

# Solar Panel Analyzer

## URD – User Requirements Document

Corso di **Software Engineering**

**Unige** - Università degli Studi di Genova

**Dibris** - Dipartimento di Informatica, Bioingegneria, Robotica e Ingegneria dei Sistemi



**Cliente:**

Wesii s.r.l.



**Versione:** 2.0

**Data:**

30/04/2018

**Autore:**

Jacopo De Luca

## Revision History

Versione	Data	Autore	Note
1.0	22/03/2018	Jacopo De Luca	Prima stesura
1.1	25/03/2018	Jacopo De Luca	Correzioni
1.2	28/03/2018	Jacopo De Luca	Correzioni
1.3	29/03/2018	Jacopo De Luca	Aggiunte
1.4	03/04/2018	Jacopo De Luca	Rifinitura
2.0	30/04/2018	Jacopo De Luca	Seconda stesura

## Sommario

<b>1.</b>	<b>Introduzione .....</b>	<b>4</b>
1.1	Scopo del documento .....	4
1.2	Ambito applicativo del documento .....	4
1.3	Definizioni e acronimi .....	4
1.4	Bibliografia.....	4
1.5	Overview del documento .....	5
<b>2.</b>	<b>Descrizione generale del sistema .....</b>	<b>5</b>
2.1	Contesto e motivazioni.....	5
2.2	Obiettivo .....	5
2.3	Utenti.....	6
2.4	Flusso del software.....	6
<b>3.</b>	<b>User requirements .....</b>	<b>7</b>
3.1	Legenda .....	7
3.2	Tabella .....	7

# 1. Introduzione

## 1.1 Scopo del documento

Questo documento (URD) è richiesto nel corso di Software Engineering (SE) e simula una pratica comune o addirittura obbligatoria nei processi di Ingegneria del Software.

L'ingegnere informatico redige l'URD che definisce le caratteristiche del software da sviluppare; il cliente è invitato a leggere e approvare quanto riportato, poiché parte del contratto che verrà eventualmente stipulato. Se il cliente non si ritenesse soddisfatto, questo documento potrà essere soggetto a modifiche in accordo con l'ingegnere (è buona norma definire l'URD prima della fase di progettazione).

In questo caso il software verrà sviluppato per Wesii s.r.l, quindi di seguito verranno descritte le funzionalità di un tool di riconoscimento e visualizzazione di pannelli solari.

## 1.2 Ambito applicativo del documento

Questo URD è parte del corso di Software Engineering (SE), il cui scopo è lo sviluppo di un software per un'azienda esterna passando per tutte le fasi dell'Ingegneria del Software.

In particolare l'URD in questione è redatto da Jacopo De Luca, studente di Laurea Magistrale in Ingegneria Informatica presso l'Università degli Studi di Genova, ed è destinato all'azienda Wesii s.r.l., la quale si è resa disponibile per la assegnazione di un progetto. Questo documento è quindi una simulazione di una pratica reale tra ingegnere informatico e cliente.

## 1.3 Definizioni e acronimi

Acronimo - Nome	Definizione
SE	Software Engineering
JD	Jacopo De Luca
SPA	Solar Panel Analyzer
DP	File descrittivo particolare
DG	File descrittivo generale

## 1.4 Bibliografia

Articoli vari riguardanti Wesii su forum e riviste:

- SMAU - [smau.it](http://smau.it)
- Tom's Hardware - [tomshw.it](http://tomshw.it)
- StartupItalia! - [startupitalia.eu](http://startupitalia.eu)
- Breaking Tech - [breakingtech.it](http://breakingtech.it)

Narizzano M., Slide del Corso di SE, Genova, 2018

Sito di Wylab - [wylab.net](http://wylab.net)

Sito di Wesii - [wesii.com](http://wesii.com)

## **1.5 Overview del documento**

Il documento è organizzato come segue: in primis verrà descritto il problema da affrontare e una possibile soluzione; nella seconda parte verranno elencate le funzionalità del sistema da sviluppare e la loro priorità di implementazione.

## **2. Descrizione generale del sistema**

### **2.1 Contesto e motivazioni**

Wesii, fondata nel 2016 dall'esperienza di oltre quindici anni di Mauro Migliazzi, è oggi una start-up innovativa che opera nel settore dei droni multispettrali.

La piattaforma include il sistema di controllo automatizzato per il volo dei droni, i sensori multispettrali non convenzionali (camera multispettrale e camera termica radiometrica) di cui sono dotati, il sistema di elaborazione delle immagini ottenute e l'interfaccia con l'utente, grazie al quale il cliente può facilmente accedere alle informazioni di proprio interesse. La soluzione di Wesii è utilizzata in impianti fotovoltaici, nel monitoraggio di discariche, nell'industria agricola, navale e anche in quella sportiva.

In particolare il progetto in questione verrà sviluppato nell'ambito degli impianti fotovoltaici. Il drone permette la raccolta dei dati, oggetto di analisi ed elaborazione, il cui risultato risiede nella geolocalizzazione dei malfunzionamenti. Grazie al riconoscimento del pannello malfunzionante, il cliente potrà intervenire per la riparazione, con il conseguente aumento della produzione energetica.

I problemi operativi si possono individuare nel numero di pannelli da analizzare (si parla di migliaia di moduli per impianto): infatti prima di Wesii i dati venivano raccolti camminando tra questi e qualsiasi operazione, dalla mappatura dell'impianto fino all'individuazione di un pannello specifico, significava dispendio di tempo. Nonostante la soluzione di successo fornita da questa start-up in crescita, tuttora alcuni passaggi di elaborazione, per quanto rapidi e precisi, necessitano ancora di intervento manuale.

### **2.2 Obiettivo**

Il progetto si propone di fornire un'automatizzazione del processo di riconoscimento di ogni singolo pannello solare. Il software è destinato ad uso interno a Wesii e dovrà quindi:

- Permettere l'inserimento di immagini dell'impianto fotovoltaico ottenute dal volo del drone;
- Raccogliere alcuni dati importanti riguardanti i pannelli durante l'elaborazione delle immagini;
- Raccogliere durante l'elaborazione delle immagini alcuni dati importanti riguardanti i pannelli;
- Utilizzare una nomenclatura per l'identificazione dei pannelli;
- Memorizzare le immagini e le informazioni di ogni singolo pannello solare;
- Usare le coordinate geografiche fornite dal volo del drone per la geolocalizzazione di ogni singolo pannello all'interno delle mappe di Google;
- Fornire un'interfaccia grafica essenziale.

Utilizzando servizi sul web (mappe di Google), il software potrebbe aver bisogno di una connessione internet attiva.

## 2.3 Utenti

Poiché il software sarà ad uso interno, utilizzato esclusivamente dai dipendenti di Wesii, l'utente è unico:

- Dipendente Wesii: si interfaccia con il software, fornisce le immagini, può avviare l'elaborazione, può visualizzare i risultati.

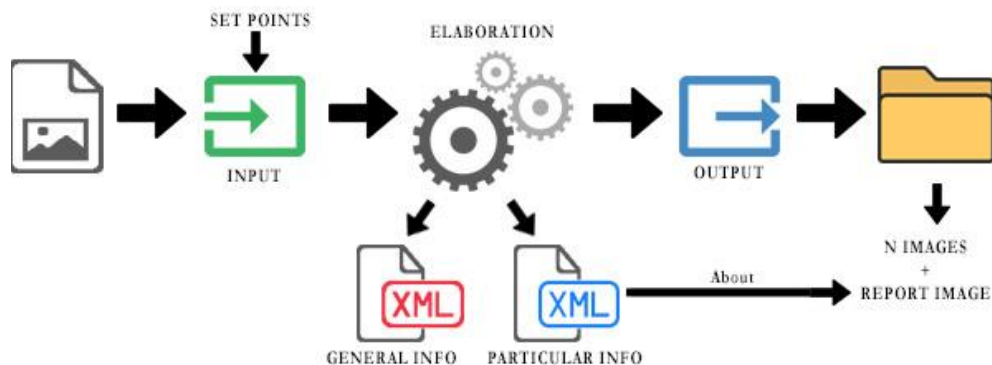
Si potrebbe simulare un utente cliente di Wesii, il quale potrà scegliere il proprio impianto e visualizzare i risultati dell'elaborazione (eventualmente con un sistema di login).

## 2.4 Flusso del software

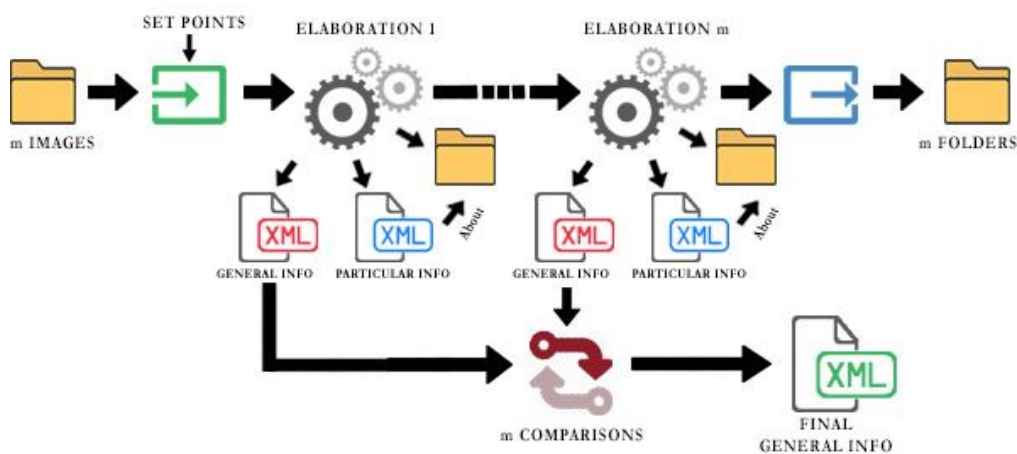
Per meglio descrivere il funzionamento concepito per SPA (Solar Panel Analyzer), di seguito verrà mostrato un grafico.

Verranno mostrati due scenari: una sola immagine in input e set di immagini in input.

### 1) Una immagine in input



### 2) Set di immagini in input



### 3. User requirements

#### 3.1 Legenda

In questa sezione saranno descritti i requisiti dello strumento chiamato SPA (Solar Panel Analyzer). In particolare ad ogni requisito sarà assegnato un ID univoco e una priorità seguendo la seguente tabella:

M	Mandatory	Requisito obbligatorio
D	Desirable	Requisito che dovrebbe essere inserito nel sistema
O	Optional	Funzionalità che può essere inserita nel sistema, a discrezione del manager di progetto. Ad esempio se il tempo di sviluppo è minore di quanto previsto oppure se il costo di implementazione non è troppo alto
E	Futuro Enhancement	Requisito rilasciato per un'eventuale release futura

#### 3.2 Tabella

ID	Descrizione	Priorità
1	<b><u>Input</u></b> Il software deve permettere l'inserimento di tutti i dati necessari per il corretto funzionamento dello stesso. Sicuramente sarà obbligatorio l'inserimento di immagini raffiguranti gli impianti fotovoltaici (RGB e/o termiche - un set di immagini oppure una sola immagine). Potrebbe essere necessario inserire dei valori in input (come ad esempio l'altezza del volo, l'angolo di inclinazione dei pannelli e altri valori ancora <u>da identificare e definire</u> ).	M
2	<b><u>Avvio elaborazione immagini</u></b> Il software deve permettere l'avvio del processo di elaborazione automatizzata. Nel caso di più immagini, il software elaborerà un'immagine per volta in ordine alfabetico/numerico e produrrà un risultato per ciascuna elaborazione.	M
3	<b><u>Interfaccia grafica</u></b> Il software necessita di un'interfaccia grafica essenziale per la scelta delle immagini in input e per l'avvio del processo di elaborazione.	M
4	<b><u>Immagini output</u></b> Il risultato di ogni singola elaborazione deve essere un set di immagini salvate in un'unica cartella: un'immagine per ogni pannello, più l'immagine sorgente con evidenziati tutti i pannelli e alcuni loro punti <u>da definire</u> (chiamata report image).	M

5	<p><b><u>Nomenclatura</u></b></p> <p>Il nome di ogni immagine in uscita deve rispettare una nomenclatura decisa a priori <u>da definire</u> (ad esempio ogni file potrebbe essere nominato con il codice relativo al pannello).</p>	M
6	<p><b><u>Produzione dati</u></b></p> <p>Ogni elaborazione produrrà anche dei dati (<u>da definire</u>) che devono essere conservati. Tali dati saranno distinti in due categorie: dati particolari (relativi ad un'immagine) e dati generali (relativi al set di immagini/impianto).</p> <p>Esempio dati particolari: coordinate pixel, numero pannello nell'immagine. Esempio dati generali: coordinate geografiche, nome del pannello.</p>	M
7	<p><b><u>Gestione dati particolari</u></b></p> <p>Per ogni immagine elaborata verrà prodotto un file descrittivo particolare (DP) contenente le informazioni dei pannelli riconosciuti in quell'immagine e visibili nella report image (requisito 4).</p>	M
8	<p><b><u>Gestione dati generali</u></b></p> <p>Esistono due casi.</p> <p>1) <u>Una sola immagine in input</u>: i dati generali vengono estratti da un'unica immagine, quindi alla fine dell'elaborazione verrà prodotto un file descrittivo contenente le informazioni generali dei pannelli.</p> <p>2) <u>Un set di immagini in input</u>: le immagini verranno elaborate una alla volta (requisito 2) e siccome un pannello può comparire in più immagini, alla fine di ogni elaborazione le informazioni dovranno essere comparate. L'obiettivo è ottenere alla fine della procedura completa un file descrittivo unico contenente le informazioni generali dei pannelli.</p> <p>Tale file descrittivo, sempre unico (anche per un eventuale set di immagini), verrà chiamato file descrittivo generale (DG).</p>	M
9	<p><b><u>Salvataggio output</u></b></p> <p>Le immagini e i file descrittivi particolari (DP) e generale (DG), ovvero l'output del processo di elaborazione delle immagini, devono essere salvati sul pc in uso. In particolare le immagini devono essere salvate in un formato comune, mentre i file descrittivi devono essere salvati in formato txt o xml (<u>da definire</u>).</p>	M
10	<p><b><u>Visualizzazione immagini output</u></b></p> <p>Il software deve permettere la visualizzazione delle immagini in uscita dall'elaborazione automatizzata, ovvero i pannelli. Sarà necessaria quindi almeno una pagina nella quale l'utente potrà visualizzare quanto descritto.</p>	D
11	<p><b><u>Scelta del tipo di immagine da visualizzare</u></b></p> <p>Il software deve permettere la visualizzazione di diverse tipologie di immagini (RGB e termica) dello stesso pannello.</p>	O



12	<b><u>Scheda del pannello</u></b> I dati immagazzinati nei file descrittivi (DP e DG) devono essere accessibili nella pagina descritta nel requisito 10. <u>Da definire</u> se s'intende dati particolari e/o quelli generali. Tali dati comporranno una scheda dettagliata del pannello. L'utente potrà quindi visualizzare la scheda di ogni pannello.	D
13	<b><u>Geolocalizzazione</u></b> Il software deve utilizzare le coordinate geografiche fornite in input (possono essere dei meta dati nelle immagini) e immagazzinate nel file descrittivo generale (DG), per identificare la posizione dell'impianto fotovoltaico e di ogni pannello.	D
14	<b><u>Visualizzazione sulle mappe di Google</u></b> Il software deve permettere la visualizzazione dell'impianto fotovoltaico e dei pannelli sulle mappe di Google. Dovrà quindi esistere una pagina dedicata a questa funzionalità.	D
15	<b><u>Login</u></b> Il software deve disporre di un sistema di login per permettere l'accesso ai dipendenti Wesii o ai clienti, limitando quest'ultimi alla sola visualizzazione delle immagini, delle schede dei pannelli e delle mappe di Google.	O
16	<b><u>Analisi dei pannelli per identificazione malfunzionamenti</u></b> Il software deve analizzare le immagini (direttamente quelle in input oppure quelle in uscita dal processo di elaborazione) per identificare malfunzionamenti del pannello.	E
17	<b><u>Risultato dell'analisi</u></b> Il risultato dell'analisi di un pannello è l'immagine dello stesso, mettendo in evidenza il malfunzionamento e il tipo (ad esempio attraverso una differenziazione di colori). I dati ottenuti dall'analisi devono essere pubblicati nella scheda del pannello.	E