



Università degli Studi di Padova



Gruppo Bug Pharma

bugpharma10@gmail.com

## Verbale esterno del 16-11-2021

Imola Informatica S.p.A. - Bot4Me

Approvazione	-
Redazione	Andrea Salmaso
	Nicla Faccioli
Verifica	-
Stato	Redatto
Uso	Esterno

# 1 Informazioni generali

## 1.1 Luogo e data dell'incontro

- **Luogo:** videoconferenza Microsoft Teams;
- **Data:** 16-11-2021;
- **Ora di inizio:** 16:00;
- **Ora di fine:** 16:30.

## 1.2 Presenze

- **Totale presenze:** 5 su 7;
- **Presenti:**
  - Michele Masetto;
  - Nicla Faccioli (segretaria);
  - Sara Nanni;
  - Andrea Salmaso;
  - Nicholas Sertori.
- **Assenti:**
  - Lorenzo Piran;
  - Silvia Giro.
- **Partecipanti esterni:**
  - Alessandro Proscia (Imola Informatica S.p.A.);
  - Giacomo Lorenzo (Imola Informatica S.p.A.).

## 2 Domande e risposte

1. **Il resoconto delle attività fatto dai dipendenti è vincolato dal timbro di entrata? Deve essere effettuato obbligatoriamente prima del timbro di uscita?**

- Dipende molto da quanto è diligente il dipendente:
  - Non si tratta di un'attività giornaliera, ma ogni dipendente può effettuarla con cadenze differenti, a propria discrezione;
  - Dipende dalle regole dell'azienda: ad Imola Informatica non vi è un orario di ingresso o un cartellino da timbrare, per cui non vi sono obblighi in questo senso.

2. **Per quanto riguarda la gestione della presenza in sede e il tracciamento delle presenze ci sono limitazioni?**

L'azienda utilizza un'applicazione per la consuntivazione completamente auto-contenuta al cui interno:

- Sono presenti le attività giornaliere su cui si può consuntivare;
- È possibile effettuare l'operazione di check-in in cui si può specificare anche la sede in cui si è presenti al momento.

Quindi il tracciamento è un'attività già inclusa nella suddetta.

3. **Quindi il bot che dobbiamo creare si appoggerebbe su questo sistema già esistente?**

- L'azienda fornirà un set di API *REST* definite mediante un file *Swagger*;
- È stato illustrato brevemente dal proponente il funzionamento di un file *Swagger*:
  - È un linguaggio che permette la definizione di API *REST*, fornendo i formalismi per la definizione dei servizi da esporre utilizzando il paradigma *REST* (path esposti, metodi e così via);
  - Un file *Swagger* si divide in tre parti:
    - \* Una parte di metadata atta ad indicare:
      - Quali sono le macro categorie di servizi esposti, attraverso una loro descrizione;
      - Quali sono le licenze con cui si stanno esponendo le suddette API (*http* o *https*, base path ecc.).
    - \* La definizione dei vari servizi esposti:
      - Ogni servizio è esposto su un path ed ha associati dei metodi *http*;
      - *REST* indica che è possibile accedere a diverse risorse, ciascuna delle quali ha associate alcune operazioni;
      - Ogni operazione restituisce un *json* di risposta in cui sono contenute informazioni rilevanti per il tipo di operazione;
      - Ogni variazione sul path identifica una risorsa differente.
    - \* Una parte di definizione degli oggetti usati dalle varie API:
      - Ogni oggetto avrà una serie di caratteristiche.

4. **Per quanto riguarda il requisito opzionale sull'accesso ai documenti: tale accesso dipende dai privilegi del dipendente (e quindi dal suo ruolo all'interno dell'azienda) oppure è una cosa più generica in cui tutti hanno accesso a tutto?**

- In azienda viene già utilizzato *Alfresco* un sistema documentale in cui l'utente ha associati alcuni ruoli;
- A seconda del ruolo l'utente vede o non vede alcune cose;
- Per quanto riguarda *Alfresco*:
  - Lo standard *CIVIS* si occupa della standardizzazione totale dell'accesso documentale;
  - È possibile trovare librerie già pronte che potrebbero tornare utili (es.: *Apache Chemistry* per *Java*).
- Per quanto riguarda le scelte tecnologiche, il gruppo è libero di scegliere ciò che vuole.

#### 5. L'azienda ha qualche preferenza sul tipo di formato dei documenti da consegnare?

- Non ci sono preferenze per quanto riguarda il formato;
- A livello di progetto software, oltre al codice i documenti solitamente consegnati sono:
  - Analisi funzionale: per descrivere a livello funzionale qual è il progetto, quali sono i requisiti e come funziona;
  - Analisi tecnica: per descrivere in dettaglio le parti tecniche;
  - Macro schema architetturale: per descrivere com'è fatta l'applicazione;
  - Analisi degli use case;
  - Manuale d'uso.

#### 6. Come si gestisce un errore nell'interazione con l'utente come un'incomprensione del messaggio?

- Viene restituito all'utente un generico messaggio di errore (come fanno *Alexa* o *Google Assistant*);
- Non è necessaria compensazione (rollback del sistema per riportarlo allo stato originale) in caso di errori.

#### 7. Consigli o aspettative da questa collaborazione?

- La risposta è stata riformulata dal proponente in: "Cosa deve aspettarsi il gruppo da Imola?":
  - Verrà messo a disposizione un canale diretto di comunicazione per chiarire dubbi o perplessità (anche su *Telegram*);
  - L'azienda rimane disponibile per meeting periodici in stile *scrum*, anche per eventuali dubbi tecnici o nel caso in cui il gruppo si trovasse bloccato su qualche aspetto.
- Dall'altro lato, il proponente si aspetta di vedere un prodotto funzionante.
- Sono state fornite altre generiche informazioni sul chatbot:
  - Usare le tecnologie che si ritengono più opportune;
  - Cercare materiale in autonomia, facilmente recuperabile.
- Sono stati forniti alcuni collegamenti utili a varie risorse:
  - Strumenti:
    - \* *Swagger*;
    - \* *Apache Chemistry*;
    - \* *Chatterbot*;
  - Contatti:
    - \* *Alessandro Poscia* (*Telegram*);
    - \* *Giacomo Lorenzo* (*Telegram*).