An Approach for a Web-based Diagnosis Cockpit for the Field of Neurology

Mirco Josefiok, Oliver Norkus and Jürgen Sauer

Deparment of Computer Science
Carl von Ossietzky Universität Oldenburg
Ammerländer Heerstr. 114-118, 26129 Oldenburg
Email: {mirco.josefiok, oliver.norkus, juergen.sauer}@uni-oldenburg.de

Abstract—**In this paper, we present an approach for a Web-based diagnosis cockpit for the field of neurology which can be deployed either in a Web-based environment or as an on-premise solution. We strongly emphasize self service capabilities by adding additional metadata. Often, diseases manifest themselves through varying combinations of symptoms, especially in the field of neurology. Supporting medical staff in the process of finding a correct diagnosis to a hypothesis in a timely manner is very desirable to improve a patients outcome.**

Keywords–Expert systems in neurology, diagnosis support systems, Web-based information system

I. INTRODUCTION

Medical diagnoses in general, and neurological diagnoses, in particular, manifest themselves in different and varying combinations of symptoms. Practitioners rely on their experience and knowledge to find a diagnosis based on a hypothesis. They have to deal with various different symptoms and need to decide whether to perform further diagnostics or to begin an appropriate treatment. Stroke is one of the most common diseases in Germany. About six billion Euros are spent each year for the treatment of the most frequent types[1]. Especially for stroke patients a rapid and early diagnosis significantly improves their prognosis and allows for taking appropriate measures. It is therefore very desirable to offer support to medical staff as much as possible.

Hospitals and resident doctors have to deal with limited resources and minimal information technology capacities. Acquisition and operation of large scale solutions is often not an option. Expert interviews also indicate a demand for a solution which is easily accessible, needs minimal maintenance and blends seamlessly into the the daily work of the medical staff. Recently, we performed several expert interviews with doctors, controllers and information technology (IT) managers from local hospitals [1]. On foundation of those interviews, we were able to create a theoretical concept with a corresponding business process for support systems, which we developed in close cooperation with the interviewed experts [2]. In addition, we propose an approach for Web-based analytical information systems with self-service of business users in mind. We evaluated the developed prototype in several scenarios with business users who had different levels of previous knowledge [3]. The overall feedback was very positive - therefore we adapted architecture, structure and approach for developing the diagnosis cockpit.

[1]see diagnosis code I60, I61, I63, I65 at www.gbe-bund.de - last visited 2015-01-14

In this paper, we present the first approach for a Web-based diagnosis cockpit for the field of neurology. Our approach comprises an expert system with an underlying knowledge base, an interface to provide additional metadata and a HTML5 frontend. With the introduction of additional metadata, we strive to make our system more accessible.

This paper is organized as follows. Section II introduces the foundations of our work, namely expert systems in neurology and Web based decision support systems (DSS). Section III analyzes the problem statement and introduces the architectural requirements for a Web-based diagnosis cockpit. Section IV presents our approach and gives an overview of the developed prototype and its architecture. Finally, the paper concludes with Section V, in which we point out remaining research and development challenges.

II. FOUNDATION

Expert systems are software solutions utilizing specialists knowledge in order to support the decision-making process [4]. Contrary to other knowledge-based systems, which expert systems belong to, a knowledge base is derived from human experts. However, expert systems seek to reduce the need for human experts in practice. This is one main advantage, because human experts are often very costly [5]. In the healthcare domain, arious attempts of applying expert systems exist and the possible benefits are widely recognized [6, 7, 8]. Research and development of medical expert systems started in the 70s and systems are still evolving. MYCIN [9] and CADUCEUS [10] represent early attempts of expert systems with an inference engine. They were superseded by systems able to deal with imprecise or incomplete information. At present, computer systems are used in various steps of a patients treatment, e.g. in detecting adverse drug events [11]. But diagnosis related expert systems are rarely used in practice [12, 13, 14].

A. Expert Systems in Neurology

In a recent survey, we searched PubMed, Mendeley and Google Scholar for the terms *neurology*, *expert systems*, *diagnosis support systems*, *decision support systems* and *healthcare* in various combinations. We identified more than 15,000 references. After further filtering, we were able to reduce this set by 14,920 references. Afterwards we identified and removed 21 duplicates, 41 unrelated references and 11 reviews and editorials. Finally, we were left with 59 full text references from which 43 turned out to be unrelated work and 11 were identified as reviews or editorials. Subsequent to this, only eight full text references, actually dealing with diagnosis support systems in neurology, still remained [15].

 ISBN: 978-1-61208-407-7

TABLE I. RESULTS OF LITERATURE ANALYSIS

Name	Last update	Number of diseases	Target diseases	Knowledge base	Method	Provision	Reference
Bunke	1988	unknown	Neurological diseases	Specialized literature	Database, rule-based	unknown	[16]
Bickel	2006	400	Neurological diseases	unknown	Database, statistical	Desktop Application	[17]
Roses	2009	6	Neurological diseases	unknown	Rule-based	Web-based	[18]
Reimers	2009	unknown	Neurological diseases	Specialized literature	Database, rule-based	Desktop Application	[19]
Borgohain	2012	4	Neuromuscular disorders	Expert interviews	Rule-based	Desktop Application	[20]
Borgohain	2012	1	Cerebral Palys Diagnosis	Expert interviews	Rule-based	Desktop Application	[21]
Ghahazi	2014	1	Multiple Sclerosis	unknown	Fuzzy logic	Spreadsheet	[22]

Table I gives an overview of our findings. We discovered different approaches for expert systems in neurology, but we found no references which show or hint at a routine application of any expert system. This emphasizes our previous findings [2]. In all cases, no considerations regarding user interface, comprehensibility of suggestions, expandability and customizability of knowledge are published. For practitioners in healthcare it is crucial that the results of the experts are comprehensible.

B. Web-based Decision Support Systems

Usually DSS consist of three basic components: A data management component, a knowledge base and a user interface. In case of Web-based DSS, they can be probably deployed on individual servers. DSS are very close to database management systems (DBMS) and are often built on top of a DBMS. More recent approaches tend to integrate the DSS into a broader application [23]. Those integrated DSS allow for an improved workflow and they are likely to be used more frequently. Most DSS, and also integrated DSS, are regular desktop applications and therefore need to be installed on every computer, where access is required. Furthermore, they do not offer an expandable knowledge base. Rare Web-based DSS offer access through common Web browser and mobile devices, which will be particularly important in hospitals.

III. Requirements and Business Drivers

During the expert interviews, we were able to derive the following requirements for a diagnosis cockpit:

Performance: Performance comparable to common desktop solutions.

Modularity and expandability: It should be possible to subsequently add complementary functionality to the prototype.

Expandability of knowledge: It should be possible to modify and expand the underlying knowledge base.

Metadata: Meaningful metadata should be provided to increase efficiency and accessibility when performing analyses.

User empowerment: It should be possible for inexperienced users to perform complex analyses.

Web standards: The prototype should use Web standards and should conform to the RESTful architecture paradigm. Representational State Transfer (REST) is widely accepted as a simper alternative to service approaches [24].

A recent study [7] lists additional success factors for clinical DSS, which include seamless integration in the workflow of clinicians, integration of the DSS into a broader system, provision of recommendations rather then decisions, minimal data input, easy access and clear presentation of the results.

In Section IV we show how we approach the requirements and in Section V we will point out what we intend to do to fulfill them.

IV. Approach for a Diagnosis Cockpit

In this section, we present our approach for a Web-based diagnosis cockpit. In Section IV-A we describe the server side of our application and especially how we utilize various technologies for our purpose. In Section IV-B, we explain the implementation of our client, with particular emphasis on self-service and the usage of metadata.

The framework conditions for our approach are determined by the theoretical concept and the corresponding business process. Both were developed in close cooperation with experts from the healthcare domain. Figure 1 depicts an excerpt of the business process. It shows how case data, meta data and patient data are prepared and provided.

A. Server side Prototype

For developing the server side part of our prototype, we chose Spring Boot, which offers rapid application development capabilities whilst being well extensible. This complies with our requirement *Modularity and expandability*. Our prototype does not require a dedicated application server but only a Java Virtual Machine. Moreover, it relies on a relational database for storing data. In future iterations we'd like to add a NoSQL database for storing / caching unstructured data. Figure 2 shows the structure of the application. We implemented a structure conforming to the Representational State Transfer (REST) software architecture style. RESTful Web services for data exchange between client and server are implemented. These standards were chosen to comply with the business driver *Web standards*. Moreover, we added the capability to work securely in a multi-client environment.

Concering the *Recommender* component, we have not yet finally decided which approach to follow. We are evaluating whether to use FuzzyLogic or to rely on crisp rules and emulate a fuzzy approach. We created small knowledge based on the expert interviews and implemented the knowledge base with Fuzzy Control Language (FCL) and JBoss Drools Domain Specific Language. Drools [25] offers a comprehensive environment for developing expert systems with a crisp rule set. There are approaches using fuzzy logic with Drools, but they are not mature, yet [26]. FCL was standardized by IEC 61131-7 and different Java implementations exits, eg. jFuzzyLogic [27].

At the moment we carry out experiments to decide which approach complies most with our requirements (see section III), especially *Performance*. Listing 1 shows an FCL example for Neuritis Vestibularis. There are four possible symptoms as inputs and the diagnosis as an output. Altogether,

 ISBN: 978-1-61208-407-7

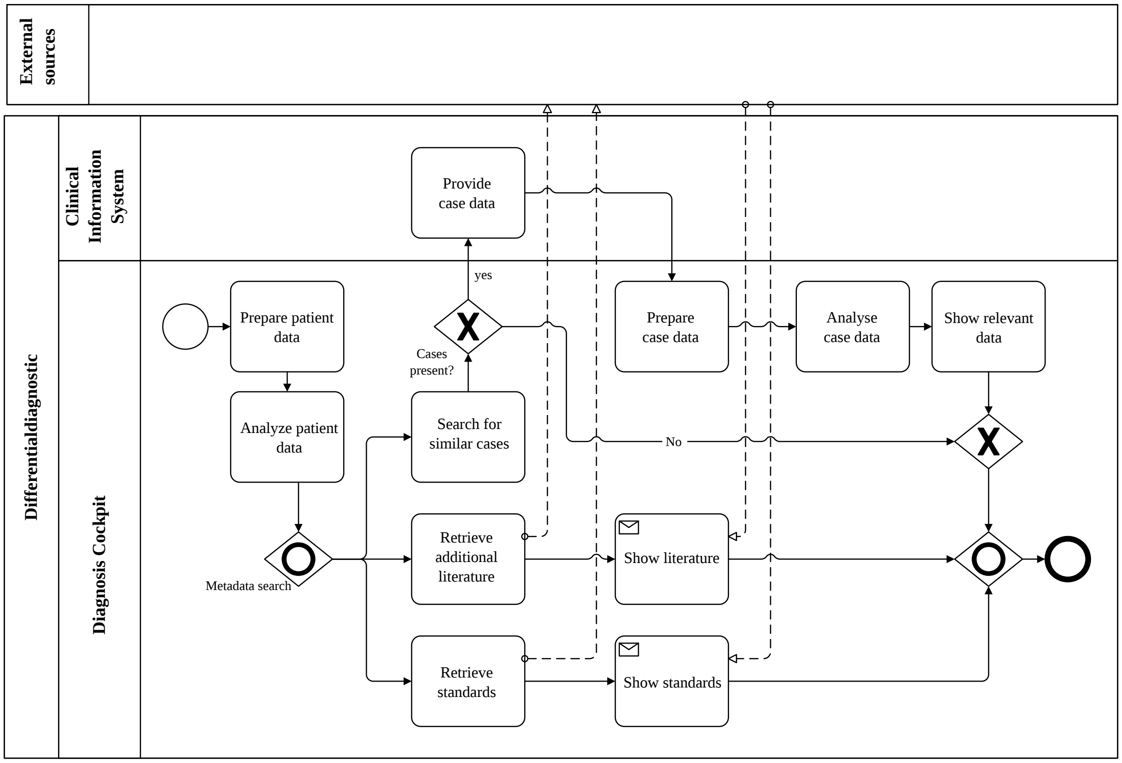

Figure 1. Preparation and Provision of Case Data, Metadata and Patient Data

we deducted seven rules from expert interviews, but only the one which produces *sure* is depicted as an example.

Additional, meaningful metadata is provided through the integration of third-party services. Our prototype fetches domain specific knowledge, regarding diagnoses, symptoms and clinical pathways, from the "Deutsche Gesellschaft für Neurologie" [28]. We search PubMed for supplementary literature to the recommendations produced by our *Recommender*. PubMed is a search engine access provider to the MedLine database of medical publications. In future iterations we plan to integrate WikiData to access correct medical classification codes. Through the *ServiceFactory* we offer a unified access point for possible clients as a RESTful interface. Data is serialized into JavaScript Object Notation (JSON), before sending it to the client.

```
FUNCTION_BLOCK Neuritis_vestibularis
VAR_INPUT
        rotary_vertigo : REAL;
        tendency_to_fall : REAL;
        spontaneous_nystagmus : REAL;
        unterberger_test : REAL;
END_VAR
VAR_OUTPUT
        Neuritis_vestibularis : REAL;
END_VAR
FUZZIFY rotary_vertigo
        TERM sudden := (0, 1) (2, 1) (4, 0) ;
        TERM attack := (2, 0) (4, 1) (6, 1) (8, 0);
        TERM constant := (6, 0) (8, 1) (10, 1);
END_FUZZIFY
FUZZIFY tendency_to_fall
        TERM left := 1;
        TERM right := 2;
END_FUZZIFY
FUZZIFY spontaneous_nystagmus
        TERM left := 1;
        TERM right := 2;
END_FUZZIFY
FUZZIFY unterberger_test
        TERM left := 1;
        TERM right := 2;
END_FUZZIFY
DEFUZZIFY Neuritis_vestibularis
        TERM possible := (0, 1) (4, 0) ;
        TERM probable := (1, 0) (4,1) (6,1) (9,0);
        TERM sure := (6, 0) (9, 1);
        METHOD : COG;
        DEFAULT := 0;
END_DEFUZZIFY
RULEBLOCK No1
        AND : MIN;
        ACT : MIN;
        ACCU : MAX;
        RULE 1 : IF rotary_vertigo IS  sudden AND tendency_to_fall IS left AND
               spontaneous_nystagmus IS right AND unterberger_test IS left THEN
               Neuritis_vestibularis IS sure;
        ...
END_RULEBLOCK
END_FUNCTION_BLOCK
```

Listing 1. FCL Example for Neuritis Vestibularis

B. *Client side Prototype*

Our prototypical client consumes the Web services of the described server side prototype. It was developed using HTML5/JavaScript, especially AngularJS [29]. In order to contribute to the business driver *Modularity and expandability*, the application is organized in a modular way with reusable components representing different aspects of the view.

Figure 3 shows a screenshot of the main view. (1) shows a selected patient. Basic information about the patient is provided (e.g. name, address etc.). Possible other fields to display would be age, sex etc. Additional information about medication, planned treatments etc can be seen at (2). (3) shows a fever curve and other corresponding information with customizeable charts is displayed. The process of filtering and selecting a patient is accessible through a dedicated modal. (4) shows the selected symptoms with a short description. Possible diagnoses with additional information can be seen at (5). Again the process of selecting symptoms is accessible through a modal. Additional information in form of literature references is provided at (6). We constantly work on the user interface to

 ISBN: 978-1-61208-407-7

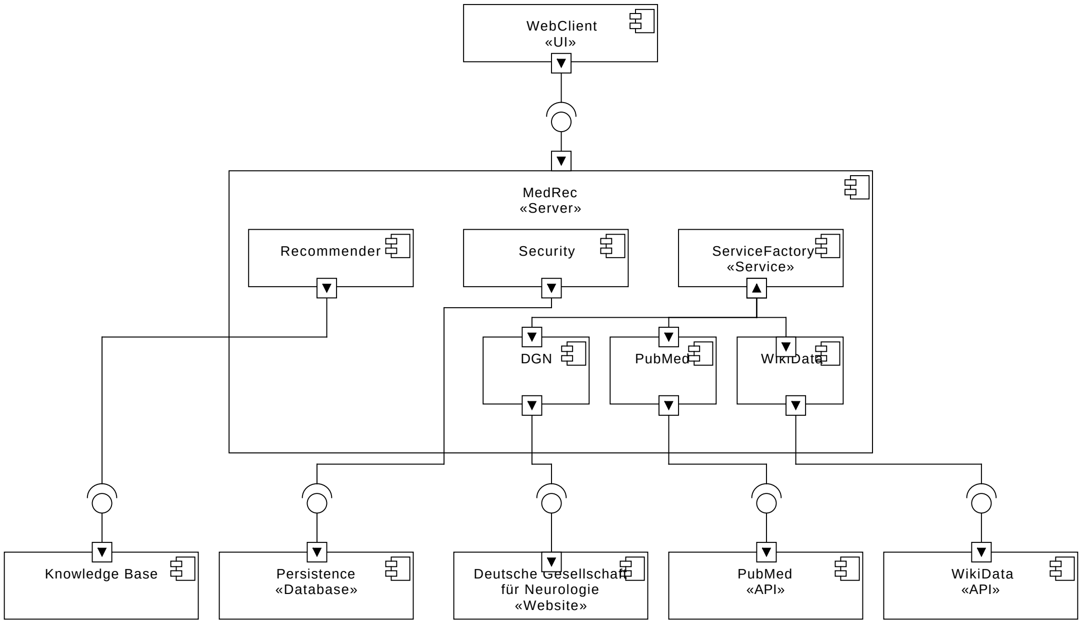

Figure 2. Server Structure

comply with our requirements and make the application easily accessible and integrate our approach into the daily routines of practitioners.

V. Conclusion and further Challenges

We presented an approach for a Web-based diagnosis cockpit. According to early feedback from doctors and medical staff our approach is very promising and a significant improvement over existing solutions. The dynamic addition of metadata, the easy access, the traceability and comprehensibility of results are most important for medical staff. But there was some criticism we need to work on in further iterations, mostly regarding our operation concept. Whilst we developed only a research prototype, a possible application accessible by end users, must take security issues into account. In addition we will rework parts of our operation concept, especially regarding the processes of selecting patients and symptoms. Feedback from IT managers was very positive as well. They considered the low hurdle to usage and the minimal effort needed to maintain our system very promising. At the moment, we mostly work with sample data and an incomplete and limited rule set. We are currently conducting a series of expert interviews with doctors from a local hospital to build a larger knowledge base. We aim to provide a reasonable set of rules with which comprehensible results can be produced and will continuously evaluate our work with practitioners from local healthcare facilities. With a larger knowledge base we will have the opportunity to perform more complete tests and finally chose an approach for our knowledge base. Another planned improvement concerns the modification of the underlying knowledge base by users at runtime. At the moment it is possible to modify the knowledge base with an external tool and afterwards the server side part needs to be recompiled. A future version could incorporate the needed user interface elements which should allow users to modify the knowledge base.

References

[1] M. Josefiok, “Measuring and Monitoring the Rehabilitation of Patients on Monitoring Stations via the Analyses of poly–structured Data,” in *Informatik 2014, 44. Jahrestagung der Gesellschaft für Informatik e.V. (GI),*, ser. LNI. GI, 2014.

[2] J.-P. Weiß, M. Josefiok, T. Krahn, and H.-J. Appelrath, “Entwicklung eines Fachkonzepts für die klinische Entscheidungsunterstützung durch Analytische Informationssysteme,” in *Proceedings of the 12th International Conference on Wirtschaftsinformatik (WI2015)*, 2015.

[3] *An Approach for a Web-based Analysis Solution with MUSTANG*, ser. The Seventh International Conferences on Advanced Service Computing. IARIA XPS Press, 2015.

[4] F. Puppe, *Einführung in Expertensysteme (Studienreihe Informatik) (German Edition)*. Springer, 1988.

[5] C. Beierle and G. Kern-Isberner, *Methoden wissensbasierter Systeme: Grundlagen, Algorithmen, Anwendungen (Computational Intelligence) (German Edition)*, 5th ed. Springer Vieweg, 2014.

[6] E. Berner, “Clinical decision support systems: state of the art,” *AHRQ Publication*, no. 09, 2009.

[7] D. Dinevski, U. Bele, T. Šarenac, U. Rajkovič, and O. Šušteršič, “Clinical decision support systems,” *Studies in health technology and informatics*, pp. 217–238, Dec. 2013.

[8] N. K. J. Adhikari, J. Beyene, J. Sam, and R. B. Haynes, “Effects of Computerized Clinical Decision Support Systems on Practitioner Performance,” vol. 293, no. 10, pp. 1223–1238, 2014.

[9] P. Van Remoortere, “Computer-based medical consultations: MYCIN,” pp. 385–386, 1979.

[10] G. Banks, “Artificial intelligence in medical diagnosis: the INTERNIST/-CADUCEUS approach.” *Critical reviews in medical informatics*, vol. 1, pp. 23–54, 1986.

[11] T. Krahn, M. Eichelberg, S. Gudenkauf, E. Laleci, B. Gokce, and A. Hans-Jürgen, “Adverse Drug Event Notification System – Reusing clinical patient data for semi–automatic ADE detection,” in *Proceedings 27th International Symposium on Computer–Based Medical Systems (IEEE CBMS 2014)*, 2014.

[12] W. Ishak and F. Siraj, “Artificial intelligence in medical application: An exploration,” *Health Informatics Europe Journal*, 2002.

[13] C. Angeli, “Diagnostic Expert Systems: From Expert’s Knowledge to Real-Time Systems,” in *Advanced Knowledge Based Systems: Model, Applications & Research*, 2010, vol. 1, ch. Chapter 4, pp. 50–73.

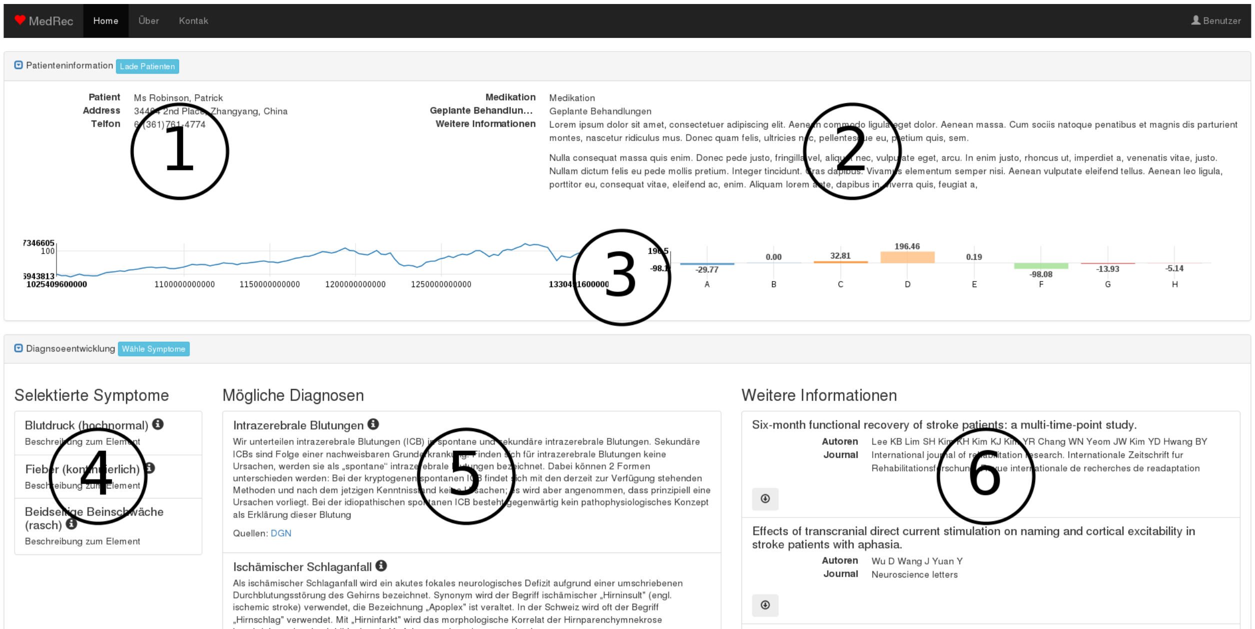

Figure 3. Client Prototype

[14] T. Sharma and S. Jain, "Survey on Expert System," vol. 2, no. 1, pp. 343–346, 2012.

[15] M. Josefiok, T. Krahn, and J. Sauer, "A Survey on Expert Systems for Diagnosis Support in the Field of Neurology," in *Intelligent Decision Technologies - Proceedings of the 4th International Conference on Intelligent Decision Technologies (IDT'2015)*, R. J. Howlett and L. C. Jain, Eds. Springer, 2015.

[16] H. Bunke, F. Flückiger, H. P. Ludin, and F. Conti, "Diagnostic Expert Systems in Neurology - A Comparison between a Rule Based Prototype and a Database Oriented Approach," in *Expert Systems and Decision Support in Medicine*. Springer, 1988, pp. 209–212.

[17] A. Bickel and M. Grunewald, "Ein Expertensystem für das Fachgebiet Neurologie-Möglichkeiten und Grenzen," *Fortschritte der Neurologie· . . .*, pp. 723–731, 2006.

[18] S. Y. M. Vaghefi and T. M. Isfahani, "ROSES: AN EXPERT SYSTEM FOR DIAGNOSING SIX NEUROLOGIC DISEASES IN CHILDREN Title," in *Int. Conf. Intelligent Systems & Agents*, 2009, pp. 259–260.

[19] C. D. Reimers and A. Bitsch, "Differenzialdiagnose Neurologie," *Kohlhammer: Stuttgart*, 2008.

[20] R. Borgohain and S. Sanyal, "Rule Based Expert System for Diagnosis of Neuromuscular Disorders," *arXiv preprint arXiv:1207.2104*, pp. 1–5, 2012. [Online]. Available: http://arxiv.org/abs/1207.2104

[21] ——, "Rule Based Expert System for Cerebral Palsy Diagnosis," *arXiv preprint arXiv:1207.0117*, pp. 1–4, 2012. [Online]. Available: http://arxiv.org/abs/1207.0117

[22] M. A. Ghahazi, M. H. Fazel Zarandi, M. H. Harirchian, and S. R. Damirchi-Darasi, "Fuzzy rule based expert system for diagnosis of multiple sclerosis," in *Norbert Wiener in the 21st Century (21CW), 2014 IEEE Conference on*. IEEE, 2014, pp. 1–5.

[23] R. Elliott, "Strathprints Institutional Repository," *Psychotherapy Research*, vol. 22, pp. 1753–1759, 2008.

[24] Alex Rodriguez, "RESTful Web services: The basics," http://www.ibm.com/developerworks/library/ws-restful/, 2015, last visited 2015-01-22.

[25] Red Hat Inc, "Drools - Drools - Business Rules Management System (Java^{TM}, Open Source," http://www.drools.org, 2015, last visited 2015-01-22.

[26] N. Wulff and D. Sottara, "Fuzzy reasoning with a Rete-OO rule engine," in *Rule Interchange and Applications*. Springer, 2009, pp. 337–344.

[27] P. Cingolani and J. Alcala-Fdez, "jfuzzylogic: a robust and flexible fuzzy-logic inference system language implementation." in *FUZZ-IEEE*. Citeseer, 2012, pp. 1–8.

[28] Deutsche Gesellschaft für Neurologie e.V., "Deutsche Gesellschaft für Neurologie e.V." http://www.dgn.org, 2015, last visited 2015-01-22.

[29] AngularJS Team, "AngularJS - HTML enhanced for web apps!" https://angularjs.org/, 2015, last visited 2015-01-22.