Streamlined Decision Support for Rheumatoid Arthritis Management for Clinicians

INFO-B 642 CLINICAL DECISION SUPPORT SYSTEMS

Table of Contents

Abstract **6**

1. Introduction **6**
1.1 Background and Motivation 6
1.2 Clinical Need Addressed 7
1.3 Objective of the Project 7
1.4 Target User Definition 7

2. Literature Review and Related Clinical Guidelines **8**
2.1 Literature Review ..... **Error! Bookmark not defined.**
2.2 Clinical guidelines and justification for the CDSS rules 9

3. Methodology ..... Error! Bookmark not defined.
3.1 Project management approach ..... **Error! Bookmark not defined.**
3.2 Team Roles and Responsibilities ..... **Error! Bookmark not defined.**
3.3 Project Phases and Key Milestones ..... **Error! Bookmark not defined.**
3.4 Tools and Processes for Collaboration ..... **Error! Bookmark not defined.**
3.5 Communication Strategy ..... **Error! Bookmark not defined.**
3.6 Critical Reflection ..... **Error! Bookmark not defined.**

4. System Design and Description **9**
4.1 System Architecture 9
4.2 User interaction 10
4.3 Cognitive Considerations 10

5. Discussion **10**
5.1 Clinical Relevance and Effectiveness 10
5.2 Interoperability Considerations 10
5.3 Legal and Ethical Considerations 11
5.4 Privacy and Data Protection 11

6. Limitations and Shortcomings **11**
6.1 Clinical and Data Scope Limitations 11
6.2 Ethical, Legal, and Privacy Limitations 11
6.3 Usability and Cognitive Design Limitations 11
6.4 Project Scope and Implementation Limitations 12

7. Future Directions **12**

8. Conclusion **12**

References **12**

Appendices **16**

Glossary of Terms

- Access token: Time-limited credential issued (e.g., via OAuth 2.0) that authorizes API calls.
- ACR (American College of Rheumatology): U.S. professional society that publishes evidence-based RA treatment guidelines.

- Agile: Iterative project-management approach emphasizing short cycles and continuous feedback.
- Alert fatigue: Desensitization to frequent alerts, leading to ignored or delayed responses.
- API (Application Programming Interface): Rules that let software systems communicate with each other.
- AUROC (Area Under the Receiver Operating Characteristic Curve): Classifier performance metric probability a model ranks a random positive higher than a random negative (0.5–1.0).
- bDMARD (Biologic Disease Modifying Antirheumatic Drugs): Biologic agents (e.g., TNF inhibitors) that modify RA disease processes.
- CDAI (Clinical Disease Activity Index): RA activity score using tender/swollen joint counts and patient/physician global assessments.
- CDSS (Clinical Decision Support System): Software that provides patient-specific, evidence-based recommendations to clinicians.
- Cognitive load: Mental effort required to process information and make decisions.
- Comorbidity: A co-existing medical condition in a patient with a primary disease.
- Condition (FHIR): FHIR resource representing a problem/diagnosis (e.g., RA).
- D2T RA (Difficult-to-Treat Rheumatoid Arthritis): RA with persistent activity or symptoms despite multiple DMARDs.
- DAS28 (Disease Activity Score in 28 joints): Composite RA activity score using 28-joint counts and inflammatory markers/symptoms.
- DMARD (Disease-Modifying Antirheumatic Drug): Drug class that slows RA progression
- EHR (Electronic Health Record): Digital longitudinal patient chart used in care delivery.
- Encryption at rest: Cryptographic protection of stored data on disk/media.
- Encryption in transit: Cryptographic protection of data sent over networks
- EULAR (European Alliance of Associations for Rheumatology): European rheumatology society issuing RA guidance.
- FHIR (Fast Healthcare Interoperability Resources): HL7 standard for exchanging healthcare data via modular “resources” and RESTful APIs.
- FHIR profile: A constrained/extended definition of a FHIR resource for a specific use case.
- Flare: Clinically meaningful worsening of RA disease activity.
- Glucocorticoid: Corticosteroid anti-inflammatory (e.g., prednisone); use should be minimized in RA.
- GRADE (Grading of Recommendations Assessment, Development and Evaluation): Framework for rating certainty of evidence and strength of recommendations.
- HCQ (Hydroxychloroquine): csDMARD often preferred in low-activity RA.
- HIPAA (Health Insurance Portability and Accountability Act): U.S. law governing privacy/security of health information.
- HL7 (Health Level Seven International): Standards-development organization behind FHIR and other health IT standards.
- HTTPS (Hypertext Transfer Protocol Secure): HTTP over TLS for encrypted web communication.

- Interoperability: Ability of different systems to exchange and use information reliably.
- JSON (JavaScript Object Notation): Lightweight text format for data exchange used by many APIs
- Kanban: Visual workflow method using “To Do / In Progress / Done” boards to manage work in progress.
- MedicationStatement (FHIR): FHIR resource documenting a patient’s reported medication use.
- Methotrexate (MTX): Anchor csDMARD and first-line therapy for many moderate/high-activity RA cases.
- OAuth 2.0: Authorization framework enabling delegated access (e.g., SMART on FHIR app to EHR data).
- Observation (FHIR): FHIR resource for measurements/assessments
- PHI (Protected Health Information): Individually identifiable health information regulated by HIPAA.
- Point-of-care: The time/place where clinical care is delivered (e.g., during a visit).
- RA (Rheumatoid Arthritis): Chronic autoimmune inflammatory arthritis affecting synovial joints.
- RCM (Rheuma Care Manager): Prototype RA CDSS that predicts flare risk to support treatment decisions.
- RheumaTool: Diagnostic CDSS that generates differential diagnoses for rheumatic conditions.
- Rules engine: Software that applies declarative “if–then” logic to data to produce outcomes/recommendations.
- Sensitivity: Proportion of true positives correctly identified by a test/model (true-positive rate).
- Shared decision-making: Collaborative process where clinicians and patients choose care based on evidence and patient preferences.
- SMART on FHIR (Substitutable Medical Applications, Reusable Technologies on FHIR): Open standards enabling apps to launch inside EHRs and securely access FHIR data via OAuth 2.0.
- SNOMED CT (Systematized Nomenclature of Medicine-Clinical Terms): Comprehensive clinical terminology used for coding problems/diagnoses.
- Specificity: Proportion of true negatives correctly identified by a test/model (true-negative rate).
- T2T (Treat-to-Target): Strategy of adjusting therapy to reach remission/low disease activity with regular measurement.
- TLS (Transport Layer Security): Protocol that secures data in transit over networks.
- TNF (Tumor Necrosis Factor): Pro-inflammatory cytokine targeted by several RA biologics.
- tsDMARD (Targeted Synthetic DMARD): Small-molecule DMARDs (e.g., JAK inhibitors) targeting specific pathways.
- WHO (World Health Organization): UN health agency providing global health guidance.

Abstract

We built a rule-based clinical decision support tool for rheumatoid arthritis (RA) in Python, integrated with a HAPI FHIR server to apply the 2021 ACR guideline at the point of care. Most EHRs lack RA-specific decision support, so clinicians must process 44 recommendations during visits. The tool pulls FHIR data (Conditions, Observations, MedicationStatement), identifies RA using SNOMED CT 69896004, and returns patient-specific options with strength and evidence notes to support treat-to-target care and limit steroid use. Primary users are rheumatologists and primary care clinicians. Unlike RheumaTool and Rheuma Care Manager,our tool provides guideline-based treatment suggestions. Built with Agile/Kanban, the pipeline fetches data, stores JSON, extracts severity, applies rules, and outputs concise results to reduce cognitive load. It leverages FHIR for interoperability but must handle vendor and terminology differences, so clinicians remain final decision-makers. For security, production use requires SMART on FHIR/OAuth 2.0 and encryption; plaintext storage is still a known risk. Some current limitations include simplified inputs, brittle matching, stand-alone workflow, static rules, and no feedback. In future we will add richer FHIR inputs, semantic normalization, a SMART on FHIR UI, an external rules engine, and a learning/ feedback loop.

Keywords: Clinical Decision Support System, Rheumatoid Arthritis, ACR 2021 guideline, Disease-Modifying Antirheumatic Drugs (DMARDs), Methotrexate, Hydroxychloroquine, HAPI FHIR, EHR interoperability, SMART on FHIR, OAuth 2.0, HIPAA, rule-based engine, CDAI, DAS28, semantic normalization, RheumaTool, Rheuma Care Manager, flare prediction (AUROC), glucocorticoids, primary care, rheumatology, JSON, API.

1. Introduction

1.1 Background and Motivation

A clinical decision support system (CDSS) is software designed to assist healthcare professionals in making patient-specific decisions by integrating a clinical knowledge base with individual patient data to generate tailored recommendations (Sutton et al., 2020). CDSSs improve healthcare quality, efficiency, and consistency by providing evidence-based, point-of-care guidance (Sutton et al., 2020).

Rheumatoid arthritis requires early diagnosis and timely DMARD initiation to prevent irreversible joint damage and disability (Tanaka, 2021; WHO, 2023). The 2021 ACR guideline provides 44 evidence-based recommendations for DMARD initiation, escalation, and prudent glucocorticoid use, stratified by disease activity, treatment history, and comorbidities (Fraenkel et al., 2021). Most EHRs lack RA-specific treatment decision support, forcing clinicians to manually process lengthy guidelines during visits. This is a time-consuming process prone to omissions (Labinsky et al., 2023).

A Python-based, HAPI FHIR–integrated CDSS can address this gap by ingesting patient data directly from FHIR resources (e.g., Conditions, Observations, MedicationStatements) and generating real-time, patient-specific treatment recommendations with strength-of-recommendation and certainty-of-evidence annotations. Embedding these recommendations into clinical workflows promotes treat-to-target (T2T) strategies, ensures more consistent adherence to guidelines, and reduces inappropriate glucocorticoid use (Fraenkel et al., 2021).

1.2 Clinical Need Addressed

The primary problem is the lack of a dedicated tool to automatically deliver ACR guideline–compliant RA treatment recommendations at the point of care. Treatment selection is complicated by variability in patient response, potential side effects, and the challenge of safely tapering therapy after remission (Tanaka, 2021; Precision Rheumatology, 2025). Many patients undergo prolonged trial-and-error therapy cycles, experiencing delayed disease control, treatment discontinuations due to adverse events, and barriers to accessing newer medications (Precision Rheumatology, 2025). Currently, there is no widely adopted CDSS that integrates disease activity, prior DMARD exposure, and glucocorticoid use to generate individualized treatment recommendations aligned with ACR guidelines, representing a significant gap in RA management.

The proposed CDSS addresses these challenges by:

- Automating the mapping of patient conditions, disease activity level, and treatment history from HAPI FHIR data to relevant ACR 2021 recommendations (Fraenkel et al., 2021).
- Presenting concise, evidence-based treatment options without requiring the clinician to review the entire guideline in real time.
- Ensuring consistent adherence to guideline-driven care, even when clinician familiarity with the latest ACR updates is variable.
- Facilitating shared decision-making by combining evidence-based suggestions with patient preferences (Messelink et al., 2024).

1.3 Objective of the Project

The main goal is to develop a pure Python, rule-based Clinical Decision Support System integrated with HAPI FHIR that delivers real-time, patient-specific RA treatment recommendations in line with the ACR 2021 guideline.

Specific objectives include:

1. Encoding corresponding ACR RA recommendations into a structured Python ruleset, optimized for minimal branching logic.
2. Retrieving patient conditions, medications, and relevant comorbidities via HAPI FHIR API calls and generating tailored recommendations with strength (strong/conditional) and certainty of evidence annotations (Fraenkel et al., 2021).
3. Enabling clinicians to review recommendations alongside key patient data to support shared decision-making and cost effectiveness for patients.

1.4 Target User Definition

Primary users:

- Rheumatologists managing RA patients.
- Primary care physicians initiating RA treatment in early stages.

Secondary users:

- Nurse practitioners and physician assistants in rheumatology practices.
- Clinical pharmacists managing DMARD therapy and safety monitoring.

The CDSS will be deployed within EHR systems connected to a HAPI FHIR server. It is intended for use during outpatient visits to guide initiation, adjustment, or continuation of DMARD therapy

according to disease activity and treatment history, enhancing adherence to guideline-driven, patient-centered RA care (Messelink et al., 2024).

2. Literature Review and Gaps in Existing Tools

2.1 Existing tools and evidence

RheumaTool is a diagnostic tool for rheumatic diseases that uses structured patient input to generate a differential diagnosis list (Alder et al., 2020). In a retrospective validation study, it correctly identified the main diagnosis in 40% of cases and included the correct diagnosis in the differential list in 63.8% of cases, demonstrating moderate diagnostic support potential (Alder et al., 2020). However, this tool is focused exclusively on diagnosis and does not provide treatment recommendations for rheumatoid arthritis (RA), representing a critical gap for therapeutic decision support.

In contrast, the Rheuma Care Manager (RCM) is a prototype CDSS developed to aid treatment decisions in RA through an AI-powered flare risk prediction tool (Labinsky et al., 2023). The RCM integrates longitudinal patient data, including disease activity, medication history, and laboratory values, and predicts flare risk under current therapy versus dose reduction scenarios. In a pilot study, the model achieved 72% sensitivity, 76% specificity, and an AUROC of 0.80, and its use increased physician decision confidence and reduced variability between clinicians (Labinsky et al., 2023).

Evidence from other domains confirms that CDSSs improve care processes such as medication dosing and preventive care when effectively integrated into workflows (Labinsky et al., 2023). In RA, best practices emphasize treat-to-target (T2T), timely escalation for active disease, and careful tapering in sustained remission (Labinsky et al., 2023; Takanashi & Kaneko, 2024). The D2T RA subset which is affecting 6–28% of RA patients, is characterized by inadequate response to multiple biologic or targeted synthetic DMARDs, persistent disease activity, and high management complexity (Takanashi & Kaneko, 2024). EULAR guidelines recommends individualized strategies, but real-world implementation is hindered by the absence of precision-guided tools that can stratify patients and suggest optimized regimens (Takanashi & Kaneko, 2024). Despite these needs, there is no presence of widely adopted CDSS that combines disease activity, prior DMARD exposure, and glucocorticoid use to deliver point-of-care treatment recommendations.

2.2 Gaps and Limitations

1. Existing RA CDSS tools are either diagnostic (RheumaTool) or focus on flare RCM without fully operationalizing guideline-based treatment pathways integrating DMARD history, disease activity, and glucocorticoid tapering criteria.
2. There is no documented system that provides real-time, patient-specific treatment recommendations aligned with ACR/EULAR T2T principles for both typical and D2T RA cases.
3. Current tools show limited external validation and lack robust evidence for impact on long-term clinical outcomes, underscoring the need for expanded datasets, prospective trials, and integration into clinical practice environments.
4. This project aims to address these gaps by developing a Python-based CDSS that integrates disease activity scores, medication exposure history, and glucocorticoid use to generate treatment recommendations grounded in ACR 2021 guidelines.

3. Clinical guidelines and justification for the CDSS rules

3.1 Clinical guideline used

The rules in the RA CDSS are grounded in the 2021 American College of Rheumatology (ACR) Guideline for the Treatment of Rheumatoid Arthritis, which specifies first-line DMARD selection, escalation sequences after inadequate response, glucocorticoid minimization, and cautious tapering after sustained remission (Fraenkel et al., 2021).

3.2 How the guidelines and literature justify the CDSS design

We encoded activity-stratified rules because the 2021 ACR guideline organizes treatment by low, moderate, and high disease activity, preferring HCQ in low activity and anchoring therapy on MTX once activity is moderate or high (Fraenkel et al., 2021). Escalation after optimized MTX favors adding a biologic or targeted synthetic DMARD rather than routine triple csDMARD, with minimization of chronic glucocorticoids and T2T reassessment at 4–12 weeks (Fraenkel et al., 2021; Tanaka, 2021). Using these rules in a point-of-care CDSS addresses known workflow barrier and improves adherence to evidence-based steps (Sutton et al., 2020). Detailed rules are outlined in Table A1 of the Appendix.

3.3 Functionality of the guideline for our CDSS

The ACR pathway incorporates branching logic based on disease activity, prior DMARD exposure, and steroid use, which is well-suited for rule-based encoding and point-of-care surfacing within an EHR via FHIR resources, thereby improving guideline adherence and decision consistency under time pressure (Sutton et al., 2020). Patient-facing qualitative research also indicates that RA patients generally support prediction or decision tools when these tools are transparent, science-based, and used to support, rather than replace, shared decision-making (Messelink et al., 2025). Our design is accommodated by presenting concise options that clinicians review with patients during the appointment (Messelink et al., 2025).

4. System Design and Description

4.1 System Architecture

The system operates through a sequence of five distinct components that handle data from its source to the final output of the recommendation. The architecture is designed to be modular and straightforward, beginning with a data fetching script that connects to the HAPI FHIR server and ending with the output of a treatment recommendation. A detailed diagram illustrating this data flow is available in **Appendix E**. The complete implementation of all system components, including the data fetching scripts, rule processing modules, and user interface code, is available in our GitHub repository (see **Appendix F** for repository link and setup instructions).

- **HAPI FHIR Server (Data Source)**: The system connects to the HAPI FHIR public server, which stores standardized healthcare data. Patient and condition records for Rheumatoid Arthritis (SNOMED code: 69896004) are queried using the FHIR API.
- **Data Fetching Script**: Two primary Python functions such as fetch_patient() and get_condition_with_details() are used to handle API requests to retrieve patient details and their associated RA condition details.
- **Local JSON Storage**: Retrieved data is stored in structured JSON files (patients.json, conditions.json) inside the data directory.

- **Data Extraction & Rule Evaluation**: The function extract_codes_and_severity() reads conditions.json, extracts diagnosis codes and severity, and matches them against a set of RA-specific treatment rules derived from clinical guidelines.
- **Recommendation Output**: If a match is found, a recommendation is generated and printed in the console. It is also appended to recommendations.txt for record-keeping and later reference.

4.2 User interaction

This CDSS is designed to assist clinicians in determining treatment recommendations for patients with RA. The user initiates the process by running the Python script, which fetches patient and condition data from the HAPI FHIR server. The system then extracts diagnosis codes and severity levels, applies pre-defined RA treatment rules, and generates recommendations. These recommendations are both displayed in the console for immediate viewing and saved into a recommendations.txt file for record-keeping.

4.3 Cognitive Considerations

The CDSS was designed to keep cognitive load low by presenting only concise, relevant treatment recommendations for RA. Each recommendation is paired with the diagnosis code and severity to support quick validation without extra searching. By showing only condition-specific guidance and using a clear, consistent format, the system avoids information overload and helps clinicians make decisions faster and with greater confidence.

5. Discussion

5.1 Clinical Relevance and Effectiveness

This CDSS directly addresses the clinical challenge of applying the ACR's 44 recommendations during busy patient visits (Fraenkel et al., 2021). Our Python-based rule engine programmatically encodes these guidelines, ingesting patient data in real-time from the FHIR Condition resource to automate a process clinicians perform manually. By providing immediate, point-of-care recommendations based on the latest FHIR data, the system is designed to enhance adherence to the treat-to-target (T2T) strategy, which is fundamental for preventing long-term joint damage (Stoffer et al., 2016; Schipper et al., 2010). The system’s output—concise, evidence-based options—facilitates shared decision-making and fills a critical functionality gap in most EHRs, which lack embedded, RA-specific decision support.

5.2 Interoperability Considerations

The architecture's reliance on the HL7 FHIR standard is a strategic choice for interoperability, enabling our get_condition_with_details() function to retrieve data from any compliant EHR (Bender & Sartipi, 2013). However, real-world deployment would require validating our scripts against vendor-specific FHIR profiles to overcome known implementation inconsistencies (Ayaz et al., 2021; Lehne et al., 2019). Semantic interoperability is equally crucial. Our extract_codes_and_severity() function's reliance on the specific SNOMED CT code for RA (69896004) highlights this dependency; without this precise semantic match within the FHIR resource, the rule-based engine would fail to trigger, underscoring the need for robust data mapping.

5.3 Legal and Ethical Considerations

The CDSS is designed as an assistive tool, not a replacement for clinical judgment, which aligns with legal principles that keep the clinician as the final decision-maker (Prictor, 2023; Fraenkel et al., 2021). Our system reinforces this by presenting recommendations annotated with evidence strength, empowering clinicians to apply their professional expertise. Ethically, its rule-based nature provides critical transparency. Unlike "black box" models, its logic is directly traceable to the published ACR guideline, as our Python engine directly encodes rules from treatment_recommendations.txt, allowing for easy auditing (Hassija et al., 2024). The system inherits any demographic biases from the guideline, so our design supports clinical oversight by displaying its recommendations alongside key patient data (e.g., severity) retrieved from FHIR, allowing clinicians to identify when a patient's unique profile may warrant deviation.

5.4 Privacy and Data Protection

As the system processes PHI, HIPAA compliance is non-negotiable (Sutton et al., 2020). A production-ready version of our Python script would operate as a registered SMART on FHIR client, using OAuth 2.0 to ensure secure, authorized data access. While data in transit is protected via HTTPS/TLS, the prototype’s most critical limitation is its current storage of PHI in unencrypted local files (patients.json, conditions.json, recommendations.txt). This method is not HIPAA-compliant and represents a significant security vulnerability. A real-world deployment would require a fundamental redesign to either process data entirely in-memory or use a secure, encrypted database, eliminating this unacceptable risk.

6. Limitations and Shortcomings

6.1 Clinical and Data Scope Limitations

The system's primary limitation is its oversimplification of RA decision-making. Its rule engine relies solely on a single, text-based "severity" label from the FHIR Condition resource, which contrasts with the multifaceted assessment mandated by the ACR 2021 guideline. It currently lacks crucial data points, including comorbidities, detailed treatment history (e.g., prior DMARD failures), and validated disease activity scores like DAS28 and CDAI. This prevents the system from executing the complex, branching logic of the full 44 ACR recommendations, limiting its utility to a simplified subset of rules. Furthermore, its dependence on exact string matching for severity text makes the system brittle, causing it to fail on valid patient records with only minor terminology variations.

6.2 Ethical, Legal, and Privacy Limitations

The current prototype has significant vulnerabilities. Its most critical privacy limitation is the storage of PHI in unencrypted local files (patients.json, recommendations_output.txt). This non-HIPAA-compliant methodology presents a major security risk that would be unacceptable in any real-world deployment. Ethically, the system currently lacks a mechanism to mitigate the inherent biases of its source clinical guideline, risking recommendations that are unsuitable for underrepresented patient populations.

6.3 Usability and Cognitive Design Limitations

The CDSS currently operates as a standalone Python script, entirely disconnected from the clinical workflow. Its recommendations are delivered to a console or a text file, requiring the clinician to

interrupt their work, run a separate program, and manually transfer the information back into their thought process or the EHR. This disjointed design creates a significant usability barrier, increases the user's cognitive load, and severely limits the likelihood of the tool's adoption in a time-constrained clinical setting. An effective CDSS must be seamlessly integrated at the point of care to provide non-disruptive, contextually relevant guidance.

6.4 Project Scope and Implementation Limitations

The project's scope led to key implementation limitations. The rule base is hardcoded in a text file (treatment_recommendations.txt), making it static and difficult to update as guidelines evolve. A more scalable approach would use an external rules engine that clinical experts could manage without requiring code changes. Additionally, the system lacks feedback or learning mechanism to track clinician agreement and patient outcomes. This absence prevents the system from being validated, refined, or improved over time, limiting its long-term accuracy and relevance.

7. Future Directions

To evolve this prototype into a clinically integrated tool, the next phase will enhance its clinical sophistication and usability. A foundational step is to expand data ingestion beyond a simple severity label to include richer FHIR resources, such as Observation for CDAI/DAS28 scores, MedicationStatement for treatment history, and Condition for comorbidities. This will be supported by a robust semantic normalization layer to overcome the current string-matching brittleness and interpret data more accurately (Rosenbloom et al., 2011). To embed the tool into clinical workflows, the system will be re-architected as a SMART on FHIR application, launching seamlessly from within an EHR. The current text output will be replaced with a user-centered graphical interface designed to present recommendations concisely, mitigating cognitive load and alert fatigue (Ancker et al., 2017). For long-term scalability, the hardcoded logic will be externalized into a dedicated rules engine, allowing clinical experts to update guidelines without software engineering intervention (Papadopoulos et al., 2022). Finally, a feedback mechanism will be introduced to create a continuous quality improvement cycle, enabling the system to learn from clinician interactions and patient outcomes.

8. Conclusion

Rheumatoid arthritis (RA) management requires early intervention, guideline-driven therapy selection, and careful treatment adjustment to prevent irreversible joint damage and disability. Despite the availability of the comprehensive 2021 ACR guideline, most electronic health records lack integrated, RA-specific clinical decision support, leaving clinicians to navigate complex recommendations during time-constrained visits manually. This creates a significant gap in care, often resulting in delayed treatment escalation, inconsistent adherence to treat-to-target strategies, and unnecessary reliance on glucocorticoids.

The proposed Python-based, HAPI FHIR–integrated CDSS directly addresses this problem by automating the mapping of patient-specific data—including disease activity, DMARD exposure, and glucocorticoid use—to guideline-aligned treatment recommendations. By delivering concise, evidence-based options at the point of care, the system enhances efficiency, supports shared decision-making, and reduces the cognitive burden on clinicians. While the current prototype

faces limitations in data scope, workflow integration, and security, it demonstrates the feasibility of embedding real-time, transparent, rule-based decision support into RA management.

Ultimately, this work highlights a clear unmet clinical need: a scalable, interoperable CDSS that transforms static RA guidelines into actionable recommendations seamlessly integrated within the clinical workflow. Addressing this need not only promotes consistent, high-quality care but also lays the foundation for a future in which precision-guided, patient-centered RA treatment is standard rather than the exception.

References

Alder, H., Marx, C., Steurer, J., Wertli, M. M., Korner-Nievergelt, P., Tamborrini, G., Langenegger, T., Eichholzer, A., Andor, M., Krebs, A., Michel, B. A., & Wildi, L. (2020). RheumaTool, a novel clinical decision support system for the diagnosis of rheumatic diseases, and its first validation in a retrospective chart analysis. *Swiss Medical Weekly*, *150*, w20369. https://doi.org/10.4414/smw.2020.20369

Ancker, J. S., Edwards, A., Nosal, S., Hauser, D., Mauer, E., Kaushal, R., & With the HITEC Investigators. (2017). Effects of workload, work complexity, and repeated alerts on alert fatigue in a clinical decision support system. *BMC Medical Informatics and Decision Making*, *17*(1), 36.

Ayaz, M., Pasha, M. F., Alzahrani, M. Y., Budiarto, R., & Stiawan, D. (2021). The Fast Healthcare Interoperability Resources (FHIR) standard: Systematic literature review of implementations, applications, challenges and opportunities. *JMIR Medical Informatics*, *9*(7), e21929.

Bender, D., & Sartipi, K. (2013). HL7 FHIR: An Agile and RESTful approach to healthcare information exchange. *Proceedings of the 26th IEEE International Symposium on Computer-Based Medical Systems*, 326-331.

Evans, E. L., & Whicher, D. (2018). What should oversight of clinical decision support systems look like? *AMA Journal of Ethics*, *20*(9), 857-863.

Fox, J., & Thomson, R. (2002). Clinical decision support systems: A discussion of quality, safety and legal liability issues. In *Proceedings of the AMIA Symposium* (p. 265). American Medical Informatics Association.

Fraenkel, L., Bathon, J. M., England, B. R., St.Clair, E. W., Arayssi, T., Carandang, K., Corrigan, C. C.,危, H., Feldman, C. H., Frieson, M. M., Ghib, G., Harrold, L. R., Horowitz, D., Hsieh, P. H., Lakin, P. R., Lal, P., Lawson, E. F., Lendowski, R., Menzin, A. W., ... Akl, E. A. (2021). 2021 American College of Rheumatology guideline for the treatment of rheumatoid arthritis. *Arthritis Care & Research*, *73*(7), 924-939.

Goodman, K. W. (2007). Ethical and legal issues in decision support. In E. S. Berner (Ed.), *Clinical decision support systems* (pp. 127-147). Springer. https://doi.org/10.1007/978-0-387-38319-4_6

Hassija, V., Chamola, V., Mahapatra, A., Singal, A., Goel, D., Huang, K., Scardapane, S., Spinelli, I., Mahmud, M., & Hussain, A. (2024). Interpreting black-box models: A review on explainable artificial intelligence. *Cognitive Computation*, *16*(1), 45-74.

Jones, C., Thornton, J., & Wyatt, J. C. (2023). Artificial intelligence and clinical decision support: Clinicians' perspectives on trust, trustworthiness, and liability. *Medical Law Review*, *31*(4), 501-520.

Labinsky, H., Ukalovic, D., Hartmann, F., Runft, V., Wichmann, A., Jakubcik, J., Gambel, K., Otani, K., Morf, H., Taubmann, J., Fagni, F., Kleyer, A., Simon, D., Schett, G., Reichert, M., & Knitza, J. (2023). An AI-powered clinical decision support system to predict flares in rheumatoid arthritis: A pilot study. *Diagnostics*, *13*(1), 148. https://doi.org/10.3390/diagnostics13010148

Lehne, M., Sass, J., Essenwanger, A., Schepers, J., & Thun, S. (2019). Why digital medicine depends on interoperability. *NPJ Digital Medicine*, *2*(1), 79.

Messelink, M. A., Fadaei, S., Verhoef, L. M., Welsing, P., Nijhof, N. C., & Westland, H. (2024). Rheumatoid arthritis patients' perspective on the use of prediction models in clinical decision-making. *Rheumatology*, *64*(6), 1045-1051. https://doi.org/10.1093/rheumatology/keae202

Papadopoulos, P., Soflano, M., Chaudy, Y., Adejo, W., & Connolly, T. M. (2022). A systematic review of technologies and standards used in the development of rule-based clinical decision support systems. *Health and Technology*, *12*(4), 713-727.

Precision Rheumatology. (2025, March 25). The biggest challenges in finding effective rheumatoid arthritis treatment—And how to overcome them. https://precisionrheum.com/the-biggest-challenges-in-finding-effective-rheumatoid-arthritis-treatment-and-how-to-overcome-them/#:~:text=Rheumatoid%20arthritis%20(RA)%20is%20a,and%20maintaining%20an%20active%20lifestyle

Prictor, M. (2023). Where does responsibility lie? Analysing legal and regulatory responses to flawed clinical decision support systems when patients suffer harm. *Medical Law Review*, *31*(1), 1-24.

Rosenbloom, S. T., Miller, R. A., Johnson, K. B., Elkin, P. L., & Brown, S. H. (2011). Interface terminologies: Facilitating direct entry of clinical data into electronic health record systems. *Journal of the American Medical Informatics Association*, *18*(4), 413-417.

Schipper, L. G., van Hulst, L. T., Grol, R., van Riel, P. L., Hulscher, M. E., & Fransen, J. (2010). Meta-analysis of tight control strategies in rheumatoid arthritis: Protocolized treatment has additional value with respect to the clinical outcome. *Rheumatology*, *49*(11), 2154-2164.

Scott, D. L., Pugner, K., Kaarela, K., Doyle, D. V., Woolf, A., Holmes, J., & Hieke, K. (2000). The links between joint damage and disability in rheumatoid arthritis. *Rheumatology*, *39*(2), 122-132.

Stoffer, M. A., Schoels, M. M., Smolen, J. S., Aletaha, D., Bijlsma, J. W., Breedveld, F. C., Emery, P., Gramatzki, D., Kvien, T. K., Montecucco, C., Scholte-Voshaar, M., Stamm, T., Takeuchi, T., Verpoort, K. N., Winthrop, K. L., & van der Heijde, D. (2016). Evidence for treating rheumatoid arthritis to target: Results of a systematic literature search update. *Annals of the Rheumatic Diseases*, *75*(1), 16-22.

Sutton, R. T., Pincock, D., Baumgart, D. C., Sadowski, D. C., Fedorak, R. N., & Kroeker, K. I. (2020). An overview of clinical decision support systems: Benefits, risks, and strategies for success. *NPJ Digital Medicine*, *3*(1), 17. https://doi.org/10.1038/s41746-020-0221-y

Takanashi, S., & Kaneko, Y. (2024). Unmet needs and current challenges of rheumatoid arthritis: Difficult-to-treat rheumatoid arthritis and late-onset rheumatoid arthritis. *Journal of Clinical Medicine*, *13*(24), 7594. https://doi.org/10.3390/jcm13247594

Tanaka, Y. (2021). Recent progress in treatments of rheumatoid arthritis: An overview of developments in biologics and small molecules, and remaining unmet needs. *Rheumatology*, *60*(Suppl 6), vi12-vi20. https://doi.org/10.1093/rheumatology/keab609

World Health Organization. (2023). *Rheumatoid arthritis*. https://www.who.int/news-room/fact-sheets/detail/rheumatoid-arthritis#:~:text=Key%20facts,higher%20risk%20of%20developing%20RA

Appendices

Appendix A:

Table A1

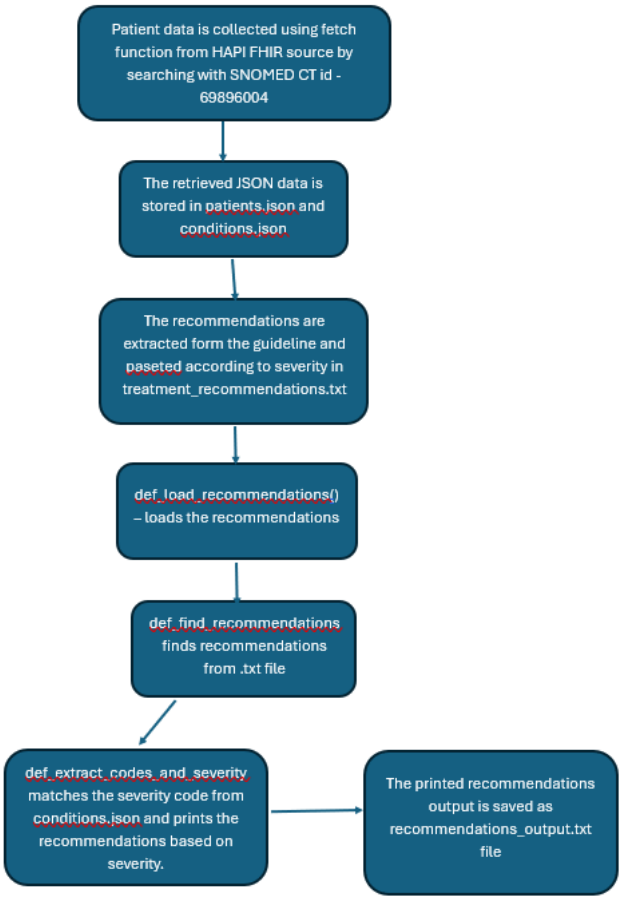
ACR-2021–derived RA treatment rules encoded for the CDSS; severity buckets map to low/moderate/high activity in the guideline (Fraenkel et al., 2021).

SNOMED (RA)	Severity	Recommendation text (exact)	Notes on ACR 2021 alignment
69896004	mild	Start a csDMARD; hydroxychloroquine preferred. Sulfasalazine over methotrexate; methotrexate over leflunomide. Avoid chronic glucocorticoids (short course, lowest dose only if needed). Reassess in 4–12 weeks (treat-to-target).	Low activity: HCQ preferred; SSZ > MTX; MTX > LEF; minimize steroids; T2T cadence 4–12 weeks.
69896004	mild to moderate	Start methotrexate for moderate activity; consider hydroxychloroquine only if close to low activity. Use treat-to-target; reassess in 4–12 weeks; escalate if not at target in 3–6 months.	Clarifies MTX preference once activity trends to moderate; explicit reassessment window.
69896004	moderate	Start methotrexate monotherapy; titrate to ≥15 mg/week within 4–6 weeks if tolerated (optimize route/split dosing and folate). Avoid chronic glucocorticoids. If not at target on maximized methotrexate, add a bDMARD or tsDMARD rather than triple therapy (shared decision-making).	Prefers add-on b/tsDMARD over triple therapy after inadequate MTX; includes dose optimization.
69896004	moderate to severe	Start methotrexate promptly (treat-to-target). Consider short course, lowest-dose glucocorticoids only if needed. Reassess in 4–12 weeks. If not at target on maximized methotrexate, add a bDMARD or	Mirrors moderate–high pathway; minimize steroids; explicit T2T review; escalation path.

		tsDMARD rather than triple therapy.	
69896004	severe	Start methotrexate promptly for high disease activity (treat-to-target) with rapid titration. Consider short course, lowest-dose glucocorticoids only if needed. Reassess in 4–12 weeks; if not at target on maximized methotrexate, add a bDMARD or tsDMARD rather than triple therapy.	High activity: rapid MTX titration; early reassessment; add b/tsDMARD rather than triple therapy.

Note: "mild" = low activity; "severe" = high activity; combined rows ("mild to moderate," "moderate to severe") ensure coverage for boundary cases while preserving ACR intent (Fraenkel et al., 2021).

Appendix E: System Architecture Diagram



Appendix F: Code Implementation

F.1 Code Snippets

```
import random
import requests
import json
from pathlib import Path
import datetime


BASE_URL = "http://hapi.fhir.org/baseR4"
# Directory to store files
data_dir = Path("data")
data_dir.mkdir(exist_ok=True)
```

```
# Extract patients records with Rheumatoid Arthritis
def fetch_patient():
    url = f"{BASE_URL}/Patient"
    f"?_has:Condition:subject:code=http://snomed.info/sct|69896004"
    f"&_sort=-_lastUpdated&_count=5"
    response = requests.get(url)
    data = response.json()
    file_path = data_dir / "patients.json"
    with open(file_path, "w", encoding="utf-8") as f:
        json.dump(data, f, indent=2, ensure_ascii=False)
    print(f"Saved {len(data.get('entry', []))} patient records to {file_path}")

# Extract condition records of patients with Rheumatoid arthritis
def get_condition_with_details():
    url = f"{BASE_URL}/Condition?code=http://snomed.info/sct|69896004&_sort=-_lastUpdated&_count=10"
    response = requests.get(url)
    data = response.json()
    file_path = data_dir / "conditions.json"
    with open(file_path, "w", encoding="utf-8") as f:
        json.dump(data, f, indent=2, ensure_ascii=False)

    print(f"Saved {len(data.get('entry', []))} patient records to {file_path}")
```

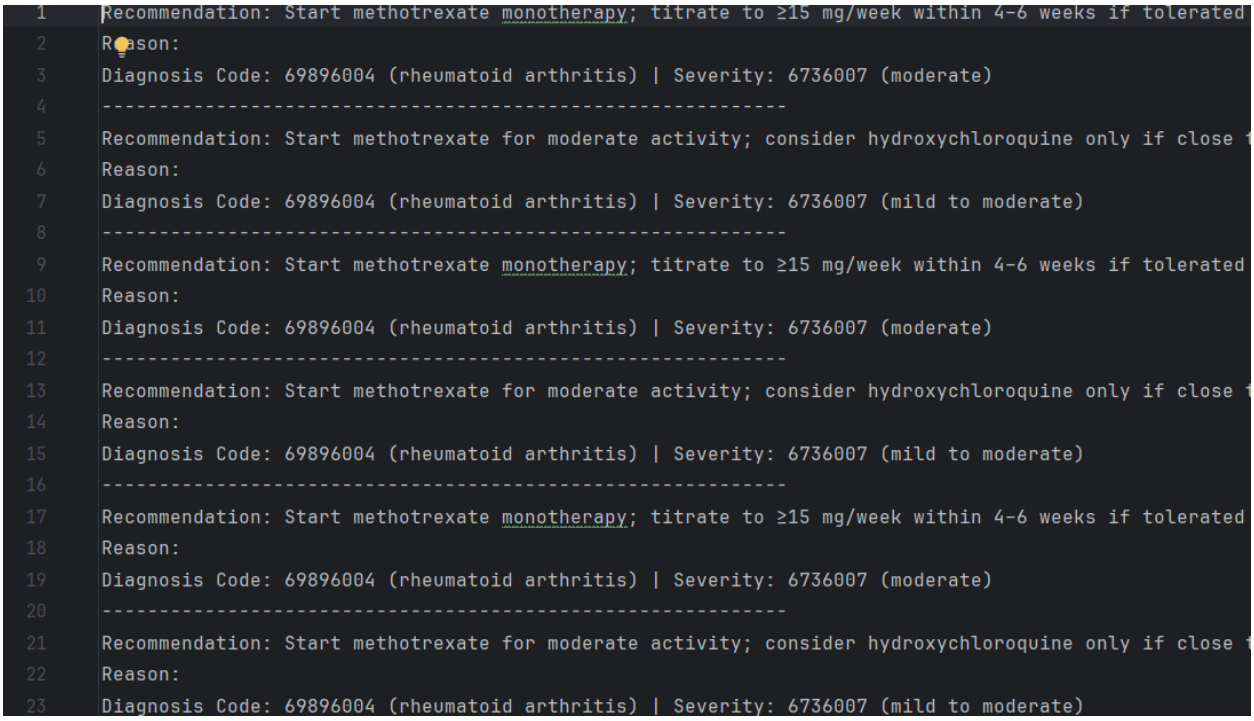
```
if __name__ == "__main__":
    fetch_patient()
```

```
rules_file = data_dir / "treatment_recommendations.txt"

def load_recommendations():  1 usage  anichiti
    rules = []
    with open(rules_file, "r", encoding="utf-8") as f:
        for line in f:
            if "|" in line:
                diagnosis_code, severity, recommendation = line.strip().split( sep: "|",  maxsplit: 2)
                rules.append({
                    "diagnosis_code": diagnosis_code,
                    "severity": severity.lower(),
                    "recommendation": recommendation
                })
    return rules

def find_recommendation(diagnosis_code, severity, rules):  1 usage  anichiti
    for rule in rules:
        if rule["diagnosis_code"] == diagnosis_code and rule["severity"] == severity.lower():
            return rule["recommendation"]
    return None

def extract_codes_and_severity():  1 usage  anichiti
    file_path = data_dir / "conditions.json"
    output_file = data_dir / "recommendations_output.txt"
```

```
def extract_codes_and_severity():  1 usage  anichiti
    # start fresh each run
    output_file.write_text( data: "", encoding="utf-8")

    # load recommendations from text file
    rules = load_recommendations()

    with open(file_path, "r", encoding="utf-8") as f:
        data = json.load(f)

    for entry in data.get("entry", []):
        resource = entry["resource"]

        # Diagnosis
        diagnosis_code = resource["code"]["coding"][0]["code"]
        diagnosis_display = resource["code"]["coding"][0]["display"]

        # Severity
        severity_code = resource["severity"]["coding"][0]["code"]
        severity_display = resource["severity"]["coding"][0]["display"].lower()

        print(f"Diagnosis Code: {diagnosis_code} ({diagnosis_display}) | "
              f"Severity: {severity_code} ({severity_display})")

        recommendation = find_recommendation(diagnosis_code, severity_display, rules)
```

```
        # Severity
        severity_code = resource["severity"]["coding"][0]["code"]
        severity_display = resource["severity"]["coding"][0]["display"].lower()

        print(f"Diagnosis Code: {diagnosis_code} ({diagnosis_display}) | "
              f"Severity: {severity_code} ({severity_display})")

        recommendation = find_recommendation(diagnosis_code, severity_display, rules)

        if recommendation:
            rec_text = (
                f"Recommendation: {recommendation}\n"
                f"Reason:\nDiagnosis Code: {diagnosis_code} ({diagnosis_display}) | "
                f"Severity: {severity_code} ({severity_display})\n"
                + "-" * 60 + "\n"
            )
            print(rec_text, end="")
            with open(output_file, "a", encoding="utf-8") as rf:
                rf.write(rec_text)

if __name__ == "__main__":
    extract_codes_and_severity()
```

```
69896004|mild|Start a csDMARD; hydroxychloroquine preferred. Sulfasalazine over methotrexate; methotrexate over leflun( ✓14
69896004|mild to moderate|Start methotrexate for moderate activity; consider hydroxychloroquine only if close to low activi
69896004|moderate|Start methotrexate monotherapy; titrate to ≥15 mg/week within 4-6 weeks if tolerated (optimize route/spli
69896004|moderate to severe|Start methotrexate promptly (treat-to-target). Consider short course, lowest-dose glucocorticoi
69896004|severe|Start methotrexate promptly for high disease activity (treat-to-target) with rapid titration. Consider shor
```



The repository includes all implementation files and a README with setup instructions.