Samenvatting

In dit document wordt het technische ontwerp van zowel de web-app als de C# applicatie

Technisch ontwerp

Gemaakt door groep 3

Inhoud

**Geen inhoudsopgavegegevens gevonden.**

# 1. Technisch ontwerp voor de web-app

Hierin komt alle informatie conform de technische funtionaliteiten van de web applicatie.

## 1.1 Hoe werkt de web-app

De web-app staat in samenhang met een database. Door de database connectie wordt het mogelijk gemaakt om data op te halen en er ook in te verwerken. Het data verwerken wordt gedaan via forms, deze dienen ingevuld te worden(door de gebruiker) en in de achtergrond worden deze dan weer geüpload in de database.

## 1.2 Login systeem

Het login systeem werkt met 4 verschillende functionaliteiten.

* HTML forms
* php
* Mysql
* Database

Het systeem gebruikt de forms om informatie van de gebruiker te krijgen, daarna wordt deze informatie doorgestuurd naar een php bestand in dit php bestand gaan we kijken of de informatie die is doorgestuurd ook echt degelijke informatie is, we gaan kijken of het email adres ook echt het formaat heeft van een email adres dus dat er een @ een . en dan een domein in zit. Voor het wachtwoord gaan we kijken of dat ze allebei overeen komen zodat we zeker weten dat de gebruiker dit wachtwoord ook echt wil hebben. Ook gaan we gebruik maken van prepare statements zodat we mysql injections kunnen voorkomen.

Op het moment dat de gebruiker ingelogd is kan de gebruiker een aantal verschillende functionaliteiten uitvoeren deze functionaliteiten zijn ook te vinden in het functioneel ontwerp.

De verplichte functies:

* Aanmaken / verwijderen van teams
* Aanmaken van een poule
* Wedstrijd schema maken
* Wedstrijd aanmaken / aanpassen
* Spelers aanmaken / linken

De optionele functies:

* Data van een team aanpassen
* Data van een speler aanpassen / verwijderen

### 1.2.1 Aanmaken / verwijderen van teams

De verschillende functionaliteiten die nodig zijn:

* Database
* Html forms
* Php
* Mysql

Voor het aanmaken of verwijderen van een team moet de website connectie maken met onze database hiervoor hebben wij gekozen om PDO te gebruiken. Als we eenmaal een connectie hebben met de database dan kunnen we ook de database aanpassen. Om een team aan te maken zou de gebruiker in een form de volgende gegevens voor een team moeten invullen.

* Team naam
* Poule nummer

Om deze gegevens te verifiëren gaan we kijken of de naam niet al bestaat en of dat het poule nummer al wel is aangemaakt, als de poule nog niet is aangemaakt dan kan de gebruiker dit ook doen op de beheer pagina.

### 1.2.2 Aanmaken van een poule

De verschillende functionaliteiten die nodig zijn:

* Database
* Html forms
* Php
* Mysql

Een poule aanmaken lijkt heel erg op het aanmaken van een team, hiervoor moet de gebruiker ook een aantal gegevens in vullen in een html form. Die gegevens zijn als volgt:

* Naam

Voor de naam gaan we verifiëren of de naam niet al bestaat.

### 1.2.3 Wedstrijd schema maken

### 1.2.4 Wedstrijd aanmaken / aanpassen

De verschillende functionaliteiten die nodig zijn:

* Database
* Html forms
* Php
* Mysql

Als je een wedstrijd wil aanmaken dan zou de gebruiker de volgende gegevens in moeten voeren in een html form:

* Team nummer voor team a
* De behaalde score voor team a
* Team nummer voor team b
* De behaalde score voor team b

Voor de team nummers gaan we nakijken of de teams wel echt bestaan en voor de score gaan we kijken of dat de score niet lager dan 0 is en niet hoger dan 100 is.

Voor het aanpassen van de behaalde score van een bepaalde wedstrijd gelden dezelfde regels.

### 1.2.5 Spelers aanmaken / linken

De verschillende functionaliteiten die nodig zijn:

* Database
* Html forms
* Php
* Mysql

Voor het aanmaken van een speler moet de gebruiker de volgende gegevens invullen:

* Studenten nummer van de speler
* ID van het team waar de speler aan toegevoegd word
* Voor en achternaam van de student

Voor het verifiëren van een ID gaan we kijken of er ook echt een team bestaat met het gegeven ID.

Voor de andere gegevens hoeven we niks te verifiëren.