

**Analysis of the Health
Watcher System Using
Viewpoint-based AORE and
the EA-Miner Tool**

Author(s): Americo Sampaio

Table of Contents

1.	Introduction.....	4
2.	Viewpoint-based AORE Model.....	4
2.1	Viewpoints	4
2.2	Early Aspects	7
2.3	Crosscutting Relationships.....	9
2.4	Trade-off points and aspects interaction	10

List of Tables

Table 1 - Viewpoints Crosscut by each Early Aspect.....	10
Table 2 - Trade-off points	11

1. Introduction

This report describes the analysis conducted, using the health watcher system documentation as input, to produce a Viewpoint-based AORE [1, 2] specification with tool support provided by EA-Miner. An overview of what was done is described next.

- The use case document describing the Health Watcher system was used as input to EA-Miner;
- EA-Miner then produces several views showing the identified viewpoints, early aspects (of functional and non-functional nature) and the relationships (i.e., the crosscutting influences between early aspects and viewpoints);
- The requirements engineer used the information produced by EA-Miner and his knowledge and experience to produce the Viewpoint-based AORE specification. As mentioned in [3, 4] EA-Miner does not automate 100% of the tasks and aims at facilitating the work of the requirements engineer by offering automation.

Therefore, in this case study, the requirements engineer analysed which viewpoints and early aspects the tool suggested and that he thought relevant and chose them to be described in the next section of this document.

2. Viewpoint-based AORE Model

This section shows the Viewpoints (Section 2.1), Early Aspects (Section 2.2), crosscutting relationships (Section 2.3). Finally, we discuss about trade-off points related to early aspects interaction (Section 2.4).

2.1 Viewpoints

Viewpoint: Employee

Requirements:

1. There are 3 types of employees: inspectors, attendants, managers
2. The employee must login so that he/she can access the various operations of the system, which are:
 - 2.1. Login: This operation allows an employee to have access to restricted operations on the Health-Watcher system.
 - 2.2. Register tables: This operation allows the registration of system tables. The following operations are possible: insert, update, delete, search and print. The available tables include:
 - 2.2.1. Health unit (unit code, unit description).
 - 2.2.2. Specialty (code and description).
 - 2.2.3. Health unit / Specialty (health unit and specialty).
 - 2.2.4. Employee (login, name and password).
 - 2.2.5. Type of disease (code, name, description, symptom and duration).
 - 2.2.6. Symptom (code and description).
 - 2.2.7. Type of disease / Symptom (type of disease and symptom).

- 2.3. Update complaint: allows the state of a complaint to be updated. Information to be provided: the complaint number, the conclusion, the date the conclusion was made, and the employee who entered the conclusion.
- 2.4. Register new employee: allows new employees to be registered on the system. Information that needs to be provided: Name, Login ID, Password.
- 2.5. Update employee: allows the employee data to be updated. Information that needs to be provided: Name, new password (with extra field for confirmation), current password.
- 2.6. Update health unit: updates the health unit's data
- 2.7. Change logged employee: makes possible the logged employee to be changed

Viewpoint: Citizen

Requirements

1. A citizen is any person who wishes to interact with the system.
2. A citizen can interact with the system to query information and to specify (register) a complaint.
3. A citizen can access the system through the internet or dialling 1520, and make their complaint or ask information about the health services.
4. The citizen might query:
 - 4.1. Which health units take care of a specific specialty.
 - 4.2. What are the specialties of a particular health unit.
 - 4.3. Information about a complaint made by a citizen:
 - 4.3.1. Complaint details.
 - 4.3.2. Situation (OPENED, SUSPENDED, or CLOSED).
 - 4.3.3. Technical analysis.
 - 4.3.4. Analysis date.
 - 4.3.5. Employee that made the analysis.
 - 4.4. Information about diseases:
 - 4.4.1. Description.
 - 4.4.2. Symptoms.
 - 4.4.3. Duration.
 - 4.5. The system shows the query result.
5. The citizen can register a new complaint. This use case makes possible for a citizen to register complaints. Complaints can be: Animal Complaint – DVA •Animals apprehension. •Control of vectors (rodents, scorpions, bats, etc.) •Diseases related to mosquitos (dengue, filarirose). •Animals maltreatment. Food Complaint - DVISA •Cases where it is suspicious the ingestion of infected food. Diverse Complaint - DVISA •Cases related to several reasons, which are not mentioned above (restaurants with hygiene problems, leaking sewerage, suspicious water transporting trucks, etc.). Complaint data: description (mandatory) and observations (optional); Complainer data: name, street, complement, district, city, state/province, zip code, telephone number and e-mail. All these fields are optional.
 - 5.1. Complaint can be an animal complaint. Specific animal complaint data: Type of animal (mandatory), amount of animals (mandatory), date problem was observed (mandatory).
 - 5.2. Complaint can be a food complaint. Specific food complaint data: Victim's name (mandatory). Victim's data: street, complement, district, city (or closest one), state/province, zip code and telephone number. All of these fields are

optional. Amount of people who ate the food, amount of sick people, amount of people who were sent to a hospital and amount of deceased people. All mandatory. Location where the patients were treated, suspicious meal. All optional.

5.3. Complaint can be a special complaint. Specific special complaint data: Age (mandatory), academic qualifications (optional), occupation (optional). Street, complement, district, city, state/province, zip code and telephone number of the closest location to the complaint location. All optional.

Viewpoint: Sanitary Surveillance (SSVS) System

Requirements:

1. The Health Watcher system must also exchange information with the SSVS system (Sanitary Surveillance System).
 - 1.1. Initially, this exchange will involve the querying of sanitary licenses.
 - 1.2. Subsequently, when the SSVS has the Complaint Control module deployed, Sanitary Surveillance complaints will be exchanged between the two systems.

Viewpoint: Complaint

Requirements:

1. The types of Complaints can be:
 - 1.1. Animal Complaint – DVA
 - 1.1.1. Sick animals.
 - 1.1.2. Infestations (rodents, scorpions, bats, etc.)
 - 1.1.3. Diseases related to mosquitoes (dengue, filarirose).
 - 1.1.4. Animal maltreatment.
 - 1.2. Food Complaint - DVISA
 - 1.2.1. Cases where there is a suspicion infected food being eaten.
 - 1.3. Special Complaint - DVISA
 - 1.3.1. Cases related to several reasons, which are not mentioned above (restaurants with hygiene problems, leaking sewerage, suspicious water transporting trucks, etc.).
2. In addition to the above data, each complaint type has its own specific data, including:
 - 2.1. Animal Complaint – DVA
 - 2.1.1. Type of animal (mandatory), amount of animals (mandatory), date problem was observed (mandatory).
 - 2.1.2. Problem location data: street, complement, district, city, state/province, zip code and telephone number. All of these fields are optional.
 - 2.2. Food Complaint - DVISA
 - 2.2.1. Victim's name (mandatory).
 - 2.2.2. Victim's data: street, complement, district, city (or closest one), state/province, zip code and telephone number. All of these fields are optional.
 - 2.2.3. Amount of people who ate the food, amount of sick people, amount of people who were sent to a hospital and amount of deceased people. All mandatory.

- 2.2.4. Location where the patients were treated, suspicious meal. All optional.
- 2.3. Special Complaint - DVISA
 - 2.3.1. Age (mandatory), academic qualifications (optional), occupation (optional).
 - 2.3.2. Street, complement, district, city, state/province, zip code and telephone number of the closest location to the complaint location. All optional.
- 3. Citizens can register a complaint while employees can register and update them.
- 4. In the event of a complaint being made, it will be registered on the system and addressed by a specific department.
 - 4.1. This department will be able to handle the complaint in an appropriate manner and return a response when complaint has been dealt with.
 - 4.2. This response will be registered on the system and available to be queried.

2.2 Early Aspects

Early Aspect: Availability

Requirements:

1. The system should be available 24 hours a day, 7 days a week.
2. As the nature of the system is not considered to be a very critical system, the system might stay off until any fault is fixed.

Early Aspect: Security

Requirements:

1. Security Protocol (encryption): The system should use a security protocol when sending data over the internet.
2. Access Control: To have access to the complaint registration features, access must be allowed by the access control sub-system. Employees have a login and password for using the system (e.g., updating complaints and tables).

Early Aspect: Performance

Requirements:

1. The system must be capable to handle 20 simultaneous users.
2. The response time must not exceed 5 seconds.

Early Aspect: Concurrency

Requirements:

1. The system must be capable to handle 20 simultaneous users.
2. The system shall be able to deal with the concurrent users respecting issues such as database transactions to avoid corruption of data. Moreover, concurrent users should have an adequate response time.

Early Aspect: Persistence

Requirements:

1. The persistence mechanism should store data about the complaints, employees, health units, deceases, specialties and citizens that complain.
2. The system must be flexible in terms of the storage format allowing the use of arrays or different databases (MySQL, Oracle, etc.)

Early Aspect: Distribution

Requirements:

1. The system should be capable of running on separate machines. For example, the system core could be running on one machine and the Servlets on another.

Early Aspect: Error and Exception handling

Requirements:

1. Several functionalities might raise errors while the user interacts with the system and require different handling techniques.
 - 1.1. General errors that apply to most cases are due to missing information (e.g., users do not fill in the required fields in an entry form) and the system signals the error and shows witch fields need to be provided.
 - 1.2. Other errors might be related to entering invalid data and the error handling mechanism should try either to avoid that or to raise the error and suggest the correction.

Early Aspect: Compatibility

Requirements:

1. As the Health Watcher system exchanges information with the SSVS system (Sanitary Surveillance System), they must comply with the same mechanism of exchanging data.
 - 1.1. Both systems should agree on a protocol for exchanging the data to avoid incompatibilities in the integration.
 - 1.2. The data exchanged should comply with a pre-defined and agreed format (e.g., XML).
 - 1.3. Security issues might be involved to guarantee a safe exchange of information between both systems.

Early Aspect: Usability

Requirements:

1. The system should have an easy to use GUI, as any person who has access to the internet should be able to use the system.
2. The system should have an on-line HELP to be consulted by any person that uses it.

- 2.1. Several functionalities of the system should provide context sensitive help for the user (e.g., explain what are the types of complaints)

Early Aspects: Standards

Requirements:

1. The system must be developed according to the standards established by Company X, responsible for the norms and standardization of systems for the City Hall.

Early Aspect: Operational environment

Requirements:

1. The user interface must be implemented using Servlets.
2. SOFTWARE: One license for the Microsoft Windows for the workstation.
3. HARDWARE: One computer with: Pentium III processor, 256 MB of RAM memory, net card 3Com 10/100. This equipment shall be used by the attendant as a workstation

2.3 Crosscutting Relationships

Table 1 shows for early aspect which viewpoints it crosscuts and which requirements the early aspect affects. This type of information is provided by EA-Miner.

Early Aspect	Viewpoints Crosscut (requirements)
Availability	Employee (requirement 2 and all sub-requirements) Citizen (requirements 2 to 4 and all sub-requirements)
Security	Employee (requirement 2 and all sub-requirements) Citizen (requirements 2 to 4 and all sub-requirements) SSVS system (all requirements)
Performance	Employee (requirement 2 and all sub-requirements) Citizen (requirements 2 to 4 and all sub-requirements)
Concurrency	Employee (requirement 2 and all sub-requirements) Citizen (requirements 2 to 4 and all sub-requirements)
Persistence	Employee (all requirements) Citizen (all requirements) Complaint (all requirements)
Distribution	Employee (requirement 2 and all sub-requirements)

	Citizen (requirements 2 to 4 and all sub-requirements)
Error and Exception Handling	Employee (requirement 2 and all sub-requirements) Citizen (requirements 2 to 4 and all sub-requirements)
Compatibility	SSVS system (all requirements)
Usability	Employee (requirement 2 and all sub-requirements) Citizen (requirements 2 to 4 and all sub-requirements)
Standards	All viewpoints (except SSVS System) and all requirements
Operational environment	All viewpoints (except SSVS System) and all requirements

Table 1 - Viewpoints Crosscut by each Early Aspect

2.4 Trade-off points and aspects interaction

The trade-off points are considered to be the requirements belonging to a viewpoint in which different early aspects apply. From the Requirements Engineering perspective these points might raise conflicts that need to be further investigated and maybe involve negotiation with stakeholders as some early aspects might negatively influence each other. Table 2 presents a summarized discussion on these trade-off issues for each viewpoint (the early aspects that crosscut each viewpoint can be obtained by EA-Miner and can be seen by observing Table 1 above).

Viewpoint	Trade-off points
Employee	Most early aspects (availability, security, performance, concurrency, persistence, distribution, error and exception handling, usability, standards, operational environment) apply to the employee viewpoint. Performance generally contributes negatively to security and positively to availability [1]. Distribution can contribute positively to performance as different servers can be used to load the balance (application and web server). Concurrency can contribute negatively to persistence as concurrent users can corrupt the data, therefore requiring a good transaction management approach. Error handling contributes positively to usability as users can know when errors happen and what actions to take to correct them.
Citizen	The same early aspects and explanation for Employee applies.
SSVS System	(Security and Compatibility) early aspects apply to all requirements of SSVS system. The health watcher system interacts with the SSVS system and the interaction must respect the same protocol and be secure at the same time. Ensuring security might contribute

	negatively for compatibility of the security mechanism selected overcomplicates the protocol for establishing the data exchange between the two systems.
Complaint	(Persistence, standards and operation environment) apply to this viewpoint. There is no restriction from the operational environment to use a specific database and with respect to standards of the company need to be checked to see if any constraints apply. Therefore, apparently no trade-offs need to be considered here.

Table 2 - Trade-off points

References

1. A. Rashid, A. Moreira, and J. Araujo. *Modularisation and Composition of Aspectual Requirements*. in *2nd International Conference on Aspect Oriented Software Development (AOSD)*. 2003. Boston, USA: ACM.
2. A. Rashid, P. Sawyer, A. Moreira, and J. Araujo. *Early Aspects: a Model for Aspect-Oriented Requirements Engineering*. in *International Conference on Requirements Engineering (RE)*. 2002. Essen, Germany: IEEE.
3. A. Sampaio, R. Chitchyan, A. Rashid, and P. Rayson. *EA-Miner: A tool for automating aspect-oriented requirements identification*. in *20th IEEE/ACM International Conference on Automated Software Engineering (ASE2005)* 2005. Long Beach, California, USA.
4. A. Sampaio, N. Loughran, A. Rashid, and P. Rayson. *Mining Aspects in Requirements*. in *Early Aspects 2005: Aspect-Oriented Requirements Engineering and Architecture Design Workshop (held with AOSD 2005)*. 2005. Chicago, Illinois, USA.