

# MagicMirror-GBM

## configurazione ad uso privato

Documento dei requisiti  
ver. 5.0

17 Aprile 2021

## Indice

<b>1</b>	<b>Premesse del progetto</b>	<b>1</b>
1.1	Obiettivi e scopo del progetto . . . . .	1
1.2	Contesto di business . . . . .	1
1.3	Stakeholders . . . . .	1
<b>2</b>	<b>Servizi del sistema</b>	<b>2</b>
2.1	Requisiti funzionali . . . . .	2
2.2	Requisiti informativi . . . . .	2
<b>3</b>	<b>Vincoli di sistema</b>	<b>3</b>
3.1	Requisiti di interfaccia . . . . .	3
3.2	Requisiti tecnologici . . . . .	3
3.3	Requisiti di prestazione . . . . .	4
3.4	Requisiti di sicurezza . . . . .	4
3.5	Requisiti operativi . . . . .	4
3.6	Requisiti politici e legali . . . . .	4
3.7	Vincoli API esterne . . . . .	4
	<b>Glossario</b>	<b>5</b>

# 1 Premesse del progetto

## 1.1 Obiettivi e scopo del progetto

Piattaforma Open Source modulare per trasformare un classico specchio in un sistema digitale multifunzione.

## 1.2 Contesto di business

Nella continua evoluzione tecnologica degli ultimi anni si è rilevata sempre più utile l'integrazione della domotica e della tecnologia in strumenti di uso quotidiano.

## 1.3 Stakeholders

Le figure che influenzano lo sviluppo del sistema software sono:

- Committente: **NonSoloTelefonia Lab**
- Clienti: **Human-centered design**
- Developers (analisti, progettisti)

## 2 Servizi del sistema

### 2.1 Requisiti funzionali

2.1.1 Il sistema dovrà consentire la modifica delle impostazioni del sistema software stesso.

2.1.1.1 Il sistema dovrà consentire la gestione delle impostazioni di connettività.

2.1.1.2 Il sistema dovrà consentire la gestione del microfono e della fotocamera.

2.1.1.3 Il sistema dovrà permettere il reset del sistema stesso.

2.1.1.4 Il sistema dovrà permettere una fase di configurazione iniziale.

2.1.1.5 Il sistema dovrà permettere la modifica della lingua.

2.1.1.6 Il sistema dovrà permettere la regolazione del volume.

2.1.1.7 Il sistema dovrà permettere la gestione delle notifiche.

2.1.2 Il sistema dovrà integrare il modulo **MMM-AirQuality**.

2.1.3 Il sistema dovrà integrare il modulo **MMM-DHT**.

2.1.4 Il sistema dovrà integrare il modulo **newsfeed**.

2.1.5 Il sistema dovrà integrare il modulo **MMM-AVStock**.

2.1.6 Il sistema dovrà integrare il modulo **clock**.

2.1.7 Il sistema dovrà integrare il modulo **weather**.

2.1.8 Il sistema dovrà integrare il modulo **weatherforecast**.

2.1.9 Il sistema dovrà integrare il modulo **calendar**.

2.1.10 Il sistema dovrà integrare il modulo **MMM-MD**.

2.1.11 Il sistema dovrà integrare il modulo **MMM-Screencast**.

2.1.12 Il sistema dovrà integrare il modulo **MMM-Mail**.

2.1.13 Il sistema dovrà integrare il modulo **MMM-Online-State**.

2.1.14 Il sistema dovrà integrare il modulo **MMM-ip**.

2.1.15 Il sistema dovrà entrare in *fase di standby* quando esso non rileva più movimenti (usa sensore: hc-sr501).

### 2.2 Requisiti informativi

Il linguaggio utilizzato per lo scambio di informazioni tra le componenti interne ed esterne del sistema è il JSON.

## **3 Vincoli di sistema**

### **3.1 Requisiti di interfaccia**

L'interfaccia proposta dal sistema è stata appositamente studiata per garantire una fruizione di contenuti intuitiva ed immediata.

#### **3.1.1 Interfaccia principale**

- 3.1.1.1 Visualizzazione della qualità dell'aria per la zona specificata.
- 3.1.1.2 Visualizzazione della temperatura e dell'umidità locali (usa sensore: DHT22).
- 3.1.1.3 Visualizzazione delle news più recenti.
- 3.1.1.4 Visualizzazione degli aggiornamenti relativi alle quotazioni in borsa.
- 3.1.1.5 Visualizzazione della data e dell'ora correnti.
- 3.1.1.6 Visualizzazione delle previsioni meteo.
- 3.1.1.7 Visualizzazione di un calendario interattivo.
- 3.1.1.8 Visualizzazione delle annotazioni.
- 3.1.1.9 Visualizzazione delle email in entrata.
- 3.1.1.10 Visualizzazione dello stato della connessione.
- 3.1.1.11 Visualizzazione dell'indirizzo ip.

#### **3.1.2 Interfaccia modulo MMM-Screencast**

- 3.1.2.1 Visualizzazione di contenuti multimediali.

### **3.2 Requisiti tecnologici**

L'intero progetto è stato realizzato utilizzando i seguenti strumenti:

- Raspberry Pi modello 2 o superiore
- Casse audio
- Schermo con interfaccia HDMI
- Telaio specchio
- Two-way mirror
- Sensore rilevatore di temperatura ed umidità locali (DHT22)
- Sensore di movimento P.I.R. (hc-sr501)

### **3.3 Requisiti di prestazione**

Non si registrano particolari esigenze in questo ambito.

### **3.4 Requisiti di sicurezza**

Non si registrano particolari esigenze in questo ambito.

### **3.5 Requisiti operativi**

L'intero progetto è stato realizzato utilizzando i seguenti linguaggi:

- JavaScript
- CSS
- HTML
- PHP

L'intero progetto è basato sulle seguenti piattaforme:

- npm + Node.js v10.x o superiore
- Electron

Si relaziona con sistemi operativi Raspberry Pi OS (full version).

### **3.6 Requisiti politici e legali**

Il sistema software open source è rilasciato sotto la licenza [Apache-2.0](#).

### **3.7 Vincoli API esterne**

L'utilizzo di API esterne è soggetto a limitazioni poste dai fornitori delle API stesse. Pertanto si invita ad una consultazione dei regolamenti di utilizzo delle singole API.

## Glossario

**Human-centered design** approccio di problem solving che coinvolge la prospettiva del cliente in tutti gli step della risoluzione stessa.

**Sensore di movimento P.I.R.** acronimo di *Passive InfraRed* è un sensore elettronico che rileva la radiazione infrarossa (IR) irradiata dagli oggetti nel suo campo visivo.

**Two-way mirror** particolare tipo di specchio che da un lato riflette la luce mentre dall'altro ne permette il passaggio.