Technisch ontwerp Hotelkamerschoonmaakplanapp

Auteur: Yusuf Kemal Öztürk

Datum: 23-11-2022

Versie: 0.1

# Inleiding

Op dit moment is er geen systeem om bij te houden hoelang het duurt om een kamer schoon te maken en of het schoonmaken haalbaar is binnen de geschatte tijd. Daarom wil de product-owner een website waar makkelijk te zien is hoelang het duurt om een kamer schoon te maken. De beheerder wil schoonmaaksters koppelen aan de kamers van het hotel en wil zien of het lukt om de kamer schoon te maken binnen de geschatte tijd.

Inhoudsopgave

[Inleiding 2](#_Toc120134642)

[Documenthistorie 2](#_Toc120134643)

[Samenvatting 3](#_Toc120134644)

[Plan van aanpak 3](#_Toc120134645)

[Ontwikkelomgeving 3](#_Toc120134647)

[Technische infrastructuur 3](#_Toc120134648)

[Activity Diagrammen 3](#_Toc120134649)

[Sequence Diagrammen 17](#_Toc120134660)

[Database ontwerp 23](#_Toc120134670)

[Entity Relationeel Datamodel 25](#_Toc120134671)

[Beveiliging 26](#_Toc120134672)

[Autorisatie 26](#_Toc120134673)

[Ongewenst gebruik van de applicatie 26](#_Toc120134674)

[Beheer 27](#_Toc120134675)

[Back-up 27](#_Toc120134676)

# 

# Documenthistorie

Hier worden alle aanpassingen en versie nummers van het document bijgezet.

|  |  |  |  |
| --- | --- | --- | --- |
| **Versie** | **Status** | **Datum** | **Opmerkingen** |
|  |  |  |  |
| 1.0 | Initiële versie | 23-11-2022 | Eerste versie van het document. |

# Samenvatting

In dit document wordt uitgelegd en laten zien hoe dit project technisch eruit gaat zien. Dit document is om beter te begrijpen hoe het project technisch in elkaar zit en een goede visualisering daarvan geven. Er wordt uitgelegd welke databases er zijn en hoe die zijn verbonden met elkaar, front-end, back-end en de database aan elkaar verbonden zijn.

# Plan van aanpak

# De planning van dit project ziet u in het bestand ‘Planning Hotelkamerschoonmaakapp’ als bijgevoegde bestand.

# Ontwikkelomgeving

## Technische infrastructuur

## **Activity Diagrammen**

Diagram

Description automatically generated

Diagram

Description automatically generated

Diagram

Description automatically generated

Diagram

Description automatically generated

# Diagram Description automatically generated

# Diagram Description automatically generated

# Diagram Description automatically generated

# Diagram Description automatically generated

# Diagram Description automatically generated

# Diagram Description automatically generated

# Diagram Description automatically generated

# Diagram Description automatically generated

# Diagram Description automatically generated

# Diagram Description automatically generated

## **Sequence Diagrammen**

Diagram

Description automatically generated

Diagram

Description automatically generated

# Diagram Description automatically generated

# Diagram Description automatically generated

# Diagram Description automatically generated

# Diagram Description automatically generated

# Diagram Description automatically generated

# Diagram Description automatically generated

# Diagram Description automatically generated with medium confidence

# Diagram Description automatically generated

Diagram

Description automatically generated

Diagram

Description automatically generated

Diagram

Description automatically generated

# Diagram Description automatically generated

# Database ontwerp

Hieronder staan de ontwerpen van de database tabellen die gebruikt zullen worden voor dit project

User Tabel

|  |  |  |  |
| --- | --- | --- | --- |
| Veld | Type | Omschrijving | Voorbeeld |
| UserId | Int(P.K) | Het Id van een user | 1 |
| Email | Varchar(60) | De E-mail van een gebruiker | [YusufK@outlook.com](mailto:YusufK@outlook.com) |
| Wachtwoord | Varchar(60) | Het wachtwoord van een gebruiker | Mis6&286sGyh |
| Naam | Varchar(40) | De naam van een gebruiker | Obejah |
| Rol | Int | De rol van een gebruiker | 2 |
| Actief | Int(F.K) | Identificatie of een schoonmaker aan het schoonmaken is | 1 |

Kamer Tabel

|  |  |  |  |
| --- | --- | --- | --- |
| Veld | Type | Omschrijving | Voorbeeld |
| KamerId | Int(P.K) | Het Id van een kamer | 1 |
| Naam | Varchar(50) | De naam van een kamer | Kamer 1 |

Taken Tabel

|  |  |  |  |
| --- | --- | --- | --- |
| Veld | Type | Omschrijving | Voorbeeld |
| TaakId | Int(P.K) | Het Id van een taak | 1 |
| Taak | Varchar(60) | De naam van de taak | Stofzuigen |

Taken van kamer Tabel

|  |  |  |  |
| --- | --- | --- | --- |
| Veld | Type | Omschrijving | Voorbeeld |
| TaakKamerId | Int(P.K) | Het Id van een taak van een kamer | 1 |
| TaakId | Int(S.K) | Het Id van een taak | 2 |
| KamerId | Int(S.K) | Het Id naam van een kamer | 4 |
| Duur | Int | De tijd die kost om een taak af te maken | 60 |

Opdrachten Tabel

|  |  |  |  |
| --- | --- | --- | --- |
| Veld | Type | Omschrijving | Voorbeeld |
| OpdrachtId | Int(P.K) | Het Id van een opdracht | 1 |
| UserId | Int(S.K) | Het Id van een gebruiker | 2 |
| KamerId | Int(S.K) | Het Id naam van een kamer | 4 |
| Starttijd | Time | Hoe laat een opdracht plaatsvindt | 14:00:00 |
| Eindtijd | Time | Hoe laat een opdracht eindigt | 15:00:00 |

Rapportages Tabel

|  |  |  |  |
| --- | --- | --- | --- |
| Veld | Type | Omschrijving | Voorbeeld |
| RapportId | Int(P.K) | Het Id van een schoonmaakrapport | 1 |
| UserId | Int(S.K) | Het Id van een gebruiker | 2 |
| KamerId | Int(S.K) | Het Id naam van een kamer | 4 |
| Datum | Datetime | De datum waarop het schoonmaakrapport is gestuurd | 15.10.2022 |
| Starttijd | Time | Hoe laat een opdracht is plaatsgevonden | 14:00:00 |
| Eindtijd | Time | Hoe laat een opdracht is beeindigt | 15:00:00 |
| Opmerking | Varchar(60) | Een opmerking die de schoonmaakster van versturen | Stoel kapot |

# Entity Relationeel Datamodel

Hier wordt in wordt aangetoond welke tabellen met elkaar verbonden zijn en op welke manier.

Table

Description automatically generated

## 

# Beveiliging

## Autorisatie

Voor dit project zijn er 3 soort gebruikers: een super beheerder, een beheerder en een schoonmaakster. Deze gebruikers hebben hun eigen admin-panel en ieder admin panel is anders. De functionaliteiten van de super beheerder en de beheerder zijn vrijwel hetzelfde. Het enige verschil is dat de super beheerder accounts kan beheren.  
  
Hoe de gebruikers onderscheid zullen zijn gaat zo: alle gebruikers worden opgeslagen op een database tabel namelijk, users. Ieder user heeft een rol die een waarde van een integer heeft. Afhankelijk van de integer, weet het systeem welke gebruiker het is en welke rol die heeft.

## Ongewenst gebruik van de applicatie

Hieronder staan alle ongewenste situaties van dit project

* **Het onnodig verwijderen van accounts**: Beheerders kunnen accounts beheren. Dus accounts aanmaken en verwijderen. Omdat er meerdere beheerders zijn, het kan zijn dat beheerders zomaar accounts kunnen verwijderen en dat is niet wenselijk. Zo kan bijvoorbeeld een beheerder niet meer inloggen en niet zien welke schoonmaakster gekoppeld is aan welke kamer en hoelang die schoonmaakster bezig is. Om dit te voorkomen is er een ‘super beheerder’. Die is de enige die accounts kan aanmaken en verwijderen
* **Taken niet afmaken maar toch afvinken en versturen:** Schoonmaaksters vinken taken af die afgemaakt zijn. Het wordt niet gecontroleerd of de schoonmaakster de kamer wel of niet heeft schoongemaakt. Een schoonmaakster kan dan bijvoorbeeld alle taken gelijk afvinken terwijl de schoonmaakster die taken niet heeft gedaan. Om dit te voorkomen is er een timer die meet hoelang de schoonmaakster bezig is. Als er een te groot verschil is tussen de ingevulde duur van de takenen de afgeronde tijd dan wordt dat vermeld aan de beheerder.

# Beheer

## Back-up

Als back-up systeem wordt GitHub gebruikt. Op GitHub is het makkelijk te zien wie, wat heeft veranderd op welke bestand. Er wordt ‘gepushed’ iedere keer als een functionaliteit afgerond is of als een grote bug is opgelost. Zo blijft iedereen in een team up-to-date.

U kunt door [hier](https://github.com/yusufkemal1001/Hotelkamerschoonmaakplanapp) te klikken de GitHub repository bekijken.