



PowerEnJoy  
Software Engineering II

## Design Document

*Giovanni Scotti, Marco Trabucchi*

Document version: 1  
November 23, 2016

# Contents

<b>Contents</b>	<b>1</b>
<b>1 Introduction</b>	<b>2</b>
1.1 Purpose . . . . .	2
1.2 Scope . . . . .	2
1.3 Definitions, Acronyms, Abbreviations . . . . .	3
1.4 Reference Documents . . . . .	3
1.5 Document Structure . . . . .	3
<b>2 Architectural Design</b>	<b>5</b>
2.1 Overview . . . . .	5
<b>A Appendix</b>	<b>6</b>
A.1 Software and tools used . . . . .	6
A.2 Hours of work . . . . .	6
<b>Bibliography</b>	<b>7</b>

# Section 1

## Introduction

### 1.1 Purpose

The Design Document is intended to provide a deeper functional description of the *PowerEnJoy* system-to-be by giving technical details and describing the main architectural components as well as their interfaces and their interactions. The relations among the different modules are pointed out using UML standards and other useful diagrams showing the structure of the system.

The document aims to guide the software development team to the architecture of the project providing a stable reference and a single vision of all parts of the software itself and clearly defining how they work.

### 1.2 Scope

The system aims to support a car-sharing service that exclusively employs electric cars.

The system is structured in a four-layered fashion, which will be thoroughly described in this document, that adapts to several forms of clients: various types of actors that interact with the system-to-be by generating a client-server dualism, hence a flow of requests-responses.

The architecture must be designed with the intent of being maintainable and extensible, also foreseeing future changes.

This document aims to drive the implementation phase so that cohesion and decoupling are increased as much as possible. In order to do so, individual components must not include too many unrelated functionalities and reduce interdependency between one another.

Specific architectural styles and design patterns will be followed in this document and used for future implementation, as well as common design paradigms that combine useful features of said concepts.

## 1.3 Definitions, Acronyms, Abbreviations

**DD:** Design Document

**RASD:** Requirements Analysis and Specification Document

## 1.4 Reference Documents

This document follows the guidelines provided by ISO/IEC/IEEE 1016:2009 [3] related to system desing and software design descriptions for complex software systems.

The indications provided in this document are also based on the ones stated in the previous deliverable for the project, the RASD document [1].

Moreover it is strictly based on the specifications concerning the RASD assignment [2] for the Software Engineering II project, part of the course held by professors Luca Mottola and Elisabetta Di Nitto at the Politecnico di Milano, A.Y. 2016/17.

## 1.5 Document Structure

This document consists of five sections:

**Section 1: Introduction.** This section provides a general introduction and overview of the Design Document and the covered topics not previously taken into account by the RASD [1].

**Section 2: Architectural Design.** It shows the main system components together with sub-components and their relationship. This section is divided into different parts whose focus is mainly on design choices, interactions, architectural styles and patterns.

**Section 3: Algorithm Design.** This section provides a high-level description and details about some of the most crucial and critical algorithms to be implemented by the system-to-be.

**Section 4: User Interface Design.** It provides an overview on how the user interface will look like and behave giving further information with respect to those contained in the RASD [1].

**Section 5: Requirements Traceability.** This section describes how the requirements defined in the RASD [1] are mapped to the design elements defined in this document.

At the end of the document are an **Appendix** and a **Bibliography**, providing additional information about the sections listed above.

## Section 2

# Architectural Design

### 2.1 Overview

In this section is a detailed view of the physical and logical infrastructure of the system-to-be, as well as the description of the main components and their interactions.

A top down approach will be adopted for the description of components:

**Section ??** A description of high level components and their interactions.

**Section ??** A detailed insight of the components described in the previous section.

**Section ??** A set of indications on how to deploy the illustrated components on physical tiers.

**Section ??** A thorough description of the dynamic behaviour of the software, complete with diagrams for the key functionalities.

**Section ??** A description of the different type of interfaces among the various described components.

**Section ??** A list of the architectural styles, design patterns and paradigms adopted in the design phase.

**Section ??** A list of all other relevant design decisions not mentioned before.

# Appendix A

## Appendix

### A.1 Software and tools used

- L<sup>A</sup>T<sub>E</sub>X, used as typesetting system to build this document.
- draw.io - <https://www.draw.io> - used to draw diagrams and mockups.
- GitHub - <https://github.com> - used to manage the different versions of the document and to make the distributed work much easier.
- GitHub Desktop, the GitHub official application that offers a seamless way to contribute to projects.

### A.2 Hours of work

The absolute major part of the document was produced in group work. The approximate number of hours of work for each member of the group is the following:

- Giovanni Scotti:
- Marco Trabucchi:

NOTE: indicated hours include the time spent in group work.

# Bibliography

- [1] AA 2016/2017 Software Engineering 2 - *Requirements Analysis and Specification Document* - Giovanni Scotti, Marco Trabucchi
- [2] AA 2016/2017 Software Engineering 2 - *Project goal, schedule and rules*
- [3] IEEE Standard 1016:2009 *System design - Software design descriptions*