INGLESE  
**Project Briefing for the Development of the Micro-Management System**

### **1. Introduction**

Hello everyone,

We need to develop a micro-management system that connects the operators of our agency with clients, enabling them to work optimally and complete all tasks effectively.

### **2. Required Technology Stack**

* **Backend:** PHP
* **Framework:** Laravel or similar
* **Frontend:** Bootstrap or a similar framework for design

### **3. Available Resources**

* **Graphic Template:** We already have a graphic template developed up to 70%, and we would like to start working from this. The template is available at: [insert link].
* **Existing Code:** We also have pure PHP code (without a framework) where a series of functionalities have already been developed, such as task management, the work diary, etc.

### **4. User Types and Permissions**

There are three types of users:

#### **4.1 Administrator**

* **Project Management:** Can create projects and assign them to the client and one or more operators.
* **Configuration Management:** Can manage variables such as passwords, emails, etc.
* **Notification Settings:** Can choose the emails that should receive notifications for specific events (e.g., task completed).
* **Project Options:** Can modify the options of individual projects (e.g., who should receive notifications and with which email or push template).
* **Miscellaneous:** Various other administrative functions as needed.

#### **4.2 Operator**

The operator is the person who works on the projects. In their dashboard, as per the template, they must be able to:

* **Respond to Client Requests:** Reply to client inquiries.
* **Create Requests:** Generate requests that the client must respond to.
* **Task Management:** Create tasks and submit them for review (only the client can confirm that the task is completed).
* **Task Deletion:** Can delete tasks but only if they know the password.
* **Project Exploration:** Explore the projects assigned to them.

#### **4.3 Client**

* **Work Monitoring:** Can monitor the progress of the work.
* **Respond to Requests:** Can respond to requests made by the operator.
* **Support Tickets:** Can create support tickets for the operator.
* **Project Briefing:** Can read the project briefing.
* **File Attachments:** Can attach requested files.

### **5. Technical Operation**

#### **5.1 General Dashboard**

1. **Project Selection:** Choose which project to select from those assigned. Destination: Operator/Client.
2. **Urgent Alerts:** View notifications that need to be responded to or completed. Once completed, they should disappear.
3. **Client Communications:** View communications requested by the client.
4. **Requests to Client:** Create requests that the operator must pose to the client, for example: “Upload the logo.”
5. **Client Tickets:** View and respond to tickets that the client opens to request support from the operator and digital culture teams.

#### **5.2 Module**

##### **5.2.1 Project Details**

* **Hours Worked:** Display the number of hours worked (calculated from declared hours or auto-calculated from tasks).

##### **5.2.2 Daily Update**

* **Daily Reporting:** Every day, the operator must write and send a “Daily update” on the project. If the operator fails to send it by the end of the day, ideally by 7:00 PM - 8:00 PM, a sad face icon should appear, and an alert should notify them that they did not send an update on time (the alert should be visible on the dashboard and also sent via email).

##### **5.2.3 Work Diary**

* **Activity Log:** The Work Diary page should allow viewing the LOG of EVERYTHING done in the project (the complete history log). Additionally, a section should be dedicated solely to daily updates, where each sent and unsent daily update can be viewed to keep track of everything.

##### **5.2.4 Project Progress**

* **Progress Bar:** Display a project progress bar from 0 to 100% (calculated based on remaining days and the deadline).
* **Data Export:** Ability to export all content in XLS format.

##### **5.2.5 Days Remaining Until Delivery**

* **Automatic Deadline Calculation:** The deadline should be automatically calculated based on the number of tasks, their latest deadlines, and the number of requests or tickets the user opens.
* **Progress Blockage:** If the user does not respond to the operator’s requests, the project progress and deadline calculation should BLOCK (this needs to be shown graphically). Subsequently, when the project resumes, the days lost waiting for client responses should be added.
* **Manual Adjustment:** It must be possible to manually modify this count to fix what is shown to the client.

### **6. Task Management**

#### **6.1 Task Creation**

* **Basic Information:** Tasks are created by providing a TITLE, DESCRIPTION, DEADLINE, ATTACHMENTS, and STATUS.
* **Sub-Tasks:** Tasks can have sub-tasks.
* **Linking to Requests:** It should be possible to link a task to a request made to the client, which will remain blocked until the client responds.
* **Priority Setting:** Tasks should have the ability to set their priority.

#### **6.2 Task Notifications**

* **Completion Alerts:** When a task is completed or submitted for review, a notification should be sent to the administrator (or to the email set as the project manager).
* **Batch Import:** It should be possible to import tasks in BATCH using a simple TXT template.
* **Deadline Management:** If a deadline is missed, the operator must view an alert and provide the reason for the delay. If the operator needs to reschedule a task deadline, they must provide the reason and request rescheduling, which the administrator can approve; otherwise, the original deadline remains.
* **Task Export:** Ability to export all task content in XLS format.

### **7. Notifications**

#### **7.1 Notification Delivery**

* **Email Notifications:** Notifications must be sent via email.
* **Template Management:** The administrator should have a panel to modify the templates of the emails being sent.
* **Project-Specific Recipients:** For each project, the administrator must be able to choose the recipient by entering one or more email addresses.

#### **7.2 Automated Notifications**

* **Client Non-Response:** If a client does not respond to the operator’s requests, they must receive regular emails prompting them to respond to the request (the administrator should be able to choose the base message template and the reception frequency).
* **Operator Task Deadlines:** If an operator has tasks nearing their deadlines, they must receive regular notifications alerting them of the upcoming task deadlines.
* **Operator Non-Response:** If an operator does not respond to client communications or requests, they must receive regular notifications (again, the administrator should be able to modify the message and notification frequency).

### **8. Client Dashboard**

* **View Alerts:** Must view all alerts that require responses or attention.
* **Request Management:** Can create requests directed at operators and digital culture teams.
* **Respond to Requests:** Can respond to requests.
* **Support Tickets:** Can create support tickets.
* **Project Details:** Can view all project details.
* **Task Management:** Can view ongoing tasks, their deadlines, and can approve a task that the operator has declared complete.
* **Project Briefing:** Must be able to read the project briefing.
* **File Attachments:** Can attach files (both in requests and tickets, and on a dedicated page).

#### **8.1 Important Note**

Every time the client creates a ticket, request, or communication, it must become a new task in the list of tasks to be resolved.

ITALIANO

Salve a tutti,

Abbiamo necessità di sviluppare un micro-gestionale che metta in connessione gli operatori della nostra agenzia con i clienti in modo che possano lavorare al meglio e portare a termine tutti i lavori

Stack Tecnologico richiesto

* PHP
* Framework - Laravel o Similare
* Bootstrap o similare per il design

Abbiamo già un template grafico sviluppato al 70% e gradiamo partire da questo per lavorare. Il template è visibile a questo indirizzo : \_\_

Abbiamo anche del codice PHP (puro senza framework) dove sono state già sviluppate una serie di funzionalità, come la gestione dei task, il diario etc…

Le tipologie di utenti sono 3

**Amministratore**

* Puoi creare progetti ed assegnarli al cliente e ad 1 o più operatori
* Può controllare le variabili come password, email etc..
* Può scegliere le email che devono ricevere le notifiche di avvenimento di qualcosa (esempio task completato)
* Può modificare le opzioni dei singoli progetti (come ad esempio chi deve ricevere le notifiche e con quale template email o push)
* Varie ed eventuali

**Operatore**

* L’operatore è la persona che lavora sui progetti
* Nel suo pannello, come da Template deve poter
* Rispondere alle richieste del cliente
* Creare delle richieste (a cui il cliente deve rispondere)
* Creare dei task e metterli in revisione (Solo il cliente può confermare che il task è completato)
* (Può eliminare dei task ma solo se è a conoscenza della password)
* Può esplorare i progetti a lui assegnati

**Cliente**

* Può monitorare l’andamento del lavoro
* Può Rispondere alle richieste che vengono poste dall’operatore
* Può creare un ticket di assistenza all’operatore
* Può leggere il briefing del progetto
* Può allegare file che gli vengono richiesti

Funzionamento Tecnico

Cruscotto Generale

1 Selezione Progetto (Scelgo quale progetto selezionare tra quelli assegnati) Destinazione Operatore/Cliete

2 Avvisi urgenti : Qui posso leggere le notifiche alle quali devo ancora rispondere o che devo completare. Quando sono completate devo scomparire

3 Comunicazioni del cliente : Qui vengono visualizzate le comunicazioni che il cliente richiede

4 Richieste al cliente : Qui vengono create le richieste che l’operatore deve porre al cliente, ad esempio : “Caricami il logo”

5 Ticket Cliente : Qui visualizza e può rispondere ai Ticket che il cliente apre per chiedere supporto all’operatore ed a culture digitali

Modulo

Dettagli progetto

In questa pagina è possibile visualizzare il numero di ore lavorate (Le deve conteggiare da quelle dichiarate o auto-calcolate nei task)

Daily Update

Ogni giorno l’operatore deve scrivere ed inviare un “Daily update” sul progetto. Se l’operatore non riesce ad inviarlo entro la fine della giornata, idealmente alle 19.00 - 20.00 una faccina piangente deve comparire ed un alert deve avvertirlo che non ha mandato un update in tempo (L’alert deve essere visibile sul cruscotto e deve essere inviato anche a mezzo mail).

Work Diary

Nella pagina Work Diary deve essere possibile visionare il LOG di TUTTO quello che viene fatto nel progetto (Proprio tutto l’history log) inoltre una sezione deve essere dedicata solo alla parte legata ai daily update, dove si deve poter visionare ogni daily update inviato ed anche quello non inviati, in modo da tenere traccia di tutto.

Progresso del progetto : Deve visualizzare una barra di caricamento del progetto da 0 a 100% (calcolato in base ai giorni rimanenti e alla deadline di scadenza)

Deve essere possibile esportare in formato XLS tutto il contenuto

Giorni rimanenti alla consegna : La deadline deve essere calcolata automaticamente in base a quanti sono i task, qual’è la loro ultima scadenza ed in base al numero di richieste o ticket che l’utente apre

Se l’utente non risponde alle richieste poste dall’operatore il progresso del progetto ed il calcolo della scadenza si BLOCCA (Bisogna mostrarlo graficamente) e successivamente, quando il progetto riparte, devono essere aggiunti i giorni persi in attesa delle risposte del cliente.

OVVIAMENTE deve essere possibile modificare a mano questo conteggio, in modo da fixare ciò che va mostrato al cliente

TASK

Creazione dei task

I task vengono creati semplicemente, mettendo un TITOLO una DESCRIZIONE una SCADENZA degli ALLEGATI e LO STATUS  
I TASK devono poter avere dei sotto-task

Deve essere possibile legare un task anche ad una richiesta da fare al cliente che chiaramente resterà bloccato fin quando il cliente non risponderà

I Task devono avere anche la possibilità di settarne la priorità

Quando un task viene completato, messo in revisione, deve arrivare una notifica all’amministratore (O alla email che è stata settata come PM del progetto)

Deve essere possibile importare dei task in BATCH con un template TXT semplice  
Se una scadenza viene saltata, l’operatore deve visionare un alert e deve inserire il motivo del ritardo

Se l’operatore deve riprogrammare la scadenza di un task, deve inserire il motivo e fare richiesta per riprogrammarla, sarà l’amministratore che può approvare la richiesta, altrimenti la scadenza resta quella messa all’inizio.

**Deve essere possibile esportare il formato XLS tutto il contenuto dei task**

**NOTIFICHE**

Le notifiche devono essere inviate a mezzo email

L’amministratore deve avere un pannello dove poter modificare il template delle email che vengono trasmesse

Per ogni progetto, l’amministratore deve poter scegliere il destinatario, inserendo uno o più indirizzi email

Se un cliente non risponde alle richieste dell’operatore, deve ricevere una email con cadenza regolare che gli chiede di rispondere alla richiesta (l’amministratore deve poter scegliere il template di base del messaggio e la cadenza di ricezione)

Se un operatore ha dei task in scadenza, deve ricevere una notifica regolare che lo avverte dei task in scadenza

Se un operatore non risponde a comunicazioni o richieste del cliente deve ricevere una notifica con cadenza regolare (Anche in questo caso l’amministratore deve poter modificare il messaggio e la cadenza della notifica)

Cruscotto Cliente

Deve visionare tutti gli avvisi ai quali deve rispondere o fare attenzione

Può creare delle richieste verso operatore e culture digitali

Può rispondere alle richieste

Può creare un ticket di assistenza

Può visionare tutti i dettagli del progetto

Può visionare i task in corso, le loro scadenze e può anche approvare un task che l’operatore ha dichiarato completo

Deve poter leggere il briefing del progetto

Deve poter allegare dei file (Sia nelle richieste che nei ticket che in una pagina dedicata).

**ATTENZIONE**

Ogni volta che il cliente crea un ticket, una richiesta o una comunicazione, la stessa deve divenire un nuovo task nella lista dei task da risolvere.