Update(obj);
        _db.SaveChanges();
        return RedirectToAction("Index");
    }
    return View(obj);
}

//GET - DELETE
```

5. Drugą opcją jest „Usuń” jest to potwierdzenie usunięcia zadania z możliwością powrotu

Google ToDo    Zadania    Historia    Logowanie

Usuń zadanie

Delete

Cofnij

Usuwanie Zadania

```
//GET - DELETE
Odwolania: 0 | thedjextreme, 3 dni temu | 1 autor, liczba zmian: 2
public IActionResult Delete(int? id)
{
    //ViewData["TaskName"] = new SelectList(_db.zadanie, "Id", "TaskName");

    if (id == null || id == 0)
    {
        return NotFound();
    }
    var obj = _db.zadanie.Find(id);
    if (obj == null)
    {
        return NotFound();
    }

    return View(obj);
}

//POST - DELETE
[HttpPost]
[ValidateAntiForgeryToken]
Odwolania: 0 | thedjextreme, 3 dni temu | 1 autor, liczba zmian: 2
public IActionResult DeletePost(int? id)
{
    var employee = _db.zadanie.FirstOrDefault(x => x.Id == id);
    _db.zadanie.Remove(employee);
    _db.SaveChanges();

    return RedirectToAction(nameof(Index));
}
```

6. Historia usuniętych zadań, będzie się cechować wszystkimi kategoriami jak przy tworzeniu zadania, dodatkowo będzie data przywrócenia.

## Google ToDo    Zadania    Historia    Logowanie

### Lista zadań wykonanych

Zadanie Wykonane	Priorytet	Data rozpoczęcia zadania	Data zakończenia zadania	Zmienić na aktywne?	
Urodziny	Bardzo ważne	dd . mm . rrrr	dd . mm . rrrr	dd . mm . rrrr	Przywróć
					Przywróć
					Przywróć
					Przywróć