**Functional Design**

**“Tu sì che mi capisci”**

Versione 24/01/2022 cambiamenti:

Correzione errori punti 1.1, 1.2, 3.2

Aggiunte 4, 4.2.1, 5.2.1, 5.2.1, 5.5.1.1, 5.5.1.2

**Table of content**

[**1**](#_heading=h.sqyw64) **Introduction 4**

[*1.1*](#_heading=h.3cqmetx) *Actual Situation and history 4*

[*1.2*](#_heading=h.1rvwp1q) *Objectives 4*

[**2**](#_heading=h.4bvk7pj) **Scope, premises and restrictions 4**

[**3**](#_heading=h.2r0uhxc) **Conceptual Design 4**

[*3.1*](#_heading=h.1664s55) *General description of the system 4*

[*3.2*](#_heading=h.3q5sasy) *Context and interfaces 4*

[**4**](#_heading=h.25b2l0r) **Design of the architecture 5**

[*4.1*](#_heading=h.kgcv8k) *Requirements related to the architecture 5*

[*4.2*](#_heading=h.34g0dwd) *Scenario and elements of the architecture 5*

[4.2.1](#_heading=h.2jxsxqh) Components of the architecture 5

[4.2.2](#_heading=h.3j2qqm3) Operating systems and hardware architecture 6

[4.2.3](#_heading=h.4i7ojhp) Scenario Diagrams 6

[*4.3*](#_heading=h.1jlao46) *Software architecture 6*

[4.3.1](#_heading=h.1ci93xb) Software components and integration model 6

[4.3.2](#_heading=h.3whwml4) Correspondence software components / system functions 6

[**5**](#_heading=h.43ky6rz) **External Design 7**

[*5.1*](#_heading=h.2iq8gzs) *Layout standards and style guides 7*

[5.1.1](#_heading=h.3as4poj) Screen standards 7

[5.1.2](#_heading=h.1pxezwc) Report and document standards 8

[*5.2*](#_heading=h.xvir7l) *Details of user screen interface 8*

[5.2.1](#_heading=h.2p2csry) Flow chart for navigation of the interface 8

[5.2.2](#_heading=h.147n2zr) Summary of user interfaces – Screens and windows 9

[5.2.3](#_heading=h.23ckvvd) Details of the user interfaces – Screens and windows 9

[5.2.3.1](#_heading=h.3hv69ve) U. I. screen (1...n) 9

[*5.3*](#_heading=h.1x0gk37) *Details of the user interface for reports 10*

[5.3.1](#_heading=h.4h042r0) Report (1 ... n) 10

[*5.4*](#_heading=h.2w5ecyt) *Details of the user interface for documents 10*

[5.4.1](#_heading=h.1baon6m) Document (1 ... n) 10

[*5.5*](#_heading=h.3vac5uf) *Interfaces with other systems / applications 10*

[5.5.1](#_heading=h.2afmg28) System Interface – (1...n) 10

[5.5.1.1](#_heading=h.1v1yuxt) Description of the interface concept 11

[5.5.1.2](#_heading=h.pkwqa1) External system 11

[5.5.1.3](#_heading=h.39kk8xu) Content 11

[**6**](#_heading=h.1opuj5n) **Logical Data Model 12**

[*6.1*](#_heading=h.48pi1tg) *Entity-Relation Diagram 12*

[*6.2*](#_heading=h.2nusc19) *Entity details 13*

[6.2.1](#_heading=h.1302m92) Entity (E - 1..n) 13

[6.2.1.1](#_heading=h.3mzq4wv) Entity description 13

[*6.3*](#_heading=h.2250f4o) *Relations between entities 13*

[*6.4*](#_heading=h.haapch) *Data Dictionary 13*

[**7**](#_heading=h.319y80a) **Logical Process Model 15**

[*7.1*](#_heading=h.1gf8i83) *Details of the subsystems 15*

[7.1.1](#_heading=h.40ew0vw) Layers or system partitions 15

[7.1.1.1](#_heading=h.2fk6b3p) Details of the subsystem 1...n 15

[*7.2*](#_heading=h.upglbi) *Detailed design of basic functions (structured design) 15*

[7.2.1](#_heading=h.3ep43zb) Function AAAAAAA (1...n) 15

1. **Introduction**

This section contains the objectives of the project, the present situation, the history, the scope of the project, as well as all the known restrictions.

* 1. **Actual Situation and history**

Present situation that is planned to solve or to improve: Business Processes and business model of the client, as well as present technological environment.

* 1. **Objectives**

Include in a brief description the objectives of the project, what it is tried to reach, to obtain, to modify, etc.

1. **Scope, premises and restrictions**

Include here the information regarding the:

* Scope of application: organisation units (end user areas, departments of involved systems, etc.) to whom the document is directed and the work to be performed by each one of them.

Con il nostro progetto, inizialmente, cerchiamo di aiutare gli utenti, con dei piccoli gesti, a migliorare il proprio umore. Fino ad arrivare ad aiutare le persone che soffrono di solitudine e che essendo sole nella quotidianità necessitano di assistenza. Inoltre il nostro progetto punta a rendere più sicure le case in quanto, grazie a delle cam che l’utente può posizionare nella sua casa e collegare dalla nostra app. Successivamente quando un estraneo entrerà in casa il proprietario verrà avvisato immediatamente.

* Assumptions: those in that it can create a significant impact on the requirements already approved, and/or in the project in general if the situation changes unexpectedly

Problemi legati alla creazione e l’implementazione delle ia oltre a possibili problemi nell’hosting del server che metterà a disposizione il servizio

Restrictions: conditions imposed on the project and its solution: target dates, technical environment, etc.

Data termina: aprile;

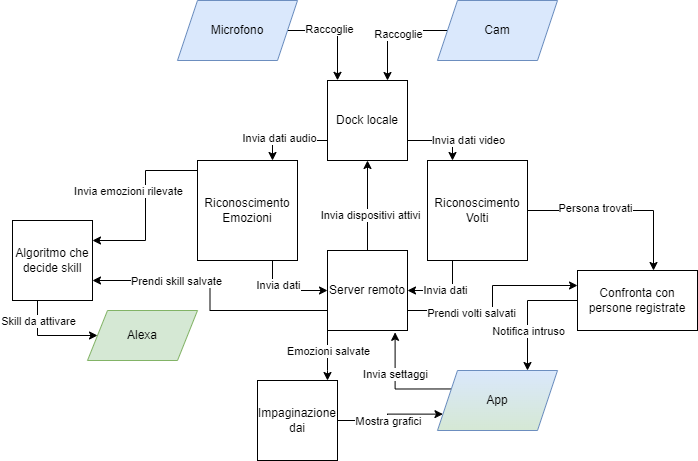
1. **Conceptual Design**

* 1. **General description of the system**

Il nostro software tramite alcuni canali di input audio estrapola le emozioni delle persone che stanno parlando e, è in gradi di attivare skill di alexa che l’utente ritiene consoni per l’umore in cui si trova in quel momento. Inoltre il prostro progetto vanta anche una funzione di riconoscimento facciale per aumentare la sicurezza e garantire al cliente di sapere in tempo reale se una persona si è introdotta in casa sua e una comoda app con la quale l’utente avrà la possibilità di gestire i propri dati e ottenere piccole soddisfazioni nel completare degli obbiettivi giornalieri.

* 1. **Context and interfaces**

Example of a DFD:



1. **Design of the architecture**

Architettura:

1. Server centrale
2. Ia riconoscimento emozioni
3. Ia riconoscimento volti
4. Dock locale
5. Microfono
6. Telecamera

* 1. **Requirements related to the architecture**

Specify these requirements that have an impact on the architecture.

|  |  |
| --- | --- |
| REQUIREMENT | CONSEQUENCES FOR ARCHITECTURE |
| RQF00001 | Functional req.00001  Xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx  Xxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxx |
| RQT00003 |  |
| RQO00006 |  |

* 1. **Scenario and elements of the architecture**

The design of the scenario should not detail the versions or specific configuration themes that may change between the different environments.

* + 1. **Components of the architecture**

Put into relation the elements of the architecture dividing them by the different element types and the functions that they are carrying out.

|  |  |
| --- | --- |
| COMPONENT | FUNCTION TO CARRY OUT |
| Alexa | Mettere a disposizione le skill decise dall’utente e poter interagire con componenti domotici della casa |
| Server remoto | 1. Store dati e metterli a disposizione tramite dei servizi 2. Mettere a disposizione le ia per il riconoscimento dei volti e per il riconoscimento delle emozioni dalla voce |
| App | Interfaccia con l’utente e possibilità di personalizzare il servizio in basse alle proprie necessità |
| Ia riconoscimento facciale | Capire che persona è entrata in casa del flusso video di input |
| Ia riconoscimento vocale | Capire le emozioni dal flusso audio che ha in input |
| Dock locale | Leggere le fonti di input, preprocessing dei dati e invio al server |

* Function to be carried out: what is it thought for within the project.

* + 1. **Operating systems and hardware architecture**

List the hardware and basic software elements of the system.

|  |  |  |  |
| --- | --- | --- | --- |
| MANUFACTURER - MODEL | CARACTERISTICS | CONFIGURATION | OPER. SYSTEM |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

* + 1. **Scenario Diagrams**

It is understood as a scenario the set of elements that form the technological environment necessary to support the requirements of the project: basic software, servers, connections, protocols, etc.

Include in this chapter a diagram of the elements of the architecture on a high level (using Visio or another product).

* 1. **Software architecture**

Include a text as introduction. Example:

“The architecture of the software is based on component types that are specified on the following pages. The specifications about the component types and the form of their integration with the other components in order to build the system functions are described.”

* + 1. **Software components and integration model**

Specify for every used component the following:

* Description (Component type)
* Component type that it uses/invokes and the methods
* Component types from which it is invoked and the used methods

See example:

|  |  |  |
| --- | --- | --- |
| TYPE OF COMPONENT | INVOKES A (TYPES) | IS INVOKED BY (TYPES) |
|  |  |  |
|  |  |  |
|  |  |  |

Add further information about the used component types if necessary.

* + 1. **Correspondence software components / system functions**

Describe in a structured form the components that are needed to create a function of the system. Normally in a system there is more than one way to build functions (transaction, to trigger, etc.) from components.

Example:

* “The presentation/ batch functions … are composed by the component CCCCCCCC and the following components of type TTTTTTTT organised in the following form:”
* “The functions of …”

It may be described also with a scheme, e.g.:



1. **External Design**

Describe the interfaces with the system. This includes those of human interaction (screens, navigator, etc.) as well as the interfaces with external systems.

The description of these interfaces should be maintained abstract and as independent as possible from the used technology. In the case of interfaces with other systems, the specification is centred in the aspects of the intercommunication and their details, including the technologies used for the interface. On the other hand, in the case of a user interface a more detailed description of each interface must be done including the image or graphical design, like the detailed relation of the actions and events of each one.

As an example the following text may be included:

“The external design describes the level of presentation and application layer (model OSI) of each of the interfaces with users and with other systems detailed in the context diagram (to see chapter 2.2).”

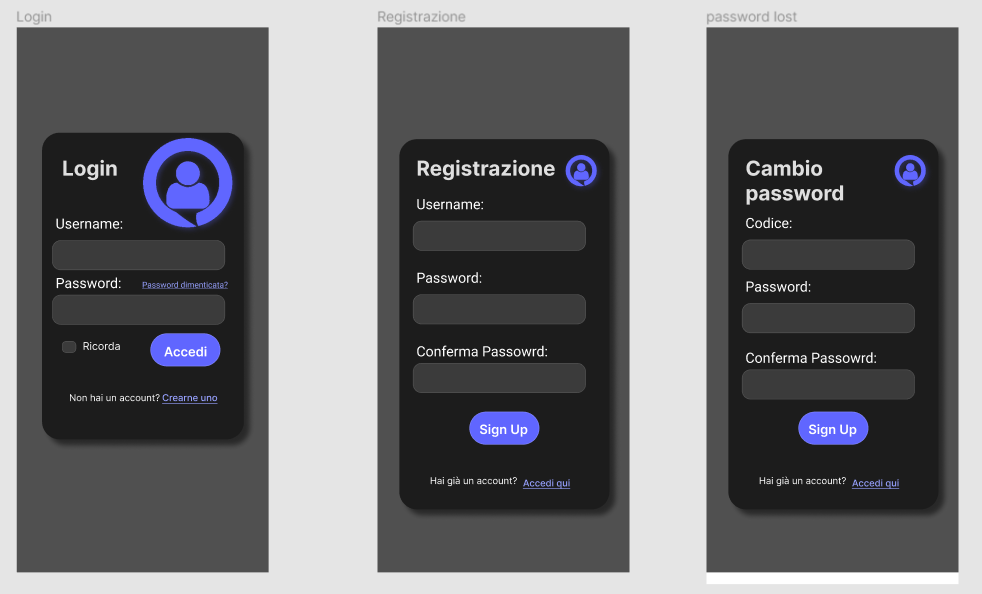
User interfaces are documents, reports and screens, understanding as screen in a wide definition any display device using text and/or graphical elements that allow to communicate with the system.

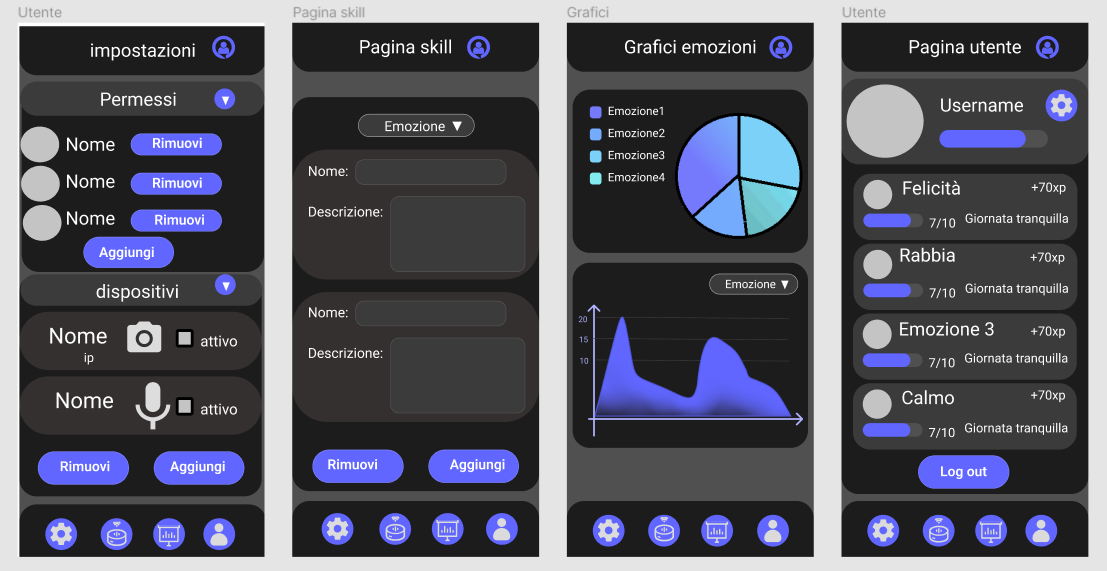
In case of using a tool for the design and/or construction of user Interfaces the part of External Design should be substituted by the information generated by the tool. The documents or objects where the designs are stored must be treated like components of the system with the object stored in repositories and of version control.

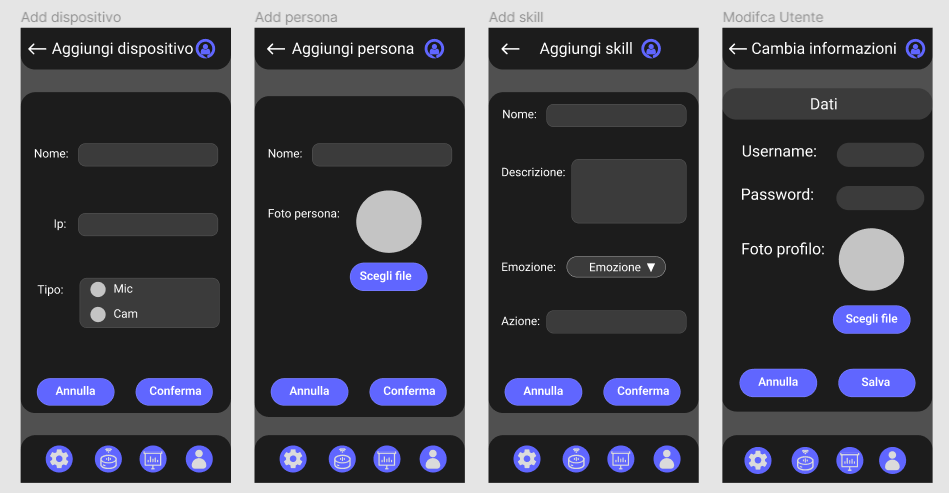
L'utente si interfaccerà con il microfono che ascolterà la sua voce e prelevando le emozioni gli consigliera azioni da fare e l’applicazione da cui potrà personalizzare il software.

* 1. **Layout standards and style guides**

Anteprima grafiche:







* + 1. **Screen standards**

Define standards for screen styles. It may be included for example:

* Headers.
* Logos.
* Typography and colours (fonts and pantones).
* Visual structure and positioning of the presentation elements.
* Main menu bar.
* Positions of the buttons.
* Banners and advertisements.
* Function keys.
* Portlets.
* Languages (multi-language).
* Personalisation.
* Message areas for responses and errors.
* Any other resource or common structure.
* Sizes (in pixels or characters).
* Supported navigators.
* Common elements in all screens.
* Location of the elements.

* + 1. **Report and document standards**

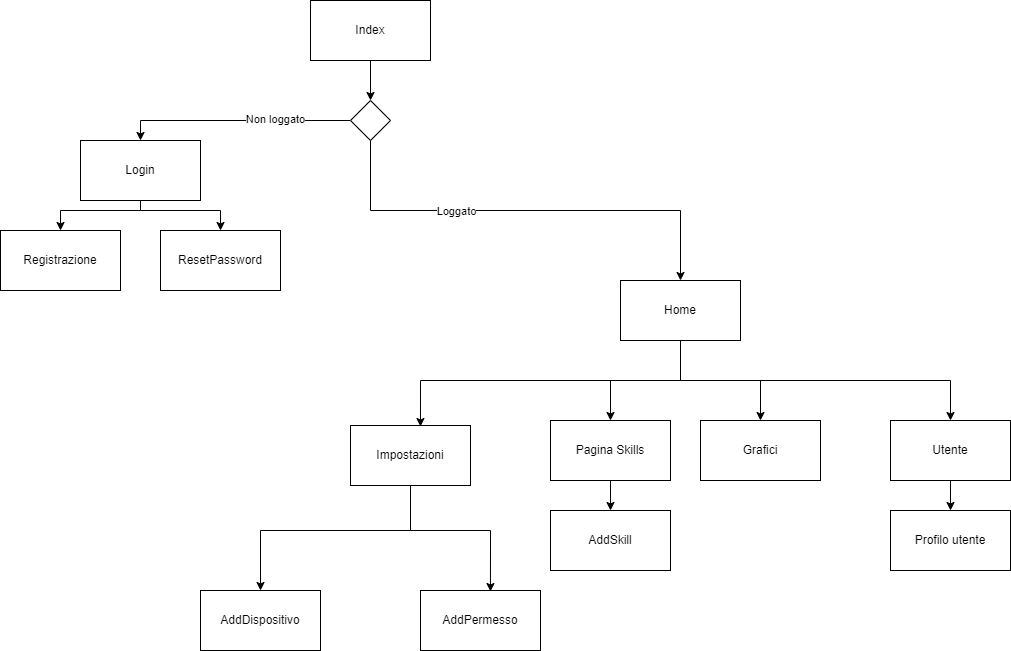
For reports and documents standards should be established for the following aspects:

* Headers.
* Logos.
* Legal elements in commercial documents (commercial register, etc.).
* Typography and colours (fonts and pantones).
* Languages (multi-language).
* Inserted advertisement.
* Sizes (DIN A4, 8,5x11”, etc.)
* Common elements for all documents and/or reports.

* 1. **Details of user screen interface**

* + 1. **Flow chart for navigation of the interface**

Flow chart of the navigation:



* + 1. **Summary of user interfaces – Screens and windows**

Describe all the screens and windows of the system.

|  |  |  |
| --- | --- | --- |
| ID. SCREEN OR WINDOW | DESCRIPTION | SUPPORTED FUNCTION |
| index | Prima pagina | Indirizza l’utente alla pagina “Home” nel caso in cui sia già loggato altrimenti lo reindirizza alla pagina “Login” per effettuare l’accesso al suo profilo |
| Login | Pagina di login | Permette all’utente di accedere al suo profilo tramite l’inserimento di username e password |
| Registrazione | Pagina di registrazione | Permette all’utente di registrare un nuovo account tramite l’inserimento di username e password |
| Reset Password | Pagina che permette il reset della password | Permette all’utente di resettare la password dimenticata grazie a un codice di ripristino |
| Home | Pagina principale | Permette la navigazione tra le pagine |
| Skill | Pagina Skills: mostra le skill inserite con le relative caratteristiche | Aggiunta e rimozione delle skill e filtraggio per emozione |
| Add skill | Pagina che permette l’inserimento di una nuova skill | Permette l’inserimento di una nuova skill inserendo le relative caratteristiche. Permette anche la rimozione di una skill |
| Impostazioni | Pagina che visualizza i dispositivi già inseriti con le relative caratteristiche e i permessi degli utenti | Permette di aggiungere e rimuovere i permessi degli utenti, permette di aggiungere e rimuovere un dispositivo |
| Add dispositivo | Pagina per aggiungere un nuovo dispositivo | Permette l’inserimento di un nuovo dispositivo con le relative caratteristiche. Permette di eliminare le informazioni inserite annullando l’inserimento |
| Grafici | Pagina che visualizza l’andamento di un’emozione specificata | In base a un’emozione specifica visualizza l’andamento di questa nel corso dei giorni |
| Utente | Pagina che mostra l’username, l’immagine profilo, gli obiettivi e l’andamento di ognuno di essi | Mostra gli obiettivi di un utente e l’andamento di essi e permette di accedere alle informazioni personali dell’utente |
| Cambia informazioni | Pagina che visualizza e permette di modificare le informazioni dell’utente | Permette di cambiare le informazioni dell’utente (username, password, foto profilo). Permette di annullare le informazioni inserite |
| AddPersona | Pagina che permette di aggiungere una persona che può entrare | Permette l’aggiunta di una nuova persona che può entrare inserendo nome e foto. Permette di annullare le informazioni inserite |

* + 1. **Details of the user interfaces – Screens and windows**

* + - 1. **U. I. screen (1...n)**

Specify the aspect and characteristics of each screen. Include so many sections as screens exist. It will not be necessary to include this section supposing that a tool is used. Include aspects such as:

* Identification of the screen or window.
* Description of them.
* Which components interact between the system and the user in this interface.
* Existing restrictions due to requirements of the user interface.
* Graphic layout of the user interface. Include a design sample or a graphic draft.
* Data fields. Depending on the technology and the special case should be defined for every screen:
  + Position.
  + Data source.
  + Input / Output.
  + Data type from the data dictionary.
  + Special Attributes: ¿Is translatable? (text multi-language), ¿Is convertible? (Numeric data in different types of units or currencies), etc.
  + Validation rules of the screen or coming from the data dictionary.
  + Previewed error cases, messages, actions (applied to input fields).
  + Default values (applied only to input fields).
  + Actions depending on field events: get-focus/lost focus, function keys, etc.
  + Pull down for context help (input fields).
* Other input/output elements. Depending on the technology and the special case should be defined for every element:
  + Position.
  + Validation rules of the screen or coming from the data dictionary.
  + Previewed errors, messages, actions.
  + Default values (input fields).
  + Actions depending on event cases of the element.
* Input/ output with peripheral equipment:
  + Bar code reader.
  + Scanner.
  + Cash Dispenser.
* Actions and transitions (function keys, buttons, menu options, links, etc). Describe each action that can take place from the screen. The description of each transition must include the action of user who qualifies it (PFs, button, Link) and the navigation destination or process that it initiates. The common transitions to all the interfaces of the system should not be described in a detailed form. This will be done in the section of Screen Standards.
* Local Personalización. The capacity of the interface to act in a different form according to profiles, groups, rolls, or even at individual level must be explained.

* 1. **Details of the user interface for reports**
     1. **Report (1 ... n)**

Specify the aspect and characteristics of each report. Include so many sections as reports exist. It will not be necessary to include this section supposing that a tool is used. In opposite case include aspects such as:

* Report description.
* Model format (template/layout).
* Header data and structure.
* Data of every column of the report.
* Report structure.
* Groups: How to group and sort the report.
* Totals: sub-totals and totals
* Print-formats: print preview, AFP, HP, POSTCRIPT, PDF, etc.
* Export formats: MS Word, MS Excel, HTML, XML, etc.

* 1. **Details of the user interface for documents**
     1. **Document (1 ... n)**

Specify the aspect and characteristics of each document. Include so many sections as documents exist. It will not be necessary to include this section supposing that a tool is used. In opposite case include aspects such as:

* Description.
* Model format (template/layout).
* Document data.
* Print formats: print preview, AFP, HP, POSTCRIPT, PDF, etc.
* Export formats: MS Word, MS Excel, HTML, XML, etc.
* Supported peripheral units (hardware).

* 1. **Interfaces with other systems / applications**

Per ora non siamo ancora a conoscenza della tecnologia che usarà il nostro progetto ma dobbiamo interfacciarci con Alexa.

* + 1. **System Interface – (1...n)**

The description of the interfaces with other systems is centred in the details of logical content, not in the solution of technological connectivity that is object of the architecture.

* + - 1. **Description of the interface concept**

1. Interfaccia di output con alexa (API self service) che permette di leggere le skill presenti e farle partire
2. (Web Api) Interfaccia interna che serve per la gestione (prendere, depositare, cambiare e cancellare) dei dati riguardanti dispositivi, emozioni, skill e informazioni dell’utente
3. (Service udp)Interfaccia interna che serve serve per l’inserimento dei dati di input provenienti dalla dock locale per andare successivamente alle ia per l’elaborazione
4. (Librerie di terze parti) Interfacce di input per la gestione di canali audio e video.

* + - 1. **External system**

1. The API self service permette di interfacciarsi con il sistema esterno di Alexa.
   * + 1. **Content**

Explain the elements of the interface. Detail more or less the object methods, API parameters, protocols, etc. depending on the technology If these elements are detailed sufficiently, they should be presented within a table.

Input:

server udp formato dati json per l'inserimento dei dati presi dai canali di input che andranno nelle ia per l'elaborazione

formato:

{

Tipo:"audio/video",

KeyUtente:” e8de33ae-0964-48b2-affc-307c74c2d481”(esempio formato key),

Dati:"dati presi dal dispositivo (vettore dello spettrogramma per audio e punti del viso per le cam)"

}

Web api:

pagine:

-=opzionale

1)  sito\login:Post return key

●   Parametri:

○   username= nome della persona

○   Password= password dell’account

●   Utilizzo:

Effettuare la login per accedere alle altre pagine restituisce un cookie

2)  sito\register:Post

●   Parametri:

○   username= nome della persona

○   password= password dell’account

○   password2= password dell’account

○   mail= mail account

●   Utilizzo:

Registrare un utente

3)  sito\resetPassword:Get

●   Parametri:

○   mail=mail a cui mandare il recupero password

●   Utilizzo:

richiedere reset password

4)  sito\cambiaPassword:PUT

●   Parametri:

○   codice=codice ricevuto per mail

○   password = nuova password dell’account

○   password2= nuova password dell’account

●   Utilizzo:

Reset password

5)  sito\key\utente\cambiaInfo  PUT

●   Parametri:      //solo quelli che cambiano

○   Username-= nome della persona

○   password-= password dell’account

○   mail-= mail account

○   immagine-= file .png che identifica l’immagine profilo

●   Utilizzo:

Serve a cambiare le info della persona

6)  sito\key\utente:GET

Utilizzo:

Restituisce le info dell’utente registrato

7)  sito\key\emozioni\add: POST

●   Parameri:

○   tipo= tipo di emozione rilevata

○   dataRilevazione-= data di quando è stata trovata

○   IdDipositivo= dispositivo che ha rilevato l’emozione

●   Utilizzo:

Serve a Inserire una nuova emozione e solo le ia avranno il

permesso di usare questa chiamata

8)  sito\key\emozioni:GET

●   Parametri:

○   Start-= int che indica da che emozione iniziare a leggere

○   Numero-= int che indica quante emozioni prendere

○   Tipo-= tipo di emozione da prelevare

○   Data-= data da cui iniziare a leggere le emozioni

●   Utilizzo:

Serve a prelevare le emozioni salvate relative alla persona che ha

eseguito la login

9)  sito\key\voltoTrovato\add:POST

●   Parametri:

○   immagine= immagine rilevata

○   dataRilevazione-= data in cui si è rilevato il volto

○   idDispositivo= id del dispositivo che ha rilevato quel volto

○   idVolto-= id del volto registrato (ia ha riconosciuto la persona)

●   Utilizzo:

Serve a Inserire un nuovo volto trovato e solo le ia avranno il

permesso di usare questa chiamata

10) sito\key\voltoTrovato:GET

●   Parametri:

○   Start-= int che indica da che volto iniziare a leggere

○   Numero-= int che indica quanti volti prendere

○   Data-= data da cui iniziare a leggere i volti

●   Utilizzo:

Serve a prelevare i volti salvate relative alla persona che ha

eseguito la login

11) sito\key\voltoRegistrato\add:POST

●   Parametri:

○   immagine= immagine inserita

○   Nome= Nome volto

●   Utilizzo:

Serve a Inserire un nuovo volto trovato e solo le ia avranno il

permesso di usare questa chiamata

12) sito\key\voltoRegistrato:GET

●   Parametri:

○   Start-= int che indica da che volto iniziare a leggere

○   numero-= int che indica quanti volti prendere

○   data-= data da cui iniziare a leggere i volti

●   Utilizzo:

Serve a prelevare i volti salvate relative alla persona che ha

eseguito la login

13) sito\key\voltoRegistrato\remove\id:Delete

14) sito\key\dispositivi\add:POST

●   Parametri:

○   nome= nome del dispositivo

○   tipo= se cam o microfono

○   ip= ip del dispositivo

○   acceso-= se il dispositivo sta lavorando, dafault true

●   Utilizzo:

Serve a Inserire un nuovo dispositivo all’account loggato

15) sito\key\dispositivi:GET

●   Parametri:

○   Start-= int che indica da che dispositivo iniziare a leggere

○   Numero-= int che indica quante dispositivo prendere

○   Tipo-= il tipo di dispositivo da legegre

●   Utilizzo:

Serve a prelevare i dispositivi  salvate relative alla persona che ha

eseguito la login

16) sito\key\dispositivi\remove\id:Delete

●   Utilizzo:

Serve a rimuovere un dispositivo

17) sito\key\skill\add:POST

●   Parametri:

○   nome= nome della skill

○   descrizione= descrizione della skill

○   azione= stringa che identifica il comando per attivare la skill

○   idEmozione= Emozione a cui legare la skill

●   Utilizzo:

Serve a Inserire una nuova skill all’account loggato

18) sito\key\skill:GET

●   Parametri:

○   Start-= int che indica da che skill iniziare a leggere

○   Numero-= int che indica quante skill prendere

●   Utilizzo:

Serve a prelevare le skill salvate relative alla persona che ha

eseguito la login

19) sito\key\skill\remove\id:Delete

20) sito\key\obiettivi\add:POST

●   Parametri:

○   Concluso-= indica se l’obiettivo è completato

○   idObiettivo= idObiettivo da assegnare

●   Utilizzo:

Serve ad assegnare l’obiettivo all’account loggato

21) sito\key\obiettivi:GET

●   Parametri:

○   Start-= int che indica da che obiettivo iniziare a leggere

○   Numero-= int che indica quanti obiettivi prendere

●   Utilizzo:

Serve a prelevare gli obiettivi salvate relative alla persona che ha

eseguito la login

22) sito\key\obiettivi\change\id:PUT

●   Parametri:

○   Concluso= indica se l’obiettivo è completato

●   Utilizzo:

Serve ad assegnare l’obiettivo all’account loggato

23) sito\key\obiettivi\remove\id:Delete

1. **Logical Data Model**

Analyze in detail the system data, generating a Entity/Relationship diagram, and the description of each one of the entities with its main attributes, the relations between organizations and the data dictionary. It should be done following the normal forms, recommending to reach third normal form.

In case of using a data design tool, replace the following sections by the information generated from the tool. The documents or objects where the designs are stored must be treated like system components. They are object of storage in repositorios and of version control.

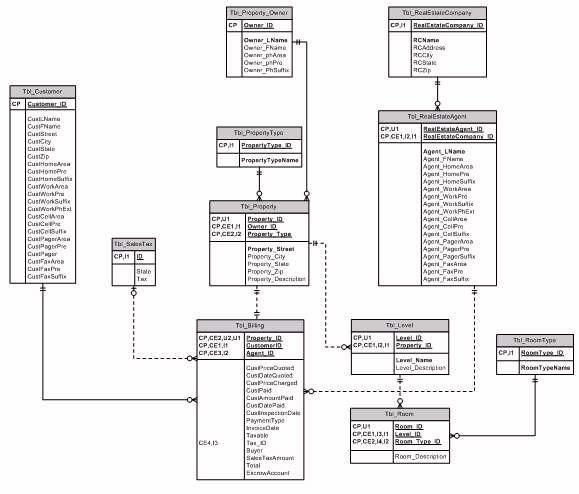
Non abbiamo ancora stabilito questi dettagli del progetto.

* 1. **Entity-Relation Diagram**

Include a graphic presentation of the Data Model observing:

* Data entities.
* Relations between the entities.
* Numbering.

This diagram may be done with the help of a data modelling tool or a painting tool. Example:



* 1. **Entity details**
     1. **Entity (E - 1..n)**
        1. **Entity description**

Describe the different entities of the system.

* + - * 1. Column details

|  |  |  |
| --- | --- | --- |
| Description of the column | Class of data type from data dictionary | Keys and other attributes |
|  |  |  |
|  |  |  |
|  |  |  |

* Column description: name of the column.
* Class of data type in the dictionary: data name in the data dictionary (name, site, DNI, etc.) Do not indicate information type N8, Char (14), etc. because this type of information is included in the Data dictionary from the Technical Design.
* Keys and other attributes: indicate if the column is part of any primary, secondary or external key.

* + - * 1. Actions by entity events

Describe the defined actions as processes of the technical design that are released by events of an entity.

|  |  |
| --- | --- |
| EVENT AND MOMENT OF THE ACTION | DESCRIPTION OF THE ACTION TO DO |
| (Event: new, delete, modification, …)  (moment: pre or post) |  |
|  |  |
|  |  |

* 1. **Relations between entities**

Describe the necessary information for every relation.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| RELATION NMAE | PRIMARY ENTITY | DIRECT ACTION | SECONDARY ENTITY | INVERSE ACTION | CARDINALITY |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

* Relation name: Id. of the relation.
* Primary entity: name of entity 1.
* Secondary entity: name of entity 2.

* 1. **Data Dictionary**

This is the catalogue of the logical data types for the system. Every column of an entity of the data model must be assigned to one data type from the dictionary. The physical data type configuration (N10, char(14), etc.) is explained in the Technical Design.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| DATA TYPE | ALIAS OF THE DATA TYPE | DESCRIPTION | VALUES AND RANGES | RULES | ORIGINAL TYPE |
| Date | Date | General dates | 01.01.1900 a 31.12.2100 |  |  |
| Expiry date | Expiry |  |  |  | Date |
| CCC |  | Account code cl. |  | CCC |  |

* Data type: data name in the dictionary. Must be unique. E.g. a data type “DNI” may exist that may be included as column in different entities.
* Alias of the data type: other names indicated as alias.
* Description: description of the data types.
* Values and ranges: notation, cases, value ranges, default values, etc.
* Rules: validation and integrity rules.
* Original type: If necessary, name of other “data type” (column 1 of the table) from the dictionary from which type gets its basic properties.

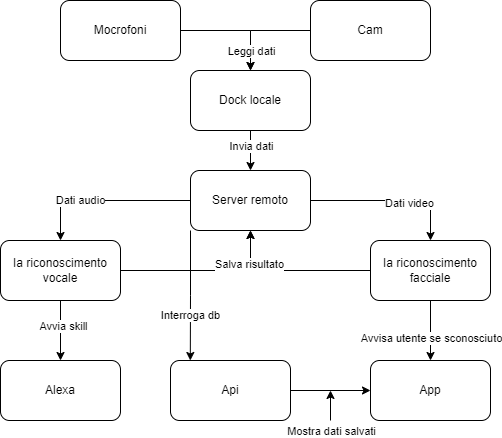
1. **Logical Process Model**

Beginning with the system defined in Conceptual Design and the Requirements of the Client, the Logical Process Model descends to a greater detail level, until a very exact description of the basic system functions, their relations with the data, and the exchanges with the external system interfaces.

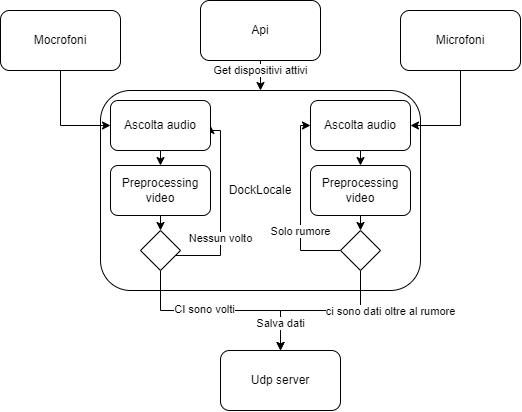
* 1. **Details of the subsystems**
     1. **Layers or system partitions**

Depending on the technology, the project type and the clients preferences this Logical Process Model (process of decomposition in elementary functions) may be done with an object oriented design. In this case it is normal to use UML (use case diagrams, collaboration diagrams, etc.), Flow charts (DFDs) or other technologies.

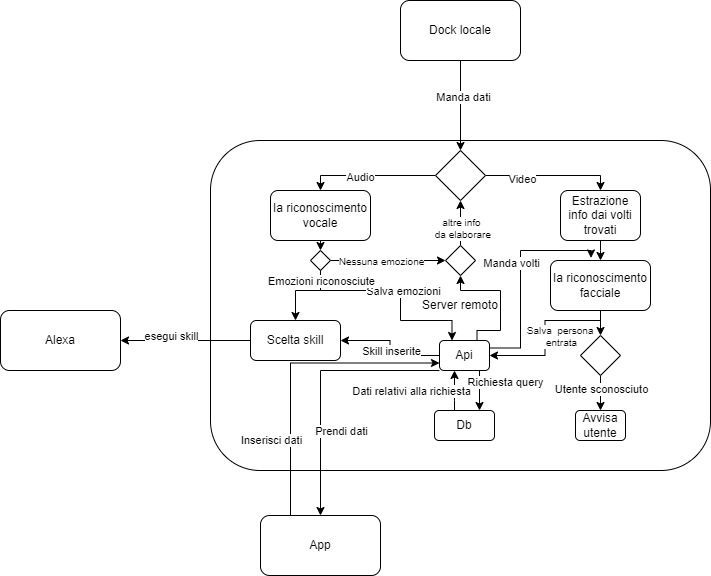
So many sections of subsystems will be created as it is necessary.



* + - 1. **Dock locale**

****

* + - 1. **Server remoto**

****

**Manca Componente app**

* 1. **Detailed design of basic functions (structured design)**
     1. **Function AAAAAAA (1...n)**

Describe:

* Storage of data related to the function
* Related business requirements
* Related external interfaces (users/external systems)
* Event or starting method
* …

Funzioni:

* Aggiungere e rimuovere I dispositivi

sottoAttività:

* + App manda al le informazioni del dispositivo da aggiungere o rimuovere
  + Il server lo elimina o lo aggiunge ai dispositivi attivi

memorizza una nuova instanza nella tabello dispositivi

interface usate:

* + Api web

start evento:

* + utente sceglie l’opzione aggiungi nella pagina relativa ai dispositivi

* Prelevare i dati dalle fonti di input

sottoAttività:

* + Dock locale prende i dispositive attivi dal server
  + La dock raccoglie I dati dagli input per ogni fonte
  + Predispone i dati raccolti per essere salvati
  + Invia i dati al server che li salva e successivamente li metterà a disposizione dell’utente

non memorizza nulla

interface usate:

* + Udp server
  + Librerie di terze parti
* Riconoscere le emozioni

sottoAttività:

* + Manda le emozioni trovate al server

memorizza una nuova instanza nella tabello emozione

interface usate:

* + Api web
  + Udp server

start evento:

* + Server udp riceve dei dati audio

* Riconoscere I volti

sottoAttività:

* + Elabora e manda I volti che ha trovato al server

memorizza una nuova instanza nella tabella VoltoRilevato

interface usate:

* + Api web
  + Udp server

start evento:

* + Server udp riceve dei dati video
* Visualizzazione grafici

sottoAttività:

* + Utente richiede I dati
  + Server manda I dati
  + App li elabora e mostra grafici

interface usate:

* + Api web

start evento:

* + utente sceglie l’opzione di vedere le emozioni

* Gestione Obiettivi giornalieri

sottoAttività:

* + Fornisce obiettivi
  + In base ai dati ricevuti controlla se hai raggiunto gli obiettivi

memorizza una nuova instanza nella tabello dispositivi

interface usate:

* + Api web

start evento:

* + utente fa l’accesso nell’app

* Start skill alexa

sottoAttività:

* + Prendere emozioni salvate
  + Attivare le skill di alexa in base allo stato d’animo

start evento:

* + il sistema rileva una emozione

Flow chart delle funzioni da eseguire dagli input fino ad arrivare a mandare le informazioni all’api che li memorizzerà:

