Contexte :

* Michaël, reconversion
* Moyens de comm' , problèmes actuels
* Objectifs
* **Montrer le site !**

Conception visuelle :

* 1ère ébauche
* Logo
* Charte graphique
* Maquettes sous **FIGMA !** => penser au déploiement en plusieurs temps

Environnement :

* Docker pour MailHog et PHP pour front, et MailPit et mySQL pour back = **Montrer COMPOSE.YAML du front**
* GitHub pour les sauvegardes
* Front : JS/HTML/CSS - Back : Symfony/PHP/JS
* APIPlatform + Test requêtes avec Postman + Dbeaver pour consultation BDD
* Bootwatch et Twig pour Espace Admin

Développement Front :

* Choix JS natif, conception objet : ex: **stage\_vacances.php pour voir sections, puis structure.js**

Cela n'empêche pas le respect des balises HTML.

**Ex: page d'accueil dans navigateur avec extension !! mode ordi**

* Code plus sensible codé en HTML pur : ex : **base\_formulaire\_contact.php**

Un Front Dynamique :

* Le Responsive :

Page d'accueil **: index.css : ligne 5 + ligne 587**

Hamburger : **style.css ligne 319 + structure.js ligne 13.**

* Une écoute active :

Loaders : **loader.css + actualite.js pour le listener**

Changement d'images de fond pour les rôles : **index.js ligne 256**

Masquer la bannière au click : **banniere.js**

* Manipulation du DOM :

Carrousel : **index.js ligne 42 à 86**

Accessibilité/SEO :

* Contrastes couleur : Colour Contrast Analyser
* Balises HTML + texte alt sur images. Ex : carrousel
* .webp
* <meta>
* Sitemap, google analytics

Modèle Conceptuel de Données

Montrer **Looping**

Relation article et photo

Montrer code SQL + migrations 2ème.

CRUD

Espace Admin, Page Actualités

Sécurité

* Formulaires : regex (formulaire\_contact.js), Asserts (dans Form, ResetPasswordType.php), reCaptcha (front>client>register\_traitement.php)
* Mots de passe sécurisés
* Nouveaux comptes validés via un lien mail : controller>clients>clientcontroller.php
* Token d’authentification (authentification client en front login.js via api/login\_check. Lexik\_JWT).
* Htmlspecialchars (dans myaccount.php pour récupérer données client dans formulaire)
* Hashage du mot de passe en base de données (ClientController route register)
* Sécurisation des routes API (dans security.yaml + routes de ArticleController + entité Client)
* Échappement avec Twig
* CSRF Token avec Symfony
* Utilisation de PHPMailer