

test-scenarios.md - Manual Testing Scenarios

Manual Testing Scenarios for Role-based Authentication

Prerequisites

- Database with migrations applied
 - Environment variables configured
 - Email service configured (optional for email tests)
 - Test social OAuth apps configured
-

Test Scenario 1: Super Admin First Login

Objective





Verify Super Admin can login with environment credentials and access dashboard

Steps




1. **Navigate to login page**
2. Go to `/auth/signin`
3. Verify all login options are visible
4. **Select Super Admin login**
5. Click "Super" tab
6. Verify Super Admin form appears
7. Check placeholder shows correct email
8. **Enter credentials**
9. Email: `sanchai5651@gmail.com`
10. Password: `Safety17`

11. Click "เข้าสู่ระบบ SUPER ADMIN"
12. **Verify redirect**
13. Should redirect to `/admin/super/dashboard`
14. Dashboard should load with user statistics
15. Should show "Super Admin Dashboard" title

Expected Results

-  Login successful
-  Redirected to Super Admin dashboard
-  Can see pending users (if any)
-  All admin functions accessible

Error Cases to Test

-  Wrong email → "ข้อมูลเข้าสู่ระบบไม่ถูกต้อง"
 -  Wrong password → "ข้อมูลเข้าสู่ระบบไม่ถูกต้อง"
 -  Empty fields → "กรุณากรอกอีเมลและรหัสผ่าน"
-



Test Scenario 2: New User Social Login (Admin)

Objective

Test complete flow from social login to role assignment

Steps

1. **Social login as new user**
2. Go to `/auth/signin`
3. Click "Admin" tab
4. Click "เข้าสู่ระบบด้วย Google"
5. Complete Google OAuth (use test account)
6. **Verify pending status**
7. Should redirect to `/pending-approval`
8. Should show "รอการอนุมัติบัญชี" message
9. Should display user info and next steps

10. Super Admin assigns role

11. Login as Super Admin in new tab
12. Go to Super Admin dashboard
13. Find new user in pending list
14. Click "👤" (Admin) button
15. Add reason: "New admin assignment"
16. Click confirm

17. Verify role assignment

18. Check success message appears
19. User should move from pending to admin list
20. Email notification should be sent (check logs)

21. Test new admin access

22. Go back to pending user tab
23. Refresh page or click "ตรวจสอบสถานะอีกครั้ง"
24. Should redirect to `/admin/dashboard`
25. Should see "Admin Dashboard"

Expected Results

- ☒ Social login creates pending user
 - ☒ Super Admin can see and assign role
 - ☒ User receives admin access
 - ☒ Email notification sent
 - ☒ Proper redirects work
-

Test Scenario 3: New User Social Login (Vendor)

Objective

Test vendor role assignment flow

Steps

1. Social login as vendor
2. Go to `/auth/signin`
3. Click "Vendor" tab

4. Click "เข้าสู่ระบบด้วย Facebook"
5. Complete Facebook OAuth
6. **Verify pending status**
7. Should redirect to `/pending-approval`
8. Should show vendor-specific messaging
9. **Admin assigns vendor role**
10. Login as Admin (not Super Admin)
11. Go to admin dashboard
12. Navigate to user management
13. Find pending vendor
14. Assign vendor role with reason
15. **Verify vendor access**
16. Pending user should get vendor access
17. Should redirect to `/vendor/dashboard`
18. Should see vendor-specific features

Expected Results

- ☒ Admin can assign vendor roles
 - ☒ Vendor gets appropriate access
 - ☒ Cannot access admin functions
-

Test Scenario 4: Permission Testing

Objective

Verify role-based access control works correctly

Test Cases

4.1 Super Admin Permissions

- ☒ Can access `/admin/super/dashboard`
- ☒ Can assign admin roles

- ☒ Can assign vendor roles
- ☒ Can assign customer roles
- ☒ Can view all users
- ☒ Can view audit logs

4.2 Admin Permissions

- ☒ Can access `/admin/dashboard`
- ☒ Cannot access `/admin/super/dashboard`
- ☒ Cannot assign admin roles
- ☒ Can assign vendor roles
- ☒ Can assign customer roles
- ☒ Can view non-admin users

4.3 Vendor Permissions

- ☒ Can access `/vendor/dashboard`
- ☒ Cannot access `/admin/*`
- ☒ Cannot assign any roles
- ☒ Can manage own products
- ☒ Can view own orders

4.4 Customer Permissions

- ☒ Can access `/customer/dashboard`
- ☒ Cannot access `/admin/*`
- ☒ Cannot access `/vendor/*`
- ☒ Can browse products
- ☒ Can place orders

4.5 Pending User Permissions

- ☒ Cannot access any dashboard
- ☒ Can only access `/pending-approval`
- ☒ Redirected from all other protected routes



Test Scenario 5: Middleware Protection





Objective

Test route protection and redirects

Steps

1. **Test unauthenticated access**
2. Clear all cookies/localStorage
3. Try to access `/admin/dashboard`
4. Should redirect to `/auth/signin`
5. **Test wrong role access**
6. Login as customer
7. Try to access `/admin/dashboard`
8. Should redirect to `/customer/dashboard`
9. **Test pending user access**
10. Login as pending user
11. Try to access any dashboard
12. Should redirect to `/pending-approval`
13. **Test public routes**
14. Access `/`, `/products`, `/about`
15. Should work without authentication

Expected Results

-  Protected routes require authentication
 -  Role-based redirects work
 -  Public routes accessible
 -  Proper error handling
-



Test Scenario 6: Email Notifications

Objective





Test email notification system

Steps

1. **Test new user notification**

2. New user signs up via social login
3. Check email logs for Super Admin notification
4. Verify email contains user details
5. **Test role assignment notification**
6. Super Admin assigns admin role
7. Check email logs for user notification
8. Verify email contains role details and login link
9. **Test email failure handling**
10. Temporarily break email configuration
11. Assign role to user
12. Verify role assignment still works
13. Check error logs for email failure

Expected Results

-  New user emails sent to Super Admin
 -  Role assignment emails sent to users
 -  System works even if email fails
 -  Proper error logging
-

Test Scenario 7: Database Integrity

Objective





Test data consistency and audit trails

Steps

1. **Test role assignment logging**
2. Assign roles to multiple users
3. Check `role_assignments` table
4. Verify all assignments are logged
5. **Test admin action logging**
6. Perform various admin actions

7. Check `admin_actions` table
8. Verify actions are properly logged
9. **Test login logging**
10. Perform various login attempts
11. Check `login_logs` table
12. Verify both success and failure logs
13. **Test data relationships**
14. Check foreign key constraints
15. Verify cascade deletes work
16. Test data integrity

Expected Results

-  All actions properly logged
 -  Data relationships maintained
 -  Audit trail complete
 -  No orphaned records
-

Test Scenario 8: Error Handling

Objective

Test system behavior under error conditions

Test Cases

8.1 Network Errors

- Disconnect internet during login
- Verify proper error messages
- Test retry mechanisms

8.2 Database Errors

- Simulate database connection issues
- Verify graceful degradation
- Test error recovery





8.3 OAuth Errors

- Test OAuth provider failures
- Verify error handling
- Test fallback options

8.4 Session Errors

- Test expired sessions
- Test invalid tokens
- Verify proper cleanup

Expected Results

-  Graceful error handling
 -  User-friendly error messages
 -  System recovery
 -  No data corruption
-



Test Scenario 9: Performance Testing

Objective





Test system performance under load

Steps

1. **Test with many users**
2. Create 100+ test users
3. Test dashboard loading times
4. Test search and filtering
5. **Test concurrent logins**
6. Simulate multiple simultaneous logins
7. Verify system stability
8. Check response times
9. **Test large datasets**
10. Test with many role assignments

- 11. Test audit log performance
- 12. Test pagination

Expected Results

-  Acceptable response times
 -  System stability under load
 -  Efficient database queries
 -  Proper pagination
-

Test Scenario 10: Security Testing

Objective

Test security measures and vulnerabilities

Test Cases

10.1 Authentication Security

- Test password brute force protection
- Test session hijacking prevention
- Test CSRF protection





10.2 Authorization Security

- Test privilege escalation attempts
- Test direct URL access
- Test API endpoint security

10.3 Data Security

- Test SQL injection prevention
- Test XSS prevention
- Test data sanitization

Expected Results

-  No security vulnerabilities
-  Proper input validation
-  Secure session handling
-  Protected API endpoints

Testing Tools and Setup

Required Tools

- Browser (Chrome/Firefox)
- Database client (pgAdmin/DBeaver)
- Email testing tool (MailHog/Mailtrap)
- Network monitoring (Browser DevTools)

Test Data Setup

```
-- Create test users for different scenarios
INSERT INTO users (email, name, role, provider) VALUES
('testadmin@example.com', 'Test Admin', 'admin', 'google'),
('testvendor@example.com', 'Test Vendor', 'vendor', 'facebook'),
('testcustomer@example.com', 'Test Customer', 'customer',
'line'),
('testpending@example.com', 'Test Pending', 'pending',
'google');
```

Environment Variables for Testing

```
# Test database
DATABASE_URL="postgresql://test_user:test_pass@localhost:5432/
test_sss_surplus"

# Test OAuth (use test apps)
GOOGLE_CLIENT_ID="test_google_client_id"
GOOGLE_CLIENT_SECRET="test_google_client_secret"

# Test email
SMTP_HOST="localhost"
SMTP_PORT="1025"
SMTP_USER="test@example.com"
SMTP_PASS="test_password"

# Super Admin
SUPER_ADMIN_EMAILS="sanchai5651@gmail.com"
SUPER_ADMIN_PASSWORD="Safety17"
SUPER_ADMIN_MODE="true"
NEXT_PUBLIC_SUPER_ADMIN_ENABLED="true"
```



Test Results Template

Test Execution Log

Date: _____
Tester: _____
Environment: _____

Scenario 1: Super Admin Login

- Step 1: ☒/☒
- Step 2: ☒/☒
- Step 3: ☒/☒
- Overall: ☒/☒
- Notes: _____

Scenario 2: New User Social Login

- Step 1: ☒/☒
- Step 2: ☒/☒
- Step 3: ☒/☒
- Overall: ☒/☒
- Notes: _____

[Continue **for** all scenarios...]

Summary:

- Total Scenarios: ____
- Passed: ____
- Failed: ____
- Critical Issues: ____
- Minor Issues: ____