ServiceNow A2A Testing - README

ServiceNow A2A Testing Tools

September 2025

1 Generic ServiceNow A2A Client

A configurable Python client for connecting to any ServiceNow Agent-to-Agent (A2A) endpoint.

1.1 Features

- ✅ **Generic Configuration**: Works with any ServiceNow instance and agent
- ✅ **Multiple Configuration Methods**: Environment variables or direct code configuration
- ✅ **Full A2A Workflow**: Agent discovery, client initialization, and message sending
- ✅ **Error Handling**: Comprehensive error handling and user-friendly messages
- ✅ **Conversation Context**: Support for context and task IDs for conversation continuity
- ✅ **Easy Integration**: Simple API for embedding in other applications

1.2 Quick Start

1.2.1 1. Install Dependencies

```
pip install httpx a2a-sdk
```

1.2.2 2. Configure Your ServiceNow Connection

Option A: Using Environment Variables (Recommended)

Copy the template and fill in your values:

```
cp .env.template .env
# Edit .env with your ServiceNow details
```

Option B: Direct Configuration in Code

```
from generic_a2a_client import ServiceNowA2AClient

client = ServiceNowA2AClient(
    instance="your-instance.service-now.com",
    agent_id="your-agent-id",
```

```
        api_key="your-api-key"
)
```

1.2.3 3. Send a Message

```python
import asyncio
from generic_a2a_client import ServiceNowA2AClient

async def main():
    client = ServiceNowA2AClient()
    response = await client.connect_and_send("What can you help me with?")
    print(response)

asyncio.run(main())
```

1.3 Configuration

1.3.1 Environment Variables

Variable	Description	Example
`SERVICENOW_INSTANCE`	Your ServiceNow instance URL (without https://)	`company.service-now.com`
`SERVICENOW_AGENT_ID`	The ID of the agent you want to connect to	`your-agent-id-here`
`SERVICENOW_TOKEN`	Your ServiceNow API key	`your-api-key-here`

1.3.2 Finding Your Agent ID

1. Log into your ServiceNow instance
2. Navigate to **Agent Intelligence** > **Agents**

3. Find your agent and copy its ID from the URL or agent details

1.3.3 Getting an API Key

1. In ServiceNow, go to **System Web Services** > **REST API Explorer**
2. Or generate one in the ServiceNow Developer Portal
3. Ensure the API key has permissions for A2A operations

1.4 Usage Examples

1.4.1 Basic Usage

```python
import asyncio
from generic_a2a_client import ServiceNowA2AClient

async def basic_example():
    client = ServiceNowA2AClient()
    response = await client.connect_and_send("Hello, agent!")
    return response

asyncio.run(basic_example())
```

1.4.2 Advanced Usage with Context

```python
import asyncio
from generic_a2a_client import ServiceNowA2AClient

async def conversation_example():
    client = ServiceNowA2AClient()

    # Start a conversation
    response1 = await client.connect_and_send("What services do you
        provide?")

    # Continue the conversation with context
```

```python
        context_id = response1.get('result', {}).get('contextId')
        response2 = await client.connect_and_send(
            "Tell me more about the first one",
            context_id=context_id
        )

        return response1, response2

asyncio.run(conversation_example())
```

1.4.3 Step-by-Step Control

```python
import asyncio
import httpx
from generic_a2a_client import ServiceNowA2AClient

async def detailed_example():
    client = ServiceNowA2AClient()

    timeout = httpx.Timeout(30.0, connect=10.0)
    async with httpx.AsyncClient(timeout=timeout) as httpx_client:
        # Step 1: Discover the agent
        agent_card = await client.fetch_agent_card(httpx_client)

        # Step 2: Initialize the A2A client
        a2a_client = client.initialize_client(httpx_client)

        # Step 3: Send multiple messages
        response1 = await client.send_message("First message")
        response2 = await client.send_message("Second message")

        return response1, response2

asyncio.run(detailed_example())
```

1.5 API Reference

1.5.1 ServiceNowA2AClient

1.5.1.1 Constructor

```
ServiceNowA2AClient(instance=None, agent_id=None, api_key=None)
```

If parameters are None, loads configuration from environment variables.

1.5.1.2 Methods

1.5.1.2.1 `connect_and_send(message_text, context_id=None, task_id=None)`

Convenience method to connect and send a message in one call.

1.5.1.2.2 `fetch_agent_card(httpx_client)`

Discover and fetch the agent card from ServiceNow.

1.5.1.2.3 `initialize_client(httpx_client)`

Initialize the A2A client with the discovered agent.

1.5.1.2.4 `send_message(message_text, context_id=None, task_id=None)`

Send a message to the agent.

1.6 Error Handling

The client provides detailed error messages for common issues:

- **Configuration errors**: Missing environment variables
- **Connection errors**: Network or authentication issues
- **Agent errors**: Invalid agent ID or unavailable agent
- **Message errors**: Invalid message format or sending failures

1.7 Files

- `generic_a2a_client.py` - Main generic client implementation
- `.env.template` - Configuration template
- `README.md` - This documentation

1.8 Migration from Specific Implementations

If you have an existing ServiceNow A2A client, you can easily migrate:

1. Replace hardcoded instance/agent values with configuration
2. Use the `ServiceNowA2AClient` class instead of direct implementation
3. Update your `.env` file with the new variable names

1.9 Contributing

Feel free to submit issues and enhancement requests!