XBRLStream MVP Product Requirements Document (PRD)

Product Requirements Document: AI-Powered SSM XBRL Filing Automation System

For Malaysia Companies – MBRS 2.0 Compliance

**🎯 1. Executive Summary**

**Problem**

Malaysian companies must file financial statements, annual returns, and exemption applications with SSM using XBRL format under MBRS 2.0. Doing this manually via SSM’s mTool is:

* Time-consuming (4–10 hours per filing)
* Error-prone (concept mapping, context errors)
* Requires financial expertise

**Solution**

An AI-powered automation system that:

1. Takes a PDF financial statement or annual return
2. Extracts data (numbers + text)
3. Maps to correct SSMxT\_2022v1.0 taxonomy concepts
4. Generates valid XBRL instance
5. Validates against SSM business rules
6. Outputs SSM-ready ZIP file

✅ Fully compliant with SSM MBRS 2.0  
✅ Supports all filing types: FS-MFRS, FS-MPERS, AR1–AR4, EA1–EA7, BNM, CLBG, FC  
✅ Built-in evaluation to prevent regressions

**👥 2. Target Users**

|  |  |
| --- | --- |
| Company Secretary | File SSM documents accurately, on time |
| Finance Officer | Automate XBRL tagging without mTool |
| Accounting Firm | Process 100s of client filings faster |
| Developer | Build/maintain a production-grade AI system |

**🧩 3. Scope & Use Cases**

**✅ In Scope**

|  |  |  |
| --- | --- | --- |
| FS-MFRS | Private/Public | MIHCM Asia Sdn Bhd |
| FS-MPERS | Private | ABC Trading Sdn Bhd |
| AR1 | Companies with share capital | XYZ Holdings Bhd |
| AR2 | Companies without share capital | Charity Org Sdn Bhd |
| AR3 | Foreign Companies | Singapore Branch Sdn Bhd |
| EA2 | Exemption apps | Startup applying for audit exemption |
| FS-BNM | Banks | Bank ABC Malaysia |
| FS-CLBG | Companies Ltd by Guarantee | Professional Body CLBG |

**❌ Out of Scope**

* Inline XBRL (iXBRL) generation
* SSM portal integration (uploading)
* Non-Malaysian filings

**📥 4. Input & Output**

**Input**

* PDF file of financial statement / annual return / exemption app  
  *(e.g., MIHCM Asia Sdn Bhd 31.3.2024.pdf)*

**Output**

* ZIP file ready for SSM upload
  + FS-MFRS\_201401019662\_20240331.zip
  + Contains: FS-MFRS\_201401019662\_20240331.xml

📌 Real-World Example:  
Input: MIHCM Asia Sdn Bhd 31.3.2024.pdf  
Output: ZIP matching SSM\_FS-MFRS\_201401019662\_20240331.xml.txt

**🧠 5. Core Requirements**

**5.1 Functional Requirements**

|  |  |  |
| --- | --- | --- |
| FR-01 | Parse PDF layout (tables, text, headers) | Use Docling to extract P&L table |
| FR-02 | Detect filing type (FS-MFRS, AR1, etc.) | “Annual Return” → AR1 |
| FR-03 | Extract company info (Reg No, Name, Status) | Reg No:201401019662 |
| FR-04 | Extract numeric facts (Revenue, P/L, Assets) | Revenue:4,787,455 |
| FR-05 | Extract narrative text (Auditor’s Report) | Full HTML text block |
| FR-06 | Map to correct taxonomy concept | “Revenue” →ifrs-full:Revenue |
| FR-07 | Generate correct XBRL contexts | fromto\_20230401\_20240331 |
| FR-08 | Validate against SSM formula rules | Block if Revenue missing |
| FR-09 | Output SSM-compliant ZIP | FS-MFRS\_<RegNo>\_<Date>.zip |
| FR-10 | Evaluate accuracy vs gold standard | Concept accuracy ≥98% |

**5.2 Non-Functional Requirements**

|  |  |
| --- | --- |
| NFR-01 | Accuracy: ≥98% concept mapping accuracy |
| NFR-02 | Reliability: Auto-fallback to human review if eval <98% |
| NFR-03 | Scalability: Support 100+ filings/day |
| NFR-04 | Maintainability: Modular design (add new filing types easily) |
| NFR-05 | Auditability: Log all decisions (why concept X was chosen) |

**🏗️ 6. System Architecture**

**Key Components**

|  |  |
| --- | --- |
| Docling Parser | Extract structured data from PDF |
| Filing Type Router | Classify PDF → FS-MFRS, AR1, etc. |
| Filing Agent | Per-type logic (concept map, validation) |
| XBRL Generator | Build valid XBRL instance |
| Arelle Validator | Run SSM formula rules |
| Evaluator | Compare vs gold standard |
| Human Review | Fallback for low-confidence cases |

**🗃️ 7. Database Design (Optional – For Scale)**

For MVP: Use file-based storage (JSON/CSV).  
For enterprise: Use database.

**Tables**

filing\_jobs

|  |  |  |
| --- | --- | --- |
| id | UUID | Job ID |
| pdf\_path | TEXT | Input PDF path |
| filing\_type | TEXT | FS-MFRS, AR1, etc. |
| reg\_no | TEXT | Company Reg No |
| status | ENUM | pending, success, failed, human\_review |
| output\_zip | TEXT | Output ZIP path |
| created\_at | DATETIME | Timestamp |

Gold\_standards

|  |  |  |
| --- | --- | --- |
| id | UUID |  |
| reg\_no | TEXT |  |
| filing\_type | TEXT |  |
| expected\_facts | JSON | List of {concept, value, context} |
| pdf\_hash | TEXT | SHA256 of PDF |

evaluation\_logs

|  |  |  |
| --- | --- | --- |
| job\_id | UUID | FK to filing\_jobs |
| concept\_accuracy | FLOAT | 0.0–1.0 |
| numeric\_error\_rate | FLOAT |  |
| passed | BOOLEAN |  |
| timestamp | DATETIME |  |

**🧪 8. Evaluation Framework**

**Why Evaluate?**

* Prevent silent regressions (e.g., “Profit” → wrong concept)
* Ensure compliance (SSM rejects invalid XBRL)
* Enable continuous improvement

**Metrics That Matter**

|  |  |  |
| --- | --- | --- |
| Concept Accuracy | ≥98% | % of facts mapped to correct concept |
| Mandatory Coverage | 100% | % of\*fields filled |
| Numeric Fidelity | ≤1% error | Relative error in numbers |
| TextBlock Faithfulness | ≥95% | Exact match or BERTScore ≥0.95 |

**Gold Standard Example (**MIHCM**)**

json

{

"reg\_no": "201401019662",

"filing\_type": "FS-MFRS",

"facts": [

{

"concept": "ifrs-full:Revenue",

"value": 4787455,

"context": "fromto\_20230401\_20240331",

"unit": "MYR"

},

{

"concept": "ifrs-full:ProfitLoss",

"value": -723253,

"context": "fromto\_20230401\_20240331",

"unit": "MYR"

},

{

"concept": "ssmt-dei:TypeOfSubmission",

"value": "FS-MFRS",

"context": "fromto\_20230401\_20240331"

}

]

}

**🤖 9. AI Code Generation Prompts**

Use these prompts with Claude Code, Lovable, or GitHub Copilot

**Prompt 1: Docling PDF Parser**

You are a Python developer building an SSM XBRL automation system.

Write a function using Docling to parse a PDF financial statement.

Output: structured JSON with tables, text, and page numbers.

Include error handling.

**Prompt 2: Filing Type Router**

text

You are a Malaysian company secretary.

Write a Python function that takes PDF text and returns filing type:

- "FS-MFRS" if contains "Revenue", "Profit", "MFRS"

- "AR1" if contains "Annual Return", "Directors", "Shareholders"

- "EA2" if contains "Exemption Application"

Use keyword matching + simple rules.

**Prompt 3: Concept Mapper**

You are an XBRL expert.

Write a Python function that maps a PDF label (e.g., "Total revenue")

to an SSMxT\_2022 concept (e.g., "ifrs-full:Revenue").

Use a CSV lookup file with columns: label\_en, concept\_id.

Handle case-insensitive and partial matches.

**Prompt 4: XBRL Generator**

You are an XBRL developer.

Write a Python function that builds an XBRL instance document.

Inputs: list of facts [{concept, value, context, unit}], reg\_no, dates.

Output: lxml Element tree.

Include:

- Correct namespaces (ifrs-full, ssmt-dei, etc.)

- Contexts: asof\_YYYYMMDD, fromto\_YYYYMMDD\_YYYYMMDD

- Units: MYR, shares, pure

**Prompt 5: Evaluator**

You are a QA engineer.

Write a Python function that compares predicted XBRL facts

against a gold standard JSON.

Return:

- concept\_accuracy: % of correct concepts

- numeric\_error\_rate: avg relative error

Assume facts are lists of {concept, value, context}.

**📊 10. Monitoring & Dashboard**

**Streamlit Dashboard (**dashboard/app.py**)**

* Show real-time metrics: concept accuracy, coverage
* List recent jobs: success/fail
* Alert if accuracy <98%

python

import streamlit as st

st.metric("Concept Accuracy", "99.2%", delta="+0.5%")

st.metric("Mandatory Coverage", "100%")

if accuracy < 0.98:

st.error("⚠️ Action required!")

**🚀 11. Deployment & CI/CD**

**GitHub Actions (**/.github/workflows/ci.yml**)**

yaml

name: CI with Eval Gates

on: [push]

jobs:

test:

runs-on: ubuntu-latest

steps:

- uses: actions/checkout@v4

- name: Install deps

run: pip install -r requirements.txt

- name: Run end-to-end test (MIHCM)

run: python tests/test\_mihcm\_end\_to\_end.py

- name: Block if accuracy < 98%

run: |

ACC=$(python -c "from evaluator import get\_accuracy; print(get\_accuracy())")

if (( $(echo "$ACC < 0.98" | bc -l) )); then exit 1; fi

**📁 12. Project Structure**

ssm-xbrl-ai/

├── input/ # Test PDFs

├── gold\_standard/ # Expected outputs

├── taxonomy/ # Concept maps per filing type

├── src/

│ ├── docling\_parser.py

│ ├── filing\_router.py

│ ├── agents/ # FS-MFRS, AR1, etc.

│ ├── xbrl\_generator.py

│ └── evaluator.py

├── tests/

│ └── test\_mihcm\_end\_to\_end.py

├── dashboard/

│ └── app.py

├── .github/workflows/ci.yml

└── main.py

**✅ 13. Success Criteria**

|  |  |
| --- | --- |
| MIHCM End-to-End Test Passes | Generates ZIP matching SSM XML |
| Supports 5+ Filing Types | FS-MFRS, FS-MPERS, AR1, AR2, EA2 |
| Concept Accuracy ≥98% | Eval vs gold standard |
| CI/CD Blocks on Fail | GitHub Actions fails if accuracy <98% |
| Human Review Fallback | Low-confidence jobs routed to human |

**📎 Appendix A: Real-World Example (MIHCM)**

**Input PDF Snippet**

Statement of Profit or Loss

Revenue: 4,787,455

Cost of sales: (1,615,530)

Profit (loss): (723,253)

**Expected XBRL Facts**

|  |  |  |  |
| --- | --- | --- | --- |
| ifrs-full:Revenue | 4787455 | fromto\_20230401\_20240331 | MYR |
| ifrs-full:CostOfSales | -1615530 | fromto\_20230401\_20240331 | MYR |
| ifrs-full:ProfitLoss | -723253 | fromto\_20230401\_20240331 | MYR |

**Output ZIP**

* FS-MFRS\_201401019662\_20240331.zip
  + Contains: FS-MFRS\_201401019662\_20240331.xml

**📎 Appendix B: SSMxT\_2022 Taxonomy Structure**

|  |  |
| --- | --- |
| FS-MFRS | ssmt-fs-mfrs\_2022-12-31\_entry\_point.xsd |
| FS-MPERS | ssmt-fs-mpers\_2022-12-31\_entry\_point.xsd |
| AR1 | ssmt-ar1\_2022-12-31\_entry\_point.xsd |
| EA2 | ssmt-ea2\_2022-12-31\_entry\_point.xsd |

📌 Key Insight: Each filing type has its own concept map CSV.

✅ This PRD is ready for AI code generation and developer implementation.