Project 2 - ServiceDesk Portaal

Projectbriefing

Projectnaam: ServiceDesk Portaal "FixIT"

Datum: 11 december 2025

Opdrachtgever: Logistiek Centrum "De Haven"

Contactpersoon: Mevr. S. Bakker (Hoofd Automatisering)

---

1. Achtergrond en Probleemstelling

Binnen ons logistiek centrum werken 80 medewerkers die dagelijks afhankelijk zijn van scanners, computers en printers. Als er iets stuk gaat, lopen medewerkers nu naar de IT-afdeling of sturen ze een WhatsApp-bericht naar één van de beheerders.

Hierdoor raken meldingen kwijt, weten we niet wat de status is van een reparatie en hebben we geen inzicht in hoeveel storingen er eigenlijk zijn. Dit moet geprofessionaliseerd worden.

2. Doelstelling

Wij willen een **web-based ticketingsysteem (FixIT)** laten ontwikkelen.

Medewerkers moeten zelf eenvoudig een storing ("Ticket") kunnen aanmaken. De IT-afdeling moet deze tickets centraal kunnen beheren, prioriteren en afmelden.

3. Doelgroepen

1. **Medewerkers (Gebruikers):** Willen snel melden dat hun PC stuk is en zien wanneer het gemaakt is.
2. **IT-Support (Beheerders):** Willen een lijst met openstaande taken, gesorteerd op prioriteit.

4. Gewenste Functionaliteiten (Must-Haves)

Voor het MVP verwachten wij de volgende functies:

- **Authenticatie:**
  - Inlogscherm voor alle gebruikers.
  - Rol-verdeling in de database: `user` (medewerker) en `admin` (IT-support).
- **Voor Medewerkers:**
  - Dashboard met overzicht van *eigen* gemelde tickets.
  - Formulier om een nieuw ticket aan te maken (Onderwerp, Beschrijving, Categorie: Hardware/Software).
  - Zichtbare status van hun ticket (bijv. "In Behandeling" of "Opgelost").
- **Voor IT-Support (Admins):**
  - Centraal dashboard met *alle* openstaande tickets van het hele bedrijf.
  - Detailpagina per ticket om deze te bewerken.
  - Mogelijkheid om de **Status** (Open/Closed) en **Prioriteit** (Laag/Hoog) aan te passen.
  - Mogelijkheid om een "Admin Opmerking" (oplossing) toe te voegen aan het ticket.

5. Technische Eisen & Randvoorwaarden

- **Taal & Database:** PHP (8.x) in combinatie met MySQL.
- **Code Architectuur:** Gebruik van `includes` voor header/footer om dubbele code te voorkomen.
- **Veiligheid:** Alle invoer (forms) moet 'sanitized' worden om XSS te voorkomen. Gebruik Prepared Statements voor SQL.
- **Data Integriteit:** Een ticket moet altijd gekoppeld zijn aan de gebruiker die het heeft aangemaakt (Relational Database ID).

6. Budget en Planning

- **Tijdsinvestering:** Wij schatten dit project op circa **40-45 uur** ontwikkeling.
- **Oplevering:** Deadline in overleg, bij voorkeur binnen 5 weken.

7. Gevraagde actie

1. Lever een **Functioneel Ontwerp** (wat gaat het systeem doen?).
2. Lever een **Technische Schets** (ERD van de database + Wireframes).
3. Realiseer de applicatie en lever de broncode op via Git.

BIJLAGE: Specifieke Design & Interface Wensen

Omdat het systeem de hele dag open staat op de schermen van IT-support, zijn de visuele eisen strikt. Wij willen een professionele "SaaS" (Software as a Service) uitstraling.

1. **Kleurcodering van Prioriteiten (Must-have):**
   - In de ticket-lijsten moet de prioriteit direct visueel herkenbaar zijn d.m.v. gekleurde labels ('badges') of randen:
     - *Laag:* Blauw of Grijs.
     - *Normaal:* Groen.
     - *Hoog / Spoed:* Fel Rood (`#D32F2F`).
2. **De "Top-Bar" Navigatie:**
   - In tegenstelling tot veel andere systemen, willen wij **geen sidebar**. Wij willen een strakke, donkere navigatiebalk (Header) over de volledige breedte bovenin.
   - Rechts in deze balk moet de naam van de ingelogde gebruiker staan met een klein profiel-icoon.
3. **Ticket Detail Weergave:**
   - Wanneer je op een ticket klikt, moet de pagina in twee kolommen verdeeld zijn:
     - *Links (70%):* De beschrijving van het probleem en de oplossing.
     - *Rechts (30%):* Een "Infobox" met metadata (Datum aangemaakt, Naam melder, Huidige Status).
   - *Toelichting:* Dit toont aan dat je met Grid of Flexbox kolommen kunt maken.
4. **De "Admin Quick-Filter":**
   - Boven de lijst met tickets moeten 3 grote knoppen staan waarmee de admin direct de lijst kan filteren:
     1. "Alles tonen"

2. "Alleen Spoed"
3. "Alleen Openstaand"

- Dit vereist het gebruik van `GET` parameters in de URL (bijv: `dashboard.php?filter=spoed`) en de verwerking daarvan in je SQL query.

Tips voor jouw Examenportfolio

Waarom is deze opdracht geschikt voor je examen?

1. **Complexiteit (W3):** Het systeem vereist dat je een gebruiker (User ID) koppelt aan een ticket. Dit is een 1-op-N relatie. Zonder deze relatie werkt het systeem niet.
2. **SQL Parameters (W3):** De eis "Admin Quick-Filter" (punt 4 in de bijlage) is de perfecte manier om te laten zien dat je veilige SQL-queries kunt bouwen op basis van input (`WHERE status = ?`).
3. **Veiligheid (W1/W3):** Omdat gebruikers input leveren (tekst in tickets), moet je "Input Sanitization" (`htmlspecialchars`) toepassen om XSS-aanvallen te voorkomen. Dit is een verplicht onderdeel van de exameneisen.
4. **Testen (W4):** Je kunt heel makkelijk scenario's testen:
   - "Als ik inlog als medewerker, mag ik de 'Verwijder Ticket' knop NIET zien."
   - "Als ik filter op Spoed, mag ik geen blauwe tickets zien."

Revision #3
Created 2025-12-11 10:47:36 UTC by Max
Updated 2026-01-08 18:53:58 UTC by Max