BerghAdmin - Documentation

table of contents

Generated on: 2022-03-11 10:01:49 +0100

1. Introduction

2. README

- get the code at https://github.com/florisz/BerghAdmin/settings/access
- SQL Server
  - install a SQL server on your local (or antoher accessible) machine
  - create a database with name 'BIHZ2021' with a user/password with read/write permissions (can actually be any name but this one is most convenient)
  - install latest entity framework client tools; see https://docs.microsoft.com/en-us/ef/core/cli/dotnet
    just calling 'dotnet tool install --global dotnet-ef' will do the job
- Secrets
  - get secrets file from another developer (is not in the repo)
  - store secrets file in solution directory (keep calm, git will not synchronize it to other repos)
  - load secrets into your private secret store with ./SetSecrets.ps1
  - change the "DatabaseConfiguration:ConnectionString" with your own specific value (use the values you've chosen above and one of the commands below)
  - run 'dotnet ef database update' from the command line (this should create an empty database)
- Source code
  - start Visual studio and load the solution file
  - Open the 'Test Explorer' window and run all unit tests (this should give you a safe feeling)
  - debug/run the default project (should be BerghAdmin)

Help with setting secrets in local store:

```
dotnet user-secrets clear --project ./BerghAdmin
dotnet user-secrets list --project ./BerghAdmin
dotnet user-secrets remove "sectie:setting" --project ./BerghAdmin
dotnet user-secrets add "sectie:setting" --project ./BerghAdmin
```

3. Domain overview

Business background

Main use cases

- secretariaat
- golfers

- ambassadeurs
- financien

4. Analysis

Functionalities

- central donateurs mgmt
- fietstochten
- connection Kentaa
- golfdagen
- facturen (ambassadeurs)
- email sending
- history

5. Design

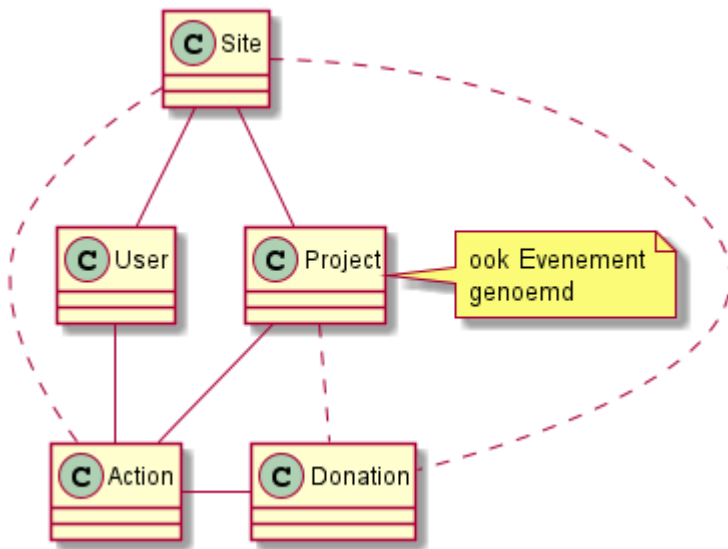The high level class model for the application:

Figure 1. High level class model BerghAdmin

Kentaa is a separate organization which is handling gifts and collecting money from a large group of donateurs. Mainly for the money earned by cyclists but also other 'projects' are handled with the aid of this service. Kentaa stores its information in the following structure:

Figure 2. Data model Kentaa

6. Infrastructure

picture with:

- webapp @Digital Ocean

- Kentaa

- Rabobank (CSV downloads)

- SQL Server MijnServer

- Azure function

- Mailjet

- Syncfusion

== Linux installation

future?

- Nginx: https://www.digitalocean.com/community/tutorials/how-to-install-linux-nginx-mysql-php-lemp-stack-in-ubuntu-16-04

- Host ASP.NET Core on Linux with Nginx https://docs.microsoft.com/en-us/aspnet/core/host-and-deploy/linux-nginx?view=aspnetcore-6.0

- Host and deploy Blazor Server https://docs.microsoft.com/en-us/aspnet/core/blazor/host-and-deploy/server?view=aspnetcore-6.0#linux-with-nginx

- ASP.NET Core SignalR hosting and scaling https://docs.microsoft.com/en-us/aspnet/core/signalr/scale?view=aspnetcore-6.0#linux-with-nginx