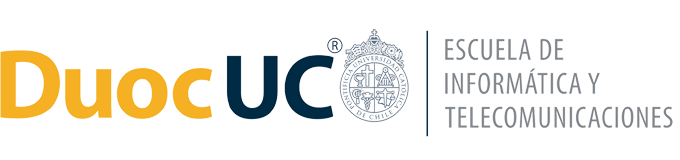
Pestaña 1



| Pruebas unitarias |
| --- |
| *Proyecto: Gustalo APP* |
|  |
| **Revisión*: [99.99]*** |
| **[09/11/24** |

| Responsable | Firma |
| --- | --- |
| Joaquin Carcamo | x |

Objetivo

Este documento cumple con el fin de realizar pruebas unitarias de nuestra app Desarrollada al 100% además de realizar algunas pruebas enfocadas en las funciones, métodos y flujos específicos del componente seleccionado.

Herramientas

* Framework de pruebas (Angular Jasmine)
* Entorno de simulación de ejecución (karma)

Casos de Pruebas unitaria

| ID | P01 |
| --- | --- |
| Funcion/metodo | Login |
| Descripción | Al iniciar las pruebas comenzamos con un error que consiste que **NullInjectorError**: Este error significa que Angular no puede encontrar un proveedor para InjectionToken angularfire2.app.options, el cual es necesario para AngularFireAuth. |
| Precondiciones | Cambiaremos de injectionToken a AngularfireAuth con el que conecta a la base de datos. |
| Pasos de Prueba | Nos dirigimos a login.Page.spec.ts a cambiar el codigo y crea uno |
| Datos de entrada | import { TestBed } from '@angular/core/testing';  import { AngularFireModule } from '@angular/fire/compat';  import { AngularFireAuthModule } from '@angular/fire/compat/auth';  import { environment } from 'src/environments/environment';  import { LoginPage } from './login.page';  describe('LoginPage', () => {  beforeEach(async () => {  await TestBed.configureTestingModule({  imports: [  AngularFireModule.initializeApp(environment.firebaseConfig), // Configuración real de Firebase  AngularFireAuthModule, // Módulo de autenticación  ],  declarations: [LoginPage],  }).compileComponents();  });  it('should create', () => {  const fixture = TestBed.createComponent(LoginPage);  const app = fixture.componentInstance;  expect(app).toBeTruthy();  });  });  importamos (Angularfiremodule,AngularFiremoduleAuth) para permitir el uso de firebase que nuestra base de datos. Luego usamos Testbed lo cual nos proporciona los servicios de firebase, eliminando el error injectionToken |
| Resultado esperado | Jasmine debería quitar de 17 Failed(error) a 16 |
| Resultado Actual | de 17 errores cambio a 16 tenemos 5 success |
| Estado | Aprobado |

| ID | P02 |
| --- | --- |
| Funcion/metodo | * Gestionar categorías * Categorías Servicio, * Ver inventario Page, * Generar Reportes, * Gestionar Productos, * configuración de perfil, * gestionar productos, * Menupage, * forgot-password.page, * ver alertas, * notificaciones page, * Registrar Venta page, * Register Page. * Reporte.Service |
| Descripción | estas paginas de nuestra app tenian el mismo error que el loginPage |
| Precondiciones | Utilizaremos el mismo codigo pero con diferente rutas de cada page que ha estado relacionada con la base de datos |
| Pasos de Prueba | * Gestionar categorías * Categorías Servicio, * Ver inventario Page, * Generar Reportes, * Gestionar Productos, * configuración de perfil, * gestionar productos, * Menupage, * forgot-password.page, * ver alertas, * notificaciones page, * Registrar Venta page, * Register Page.   Estaremos en sus page pero .spec.ts donde realizaremos pruebas unitarias |
| Datos de entrada | import { TestBed } from '@angular/core/testing';  import { AngularFireModule } from '@angular/fire/compat';  import { AngularFireAuthModule } from '@angular/fire/compat/auth';  import { environment } from 'src/environments/environment';  import { LoginPage } from './login.page';  describe('LoginPage', () => {  beforeEach(async () => {  await TestBed.configureTestingModule({  imports: [  AngularFireModule.initializeApp(environment.firebaseConfig), // Configuración real de Firebase  AngularFireAuthModule, // Módulo de autenticación  ],  declarations: [LoginPage],  }).compileComponents();  });  it('should create', () => {  const fixture = TestBed.createComponent(LoginPage);  const app = fixture.componentInstance;  expect(app).toBeTruthy();  });  }); |
| Resultado esperado | 21 specs habian 17 errores de esos bajo a 3 errores al tener la importacion de AngularFire AuthModule |
| Resultado Actual | Tenemos solo 3 errores con la plataforma de jasmine con karma |
| Estado | Aprobado |

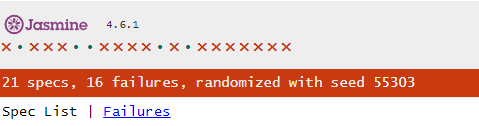
| ID | P03 |
| --- | --- |
| Funcion/metodo | Editar Categoria |
| Descripción | El editar Categoria tenia un error en la introduccion de nombre, should create Failed: Cannot read properties of undefined (reading 'nombre') at templateFn (ng:///EditarCategoriaComponent.js:35:56) at executeTemplate (http://localhost:9876/\_karma\_webpack\_/webpack:/node\_modules/@angular/core/fesm2022/core.mjs:12159:9) |
| Precondiciones | El objeto categoria se inicializa antes de llamar a fixture.detectChanges(). Esto asegura que no sea undefined cuando la vista intente acceder a sus propiedades. |
| Pasos de Prueba | Editar.categoria.spect.ts |
| Datos de entrada | import { ComponentFixture, TestBed } from '@angular/core/testing';  import { EditarCategoriaComponent } from './editar-categoria.component';  import { FormsModule } from '@angular/forms';  import { IonicModule, ModalController } from '@ionic/angular';  describe('EditarCategoriaComponent', () => {  let component: EditarCategoriaComponent;  let fixture: ComponentFixture<EditarCategoriaComponent>;  beforeEach(async () => {  await TestBed.configureTestingModule({  declarations: [EditarCategoriaComponent],  imports: [  IonicModule.forRoot(), // Configura Ionic  FormsModule, // Importa FormsModule si se usa [(ngModel)]  ],  providers: [ModalController], // Proveedor del ModalController  }).compileComponents();  fixture = TestBed.createComponent(EditarCategoriaComponent);  component = fixture.componentInstance;  // Inicializa las propiedades necesarias  component.categoria = { nombre: 'Categoría de prueba' }; // Datos simulados  fixture.detectChanges();  });  it('should create', () => {  expect(component).toBeTruthy();  });  }); |
| Resultado esperado | El Error Cambio de Succes en la prueba de jasmine karma |
| Resultado Actual | de 5 errores Cambiaron a 4 |
| Estado | Aprobado |

| ID | P04 AdminMenuPage |
| --- | --- |
| Funcion/metodo |  |
| Descripción | El injector de firebase pero tuvbimos que importa el popovercontroller para poder ejecutar algunas funciones.  NullInjectorError: R3InjectorError(DynamicTestModule)[PopoverController -> PopoverController]: NullInjectorError: No provider for PopoverController! error properties: Object({ ngTempTokenPath: null, ngTokenPath: [ 'PopoverController', 'PopoverController' ] }) |
| Precondiciones | **PopoverController como proveedor:**   * El servicio PopoverController debe incluirse en los providers del entorno de pruebas para que pueda ser inyectado correctamente en el componente. |
| Pasos de Prueba | Estamos en Admin.page.spect.ts |
| Datos de entrada | import { TestBed } from '@angular/core/testing';  import { AdminMenuPage } from './admin-menu.page';  import { IonicModule, PopoverController } from '@ionic/angular';  describe('AdminMenuPage', () => {  let component: AdminMenuPage;  let fixture: any;  beforeEach(async () => {  await TestBed.configureTestingModule({  declarations: [AdminMenuPage],  imports: [IonicModule.forRoot()], // Configuración de Ionic  providers: [PopoverController], // Proveedor del PopoverController  }).compileComponents();  fixture = TestBed.createComponent(AdminMenuPage);  component = fixture.componentInstance;  fixture.detectChanges();  });  it('should create', () => {  expect(component).toBeTruthy();  });  }); |
| Resultado esperado | Cambiaron los resultado del ng test a 3 errores |
| Resultado Actual | Ng test jasmine karma ha informado que el error ha sido resuelto |
| Estado |  |

| ID | P05 MenuPage |
| --- | --- |
| Funcion/metodo |  |
| Descripción | El mismo error que de admin asi que solo importamos el popovercontroller y el Fireatuhmodule  MenuPage > should create NullInjectorError: R3InjectorError(DynamicTestModule)[PopoverController -> PopoverController]: NullInjectorError: No provider for PopoverController! error properties: Object({ ngTempTokenPath: null, ngTokenPath: [ 'PopoverController', 'PopoverController' ] }) at NullInjector.get (http://localhost:9876/\_karma\_webpack\_/webpack:/node\_modules/@angular/core/fesm2022/core.mjs:1663:27) |
| Precondiciones | **Incluir PopoverController en los providers:**   * Esto asegura que Angular pueda inyectar PopoverController en el componente durante las pruebas.   **Opción 1 (Firebase real):**   * Usa AngularFireModule.initializeApp y AngularFireAuthModule. * Proporciona la configuración real de Firebase desde environment.firebaseConfig. |
| Pasos de Prueba | Menu.page.Spect.ts |
| Datos de entrada | import { TestBed } from '@angular/core/testing';  import { IonicModule, PopoverController } from '@ionic/angular';  import { MenuPage } from './menu.page';  describe('MenuPage', () => {  let component: MenuPage;  let fixture: any;  beforeEach(async () => {  await TestBed.configureTestingModule({  declarations: [MenuPage],  imports: [IonicModule.forRoot()], // Configuración de Ionic  providers: [PopoverController], // Incluye el PopoverController  }).compileComponents();  fixture = TestBed.createComponent(MenuPage);  component = fixture.componentInstance;  fixture.detectChanges();  });  it('should create', () => {  expect(component).toBeTruthy();  });  }); |
| Resultado esperado | errores 0 de jasmine con karma |
| Resultado Actual | ya no arroja ningún error en jasmine karma |
| Estado | Aprobado |

Resultado esperado de las pruebas unitarias:

Antes:



Despues:

