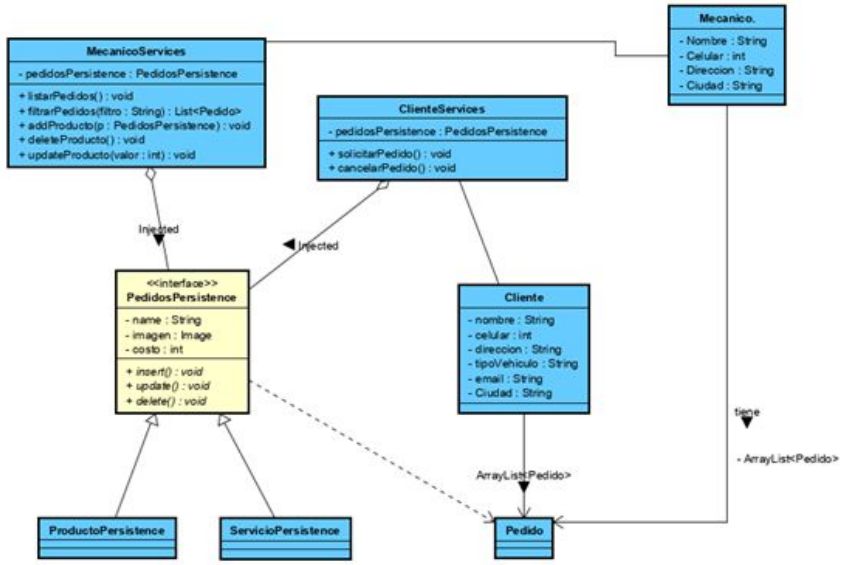
TABLE OF CONTENTS

1 Introduction

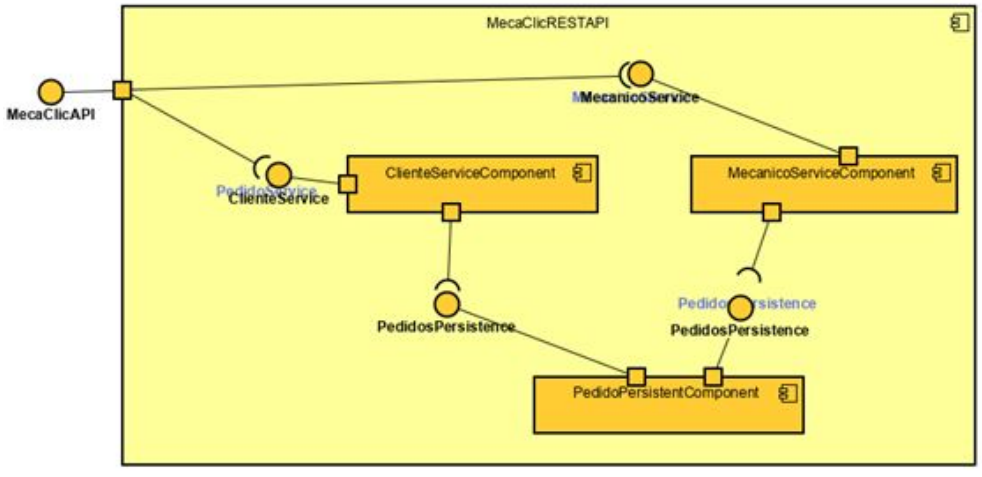
1.1 Document overview

This document describes the architecture of the MecaClick system.
It describes:

- A general description of the system:
  The objective of the application is to connect automotive service providers and their customers so that customers can consume their products and services from the comfort of their homes. The application will allow you to see all the providers and the services and products they offer. The application will allow that when a service is requested, the supplier will be chatted with to finalize details or to clarify any doubts that the customer may have.
- The logical architecture of software, the layers and top-level components



- The physical architecture of the hardware on which runs the software



- The justification of technical choices made
- The traceability between the architecture and the system requirements.

1.2 Abbreviations and Glossary

1.2.1 Abbreviations

Add here abbreviations

1.2.2 Glossary

Add here words definitions

1.3 References

1.3.1 Project References

#	Document Identifier	Document Title
[R1]	ID	Add your documents references. One line per document

1.3.2 Standard and regulatory References

#	Document Identifier	Document Title
[STD1]		Add your documents references. One line per document

1.4 Conventions

Add here conventions
For example for diagrams.

2 Architecture

2.1 Architecture overview

Give a general description of the system, from the point of view of the user :

- In what environment it works (home, near patient bed, operating room …)
- Who the users are
- What it is for,
- The main functions,
- The main interfaces, inputs and outputs.

2.2 Logical architecture overview

[SEI: Documenting software architectures: Chapter 2 (2.4, 2.5)

Describe the top level software components and their interactions/relationships.
Use UML package diagrams and/or la y eructos diagrams and/or components diagrams.
Describe also the operating systems on which the software runs.

2.2.1 Software Component 1 description

Describe the content of each top level software component in the architecture
Optional , you may not do it for 2 rationale :
1. either your software is not class C according to IEC 62366
2. or you describe each top level component in a SDD.

The description shall contain :

- **Its identification**
- The purpose of the component,
- Its interfaces with other components,
- Its network interfaces,
- The hardware resources it uses, for example : average RAM usage, peak RAM usage and peak frequency and duration, disk space for permanent data, disk space for cache data, average CPU usage, peak CPU usage and peak frequency and duration …

2.2.2 Software Component 2 description

Repeat the pattern for each top level component.

2.2.3 Software Component 3 description

Repeat the pattern for each top level component.

2.3 Physical architecture overview

Describe the hardware components on which software runs and their interactions/relationships
Use components diagrams, deployment diagrams, network diagrams, interface diagrams

[SEI: Documenting software architectures: Chapter 5 (5.2)

2.3.1 Hardware Component 1 description

Describe the content of each hardware component in the architecture
Optional , you may not do it if your software is not class C according to IEC 62366
The description shall contain :

- Its identification
- The purpose of the component
- The software component it receives
- Its technical characteristics : type of machine, CPU, RAM, disk and so on.
- Its network hardware interfaces

2.3.2 Hardware Component 2 description

Repeat the pattern for each top level component.

2.3.3 Hardware Component 3 description

Repeat the pattern for each top level component.

2.4 Software COTS

If you use COTS (Components Off the Shelf, also named SOUP, Software Of Unknown Provenance), list them here.
For each COTS, describe :

- Its identification and version
- Its purpose
- Where it comes from : manufacturer, vendor, university …
- Wether it is maintained by a third party or not
- If this is an executable,
  - what are the hardware / sotfware resources it uses
  - Wether it is insulated in the architecture and why
- Its interfaces and data flows

Note : have a look at FDA Guidance « Off-The-Shelf Software Use in Medical Devices » to determine if you need specific or special documentation for your COTS.

3 Dynamic behaviour of architecture

The architecture was designed to answer to functional requirements.
For each main function of the system, add a description of the sequences / data flow that occur.
Use sequence diagrams, collaboration diagrams

[SEI: Documenting software architectures: Chapter 8 (8.2) : Sequence Diagrams, ActivityDiagrams (for event-based behavior).

3.1 Workflow / Sequence 1

Describe here the workflow / sequence of a main function
For example, the user queries data, what happens, from his terminal to the database.

3.2 Workflow / Sequence 2

Repeat the patern for each main function of the system

4 Justification of architecture

4.1 System architecture capabilities

Describe here the rationale of the hardware / software architecture in terms of capabilities :

- Performances (for example response time, user mobility, data storage, or any functional performance which has an impact on architecture)
- User / patient safety (see §4.3 and §4.4)
- Protection against misuse (see 4.4)
- Maintenance (cold maintenance or hot maintenance),
- Adaptability, flexibility
- Scalability, availability
- Backup and restore
- Hardware and Software security : fault tolerance, redondancy, emergency stop, recovery after crash …
- Administration,
- Monitoring, audit
- Internationalization

4.2 Network architecture capabilities

If the medical device uses/has a network, describe here the rationale of the hardware / network architecture :

- Bandwidth
- Network failures
- Loss of data
- Inconsistent data
- Insconsistent timing of data
- Cyber security (see FDA Guidance on Cyber Security of networked medical devices)

4.3 Risk analysis outputs

If the results of risk analysis have an impact on the architecture, describe here for each risk analysis output what has been done to mitigate the risk in the architecture.
Use diagrams if necessary, like architecture before risk mitigation and architecture after risk mitigation, to explain the choices.

4.4 Human factors engineering outputs

If the results of human factors analysis have an impact on the architecture, describe here for each risk human factors output what has been done to mitigate the risk in the architecture.

5 Requirements traceability

Add a table with traceability of components of this document with functional requirements.

Requirement	Component	Comment
REQ-001 The device shall do foo	COMPO-001: foo maker	COMP-001 does foo. COMP-002 also does verification of foo.

This may be a difficult job. A high level function is usually handled by many components. In this case, quote only the component(s) which has(have) the major role.