Deliverable D7.4

Partners’ webservices final release

Revision: Final

Due date of delivery: 31.12.2013
Actual submission date: 21/01/2014

Start date of project: 01.01.2011
Duration: 36 months

Organisation name of lead contractor for this deliverable:
FDCGO
Revision History

<table>
<thead>
<tr>
<th>Revision</th>
<th>Date</th>
<th>Author</th>
<th>Organization</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>31.12.2013</td>
<td>FDCGO</td>
<td>FDCGO</td>
<td></td>
</tr>
<tr>
<td>1.1</td>
<td>13.01.2014</td>
<td>FDCGO</td>
<td>FDCGO</td>
<td></td>
</tr>
</tbody>
</table>

Statement of originality

This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both.

Authors

Andrea Agnoletto, FDCGO
Valerio Gower, FDCGO
Executive Summary

The overall goal of the ETNA thematic network was to establish a European Web Portal able to provide a unified access to information on European ICT assistive products, on related organizations, services, and to allow access to repositories of freeware, open source software products and tools useful for eAccessibility.

The initial vision was that of a single Portal evolving from the already-existing EASTIN system (European Assistive Technology Information Network) and including two major components: the ETNA information system – a search engine that aggregates information from various providers and repositories all over Europe and beyond, under responsibility of the ETNA network – and a virtual community connecting all stakeholders, under responsibility of ATIS4all, another Thematic Network belonging to the same cluster.

In the course of both projects, as tangible results gradually took shape, technical considerations and thoughts related to future sustainability led to the decision of having two distinct but coexisting Portals, communicating with each other:

- an Information System – a search engine that aggregates information from various providers and repositories all over Europe and beyond – to be developed by the ETNA network and later taken up by the EASTIN Association (the European Assistive Technology Information Network);
- a Community – connecting all stakeholders through a collaborative portal – to be developed by the ATIS4All Network and later taken up by Technosite (the ATIS4all coordinator).

The search engine has been developed as an extension of the current EASTIN system, thus it will be referred to as the EASTIN search engine.

This deliverable, intended for a technical audience, updates and substitutes the deliverable D6.2 “Partners’ web services 2nd release”. It provides the specifications for the web services that should be developed by the partners who wish to make available the data contained in their local information system to the EASTIN search engine. These partners have been called “Providers type A” (see Deliverable D3.2), contrary to the partners who prefer to upload information manually through a purposely developed upload tool (“Providers type B”).

In Annex 2 the three examples of wsdl (Web Services Description Language) files for web services implemented according to the specifications given in this deliverable are reported.
Table of contents

- Executive Summary ........................................................................................................................................ 3
- Introduction .................................................................................................................................................. 5
- Web service specifications .......................................................................................................................... 5
- Batch methods ............................................................................................................................................. 6
- Live methods ................................................................................................................................................ 8
- Custom data types ....................................................................................................................................... 13
- Annex 1 – ETNA features vocabulary ......................................................................................................... 18
- Annex 2 – Example of partner’s WSDL files ............................................................................................... 26
Introduction

This deliverable provides the specifications for the web services that should be developed by the partners who wish to make available the data contained in their local information system to the EASTIN search engine.

Annex 2 reports, as examples, the wsdl files of three ETNA partners that implemented the web services according to the specifications given in this document.

The specifications described in this deliverable, as well as the ETNA search engine, have been designed with backward compatibility with the web services of the current EASTIN system in mind. Therefore, the data provided by the EASTIN partners will always be visible in the ETNA system, even if they have not yet updated their web services according to the new specifications.

Web service specifications

In EASTIN Web services there are two groups of Web methods: the batch methods and the live methods.

Batch methods are invoked by automatic processes which run in EASTIN central server and are used to update some almost-static information inside the EASTIN Portal (for example the ISO tree, the keyword lists, etc.). These methods are called with different frequencies (from once a day to once a month), depending on how often the retrieved information content is supposed to change inside each EASTIN partner’s local system. For instance, the method which returns the description of ISO classes (used to update the ISO tree in EASTIN Portal) is called once a month, because the ISO classification is supposed to be almost constant. Instead the method which returns the number of products for a given ISO class is invoked once a day because new products could be often added inside the EASTIN partners’ local systems or perhaps their description could have been modified.

Live methods are invoked directly by the end users through the EASTIN Portal Web pages. They return the results of searches inside EASTIN partners’ databases about products, actors (also called “organisations” in the EASTIN Web pages), and associated information (also called “library in the EASTIN Web pages).

In the following description the name of basic data types derives from the SOAP – XML Schema Definition standard (XSD). Each partner must cast these types to the specific types of the language/platform adopted to implement the Web services.
Batch methods

integer GetIsoClassProductCount(string isoCode)

Input parameters:
- string isoCode: a string representing a single ISO class (for example “12.22”).

Returns:
- integer representing the number of products contained in the ISO class passed in the input parameter. Returns zero if no product belongs to the ISO class.

Frequency:
- Once a day; the first call to the Web method is executed at any moment between 04.00 AM (GMT +1:00) and 04.59 AM (GMT +1:00). The time at which the last call is executed is not defined.

This is a batch method which returns the number of products belonging to the ISO class whose ISO code is passed as a string parameter to the method. If no product belonging to the ISO class is found the method returns zero. The method is designed to work in batch mode. Once a day the ISO classification tree which is stored in the EASTIN central repository is visited by the batch process and for each node which is a leaf of the ISO tree the method is called, passing the ISO code of that node as parameter (so the number of calls to the Web method is equal to the number of leaf nodes of the EASTIN ISO classification tree). The method retrieves the number of products belonging to that ISO class and this information is updated in the EASTIN ISO tree.

IsoClassLocalizationDto GetIsoClassLocalization(string isoCode)

Input parameters:
- string isoCode: a string representing a single ISO class (for example “12.22”).

Returns:
- IsoClassLocalizationDto object containing the description of the ISO class passed in the input parameter. If no description is found returns the null object.

Frequency:
- Once a month; the first call to the Web method is executed on the second day of every month at any moment between 03.00 AM (GMT +1:00) and 03.59 AM (GMT +1:00). The time at which the last call is executed is not defined.

This is a batch method which returns a single object belonging to the class IsoClassLocalizationDto, which represents an element of the ISO classification. The method searches into the local database for information about the ISO class whose ISO code is passed as string parameter to the method. For example if the value “12.22” is passed, the method will search for information about 12.22 ISO class. The information items retrieved by the method and stored in the IsoClassLocalizationDto object are:

- the ISO code;
- the title of the ISO class;
- the scope note of the ISO class (if it exists).

If no information for the ISO class is found the method returns the null object. The

1 This method has to be implemented only by a restricted set of authorized partners. For further information please contact the EASTIN portal administrators.
method is designed to work in batch mode. Once a month the ISO classification tree which is stored in EASTIN central repository is visited by the batch process and for each node, which represents an ISO class, the method is called, passing the ISO code of that node as parameter (so the number of calls to the Web method is equal to the number of nodes of the EASTIN ISO classification tree). The method retrieves the information about that ISO class and this information is updated in the EASTIN Portal ISO tree. For a complete description of the IsoClassLocalizationDto object see below.

### KeywordDto[] GetKeywords()

**Input parameters:**
- none.

**Returns:**
- KeywordDto[]: an array of KeywordDto objects containing information about keywords. If no keyword is found returns a not null KeywordDto[] array with zero elements.

**Frequency:**
- Once a month; the unique call to the Web method is executed on the third day of every month at any moment between 03.00 AM (GMT +1:00) and 03.59 AM (GMT +1:00).

This is a batch method which returns an array of objects belonging to the class KeywordDto. The method searches into the EASTIN partners’ local databases for the dictionary of (keywords -> ISO classes) which will be used in the keyword research of the EASTIN portal. This method requires no parameter. Each KeywordDto object contains the following information:
- the keyword id in the partner’s local database;
- the keyword text;
- an array of ISO codes which are related to this keyword.

If no keyword is found the method returns a not null array with zero elements. The method is designed to work in batch mode. Once a month the method is called and the returned information are updated in the EASTIN portal keyword lists. For a complete description of the KeywordDto object see below.

---

2 This method has to be implemented only by a restricted set of authorized partners. For further information please contact the EASTIN portal administrators.
Live methods

Product searches

```java
SmallProductDto[] FindSmallProducts(string[] isoCodes, FeatureDto[] features, string commercialName, string manufacturer, dateTime insertDateMin, dateTime insertDateMax)
```

Input parameters:
- `string[] isoCodes`: an array of strings representing ISO classes (for example ["12.22", "09.03.03"]);
- `FeatureDto[] features`: an array of `FeatureDto` objects (for a complete description of the `FeatureDto` object see below);
- `string commercialName`: the whole or a part of the commercial name of the products to be searched;
- `string manufacturer`: the whole or a part of the manufacturer name of the products to be searched;
- `dateTime insertDateMin`: the lower bound for the insert date of the products to be searched;
- `dateTime insertDateMax`: the upper bound for the insert date of the products to be searched.

Returns:
- `SmallProductDto[]`: array of `SmallProductDto` objects containing each a light set of information about a product (for a complete description of the `SmallProductDto` object see below). If no product is found returns a not null `SmallProductDto[]` array with zero elements.

This method returns an array of objects belonging to the class `SmallProductDto`. The method implements five different kinds of searches:

1. If the `isoCodes` array is not void the method searches for all products belonging to the ISO classes passed, using an `OR` statement. For example if ["12.22", "09.03.03"] is the isoCodes array, all products belonging to the 12.22 ISO class `OR` to the 09.03.03 class are returned.

2. If the `features` array is not void the method searches for all products that possess the indicated `FeatureDto` objects and whose measures for the respective features are compatible with the measure boundaries specified in the `FeatureDto` objects. For example if features contains the `FeatureDto`s ["Width (cm)", 30, 50], ["Height (cm)", 80, 100] the method will search for all products having some widths in the range [30, 50] `AND` having some heights in the range [80, 100]. Note that if a product declares for example to have a fixable width between 20 and 40 it should be included in the search results since for some of its configurations it satisfies the boundaries. The product is included in the search results only if the compatibility between its measures and the given boundaries present in the `FeatureDto` objects are satisfied for all `FeatureDto` objects. For a complete list of Features see Annex I – ETNA features vocabulary.

3. If `commercialName` is not void the method searches a matching between the words contained in the `commercialName` parameter and the respective data in the EASTIN partner’s local database. Since into the `commercialName` parameter there could be one or more words, the method must split the words and search inside its
database for products whose commercial name contains all these words (even if present as substrings inside of biggest strings). For example if commercialName = “quickie xenon” the method must search for all products whose commercial name contains both words “quickie” AND “xenon”.

4. If manufacturer is not void the method executes the search using the same criteria specified in 2 but applied to products’ manufacturer name.

5. If insertDateMin and insertDateMax are both not null all products whose insert date is included within the interval [insertDateMin, insertDateMax], endpoints included, are returned. These two parameters must be both not null or both null. If more than one parameter is not void at the same time, the results coming from the matches for each parameter are merged together with an AND logic: only results satisfying the conditions specified for each parameter are returned. If no product is found the method returns a not null SmallProductDto[] array with zero elements.

<table>
<thead>
<tr>
<th>ProductDto GetProduct(string productCode)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input parameters:</td>
</tr>
<tr>
<td>– string productCode: the id of the product in the EASTIN partner’s system.</td>
</tr>
<tr>
<td>Returns:</td>
</tr>
<tr>
<td>– ProductDto: an object containing detailed informations about a single product. If no product is found than returns the null object.</td>
</tr>
</tbody>
</table>

This method returns an object belonging to the class ProductDto (for a complete description of the ProductDto object see below). The method searches into EASTIN partner’s local databases for the product which has the id matching with the method parameter productCode. If no product is found the method returns the null object.

**Actor searches**

<table>
<thead>
<tr>
<th>SmallActorDto[] FindSmallActors(string actorType, string[] isoCodes, string[] icfCodes, string actorName, dateTime insertDateMin, dateTime insertDateMax)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input parameters:</td>
</tr>
<tr>
<td>– string actorType: the type of the actor;</td>
</tr>
<tr>
<td>– string[] isoCodes: an array of strings representing ISO classes (for example [“12.22”, “09.03.03”]);</td>
</tr>
<tr>
<td>– string[] icfCodes: an array of strings representing the EASTIN ICF classes (for example [“b1”, “d2”]) which are a subset of the official ICF classification;</td>
</tr>
<tr>
<td>– string actorName: the whole or a part of the name of the searched actor;</td>
</tr>
<tr>
<td>– dateTime insertDateMin: the lower bound for the insert date of the actors to be searched;</td>
</tr>
<tr>
<td>– dateTime insertDateMax: the upper bound for the insert date of the actors to be searched.</td>
</tr>
<tr>
<td>Returns:</td>
</tr>
<tr>
<td>– SmallActorDto[]: an array of SmallActorDto objects containing each a light set of information about an actor (for a complete description of the SmallActorDto object see below). If no actor is found returns a not null SmallActorDto[] array with zero elements.</td>
</tr>
</tbody>
</table>
This method returns an array of objects belonging to the class `SmallActorDto`. The method implements five different kinds of searches:

1. If the `type` parameter is not void the method searches for all actors belonging to the specified type; the possible values for type are: “companies”, “projects” and “serviceproviders”.
2. If the `isoCodes` array is not void the method searches for all actors belonging to the ISO classes passed, using an `OR` statement. For example if `[“12.22”, “09.03.03”]` is the isoCodes array, all actors belonging to the 12.22 ISO class `OR` to the 09.03.03 class are returned.
3. If the `icfCodes` array is not void the method searches for all actors belonging to the ICF classes passed, using an `OR` statement. For example if `[“b1”, “d2”]` is the icfCodes array, all actors belonging to the b1 ICF class `OR` to the d2 class are returned.
4. If `actorName` is not void the method searches a matching between the words contained in the actorName parameter and the respective data in the EASTIN partner’s local database. Since into the actorName parameter there could be one or more words, the method must split the words and search inside its database for actors whose name contains all these words (even if present as substrings inside of biggest strings). For example if `actorName = “metlex ltd”` the method must search for all actors whose name contains both words “metlex” `AND` “ltd”.
5. If `insertDateMin` and `insertDateMax` are both not null all actors whose insert date is included within the interval `[insertDateMin, insertDateMax]`, endpoints included, are returned. These two parameters must be both not null or both null. If more than one parameter is not void at the same time, the results coming from the matches for each parameter are merged together with an `AND` logic: only results satisfying the conditions specified for each parameter are returned. If no actor is found returns a not null `SmallActorDto[]` array with zero elements.

**ActorDto GetActor(string actorType, string actorCode)**

Input parameters:
- `string actorType`: the type of the actor;
- `string actorCode`: the id identifying a single actor inside the EASTIN partner’s local system.

Returns:
- `ActorDto`: an object containing detailed informations about a single actor (for a complete description of the `ActorDto` object see below). If no actor is found than returns the null object.

This method returns an object belonging to the class `ActorDto`. The method searches into EASTIN partner’s local database for the actor of the type specified in the `actorType` parameter which has the id matching with the method parameter `actorCode`. If no actor is found the method returns the null object.

**Associated information searches**

**SmallAssociatedInfoDto[] FindSmallAssociatedInfos(string infoType, string[] isoCodes, string[] icfCodes, string title, string author, dateTime insertDateMin, dateTime insertDateMax)**
Input parameters:
- `string infoType`: the type of the associated information document;
- `string[] isoCodes`: an array of strings representing ISO classes (for example ["12.22", "09.03.03"]);
- `string[] isoCodes`: an array of strings representing EASTIN ICF classes (for example ["b1", "d2"]);
- `string title`: the whole or a part of the title (in the original language or in English) of the searched associated information document;
- `string author`: the whole or a part of the author names of the searched associated information document;
- `date Time insertDateMin`: the lower bound for the insert date of the associated information documents to be searched;
- `date Time insertDateMax`: the upper bound for the insert date of the associated information documents to be searched.

Returns:
- `SmallAssociatedInfoDto[ ]`: an array of `SmallAssociatedInfoDto` objects containing each a light set of information about an associated information document (for a complete description of the `SmallAssociatedInfoDto` object see below). If no associated information document is found returns a not null `SmallAssociatedInfoDto[ ]` array with zero elements.

This method returns an array of objects belonging to the class `SmallAssociatedInfoDto`. The method implements six different kinds of searches:

1. If the `type` parameter is not void the method searches for all associated information documents belonging to the specified type; the possible values are: "articles", "casedescriptions", "ideas", "faqs", "forums", "news" and "regulations".
2. If the `isoCodes` array is not void the method searches for all associated information documents belonging to the ISO classes passed, using an OR statement. For example if ["12.22", "09.03.03"] is the `isoCodes` array, all associated information documents belonging to the 12.22 ISO class OR to the 09.03.03 class are returned.
3. If the `icfCodes` array is not void the method searches for all associated information documents belonging to the ICF classes passed, using an OR statement. For example if ["b1", "d2"] is the `icfCodes` array, all associated information documents belonging to the b1 ICF class OR to the d2 class are returned.
4. If `title` is not void the method searches a matching between the words contained in the title parameter and the respective data in the EASTIN partner’s local database. Since into the title parameter there could be one or more words, the method must split the words and search inside its database for associated information documents whose title (in original language OR in English if present) contains all these words (even if present as substrings inside of biggest strings). For example if title = “a guide to wheeled walking frames” the method must search for all associated information documents whose original title or whose English title contain all words “a”, “guide”, “to”, “wheeled”, “walking” and “frames”.
5. If the `author` parameter is not void the method executes the search using the same criteria specified in 4 but applied to the name of the authors of the associated information document (in this case no distinction is needed between original language and English).
6. If `insertDateMin` and `insertDateMax` are both not null all associated information documents whose insert date is included within the interval \([\text{insertDateMin}, \text{insertDateMax}],\) endpoints included, are returned. These two parameters must be both not null or both null.

If more than one parameter is not void at the same time, the results coming from the matches for each parameter are merged together with an \(\text{AND}\) logic: only results satisfying the conditions specified for each parameter are returned. If no associated information document is found the method returns a not null `SmallAssociatedInfoDoc[]` array with zero elements.

```java
AssociatedInfoDto GetAssociatedInfo(string infoType, string associatedInfoCode)
```

Input parameters:
- `infoType`: the type of the associated information document;
- `associatedInfoCode`: the id identifying a single associated information document inside the EASTIN partner’s local systems.

Returns:
- `AssociatedInfoDto`: an object containing detailed information about a single associated information document (for a complete description of the `AssociatedInfoDto` object see below). If no associated information document is found than returns the null object.

The method searches into the EASTIN partner’s local database for the associated information document of the type specified in the `infoType` parameter which has the id matching with the method parameter `associatedInfoCode`. If no associated information document is found the method returns the null object.
Custom data types

As we have seen the EASTIN Web services return basic SOAP types, such as String, Int and Date, but also custom defined types. A complete description of EASTIN custom defined types follows below. All mandatory fields are marked with a "*" (all the other fields can be considered as nullable). For the array fields in case they are empty do not assign a null value to them but a not null array of zero elements.

IsoClassLocalizationDto
- string IsoCode*: the code of the ISO class;
- string Title*: the name of the ISO class;
- string ScopeNote: the ISO class description.

KeywordDto
- string KeywordId*: the id of the keyword in the partner’s local database;
- string Text*: the keyword text;
- string[] IsoCodes*: the array of all ISO classification codes related to the keyword (for example ["12.22", "09.03.03"]).

FeatureDto
- integer FeatureId*: the id of the EASTIN Taxonomy item;
- decimal ValueMin: the lower bound value of the measure specified for this feature;
- decimal ValueMax: the upper bound value of the measure specified for this feature.

SmallProductDto
- string ProductCode*: the id of the product in the partner’s local database;
- string IsoCodePrimary*: the primary ISO Code of the product (for example “09.03.03”);
- string[] IsoCodesOptional: the array of all secondary ISO classification codes of the product (for example [“12.22”, “09.03.03”]);
- string CommercialName*: the commercial name of the product;
- string ManufacturerCode*: the id of the product’s manufacturer in the partner’s local database;
- string ManufacturerOriginalFullName*: the full name in the original language of the product’s manufacturer;
- dateTime InsertDate*: the insert date of the product;
- dateTime LastUpdateDate*: the last update date of the product;
- string ThumbnailImageUrl: the URL of the small format picture of the product (used when displaying list of products in EASTIN Portal). The URL must be accessible on the Web by the end user’s browser.

ProductDto
- string ProductCode*: the id of the product in the partner’s local database;
- string IsoCodePrimary*: the primary ISO Code of the product (for example “09.03.03”);
- string[] IsoCodesOptional: the array of all secondary ISO classification codes of the product (for example [“12.22”, “09.03.03”]);
string CommercialName*: the commercial name of the product;
string ManufacturerCode*: the id of the product’s manufacturer in the partner’s local database;
string ManufacturerOriginalFullName*: the full name in the original language of the product’s manufacturer;
dateTime InsertDate*: the insert date of the product;
dateTime LastUpdateDate*: the last update date of the product;
string ThumbnailImageUrl: the URL of the small format image of the product (used when displaying list of products in the EASTIN portal). The URL must be accessible on the Web by the end user’s browser.
string ManufacturerAddress: the address of the product’s manufacturer;
string ManufacturerPostalCode: the postal code of the product’s manufacturer;
string ManufacturerTown: the town of the product’s manufacturer;
string ManufacturerCountry*: the country code of the product’s manufacturer in ISO 3166-1-alpha-2 code (for example “IT”, “US”, etc.);
string ManufacturerPhone: the phone of the product’s manufacturer;
string ManufacturerFax: the fax of the product’s manufacturer;
string ManufacturerEmail: the email of the product’s manufacturer;
string ManufacturerSkype: the Skype account name of the product’s manufacturer;
string ManufacturerWebSiteUrl: the Web site URL of the product’s manufacturer;
string[] ManufacturerSocialNetworkUrls: an array of URLs linking to the product’s manufacturer page inside the main social networks (for example Facebook, Twitter, LinkedIn, etc.);
string ImageUrl: the URL of the big format image of the product (used when displaying the detail view of the product in the EASTIN portal). The URL must be accessible on the Web by the end user’s browser;
string OriginalDescription: the description of the product in the original language;
string EnglishDescription: the description of the product in English;
string OriginalUrl: the URL of the Web page in the original language on the original EASTIN partner’s Web site in which the product is presented. The URL must be accessible on the Web by the end user’s browser;
string EnglishUrl: the URL of the Web page in English on the original EASTIN partner’s Web site in which the product is presented. The URL must be accessible on the Web by the end user’s browser;
string OriginalDownloadUrl: the URL of the download Web page in the original language on the original EASTIN partner’s Web site in which the product is presented. The URL must be accessible on the Web by the end user’s browser;
string EnglishDownloadUrl: the URL of the download Web page in English on the original EASTIN partner’s Web site in which the product is presented. The URL must be accessible on the Web by the end user’s browser;
string[] UserManualUrls: an array containing the URLs of product’s user manuals;
string[] VideoUrls: an array containing the URLs of product’s demo videos;
string[] BrochureUrls: an array containing the URLs of product’s brochures;
string[] FurtherInfoUrls: an array containing the URLs of eventual other information present on the Web related to the product;
FeatureDto[] Features: an array of FeatureDto objects containing all the EASTIN Taxonomy features (with the eventual measure values) for this product.

SmallActorDto
string ActorCode*: the id of the actor in the EASTIN partner’s local database;
string OriginalFullName*: the full name of the actor in the original language;
string Country*: the country code of the actor in ISO 3166-1-alpha-2 code (for example “IT”, “US”, etc.);
dateTime InsertDate*: the insert date of the actor in the EASTIN partner’s local database;
dateTime LastUpdateDate*: the insert date of the actor in the EASTIN partner’s local database;

ActorDto
- string ActorCode*: the id of the actor in the EASTIN partner’s local database;
- string OriginalFullName*: the full name of the actor in the original language;
- string Country*: the country code of the actor in ISO 3166-1-alpha-2 code (for example “IT”, “US”, etc.);
- dateTime InsertDate*: the insert date of the actor in the EASTIN partner’s local database;
- dateTime LastUpdateDate*: the insert date of the actor in the EASTIN partner’s local database;
- string ShortName*: the short name of the actor;
- string EnglishFullName*: the full name of the actor in English;
- string OriginalDescription: the description of the Actor in the original language;
- string EnglishDescription: the description of the Actor in English;
- dateTime StartDate*: the start date of the actor
- dateTime EndDate: the end date of the actor
- string ContactBody: the reference organization of the actor;
- string Address: the address of the actor;
- string PostalCode: the postal code of the actor;
- string Town: the town of the actor;
- string Phone: the phone of the actor;
- string Fax: the fax of the actor;
- string Email: the email of the actor;
- string Skype: the Skype account name of the actor;
- string WebSiteUrl: the Web site URL of the actor. The URL should be accessible on the Web by the end user’s browser;
- string ContactPersonFullName: the complete name of the contact person for the actor;
- string OriginalUrl: the URL of the Web page in the original language on the original EASTIN partner’s Web site in which the actor is presented. The URL must be accessible on the Web by the end user’s browser;
- string EnglishUrl: the URL of the Web page in English on the original EASTIN partner’s Web site in which the actor is presented. The URL must be accessible on the Web by the end user’s browser
- string[] SocialNetworkUrls: an array of URLs linking to the actor page inside the main social networks (for example Facebook, Twitter, LinkedIn, etc.);
- string[] IcfCodes*: the array of all EASTIN ICF classification codes of the actor (for example [“b1”, “d2”]);
- string[] IsoCodes*: the array of all ISO classification codes of the actor (for example [“12.22”, “09.03.03”]);

SmallAssociatedInfoDto
- **string** AssociatedInfoCode*: the ID of the associated information document in the EASTIN partner’s local database;
- **string** Authors*: a string containing the names (or the initials) of the authors of the associated information document (this is not an array but a single string);
- **string** OriginalTitle*: the original title in the native language of the associated information document
- **string** EnglishTitle*: the English translation of the original title of the associated information document
- **string** OriginalLanguage*: the ISO 639-1 code of the native language of the associated information document (for example: "en", "it", "de");
- **dateTime** InsertDate*: the insert date of the associated information document in the EASTIN partner’s local database;
- **dateTime** LastUpdateDate*: the last update date of the associated information document in EASTIN partner’s local database.

**AssociatedInfoDto**
- **string** AssociatedInfoCode*: the ID of the associated information document in the EASTIN partner’s local database;
- **string** Authors*: a string containing the names (or the initials) of the authors of the associated information document (this is not an array but a single string);
- **string** OriginalTitle*: the original title in the native language of the associated information document
- **string** EnglishTitle*: the English translation of the original title of the associated information document
- **string** OriginalLanguage*: the ISO 639-1 code of the native language of the associated information document (for example: "en", "it", "de");
- **dateTime** InsertDate*: the insert date of the associated information document in the EASTIN partner’s local database;
- **dateTime** LastUpdateDate*: the last update date of the associated information document in EASTIN partner’s local database
- **integer** PublicationYear*: the publication year of the associated information document;
- **string** PublishingDetails: the publishing details (for example the publishing house) of the associated information document;
- **string** OriginalAbstract: the abstract of the associated information document in the original language;
- **string** EnglishAbstract: the abstract of the associated information document in the original language;
- **string** OriginalUrl: the URL of the Web page in the original language on the original EASTIN partner’s Web site in which the associated information document is presented. The URL must be accessible on the Web by the end user’s browser;
- **string** EnglishUrl: the URL of the Web page in English on the original EASTIN partner’s web site in which the associated information document is presented. The URL must be accessible on the Web by the end user’s browser;
- **string** OriginalDownloadUrl: the URL for the download of the associated information document in the original language;
- **string** EnglishDownloadUrl: the URL for the download of the associated information document in English;
- **string** ImageUrl: the URL of the picture related to the associated information document (used when displaying the detail view of the associated information
document in EASTIN Portal). The URL must be accessible on the Web by the end user’s browser;

- **string[]** FurtherInfoUrls: an array containing the URLs of eventual other information present on the Web related to the associated information document;
- **string[]** IcfCodes*: the array of all EASTIN ICF classification codes of the associated information document (for example [“b1”, “d2”]);
- **string[]** IsoCodes*: the array of all ISO classification codes of the associated information document (for example [“12.22”, “09.03.03”]);
Annex 1 – ETNA features vocabulary

As part of the ETNA taxonomy a vocabulary of features has been introduced in the ETNA information system to standardize the description of products’ technical details. The Vocabulary is based on a two level hierarchy made up of Groups and Features. Homogeneous Features are grouped together in the same Group. For example the Features “Windows”, “Mac OS”, “Linux”, “Chrome OS”, etc... are all grouped in the Group “Operating System”, while “Printer”, “Visual display”, “Tactile display”, etc... are grouped in the Group “Output devices”. Features can be of two types: Measures, that can have a numeric value or an interval specified (e.g. weight, length, ...), and Attributes, that do not have a specified value (i.e. are Boolean features). The table below lists all the features (and their respective IDs) identified at the time this deliverable has been written.

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Overall dimensions</td>
<td>cluster</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Width (cm)</td>
<td>measure</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Length (cm)</td>
<td>measure</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Height (cm)</td>
<td>measure</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Weight (kg)</td>
<td>measure</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Capacity/Range</td>
<td>cluster</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Magnification (x)</td>
<td>measure</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Number of keys</td>
<td>measure</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Number of input channels/devices/messages</td>
<td>measure</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Number of output channels/devices/messages</td>
<td>measure</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Signal range (m)</td>
<td>measure</td>
<td></td>
</tr>
<tr>
<td>325</td>
<td>Max sound/speech volume (dB)</td>
<td>measure</td>
<td></td>
</tr>
<tr>
<td>326</td>
<td>Max ringer/alarm volume (dB)</td>
<td>measure</td>
<td></td>
</tr>
<tr>
<td>327</td>
<td>M rating (Hearing Aid Compatibility)</td>
<td>measure</td>
<td>M rating corresponds to interference of Mobile phones with hearing aids setted to “microphone mode”. The higher the number following the ‘M’ the clearer the sound should be.</td>
</tr>
<tr>
<td>328</td>
<td>T rating (Hearing Aid Compatibility)</td>
<td>measure</td>
<td>T rating corresponds to interference of mobile phones with hearing aids setted in “t-coil mode”. The higher the number following the ‘T’ the clearer the sound should be.</td>
</tr>
<tr>
<td>12</td>
<td>Power sources</td>
<td>cluster</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>Battery - disposable</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>Battery - rechargeable</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>Mains electric</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>Power via USB</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>Activation methods</td>
<td>cluster</td>
<td>How the device (or software) is activated</td>
</tr>
<tr>
<td>18</td>
<td>Electro Myo Graphic Signal (EMG)</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>Eye blink</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>Acoustic</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>Eye gaze</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>Speech Recognition</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>Mechanical (push, pull, grasp,...)</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>Sip/Puff</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>Tilt</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>57</td>
<td>Browsers</td>
<td>cluster</td>
<td>Type of browser supported by the device or software</td>
</tr>
</tbody>
</table>

---

3 For more information about the ETNA taxonomy and the vocabulary of features see deliverables D4.2 and D5.1
<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>58</td>
<td>Chrome</td>
<td>attribute</td>
</tr>
<tr>
<td>59</td>
<td>Firefox</td>
<td>attribute</td>
</tr>
<tr>
<td>60</td>
<td>Internet Explorer</td>
<td>attribute</td>
</tr>
<tr>
<td>61</td>
<td>Safari</td>
<td>attribute</td>
</tr>
<tr>
<td>62</td>
<td>Opera</td>
<td>attribute</td>
</tr>
<tr>
<td>63</td>
<td><strong>Languages</strong></td>
<td>cluster</td>
</tr>
<tr>
<td>64</td>
<td>Danish</td>
<td>attribute</td>
</tr>
<tr>
<td>65</td>
<td>Dutch</td>
<td>attribute</td>
</tr>
<tr>
<td>66</td>
<td>English</td>
<td>attribute</td>
</tr>
<tr>
<td>67</td>
<td>French</td>
<td>attribute</td>
</tr>
<tr>
<td>68</td>
<td>German</td>
<td>attribute</td>
</tr>
<tr>
<td>294</td>
<td>Greek</td>
<td>attribute</td>
</tr>
<tr>
<td>295</td>
<td>Italian</td>
<td>attribute</td>
</tr>
<tr>
<td>296</td>
<td>Portuguese</td>
<td>attribute</td>
</tr>
<tr>
<td>297</td>
<td>Spanish</td>
<td>attribute</td>
</tr>
<tr>
<td>298</td>
<td>Bulgarian</td>
<td>attribute</td>
</tr>
<tr>
<td>299</td>
<td>Czech</td>
<td>attribute</td>
</tr>
<tr>
<td>300</td>
<td>Estonian</td>
<td>attribute</td>
</tr>
<tr>
<td>301</td>
<td>Finnish</td>
<td>attribute</td>
</tr>
<tr>
<td>302</td>
<td>Hungarian</td>
<td>attribute</td>
</tr>
<tr>
<td>303</td>
<td>Latvian</td>
<td>attribute</td>
</tr>
<tr>
<td>304</td>
<td>Lithuanian</td>
<td>attribute</td>
</tr>
<tr>
<td>305</td>
<td>Maltese</td>
<td>attribute</td>
</tr>
<tr>
<td>306</td>
<td>Other European Languages</td>
<td>attribute</td>
</tr>
<tr>
<td>307</td>
<td>Polish</td>
<td>attribute</td>
</tr>
<tr>
<td>308</td>
<td>Romanian</td>
<td>attribute</td>
</tr>
<tr>
<td>309</td>
<td>Slovak</td>
<td>attribute</td>
</tr>
<tr>
<td>310</td>
<td>Slovenian</td>
<td>attribute</td>
</tr>
<tr>
<td>311</td>
<td>Swedish</td>
<td>attribute</td>
</tr>
<tr>
<td>330</td>
<td>Irish</td>
<td>attribute</td>
</tr>
<tr>
<td>332</td>
<td>Other European Languages</td>
<td>attribute</td>
</tr>
<tr>
<td>69</td>
<td><strong>Display characteristics</strong></td>
<td>cluster</td>
</tr>
<tr>
<td>70</td>
<td>Black/white display</td>
<td>attribute</td>
</tr>
<tr>
<td>71</td>
<td>Colour display</td>
<td>attribute</td>
</tr>
<tr>
<td>72</td>
<td>3D</td>
<td>attribute</td>
</tr>
<tr>
<td>73</td>
<td><strong>Linguistic representations</strong></td>
<td>cluster</td>
</tr>
<tr>
<td>74</td>
<td>Sign language</td>
<td>attribute</td>
</tr>
<tr>
<td>75</td>
<td>Braille</td>
<td>attribute</td>
</tr>
<tr>
<td>76</td>
<td>Alphabetic</td>
<td>attribute</td>
</tr>
<tr>
<td>77</td>
<td>Symbolic</td>
<td>attribute</td>
</tr>
<tr>
<td>78</td>
<td>Simplified</td>
<td>attribute</td>
</tr>
<tr>
<td>79</td>
<td><strong>Output devices</strong></td>
<td>cluster</td>
</tr>
<tr>
<td>80</td>
<td>Speakers/headphones</td>
<td>attribute</td>
</tr>
<tr>
<td>81</td>
<td>Printer</td>
<td>attribute</td>
</tr>
<tr>
<td>82</td>
<td>Visual screen/display</td>
<td>attribute</td>
</tr>
<tr>
<td>83</td>
<td>Tactile display</td>
<td>attribute</td>
</tr>
<tr>
<td>84</td>
<td>Vibrator</td>
<td>attribute</td>
</tr>
<tr>
<td>85</td>
<td>Voice synthesis</td>
<td>attribute</td>
</tr>
<tr>
<td>86</td>
<td>Recorded sound</td>
<td>attribute</td>
</tr>
<tr>
<td>87</td>
<td>Environmental control devices</td>
<td>attribute</td>
</tr>
<tr>
<td>89</td>
<td><strong>Functionalities</strong></td>
<td>cluster</td>
</tr>
<tr>
<td>91</td>
<td>Word prediction/completion</td>
<td>attribute</td>
</tr>
<tr>
<td>92</td>
<td>Spell correction</td>
<td>attribute</td>
</tr>
<tr>
<td>93</td>
<td>Abbreviation expansion</td>
<td>attribute</td>
</tr>
<tr>
<td>94</td>
<td>Highlights each word/sentence as it is read aloud</td>
<td>attribute</td>
</tr>
</tbody>
</table>

*Output devices (or software components) the product includes or is designed to be used with*
<table>
<thead>
<tr>
<th></th>
<th>Attribute Description</th>
<th>Cluster</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>95</td>
<td>Allows creation of macro function</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>96</td>
<td>Programmable/configurable</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>97</td>
<td>Calendar function</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>98</td>
<td>Reminder</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>Portable</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>101</td>
<td>Built-in microphone</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>102</td>
<td>Speech or acoustic signals on menus</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>103</td>
<td>Switch controlled scanning</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>104</td>
<td><strong>Input devices</strong></td>
<td>cluster</td>
<td><em>input devices (or software components) the product includes or is designed to be used with</em></td>
</tr>
<tr>
<td>105</td>
<td>Joystick</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>107</td>
<td>Keys/keyboard</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>108</td>
<td>Chording keyboard (e.g. Braille keyboard)</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>109</td>
<td>Mouse</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>110</td>
<td>Speech recognition</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>111</td>
<td>Switch</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>112</td>
<td>Touch screen</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>113</td>
<td>Track pad (touch pad)</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>114</td>
<td>Trackball</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>115</td>
<td>Movement tracking system</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>116</td>
<td>Eyegaze control system</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>117</td>
<td>Video camera/webcam</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>118</td>
<td>Microphone</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>119</td>
<td>Accelerometer</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>120</td>
<td>Biosignals sensor (EMG, EOG, EEG)</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>121</td>
<td><strong>Input adjustments</strong></td>
<td>cluster</td>
<td><em>Available adjustments or filtering options for the input devices (or software components)</em></td>
</tr>
<tr>
<td>122</td>
<td>Speed</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>123</td>
<td>Controls/keys activation delay</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>124</td>
<td>Type of scanning</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>125</td>
<td>Sensitivity</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>126</td>
<td>Scanning speed</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>127</td>
<td>Size of controls/keys</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>128</td>
<td>Number of controls/keys configuration</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>129</td>
<td>Font size on controls/keys</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>130</td>
<td>Colour of controls/keys</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>131</td>
<td>Filter on repeated activations</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>132</td>
<td>Key repeat rate</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>133</td>
<td>Microphone sensitivity</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>134</td>
<td><strong>Output adjustments</strong></td>
<td>cluster</td>
<td><em>Available adjustments or options for the output devices (or software components)</em></td>
</tr>
<tr>
<td>135</td>
<td>Enlargement/zoom</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>136</td>
<td>Font size</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>137</td>
<td>Contrast</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>138</td>
<td>Colours</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>139</td>
<td>Image reversal</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>140</td>
<td>Equalization control</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>141</td>
<td>Volume</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>142</td>
<td>Sound feedback</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>143</td>
<td><strong>Connectivity</strong></td>
<td>cluster</td>
<td><em>How the device (or software component) connects to other devices or services</em></td>
</tr>
<tr>
<td>144</td>
<td>PS2</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>145</td>
<td>Serial</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>146</td>
<td>USB</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>147</td>
<td>Bluetooth</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>148</td>
<td>infrared</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>149</td>
<td>Jack</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>150</td>
<td>Other wireless</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>151</td>
<td>WiFi</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>152</td>
<td>Cloud or internet based application</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>153</td>
<td>induction loop</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>154</td>
<td>Inductive coupling</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>157</td>
<td>Software license policies</td>
<td>cluster</td>
<td></td>
</tr>
<tr>
<td>158</td>
<td>Free and open source software</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>159</td>
<td>Proprietary</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>160</td>
<td>Software price policies</td>
<td>cluster</td>
<td></td>
</tr>
<tr>
<td>161</td>
<td>Free of charge</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>162</td>
<td>Bundled with operating system</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>331</td>
<td>Priced</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>163</td>
<td>Subdivisions</td>
<td>cluster</td>
<td></td>
</tr>
<tr>
<td>164</td>
<td>Stationary image-enlarging reading apparatus</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>165</td>
<td>Stationary image-enlarging reading apparatus with connection units for computers</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>166</td>
<td>Portable image-enlarging reading apparatus</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>167</td>
<td>Accessories for image-enlarging reading apparatus</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>169</td>
<td>Concha/in-the-ear hearing aids</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>170</td>
<td>Completely in-the-canal hearing aids</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>172</td>
<td>Behind-the-ear hearing-aids</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>173</td>
<td>Power behind-the-ear hearing-aids</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>175</td>
<td>Voice amplifiers for personal use</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>177</td>
<td>Electric typewriters without memory</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>178</td>
<td>Electric typewriters with memory</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>179</td>
<td>Braille typewriters</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>180</td>
<td>Stenotype machines</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>181</td>
<td>Electric Braille typewriters</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>183</td>
<td>Word-processing software</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>184</td>
<td>Desktop publishing software</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>186</td>
<td>Equipment for recording and/or replaying digital books</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>187</td>
<td>Digital note recorders</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>188</td>
<td>Cassette recorders</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>189</td>
<td>Accessories for recording and/or replaying sound</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>191</td>
<td>Real time captioning systems</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>192</td>
<td>Delayed captioning systems</td>
<td>attribute</td>
<td></td>
</tr>
</tbody>
</table>

**147** Bluetooth attribute

**148** infrared attribute

**149** Jack attribute

**150** Other wireless attribute

**151** WiFi attribute

**152** Cloud or internet based application attribute

**153** induction loop attribute

**154** Inductive coupling attribute

**157** Software license policies cluster

**158** Free and open source software attribute

**159** Proprietary attribute

**160** Software price policies cluster

**161** Free of charge attribute

**162** Bundled with operating system attribute

**331** Priced attribute

**163** Subdivisions cluster

**164** Stationary image-enlarging reading apparatus attribute

**165** Stationary image-enlarging reading apparatus with connection units for computers attribute

**166** Portable image-enlarging reading apparatus attribute

**167** Accessories for image-enlarging reading apparatus attribute

**169** Concha/in-the-ear hearing aids attribute

**170** Completely in-the-canal hearing aids attribute

**172** Behind-the-ear hearing-aids attribute

**173** Power behind-the-ear hearing-aids attribute

**175** Voice amplifiers for personal use attribute

**177** Electric typewriters without memory attribute

**178** Electric typewriters with memory attribute

**179** Braille typewriters attribute

**180** Stenotype machines attribute

**181** Electric Braille typewriters attribute

**183** Word-processing software attribute

**184** Desktop publishing software attribute

**186** Equipment for recording and/or replaying digital books attribute

**187** Digital note recorders attribute

**188** Cassette recorders attribute

**189** Accessories for recording and/or replaying sound attribute

**191** Real time captioning systems attribute

**192** Delayed captioning systems attribute
<p>| 193 | Captioning services | attribute | Devices for receiving or transmitting audio information using infrared light; included are, e.g., systems, transmitters and receivers for local one-way communication, e.g. personal remote voice transmission and voice transmission systems for auditorium. |
| 195 | Infrared (IR) systems for audio information | attribute | Loop amplifiers using electromagnetic waves to transmit audio information to hearing aids. Designed for use in one or more rooms. |
| 197 | Induction-loop amplifiers | attribute | Amplifiers for small loops, designed for one person. Included are, e.g., pillow loops, neck loops and clip-on equipment transmitting audio magnetically by using the pick-up loop inside the users hearing aids. |
| 201 | Symbolic voice output communication devices | attribute | Communication devices consisting of a touch-sensitive screen divided into a given number of fields. When activating a field an auditive output with digital or synthetic speech is produced. |
| 202 | Alphabetic communication devices | attribute | Writing based communication devices with a standard keyboard. Features screen output, synthetic speech output or printed output. |
| 204 | Face-to-face communication software | attribute | Software that allow a computer or a mobile device (smart phone, PDA, tablet, and other) to work as a communicator. |
| 205 | Tools for developing grids for communication software | attribute | |
| 207 | Mobile telephones | attribute | Mobile phones used for wireless calls on the public mobile network. |
| 209 | Telecommunication and telematics software | attribute | Software, specifically designed for person with motor, sensory or cognitive disability, for verbal and visual communication between computers via the computer network |
| 210 | Voice over IP Services | attribute | |
| 212 | Indicators with visual signal | attribute | Devices that indicate with light or other visual signal that something is happening in the place where the transmitter is; they can transform, e.g. audible signal to visual signal; Included are, e.g., electronic babysitters, door signals, door signal indicators and door warners. |
| 214 | Indicators with acoustic signals | attribute | Devices that indicate with sound that something is happening in the place where the transmitter is; they can transform, e.g., visual signal to audible signal or they can increase the volume of a normal device; Included are, e.g., rain indicators and computer-signal indicators. |
| 216 | Indicators with mechanical signals | attribute | Devices that indicate with tactile signal that something is happening in the place where the transmitter is; they can transform, e.g. audible or visual signal to vibration or other tactile signal; Included are, e.g., indicating devices with vibration. |
| 218 | Calendar software | attribute | Software designed to help users to manage daily life. Included are also software or equipment for mobile phones, paging receivers etc. |
| 219 | Electronic calendars | attribute | Devices designed to help users to manage daily life. Usually, featuring a watch or maybe a calendar and telling the user when an activity is about to begin. The device may be stationary, portable or of pocket size. Output is available as text, speech, sound or symbols. |
| 221 | Memory support products | attribute | Devices for notifying or reminding a person about |</p>
<table>
<thead>
<tr>
<th>ID</th>
<th>Description</th>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>223</td>
<td>Activity monitoring systems without personal identification</td>
<td>attribute</td>
<td>Alarm systems not featuring identification. The alarm is activated when a person leaves a certain area.</td>
</tr>
<tr>
<td>224</td>
<td>Remote video monitoring systems</td>
<td>attribute</td>
<td>Monitoring and positioning systems that operate via satellite navigation. Included are, e.g., global positioning systems (GPS).</td>
</tr>
<tr>
<td>225</td>
<td>Satellite navigation systems</td>
<td>attribute</td>
<td>Software based systems able to transform digital documents (e.g. text files) into voice output</td>
</tr>
<tr>
<td>226</td>
<td>Digital documents readers</td>
<td>attribute</td>
<td>Web based services that transform digital documents into audio files.</td>
</tr>
<tr>
<td>227</td>
<td>Digital document reading (text to speech) service</td>
<td>attribute</td>
<td>Hardware systems that transform the text written in paper document into alternative forms (e.g. enlarged text, synthetic speech, or tactile).</td>
</tr>
<tr>
<td>228</td>
<td>Paper documents reading devices</td>
<td>attribute</td>
<td>Software used for the scanning and recognition of documents. Included is e.g. OCR software with text-to-speech technology.</td>
</tr>
<tr>
<td>229</td>
<td>OCR software</td>
<td>attribute</td>
<td>Portable devices featuring dictionary lookup.</td>
</tr>
<tr>
<td>230</td>
<td>Portable scanner with electronic dictionary</td>
<td>attribute</td>
<td>Electric portable note taking devices with Braille reading line.</td>
</tr>
<tr>
<td>231</td>
<td>Software interfaces for computers and mobile devices</td>
<td>attribute</td>
<td>Complete software interfaces to facilitate the use of personal computer or mobile devices (e.g. tablet pc, smart phones).</td>
</tr>
<tr>
<td>232</td>
<td>Operating systems</td>
<td>attribute</td>
<td>Web browsers with special features (e.g. voice output) to navigate the web.</td>
</tr>
<tr>
<td>233</td>
<td>Keyboards with a special design</td>
<td>attribute</td>
<td>Keyboards including e.g. enlarged and miniaturized keyboards, headpointer keyboards, ergonomic keyboards and one-hand keyboards.</td>
</tr>
<tr>
<td>234</td>
<td>Programmable (concept) Keyboards</td>
<td>attribute</td>
<td>Touch-sensitive programmable boards which can be divided into different numbers and sizes of active areas (keys). Each active area can be programmed to perform different actions.</td>
</tr>
<tr>
<td>235</td>
<td>Keyboard shields and keyboard gloves</td>
<td>attribute</td>
<td>Software tools that allow to configure programmable keyboard and or to print overlay.</td>
</tr>
<tr>
<td>236</td>
<td>Software for accessing the computer in scanning mode</td>
<td>attribute</td>
<td>Systems that allow to control a computer, or other devices, through gaze.</td>
</tr>
<tr>
<td>237</td>
<td>Eyegaze systems</td>
<td>attribute</td>
<td>Systems that allow to control a computer, or other devices, through gaze.</td>
</tr>
<tr>
<td>238</td>
<td>Speech recognition software</td>
<td>attribute</td>
<td>Software for command and control or text input to computers by speech (speech-to-text programs).</td>
</tr>
<tr>
<td>239</td>
<td>Optical scanner, stationary</td>
<td>attribute</td>
<td>Stationary devices that can transform text or illustrations printed on paper into an electronic format.</td>
</tr>
<tr>
<td>240</td>
<td>Optical scanner, hand held</td>
<td>attribute</td>
<td>Handheld devices that transform text or illustrations printed on paper into an electronic format.</td>
</tr>
<tr>
<td>241</td>
<td>Datagloves</td>
<td>attribute</td>
<td>Glove fitted with sensors, which records the movements of different parts of the glove and translates the movement to an input.</td>
</tr>
<tr>
<td>242</td>
<td>EEG, EOG or EMG controlled input devices</td>
<td>attribute</td>
<td>Input devices controlled by electric signals activated by brainwave signals (EEG), by facial muscle movements (EMG) or by eye movements (EOG).</td>
</tr>
<tr>
<td>243</td>
<td>Switch interface</td>
<td>attribute</td>
<td>Interface to connect switches to a device, to allow, for example, the control in scanning mode.</td>
</tr>
<tr>
<td>Attribute Number</td>
<td>Description</td>
<td></td>
<td></td>
</tr>
<tr>
<td>------------------</td>
<td>-------------------------------------------------------------------------------------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>255</td>
<td>Accessories for input devices</td>
<td></td>
<td></td>
</tr>
<tr>
<td>257</td>
<td>On screen keyboards</td>
<td></td>
<td></td>
</tr>
<tr>
<td>258</td>
<td>Mouse control software</td>
<td></td>
<td></td>
</tr>
<tr>
<td>259</td>
<td>Word prediction and word completion software</td>
<td></td>
<td></td>
</tr>
<tr>
<td>260</td>
<td>Software for adjusting input devices response</td>
<td></td>
<td></td>
</tr>
<tr>
<td>261</td>
<td>Software based electronic dictionaries</td>
<td></td>
<td></td>
</tr>
<tr>
<td>262</td>
<td>Computer based sound collections</td>
<td></td>
<td></td>
</tr>
<tr>
<td>264</td>
<td>Touch screens</td>
<td></td>
<td></td>
</tr>
<tr>
<td>265</td>
<td>Trackballs, mousetrappers and touchpads</td>
<td></td>
<td></td>
</tr>
<tr>
<td>266</td>
<td>Traditional mouse devices and pen mouse devices</td>
<td></td>
<td></td>
</tr>
<tr>
<td>267</td>
<td>Joystick mouse device</td>
<td></td>
<td></td>
</tr>
<tr>
<td>268</td>
<td>Switch operated computer mice</td>
<td></td>
<td></td>
</tr>
<tr>
<td>269</td>
<td>Computer and console joysticks</td>
<td></td>
<td></td>
</tr>
<tr>
<td>271</td>
<td>Computer monitors</td>
<td></td>
<td></td>
</tr>
<tr>
<td>272</td>
<td>Screen filters</td>
<td></td>
<td></td>
</tr>
<tr>
<td>274</td>
<td>Braille displays                                      Returns text to Braille</td>
<td></td>
<td></td>
</tr>
<tr>
<td>276</td>
<td>Speech synthesizers</td>
<td></td>
<td></td>
</tr>
<tr>
<td>278</td>
<td>Magnifying software</td>
<td></td>
<td></td>
</tr>
<tr>
<td>279</td>
<td>Screen reader software</td>
<td></td>
<td></td>
</tr>
<tr>
<td>280</td>
<td>Software for adjusting color combination and text size</td>
<td></td>
<td></td>
</tr>
<tr>
<td>281</td>
<td>Software to modify the pointer appearance</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Descriptions:**

- **Accessories for input devices:** Includes, e.g., adaptors, cables, boards, multi ports and joysticks.
- **On screen keyboards:** Software applications that reproduce the keyboard on the device screen.
- **Mouse control software:** Software that allow controlling the mouse movement and/or click functions.
- **Word prediction and word completion software:** Software designed to facilitate typing by completing words and/or predicting the next word in a sentence.
- **Software for adjusting input devices response:** Software that allow to modify the functioning and behavior of input devices through adjustments and filtering (e.g., filtering out involuntary repeated keypress, or allowing “hot keys” and “short cuts”).
- **Software based electronic dictionaries:** Electronic dictionaries working as independent programs or in conjunction with other software e.g., word processing software programs. Included are e.g., spelling dictionaries, foreign language dictionaries etc. Included are also picture-, symbol-, and sign language dictionaries.
- **Computer based sound collections:** Collections of recorded words and sound effects for computers.
- **Touch screens:** Touch screens consist of a touch sensitive display, divided into fields. The size, number and function of the fields can be customized.
- **Trackballs, mousetrappers and touchpads:** A trackball in an upside-down mouse that rotates in place within a socket. The user rolls the ball to direct the cursor to the desired place on the screen. When using mouse trappers and touch pads movement of the finger produces a corresponding cursor movement.
- **Traditional mouse devices and pen mouse devices:** Controlled by one hand. The mouse pointer is controlled by moving the mouse device on a given surface.
- **Joystick mouse device:** Mouse devices with a joystick. Used to control the mouse pointer. Included are also mouth controlled joysticks.
- **Switch operated computer mice:** Type of computer mouse where you can control all the mouse functions through switches.
- **Computer and console joysticks:** Input devices, e.g., controllers, for playing electronic games on pc, Mac, Playstation, Nintendo, Xbox, or other platforms.
- **Computer monitors:** Monitors for desktop computers.
- **Screen filters:** Filters for computer monitors reducing specular reflection.
- **Braille displays:** Displays converting text to Braille.
- **Speech synthesizers:** Hardware or software system able to generate artificial human speech, also known as Text to Speech system.
- **Magnifying software:** Software that enlarges the text and graphics displayed on the screen of a computer or other electronic devices. May feature screen reading, colour choice and focus enhancement etc.
- **Screen reader software:** Software that interpret what is being displayed on the screen and present it to the user with text-to-speech, sound icons, or a Braille output device.
- **Software for adjusting color combination and text size:** Software that allow adjusting the color of text, background, images and other elements displayed on the screen, and/or to adjust the font size, to improve visualization.
- **Software to modify the pointer appearance:** Software to modify the size, color, and/or shape of the pointer on the screen.
<table>
<thead>
<tr>
<th>ID</th>
<th>Description</th>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>283</td>
<td>Single switches (switches with only one function)</td>
<td>Single switches</td>
<td>On/off switches (0/1 switches) which can be activated in different ways e.g. push activated, touch activated or sound activated etc. Single switches are used to control different products/assistive products.</td>
</tr>
<tr>
<td>284</td>
<td>Two-four function control switches</td>
<td>attribute</td>
<td>Switches controlling two to four functions.</td>
</tr>
<tr>
<td>285</td>
<td>Five-or-more-function-contacts</td>
<td>attribute</td>
<td>Five-or-more-function-contacts or wafer or star switch joysticks, where the function is similar to that of a digital joystick.</td>
</tr>
<tr>
<td>287</td>
<td>Remote controller</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>288</td>
<td>Receiver unit for environmental control</td>
<td>attribute</td>
<td></td>
</tr>
<tr>
<td>289</td>
<td>Switch latches and timers</td>
<td>attribute</td>
<td>Units controlling high current and low current devices with single switches.</td>
</tr>
<tr>
<td>291</td>
<td>Environmental control software</td>
<td>attribute</td>
<td>Software, standard or specifically designed, for controlling devices and automation systems.</td>
</tr>
<tr>
<td>293</td>
<td>Software for composing music</td>
<td>attribute</td>
<td>Software that allows a person to read and or compose music</td>
</tr>
<tr>
<td>312</td>
<td>Body movement controlled mice</td>
<td>attribute</td>
<td>Hardware devices that, using special sensors (e.g. video cameras, accelerometers, …), allow to control the mouse functions by moving a body part (e.g. the head)</td>
</tr>
<tr>
<td>324</td>
<td>Tools and components for development of software products</td>
<td>attribute</td>
<td>Tools and components for the development of accessible applications and assistive technology software products and services. Included are, for example, authoring tools for the development of accessible user interfaces</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ID</th>
<th>Operating systems</th>
<th>Cluster</th>
</tr>
</thead>
<tbody>
<tr>
<td>316</td>
<td>Windows</td>
<td>attribute</td>
</tr>
<tr>
<td>317</td>
<td>Mac OS</td>
<td>attribute</td>
</tr>
<tr>
<td>318</td>
<td>Linux</td>
<td>attribute</td>
</tr>
<tr>
<td>319</td>
<td>Chrome OS</td>
<td>attribute</td>
</tr>
<tr>
<td>320</td>
<td>iOS</td>
<td>attribute</td>
</tr>
<tr>
<td>321</td>
<td>Android</td>
<td>attribute</td>
</tr>
<tr>
<td>322</td>
<td>Windows mobile/phone</td>
<td>attribute</td>
</tr>
<tr>
<td>323</td>
<td>Symbian</td>
<td>attribute</td>
</tr>
</tbody>
</table>
Annex 2 – Example of partner’s WSDL files

This annex reports, as examples, the wsdl files of three ETNA partners that implemented the web services according to the specifications given above. Two of the partners (ITD-CNR and CERTH) had to implement the web services from scratch (because they were not yet connected to the current EASTIN system) while the other (FDGGO) had to upgrade the existing EASTIN web service.

ITD-CNR wsdl (Essediquadro database)

```xml
<?xml version="1.0" encoding="ISO-8859-1"?>
  <xsd:schema targetNamespace="http://sd2.itd.cnr.it/ws"  
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"  
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">  
  </xsd:schema>  
</definitions>
```
<xsd:complexType name="ETNAProductionDto">
  <xsd:complexContent>
    <xsd:restriction base="SOAP-ENC:Array">
      <xsd:attribute ref="SOAP-ENC:arrayType" use="required" type="xsd:string"/>
    </xsd:restriction>
  </xsd:complexContent>
  <xsd:all>
    <xsd:element name="ProductCode" type="xsd:string"/>
    <xsd:element name="IsoCodePrimary" type="xsd:string"/>
    <xsd:element name="IsoCodesOptional" type="tns:IsoCodesOptional"/>
    <xsd:element name="CommercialName" type="xsd:string"/>
    <xsd:element name="ManufacturerCode" type="xsd:string"/>
    <xsd:element name="ManufacturerOriginalFullName" type="xsd:string"/>
    <xsd:element name="InsertDate" type="xsd:dateTime"/>
    <xsd:element name="LastUpdateDate" type="xsd:dateTime"/>
    <xsd:element name="Issuer" type="xsd:string"/>
    <xsd:element name="Publisher" type="xsd:string"/>
    <xsd:element name="Manufacturers" type="SmallAssociatedInfoDto"/>
    <xsd:element name="Features" type="FeatureDto"/>
  </xsd:all>
</xsd:complexType>

<xsd:complexType name="SmallAssociatedInfoDto">
  <xsd:complexContent>
    <xsd:restriction base="SOAP-ENC:Array">
      <xsd:attribute ref="SOAP-ENC:arrayType" use="required" type="xsd:string"/>
    </xsd:restriction>
  </xsd:complexContent>
  <xsd:all>
    <xsd:element name="AssociatedInfoCode" type="xsd:string"/>
    <xsd:element name="CommercialName" type="xsd:string"/>
    <xsd:element name="Children" type="SmallAssociatedInfoDto"/>
    <xsd:element name="Comment" type="xsd:string"/>
    <xsd:element name="InsertDate" type="xsd:dateTime"/>
    <xsd:element name="LastUpdateDate" type="xsd:dateTime"/>
    <xsd:element name="Language" type="xsd:string"/>
    <xsd:element name="ScopeNote" type="xsd:string"/>
  </xsd:all>
</xsd:complexType>

<xsd:complexType name="FeatureDto">
  <xsd:complexContent>
    <xsd:restriction base="SOAP-ENC:Array">
      <xsd:attribute ref="SOAP-ENC:arrayType" use="required" type="xsd:string"/>
    </xsd:restriction>
  </xsd:complexContent>
  <xsd:all>
    <xsd:element name="FeatureId" type="xsd:string"/>
    <xsd:element name="ValueMin" type="xsd:decimal"/>
    <xsd:element name="ValueMax" type="xsd:decimal"/>
    <xsd:element name="MetricType" type="xsd:string"/>
    <xsd:element name="Units" type="xsd:string"/>
    <xsd:element name="FeatureDescription" type="xsd:string"/>
    <xsd:element name="Icon" type="xsd:string"/>
    <xsd:element name="PortalLink" type="xsd:string"/>
    <xsd:element name="Email" type="xsd:string"/>
    <xsd:element name="Phone" type="xsd:string"/>
    <xsd:element name="DownloadUrl" type="xsd:string"/>
    <xsd:element name="ReleaseDate" type="xsd:dateTime"/>
    <xsd:element name="LastUpdateDate" type="xsd:dateTime"/>
    <xsd:element name="PublishDate" type="xsd:dateTime"/>
    <xsd:element name="Publisher" type="xsd:string"/>
    <xsd:element name="Language" type="xsd:string"/>
    <xsd:element name="ConfigData" type="xsd:string"/>
    <xsd:element name="SortOrder" type="xsd:int"/>
    <xsd:element name="Visible" type="xsd:boolean"/>
    <xsd:element name="ProductCode" type="xsd:string"/>
    <xsd:element name="ProductFamilyCode" type="xsd:string"/>
  </xsd:all>
</xsd:complexType>

<xsd:complexType name="SmallProductDto">
  <xsd:complexContent>
    <xsd:restriction base="SOAP-ENC:Array">
      <xsd:attribute ref="SOAP-ENC:arrayType" use="required" type="xsd:string"/>
    </xsd:restriction>
  </xsd:complexContent>
  <xsd:all>
    <xsd:element name="ProductCode" type="xsd:string"/>
    <xsd:element name="IsoCodePrimary" type="xsd:string"/>
    <xsd:element name="IsoCodesOptional" type="tns:IsoCodesOptional"/>
    <xsd:element name="CommercialName" type="xsd:string"/>
    <xsd:element name="ManufacturerCode" type="xsd:string"/>
    <xsd:element name="ManufacturerOriginalFullName" type="xsd:string"/>
    <xsd:element name="InsertDate" type="xsd:dateTime"/>
    <xsd:element name="LastUpdateDate" type="xsd:dateTime"/>
    <xsd:element name="CommercialName" type="xsd:string"/>
    <xsd:element name="CanonicalTitle" type="xsd:string"/>
    <xsd:element name="PublishDate" type="xsd:dateTime"/>
    <xsd:element name="Publisher" type="xsd:string"/>
    <xsd:element name="Language" type="xsd:string"/>
    <xsd:element name="ConfigData" type="xsd:string"/>
    <xsd:element name="SortOrder" type="xsd:int"/>
    <xsd:element name="Visible" type="xsd:boolean"/>
    <xsd:element name="ProductCode" type="xsd:string"/>
    <xsd:element name="ProductFamilyCode" type="xsd:string"/>
  </xsd:all>
</xsd:complexType>
<xsd:schema>
  <xsd:complexType name="SmallActorDtoArray">
    <xsd:sequence>
      <xsd:element name="SmallActorDto" type="xsd:smallActorDto" maxOccurs="unbounded" />
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="AssociatedInfoDtoArray">
    <xsd:sequence>
      <xsd:element name="AssociatedInfoDto" type="xsd:associatedInfoDto" maxOccurs="unbounded" />
    </xsd:sequence>
  </xsd:complexType>

  <xsd:element name="GetIsoClassProductCountRequest" type="tns:GetIsoClassProductCountRequest" />
  <xsd:element name="GetIsoClassProductCountResponse" type="tns:GetIsoClassProductCountResponse" />
  <xsd:element name="GetAssociatedInfoRequest" type="tns:GetAssociatedInfoRequest" />
  <xsd:element name="GetAssociatedInfoResponse" type="tns:GetAssociatedInfoResponse" />
  <xsd:element name="FindSmallAssociatedInfoRequest" type="tns:FindSmallAssociatedInfoRequest" />
  <xsd:element name="FindSmallAssociatedInfoResponse" type="tns:FindSmallAssociatedInfoResponse" />
</xsd:schema>
<part name="title" type="xsd:string"/>
<part name="insertDateTime" type="xsd:dateTime"/>
<part name="insertDateTimeMax" type="xsd:dateTime"/>
<message name="FindSmallAssociatedInfoResponse">
  <message name="return" type="tst:SmallAssociatedInfoArray"/>
</message>
<message name="FindSmallActorsRequest">
  <part name="actorName" type="xsd:string"/>
  <part name="actorType" type="xsd:token"/>
  <part name="isoCodes" type="tst:IsoCodeArray"/>
  <part name="actCode" type="tst:ActCodeArray"/>
  <part name="insertDateTime" type="xsd:dateTime"/>
  <part name="insertDateTimeMax" type="xsd:dateTime"/>
</message>
<message name="FindSmallActorsResponse">
  <message name="return" type="tst:SmallActorDtoArray"/>
</message>
<message name="FindSmallProductsRequest">
  <part name="isoCodes" type="tst: IsoCodeArray"/>
  <part name="features" type="tst:FeatureDtoArray"/>
  <part name="turnover" type="xsd:double"/>
  <part name="commercialName" type="xsd:string"/>
  <part name="manufacturer" type="xsd:string"/>
  <part name="insertDateTime" type="xsd:dateTime"/>
  <part name="insertDateTimeMax" type="xsd:dateTime"/>
</message>
<message name="FindSmallProductsResponse">
  <message name="return" type="tst:SmallProductDtoArray"/>
</message>
<message name="GetProductRequest">
  <part name="productId" type="xsd:string"/>
  <part name="productCode" type="xsd:string"/>
  <part name="productName" type="xsd:string"/>
  <part name="productDesc" type="xsd:string"/>
  <part name="productCategory" type="xsd:string"/>
  <part name="productPrice" type="xsd:decimal"/>
  <part name="productAttribute" type="xsd:string"/>
  <part name="productSpec" type="xsd:string"/>
  <part name="productStock" type="xsd:integer"/>
  <part name="productImage" type="xsd:string"/>
  <part name="insertDateTime" type="xsd:dateTime"/>
  <part name="insertDateTimeMax" type="xsd:dateTime"/>
</message>
<message name="GetProductResponse">
  <message name="return" type="tst:ProductDto"/>
</message>
<message name="GetActorRequest">
  <part name="actorName" type="xsd:string"/>
  <part name="actorType" type="xsd:token"/>
  <part name="isoCodes" type="tst: IsoCodeArray"/>
  <part name="actCode" type="tst:ActCodeArray"/>
  <part name="insertDateTime" type="xsd:dateTime"/>
  <part name="insertDateTimeMax" type="xsd:dateTime"/>
</message>
<message name="GetActorResponse">
  <message name="return" type="tst:ActorDto"/>
</message>
<message name="JustTestRequest">
  <part name="input" type="xsd:string"/>
</message>
<message name="JustTestResponse">
  <part name="output" type="xsd:string"/>
</message>
<portType name="ETNA-SO2-SearchServicePortType">
  <operation name="GetIsoclassProductCount">
    <input message="tsn:GetIsoclassProductCountRequest"/>
    <output message="tsn:GetIsoclassProductCountResponse"/>
  </operation>
  <operation name="GetAssociatedInfo">
    <input message="tsn:GetAssociatedInfoRequest"/>
    <output message="tsn:GetAssociatedInfoResponse"/>
  </operation>
  <operation name="FindSmallAssociatedInfo">
    <input message="tsn:FindSmallAssociatedInfoRequest"/>
    <output message="tsn:FindSmallAssociatedInfoResponse"/>
  </operation>
  <operation name="FindSmallActors">
    <input message="tsn:FindSmallActorsRequest"/>
    <output message="tsn:FindSmallActorsResponse"/>
  </operation>
  <operation name="FindSmallProducts">
    <input message="tsn:FindSmallProductsRequest"/>
    <output message="tsn:FindSmallProductsResponse"/>
  </operation>
  <operation name="GetProduct">
    <input message="tsn:GetProductRequest"/>
    <output message="tsn:GetProductResponse"/>
  </operation>
  <operation name="GetActor">
    <input message="tsn:GetActorRequest"/>
    <output message="tsn:GetActorResponse"/>
  </operation>
  <operation name="JustTest">
    <input message="tsn:JustTestRequest"/>
    <output message="tsn:JustTestResponse"/>
  </operation>
</portType>
<binding name="ETNA-SO2-SearchServiceBinding" type="tns:ETNA-SO2-SearchServicePortType">
  <soap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http"/>
  <operation name="GetIsoclassProductCount">
    <input>soap:body use="encoded" namespace="http://sd2.istc.cnr.it/ws" encodingStyle="http://schemas.xmlsoap.org/soap/encoding"/>
  </operation>
  <operation name="GetAssociatedInfo">
    <input>soap:body use="encoded" namespace="http://sd2.istc.cnr.it/ws" encodingStyle="http://schemas.xmlsoap.org/soap/encoding"/>
  </operation>
  <operation name="FindSmallAssociatedInfo">
    <input>soap:body use="encoded" namespace="http://sd2.istc.cnr.it/ws" encodingStyle="http://schemas.xmlsoap.org/soap/encoding"/>
  </operation>
  <operation name="FindSmallActors">
    <input>soap:body use="encoded" namespace="http://sd2.istc.cnr.it/ws" encodingStyle="http://schemas.xmlsoap.org/soap/encoding"/>
  </operation>
</binding>
CERTH wsdl (OAEG database)

<?xml version="1.0" encoding="UTF-8"?>
  <wsdl:documentation>
    Please type your service description here.
  </wsdl:documentation>
  <wsdl:types>
    <xsi:schema xmlns:xsi="http://www.w3.org/2001/XMLSchema" targetNamespace="http://services.certh.gr/id/" attributeFormDefault="qualified" elementFormDefault="qualified">
      <xsi:import namespace="http://services.certh.gr/id/"></xsi:import>
      <xsi:element name="GetProduct">
        <xsi:complexType>
          <xsi:sequence>
            <xsi:element minOccurs="0" name="productCode" nillable="true" type="xs:string"/>
          </xsi:sequence>
        </xsi:complexType>
      </xsi:element>
      <xsi:element name="GetProductResponse">
        <xsi:complexType>
          <xsi:sequence>
            <xsi:element minOccurs="0" name="return" nillable="true" type="ax21:ProductDto"/>
          </xsi:sequence>
        </xsi:complexType>
      </xsi:element>
      <xsi:element name="GetIsoClassProductCount">
        <xsi:complexType>
          <xsi:sequence>
            <xsi:element minOccurs="0" name="IsoCode" nillable="true" type="xs:string"/>
          </xsi:sequence>
        </xsi:complexType>
      </xsi:element>
      <xsi:element name="GetIsoClassProductCountResponse">
        <xsi:complexType>
          <xsi:sequence>
            <xsi:element minOccurs="0" name="return" type="xs:int"/>
          </xsi:sequence>
        </xsi:complexType>
      </xsi:element>
      <xsi:element name="GetAssociatedInfo">
        <xsi:complexType>
          <xsi:sequence>
            <xsi:element minOccurs="0" name="InfoType" nillable="true" type="xs:string"/>
          </xsi:sequence>
        </xsi:complexType>
      </xsi:element>
    </xsi:schema>
  </wsdl:types>
  <wsdl:message name="GetProductRequest">
    <wsdl:part name="productCode" element="ax21:productCode" use="required"/>
  </wsdl:message>
  <wsdl:message name="GetProductResponse">
    <wsdl:part name="return" element="ax21:ProductDto" use="optional"/>
  </wsdl:message>
  <wsdl:message name="GetIsoClassProductCountRequest">
    <wsdl:part name="IsoCode" element="ax21: IsoCode" use="required"/>
  </wsdl:message>
  <wsdl:message name="GetIsoClassProductCountResponse">
    <wsdl:part name="return" element="xs:int" use="optional"/>
  </wsdl:message>
  <wsdl:message name="GetAssociatedInfoRequest">
    <wsdl:part name="InfoType" element="xs:string" use="required"/>
  </wsdl:message>
  <wsdl:message name="GetAssociatedInfoResponse">
    <wsdl:part name="InfoType" element="xs:string" use="optional"/>
  </wsdl:message>
  <wsdl:portType name="SearchServicePort">
    <wsdl:operation name="GetProduct">
      <wsdl:input message="ax21:GetProductRequest"/>
      <wsdl:output message="ax21:GetProductResponse"/>
    </wsdl:operation>
    <wsdl:operation name="GetIsoClassProductCount">
      <wsdl:input message="ax21:GetIsoClassProductCountRequest"/>
      <wsdl:output message="ax21:GetIsoClassProductCountResponse"/>
    </wsdl:operation>
    <wsdl:operation name="GetAssociatedInfo">
      <wsdl:input message="ax21:GetAssociatedInfoRequest"/>
      <wsdl:output message="ax21:GetAssociatedInfoResponse"/>
    </wsdl:operation>
  </wsdl:portType>
  <wsdl:binding name="SearchServiceBinding" type="ax21:SearchServicePort">
    <soap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http"/>
    <wsdl:operation name="GetProduct">
      <wsdl:input message="ax21:GetProductRequest"/>
      <wsdl:output message="ax21:GetProductResponse"/>
    </wsdl:operation>
    <wsdl:operation name="GetIsoClassProductCount">
      <wsdl:input message="ax21:GetIsoClassProductCountRequest"/>
      <wsdl:output message="ax21:GetIsoClassProductCountResponse"/>
    </wsdl:operation>
    <wsdl:operation name="GetAssociatedInfo">
      <wsdl:input message="ax21:GetAssociatedInfoRequest"/>
      <wsdl:output message="ax21:GetAssociatedInfoResponse"/>
    </wsdl:operation>
  </wsdl:binding>
  <wsdl:service name="ETNA-SD2-SearchService">
    <wsdl:port name="ETNA-SD2-SearchServicePort" binding="ax21:SearchServiceBinding">
      <soap:address location="http://150.145.2.12/webservice/index.php/GetProduct"/>
    </wsdl:port>
  </wsdl:service>
</wsdl:definitions>
<wsdl:message name="FindSmallActorsResponse">
  <wsdl:part name="parameters" element="ns:FindSmallActorsResponse"/>
</wsdl:message>

<wsdl:element name="FindSmallActorsRequest" type="ns:FindSmallActorsRequest"/>

<xs:complexType name="FindSmallActorsRequest">
  <xs:sequence>
    <xs:element name="actorInfo" type="ns:ActorDto" minOccurs="0" maxOccurs="unbounded" nillable="true"/>
  </xs:sequence>
</xs:complexType>

<wsdl:message name="FindSmallActorsRequest">
  <wsdl:part name="parameters" element="ns:FindSmallActorsRequest"/>
</wsdl:message>

<wsdl:element name="FindSmallActorsResponse" type="ns:FindSmallActorsResponse"/>

<xs:complexType name="FindSmallActorsResponse">
  <xs:sequence>
    <xs:element name="GetActor" type="ns:ActorDto" minOccurs="0" maxOccurs="unbounded" nillable="true"/>
  </xs:sequence>
</xs:complexType>

<wsdl:message name="GetActorRequest">
  <wsdl:part name="parameters" element="ns:GetActorRequest"/>
</wsdl:message>

<wsdl:element name="GetActorResponse" type="ns:GetActorResponse"/>

<xs:complexType name="GetActorResponse">
  <xs:sequence>
    <xs:element name="GetActor" type="ns:ActorDto" minOccurs="0" maxOccurs="unbounded" nillable="true"/>
  </xs:sequence>
</xs:complexType>

<wsdl:message name="GetActorRequest">
  <wsdl:part name="parameters" element="ns:GetActorRequest"/>
</wsdl:message>

<wsdl:element name="GetActorResponse" type="ns:GetActorResponse"/>

<xs:complexType name="GetActorResponse">
  <xs:sequence>
    <xs:element name="GetActor" type="ns:ActorDto" minOccurs="0" maxOccurs="unbounded" nillable="true"/>
  </xs:sequence>
</xs:complexType>
<soap:operation soapAction="urn:GetAssociatedInfo" style="document"/>
  <wsdl:input>
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>

<wsdl:binding name="EtnaService8Soap12Binding" type="ns:EtnaService8PortType">
  <soap12:binding transport="http://schemas.xmlsoap.org/soap/http" style="document"/>
  <soap12:operation soapAction="urn:GetAssociatedInfos" style="document"/>
  <soap12:body use="literal"/>
  <wsdl:output>
    <soap12:body use="literal"/>
  </wsdl:output>
</wsdl:binding>

<soap12:operation soapAction="urn:GetAssociatedInfos" style="document"/>
  <soap12:body use="literal"/>
  <wsdl:output>
    <soap12:body use="literal"/>
  </wsdl:output>
</wsdl:operation>

<soap12:operation soapAction="urn:GetProduct" style="document"/>
  <soap12:body use="literal"/>
  <wsdl:output>
    <soap12:body use="literal"/>
  </wsdl:output>
</wsdl:operation>

<soap12:operation soapAction="urn:GetActor" style="document"/>
  <soap12:body use="literal"/>
  <wsdl:output>
    <soap12:body use="literal"/>
  </wsdl:output>
</wsdl:operation>

<soap12:operation soapAction="urn:GetSmallProducts" style="document"/>
  <soap12:body use="literal"/>
  <wsdl:output>
    <soap12:body use="literal"/>
  </wsdl:output>
</wsdl:operation>

<soap12:operation soapAction="urn:GetIsoClassProductCount" style="document"/>
  <soap12:body use="literal"/>
  <wsdl:output>
    <soap12:body use="literal"/>
  </wsdl:output>
</wsdl:operation>

<soap12:operation soapAction="urn:GetSmallActors" style="document"/>
  <soap12:body use="literal"/>
  <wsdl:output>
    <soap12:body use="literal"/>
  </wsdl:output>
</wsdl:operation>

<soap12:operation soapAction="urn:GetAssociatedActor" style="document"/>
  <soap12:body use="literal"/>
  <wsdl:output>
    <soap12:body use="literal"/>
  </wsdl:output>
</wsdl:operation>

<wsdl:binding name="EtnaService8HttpBinding" type="ns:EtnaService8PortType">
  <http:binding verb="POST"/>
  <wsdl:operation name="FindSmallAssociatedInfos">
    <http:operation location="FindSmallAssociatedInfos"/>
    <wsdl:input>
      <wsdl:part name="parameters" contentType="application/xml"/>
    </wsdl:input>
    <wsdl:output>
      <wsdl:part name="parameters" contentType="application/xml"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetProduct">
    <http:operation location="GetProduct"/>
    <wsdl:input>
      <wsdl:part name="parameters" contentType="application/xml"/>
    </wsdl:input>
    <wsdl:output>
      <wsdl:part name="parameters" contentType="application/xml"/>
    </wsdl:output>
  </wsdl:operation>
</wsdl:binding>
FDCCG wsdl (Portale SIVA database)

```xml
<wSDL:binding>
  <wSDL:service name="EtnaService8"/>
  <wSDL:port name="EtnaService8HttpSoap11Endpoint" binding="ns:EtnaService8Soap11Binding"/>
</wSDL:binding>

<wSDL:operation name="GetActor"><wSDL:input><q1:SoapBody use="literal"/></wSDL:input><wSDL:output/></wSDL:operation>

<wSDL:operation name="FindSmallProducts"><wSDL:input><q1:SoapBody use="literal"/></wSDL:input><wSDL:output/></wSDL:operation>

<wSDL:operation name="GetIsoClassProductCount"><wSDL:input><q1:SoapBody use="literal"/></wSDL:input><wSDL:output/></wSDL:operation>

<wSDL:operation name="FindSmallActors"><wSDL:input><q1:SoapBody use="literal"/></wSDL:input><wSDL:output/></wSDL:operation>

<wSDL:operation name="GetAssociatedInfo"><wSDL:input><q1:SoapBody use="literal"/></wSDL:input><wSDL:output/></wSDL:operation>

<wSDL:operation name="FindSmallProducts"><wSDL:input><q1:SoapBody use="literal"/></wSDL:input><wSDL:output/></wSDL:operation>

<wSDL:operation name="GetIsoClassProductCount"><wSDL:input><q1:SoapBody use="literal"/></wSDL:input><wSDL:output/></wSDL:operation>

<wSDL:operation name="FindSmallActors"><wSDL:input><q1:SoapBody use="literal"/></wSDL:input><wSDL:output/></wSDL:operation>

<wSDL:operation name="GetAssociatedInfo"><wSDL:input><q1:SoapBody use="literal"/></wSDL:input><wSDL:output/></wSDL:operation>
```
<wsdl:definitions name="Etna project"

<wsdl:service name="GetKeywords">
<wsdl:output>
<soap:body use="literal"/>
</wsdl:output>
</wsdl:operation>
</wsdl:definition>

<wsdl:service name="BasicHttpBinding/Searches">
<wsdl:output>
<soap:body use="literal"/>
</wsdl:output>
</wsdl:operation>
</wsdl:definitions>