**Technisch Ontwerp**

Uitgaanscentrum De Bontekoe

Inhoudsopgave

[1 Samenvatting 3](#_Toc401263144)

[1.1 Samenvatting voor de klant 3](#_Toc401263145)

[1.2 Versiebeheer 3](#_Toc401263146)

[1.3 Verzendlijst 3](#_Toc401263147)

[2 Plan van aanpak 4](#_Toc401263148)

[2.1 Op te leveren producten 4](#_Toc401263149)

[2.2 Planning 4](#_Toc401263150)

[3 Ontwikkelomgeving 5](#_Toc401263151)

[3.1 De technische infrastructuur 5](#_Toc401263152)

[3.2 programmeertaal / ontwikkeltool 5](#_Toc401263153)

[4 Specificaties van de interface 6](#_Toc401263154)

[4.1 Interface 6](#_Toc401263155)

[5 Specificatie van de database 7](#_Toc401263156)

[5.1 Specificaties van de database 7](#_Toc401263157)

[5.2 De grootte van de database 7](#_Toc401263158)

[6 Beveiliging en onderhoud 8](#_Toc401263159)

[6.1 Beveiliging 8](#_Toc401263160)

[6.2 Beheer 8](#_Toc401263161)

[7 Slotconclusie 9](#_Toc401263162)

[7.1 Verantwoording 9](#_Toc401263163)

[7.2 Brondocumentatie 9](#_Toc401263164)

## 1 Samenvatting

### 1.1 Samenvatting voor de klant

Dit document omvat de rapportage over de fase Technisch Ontwerp van <project>. Alle

hoofdonderdelen alsmede conclusies en aanbevelingen zijn verwerkt in dit totaaloverzicht,

met eventuele verwijzingen naar separate uitgebreide documenten, die gedurende deze fase

zijn opgesteld.

### 1.2 Versiebeheer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Versie** | **Status** | **Datum** | **Auteur** | **Opmerkingen** |
| **0.1** | **Niet definitief** | **16/05/16** | **C.Parrott** | **Eerste oplage** |
| **X.X** |  |  |  |  |
| **X.X** |  |  |  |  |
| **X.X** |  |  |  |  |

### 1.3 Verzendlijst

Dit document wordt ter beschikking gesteld aan:

Dit document wordt ter beschikking gesteld aan:

**Werknemers:**

Dennis van Zanten - Project Leider / Backend Developer

Curig Parrott - Back-end Developer / Documentatie

Vincent Toonen - Back-end Developer / Documentatie

Gianluigi Overtoom - Front-end Developer / Logo ontwerper

Killian van der Water - Front-end Developer / Logo ontwerper

**Werkgevers/Opdrachtgever:**

Dhr. Pols - Opdrachtgever

Dhr. Volwater - Budgethouder CodeIT

Dhr. Prajo - Projectbegeleider

Dhr. Wijnands - Marketeer CodeIT

## 2 Plan van aanpak

### 2.1 Op te leveren producten

De producten die we opleveren zijn :

* Het logo van CodeIT.
* Rapport Informatiebehoefte.
* Functioneel Ontwerp.
* Technisch Ontwerp.
* Een responsive webapplicatie van het restaurant.
* Een responsive webapplicatie voor het uitgaan.

### 2.2 Planning

Onze planning die we gebruiken:

**Bezigheid Gemaakt door Einddatum**

Rapport Informatiebehoefte Dennis van Zanten 10-05-2016

Functioneel Ontwerp Vincent Toonen 11-05-2016

Logo CodeIT Kilian van der water/ 15-05-2016

Gianluigi Overtoom

Technisch Ontwerp Curig Parrott 18-05-2016

Responsive webapplicatie van het restaurant Alle werknemers 31-05-2016

Responsive webapplicatie voor het uitgaan Alle werknemers 07-06-2016

## 3 Ontwikkelomgeving

### 3.1 De technische infrastructuur

/\*Hier ga je beschrijven in welke omgeving de applicatie wordt toegepast. In je Functioneel Ontwerp heb je deze omgeving al globaal beschreven.\*/

De 2 webapplicaties moeten uiteindelijk gehost worden op een server, die ook gebruik maakt van een SQL Database.

/\*Is het standalone, een netwerk of een web applicatie. Geef bij elke omgeving aan welke technische eisen eraan worden gesteld. Denk bij standalone bv. aan de eisen aan de PC zowel hardware matig als softwarematig. Lever je een executable versie of is er aparte software nodig om de applicatie te laten draaien.\*/

Het eind product wordt geleverd in een .zip bestaand, in dit bestand staan de PHP, CSS, Javascript, afbeeldingen en de database (deze wordt geleverd als een .sql bestand). Deze zullen vervolgens een eigen map krijgen in op de server om hierna met gemakt gehost te kunnen worden.

/\*Bij een netwerkomgeving heb je een server nodig en een directory op de server waar je applicatie wordt geplaatst. Let hierbij op de rechten; wie heeft toegang en wie niet. Dit geldt zowel voor beheer als voor gebruik.\*/

Moet nog besproken worden?

/\*Als het gaat om een grote applicatie is het wellicht verstandig om een eigen webserver te installeren. Hier beschrijf je wat er dan allemaal moet gebeuren, evt. in overleg met andere specialisten in het bedrijf. Bij een kleine applicatie kun je wellicht gebruik maken van een provider. Dan ben je afhankelijk van de provider voor de voorzieningen die er zijn om een applicatie uit te voeren. Bv. welke scripting talen worden ondersteund. Let ook op de performance.\*/

Voor de afronding wordt alles nog getest en de performance wordt gemeten d.m.v. verschillende tools zoals <https://gtmetrix.com/>. Voor de hosting maken wij gebruik van een derde partij hosting, wij hebben hiervoor gekozen om het gemakkelijker te maken voor de klant zelf.

In principe wordt alles ondersteund als het maar kan communiceren met de rest, denk hiervoor bijvoorbeeld aan HTML, jQuery, Less en Sass.

### 3.2 programmeertaal / ontwikkeltool

/\*Om de applicatie te ontwikkelen en te testen gebruik je meestal een ontwikkelomgeving. Deze is uiteraard afhankelijk van de echte omgeving waarin de applicatie draait. Natuurlijk is ook van belang in welke programmeertaal de applicatie wordt ontwikkeld. Is dit een Access applicatie met VB of een VB.NET applicatie met daarbij de ontwikkelomgeving. Is het een WEB applicatie met ASP.NET en MSSQL, etc.

Hier geef je aan welke taal je gebruikt en hoe je de ontwikkelomgeving moet inrichten (client-server, standalone, web omgeving), welke hard- en software je nodig hebt voor de ontwikkeling en het testen.\*/

De talen die worden gebruikt om de webapplicatie te bouwen zijn:

* PHP Versie 5
* Symphony
* CSS
* Javascript
* SQL
* HTML

## 4 Specificaties van de interface

### 4.1 Interface

/\*Alle schermen die je in het functioneel ontwerp hebt gedefinieerd ga je hier gedetailleerd beschrijven.

* Wat doen de knoppen als je er op klikt.
* Wat moet er worden geregeld voor het scherm wordt geopend?
* Hoe groot zijn de windows die worden geopend.
* Welk lettertype en andere eigenschappen hebben ze.
* Als er Macro’s zijn gebruikt beschrijf dan de functionaliteit en geef aan bij welk veld of welke knop van een scherm ze horen.
* Als er queries nodig zijn, geef dan aan waar ze worden aangeroepen en hoe ze eruit zien bv. in SQL.

Let erop dat het hier gaat om een overdracht naar de programmeurs!

Alle code die nodig is om de applicatie te realiseren is hier beschreven in de vorm van PSD’s.

Als je via het OO model gaat werken komen hier de diagrammen die bij het OO model gebruikt worden, zoals klassendiagram en sequentiediagrammen

Ook andere schema’s en diagrammen die inzicht geven in de technische werking van de applicatie kun je hier opnemen.\*/

## 5 Specificatie van de database

### 5.1 Specificaties van de database

/\*In het functioneel ontwerp heb je al beschreven hoe de database eruit ziet, welke tabellen worden gebruikt en welke relaties er zijn tussen de tabellen. Wellicht heb je ook al de velden van de tabellen beschreven.

Hier neem je alle ander zaken op die van belang zijn om te weten over de te bouwen database zoals:

* welke typering heeft een veld en waarom, alfanumeriek, numeriek, binair, hoeveel tekens
* Zijn er formats gemaakt voor de velden.
* Moeten velden worden gevuld of mogen ze leeg blijven.
* Zijn er sleutels gedefinieerd (m.a.w. is een veld uniek).
* Is er een format op het veld gelegd, bv. een postcode.

Geef alle queries die worden gebruikt. Als je rapporten genereert leg dan uit hoe datgene dat afgedrukt wordt tot stand komt. \*/

De Queries zijn nog onbekend wel weten we een aantal dingen zeker.

* We maken gebruik van **SELECT**
* We maken gebruik van **INSERT INTO**

**Rest\_Klant**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Kolom Naam | Kolom lengte | Kolom type | Kolom extra | Kolom Note |
| ID | 9000 | INT | Primair | Dit is het ID van de klant |
| Naam | 20 | TEXT | Niks | De naam + achternaam van de klan |
| Email | 20 | TEKST | Niks | Het email van de klant |
| Datum | 20 | INT | Niks | De datum dat de klant een tafel wilt reserveren\* |
| Nummer | 3 | INT | Foreign (Rest\_Tafel) | De foreign key van het tafel nummer |

\* De foreign key is er om te kijken of het tafel nummer al in een andere entry zit, anders wordt er een error weergeven

**Rest\_Tafel**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Kolom Naam | Kolom lengte | Kolom type | Kolom extra | Kolom Note |
| Nummer | 9000 | INT | Primair, Uniek | Het nummer van de tafel |
| Plek | 20 | TEXT | Niks | Dit is de plek van de tafel\* |

\* De plek wordt alleen gebruikt om aan te duiden waar deze tafel zich bevindt, Het getal wordt omgezet in positie op een visueel duidelijke map

/\*<<code en specifieke formats horen bij het technisch ontwerp, tenzij de klant dat moet weten>>\*/

### 5.2 De grootte van de database

/\*

Beschrijf welke grootte de database nu heeft en welke er wordt verwacht. Wat is de groeiverwachting. Wat betekent dit voor de technische eisen die aan de hard- en software wordt gesteld. Hoe ziet de testdatabase eruit, hoeveel records neem je op en welke? \*/

***Zie 5.1 voor nu***

## 6 Beveiliging en onderhoud

### 6.1 Beveiliging

/\*

Hierin geef je aan hoe de applicatie wordt beveiligd en hoe de back-up geregeld is.

Let hierbij op:

* autorisatie
* hoe zorg je ervoor dat de gebruiker de applicatie niet per ongeluk overhoop helpt
* hoe zorg je ervoor dat er geen ongewenst gebruik wordt gemaakt van de applicatie
* Op welke manier is back-up geregeld
* Wat heb je opgenomen als eisen in het functioneel ontwerp.\*/

Van elke versie wordt een back-up gemaakt, deze is te vinden in onze GitHub repo.

Onze repo is vindbaar via [deze link](https://github.com/KiwisVersusKoalas/BonteKoe). Er wordt voor deployment zorgvuldig getest, zowel op uithoudingsvermogen als op exploits en backdoors, dit wordt allemaal gedaan via het OS genaamd Kali-Linux

### 6.2 Beheer

/\*

Hierin geef je aan wat er komt kijken bij het beheer. Moeten bestanden worden geschoond op zijn tijd en hoe gaat dat dan. Welke zaken moeten regelmatig worden gecheckt?\*/

De enige dingen die bijgehouden kunnen worden zijn de hoeveelheid entries per dag, indien dit er massief veel zijn kan dit een mogelijke poging tot database overload zijn.

Voorderest hoeft er nauwelijks tot niks bijgehouden worden. Alles is fully automated.

## 7 Slotconclusie

/\*Hierin geef je een verantwoording van de keuzes die je hebt gemaakt in je technisch ontwerp.\*/

Ik heb voor een duidelijke layout gekozen, matchend met het Functioneel ontwerp van Vincent om alles zo duidelijk mogelijk te maken, de reden voor dezelfde stijl is puur zodat de potentiele lezer dan niet opnieuw moet leren navigeren.

### 7.1 Verantwoording

* Dit is het eerste model, nog geen code is geschreven nor bedacht

### 7.2 Brondocumentatie

De bronnen die wij (gaan) gebruiken zijn:

* Stackoverflow
* Stackoverflow
* Google
* Stackoverflow