ChatGPT

Acceptance Test Specification: MCP–Fess Bridge Server

This specification defines acceptance tests that verify the MCP–Fess bridge server satisfies the functional requirements described in the Functional Requirements Specification (FRS). Each test includes preconditions, steps and expected results. Tests assume MCP revision **2025-03-26** unless otherwise stated. When running with the `--cody` flag, adjust expected protocol versions accordingly.

1 Configuration and Startup

AT-CFG-001 Missing configuration file

- **Preconditions:** The file `<USER_HOME_DIR>/.mcp-feiss/config.json` does not exist.
- **Steps:**
- Start the server with no command-line flags.
- **Expected:**
- The server immediately exits with a non-zero status.
- Error output clearly states that `config.json` is missing and specifies the expected path.

AT-CFG-002 Invalid JSON configuration

- **Preconditions:** A file exists at `<USER_HOME_DIR>/.mcp-feiss/config.json` containing invalid JSON (e.g., `{ bad json }`).
- **Steps:**
- Start the server.
- **Expected:**
- The server exits with a non-zero status.
- Error output identifies the parse error and points to the invalid file.

AT-CFG-003 Directory creation and logging

- **Preconditions:** `~/.mcp-feiss/` exists but the `log/` subdirectory does not.
- **Steps:**
- Place a valid `config.json` in `~/.mcp-feiss/` with minimal required fields.
- Start the server.
- **Expected:**
- The server creates the `log/` directory automatically.
- The server runs without error.

AT-CFG-004 Non-localhost bind default rejection

- **Preconditions:** `config.json` sets `httpTransport.bindAddress` to `0.0.0.0` and does not enable `allowNonLocalhostBind`.
- **Steps:**
- Start the server with `--transport http`.
- **Expected:**

- The server fails fast with an error indicating that non-loopback binding is disabled and `allowNonLocalhostBind` must be set true to allow it.

AT-CFG-005 Debug logging file naming

- **Preconditions:** Valid configuration present.
- **Steps:**
- Start the server with `--debug`.
- Wait until at least two log messages are generated.
- **Expected:**
- A log file named `<Date and Time>_server.log` appears in `~/.mcp-feiss/log/`.
- Each log line begins with `[HH:MM:SS]` indicating elapsed time since server start.

AT-CFG-006 Default transport selection

- **Preconditions:** Valid configuration present.
- **Steps:**
- Start the server without the `--transport` flag.
- **Expected:**
- The server uses stdio transport (no HTTP listener is created).
- The server reads/writes JSON-RPC on stdin/stdout.

2 MCP Lifecycle & Version Negotiation

AT-MCP-001 Initialize/initialized flow

- **Preconditions:** Server is running with valid configuration.
- **Steps:**
- Connect to the server via the selected transport.
- Send an MCP `initialize` request with a supported protocol version (`"2025-03-26"`).
- Receive the `initialize` response.
- Send `notifications/initialized`.
- Then call a tool method (e.g., health).
- **Expected:**
- The server does not process tool calls until the `notifications/initialized` message is received (pre-initialization tool calls should return an error).
- The `initialize` response advertises protocol version `"2025-03-26"` and includes capabilities for tools and resources [1] [2].

AT-MCP-002 Cody revision pinning

- **Preconditions:** Server is started with the `--cody` flag and a valid configuration.
- **Steps:**
- Send an MCP `initialize` request.
- **Expected:**
- The `initialize` response indicates protocol revision `"2024-11-05"`.

3 Tools and Domain Metadata

AT-TOOL-001 Tool listing includes domain block

- **Preconditions:** Server running with a domain configuration (id = `finance`).
- **Steps:**
- Send `tools/list` via MCP [3].
- **Expected:**
- The response lists tools including search, suggest, popular words, labels, health and job progress.
- Each tool entry's `description` contains a **Knowledge Domain** block with the correct domain id, name, description and label filter.
- Tool names include the domain id (e.g., `fess_finance_search`).

AT-TOOL-002 Search tool basic request

- **Preconditions:** Valid configuration and Fess server with accessible index.
- **Steps:**
- Send `tools/call` for `fess_finance_search` with `{ "query": "test" }` [4].
- **Expected:**
- The bridge requests `/api/v1/documents?q=test&fields.label=<label>` from Fess [5].
- The response contains an array of search results with titles, urls, digests and doc_ids.
- The number of results does not exceed the default `pageSize` of 20 [6].

AT-TOOL-003 Search tool pagination and size limits

- **Preconditions:** Fess has more than 30 matching documents.
- **Steps:**
- Call search with `{ "query": "test", "pageSize": 30 }`.
- Next, call search with `{ "query": "test", "pageSize": 150 }`.
- **Expected:**
- In the first call, the bridge requests `num=30` and returns 30 results if available.
- In the second call, the bridge rejects the request because `150` exceeds `limits.maxPageSize` (100) [6] and returns an MCP error explaining the limit.

AT-TOOL-004 Suggest tool

- **Preconditions:** Fess suggestion is enabled.
- **Steps:**
- Call `fess_finance_suggest` with `{ "prefix": "foo", "num": 5 }` [7].
- **Expected:**
- The bridge calls `/api/v1/suggest-words` with `q=foo`, `num=5`, and `label=<domain.labelFilter>` [7].
- The response contains suggestions as returned by Fess.

AT-TOOL-005 Popular words tool

- **Preconditions:** Fess popular words feature is enabled.
- **Steps:**
- Call `fess_finance_popular_words` with no parameters.
- **Expected:**

- The bridge calls `/api/v1/popular-words?label=<label>` [8].
- The response lists popular words.

AT-TOOL-006 Labels tool

- **Preconditions:** Fess server has multiple labels.
- **Steps:**
- Call `fess_finance_list_labels` with no parameters.
- **Expected:**
- The bridge calls `/api/v1/labels` [9].
- The response lists all labels configured in Fess, not just the domain label.

AT-TOOL-007 Health tool

- **Steps:**
- Call `fess_finance_health`.
- **Expected:**
- The bridge calls `/api/v1/health` [10].
- The response indicates status (`green`/`yellow`/`red`) and `timed_out` values [10].

AT-TOOL-008 Job progress tool

- **Preconditions:** A long-running search or content fetch is in progress and returns a jobId.
- **Steps:**
- Call `fess_finance_job_get` with the jobId.
- **Expected:**
- The response includes the job's current `state`, `progress` value, optional `total` and `message`, plus timestamps.

4 Resources

AT-RES-001 Resource listing pagination

- **Preconditions:** Fess search yields more than `limits.maxPageSize` documents.
- **Steps:**
- Call `resources/list` with no cursor.
- Record the returned `nextCursor`.
- Call `resources/list` again with the cursor.
- **Expected:**
- The first response returns up to `maxPageSize` resources and a non-empty cursor [11].
- The second call returns the next page of resources.

AT-RES-002 Resource description contains domain block

- **Steps:**
- Call `resources/list`.
- **Expected:**
- Each resource's `description` includes the Knowledge Domain block (id, name, description, fessLabel).

AT-RES-003 Read document metadata

- **Preconditions:** A valid resource URI `fess://<domainId>/doc/<docId>` exists.
- **Steps:**
- Call `resources/read` with the URI.
- **Expected:**
- The response contains document metadata: title, url, digest, doc_id and timestamp from the Fess result.
- The response includes `doc_id`, `url`, and `hash` fields in metadata.

AT-RES-004 Read full content with chunking

- **Preconditions:** A valid resource URI `fess://<domainId>/doc/<docId>/content>` where the document content is larger than `limits.maxChunkBytes`.
- **Steps:**
- Call `resources/read` with the URI.
- Record `nextCursor` and `isLast` from the response.
- If `isLast` is false, call `resources/read` again with the cursor; repeat until `isLast` is true.
- **Expected:**
- The first response returns the first text chunk and a non-empty `nextCursor`.
- Subsequent calls return additional chunks until `isLast=true`. No chunks are duplicated or omitted.

AT-RES-005 Content acquisition from URL (HTML)

- **Preconditions:** The Fess result for a document does not include full content, but the `url` points to an accessible HTML page.
- **Steps:**
- Call `resources/read` on `fess://<domain>/doc/<docId>/content`.
- **Expected:**
- The bridge fetches the external URL respecting `contentFetch` limits and returns plain-text content converted from HTML.
- If the operation is slow, the bridge returns a jobId immediately; progress can be monitored with `notifications/progress` and `job_get`.

AT-RES-006 Host allowlist enforcement

- **Preconditions:** `contentFetch.allowedHostAllowlist=["example.com"]`.
- **Steps:**
- Attempt to read the content of a document whose `url` host is not `example.com`.
- **Expected:**
- The bridge rejects the request with an error indicating the host is not allowed.

AT-RES-007 PDF conversion disabled by default

- **Preconditions:** The document's `url` returns a PDF (Content-Type `application/pdf`) and `contentFetch.enablePdf` is not set.
- **Steps:**
- Call `resources/read` on the content URI.
- **Expected:**
- The bridge returns an error stating that PDF conversion is disabled and instructs to enable `contentFetch.enablePdf` in the configuration to process PDF content.

5 Long-Running Operations & Progress

AT-ASYNC-001 Progress notifications

- **Preconditions:** A long-running operation is invoked (e.g., large search or content fetch), and the request includes a `progressToken` in `_meta`.
- **Steps:**
- Call the search or read tool with a `progressToken`.
- Observe incoming notifications on the chosen transport.
- **Expected:**
- The server sends `notifications/progress` messages containing the original token and increasing progress values [12] .
- Notifications MAY include `total` and `message`. Progress continues until the job completes or fails.

AT-ASYNC-002 Job polling

- **Preconditions:** A long-running operation returns a `jobId`.
- **Steps:**
- Call `fess_<domainId>_job_get` with the jobId periodically.
- **Expected:**
- The job status transitions through `queued` → `running` → `done` or `failed`.
- `progress` values are non-decreasing. When `state=done`, the associated tool call's final result can be retrieved.

AT-ASYNC-003 Cancellation

- **Preconditions:** A long-running search or content fetch is in progress with request id `123` and a client wishes to cancel it.
- **Steps:**
- Send a `notifications/cancelled` message with `requestId=123` [13] .
- **Expected:**
- The server stops processing the request and frees resources [14] .
- No final response is returned for the cancelled request [14] .

6 Error Handling & Security

AT-ERR-001 Invalid tool parameters

- **Steps:**
- Call the search tool with invalid parameters (e.g., missing `query`, or `pageSize=-5`).
- **Expected:**
- The server returns an MCP error with a clear message explaining the invalid parameters.

AT-ERR-002 Fess error propagation

- **Preconditions:** Induce an error in Fess (e.g., Fess is offline or invalid endpoint).
- **Steps:**
- Call any tool that proxies to Fess.
- **Expected:**

- The server returns an MCP error that includes the HTTP status, Fess endpoint and a sanitized message.

AT-SEC-001 HTTP token enforcement

- **Preconditions:** `httpAuthToken` is set in the configuration.
- **Steps:**
- Start the server in HTTP mode.
- Send an MCP request without the `Authorization` header.
- Send another request with `Authorization: Bearer <token>` using the configured token.
- **Expected:**
- The first request is rejected with HTTP 401.
- The second request is processed normally.

AT-SEC-002 Private network fetch blocked

- **Preconditions:** `contentFetch.allowPrivateNetworkTargets=false`.
- **Steps:**
- Attempt to read content for a document whose `url` resolves to an internal IP (e.g., `http://192.168.0.10`).
- **Expected:**
- The server rejects the request with an error indicating private network targets are not allowed.

AT-SEC-003 Non-localhost HTTP bind explicit opt-in

- **Preconditions:** `security.allowNonLocalhostBind=true` and `httpTransport.bindAddress=0.0.0.0`.
- **Steps:**
- Start the server with `--transport http`.
- **Expected:**
- The server starts and binds to all addresses.
- Without `allowNonLocalhostBind=true`, this would have failed (covered in AT-CFG-004).

---

[1] [11] Resources - Model Context Protocol
https://modelcontextprotocol.io/specification/draft/server/resources

[2] [3] Tools - Model Context Protocol
https://modelcontextprotocol.io/specification/draft/server/tools

[4] [5] [6] Search API
https://fess.codelibs.org/15.4/api/api-search.html

[7] Suggest API
https://fess.codelibs.org/15.4/api/api-suggest.html

[8] Popular Words API
https://fess.codelibs.org/15.4/api/api-popularword.html

[9] Label API
https://fess.codelibs.org/15.4/api/api-label.html

[10] Health API
https://fess.codelibs.org/15.4/api/api-health.html

[12] Progress - Model Context Protocol
https://modelcontextprotocol.io/specification/draft/basic/utilities/progress

[13] [14] Cancellation - Model Context Protocol
https://modelcontextprotocol.io/specification/draft/basic/utilities/cancellation