

1. Projectoverzicht

Doel: Een broodjesbestelplatform voor school waarbij leerlingen en leerkrachten bestellingen kunnen plaatsen. Admin kan het overzicht behouden en het broodjesbedrijf krijgt een gekoppeld dashboard om bestellingen te verwerken.

Belangrijkste features:

- Smartschool-login (SSO) voor leerlingen en leerkrachten.
 - Rollen: leerling, broodjeszaak, admin.
 - Real-time bestellingen (pulling interval).
 - Betaalopties via Stripe (Apple Pay, Google Pay).
 - Adminpaneel voor statistieken en menu-aanpassing.
 - Broodjeszaak-dashboard voor bestellingverwerking en realtime status.
 - Responsive design voor mobiel en desktop.
-

2. Architectuur & Technologie

Frontend: HTML, CSS, JavaScript

Backend: PHP of Node.js

Database: MySQL

Betaalprovider: Stripe (Apple/Google Pay via Stripe)

Realtime communicatie: Pulling (bijv. elke 5-10s)

Hosting: Externe hosting (schoolserver of externen provider)

Beveiliging: HTTPS, input-validatie, wachtwoord hashing, rolgebaseerde toegang.

Databasevoorstel (MySQL):

- Users (id, naam, rol, klas/bedrijf, smartschool_id)
 - Broodjes (id, naam, beschrijving, prijs, beschikbaar)
 - Menu (id, datum, broodje_id, beschikbaar)
 - Orders (id, user_id, datum, status, totaalprijs)
 - OrderItems (id, order_id, broodje_id, aantal)
 - Statistieken (optioneel, kan berekend worden via queries)
-

3. Functionaliteiten per rol

Leerling

- Smartschool SSO login.
- Bestellingen plaatsen.
- Bekijk status van bestellingen (ontvangen → in bereiding → klaar → geleverd).
- Eventueel geschiedenis van bestellingen.

Broodjeszaak

- Dashboard met realtime bestellingen.
- Kan bestellingen markeren als “in bereiding” of “afgerond”.
- Kan beschikbaarheid broodjes aanpassen (indien nodig).
- Statistieken over omzet, populaire broodjes.

Admin

- Kan alle bestellingen bekijken en beheren.
 - Menu aanpassen (dagen/menu-items/prijs/aanbod).
 - Statistieken bekijken (verkoop per dag, top broodjes, omzet).
 - Status van broodjeszaak zien (hoeveel bestellingen open, verwerkt).
-

4. Development Stappen

Stap 1: Analyse & Planning

- Definieer alle rollen en rechten.
- Plan Smartschool SSO integratie met school.
- Bepaal de betaalmethode via Stripe.
- Beslis over database structuur en hosting.

Stap 2: UI/UX Design

- Maak wireframes voor: leerling-site, adminpaneel, broodjeszaak-dashboard.
- Responsive design: mobiel + desktop.
- Simpele en duidelijke navigatie: bestellen moet in max 3 klikken.

Stap 3: Backend Development

- Gebruikersauthenticatie via Smartschool SSO.
- Rolgebaseerde toegang implementeren.
- Database-opzet in MySQL.
- REST API endpoints voor bestellingen, menu, gebruikers.
- Pulling mechanisme voor realtime updates.

Stap 4: Frontend Development

- Leerlingen- en leerkrachteninterface voor bestellen.
- Adminpaneel: bestellingen, menu, statistieken.
- Broodjeszaak-dashboard: realtime bestellingen, status updates.
- Betaalintegratie met Stripe (Apple/Google Pay).

Stap 5: Integratie

- Verbind frontend en backend via API.
- Test Smartschool SSO login.
- Test betalingen en bestelworkflow.
- Test pulling updates tussen admin en broodjeszaak.

Stap 6: Testing

- Unit tests: database en backend functies.
- UI tests: mobiel en desktop.
- Role-based access tests (leerling, admin, broodjeszaak).
- Betaalfunctionaliteit testen in sandbox omgeving.

Stap 7: Deployment

- Upload website naar gekozen hosting.
- Configureer HTTPS.
- Zorg dat Smartschool-link werkt en gebruikers automatisch inloggen.

Stap 8: Onderhoud

- Dagelijks menu en broodjes beschikbaarheid updaten.
- Monitoren van statistieken.
- Beveiligingsupdates voor backend en frontend.

5. Extra Functionaliteiten (optioneel)

- Notificaties via mail of push bij statusveranderingen van bestellingen.
 - Export van statistieken (CSV, PDF).
 - Extra rol “manager” voor broodjeszaak.
 - Koppeling met mobiele app voor snelle toegang.
-