



*Platform for European Medical Support
During Major Emergencies*

D4.4 Surge Capacity Tool





PULSE

Platform for European Medical Support during major emergencies

WP4 Tools

Deliverable D4.4 - Surge Capacity Tool

30/11/2015



Contractual Delivery Date:	Actual Delivery Date:	Deliverable Type*-Security**:
30/11/2015	30/11/2015	R - PU
607799	PULSE	Platform for European Medical Support during major emergencies

*Type: P: Prototype; R: Report; D: Demonstrator; O: Other.

**Security Class: PU: Public; PP: Restricted to other programme participants (including the Commission); RE: Restricted to a group defined by the consortium (including the Commission); CO: Confidential, only for members of the consortium (including the Commission).

Responsible:	Organisation:	Contributing WP:
Francesco Malmignati	Selex ES	WP4

Authors (organisation)
Alessandro Borri (UCSC), Andrea De Gaetano (UCSC), Sabina Magalini (UCSC), Francesco Malmignati (SES)

Abstract:
<p>The purpose of this document is to provide a synthetic supporting document about the software prototype of Deliverable D4.4 "Surge capacity tool" of WP4 "Tools".</p> <p>The document describes the tool aiming at providing support for the creation of surge capacity in the event of a major health crisis focusing on the availability of critical resources and the prediction of their evolution. The tool will be designed for use both in the preparedness phase and in the response phase.</p>

Keywords:
Surge capacity tool, Health Care Facilities Model, Stadium Crush scenario, SARS scenario



Revisions:

Revision	Date	Description	Author (Organisation)
0.1	05/11/2015	First contribution	Alessandro Borri (UCSC), Sabina Magalini (UCSC), Andrea De Gaetano (UCSC)
0.2	10/11/2015	Integration of the SES contribution	Francesco Malmignati (SES)
1.0	19/11/2015	Final version to be reviewed	Francesco Malmignati (SES)
1.1	26/11/2015	Internal review	Paul Kiernan (SKY)
1.2	27/11/2015	Addressing internal review comments	Alessandro Borri (UCSC), Francesco Malmignati (SES)
1.3	27/11/2015	Final version	Francesco Malmignati (SES)



Table of contents

Revisions:	3
List of figures	5
List of Tables	6
1 List of acronyms	7
2 Executive Summary	8
3 Introduction.....	9
3.1 Hospital Surge Capacity	9
3.1.1 Classification of Hospital Surge Capacity	9
3.1.2 Existing models of Hospital Surge Capacity	10
3.1.3 Creation of Surge Capacity through Early Discharge	10
3.1.4 Management of Logistics and Human Resources.....	10
3.2 Scope of the Document	12
3.3 Structure of the Document.....	12
3.4 Relations with other Deliverables	13
4 Tool description	14
4.1 Objective and general description	14
4.2 Functionalities	14
4.3 Relations with WP2 Use Cases.....	17
5 Architecture	18
5.1 Software Component Architecture.....	18
5.1.1 SCGT Engine	18
5.1.2 GUI client.....	21
5.2 Component relation to overall PULSE architecture	22
6 Component Technologies	24
6.1 List of core technologies	24
6.2 3 rd Party libraries and licenses.....	24
7 Software Test/Validation	25
References	26
Appendix A	27
A.1 WSDL.....	27



List of figures

Figure 1: Input parameters	19
Figure 2: Output parameters	19
Figure 3: WSDL Messages	20
Figure 4: WSDL Operations	20
Figure 5: Scheme of the client-server implementation of tool	21
Figure 6: Java Client example.....	22
Figure 7: PULSE Platform Architecture - Component diagram	23



List of Tables

Table 1: Levels of Capacity	9
Table 2: Critical resources to be increased during a Major Emergency in case of Stadium Crush scenario	15
Table 3: Critical resources to be increased during a Major Emergency in case of SARS scenario	16
Table 4: 3rd party libraries and licenses	24



1 List of acronyms

Acronym	Definition
DSVT	Decision Support Validation Tool
ED	Emergency Department
EMS	Emergency Medical System
GUI	Graphical User Interface
HT	Hospital Types
ICU	Intensive Care Unit
ID	Infectious Diseases
JVM	Java Virtual Machine
SCGT	Surge Capacity Generation (support) Tool
SOAP	Simple Object Access Protocol
US	United States
WSDL	Web Services Description Language



2 Executive Summary

The aim of the deliverable D4.4 is to describe the Surge Capacity Generation Tool (SCGT), one of the software components composing the PULSE Platform Architecture.

The tool's objective is to provide support for the creation of surge capacity or, in other words, the expected evolution of some critical medical resources during a major health crisis. The tool accepts as input (1) the number of people involved in the crisis scenario and (2) a desired prediction interval. It returns the amount of resources (depending on the specified number of people) which is possible to make available within the prediction interval.

Legal implications, cross-border support and the possible use of volunteers can be taken into account by means of a variation of the nominal parameters of the prediction model. This could be done off-line, in the preparedness phase.

At the basis of the Surge capacity tool resides the Health Care Facilities Model (T3.3 of WP3), which has been deeply described in D3.1.

The SCGT provides its functionalities through a SOAP-based Web Service and the *Decision Support and Validation Tool (DSVT)* is the only architectural component that directly interfaces with it; the DSVT invokes the SCGT interface during the crisis simulation process in order to obtain the possible variation of the required resources.



3 Introduction

3.1 Hospital Surge Capacity

Hospital Surge Capacity is defined as “a term of art referring to the ability of a hospital in a mass casualty incident to augment bed availability by maximising resources and discharging as many patients as safely possible” [1].

Disaster medicine literature is very often anecdotal being it based on single offer different cases of disasters and catastrophes. The many variables, the availability of data and the absence of parameters make it very difficult to deal with surge capacity in hospitals. This makes it so that most of the models of surge capacity are in reality based on “generic” management systems.

In reality it is very difficult to understand exactly what surge capacity is. There has been a recent recommendation to express hospital surge capacity as and hospital bed surge capacity as rates and define them as the maximum number of critically or moderately injured patients for whom a hospital can adequately care during a defined period of time. The upper limit of hospital surge capacity is conceptually the beginning of hospital acute care disaster surge capacity, above which patient care deteriorates exponentially even after recruitment of assets. This because hospital acute care surge capacity is not infinite and is limited not only by equipment and staff but also by access to services.

3.1.1 Classification of Hospital Surge Capacity

In Major Incidents hospital surge capacity is the core of the total medical capacity of the community. A taxonomy system of surge capacity was proposed by Hick et al. [2]

In Table 1 this is confronted with the more classical Level of EMS (Emergency Medical Systems) response at European Level as classified by Lennquist [3].

Table 1: Levels of Capacity

Conventional capacity (implemented in major incidents and representing care as usually provided at the institution)	Level 1: by adjusting organization and methodology, we can maintain the level of ambition of our medical care and save normally salvageable patients, also called major incidents, major emergencies, compensated incidents
Contingency capacity (using adaptations to medical care spaces, staffing constraints, and supply shortage without significant impact on delivered medical care)	Level 2: load of casualties is so high that even by adjusting organization and methodology we cannot maintain the level of ambition of our medical care and save normally salvageable patients, also called Mass Casualty Incidents, disasters, decompensated incidents.
Crisis capacity (implemented in catastrophic situations with a significant impact on standard of care)	Level 3: similar to level 2 including destruction of the infrastructure in a region or in a country; requires external assistance, also called complex



	emergencies or compound incidents
--	-----------------------------------

3.1.2 Existing models of Hospital Surge Capacity

De Boer proposed that each hospital should care under regular circumstances for a number of critically and moderately injured patients equivalent to 0.5-1 patient per 100 hospital beds per hour. With the activation of the hospital surge plan this number rises to 2-3/100 hospital beds per hour for a maximum of 8 hours [4].

Hirshberg et al did a modeling on data available from the Ben Taub Hospital (Houston, TX) regressing the number of critically ill patients per hour on the mean level of care (made dependent on 6 variables). They defined the adequate level as a mean level of care of 90 % or over of the optimal care (care of a single critical trauma patient during normal daily operations). They established that the cut off for their hospital was approximately a casualty load of 50 [5].

Another group (Bayram et al) modelled hospital acute care surge capacity to be equal to the number of emergency department beds, divided by the emergency department time that in Major Incidents (trauma related) is approximately 2.5 hours. Since trauma patients reach the Emergency Department (ED) in 6 hours (by definition), the hospital bed surge capacity must match the hospital acute care surge capacity at 6 hours to ensure coordinated care, the rate was mathematically benchmarked to be 18 % of the staffed hospital bed capacity [6].

These proposed models can provide framework for conducting a systematic and comprehensive estimation of surge capacities of hospitals, which could be made available to all concerned parties, including prehospital and hospital managers and directors. This task can be lead by public health departments because all-hazards preparedness is one of their emerging roles. Comprehensive data banks for hospital surge capacity of hospital within each local, regional, multiregional and national level can be shared among neighboring countries to coordinate efforts in the international cross-border events.

3.1.3 Creation of Surge Capacity through Early Discharge

US hospitals are expected to function without external aid for up to 96 hours during a disaster. Creation of Surge Capacity through early discharge (reverse triage) is a very important method to create Hospital surge capacity and represents the newest health policy approach in these situations.

Usually patients that do not require any previously defined critical intervention for 4 days are deemed suitable for early discharge. In a recent study on 3 capacity-constrained hospitals it was shown that 44% of patients do not require any critical intervention and are suitable for discharge, that added to routine patient discharges, full use of staffed and unstaffed beds (were present), gross surge capacity can be estimated at 66%-81%. This can be done in 24-48 hours [7].

3.1.4 Management of Logistics and Human Resources

Surge capacity is defined by 3S system: Staff (personnel), Stuff (supplies and equipment), Structure (physical structure/location plus management structure).

Staff: Maintaining adequate number of personnel with appropriate expertise is



challenging even in basal conditions and becomes more so during a disaster. A percentage of the regular staff may be unable or unavailable or unwilling to report to work, so it is important to have a system for emergency credentializing of volunteers.

Another issue with surge capacity is that human behavior or psychosocial influences may override duty in face of disaster. Some staff will have concerns with families or property, may be victims of disaster, may fear contamination. Another challenge is called “convergent volunteerism”. These volunteer may not be adequately trained and know how to protect themselves. Well meaning people may be a burden.

Stuff: A variety of supplies and equipment are needed to support surge capacity. These include cots/litters, bedding, medical-surgical supplies, oxygen, pharmaceuticals and sanitation supplies. Pre-event planning to acquire and store these items is necessary whether it be on the hospital grounds or held on standby by a vendor. Some hospitals have acquired excess inventory by slowly buying the supplies over time and then rotating them into normal hospital operations. Other hospitals have stored supplies in trailers or cargo containers.

Structure: while a catastrophic event will certainly cause shortages in staff and supplies, in many cases it is the lack of coordination and communication that hinders effective disaster response. Therefore structure includes two components:

- a physical structure of location to care for patients (either within the hospital or at alternate care center) and
- the incident management structure to organize and manage the overall event.

Given that different kinds of threats and events require different responses and identified the accident types according to the well known terminology of sudden impact event (big bang) or an obscure event or one that progresses slowly or chronic disaster (rising tide), the evaluation of hospital capacity to respond to surge depends on the hospital capacity to focus on “patient care capacity” and this is difficult to be achieved by standardized methods. At minimum a set of metrics should be available from hospitals and may be utilized by modeling systems:

- number of inhabitants of the referral region
- number of hospitals
- numbers of EDs
- number of hospital beds
- percentage of beds occupation
- number of ED visits (hour/day/week/month)
- total number of empty occupied ED beds
- total number of occupied overflow ED beds
- total number of patients in the waiting room
- acuity level of triaged waiting room patients
- acuity level of ED patients
- number of intensive care beds
- total number of occupied and available intensive care unit beds
- number of physicians, nurses, pharmacist, porters
- number of ED nurses
- number of emergency physicians
- number of ED patients waiting for inpatient beds
- mean/median overall length of stay (for patients in licensed and overflow beds)
- mean/median waiting room times
- mean/median boarding times



More in detail:

- number of physicians on call or possibly to be on call and time to be available (hour/day/month)
- number of nurses on call or possibly to be on call and time to be available (hour/day/month)
- number of pharmacists on call or possibly to be on call and time to be available (hour/day/month)
- number of porters and technicians on call or possibly to be on call and time to be available (hour/day/month)

As regards the staff and structures, we should have an approximate evaluation of hospital resources considering the ones which will be likely to be scarce in a crisis care environment, possibly justifying specific planning and tracking.

These include:

1. Ventilators and components
2. Oxygen and oxygen delivery devices
3. Intensive care unit patient care capacity
4. Health care providers, particularly critical care, emergency medicine, burn and surgical/anesthesia staff (nurses and physicians) and respiratory therapists
5. Hospitals and related health care facilities (because of infrastructure damage or compromise)
6. Speciality medications, intravenous fluids, blood products (es. Sedatives/analgesics, specific antibiotics, antivirals and vaccines)
7. Medical transportation

Specific important points to be known are:

1. the total number of operating rooms
2. the total number of surgical beds divided by medical specialities
3. the total number of surgical beds for emergencies.

3.2 Scope of the Document

This document is a covering document to support the software delivery D4.4 – Surge Capacity Tool.

This report summarizes the software component delivery and provides high level details on the architecture, technologies and underlying libraries on which the software component has been developed.

3.3 Structure of the Document

This document is structured into the following sections.

- Introduction to the Surge Capacity concept.
- Description of the *Surge Capacity Generation Tool* objective.
- Main functionalities of the SGCT.
- Description of the *Surge Capacity Generation Tool* internal architecture and relation of this component with other elements of the PULSE platform architecture;



- List of technologies adopted for the implementation of the *Surge Capacity Generation Tool*
- List of underlying 3rd party libraries used for *Surge Capacity Generation Tool* implementation and summary of the corresponding licenses
- Software Test/Validation

3.4 Relations with other Deliverables

The work presented in this deliverable is related to the following WP2 and WP4 deliverables:

- D2.2 - Use case specification [8] – This document describes the use cases related to two PULSE reference scenarios: SARS-like epidemics and Stadium crush.
- D4.1 – Decision support Validation tool [9] – It is the tool that invokes the API provided by the Surge Capacity Generation tool. The DSVT provides a GUI where the user can actually exploit the functionalities exposed by the SCGT.



4 Tool description

4.1 Objective and general description

The objective of the *Surge Capacity Generation support Tool* (SCGT) is to provide the expected evolution of some critical resources during a major crisis. Two distinct scenarios (Stadium crush and SARS epidemics) are evaluated, in order to test the tool in the different contexts of trauma events and biological events, respectively.

Given the number “num_affected” of the people involved in the crisis scenario and a desired prediction interval “delta_t”, the tool returns the prediction of the amount of the resources at time $t_0 + \text{delta}_t$ (the level of the resource at time t_0 is assumed to be known).

The input/output structure of the service is the following:

- **INPUT**
 - Number of affected num_affected, natural number.
 - Prediction time step delta_t, nonnegative real number [unit of time].
- **OUTPUT**
 - Prediction of the provision of the resources at time $t = t_0 + \text{delta}_t$.

4.2 Functionalities

The tool provides is applied to different sets of facilities in the Stadium Scenario (trauma) and in the SARS Scenario (epidemics). Since the tool matches the corresponding Health Care Facilities Model in WP3, we here recall that the prediction of the future availability of resources depends on the category of the medical structure considered.

In the case of Stadium Crush scenario, the following Hospital Types (HT) are considered:

- Trauma Center high level;
- Trauma Center low level;
- General Hospital;
- Specialized Hospital (Burns, Pediatric, Neuro, Cardio);
- First Aid (without beds).

For each category of hospital, it is possible to evaluate the availability (in unit of resource) of the following resources:

- Triage [patients/h];
- Emergency Room [number beds];
- Intensive Care Unit (ICU) [number of beds/function];
- General Surgery [number of teams in house/on call];
- Cardio Surgery [number of teams in house];
- Neuro Surgery (NCH) [number of teams in house];
- Emergency & Trauma Surgery [number of teams in house];
- Drugs/Devices [% of the basal value];
- Ambulance [number];



- Radiology [basic/standard];
- Radiology TC [number];
- Radiology ANGIO [number];
- General ward [number of beds];
- Operative Theater [number];
- Blood bank [level of function 1-5] ;
- Emergency Department [number treated patients/h].

In the case of SARS scenario, the considered Hospital Types (HT) are the following:

- Level 1 Infectious Disease (ID) Specialized Hospital;
- Level 2 Infectious Disease (ID) Specialized Hospital;
- Level 1 General Hospital;
- Level 2 General Hospital;
- General Practitioner Clinic.

For these categories of hospital, it is possible to evaluate the availability (in unit of resource) of the following resources:

- ID ward;
- High level isolation unit;
- ID specialist;
- General Practitioner;
- Infection control protocol for SARS;
- Treatment Protocol for SARS;
- ID specialist on call 7/24 (guard);
- ID specialist on call.

We recall that the following attributes need to be estimated for each resource of each HT:

- Basal level (L_0) [in resource units] (level in normal, non critical conditions);
- The time in hours (t_{50}) required to reach the 50% of the maximal increment of the resource with respect to the beginning of the crisis (t_0).

Table 2 and Table 3 (taken from D3.1 [10]) report two examples of a subset of possible resources to be incremented to meet the needs during a Major Emergency in case of Stadium Crush and SARS scenario respectively (estimated levels are given for Trauma Center high level and for Level 1 Infectious Disease (ID) Specialized Hospital).

Table 2: Critical resources to be increased during a Major Emergency in case of Stadium Crush scenario

Resource name	Unit	Basal Level (L_0) [in resource units]	Time t_{50} [h]
Triage	patients/h	2	20



Emergency Room	number of beds	2	6
Intensive Care Unit (ICU)	number of beds/function	4	70
General Surgery	number of teams in house/on call	1	1
Cardio Surgery	number of teams in house	2	1
Neuro Surgery (NCH)	number of teams in house	2	1
Emergency & Trauma Surgery	number of teams in house	2	1
Drugs/Devices	% of the basal value	1	100
Ambulances	number	1	1
Radiology	basic/standard	0.5	1
Radiology TC	number	0.5	3
Radiology ANGIO	number	1	1
General ward	number of beds	5	200
Operative Teathers	number	0.5	2
Blood bank	function level 1-5	1	5
Emergency Department	number treated patients/h	2	10

Table 3: Critical resources to be increased during a Major Emergency in case of SARS scenario

Resource name	Unit	Basal Level (L ₀) [in resource units]	Time t ₅₀ [h]
ID ward	number of beds	40	2
High level isolation unit	number of beds	2	N/A
ID specialist	number	3	1
ID specialist on call 7/24 (guard)	number	1	2



ID specialist on call	number	1	2
-----------------------	--------	---	---

4.3 Relations with WP2 Use Cases

Part of the work carried out in WP2, was focused on the definition of a set of use cases for PULSE that have been detailed in the deliverable D2.2 [8]. In the following paragraph, we will try to associate the SCGT's functionalities with the use cases where the SCGT is actually involved.

The Intelligence Analysis Tool has a strong relation with just one of the use cases defined in D2.2, specifically the *UC – STADIUM CRUSH -04* where the PULSE platform provides summarised information to support decision making (e.g. Hospital controllers, regional authorities and crisis management teams) in regard to hospital admissions. In this case, the Surge Capacity Generation Tool, is invoked by the DSVT (see [9]) to obtain the possible evolution of the hospital resources during the crisis.



5 Architecture

5.1 Software Component Architecture

The *Surge Capacity Generation* tool is constituted by a sub-module called SCGT Engine that implements a Web Service equipped with a SOAP-based interface. This interface provides the same functionalities defined for the WP3 prototype (described in D3.1 [10]) and that are based on the context models defined within the same workpackage. More info can be found in 5.1.1.

An external GUI client has been also defined to facilitate the development and testing of the SCGT Engine prototype. A short description of this client is provided in section 5.1.2.

5.1.1 SCGT Engine

The SCGT Engine has been implemented in PHP language, and provides a Web Service interface based on the SOAP protocol. In such a way, an external client (e.g. the DSVT in the PULSE context) is able to send requests even in another programming language (i.e. Java) with respect to the server, though respecting the SOAP specifications.

The SCGT interface is described through the Web Services Description Language (WSDL) and is located at the following URL: <http://biomat1.iasi.cnr.it/webservices/pulse/M18/webservice.wsdl> (hosted in the CNR/IASI server). You can find the WSDL file in Annex A.1.

The WSDL file contains the definition of complex types, elements, messages and operations for the Surge Capacity Tool.

Figure 1 shows the input parameters defined into the WSDL file. The keyword "Parameters" (in `facilitiesModelParametersStadium` and `facilitiesModelParametersSARS`) is used to define the input of the server function, matching (in number and type) the input described in section 4.1.



```
<complexType name="FacilitiesModelParametersSARS">
  <sequence>
    <element name="num_patients" type="xsd:integer" maxOccurs="1"/>
    <element name="delta_T" type="xsd:double" maxOccurs="1"/>
  </sequence>
</complexType>
<complexType name="FacilitiesModelParametersStadium">
  <sequence>
    <element name="num_patients" type="xsd:integer" maxOccurs="1"/>
    <element name="delta_T" type="xsd:double" maxOccurs="1"/>
  </sequence>
</complexType>
  <complexType name="ListOfFacilitiesStadium">
    <sequence>
      <element name="facilityStadium" type="xsd:integer" maxOccurs="unbounded" />
    </sequence>
  </complexType>
  <complexType name="FacilitiesLevelsStadium">
    <sequence>
      <element name="facility_level_Stadium" type="xsd:double" maxOccurs="unbounded" />
    </sequence>
  </complexType>
  <complexType name="ListOfFacilitiesLevelsStadium">
    <sequence>
      <element name="facilities_levels_per_HT_Stadium" type="tns:FacilitiesLevelsStadium" maxOccurs="unbounded" />
    </sequence>
  </complexType>
  <complexType name="ListOfFacilitiesSARS">
    <sequence>
      <element name="facilitySARS" type="xsd:integer" maxOccurs="unbounded" />
    </sequence>
  </complexType>
```

Figure 1: Input parameters

In Figure 2 are listed instead the output parameters of the SCGT Engine Web Service. The keyword Results (in facilitiesModelResultsStadium and facilitiesModelResultsSARS) is used to define the output of the server function as a list of facility levels, matching the output described in section 4.1.

```
<xsd:element name="facilitiesModelParametersStadium" type="tns:FacilitiesModelParametersStadium"/>
<xsd:element name="facilitiesModelResultsStadium" type="tns:ListOfFacilitiesLevelsStadium"/>
<xsd:element name="facilitiesModelParametersSARS" type="tns:FacilitiesModelParametersSARS"/>
<xsd:element name="facilitiesModelResultsSARS" type="tns:ListOfFacilitiesLevelsSARS"/>
```

Figure 2: Output parameters



The request and response messages exploit Parameters (inputs) and Results (output) respectively. The messages defined in the WSDL file are depicted in Figure 3.

```
<message name="FacilitiesModelRequestStadium">
  <part name="facilitiesModelparStadium" element="tns:facilitiesModelParametersStadium"/>
</message>
<message name="FacilitiesModelResponseStadium">
  <part name="facilitiesModelresStadium" element="tns:facilitiesModelResultsStadium"/>
</message>
<message name="FacilitiesModelRequestSARS">
  <part name="facilitiesModelparSARS" element="tns:facilitiesModelParametersSARS"/>
</message>
<message name="FacilitiesModelResponseSARS">
  <part name="facilitiesModelresSARS" element="tns:facilitiesModelResultsSARS"/>
</message>
```

Figure 3: WSDL Messages

The operations “FacilitiesModelSARS” and “FacilitiesModelStadium”, as shown in Figure 4, match the names of the corresponding PHP function files on the Web Server and are the operations that can be invoked by an external SCGT client.

```
<operation name="FacilitiesModelSARS">
  <input message="tns:FacilitiesModelRequestSARS"/>
  <output message="tns:FacilitiesModelResponseSARS"/>
</operation>
<operation name="FacilitiesModelStadium">
  <input message="tns:FacilitiesModelRequestStadium"/>
  <output message="tns:FacilitiesModelResponseStadium"/>
</operation>
```

Figure 4: WSDL Operations

Note that the WSDL declarations are different for the two scenarios (they have different suffixes “SARS” and “Stadium”).



```
private static ListOfFacilitiesLevelsSARS facilitiesModelSARS
(FacilitiesModelParametersSARS logevoparSARS)
{
    PULSEWebServices service = new PULSEWebServices();
    PULSEModelsPortType port = service.getPULSEModelsPort();
    return port.facilitiesModelSARS(logevoparSARS);
}

private static ListOfFacilitiesLevelsStadium facilitiesModelStadium
(FacilitiesModelParametersStadium facilitiesModelparStadium)
{
    PULSEWebServices service = new PULSEWebServices();
    PULSEModelsPortType port = service.getPULSEModelsPort();
    return port.facilitiesModelStadium(facilitiesModelparStadium);
}
```

Figure 6: Java Client example

5.2 Component relation to overall PULSE architecture

As described in 5.1, the SCGT can be easily invoked by an external client. Within the PULSE context, the client in question is the DSVT that invokes the Surge Capacity Generation tool whenever an update regarding the possible variation of the hospital resources (e.g. medical staff) is requested. This update is specifically requested by the DSVT during the Simulation phase, as described in D4.1 [9]. Figure 7 shows a component diagram of the PULSE Platform Architecture where the relation between SCGT and DSVT is highlighted.

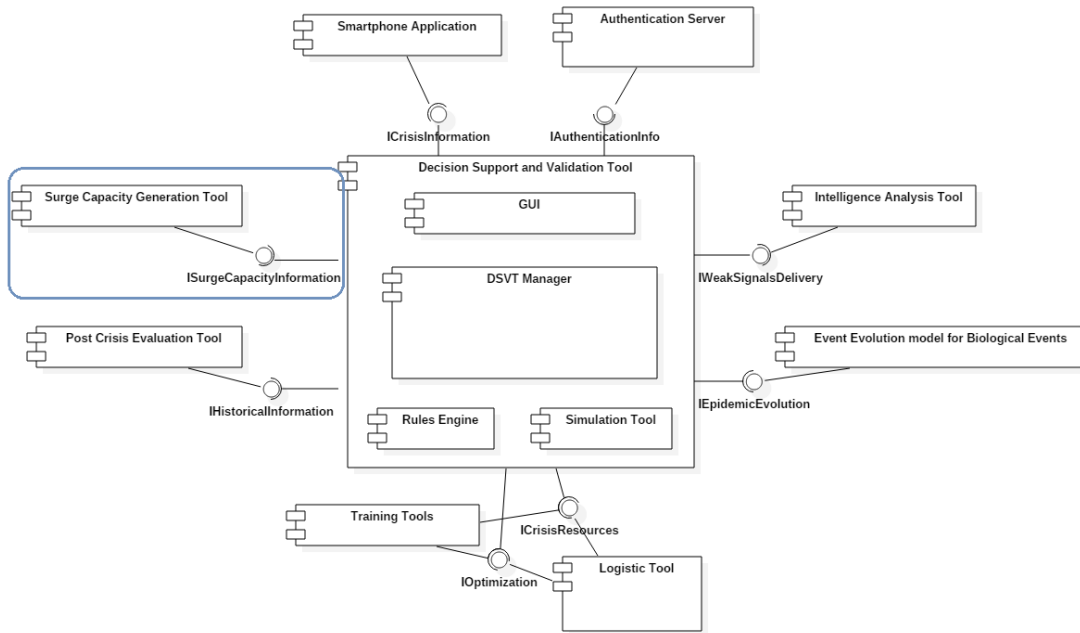


Figure 7: PULSE Platform Architecture - Component diagram



6 Component Technologies

6.1 List of core technologies

The core technologies adopted for the implementation of the SCGT are:

- *SCGT Engine:*
 - **PHP:** it is a popular general-purpose scripting language that is especially suited to web development. It has been used to define the Web Service exposed by the SCGT (see 5.1.1) and to implement, as software solution, the mathematical model defined in WP3.
 - **Apache HTTP Server:** it is an open-source HTTP server and it is the most popular web server on the Internet since 1996. It hosts the PHP web application that exposes the Web Service described in 5.1.1.
- *Example client:*
 - **Java:** The test client has been developed by using the Java programming language

6.2 3rd Party libraries and licenses

Below is a list of third party libraries/frameworks used and the licenses under which they are distributed.

Table 4: 3rd party libraries and licenses

Product	Version	Vendor	License
PHP	5	PHP Group	PHP License 3.01
Apache HTTP Server	2.4	Apache Software Foundation	Apache 2.0
Java SE Development Kit	8	Oracle	Oracle Binary Code License Agreement
Beans Binding	1.2.1	Java.net	GNU LESSER GENERAL PUBLIC LICENSE
Swing Layout Extensions	1.0.4	Java.net	GNU LESSER GENERAL PUBLIC LICENSE



7 Software Test/Validation

The Surge capacity tool (T.4.4) which is object of the present deliverable exploits as basic modeling layer the Health Care Facilities Model (T3.3), which is exposed as a webserver. The webservice is called with a subset of inputs selectable by the user by means of the tool graphical GUI, while some other inputs are fixed.

Parameter tuning and implementation refinements are object of future deliverables in the final year of the project (M19-M30), wherein the processes of integration and validation/trials in WP6-WP7 will provide further occasion for tuning the model/tool parameters as a consequence of the lessons learnt, of the procedures from the WP5 deliverables and of the application of the PULSE platform in realistic conditions for the considered scenarios. The integrated PULSE platform will also allow to set a larger set of parameters and to retrieve some results in a graphical form. The current parameter values have been determined from the study of the related literature and exploiting the expertise of UCSC medical staff. The validation of these parameters (by means of repeated runs of the tool) will be object of the last year of the project.



References

- [1] Segen's Medical Dictionary. © 2012 Farlex, Inc.
- [2] D Hick JL1, Barbera JA, Kelen G: Refining surge capacity: conventional, contingency, and crisis capacity. *Disaster Med Public Health Prep.* 2009 Jun;3(2 Suppl):S59-67.
- [3] *Medical Response to Major Incidents and Disasters. A Practical Guide for All Medical Staff.* Editors: Lennquist, Sten (Ed.), Springer Verlag, Berlin 2012.
- [4] Boer de, J, et al.: The medical severity index of disasters. 1989, *The Journal of Emergency Medicine*, Vol. 7, pp.269-273.
- [5] Hirshberg A, Scott BG, Granchi T, Wall MJ, Mattox KL, Stein M. How Does Casualty Load Affect Trauma Care in Urban Bombing Incidents? A Quantitative Analysis. *Journal of Trauma-Injury Infection and Critical Care.* 2005;58:686–93.
- [6] Bayram JD, Zuabi S, Subbarao I. Disaster Metrics: Quantitative Benchmarking of Hospital Surge Capacity in Trauma-Related Multiple Casualty Events. *Disaster Medicine and Public Health Preparedness.* 2011;5:117–24.
- [7] Kelen GD1, McCarthy ML, Kraus CK, Ding R, Hsu EB, Li G, Shahan JB, Scheulen JJ, Green GB.: Creation of surge capacity by early discharge of hospitalized patients at low risk for untoward events. *Disaster Med Public Health Prep.* 2009 Jun;3(2 Suppl):S10-6.
- [8] PULSE Project Deliverable – D2.2 Use case specification
- [9] PULSE Project Deliverable – D4.1 Decision Support Validation tool
- [10] PULSE Project Deliverable – D3.1 Context Models



Appendix A

A.1 WSDL

```
<?xml version="1.0" encoding="UTF-8" ?>
<definitions name="PULSEmodels"
  targetNamespace="http://biomat1.iasi.cnr.it/webservices/pulse/M
18/"
  xmlns:tns="http://biomat1.iasi.cnr.it/webservices/pulse/M18/"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:wSDL="http://schemas.xmlsoap.org/wsdl/"
  xmlns="http://schemas.xmlsoap.org/wsdl/">

  <types>
    <schema xmlns="http://www.w3.org/2001/XMLSchema"
      targetNamespace="http://biomat1.iasi.cnr.it/webservices/pulse/M
18/">
      <xsd:import
        namespace="http://schemas.xmlsoap.org/soap/encoding/" />
      <xsd:import
        namespace="http://schemas.xmlsoap.org/wsdl/" />
      <complexType name="Therapy">
        <sequence>
          <element name='code' type='xsd:integer'
            maxOccurs='1' />
          <element name='ther_A_delta'
            type='xsd:double' maxOccurs='1' />
          <element name='ther_A_alpha'
            type='xsd:double' maxOccurs='1' />
          <element name='ther_B1_delta'
            type='xsd:double' maxOccurs='1' />
          <element name='ther_B1_alpha'
            type='xsd:double' maxOccurs='1' />
          <element name='ther_B2_delta'
            type='xsd:double' maxOccurs='1' />
          <element name='ther_B2_alpha'
            type='xsd:double' maxOccurs='1' />
          <element name='ther_B3_delta'
            type='xsd:double' maxOccurs='1' />
          <element name='ther_B3_alpha'
            type='xsd:double' maxOccurs='1' />
          <element name='ther_C1_delta'
            type='xsd:double' maxOccurs='1' />
          <element name='ther_C1_alpha'
            type='xsd:double' maxOccurs='1' />
          <element name='ther_C2_delta'
            type='xsd:double' maxOccurs='1' />
          <element name='ther_C2_alpha'
            type='xsd:double' maxOccurs='1' />
          <element name='ther_D1_delta'
            type='xsd:double' maxOccurs='1' />
          <element name='ther_D1_alpha'
            type='xsd:double' maxOccurs='1' />
          <element name='ther_D2_delta' type='xsd:double' maxOccurs='1' />
        </sequence>
      </complexType>
    </schema>
  </types>
</definitions>
```



```

                                <element name='ther_D2_alpha'
type='xsd:double' maxOccurs='1' />
                                <element name='ther_D3_delta'
type='xsd:double' maxOccurs='1' />
                                <element name='ther_D3_alpha'
type='xsd:double' maxOccurs='1' />
                                <element name='ther_E_delta'
type='xsd:double' maxOccurs='1' />
                                <element name='ther_E_alpha'
type='xsd:double' maxOccurs='1' />
                                </sequence>
                                </complexType>
                                <complexType name="ListOfTherapies">
                                <sequence>
                                <element name='therapy'
type='tns:Therapy' maxOccurs='unbounded' />
                                </sequence>
                                </complexType>
                                <complexType name="PatientStatus">
                                <sequence>
                                <element name='x_A' type='xsd:double'
maxOccurs='1' />
                                <element name='x_B1' type='xsd:double'
maxOccurs='1' />
                                <element name='x_B2' type='xsd:double'
maxOccurs='1' />
                                <element name='x_B3' type='xsd:double'
maxOccurs='1' />
                                <element name='x_C1' type='xsd:double'
maxOccurs='1' />
                                <element name='x_C2' type='xsd:double'
maxOccurs='1' />
                                <element name='x_D1' type='xsd:double'
maxOccurs='1' />
                                <element name='x_D2' type='xsd:double'
maxOccurs='1' />
                                <element name='x_D3' type='xsd:double'
maxOccurs='1' />
                                <element name='x_E' type='xsd:double'
maxOccurs='1' />
                                <element name='v_A' type='xsd:double'
maxOccurs='1' />
                                <element name='v_B1' type='xsd:double'
maxOccurs='1' />
                                <element name='v_B2' type='xsd:double'
maxOccurs='1' />
                                <element name='v_B3' type='xsd:double'
maxOccurs='1' />
                                <element name='v_C1' type='xsd:double'
maxOccurs='1' />
                                <element name='v_C2' type='xsd:double'
maxOccurs='1' />
                                <element name='v_D1' type='xsd:double'
maxOccurs='1' />
                                <element name='v_D2' type='xsd:double'
maxOccurs='1' />
                                <element name='v_D3' type='xsd:double'
maxOccurs='1' />

```



```

                                <element name='v_E' type='xsd:double'
maxOccurs='1' />
                                </sequence>
                                </complexType>
                                <complexType name="PopulationStatus">
                                <sequence>
                                <element name='patient_status'
type='tns:PatientStatus'
                                maxOccurs='unbounded' />
                                </sequence>
                                </complexType>
                                <complexType name="ListOfIntegers">
                                <sequence>
                                <element name='int_element'
type='xsd:integer' maxOccurs='unbounded' />
                                </sequence>
                                </complexType>
                                <complexType name="Lesions">
                                <sequence>
                                <element name='lesion'
type='xsd:double' maxOccurs='unbounded' />
                                </sequence>
                                </complexType>
                                <complexType name="ListOfLesions">
                                <sequence>
                                <element name='pat_lesions'
type='tns:Lesions' maxOccurs='unbounded' />
                                </sequence>
                                </complexType>
                                <complexType name="Symptoms">
                                <sequence>
                                <element name='symptom'
type='xsd:integer' maxOccurs='unbounded' />
                                </sequence>
                                </complexType>
                                <complexType name="ListOfSymptoms">
                                <sequence>
                                <element name='pat_symptoms'
type='tns:Symptoms' maxOccurs='unbounded' />
                                </sequence>
                                </complexType>
                                <complexType name="ListOfFacilitiesStadium">
                                <sequence>
                                <element name='facilityStadium'
type='xsd:integer'
                                maxOccurs='unbounded' />
                                </sequence>
                                </complexType>
                                <complexType name="FacilitiesLevelsStadium">
                                <sequence>
                                <element name='facility_level_Stadium'
type='xsd:double'
                                maxOccurs='unbounded' />
                                </sequence>
                                </complexType>
                                <complexType name="ListOfFacilitiesLevelsStadium">
                                <sequence>
                                <element

```



```
name='facilities_levels_per_HT_Stadium'  
type='tns:FacilitiesLevelsStadium'  
maxOccurs='unbounded' />  
  </sequence>  
</complexType>  
<complexType name="ListOfFacilitiesSARS">  
  <sequence>  
    <element name='facilitySARS'  
type='xsd:integer' maxOccurs='unbounded' />  
  </sequence>  
</complexType>  
<complexType name="FacilitiesLevelsSARS">  
  <sequence>  
    <element name='facility_levelSARS'  
type='xsd:double'  
maxOccurs='unbounded' />  
  </sequence>  
</complexType>  
<complexType name="ListOfFacilitiesLevelsSARS">  
  <sequence>  
    <element  
name='facilities_levels_per_HT_SARS' type='tns:FacilitiesLevelsSARS'  
maxOccurs='unbounded' />  
  </sequence>  
</complexType>  
<complexType name="ListOfETDs">  
  <sequence>  
    <element name='etd' type='xsd:double'  
maxOccurs='unbounded' />  
  </sequence>  
</complexType>  
<complexType name="ListOfGCSs">  
  <sequence>  
    <element name='gcs' type='xsd:integer'  
maxOccurs='unbounded' />  
  </sequence>  
</complexType>  
<complexType name="ListOfISSs">  
  <sequence>  
    <element name='iss' type='xsd:integer'  
maxOccurs='unbounded' />  
  </sequence>  
</complexType>  
<complexType name="ListOfColorCodes">  
  <sequence>  
    <element name='code' type='xsd:integer'  
maxOccurs='unbounded' />  
  </sequence>  
</complexType>  
<complexType name="PatGenerateParameters">  
  <sequence>  
    <element name='num_people_in_area'  
type='xsd:integer'  
maxOccurs='1' />  
    <element name='event_code'  
type='xsd:integer' maxOccurs='1' />  
    <element name='event_severity'  
type='xsd:double' maxOccurs='1' />
```



```

                                <element name='event_longitude'
type='xsd:double'
                                maxOccurs='1' />
                                <element name='event_latitude'
type='xsd:double' maxOccurs='1' />
                                <element name='event_radius'
type='xsd:double' maxOccurs='1' />
                                </sequence>
                                </complexType>
                                <complexType name="ScenGenParameters">
                                <sequence>
                                <element name='num_bystanders'
type='xsd:integer'
                                maxOccurs='1' />
                                <element name='event_type'
type='xsd:integer' maxOccurs='1' />
                                <element name='event_size'
type='xsd:double' maxOccurs='1' />
                                <element name='event_longitude'
type='xsd:double'
                                maxOccurs='1' />
                                <element name='event_latitude'
type='xsd:double' maxOccurs='1' />
                                <element name='event_dimension'
type='xsd:double'
                                maxOccurs='1' />
                                <element name='event_min_latency'
type='xsd:double'
                                maxOccurs='1' />
                                </sequence>
                                </complexType>
                                <complexType name="FacilitiesModelParametersSARS">
                                <sequence>
                                <element name='num_patients'
type='xsd:integer' maxOccurs='1' />
                                <element name='delta_T'
type='xsd:double' maxOccurs='1' />
                                </sequence>
                                </complexType>
                                <complexType
name="FacilitiesModelParametersStadium">
                                <sequence>
                                <element name='num_patients'
type='xsd:integer' maxOccurs='1' />
                                <element name='delta_T'
type='xsd:double' maxOccurs='1' />
                                </sequence>
                                </complexType>
                                <complexType name="GeoLocalizedPatient">
                                <sequence>
                                <element name='health_status'
type='tns:PatientStatus'
                                maxOccurs='1' />
                                <element name='longitude'
type='xsd:double' maxOccurs='1' />
                                <element name='latitude'
type='xsd:double' maxOccurs='1' />
                                </sequence>

```




```
        </complexType>
        <complexType
name="GeoLocalizedPatientWithLatencyAndLesions">
            <sequence>
                <element name='health_status'
type='tns:PatientStatus'
                    maxOccurs='1' />
                <element name='longitude'
type='xsd:double' maxOccurs='1' />
                <element name='latitude'
type='xsd:double' maxOccurs='1' />
                <element name='latency'
type='xsd:double' maxOccurs='1' />
                <element name='lesions'
type='tns:Lesions' maxOccurs='1' />
            </sequence>
        </complexType>
        <complexType
name="GeoLocalizedPatientWithTriageAndETD">
            <sequence>
                <element name='health_status'
type='tns:PatientStatus'
                    maxOccurs='1' />
                <element name='longitude'
type='xsd:double' maxOccurs='1' />
                <element name='latitude'
type='xsd:double' maxOccurs='1' />
                <element name='triagecode'
type='xsd:integer' maxOccurs='1' />
                <element name='etd' type='xsd:double'
maxOccurs='1' />
            </sequence>
        </complexType>
        <complexType
name="GeoLocalizedPatientWithTriageAndETDAndSigns">
            <sequence>
                <element name='health_status'
type='tns:PatientStatus'
                    maxOccurs='1' />
                <element name='longitude'
type='xsd:double' maxOccurs='1' />
                <element name='latitude'
type='xsd:double' maxOccurs='1' />
                <element name='triagecode'
type='xsd:integer' maxOccurs='1' />
                <element name='etd' type='xsd:double'
maxOccurs='1' />
                <element name='signs' type='xsd:string'
maxOccurs='1' />
            </sequence>
        </complexType>
        <complexType name="ListOfGeoLocalizedPatients">
            <sequence>
                <element name='geo_localized_patient'
type='tns:GeoLocalizedPatient'
                    maxOccurs='unbounded' />
            </sequence>
        </complexType>
```



```
        <complexType
name="ListOfGeoLocalizedPatientsWithLatencyAndLesions">
        <sequence>
            <element
name='geo_localized_patient_with_latency_and_lesions'

            type='tns:GeoLocalizedPatientWithLatencyAndLesions'
maxOccurs='unbounded' />
        </sequence>
    </complexType>
    <complexType
name="ListOfGeoLocalizedPatientsWithTriageAndETD">
        <sequence>
            <element
name='geo_localized_patient_with_triage_and_etd'

            type='tns:GeoLocalizedPatientWithTriageAndETD'
maxOccurs='unbounded' />
        </sequence>
    </complexType>
    <complexType
name="ListOfGeoLocalizedPatientsWithTriageAndETDAndSigns">
        <sequence>
            <element
name='geo_localized_patient_with_triage_and_etd'

            type='tns:GeoLocalizedPatientWithTriageAndETDAndSigns'
maxOccurs='unbounded' />
        </sequence>
    </complexType>
    <complexType name="PatientEvolveParameters">
        <sequence>
            <element name='pat_status'
type='tns:PatientStatus'
maxOccurs='1' />
            <element name='therapy'
type='xsd:integer' maxOccurs='1' />
            <element name='t_fin' type='xsd:double'
maxOccurs='1' />
        </sequence>
    </complexType>
    <complexType name="SickEvoParameters">
        <sequence>
            <element name='pop_status'
type='tns:PopulationStatus'
maxOccurs='1' />
            <element name='assets'
type='tns:ListOfFacilitiesStadium'
maxOccurs='1' />
            <element name='delta_t'
type='xsd:double' maxOccurs='1' />
        </sequence>
    </complexType>
    <complexType name="SickEvoResults">
        <sequence>
            <element name='pop_status'
type='tns:PopulationStatus'
maxOccurs='1' />
```



```
        </sequence>
    </complexType>
    <complexType name="DamagerParameters">
        <sequence>
            <element name='event_type'
type='xsd:integer' maxOccurs='1' />
            <element name='patients_lesions'
type='tns:ListOfLesions'
                maxOccurs='1' />
        </sequence>
    </complexType>
    <complexType name="DamagerResults">
        <sequence>
            <element name='pop_status'
type='tns:PopulationStatus'
                maxOccurs='1' />
        </sequence>
    </complexType>
    <complexType name="StatScoringParameters">
        <sequence>
            <element name='pop_status'
type='tns:PopulationStatus'
                maxOccurs='1' />
        </sequence>
    </complexType>
    <complexType name="StatScoringResults">
        <sequence>
            <element name='expected_time_of_death'
type='tns:ListOfETDs'
                maxOccurs='1' />
            <element name='glasgow_coma_scale'
type='tns:ListOfGCSs'
                maxOccurs='1' />
            <element name='injury_severity_score'
type='tns:ListOfISSs'
                maxOccurs='1' />
            <element name='color_code'
type='tns:ListOfColorCodes'
                maxOccurs='1' />
        </sequence>
    </complexType>
    <complexType name="SymptScoringParameters">
        <sequence>
            <element name='pop_symptoms'
type='tns:ListOfSymptoms'
                maxOccurs='1' />
        </sequence>
    </complexType>
    <complexType name="SymptScoringResults">
        <sequence>
            <element name='expected_time_of_death'
type='tns:ListOfETDs'
                maxOccurs='1' />
            <element name='glasgow_coma_scale'
type='tns:ListOfGCSs'
                maxOccurs='1' />
            <element name='injury_severity_score'
type='tns:ListOfISSs'
```



```

                maxOccurs='1' />
            <element name='color_code'
type='tns:ListOfColorCodes'
                maxOccurs='1' />
        </sequence>
    </complexType>
    <complexType name="PopulationEvolveParameters">
        <sequence>
            <element name='patient_parameters'
type='tns:PatientEvolveParameters'
                maxOccurs='unbounded' />
        </sequence>
    </complexType>
    <complexType name="Void">
        <sequence>
        </sequence>
    </complexType>
    <complexType name="ExposedsAtStart">
        <sequence>
            <element name='row' type='xsd:integer'
maxOccurs='1' />
            <element name='col' type='xsd:integer'
maxOccurs='1' />
            <element name='exposeds'
type='xsd:integer' maxOccurs='1' />
        </sequence>
    </complexType>
    <complexType name="ListOfExposedsAtStart">
        <sequence>
            <element name='exposeds_data'
type='tns:ExposedsAtStart'
                maxOccurs='unbounded' />
        </sequence>
    </complexType>
    <complexType name="InfectivesAtStart">
        <sequence>
            <element name='row' type='xsd:integer'
maxOccurs='1' />
            <element name='col' type='xsd:integer'
maxOccurs='1' />
            <element name='infectives'
type='xsd:integer' maxOccurs='1' />
        </sequence>
    </complexType>
    <complexType name="ListOfInfectivesAtStart">
        <sequence>
            <element name='infectives_data'
type='tns:InfectivesAtStart'
                maxOccurs='unbounded' />
        </sequence>
    </complexType>
    <complexType name="EnsirParameters">
        <sequence>
            <element name='exposeds_at_start'
type='tns:ListOfExposedsAtStart'
                maxOccurs='1' />
            <element name='infectives_at_start'

```



```
type='tns:ListOfInfectivesAtStart'
    maxOccurs='1' />
    <element name='assets'
type='tns:ListOfFacilitiesStadium'
    maxOccurs='1' />
    <element name='t_fin' type='xsd:double'
maxOccurs='1' />
    <element name='delta_t'
type='xsd:double' maxOccurs='1' />
    <element name='event_type'
type='xsd:integer' maxOccurs='1' />
    </sequence>
</complexType>
<complexType name="EnsirResults">
    <sequence>
        <element name='susceptibles_population'
type='xsd:double'
            maxOccurs='unbounded' />
        <element name='exposeds_population'
type='xsd:double'
            maxOccurs='unbounded' />
        <element name='infectives_population'
type='xsd:double'
            maxOccurs='unbounded' />
        <element name='recovered_population'
type='xsd:double'
            maxOccurs='unbounded' />
        <element name='total_infecteds'
type='xsd:double'
            maxOccurs='unbounded' />
    </sequence>
</complexType>

    <xsd:element name="triageParameters"
type="tns:PatientStatus" />
    <xsd:element name="triageResults"
type="xsd:integer" />
    <xsd:element name="popTriageParameters"
type="tns:PopulationStatus" />
    <xsd:element name="popTriageResults"
type="tns:ListOfIntegers" />
    <xsd:element name="patEvolveParameters"
type="tns:PatientEvolveParameters" />
    <xsd:element name="patEvolveResults"
type="tns:PatientStatus" />
    <xsd:element name="getTherapiesParameters"
type="tns:Void" />
    <xsd:element name="getTherapiesResults"
type="tns:ListOfTherapies" />
    <xsd:element name="popEvolveParameters"
type="tns:PopulationEvolveParameters" />
    <xsd:element name="popEvolveResults"
type="tns:PopulationStatus" />
    <xsd:element name="sickEvoParameters"
type="tns:SickEvoParameters" />
    <xsd:element name="sickEvoResults"
type="tns:SickEvoResults" />
    <xsd:element
```



```
name="facilitiesModelParametersStadium"
    type="tns:FacilitiesModelParametersStadium"
/>
    <xsd:element name="facilitiesModelResultsStadium"
type="tns:ListOfFacilitiesLevelsStadium" />
    <xsd:element name="facilitiesModelParametersSARS"
type="tns:FacilitiesModelParametersSARS" />
    <xsd:element name="facilitiesModelResultsSARS"
type="tns:ListOfFacilitiesLevelsSARS" />
    <xsd:element name="patGenerateParameters"
type="tns:PatGenerateParameters" />
    <xsd:element name="patGenerateResults"
type="tns:ListOfGeoLocalizedPatients" />
    <xsd:element name="ScenGenParameters"
type="tns:ScenGenParameters" />
    <xsd:element name="ScenGenResults"

    type="tns:ListOfGeoLocalizedPatientsWithLatencyAndLesions" />
    <xsd:element name="statScoringParameters"
type="tns:StatScoringParameters" />
    <xsd:element name="statScoringResults"
type="tns:StatScoringResults" />
    <xsd:element name="symptScoringParameters"
type="tns:SymptScoringParameters" />
    <xsd:element name="symptScoringResults"
type="tns:SymptScoringResults" />
    <xsd:element name="damagerParameters"
type="tns:DamagerParameters" />
    <xsd:element name="damagerResults"
type="tns:DamagerResults" />
    <xsd:element
name="patGenerateWithTriageAndETDParameters"
    type="tns:PatGenerateParameters" />
    <xsd:element
name="patGenerateWithTriageAndETDAndSignsParameters"
    type="tns:PatGenerateParameters" />
    <xsd:element
name="patGenerateWithTriageAndETDResults"

    type="tns:ListOfGeoLocalizedPatientsWithTriageAndETD" />
    <xsd:element
name="patGenerateWithTriageAndETDAndSignsResults"

    type="tns:ListOfGeoLocalizedPatientsWithTriageAndETDAndSigns"
/>
    <xsd:element name="popETDParameters"
type="tns:PopulationStatus" />
    <xsd:element name="popETDResults"
type="tns:ListOfETDs" />
    <xsd:element name="EnsirParameters"
type="tns:EnsirParameters" />
    <xsd:element name="EnsirResults"
type="tns:EnsirResults" />

</schema>

</types>
```



```
<message name="triageRequest">
  <part name="triagepar" element="tns:triageParameters" />
</message>

<message name="triageResponse">
  <part name="triageres" element="tns:triageResults" />
</message>

<message name="popTriageRequest">
  <part name="poptriagepar"
element="tns:popTriageParameters" />
</message>

<message name="popTriageResponse">
  <part name="poptriageres" element="tns:popTriageResults"
/>
</message>

<message name="patEvolveRequest">
  <part name="patevolvepar"
element="tns:patEvolveParameters" />
</message>

<message name="patEvolveResponse">
  <part name="patevolveres" element="tns:patEvolveResults"
/>
</message>

<message name="getTherapiesRequest">
  <part name="gettherpar"
element="tns:getTherapiesParameters" />
</message>

<message name="getTherapiesResponse">
  <part name="gettherres" element="tns:getTherapiesResults"
/>
</message>

<message name="popEvolveRequest">
  <part name="popevolvepar"
element="tns:popEvolveParameters" />
</message>

<message name="popEvolveResponse">
  <part name="popevolveres" element="tns:popEvolveResults"
/>
</message>

<message name="SickEvoRequest">
  <part name="sickevopar" element="tns:sickEvoParameters"
/>
</message>

<message name="SickEvoResponse">
  <part name="sickevores" element="tns:sickEvoResults" />
</message>

<message name="FacilitiesModelRequestStadium">
```



```
        <part name="facilitiesModelparStadium"
element="tns:facilitiesModelParametersStadium" />
    </message>

    <message name="FacilitiesModelResponseStadium">
        <part name="facilitiesModelresStadium"
element="tns:facilitiesModelResultsStadium" />
    </message>

    <message name="FacilitiesModelRequestSARS">
        <part name="facilitiesModelparSARS"
element="tns:facilitiesModelParametersSARS" />
    </message>

    <message name="FacilitiesModelResponseSARS">
        <part name="facilitiesModelresSARS"
element="tns:facilitiesModelResultsSARS" />
    </message>

    <message name="patGenerateRequest">
        <part name="patgeneratepar"
element="tns:patGenerateParameters" />
    </message>

    <message name="patGenerateResponse">
        <part name="patgenerateres"
element="tns:patGenerateResults" />
    </message>

    <message name="ScenGenRequest">
        <part name="scengenpar" element="tns:ScenGenParameters"
/>
    </message>

    <message name="ScenGenResponse">
        <part name="scengenres" element="tns:ScenGenResults" />
    </message>

    <message name="statScoringRequest">
        <part name="statscoringpar"
element="tns:statScoringParameters" />
    </message>

    <message name="statScoringResponse">
        <part name="statscoringres"
element="tns:statScoringResults" />
    </message>

    <message name="symptScoringRequest">
        <part name="symptscoringpar"
element="tns:symptScoringParameters" />
    </message>

    <message name="symptScoringResponse">
        <part name="symptscoringres"
element="tns:symptScoringResults" />
    </message>
```




```
<message name="damagerRequest">
  <part name="damagerpar" element="tns:damagerParameters"
/>
</message>

<message name="damagerResponse">
  <part name="damagerres" element="tns:damagerResults" />
</message>

<message name="patGenerateWithTriageAndETDRequest">
  <part name="patgeneratetriageetdpar"
element="tns:patGenerateWithTriageAndETDParameters" />
</message>

<message name="patGenerateWithTriageAndETDAndSignsRequest">
  <part name="patgeneratetriageetdpar"
element="tns:patGenerateWithTriageAndETDAndSignsParameters" />
</message>

<message name="patGenerateWithTriageAndETDResponse">
  <part name="patgeneratetriageetdres"
element="tns:patGenerateWithTriageAndETDResults" />
</message>

<message name="patGenerateWithTriageAndETDAndSignsResponse">
  <part name="patgeneratetriageetdres"
element="tns:patGenerateWithTriageAndETDAndSignsResults" />
</message>

<message name="popETDRequest">
  <part name="popETDpar" element="tns:popETDParameters" />
</message>

<message name="popETDResponse">
  <part name="popETDres" element="tns:popETDResults" />
</message>

<message name="ensirRequest">
  <part name="ensirpar" element="tns:EnsirParameters" />
</message>

<message name="ensirResponse">
  <part name="ensirres" element="tns:EnsirResults" />
</message>

<portType name="PULSEModelsPortType">
  <operation name="patTriage">
    <input message="tns:triageRequest" />
    <output message="tns:triageResponse" />
  </operation>
  <operation name="popTriage">
    <input message="tns:popTriageRequest" />
    <output message="tns:popTriageResponse" />
  </operation>
  <operation name="patEvolve">
    <input message="tns:patEvolveRequest" />
    <output message="tns:patEvolveResponse" />
  </operation>
</portType>
```



```
</operation>
<operation name="getTherapies">
  <input message="tns:getTherapiesRequest" />
  <output message="tns:getTherapiesResponse" />
</operation>
<operation name="popEvolve">
  <input message="tns:popEvolveRequest" />
  <output message="tns:popEvolveResponse" />
</operation>
<operation name="SickEvo">
  <input message="tns:SickEvoRequest" />
  <output message="tns:SickEvoResponse" />
</operation>
<operation name="FacilitiesModelSARS">
  <input message="tns:FacilitiesModelRequestSARS" />
  <output message="tns:FacilitiesModelResponseSARS"
/>
  </operation>
<operation name="FacilitiesModelStadium">
  <input message="tns:FacilitiesModelRequestStadium"
/>
  <output
message="tns:FacilitiesModelResponseStadium" />
  </operation>
<operation name="patGenerate">
  <input message="tns:patGenerateRequest" />
  <output message="tns:patGenerateResponse" />
</operation>
<operation name="ScenGen">
  <input message="tns:ScenGenRequest" />
  <output message="tns:ScenGenResponse" />
</operation>
<operation name="Damager">
  <input message="tns:damagerRequest" />
  <output message="tns:damagerResponse" />
</operation>
<operation name="StatScoring">
  <input message="tns:statScoringRequest" />
  <output message="tns:statScoringResponse" />
</operation>
<operation name="SymptScoring">
  <input message="tns:symptScoringRequest" />
  <output message="tns:symptScoringResponse" />
</operation>
<operation name="patGenerateWithTriageAndETD">
  <input
message="tns:patGenerateWithTriageAndETDRequest" />
  <output
message="tns:patGenerateWithTriageAndETDResponse" />
  </operation>
  <operation name="patGenerateWithTriageAndETDAndSigns">
    <input
message="tns:patGenerateWithTriageAndETDAndSignsRequest" />
    <output
message="tns:patGenerateWithTriageAndETDAndSignsResponse" />
  </operation>
  <operation name="popETD">
    <input message="tns:popETDRequest" />
```



```
        <output message="tns:popETDResponse" />
    </operation>
    <operation name="EnSIR">
        <input message="tns:ensirRequest" />
        <output message="tns:ensirResponse" />
    </operation>
</portType>

<binding name="PULSEModelsBinding"
type="tns:PULSEModelsPortType">
    <soap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http" />
    <operation name="patTriage">
        <soap:operation soapAction="patTriage" />
        <input>
            <soap:body use="literal" />
        </input>
        <output>
            <soap:body use="literal" />
        </output>
    </operation>
    <operation name="popTriage">
        <soap:operation soapAction="popTriage" />
        <input>
            <soap:body use="literal" />
        </input>
        <output>
            <soap:body use="literal" />
        </output>
    </operation>
    <operation name="patEvolve">
        <soap:operation soapAction="patEvolve" />
        <input>
            <soap:body use="literal" />
        </input>
        <output>
            <soap:body use="literal" />
        </output>
    </operation>
    <operation name="getTherapies">
        <soap:operation soapAction="getTherapies" />
        <input>
            <soap:body use="literal" />
        </input>
        <output>
            <soap:body use="literal" />
        </output>
    </operation>
    <operation name="popEvolve">
        <soap:operation soapAction="popEvolve" />
        <input>
            <soap:body use="literal" />
        </input>
        <output>
            <soap:body use="literal" />
        </output>
    </operation>
</binding>
</service>
</targetNamespace>
</schema>
```



```
<operation name="SickEvo">
  <soap:operation soapAction="SickEvo" />
  <input>
    <soap:body use="literal" />
  </input>
  <output>
    <soap:body use="literal" />
  </output>
</operation>
<operation name="FacilitiesModelSARS">
  <soap:operation soapAction="FacilitiesModelSARS" />
  <input>
    <soap:body use="literal" />
  </input>
  <output>
    <soap:body use="literal" />
  </output>
</operation>
<operation name="FacilitiesModelStadium">
  <soap:operation soapAction="FacilitiesModelStadium" />
  <input>
    <soap:body use="literal" />
  </input>
  <output>
    <soap:body use="literal" />
  </output>
</operation>
<operation name="patGenerate">
  <soap:operation soapAction="patGenerate" />
  <input>
    <soap:body use="literal" />
  </input>
  <output>
    <soap:body use="literal" />
  </output>
</operation>
<operation name="ScenGen">
  <soap:operation soapAction="ScenGen" />
  <input>
    <soap:body use="literal" />
  </input>
  <output>
    <soap:body use="literal" />
  </output>
</operation>
<operation name="Damager">
  <soap:operation soapAction="Damager" />
  <input>
    <soap:body use="literal" />
  </input>
  <output>
    <soap:body use="literal" />
  </output>
</operation>
<operation name="StatScoring">
  <soap:operation soapAction="StatScoring" />
  <input>
    <soap:body use="literal" />
  </input>
  <output>
    <soap:body use="literal" />
  </output>
</operation>
```



```
</input>
<output>
  <soap:body use="literal" />
</output>
</operation>
<operation name="SymptScoring">
  <soap:operation soapAction="SymptScoring" />
  <input>
    <soap:body use="literal" />
  </input>
  <output>
    <soap:body use="literal" />
  </output>
</operation>
<operation name="patGenerateWithTriageAndETD">
  <soap:operation soapAction="patGenerateWithTriageAndETD"
/>
  <input>
    <soap:body use="literal" />
  </input>
  <output>
    <soap:body use="literal" />
  </output>
</operation>
<operation name="patGenerateWithTriageAndETDAndSigns">
  <soap:operation
soapAction="patGenerateWithTriageAndETDAndSigns" />
  <input>
    <soap:body use="literal" />
  </input>
  <output>
    <soap:body use="literal" />
  </output>
</operation>
<operation name="popETD">
  <soap:operation soapAction="popETD" />
  <input>
    <soap:body use="literal" />
  </input>
  <output>
    <soap:body use="literal" />
  </output>
</operation>
<operation name="EnSIR">
  <soap:operation soapAction="EnSIR" />
  <input>
    <soap:body use="literal" />
  </input>
  <output>
    <soap:body use="literal" />
  </output>
</operation>

  <documentation>PULSE Webservices - BioMatLab</documentation>

</binding>

<service name="PULSEWebServices">
```



```
<port name="PULSEModelsPort"
binding="tns:PULSEModelsBinding">
  <soap:address
location="http://biomat1.iasi.cnr.it/webservices/pulse/M18/services.p
hp" />
  </port>
</service>
</definitions>
```