

TIENDA ONLINE

**PROYECTO FINAL DE CICLO
DESARROLLO DE APLICACIONES MULTIPLATAFORMA**

CURSO 2021-22

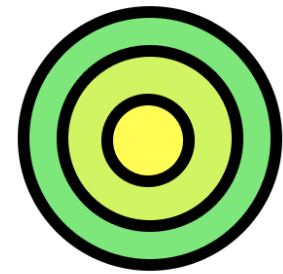


Alejandro Fandos Ferrada



/alejandrofan2





 Meta for Developers

TIENDA ONLINE · Planificación



MARZO

ABRIL

MAYO

JUNIO

Fase 0	Fase 1	Fase 2	Fase 3	Fase 4	Fase 5	Fase 6	Fase 7	Fase 8
+	+	+	+	+	+	+	+	+
Instalación de Java,Flutter,Dart	Re-factorización directorios.	Creación sistema de páginas y navegación.	Creación del proyecto en FireBase y configuración básica.	Creación de entidades para la asimilación de los datos en Flutter.	Creación de IU de las paginas de cambio de nombre, correo, teléfono, contraseña y avatar.	Creación de un sistema de carrito local funcional.	Creación de un sistema de tags para los productos y categorías.	Creación de IU para la selección de coche.
Configuraciones varias de VS Code y GitHub repo.	Eliminación de código residual de 'flutter create'.	Diseño de las páginas del sistema de autenticación.	Creación del Login Service para la autenticación.	Creación de JSON para Productos / Categorías.	Creación de sistema de cambio de nombre, correo, teléfono, contraseña y avatar.	Implementación de 'FireBaseStorage' en el carrito.	Implementación del sistema de productos y tags en 'FireBaseStorage'	Creación de un sistema de selección de coche.
Instalación y configuración de dependencias.		Diseño del resto de paginas ('Home', 'Cart', 'Profile').		Creación de ListViews para mostrar el contenido.			Creación de un sistema de búsqueda por tags.	Implementación del sistema en 'FireBaseStorage'.
							Creación de IU para la búsqueda.	Implementar la selección de coche en el sistema de Búsqueda.



TIENDA ONLINE - SISTEMA DE AUTENTICACIÓN



Meta for Developers

Cueva-Auth

Identificador de la aplicación

Panel

Configuración

Roles

Alertas

Revisión de la aplicación

Productos

Añadir producto

Inicio de sesión con Facebook

Registro de actividad

Descripción general de

Compilación

Authentication

App Check

Firestore Database

Realtime Database

Extensions

Storage

Hosting

Functions

Machine Learning

CuevaDelRecambio

Authentication

Ir a la documentación

Proveedores de acceso

Proveedor

Estado

Correo electrónico/contraseña

Habilitado

Teléfono

Habilitado

Google

Habilitado

Facebook

Habilitado

Agregar proveedor nuevo

Limitación de frecuencia de la aplicación

Cueva-Auth

Identificador de la aplicación: 1005285076793837

0 % del límite usado

Ver detalles

100 % restante

Limites de frecuencia del usuario

0

Usuarios limitados



```
Future<bool> login(BuildContext context, String email,
String password) async {
  try {
    GetIt.I<FirebaseAuth>()
      .signInWithEmailAndPassword(email: email, password:
password);

    return true;
  } on FirebaseAuthException catch (e) {
    if (e.code == 'weak-password')
      ScaffoldMessenger.of(context).showSnackBar(
        SnackBar(content: Text(
          "Try using a strong password.")));
    else if (e.code == 'email-already-in-use')
      ScaffoldMessenger.of(context)
        .showSnackBar(SnackBar(content: Text(
          "That email is in use yet.")));
    else
      ScaffoldMessenger.of(context)
        .showSnackBar(SnackBar(content: Text(
          "An error occurred.")));
    return false;
  }
}
```

```
final GoogleSignInAccount googleSignInAccount =
  await GoogleSignIn().signIn();

if (googleSignInAccount != null) {
  final GoogleSignInAuthentication googleSignInAuthentication =
    await googleSignInAccount.authentication;

  final AuthCredential credential = GoogleAuthProvider.credential(
    accessToken: googleSignInAuthentication.accessToken,
    idToken: googleSignInAuthentication.idToken,
  );
}
```

```
FacebookAuth instance = FacebookAuth.instance;
final LoginResult result = await instance.login();

if (result.status == LoginStatus.success) {
  final AuthCredential credential =
    FacebookAuthProvider.credential(result.accessToken.token);
}
```

```
class CartProvider with ChangeNotifier {  
  CartProvider(this._cart) {  
    _cart.forEach((element) {  
      if (element.quantity > 0) {  
        totalPrice += element.quantity * element.price;  
      }  
    });  
  }  
  
  List<Product> _cart;  
  double totalPrice = 0;  
  bool loading = false;  
  
  List<Product> get cart => _cart;  
  ...  
}
```

```
void addProduct(Product product) {  
  int index = _cart.indexWhere((element) => element.id == product.id);  
  
  if (index == -1) {  
    totalPrice += product.price;  
    _cart.add(product);  
  } else {  
    _cart[index].quantity++;  
    totalPrice += product.price;  
  }  
  
  updateCartToFB(_cart);  
  notifyListeners();  
}
```

TIENDA ONLINE - CARRITO AVANZADO (SYNC)



```
Future<bool> updateCartToFB(List<Product> cart) async {
  User user = FirebaseAuth.instance.currentUser;

  if (user != null) {
    try {
      if (cart.isEmpty) {
        DocumentSnapshot snap = await FirebaseFirestore.instance
          .collection('users')
          .doc(user.uid)
          .get();
        if (snap.exists) {
          await FirebaseFirestore.instance
            .collection('users')
            .doc(user.uid)
            .update({
              'ids': FieldValue.delete(),
              'quantities': FieldValue.delete()
            });
        }
        return true;
      }
    }
  }
  ...
}
```

```
...

List<int> ids = <int>[];
List<int> quantities = <int>[];

cart.forEach((element) {
  print(
    element.id + ' ' + element.quantity.toString() + ' added to cart');
  ids.add(parseInt(element.id));
  quantities.add(element.quantity);
});

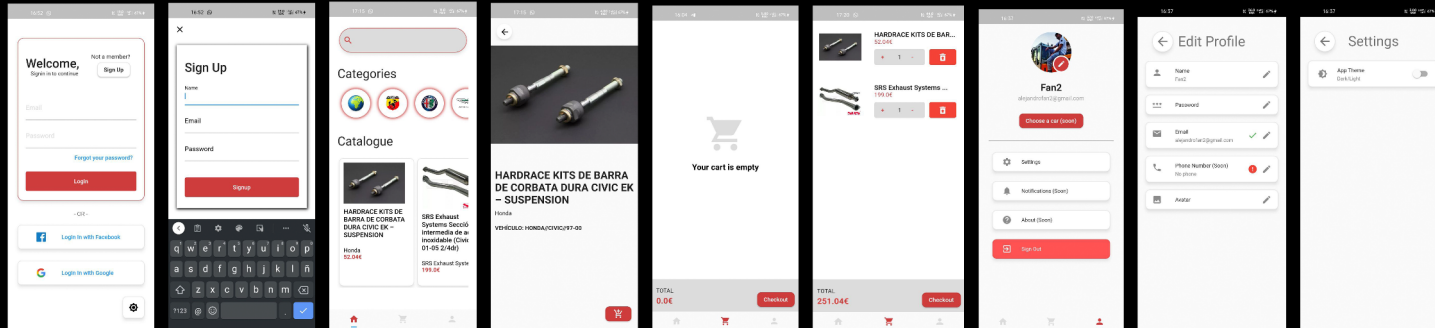
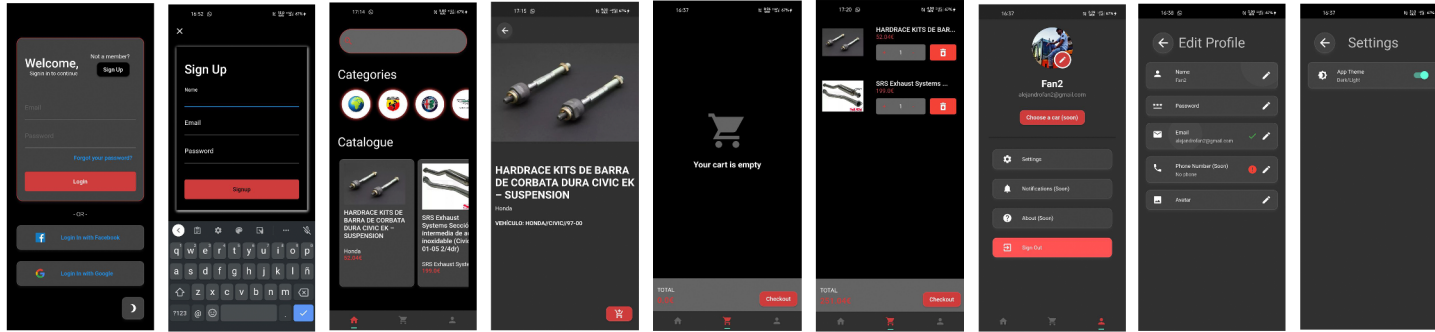
await FirebaseFirestore.instance
  .collection('users')
  .doc(user.uid)
  .set({'ids': ids, 'quantities': quantities});
} on Exception catch (e) {
  print('$e');
  return false;
}
return true;
}
return false;
}
```



TIENDA ONLINE · DEMO FUNCIONALIDAD



android 





TIENDA ONLINE

**PROYECTO FINAL DE CICLO
DESARROLLO DE APLICACIONES MULTIPLATAFORMA**

CURSO 2021-22



Alejandro Fandos Ferrada

