Finsoft srl

Osai AS SpA

Marzo 2021

Piattaforma gestione cassettiere

## Abstract

Si desidera offrire uno strumento che permetta ai montatori di segnalare quali materiali di consumo siano in esaurimento nei capannoni di produzione e chiederne il rabbocco senza interrompere l'attività.

## Interfaccia Hardware

L'idea è quella di preparare una stazione simile al bolla&stampa da montare vicino alle cassettiere (touchscreen da ~7” in landscape con browser in modalità kiosk) per il frontend.

I dispositivi saranno dotati di lettori QRCODE e RFID in modalità emulazione tastiera.

Gli operatori utilizzeranno il backend dal proprio pc.

## Attività interna OSAI

Per quanto riguarda la configurazione della stazione è possibile riciclare buona parte del sw già preparato per il tabellone presenze, ma sarebbe ragionevole considerare 1 giornata di aggiornamento sw e configurazione pistola del QR.

Valutare eventuali API per attivazione/disattivazione hardware (lettore QRCODE e lettore RFID).

## Architettura

1. Ipotizziamo di sviluppare due applicazioni web su stack Linux+Apache+MySQL+PHP, frontend e backend.
2. Il frontend deve essere pensato per l’utilizzo su tablet, mentre il backend per l’utilizzo da postazioni desktop.

## Funzionalità previste

La piattaforma di gestione delle cassettiere prevede queste funzionalità:

* (frontend) Form di segnalazione esaurimento (per utente):
  1. L’utente scansiona il QR code
  2. Il software decodifica il codice, recupera l’anagrafica articolo e ne mostra i dettagli
  3. Se l’articolo ha già una segnalazione attiva viene mostrato un avviso con un pulsante per tornare indietro (dopo 15 secondi di inattività si torna alla pagina principale)
  4. Se l’articolo non ha segnalazioni attive viene richiesta conferma tramite lettura del badge RFID dipendente per segnalare l’esaurimento (5 tentativi) oppure pulsante per annullamento
  5. Alla conferma o all’annullamento (o dopo 30 secondi di inattività) si torna alla pagina principale
  6. La lettura del badge garantisce l’autenticazione per 60 secondi:
     1. quando l’autenticazione è “attiva” la pagina principale mostra il nome utente e il pulsante “logout”
     2. le richieste presenteranno solo i pulsanti “conferma” e “annulla”
     3. ciascuna richiesta garantisce l’autenticazione per i 60 secondi successivi
* (frontend) Form di rabbocco per logistica: raggiunto attraverso un pulsante apposito nel form di segnalazione esaurimento permette all’operatore di annullare la segnalazione esaurimento.
  1. L’operatore preme il pulsante per accedere alla schermata di rabbocco
  2. L’operatore si autentica tramite badge RFID (5 tentativi)
  3. L’operatore prende la pistola QR e scansiona i codici delle ubicazioni vuote mentre le rabbocca; dato che potrebbe essere distante dal display un suono confermerà la corretta lettura (o la lettura di un codice non corretto)
  4. Le ubicazioni verranno accodate su una lista a video in modo da poterle consultare agevolmente; eventuali doppioni verranno automaticamente rimossi
  5. Al termine dell'operazione l’operatore ritorna al terminale e conferma il rabbocco delle ubicazioni scansionate
  6. Alla conferma viene effettuato il logout in automatico. L’operatore può annullare l’operazione facendo logout manualmente (timeout da definire per l’intera operazione)
* (backend) Report stato ubicazioni (per gruppo logistica): uno schema grafico che permetta di identificare rapidamente lo stato delle scorte, va concordato il layout. Un esempio potrebbe essere un primo report per area (E1, M6, M7) da cui selezionare l'area per analizzare le singole ubicazioni;
* (backend) Report segnalazioni attive in formato tabellare, esportabile in CSV. Il file esportato serve per alimentare un’altra piattaforma per lo scarico del magazzino.
* (backend) Storico richieste di rabbocco/operazioni di rabbocco con utente e data. Occorre prevedere numerosi filtri sui dati: per area, utente, range temporale, articolo contenuto, tipologia di operazione.
* (backend) Maschera di modifica delle aree
* (backend) Maschera di modifica delle ubicazioni

L’autenticazione lato backend sarà effettuata tramite LDAP (utente, password, profili utente). Sul frontend sarà disponibile solo l’accesso/conferma tramite badge.

Prevedere che tutte le tabelle siano esportabili in formato XLSX e in CSV (il formato CSV serve come input per altri applicativi).

Se possibile il lettore badge dovrebbe essere attivato solo durante l’autenticazione per evitare interferenze durante l’operatività.

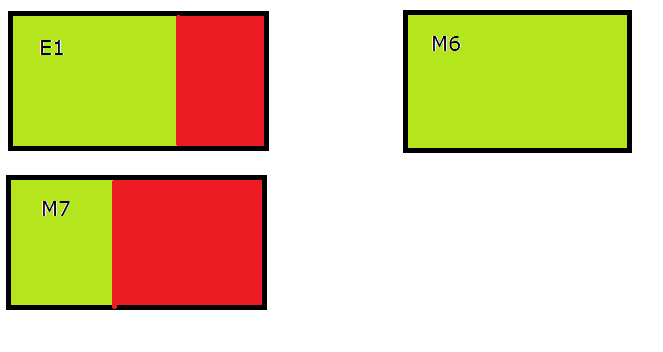
## Dati gestiti

Le informazioni gestite dalla piattaforma sono:

* ubicazioni: contenitore fisico in cui è contenuto il materiale
  + stato: pieno o vuoto
  + codice articolo contenuto
  + quantità
  + codice area
* articolo: prelevato da Panthera
  + codice articolo
  + codice disegno
  + descrizione
* area: raccoglie tutti i contenitori geograficamente vicini (es. tutti i contenitori di M7)
  + codice area
  + descrizione
* operazione: segnalazione esaurimento o rabbocco
  + codice operazione
  + descrizione
* utente (questi dati non stanno su database ma su server LDAP): utente che sta segnalando la mancanza o operatore che sta rabboccando
  + codice: codice utente Panthera
  + username: utente LDAP
  + descrizione: nome cognome
  + profilo utente (utente normale oppure validatore)
* badge: tabella presente su altro db MySQL con abbinamento badge - utente
  + codice badge
  + codice utente (v. sopra)
* storico operazioni:
  + timestamp
  + codice utente
  + codice articolo
  + codice ubicazione
  + codice operazione

## Report stato ubicazioni

Una prima schermata mostra tutte le aree, colorate in verde/rosso proporzionalmente al numero di cassettiere in esaurimento presenti nell’area:



Selezionando una singola area, verrà proposto un report tabellare di dettaglio delle singole ubicazioni:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Cod. ubicazione** | **Articolo** | **Descrizione** | **Quantità prevista** | **In Esaurimento** |
| A1 | Xxx |  |  | X |
| A2 | yyy |  |  |  |
| A3 |  |  |  |  |
| A4 |  |  |  | X |
| … |  |  |  |  |

## Esempio cassettiera



## Ipotesi di base dati

