

Combining Patient Pathway Visualisation with Prediction Outcomes for Chemotherapy Treatments

Agastya Silvina and Juliana Bowles

School of Computer Science
University of St Andrews
St Andrews KY16 9SX, UK
email: {as362|jkbfb}@st-andrews.ac.uk

Peter Hall

Edinburgh Cancer Research Centre
University of Edinburgh
Crewe Road South, EH4 2XR, UK
email: p.s.hall@ed.ac.uk

Abstract—The Edinburgh Cancer Centre (ECC) contains National Health Service (NHS) Lothian cancer patient data from multiple resources. However, the lack of proxy between numerous scattered resources hinders the capability to use the information collected in a useful way. ECC data is very varied and includes patient characteristics (e.g., age, weight, height, gender), information on diagnosis (e.g., stage, site, comorbidities) and treatments (e.g., surgery, chemotherapy, radiotherapy). The visualisation of a fraction of ECC data in the form of a patient timeline can aid and enhance the process of observing and identifying the overall condition of the patient, as well as understand how it may compare with cohorts of patients with similar characteristics. We have previously developed machine learning models for predicting treatment outcomes for breast cancer patient data that have undergone chemotherapy. In this paper, we describe, examine, and propose a solution to connect all these aspects and provide a bridge for several resources. This will make it easier for clinicians and other healthcare professionals to support service planning, deliver better quality of care and consequently improve service outcome within NHS Lothian.

Keywords—Distributed Health Data; Diagnosis; Treatment Timeline; Machine Learning; Oncology.

I. INTRODUCTION

Data is a precious asset in many organisations, but it is often too fragmented and crude to be useful. By integrating data from a variety of sources, we can gain considerably more insight from the information it contains. It is well known that understanding, analysing and building models from complex data has the potential to improve decision making. This is the case for many domains and is particularly relevant in the healthcare domain to improve cancer care.

Data integration is the process of combining/aggregating data from different sources to provide meaningful and valuable information to end-users. There are several ways to perform data integration [1], such as building an enterprise warehouse or creating a proxy server. In the case of the former, it corresponds to creating a centralised database that holds all the business information of an organisation and makes it accessible across the organisation. In the latter case, it corresponds to creating an application which provides the data to the end-users directly from various sources/servers.

Furthermore, the use of healthcare data and data mining/machine learning techniques enables us to develop models which can be used to observe, predict, and analyse the outcome of a specific cancer treatment (for instance, for chemotherapy [2]) by feeding patient characteristics, diagnosis and treatments into the learning algorithm.

The Edinburgh Cancer Centre (ECC) contains NHS Lothian cancer patient data from multiple resources [3]. The

Edinburgh Cancer Centre (ECC) is data-rich, however, the use of this resource for improving patient care and cancer treatment has so far been superficial and its potential under-explored. Cancer patient data is currently scattered across disparate systems and repositories. We show several components (some details purposely omitted) of the ECC system in Figure 1.

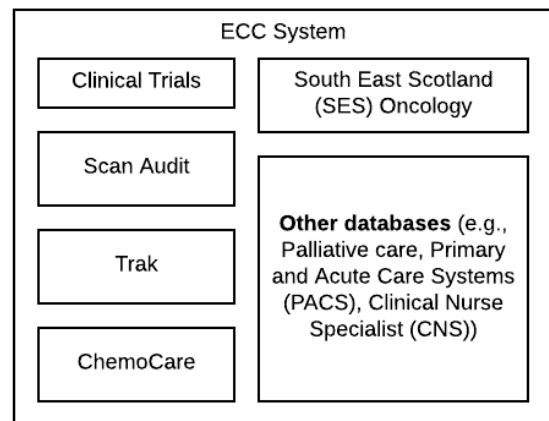


Figure 1. ECC cancer patient data sources

At the moment, it requires significant time and effort to carry out a manual retrospective analysis of data held across multiple systems [3]. In this paper, we describe the work carried out to provide a proxy between the different (sub)systems within ECC. The developed system, called *South East Scotland Oncology Gateway (SESO Gateway)*, can be seen as one analytical solution for improving the quality and capability of real-time outcomes reporting within South East Scotland using routinely captured and integrated electronic healthcare data. With this application, the users can view the pathway progress at an individual level (e.g., during the clinic and when deciding suitable treatments for a patient), at a cohort basis for analysing a treatment against a set of metrics (e.g., waiting time, following a particular pathway, outcome against disease management protocols) as well as examining the possible outcome of upcoming treatments.

This paper is structured as follows. We present related work in Section II, and describe the data source, analysis and the main features of the system in Sections III and IV. We illustrate the implementation of the system (i.e., front-end, back-end) in Section V. Lastly, we discuss the test results in Section VI and conclude the paper with suggestions for future work in Section VII.

II. RELATED WORK

SESO Gateway continues a line of work by the University of Edinburgh to discover predictors of chemotherapy side effects (e.g., toxicity) in breast cancer patients from a *ChemoCare* data extraction [4]. The data is extracted, transformed, and processed using multivariate logistic regression, as shown in Figure 2.

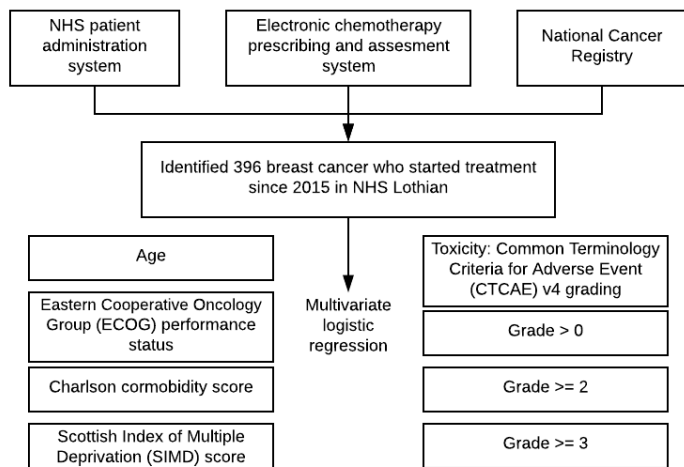


Figure 2. Example of the *ChemoCare* data processing methodology

This study aims to use the data to identify the toxicity rate from commonly used anti-cancer regimens for NHS Lothian patient populations, and find the baseline patient characteristics for predicting excess toxicity. It is possible to profile the pattern of cancer patients, which helps in predicting the side-effects of the chemotherapy regimen. We then add other features to predict and analyse the result of upcoming chemotherapy treatments by using machine learning models that have been developed previously [2].

The use of machine learning models and data visualisation helps users to understand the concept and underlying patterns of patient data. The data visualisation amplifies the benefits of health informatics databases and networks by expanding the capability of clinicians and public health policy-makers to make better decisions [1]. Indeed, multivariate applications like the *SESO Gateways* are becoming more widely developed in the healthcare domain (e.g., IBM Watson [5], Microsoft Research [6], NHS [7]). Integration with existing healthcare systems may help healthcare professionals make decisions, personalise cancer treatments and hence improve the treatment of their patients.

Several existing tools have inspired the development of the *SESO Gateway* (e.g., *LifeLine* [8], NHS Predict [7]). *LifeLine* is a visualisation tool to enhance navigation and analysis of patient records. It provides a general visualisation environment for personal histories. The prototype for this system was developed in 1997 as a research project between IBM Research and the University of Maryland. The basis for modelling the record was a newly operational clinical information system at Kaiser Permanente Colorado.

Predict [7] is an online prognostication tool that helps both patients and clinicians analyse how different treatments for early invasive breast cancer might improve the survival rates of patients after surgery. NHS England developed the tool by training the survival model against women with breast cancer data in England and has been tested on women with breast

cancer data from around the world.

III. USE CASE

We are developing a dashboard to help oncologists observe, monitor, and analyse the condition of their patients over time. We describe a hypothetical but realistic scenario of a user of the system as follows.

Emma is 38 years old and had been diagnosed with breast cancer. To prevent the spreading of the tumour, she underwent breast surgery. After surgery, chemotherapy treatment was given as a follow-up to her surgery. She comes to the hospital for her chemotherapy appointments. To ensure her recovery is as best as possible, a treatment plan and regimen have been established over several months with chemotherapy treatments in the hospital every three weeks. Emma also has a comorbidity. As any cancer patient on chemotherapy, she might have higher toxicity levels as a result, but it is crucial to guarantee that the scale does not go above level two. Toxicity levels range from 0 (no toxicity) to 5 (very high toxicity).

Emma agrees on using and sharing data between treatment visits via the cancer data gateway and patient portal. Emma determines who in the medical team sees this information: The oncologist/nurse and her GP. Emma is also informed about how to use the web application and pass on relevant information to the clinical team.

Via a user-friendly web application, Emma can provide information on symptoms daily throughout the treatment. Severe reported symptoms can be picked up by the clinical team and acted upon asap. The conditions that are being monitored and provided by patients are *nausea, vomiting, diarrhoea, constipation, oral mucositis, oesophagitis, neurotoxicity, hand/foot, skin, hypersensitivity* and *fatigue*. The information Emma provides is combined with further data on her patient characteristics, cancer information, hospitalisation details and comorbidity.

This combined data will help clinicians adapt treatments better to Emma as an individual patient, keep toxicity levels under control and improve her health outcomes. It uses data from several patients treated over the years with comparable characteristics.

If during the treatment there are signs that toxicity levels are high or that Emma's condition is deteriorating, one of the members of the clinical team (e.g., oncologist, specialist consultant, nurse, GP) will be alerted and contact Emma to intervene.

Overall, Emma can have a more personalised treatment. If a complication arises, the clinical team can act more quickly. Furthermore, Emma's well-being increases as she gets more involved in her treatment plans.

IV. DATA ANALYSIS

A. Data Source

As mentioned before, the ECC consists of several scattered databases. Usually, the databases are divided into different categories based on their function: Direct Clinical Care Databases (DCCD) for recording, observing and analysing the direct patient care and Secondary/Derived Databases (SDD) for auditing and reporting purposes.

Table I shows a brief overview of several DCCD and SDD databases. The system features the patient pathway at specific points. Hence, there is some transfer of information within each system (e.g., laboratory, radiology, chemotherapy).

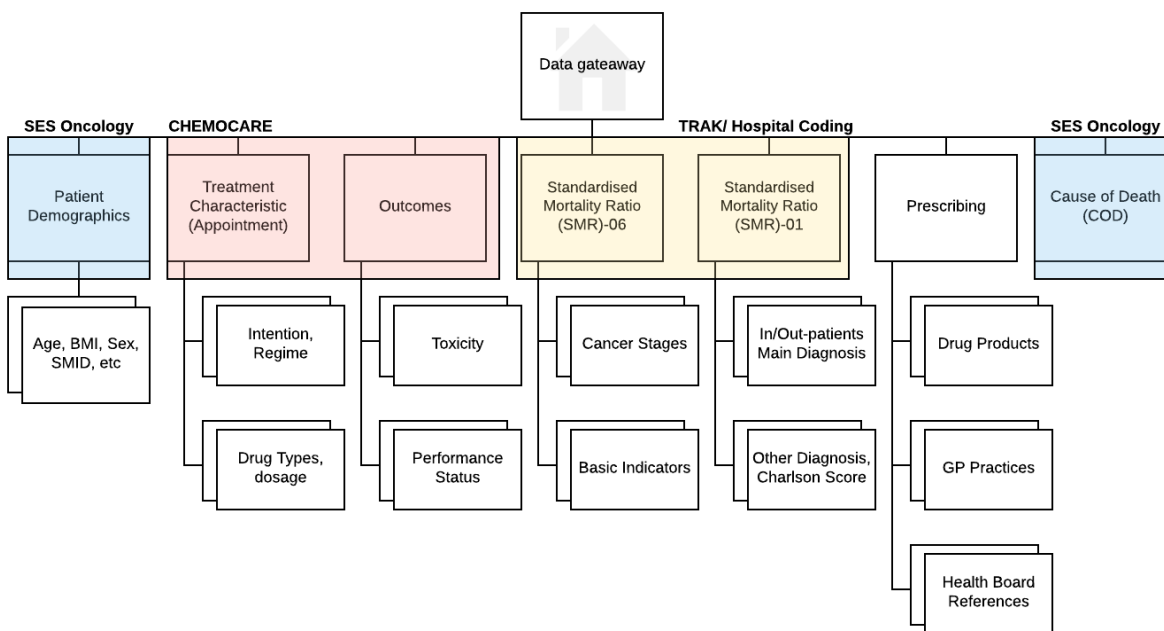


Figure 3. SESO Gateway data structure

TABLE I. ECC DATASOURCES

System/Team	Function
Trak	Patient Administration System (PAS) (DCCD)
PACS	Radiology Investigation System (DCCD)
ChemoCare	Chemotherapy ePrescribing System (DCCD)
CNS	Shared patient records (DCCD)
Trak Questionnaire	Ensure timely diagnosis and treatment (SDD)
Trak Module	Collect the quality performance indicator dataset (SDD)
Oncology Coding	Collect the whole patient pathway (SDD)
Hospital Coding	Collect the submission to national cancer registry (SDD)

However, because of the lack of proxy, the system has no cohesive view of the patient journey through cancer care. To facilitate this functionality, hospitals usually rely on several teams or manual input.

Figure 3 shows the data structure of the databases connected by the *SESO Gateway*. We developed the *SESO Gateway*, as a proxy server to connect two databases from the DCCD (i.e., *Trak* and *ChemoCare*) and one database from the SDD (i.e., oncology coding) to simplify the observation of the patients’ journey through their cancer care for the first version of our application.

B. Patient Information Dashboard

The *SESO Gateway* has two main components: the patient information dashboard and the prognostication tool. The information dashboard facilitates both individual patient analysis (by means of timeline visualisation) and cohort patient analysis.

1) *Individual Patient Analysis*: The patient timeline summarises the medical records of the patient as a set of lines and icons on an adjustable timeline as shown in Figure 4.

Figure 4 shows an example of patient data spanning about 5 years. The timeline is colour-coded based on the type of event (e.g., follow up, hormone treatment, chemotherapy treatment). The timeline has other features, such as zooming in/out, tab hiding, and scrolling as shown in Figure 5.

With a quick glance, the user can get the idea of the overall patient condition. If the user needs to see the detail of an event, the user can click on the icon. The detail of the event will be shown below the timeline.

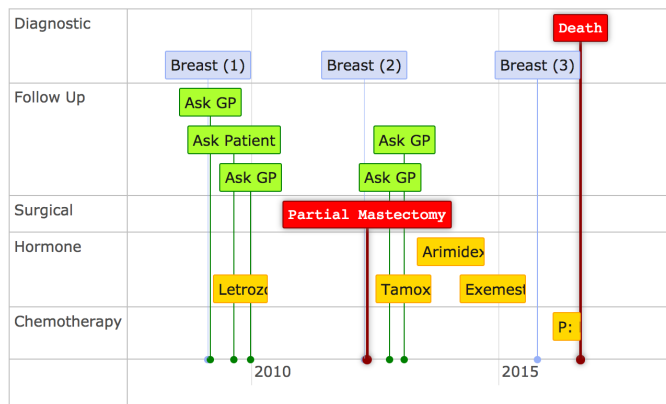


Figure 4. An example of individual patient timeline

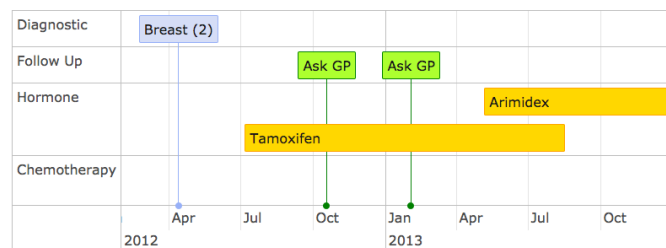


Figure 5. Zoom, hide, and scroll features

2) *Cohort Patients Analysis*: While several cancer treatments are proven to cure and reduce the (re)occurrence the cancer, the burden of drug-induced toxicity can be substantial. Therefore, there is a need to improve the ability to observe and predict the pattern of several different treatments. To improve the observation of drug effectiveness, we show cohort information in graphs/charts.

Figure 6 shows the wire-frame for the chemotherapy page. It consists of two tab components. The first tab shows the summary obtained from the Oncology DB while the second tab shows the cohort summary from the ChemoCare data extractions. As shown in Figure 7, the graphs show more complex clinical relations (for instance the surgeries that have

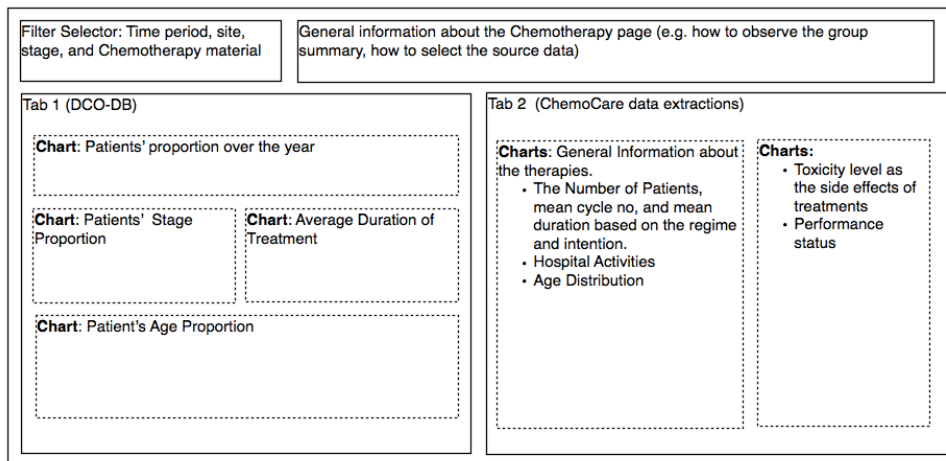


Figure 6. Chemotherapy page wireframe

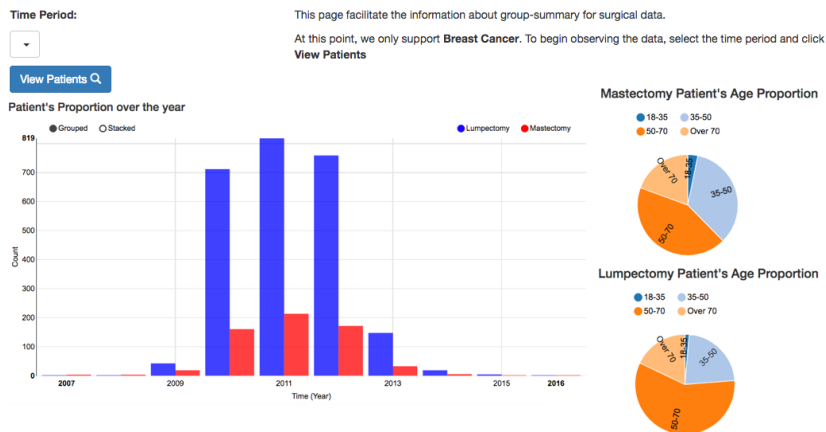


Figure 7. Surgical page

been done). They are useful when there is a point to be made in the shape of the data, or for showing how different things (variables) relate to each other [9]. It helps the user to observe the complex relationships, patterns and trends concisely. The *SESO Gateway* provides the users with graphs and charts for the patients' data within NHS Lothian. These components are shown in several different pages based on the type of treatments (e.g., hormone therapy, chemotherapy, radiotherapy, surgery). The clinical team (e.g., consultants, pharmacists, managers) can use this information for rapid auditing or scope out the data prior to more formal analysis.

C. Machine Learning Models

In our previous paper [2], we developed several models to predict the toxicity outcome of a chemotherapy treatment. We trained the models with the data extracted from *ChemoCare*, which contains the data for 13030 breast cancer treatments (with 933 unique patients). Each patient undergoes two or three different intentions and several regimes/protocols during their cancer care. Each regime might have several cycles (i.e., from 3 cycles to 85 cycles). Table II shows the number of patients in the dataset calculated based on the intention of treatment. Because of the lack of data for the curative intention, for the first version of *SESO Gateway*, we developed models for three different intentions, namely, Adjuvant, Neo-Adjuvant, Palliative.

We used the Random Forest (RF) model for our system. Our RF model performed better than the Hidden Markov

TABLE II. TREATMENT INTENTION

Intention	Total patients
Adjuvant	620
Neo-Adjuvant	427
Palliative	483
Curative	17

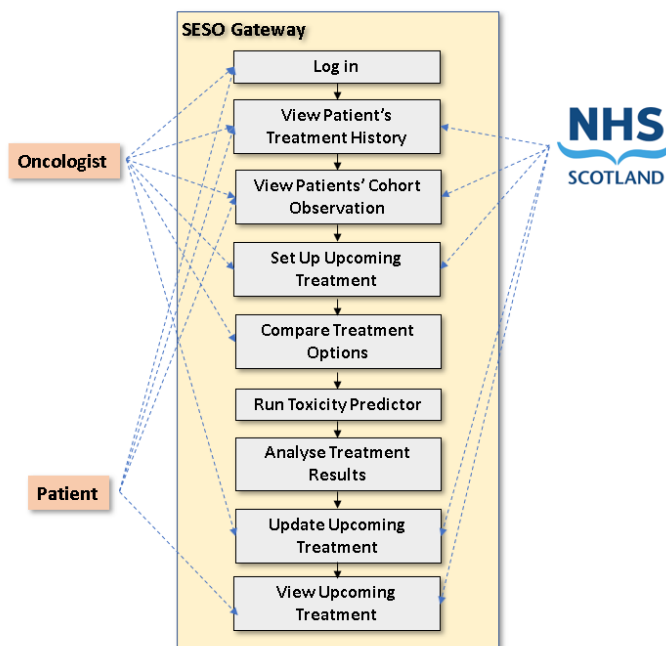
Model (HMM) models (i.e., especially in predicting the toxicity outcome during the patients' treatments) [2]. Although the Recurrent Neural Network (RNN) model performed better than the Random Forest model - mainly because of its capability to integrate all the previous treatments for predicting the treatment result - it is prone to over-fitting. This is due to the size of our dataset, as we presently do not have enough data to train the RNN models optimally [10].

V. IMPLEMENTATION

The *SESO Gateway* consists of three main components, the user interface (i.e., web-client), the software integration which acts as the proxy between each data source (i.e., Representational State Transfer Application Programming Interface (REST API) services), and the data sources (e.g., the *SESCD*, *Trak*, and *ChemoCare*). This separation contributes to the modularity of the system, which allows us to easily change each module for any future changes that may happen in the NHS infrastructure.

A. Front-End

The system presentation layer (i.e., Front-End) is a web-client developed using *AngularJS 7* framework [11]. The use of

Figure 8. *SESO Gateway* features

the framework helps the application to be more modular (i.e., reusable, Do not Repeat Yourself (DRY), readable, secure).

Our web-client facilitates the user to observe the patients history, cohort data from the NHS data sources. As shown in Figure 8, The web-client has three main features. It shows the event timeline for a particular patient during their treatments in the NHS. It allows the user to observe patterns for various treatments from the cohort observations. Lastly, the web-client has a feature to facilitate the users to analyse the outcome of the upcoming treatments by using the machine learning models that were previously developed.

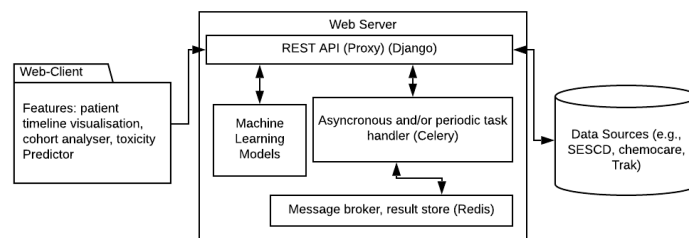
We implement the *Model-View-View-Model* (MVVM) pattern for the *SESO Gateway* web-client. The MVVM separates the development of the GUI (Graphical User Interface) with the business logic or the data model. Each functionality is separated into different modules. These modules are being integrated into a root module called *Apps*. In *Apps*, we register the built-in *AngularJS* modules (e.g. *ngRoute*, *ngResources*) as well as the custom modules that we create (e.g. *event-timeline*, *group-detail*). It contains the main controller for the *SESO Gateway* client. With the main controller, we generate a Single Page Application (SPA) by only updating specific divisions on the client. We register several *urls* that can be accessed by the users in the *Apps*.

We separate the modules for the individual timeline, the upcoming treatments analysers, and the cohort/group summaries. With this separation, we have loosely coupled components. Therefore, when we need to change the behaviour of one of the module, we do not need to update any components in the group summaries or the upcoming treatments predictor module.

For the cohort reporting service, we create several different modules/pages based on the type of treatments (e.g. surgical, radiotherapy, hormone therapy, chemotherapy) registered in the *SESO Gateway* for the cohort summary. These modules have a similar structure. For each treatment, we have a module which contains methods to allow the users to apply several filters and

then fetch the data from the REST API. The same principle applies to analyse the upcoming treatment feature. Once the data is fetched, we use a third-party library for generating the data visualisation (i.e. *D3* [12]).

B. Back-End

Figure 9. *SESO Gateway* system architecture

The *SESO Gateway* back-end system is a REST API service. The system has an asynchronous and periodic task handler and an internal database as the message broker and asynchronous requests' result storage, as shown in figure 9.

The REST API service of the *SESO Gateway* acts as a proxy between the web-client and the data sources. This uses several database engines to connect a different kind of data sources.

The application consists of several isolated sub-application (i.e., modules) based on the database access and main functionality (e.g., *sescd*, *chemocare*, *toxicity_predictor*). The *sescd* module handles the REST API request for the *SESCD* database, while the *chemocare*, as the name suggests, processes the request for the *ChemoCare* access. Each module has its own REST API for their appropriate function and database access.

To facilitate the functionality to predict the outcome of the upcoming treatment, we serialise the machine learning models which will be loaded the moment the server is up and running. We create a separate API for the user to request an analysis of the upcoming treatment. We develop both *POST* and *GET* request to facilitate this. To analyse the treatment, first the user performs a *POST* request. In the request body, the user specifies the treatment's parameters, such as patient characteristic (e.g., age, weight, height, cancer stage and site), treatment intention (e.g., *adjuvant*), treatment regime (e.g., *FEC 80*) and duration (e.g., 3 months). The request is processed asynchronously by the asynchronous task handler as it might require a longer time to be processed (i.e., in comparison with requesting the patient's medical history). The handler puts the request in the tasks queue and tags it with a unique *task id*. The *task id* is later given to the user/client as the response of the *POST* request. Once the *SESO Gateway* finishes predicting the result of the upcoming treatment, the outcome of the prediction is saved in the result storage which can be retrieved by the client via a *GET* request with the *task id* as the query parameter if the task is complete.

VI. CASE STUDY: FEEDBACK AND EVALUATION

A. Testing

To evaluate the *SESO Gateway*, we conducted different kinds of testing, organised by the level and purpose of the tests themselves. For the *SESO Gateway*, this included unit-testing, integration-testing and performance-testing.

Our unit tests are performed on the developed module to clarify if the method performs as expected in a set of

conditions. There are several different purposes for our unit tests. First, we use them to check the correctness of our queries to retrieve patient data or cohort information from an appropriate data source. We also use it to check if we load the correct machine learning model for our upcoming treatment analyses request. Lastly, we also use it for utility methods. Unlike for other forms of testing, we create several mock objects for our unit testing.

For the *SESO Gateway* integration testing, we combine several modules within the application and test those modules as a group. The integration testing takes the modules that have been tested previously in the unit testing. Our integration testing focuses on examining the pathway from the first time the client makes a request to retrieving the correct data from the databases and returning the right response to that request. Similarly, we use the integration testing to simulate the process of the user/client requesting to analyse the upcoming treatment.

Figure 8 shows the potential users of the system. We estimate the number of users between 0 to 150 users accessing the application at the same time. For testing the performance of the *SESO Gateway*, we assume that there could be a time when all users try to access the application simultaneously. Based on this assumption, we need to perform a performance test to observe the robustness of our system.

One issue of *go-live* is performance, such as latency from highly sensitive and slow responses. The performance test allows us to simulate a situation where many users try to access our application at the same time. We use a third party library (i.e., *Gatling-tool* [13]) to perform such a performance test. The tool has the support for HTTP-Protocol and is asynchronous as long as its underlying protocol can be implemented in a non-blocking way. Henceforth, it allows us to do the load-testing of the *SESO Gateway*'s HTTP Server by creating hundred of virtual users.

The *SESO Gateway* is capable of handling 150 simultaneous requests from the user, as requested by the customer, but there is no reason why it cannot process more requests. However, we have not performed any performance test for the system against a million of patient data. The result of performance testing will be highly dependant on the server we access and the environment where we deploy our application.

B. System evaluation

During the development process, we conducted several presentations and demonstrations of the *SESO Gateway* to different users and clinical oncologists. We also conducted a user evaluation of our application. This process allows us to evolve the *SESO Gateway* application to target more concretely the needs and requirements of the end users. Based on users' feedback, we expanded the *SESO Gateway* accordingly, we updated the timeline by adding group functionality as well as better colour-codings after a demonstration to an oncologist. We decided to change the colour of individual timelines and add groups for different kinds of events (e.g., *Diagnosis*, *Surgery*).

One of the oncologists expressed how having an extensive filter for the cohort functions (e.g., filter the patients based on their oncologist which handle the patient) could make the observation faster. We will implement the functionality for generating the list of patients used in the cohort graph in the future because it can further ease the auditing process.

VII. CONCLUSION

Our primary goal is to design a method to accurately visualise patient pathway data from the South-East Scotland Oncology Database. The users can view the pathway progress at an individual level (for example in clinic) as well as a cohort basis to analyse a treatment pathway against a set of metrics (e.g., time between treatment, following a particular pathway, outcome against disease management protocols). With *SESO Gateway*, we have a system which enhances observations for cancer patients in South East Scotland by incorporating a data visualisation tool with additional capabilities to analyse the upcoming treatments. The integration to the new database enables users to effectively see connections and patterns, between patient treatments and their condition, in real-time. Finding these correlations among the data is essential to improve the healthcare systems for ECC.

ACKNOWLEDGEMENT

This research is supported by the NHS Lothian, the DATA Lab, and the EU H2020 project *Serums: Securing Medical Data in Smart Patient-Centric Healthcare Systems* (grant code: 826278). We thank Christina Lilley for her support and contribution to the development of *SESO Gateway*.

REFERENCES

- [1] B. Shneiderman, C. Plaisant, and B. W. Hesse, "Improving Healthcare with Interactive Visualization," *Computer*, vol. 46, no. 5, May 2013, pp. 58–66.
- [2] A. Silvina, J. Bowles, and P. Hall, "On Predicting the Outcomes of Chemotherapy Treatments in Breast Cancer," in *Artificial Intelligence in Medicine*, D. Riaño, S. Wilk, and A. ten Teije, Eds. Cham: Springer International Publishing, 2019, pp. 180–190.
- [3] C. Lilley, "Cancer information Programme Plan NHS Lothian," NHS Lothian, 2016.
- [4] Z. H. Ng and P. Hall, "Using real-world data to discover predictors of chemotherapy toxicity in breast cancer patients," University of Edinburgh, 2016.
- [5] J. L. Malin, "Envisioning watson as a rapid-learning system for oncology," *Journal of Oncology Practice*, vol. 9, no. 3, 2013, pp. 155–157, pMID: 23942497. [Online]. Available: <https://doi.org/10.1200/JOP.2013.001021>
- [6] "Project Hanover," 2019, URL: <https://hanover.azurewebsites.net/> [accessed: 2019-23-09].
- [7] G. Wishart, E. Azzato, D. Greenberg, J. Rashbass, O. Kearins, G. Lawrence, C. Caldas, and P. Pharoah, "PREDICT: A new UK prognostic model that predicts survival following surgery for invasive breast cancer," *Breast cancer research : BCR*, vol. 12, 01 2010, p. R1.
- [8] C. Plaisant, R. Mushlin, A. Snyder, J. Li, D. Heller, and B. Shneiderman, "Lifelines: using visualization to enhance navigation and analysis of patient records," *The craft of information visualization*, 01 2003, pp. 308–312.
- [9] S. Powsner and E. Tufte, "Graphical summary of patient status," *The Lancet*, vol. 344, no. 8919, 1994, pp. 386 – 389, originally published as Volume 2, Issue 8919. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0140673694914060>
- [10] M. Z. Alom, T. M. Taha, C. Yakopcic, S. Westberg, P. Sidike, M. S. Nasrin, M. Hasan, B. C. Van Essen, A. A. S. Awwal, and V. K. Asari, "A State-of-the-Art Survey on Deep Learning Theory and Architectures," *Electronics*, vol. 8, no. 3, 2019. [Online]. Available: <https://www.mdpi.com/2079-9292/8/3/292>
- [11] "AngularJS," 2019, URL: <https://angular.io/> [retrieved: 09, 2019].
- [12] "D3.js-Data-Driven Documents," 2019, URL: <https://d3js.org/> [retrieved: 09, 2019].
- [13] "Gatling Open-Source Load Testing," 2019, URL: <https://gatling.io/> [retrieved: 09, 2019].