

PROJECT STATUS - QA_Framework_SauceDemo

Última actualización: 11 Diciembre 2024 - Sesión 3
Estado: En proceso de profesionalización (Opción B: Full Professional)













OBJETIVO DE HOY


Llevar el proyecto de "funcional" a "portfolio-ready profesional" con:

- Pytest markers
- Screenshots on failure
- Allure reports
- Documentation completa
- GitHub deploy



COMPLETADO (100%)

Tests Implementados (14/14)

-  TC001: test_login_success (test_login.py)
-  TC002: test_add_single_product (test_products.py)
-  TC003: test_add_multiple_products (test_products.py)
-  TC004: test_view_cart_with_products (test_cart.py)
-  TC005: test_remove_product_from_cart (test_cart.py)
-  TC006: test_start_checkout_button (test_checkout.py) - NUEVO ✨
-  TC007: test_checkout_with_valid_data (test_checkout.py)
-  TC008: test_checkout_without_data (test_checkout.py)
-  TC009: test_validate_checkout_overview (test_checkout.py) - NUEVO ✨
-  TC010: test_complete_purchase (test_checkout.py)
-  TC011: test_complete_purchase_flow (test_e2e.py)
-  TC012: test_continue_shopping_from_cart (test_checkout.py) - NUEVO ✨

Último test run: 6 passed in 30.11s 

Page Objects (5/5)

-  base_page.py
-  login_page.py

- ✓ products_page.py
- ✓ cart_page.py
- ✓ checkout_page.py
- ✓ confirmation_page.py

Estructura Base ✓

```
QA_Framework_SauceDemo/  
├── pages/      ✓ Completo  
├── tests/      ✓ 14 tests funcionando  
├── config/     ✓ Configuración base  
├── core/       ✓ Browser manager  
├── utils/      ✓ Helpers  
├── conftest.py ✓ Fixtures básicos  
├── pytest.ini  ✓ Config básica  
└── README.md   ⌚ Necesita actualización profesional
```

⌚ EN PROCESO - OPCIÓN B: FULL PROFESSIONAL

🔥 FASE 1: Markers & Screenshots (⌚ NEXT)

Objetivo: Organizar tests + screenshots automáticos en failures

Archivos a actualizar:

- ☐ tests/test_login.py - Agregar @pytest.mark.smoke, @pytest.mark.regression
- ☐ tests/test_products.py - Agregar markers
- ☐ tests/test_cart.py - Agregar markers
- ☐ tests/test_checkout.py - Agregar markers
- ☐ tests/test_e2e.py - Agregar @pytest.mark.e2e
- ☐ conftest.py - Agregar hook para screenshots on failure
- ☐ pytest.ini - Actualizar con markers custom

Resultado esperado:

```
bash  
  
pytest -m smoke -v      # Solo tests críticos  
pytest -m regression -v # Suite completa  
pytest -m e2e -v        # Solo E2E
```

Screenshots: screenshots/ con capturas automáticas al fallar

🔥 FASE 2: Allure Reports (⌚ PENDING)

Objetivo: Reports profesionales con gráficos y timeline

Tareas:

- ☐ Instalar: `pip install allure-pytest`
- ☐ Configurar en `pytest.ini`
- ☐ Agregar decorators Allure a tests
- ☐ Generar reports con: `allure generate`
- ☐ Crear carpeta `reports/allure-results/`

Resultado esperado:

- Reports HTML con gráficos
 - Timeline de ejecución
 - Screenshots integrados en reports
-

🔥 FASE 3: Professional Structure (⌚ PENDING)

Objetivo: Organización y configuración nivel senior

Archivos a crear/actualizar:

- ☐ `requirements.txt` - Con versiones fijas de todas las dependencias
- ☐ `.gitignore` - Ignorar venv, **pycache**, reports, screenshots
- ☐ `pytest.ini` - Optimizado con todas las configs
- ☐ Crear carpeta `screenshots/` (si no existe)
- ☐ Crear carpeta `reports/` (si no existe)

Dependencias actuales:

```
pytest
pytest-playwright
playwright
allure-pytest (agregar)
```

🔥 FASE 4: Documentation (⌚ PENDING)

Objetivo: README profesional + documentación completa

README.md debe incluir:

- ☐ Badges (Python version, pytest, tests passing)
- ☐ Descripción del proyecto
- ☐ Features del framework
- ☐ Instalación paso a paso
- ☐ Cómo ejecutar tests
- ☐ Estructura del proyecto
- ☐ Screenshots o GIF demo
- ☐ Reporting (Allure)
- ☐ Contacto/LinkedIn

Archivos adicionales:

- ☐ ARCHITECTURE.md - Explicar diseño del framework
 - ☐ CHANGELOG.md - Historial de cambios (opcional)
-

🔥 FASE 5: GitHub Deploy (⌚ PENDING)

Objetivo: Proyecto público en GitHub para portfolio

Tareas:

- ☐ Crear repo en GitHub (si no existe)
 - ☐ Configurar .gitignore
 - ☐ Commit inicial/actualización
 - ☐ Push a main branch
 - ☐ Verificar que se vea bien en GitHub
 - ☐ Agregar descripción y topics al repo
-

📁 ESTRUCTURA FINAL ESPERADA

```
QA_Framework_SauceDemo/
├── .github/
│   └── workflows/      (futuro: CI/CD)
├── pages/
│   ├── __init__.py
│   ├── base_page.py    ✓
│   ├── login_page.py   ✓
│   ├── products_page.py ✓
│   └── cart_page.py     ✓
```

```
| | checkout_page.py  ✓
| | confirmation_page.py  ✓
| tests/
| | __init__.py
| | test_login.py  ✓ (pendiente: markers)
| | test_products.py  ✓ (pendiente: markers)
| | test_cart.py  ✓ (pendiente: markers)
| | test_checkout.py  ✓ (pendiente: markers)
| | test_e2e.py  ✓ (pendiente: markers)
| config/
| | config.py  ✓
| core/
| | browser_manager.py  ✓
| utils/
| | helpers.py  ✓
| reports/
| | allure-results/ (crear)
| screenshots/ (crear)
| venv/  ✓
| .gitignore (crear)
| conftest.py  ✓ (pendiente: screenshots)
| pytest.ini  ✓ (pendiente: actualizar)
| requirements.txt (crear)
| README.md ⚠ (actualizar completo)
| ARCHITECTURE.md (crear)
| PROJECT_STATUS.md  ✓ (este archivo)
```

CHECKPOINT ACTUAL

ACABAMOS DE TERMINAR:

- ✓ Todos los tests (14/14) funcionando
- ✓ Tests de checkout completos con 3 nuevos
- ✓ Último test run: 6 passed in 30.11s

SIGUIENTE PASO:

- ⚠ FASE 1: Agregar markers a test_login.py

TIEMPO ESTIMADO RESTANTE: 1-2 horas para completar Opción B

NOTAS IMPORTANTES

Credenciales de Test:

- Usuario: `standard_user`
- Password: `secret_sauce`
- URL: `https://www.saucedemo.com`

Comandos Útiles:

```
bash

# Activar venv
.\venv\Scripts\activate

# Correr todos los tests
pytest -v

# Correr solo checkout
pytest tests/test_checkout.py -v

# Correr con markers (después de Fase 1)
pytest -m smoke -v
pytest -m regression -v

# Generar Allure report (después de Fase 2)
pytest --alluredir=reports/allure-results
allure serve reports/allure-results
```

WORKFLOW DE RETORNO

Si interrumpimos la sesión:

```
TÚ: "Retomamos proyecto SauceDemo"
YO: [Leo este PROJECT_STATUS.md]
YO: "OK, estábamos en FASE X, paso Y. ¿Continuamos?"
```

CONTEXTO DEL PROYECTO

- **Estudiante:** Curso QA Automation con Okulik (YouTube)

- **Objetivo:** Primer proyecto portfolio-ready profesional
 - **Timeline:** 6 meses de estudio, en Clase 3 actualmente
 - **Tech Stack:** Python + Pytest + Playwright + Page Object Model
 - **Next Projects:** Más frameworks después de dominar este
-

PRÓXIMOS PASOS (DESPUÉS DE OPCIÓN B)

Futuras mejoras opcionales (otro día):

- 🚀 CI/CD con GitHub Actions
- 🚀 Multi-environment configs (DEV/QA/PROD)
- 🚀 Logging profesional
- 🚀 Parallel execution
- 🚀 Video recording

Pero primero: ✅ Completar Opción B y subirlo a GitHub

Última sesión: Diciembre 11, 2024

Estado general: 🟢 En buen camino, tests completos, falta profesionalización

Próximo milestone: FASE 1 - Markers & Screenshots