

A Decision Support System for Diagnostics and Treatment Planning in Traumatic Brain Injury

Adil Umer, Jussi Mattila, Hilikka Liedes, Juha Koikkalainen, Jyrki Lötjönen,
Ari Katila, Janek Frantzén, Virginia Newcombe, Olli Tenovuo, David Menon,
Mark van Gils, *Member IEEE*

Abstract— Traumatic Brain Injury (TBI) occurs when an external force causes functional or structural alterations in the brain. Clinical characteristics of TBI vary greatly from patient to patient, and a large amount of data is gathered during various phases of clinical care in these patients. It is hard for clinicians to efficiently integrate and interpret all of these data and plan interventions in a timely manner. This paper describes the technical architecture and functionality of a web-based Decision Support System (DSS), which not only provides advanced support for visualizing complex TBI data but also predicts a possible outcome by using a state-of-the-art Disease State Index machine-learning algorithm. The DSS is developed by using a three-layered architecture and by employing modern programming principles, software design patterns and using robust technologies (C#, ASP.NET MVC, HTML5, JavaScript, and Entity Framework etc.). The DSS is comprised of a patient overview module, a disease-state prediction module and an imaging module. After deploying it on a web-server, the DSS was made available to two hospitals in UK and Finland. Afterwards, we conducted a validation study to evaluate its usability in clinical settings. Initial results of the study indicate that especially less experience clinicians may benefit from this type of decision support software tool.

Index Terms— clinical decision support, traumatic brain injury, web-based tool.

I. INTRODUCTION

Traumatic brain injury (TBI) is defined as an alteration in brain function, or other evidence of brain pathology, caused by an external force [1]. TBI is a major health problem and one of the most common causes of permanent disability and death, resulting in hospitalization rates of 235 patients per 100,000 population across Europe [2]. TBI is extremely variable, both in presentation and in outcome. Particularly in more severely injured patients, investigations and monitoring tools, clinical characteristics of the patient, sociodemographic factors, and pre-injury characteristics (e.g., age, gender, education, employment, etc.) [3, 4], provide a large amount of data, but the quantity and complexity of such data make it extremely difficult to predict outcome or plan treatments reliably. Indeed, current

prognostic schemes account for less than half the variability in outcome across patients [5]. Further, we currently lack clinically useful tools for robust data visualization and decision support for intervention planning.

Hospitals collect a large amount of data in day-to-day practice using electronic health record systems. These data from past patients play a vital role for predicting outcome and planning treatments of prospective patients. Various prediction models based on statistical and data-mining techniques use such data to predict probable outcomes for individual patients with TBI [5, 6, 7]. For example, the prognostic calculator developed in the IMPACT study (International Mission for Prognosis and Analysis of Clinical Trials in TBI), and the logistic regression model that underpins it are widely known [8]. However, although these prognostic models provide useful estimates of likely outcome in moderate-severe TBI [9], there is a lack of a comprehensive computer-based decision support system for the treatment of TBI, which could provide outcome prediction along with high-level visualization of complex patient data.

There are Decision Support Systems (DSS) available to clinicians for various diseases, such as a discovery engine (DE), which provides personalized treatment recommendations after analysing the past data from similar patients. The DE was initially designed to provide informed decisions for patients with breast cancer, but it can be useful in other types of complex diseases. There are some other clinical decision support systems that provide generalized treatment recommendations and drug therapies for example, WizOrder was designed to reduce medical errors, Assessment and treatment of hypertension: evidence based automation (ATHENA) system was developed for hypertension management in primary care, and TherapyEdge-HIV DSS was developed for the treatment of HIV [10]. These systems help clinicians in making decisions. However, they lack the ability to provide visualization of complex patient data in a user-friendly manner. Data visualization functionality is important for clinicians as they can assess the data by themselves and be more confident that the decision provided by the DSS is an appropriate choice for a particular patient.

This work was supported in part by the 7th Framework Program by the European Commission in the co-funded project TBICare (GA-270259) (<http://www.tbicare.eu/>).

Adil Umer, Hilikka Liedes, and Mark van Gils are with the VTT Technical Research Centre of Finland Ltd., Tampere, Finland, (e-mail: adil.umer@vtt.fi, hilikka.liedes@vtt.fi, mark.vangils@vtt.fi).

Jussi Mattila, Juha Koikkalainen and Jyrki Lötjönen are with Combinostics Oy, Finland. (e-mail: jussi.mattila@combinostics.com, juha.koikkalainen@combinostics.com, jyrki.lotjonen@combinostics.com).

Ari Katila, Janek Frantzén and Olli Tenovuo are based at the Turku University Central Hospital (TUCH) and University of Turku, Finland (e-mail: ari.katila@tyks.fi, janek.frantzen@tyks.fi, olli.tenovuo@tyks.fi).

David Menon and Virginia Newcombe are based at the University of Cambridge, Cambridge, UK (e-mail: dkm13@cam.ac.uk, vfjn2@cam.ac.uk).

The web-based DSS described in this paper represents one-step in a solution to address this need. It allows clinicians to explore all available (highly multi-modal) data, collected at different points in time and at various locations, on a dashboard. It not only provides advanced support for visualizing complex data, but also predicts the probable outcome after TBI by using a state-of-the-art prognostic model based on the Disease State Index (DSI) machine-learning algorithm [11]. The DSI was specifically designed together with clinicians to address data analytics challenges in actual clinical practice. It has similar performance to state-of-the-art classification methods, but has the advantages that it does not require pre-processing of data, feature selection or data clean up, and it has been designed to work naturally with missing data. Moreover, the results of the DSI algorithm can be presented as a Disease State Fingerprint (DSF) tree, which allows clinicians to quickly and easily interpret the heterogeneous and multiscale data of a patient and explore different scenarios such as outcome estimations when different variables would (or would not be) available [12]. Moreover, the integrated imaging module of the DSS provides visualization of the Computerized Tomography—CT and Magnetic Resonance Imaging—MRI images along with various overlays, and it also shows the DSF tree after calculating the DSI, specifically for the imaging features. In addition, a wide variety of data from different sources is presented in chronological order as an interactive timeline. The clinicians can navigate between data events and can also access the CT or MRI images directly from the timeline. These state-of-the-art data visualization features are not available in the existing electronic medical record systems.

The main goal of this paper is the presentation of the software solution and its architecture. In addition, the paper also briefly describes the validation study that was conducted at two hospitals in UK and Finland to evaluate the usability of the DSS in clinical settings. Subsequent sections describe the process of requirement elicitation, architecture of the system, and functionality of each major module, along with implementation details and brief illustration of results of the validation study.

II. REQUIREMENT ELICITATION

In an iterative process of design and development, we gathered user requirements at different phases of development. An initial set of requirements was collected by conducting a preliminary user study through individual interviews with eleven clinical specialists (at University of Cambridge (UCA), UK and Turku University Central Hospital (TUCH), Finland) with expertise in TBI management. The main result of the user study was a description of treatment process and diagnosis of TBI, including a workflow diagram, which highlighted possible usage points for the DSS. We developed a first version of the DSS based on the initial set of requirements. Later, in an iterative process of feedback and development, five clinical specialists from TUCH and UCA evaluated the first version of the DSS. As a result, designers and TBI clinicians co-developed user interface (UI) prototypes. Two formal one-to-one interview and UI co-creation sessions were organized with clinicians. Results of these co-creation sessions were used to

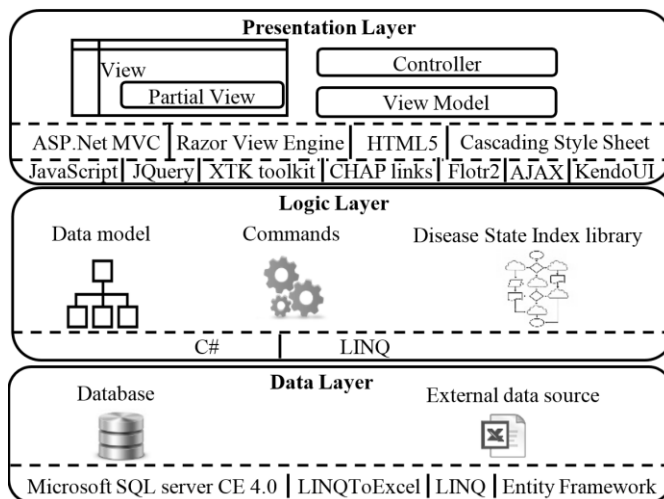


Fig. 1. Architecture of the Decision Support System.

identify, analyse and specify relevant design insights for the system.

III. SYSTEM ARCHITECTURE

The DSS was developed by employing modern programming principles, software design patterns, and latest technologies (C#, ASP.NET MVC, HTML5, JavaScript, Entity Framework etc.). Following a three-layered architecture, the DSS was divided into data, logic and presentation layers (Fig. 1). Moreover, functionality of the DSS was divided into a patient overview module, a disease-state prediction module, and an imaging module. These modules allow exploration of complex patient data, prediction of probable outcome, and visualization for CT and MRI scans.

The presentation layer holds functionality related to UI, the logic layer is responsible for processing of specific application logic, as well as handling communication between presentation and data layers. The data layer processes tasks related to access and management of data. This architectural approach helps to build web-application in a modular manner and supports strong separation of development focus for each aspect [13, 14]. Thus, web-application becomes more robust, scalable, and flexible. In addition to the multi-layer architecture, loose coupling between different DSS modules was ensured by the use of design patterns and modern programming principles such as dependency injection (DI) [15], command-query separation (CQS) [16], repository [17], unit-of-work [18] and model-view-controller (MVC) patterns [19, 20]. The following sections provide in-depth details about layered architecture of the DSS.

A. Data Layer

The major aim of the DSS was to provide statistics and analytics support to clinicians for an optimal decision-making process. The DSS not only provides outcome prediction for decision support but it can also be considered a highly sophisticated viewer, a dashboard, for data that is already being collected from patients with TBI, allowing the clinicians to make the best use of available patient data. The original raw patient data are imported into the DSS from external sources (e.g. hospital information systems).

The ETL (Extract, Transform and Load) process [21] was followed to make data ready for the DSS. Patient data from external data-sources were extracted, cleaned, transformed and combined into MS Excel file format before it was imported into the DSS. The data layer is responsible for reading the data from those MS Excel files and storing it into logically divided data entities appropriate for decision support in TBI. Moreover, it also handles data access requests initiated by the presentation layer/client.

The data were exported from MS Excel files into the DSS by using LINQToExcel (open source .Net library). LINQToExcel facilitates querying MS Excel spreadsheets by using LINQ (Language Integrated Query) syntax. Moreover, the entity framework was used by following a code-first approach, which allows importing data of each patient case into the application through object-relational mappings. Therefore, all types of data entities were first defined as domain objects in a (logic layer) source code file of the DSS. These domain objects represent what type of data can be stored into and retrieved from the system. Once the data are imported, Microsoft SQL server (SQL CE 4.0) becomes responsible for managing imported data by organizing them into interrelated entities and by handling database management operations.

There are six core data entities: patient, injury, situation, condition, measurement and treatment (Fig. 2). These entities are interlinked to each other with one-to-many relationship. A patient can have multiple injuries, there can be multiple measurements regarding one injury, and several treatments exist for addressing one injury. In addition to these entities, there are many specialized sub-types/entities, derived from measurements and treatments. For example, CT, MRI and Glasgow Coma Scale (GCS) measurements are some of the specialized entities derived from generic measurement entity. Similarly, pharmacology and surgery are examples of more specialized treatment entities.

B. Logic Layer

This layer handles the application logic and data manipulation operations. It contains the definition of domain objects, which represent a data model of the DSS, and dictates the creation and layout of a database. C-Sharp (C#) was used as a main programming language for development of different logic layer/server side components. This layer contains a number of commands, invoked by controllers of a presentation layer (see next section). These commands represent actions that are available in the domain of the DSS. Through command interfaces, a controller can query the database and retrieve data that it needs, based on user inputs. Subsequently, the logic layer processes the data accessed by the controller and organizes it into the respective domain objects. Afterwards, it sends the processed data back to the controller, which is responsible for forwarding results to the view. Consequently, less code is required in the presentation layer to determine what to do with particular user inputs.

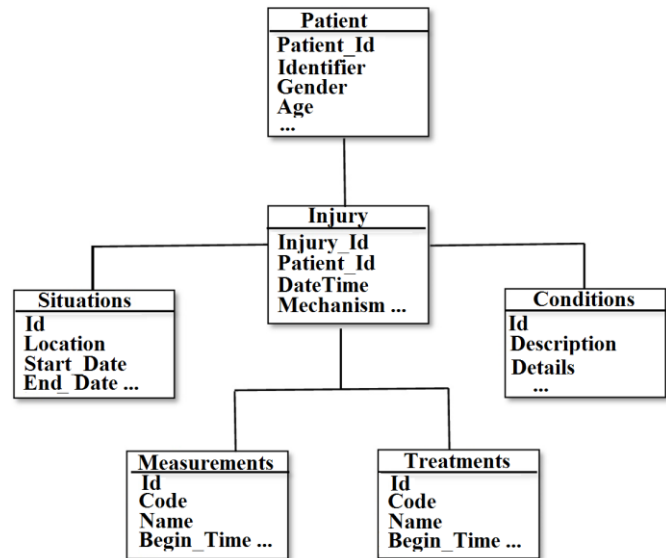


Fig. 2. Data model of the Decision Support System.

C. Presentation Layer

The presentation layer is responsible for displaying processed data to the user in a graphical format within a web-browser. ASP.NET was used as a development framework followed by MVC (Model-View-Controller) design pattern, which is one of the programming models supported by the ASP.NET. It facilitates developing web applications with HTML (Hypertext Markup Language), CSS (Cascading Style Sheets) and JavaScript.

Applying a MVC design pattern is a common practice to develop web-based systems. In the MVC, a “Model” represents either a domain object or combination of multiple domain objects, the “View” is responsible for creating visualizations of data, and the “Controller” is responsible for intercepting all requests invoked by a user through the View (user-interface). It sends user input to the logic layer, which then returns the requested data in a form of domain objects. The controller binds the received data to the view [19, 20].

Most of the controls used in the DSS were based on JavaScript libraries. JQuery¹, JQuery Easy UI², XTK toolkit³, CHAP links⁴, and Flotr2⁵ were some of the main libraries used for the development. Moreover, request and response to and from the server respectively, were handled through AJAX (Asynchronous JavaScript and XML)⁶ calls. This practice helps to improve user experience as well as performance of the application.

IV. USER INTERFACE & FUNCTIONALITY

The DSS comprised three major modules: patient overview, disease-state prediction and imaging module. These modules provide information to the clinician about a patient in a way that can facilitate decision-making process.

¹ <https://jquery.com/>

² <http://www.jeasyui.com/>

³ <https://github.com/xtk/X/blob/master/README.md>

⁴ <http://almende.github.io/chap-links-library/>

⁵ <https://code.google.com/p/flotr/>

⁶ http://www.w3schools.com/ajax/ajax_intro.asp

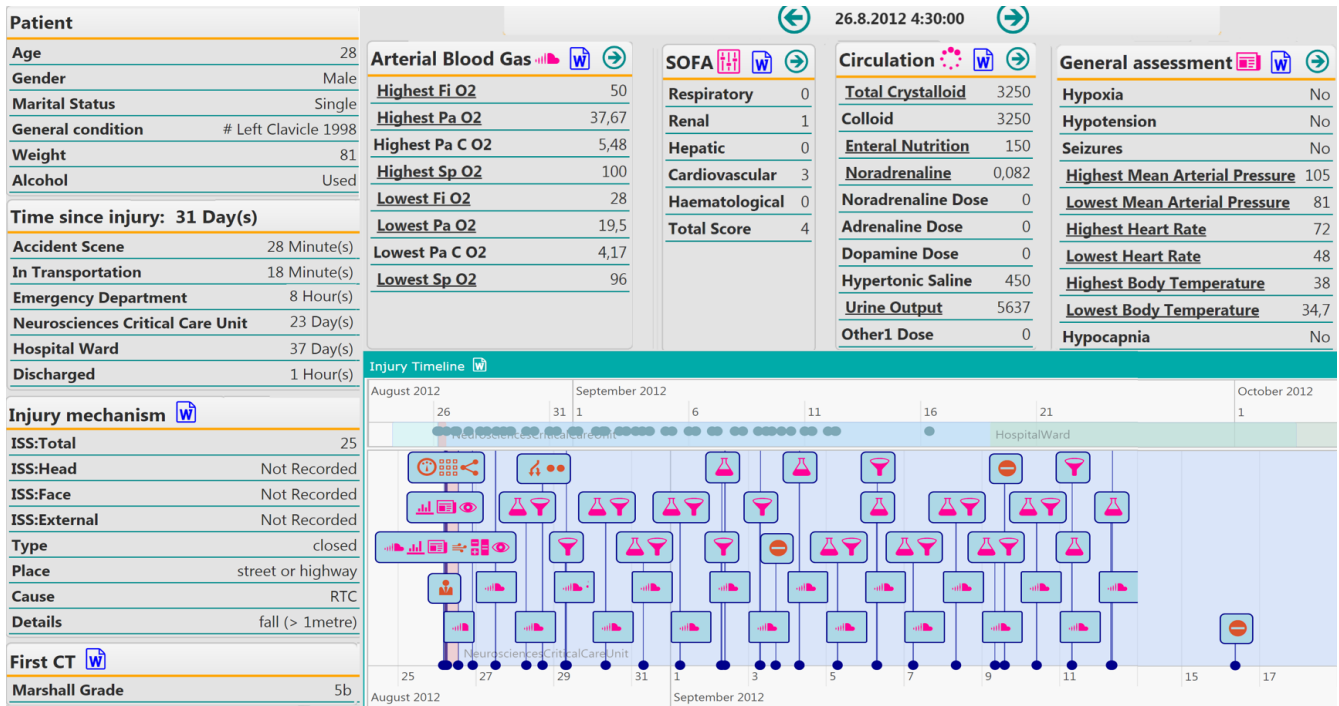


Fig. 3a. Patient overview module: left panel shows the patient background information and injury status whereas lower panel shows timeline of treatments & measurements grouped into events and upper panel shows the details of one of the selected events.

A. Patient Overview Module

This module provides data of the patient being presented in a comprehensive form. It provides demographic and past clinical information about the patient, a summary of the patient’s injury status, and access to diagnostic investigations, monitored data, and therapy details (Fig. 3a). Events are grouped according to the location, (accident scene, emergency department, hospital ward etc.) where specific measurements were made or any interventions were delivered.

By selecting any event, the clinician can view all the details of measurements and treatments associated with that event. Therefore, this interactive navigation allows clinicians to easily browse through a large amount of patient data at a high level, with the ability to focus on and obtain detailed information about any feature of interest.

In addition, this module also contains a line graph, which depicts the GCS measurements that were taken at different points in time. Three lines in a graph show variation in score of eyes, verbal, and motor measurements respectively (Fig. 3b).

B. Disease-state Prediction Module

The Disease-state prediction module is a core feature of the DSS. This module is divided into two parts: prediction of outcome and prediction of prolonged elevation of intra-cranial pressure (ICP). Prediction of outcome shows whether the subject is likely to have an unfavourable outcome (Glasgow Outcome Scale (GOS) 1-3) or a favourable outcome (GOS 4-5) after TBI. Prediction of need for prolonged ICP monitoring shows whether the patient is likely to need prolonged (>7 days) ICP monitoring and therapy. These predictions are calculated using the DSI machine-learning algorithm. This algorithm has proven its power in the case of an early differential diagnosis of

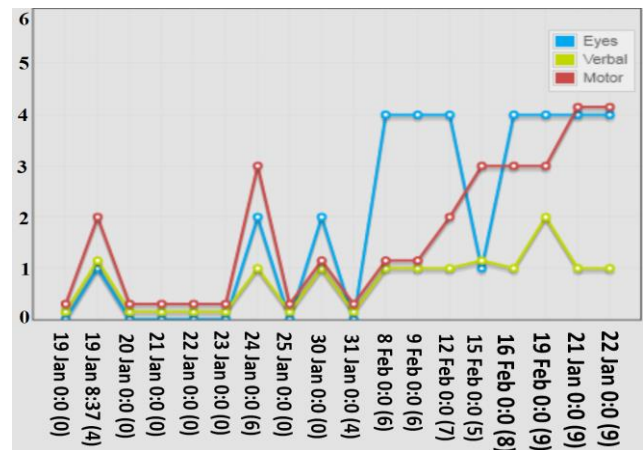


Fig. 3b. Glasgow Coma Scale (GCS) graph: three lines in a graph show variation in scores of eyes, verbal and motor measurements with respect to different date and time.

Alzheimer’s disease by using multivariate data [12, 22]. For these calculations, the algorithm uses data of a patient under examination and data of previously diagnosed patients (training data—which are stored in the database of the DSS) as input. The algorithm returns a hierarchical set of statistical and classification results. The computation results are then transformed into a hierarchical tree for interactive visualization and exploration, called a DSF tree (Fig. 4a). The “TreeView” control of the KendoUI framework was used for generating this tree. JavaScript and HTML5 based KendoUI framework offers various web-based controls and TreeView is one of them. Each node within the tree represents one feature from the patient’s data. The node comprises a box, feature name, DSI value as well as the original value of the feature, if available. The colour of the box represents a DSI value, indicating how similar the patient data are to the cases with unfavourable outcome (or

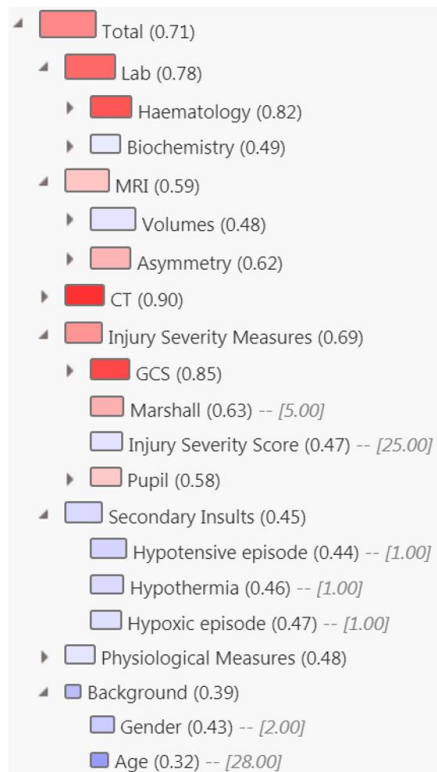


Fig. 4a. Disease State Fingerprint (DSF) tree: the colour of the box in each node represents DSI value—indicating how similar the patient is to the training data, with shades of red and blue representing a scaled similarity to patients with unfavourable and favourable outcome respectively — and the size of the box denotes relevance value—indicating how important a particular measure is for predicting outcome.

cases with prolonged ICP monitoring) in the training data. Red colour denotes an unfavourable outcome (or prolonged ICP monitoring > 7 days) whereas blue colour represents a favourable outcome (or short ICP monitoring ≤ 7 days). The size of the box denotes relevance value, indicating how important a particular measure is for the prediction. The bigger the box, the more relevant is the feature in the prediction. In addition to the DSF tree, users can also view a graph for each node, which shows the DSI value with respect to the GOS or need for prolonged ICP monitoring (Fig. 4b). The user can select any node from the tree and this graph updates according to the selected node. Blue and red dots in a scatter graph show the distribution of patients (training data) according to the DSI and GOS values (or ICP monitoring length) whereas a black vertical line represents the disease state index value of the patient being examined.

C. Imaging Module

The imaging module provides visualization functionality for CT and MRI images. The user can select an image from a list of CT/MRI images taken at different points in time and the selected image loads into a full-view (Fig. 5). Different controls are available for adjusting zoom, contrast or brightness as well as for loading various image overlays. The image overlays provide valuable information to the user about TBI. The user can select any type of overlay from the list of overlays (one at a time) and the image re-renders itself to display the selected overlay. For example, a blood overlay shows suspected

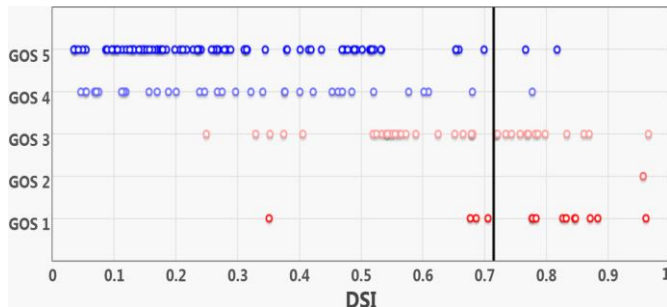


Fig. 4b. Disease State Index (DSI) graph: blue and red dots in the graph show the distributions of patients in the training data according to the DSI and Glasgow Outcome Scale (GOS 1-5) values, whereas a black vertical line represents the DSI value of the patient being examined.

bleeding with red colour. Other overlays show for example segmentations of brain structures, midline shift, and z-scoring of deviations from a healthy brain.

X Toolkit (WebGL for Scientific Visualization) was used for CT and MRI image visualization in the DSS. This is an open source toolkit and is available for use under the MIT license. X-toolkit provides a JavaScript library to load medical images in a web-browser. This library supports various formats of medical images. In the DSS, NIFTI image format was used. Image data were maintained by the DSS on the file system of the web-server.

The Logic layer reads the NIFTI image file from a directory, converts image data into a byte array and serializes it to transmit to a custom JavaScript library. This custom library is responsible for passing data between controller and view. The toolkit parses the received image data by using a NIFTI parser and sends the parsed data to HTML canvas, which then renders the image within a browser. Moreover, X-toolkit is also responsible for rendering different image overlays and their associated colour tables. Colour tables specify different colours for various greyscale intensities in the overlay image. When the user selects a specific overlay type, the custom JavaScript library sends a request to the logic tier that retrieves both overlay image as well as its associated colour table from the file directory and sends them back to the X-toolkit for rendering.

The imaging module also displays general information about the patient and the list of CT/MRI images taken at different points-in-time. Therefore, the clinician can select a specific image to view it in detail. Moreover, DSF tree and graph are also rendered based on the selected image and depicts CT/MRI features and their associated DSI values.

V. DEPLOYMENT AND VALIDATION STUDY

After deploying the DSS on a web-server, it was made available to the clinicians at UCA in UK and TUCH in Finland. Three clinicians from each site evaluated the usability of the DSS in clinical settings. To conduct the validation study, an extra module was added to the DSS. The clinicians participating in the validation study were: neuroanaesthesiologist, neurosurgeon, neurosurgeon-in-training, neurosurgical registers/ attending, and registrar/resident (trainee) in emergency medicine. The validation study was conducted in two phases. In the first phase, the clinicians had to assess each

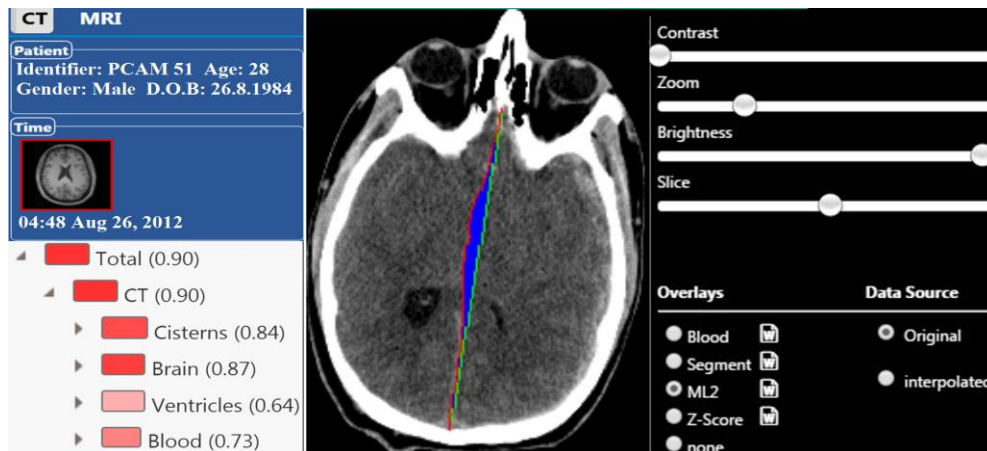


Fig. 5. Example of CT image with midline-shift overlay. Left panel shows general patient information as well as Disease State Fingerprint tree and graph based only on imaging. On the upper right corner are the controls to change visualization of the image, such as increase/decrease zoom, contrast or brightness. In addition, controls are available to change overlays as well as data source of the image.

patient by using the validation module of the DSS without having access to decision support features of the DSS. In the second phase, the clinicians had to re-assess the patient with full functionality of the DSS.

Data of 400 patients with TBI were used as an input for the DSS. Two hundred patients with TBI were selected at each study site. This database of TBI patients served as the training data for the disease-state prediction module (see section IV B).

In total, 60 patients with TBI were selected for the validation study from the database and imported to the DSS for analysis. Patients with mild, moderate and severe TBI were included in the validation study (see Table I). All those patients were included, where data were available for CT or MRI, primary injury and outcome. To minimize the bias that the clinicians involved would remember their patients and their outcomes, the Cambridge data was analysed in Turku and the Turku data in Cambridge.

Each clinician at both sites assessed the 30 TBI subjects, which were available to them according to their site location (Turku or Cambridge). The clinicians predicted the possible outcomes at 6-month for each subject on GOS scale from 1-5. However, as DSI supports only two classes, a two-step dichotomised scale was used as unfavourable outcome (GOS 1-3) and favourable outcome (GOS 4-5). Since no gold standard is available for predicting TBI outcome, GOS, a well-accepted measure, was used.

Table II shows the results of the validation study. The results of the validation study showed that the more experienced clinicians predicted the outcomes as well with the DSS as without the DSS. However, clinicians with less experience predicted the outcome more accurately after using the DSS. Automatic prediction by the DSS achieved accuracies of 80.0% and 83.3% using the studied patients. These values are higher than the values we obtained when applying the DSI to the IMPACT data (accuracy: 70.8-72.3%) that could be considered as baseline [9]. Different data sets used in the analyses explain the differences: the IMPACT data included more subjects, and the IMPACT models included less variables. In this study, the doctors' predictions improved slightly from the average of 78.9% in Phase 1 to 82.2% in Phase 2.

Clinicians gave positive feedback about the usability and support of the DSS in the decision-making process. Moreover, they also suggested various new features and asked for changes to the existing one. Most of these change requests were considered and implemented in the final version of the DSS.

TABLE I
PATIENTS USED IN THE VALIDATION STUDY

CAMBRIDGE		TURKU	
Injury Type	No. of patients	Injury Type	No. of patients
Severe	12	Severe	7
Moderate	3	Moderate	2
Mild	15	Mild	21

TABLE II
COMPARISON OF PREDICTED AND OBSERVED DICHOTOMIZED GOS (1-3 VS. 4-5)

	PHASE 1		PHASE 2		Automatic prediction	
	N correct	% correct	N correct	% correct	N correct	% correct
Doctor 1	23	76.7	25	83.3	24	80.0
Doctor 2	23	76.7	24	80.0	25	83.3
Doctor 3	26	86.7	25	83.3	24	80.0
Doctor 4	25	83.3	24	80.0	25	83.3
Doctor 5	23	76.7	24	80.0	25	83.3
Doctor 6	22	73.3	26	86.7	24	80.0

Phase 1: assessment without decision support features, Phase 2: assessment with decision support features, N correct: number of correct predictions; % correct: percentage of correct predictions; automatic prediction by the decision support system which was based on the Disease State Index model. Doctor 1 and 3 are neurosurgical registers, Doctor 2 is neuroanesthesiologist, Doctor 4 is neurosurgeon, Doctor 5 is neurosurgeon-in-training and Doctor 6 is registrar (trainee) in emergency medicine.

VI. DISCUSSION AND CONCLUSIONS

The decision support system described in this paper provides a practical first step towards the practice of precision medicine in TBI. We described the system architecture and technologies used to develop a web-based decision support system for the diagnosis and treatment planning of TBI. Being web-based, the system guarantees maximal access to the system in various clinical situations.

As the DSS was developed as R&D prototype, it was not the aim, in this phase, to formally integrate it to the hospital record

system. This is a task for industrial partners when exploiting DSS as a commercial product. Once integrated with the hospital health record system, the DSS can be used in daily clinical practices, and allows clinicians to match a patient's individual data with injury characteristics. Thus, it facilitates efficient diagnosis, prognostication, and management decisions for individual patients with TBI by clinicians.

The DSS shows promise for providing computer-based decision support in TBI. Clinical feedback has been positive and the validation study has provided initial positive signals regarding the clinical performance of the system. In the presented implementation and evaluation we concentrated on clinical questions that deal with the shorter term management of patients: outcome prediction and patient state assessment that help in intervention planning in the hospital setting. Assessment of true prognostic performance would need a longitudinal set-up, which is the subject of ongoing research activities. The potential of the approach is wider though, and could be expanded to give input to decision support for long-term rehabilitation, e.g. at home.

To the authors' knowledge there exists no other systems that provide this kind of highly multi-modal data visualizations combined with evidence-based outcome predictions in TBI. This system allows clinicians to quickly interpret patient data and view probable outcomes, based on past patients in their own institution. Currently, the DSS does not support direct data entry and all the data must be fed into it from external databases. Databases differ from each other in terms of number of variables/attributes as well as data formats. Therefore, curation of datasets for the DSS required considerable effort and time. Therefore, future development work could focus on developing efficient ways for data preparation as well as incorporation of functionality that permits direct/automatic data entry into the system. While this paper provides an initial attempt at cross validation between Cambridge and Turku, the datasets used are not fully independent. Robust validation and calibration of the algorithms used to predict outcome need to test their performance on completely independent data in large studies, such as e.g. the EU funded CENTER-TBI project.

ACKNOWLEDGMENT

The authors thank all participants for their contribution to this project. Special thanks to clinicians of Addenbrooke's hospital, UK, and TUCH, Finland, for evaluating the tool and for providing useful feedback. Two clinical partners UCA and TUCH provided data for the tool.

REFERENCES

- [1] D. Menon, K. Schwab, D.W. Wright, A.I. Maas, and the Demographics and Clinical Assessment Working Group of the International and Interagency Initiative toward Common Data Elements for Research on Traumatic Brain Injury and Psychological Health, Position statement: definition of traumatic brain injury, *Arch. Phys. Med. Rehabil.*, vol. 91, pp. 1637–40, Nov 2010.
- [2] F. Tagliaferri, C. Compagnone, M. Korsic, F. Servadei and J. Kraus, A systematic review of brain injury epidemiology in Europe, *Acta Neurochir.*, vol. 148(3), pp. 255-268, Mar, 2006.
- [3] J. Ponsford, K. Draper, M. Schönberger, Functional outcome 10 years after traumatic brain injury: its relationship with demographic, injury severity, and cognitive and emotional status, *J. Int. Neuropsychol. Soc.*, vol. 14(2), pp. 233-42, Mar, 2008.
- [4] S. Dikmen, J. Machamer, J.R. Fann, N.R. Temkin, Rates of symptom reporting following traumatic brain injury, *J. Int. Neuropsychol. Soc.*, vol. 16(3), pp. 401-11, May, 2010.
- [5] H.F. Lingsma, B. Roozenbeek, E.W. Steyerberg, G.D. Murray, A.I. Maas, Early prognosis in traumatic brain injury: from prophecies to predictions, *Lancet Neurol.*, vol. 9(5), pp. 543-54, May, 2010.
- [6] K.E. Saatman, A.C. Duhaime, R. Bullock, A.I. Maas, A. Valadka, and G. T. Manley, Classification of traumatic brain injury for targeted therapies, *J. Neurotrauma.*, vol. 25(7), pp. 719-738, Jul, 2008.
- [7] N.A. Mushkudiani, C.W. Hukkelhoven, A.V. Hernández, G.D. Murray, S.C. Choi, A.I. Maas, E.W. Steyerberg, A systematic review finds methodological improvements necessary for prognostic models in determining traumatic brain injury outcomes, *J. Clin. Epidemiol.*, vol. 61(4), pp. 331–343, Apr, 2008.
- [8] E.W. Steyerberg, N. Mushkudiani, P. Perel, I. Butcher, J. Lu, G. McHugh, G.D. Murray, A. Marmarou, I. Roberts, J.D.F. Habbema, A.I. Maas, Predicting outcome after the traumatic brain injury: development and international validation of prognostic scores based on admission characteristics, *PLoS Med.*, vol. 5, e165, Aug, 2008.
- [9] H. Liedes, J. Mattila, H. Lingsma, J. Lötjönen, D. Menon, O. Tenovuo, M. Gils, Prediction of Outcome after Traumatic Brain Injury: Comparison of Disease State Index and IMPACT Calculator, *Stud. Health Technol. Inform.*, vol. 224, pp. 175-180, Jan, 2015.
- [10] J. Yoon, C. Davtyan, M. van der Schaar, Discovery and Clinical Decision Support for Personalized Healthcare, *IEEE J. Biomed. Health Inform.*, vol. 21(4), pp. 1133-1145, Jul, 2017.
- [11] J. Mattila, J. Koikkalainen, A. Virkki, M. Gils, J. Lötjönen, Design and application of a generic clinical decision support system for multiscale data, *IEEE Trans. Biomed. Eng.*, vol. 59(1), pp. 234-240, Jan, 2012.
- [12] J. Mattila, J. Koikkalainen, A. Virkki, A.H. Simonsen, M. Gils, G. Waldemar, H. Soininen, J. Lötjönen, A Disease State Fingerprint for Evaluation of Alzheimer's disease, *J. Alzheimers Dis.*, vol. 27, pp. 1–3, May, 2011.
- [13] Microsoft Patterns & Practices Team, *Architectural Patterns and Styles*, 2009. [Online]. Available: <https://msdn.microsoft.com/en-us/library/ee658117.aspx>.
- [14] Using a Three-Tier Architecture Model, 2013. [Online]. Available: [https://msdn.microsoft.com/en-us/library/windows/desktop/ms685068\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/ms685068(v=vs.85).aspx).
- [15] D. Betts, G. Melnik, F. Simonazzi, M. Subramanian, *Dependency Injection with Unity*, USA, Microsoft, 2013, ch. 2, pp.11-20.
- [16] D. Betts, J. Domínguez, G. Melnik, F. Simonazzi, M. Subramanian, *Exploring CQRS and Event Sourcing*, USA, Microsoft, 2012, ch.8, pp.201-210.
- [17] Microsoft Patterns & Practices Team, *The Repository Pattern*, 2009. [Online]. Available: <https://msdn.microsoft.com/en-us/library/ff649690.aspx>.
- [18] The Microsoft Dynamics AX Team, *How to: Use the UnitOfWork Class to Manage Database Transactions*, 2012. [Online]. Available: <https://msdn.microsoft.com/en-us/library/gg846338.aspx>.
- [19] L.S. Jadhav and B. A. Sodawala, Study of MVC Architecture and Implementation, *IJAR CET*, vol. 4 (6), Jun, 2015.
- [20] ASP.Net MVC 4 Overview, 2013. [Online]. Available: [https://msdn.microsoft.com/en-us/library/dd381412\(v=vs.108\).aspx](https://msdn.microsoft.com/en-us/library/dd381412(v=vs.108).aspx).
- [21] P. Vassiliadis, A survey of extract-transform-load technology, *IJDWM*, vol. 5(3), pp. 1-27, July, 2009.
- [22] H. Soininen, J. Mattila, J. Koikkalainen, M. Gils, A.H. Simonsen, G. Waldemar, D. Rueckert, L. Thurfjell, J. Lötjönen, Software Tool for Improved Prediction of Alzheimer's Disease, *Neurodegener Dis.*, vol. 10, pp. 149–152, 201, Dec, 2011.