

Documentazione Tecnica



Versione del 12/02/2025

Prerequisiti

- .NET 8 (richiesto per servizio <https://dotnet.microsoft.com/it-it/download/dotnet/8.0> [🔗](#)):
 - ASP .NET Core Runtime 8.0.11 <https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-aspnetcore-8.0.11-windows-hosting-bundle-installer> [🔗](#)
 - .NET Desktop Runtime 8.0.11 <https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-desktop-8.0.11-windows-x64-installer> [🔗](#)
 - .NET Runtime 8.0.11 <https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-8.0.11-windows-x64-installer> [🔗](#)
- NodeJs (richiesto per l'applicativo client <https://nodejs.org/en> [🔗](#))
- Microsoft SQL Server dal 2022 (qualsiasi licenza)
- PM2 (<https://pm2.io/docs/runtime/guide/installation/> [🔗](#))
- Elasticsearch (in \\main5\Prodotti_scao come file EKC.7z)

SCHEMA GENERALE SERVIZI E MICROSERVIZI

- **eTabula per la schedulazione e la gestione delle fasi**
 - porta 9090 (o qualsiasi altra disponibile)
 - richiede: IIS, SQL, SSMS, .NET 8
- **PM2 gestore microservizi (eTabula, RabbitMq)**
 - richiede: NodeJS
- **BACKUP**
 - pianificazione dei backup dei DB in windows
- **LICENZE**
 - chiavi di attivazione per eTabula
- **VARI**
 - Browser Chrome (chromium), Connessione Desktop Remoto(MSTSC)

Procedura d'installazione

- Connettersi al Server del cliente con Connessione Desktop Remoto(MSTSC)
- Creare le cartelle
 - C:\scao (qui avremo il software necessario all'utente)
 - C:\scao\setup (qui copieremo tutti i file d'installazione ed info varie del cliente)

Installazione NodeJs, Visual Studio Code, 7zip e Chrome

Procedere alle installazioni dei rispettivi file .exe



Per Visual Studio Code, durante l'installazione abilitare le checkbox riguardanti "apri con visual studio code".

Installare Internet Information Services (IIS)

Installazione PM2

Installare RabbitMQ

Installare eTabula

Di seguito si elencano i passi principali per l'installazione di eTabula:

1. Copiare le cartelle client e server dell'ultima versione di etabula da **\main5\Prodotti_scao\e-tabula2** sotto:
C:\inetpub\wwwroot\eTabula (NOTA: copiare il file .zip sul desktop e poi estrarlo in
C:\inetpub\wwwroot\eTabula così evito problemi con i permessi).
2. Sotto la cartella server, modificare il file **appsettings.json** come indicato di seguito nella sezione "*File di configurazione -> Server (appsettings.json)*"
3. Sotto la cartella client C:\inetpub\wwwroot\eTabula\client\assets modificare il file **config.json** modificando appositamente gli url a <http://IPxx> :porta_libera/server/ quindi l'IP, una porta libera del sito web su cui installeremo etabula (dovrebbe essere libera la 9090) quindi aggiungere "/server" agli url in quanto il server si trova sotto l'applicazione server del sito IIS dove risiede il client.
Se è libera la porta 80 posso utilizzare anche quella e quindi poi non sarà più necessario specificarla in quanto è quella di default.
4. Da IIS andare su Siti-> Aggiungi sito web -> settando nome:etabula, puntare alla cartella del client e porta 9090 (se libera), successivamente dal sito etabula creare una nuova applicazione facendo "Aggiungi applicazione" mettendo come alias: "**server**" e puntando al percorso fisico della cartella "C:\inetpub\wwwroot\eTabula\server"; dalle impostazioni avanzate del Pool di applicazioni -> etabula -> tasto dx->impostazioni avanzate e impostare correttamente i parametri come da Figura 8
5. In caso di Virtual Directory per accedere ad una cartella esterna di file contenente ad es. .pdf, dal sito IIS di etabula andare sotto "Directory Browsing" o "Esplorazione Directory" e fare **enable** dal pannello a dx.
6. Modificare i permessi all'intera cartella C:\inetpub\wwwroot\eTabula facendo -> tasto dx-> proprietà -> protezione ed aggiungere l'utente IIS_IUSRS impostando il controllo completo. Come da Figura 7.

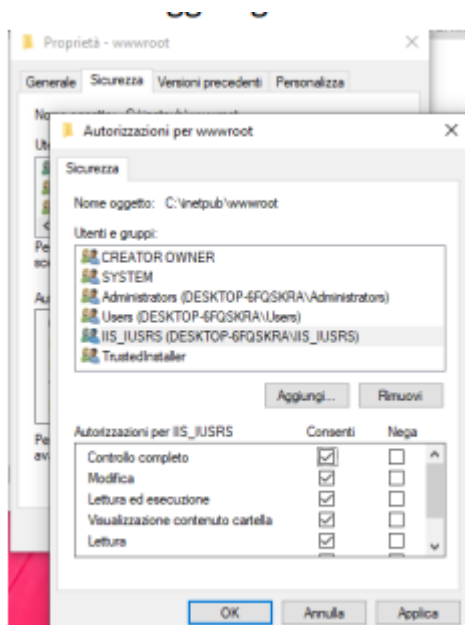


Figura 7- Permessi cartella wwwroot\etabula

8. Dal browser Chrome accedere al sistema tramite l'url: <http://localhost:9090/> ed inserire per il primo accesso il codice licenza di attivazione per registrare eTabula.
In caso di errore su file non accessibile verificare i permessi.

Aggiornare eTabula

1. Arrestare da IIS il pool relativo all'applicazione eTabula da aggiornare
2. Rinominare le precedenti cartelle client e server in modo da avere un backup delle stesse e copiare le cartelle della nuova versione
3. Effettuare un backup del DB
4. Copiare i precedenti file di configurazione delle cartelle client e server, in particolare:
 - a) C:\inetpub\wwwroot\etabula\server\appsettings.json
 - b) C:\inetpub\wwwroot\etabula\client\zoom.js
 - c) C:\inetpub\wwwroot\etabula\client\config.json
 - e) Copiare anche la cartella "exec" del server sotto quella del client
 - f) In caso di Virtual Directory per accedere ad una cartella esterna di file contenente ad es. .pdf, dal sito IIS di etabula andare sotto "Directory Browsing" o "Esplorazione Directory" e fare enable dal pannello a dx.
5. Modificare i permessi con controllo completo con utente IIS_IUSRS a tutta la cartella C:\scao\etabula\
6. Da IIS avviare il pool di applicazione eTabula
7. Testare il funzionamento verificare che abbia preso il numero della nuova versione ed effettuare una ricerca di test da una soluzione per vedere che funzioni, eventualmente analizzare i log.

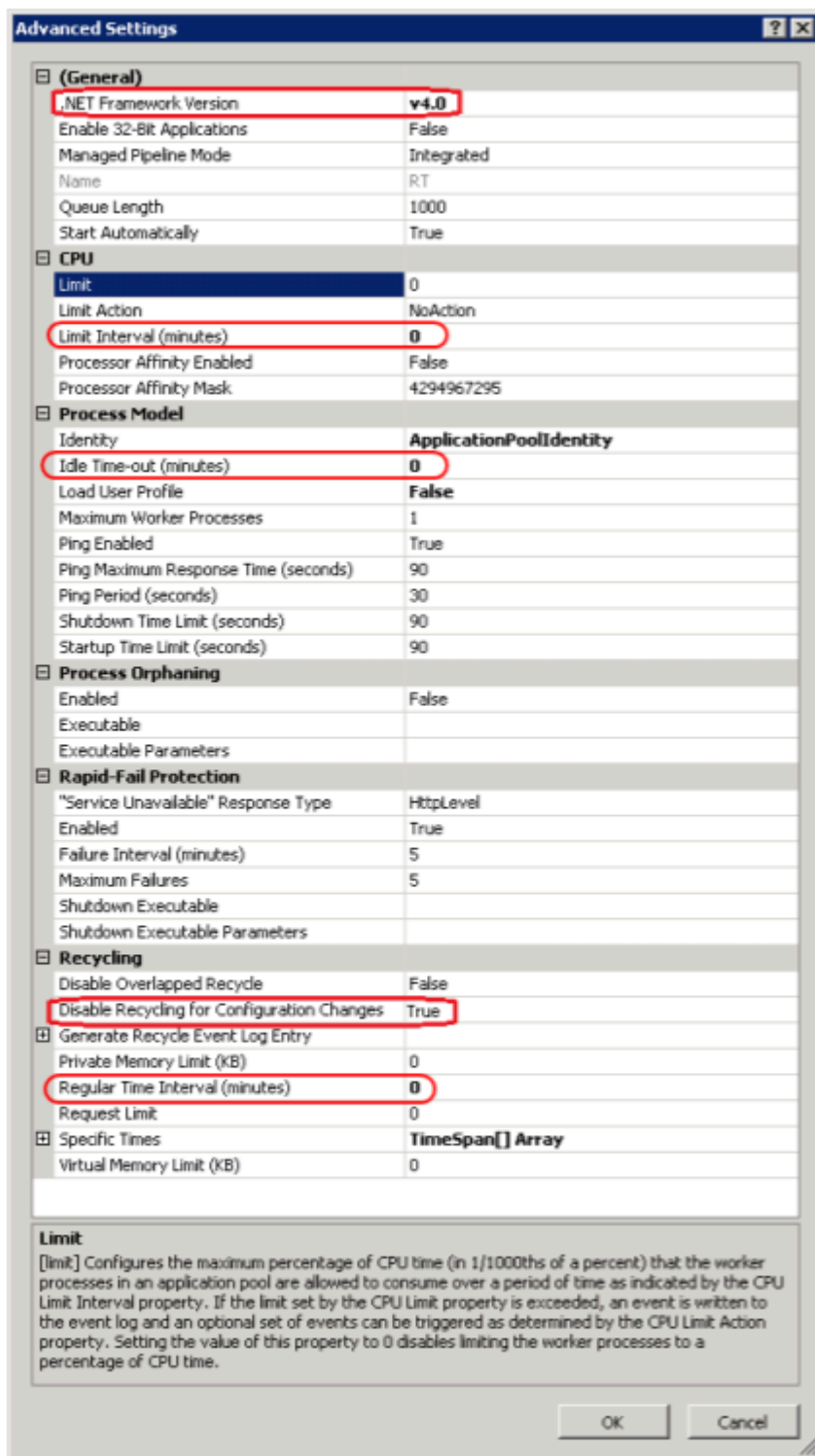


Figura 8- IIS configurazione applicazione

Quindi impostare le seguenti (come da Figura 8):

- General -> .NET Framework Version = v4.0
- CPU -> Limit Interval (minutes) / Intervallo limite (minuti) = 0
- Process MODEL -> Idle Time-out (minutes) / Timeout di Inattività (minuti) = 0
- Recycling -> Regular Time Interval (minutes) / Intervallo tempo regolare (minuti) = 0
- Recycling -> Disable Recycling for Configuration Changes / Disabilita riciclo per modifiche di configurazione = TRUE

File di configurazione

Client - file di configurazione *config.json*

File di configurazione della parte client, contenente principalmente il collegamento con il servizio e rabbit

baseUrl

Indirizzo url dove sarà esposto il servizio

rabbitMq

- baseUrl
Url websocket dove viene esposto rabbitmq
- basicAuth
Nome utente (username) e password (password) di rabbitMq, indicati come in un JSON:

```
{ "username": "---", "password": "---" }
```
- vhost
Nome dell'host da utilizzare (**É LO STESSO DEL SERVIZIO**)
- heartBeatCheckSec
Ogni quanto il client deve mandare un messaggio di controllo al servizio per sapere se le code di rabbit sono ancora attive

Client - file di configurazione *zoom.js*



La libreria utilizzata è quella di Dhtmlx, la sua documentazione per lo zoom è disponibile a questo

link https://docs.dhtmlx.com/gantt/desktop_zoom.html

File di configurazione dello zoom, ogni livello di zoom del gantt può essere modificato a proprio piacimento e a proprio rischio e pericolo (il linguaggio utilizzato è javascript)

zoomLevels

- name
Nome dell'unità di misura



Il nome default è utilizzato come discriminante per indicare quale livello di zoom viene utilizzato appena viene aperto il piano di produzione

- scales
- unit
Unità di tempo ("minute", "hour", "day", "week", "quarter", "month", "year")
- step
Indica il raggruppamento dei periodi: se all'unit month mettiamo uno step di 3, avremo dei mesi raggruppati

a trimestre; se invece lo step lo metteremo a 1, avremo visibilità mensile e così via

08 Gen	09 Gen	10 Gen	11 Gen	12 Gen	13 Gen	14 Gen	15 Gen	16 Gen	17 Gen	18 Gen	19 Gen	20 Gen	21 Gen	22 Gen	23 Gen	24 Gen	25 Gen	26 Gen	27 Gen

Step 1 giorni

Dicembre 2024			Gennaio 2025					Febbraio 2025					Marzo 2025				Aprile 2025		
09 - 15	16 - 22	23 - 29	30 - 05	06 - 12	13 - 19	20 - 26	27 - 02	03 - 09	10 - 16	17 - 23	24 - 02	03 - 09	10 - 16	17 - 23	24 - 30	31 - 06	07 - 13	14 - 20	21 - 27

Step 1 mesi

Step 5 minuti

Nov 2024 - Gen 2025								Lug 2025 - Set 2025								Apr 2			
Nov	Dic	Gen	Feb	Mar	Apr	Mag	Giu	Lug	Ago	Set	Ott	Nov	Dic	Gen	Feb	Mar	Apr	Mag	Giu

Step 3 mesi

Step 1 mese

- format
- Indica come dovrà essere descritto il periodo. Si può inserire una qualsiasi descrizione per il formato utilizzato, di seguito è indicato quali sono i codici utilizzati per le date e orari

Proprietà	Note
%y	L'anno come numero a due cifre (00 a 99)
%Y	L'anno come numero a quattro cifre (1900-9999)
%m	Il mese come numero con zero iniziale (01 a 12)
%n	Il mese come numero senza zero iniziale (1 a 12)
%M	Il mese come abbreviazione (Gen a Dic)
%F	Il mese come nome completo (Gennaio a Dicembre)
%W	Il numero della settimana ISO-8601 dell'anno. Le settimane iniziano il lunedì
%w	Il numero della settimana, le settimane iniziano il lunedì o la domenica a seconda del valore della proprietà di configurazione start_on_monday
%d	Il giorno come numero con zero iniziale (01 a 31)
%j	Il giorno come numero senza zero iniziale (1 a 31)
%D	Il giorno come abbreviazione (Dom a Sab)

Proprietà	Note
%l	Il giorno come nome completo (Domenica a Sabato)
%h	L'ora basata sul formato a 12 ore (00 a 11)
%H	L'ora basata sul formato a 24 ore (00 a 23)
%g	L'ora basata sul formato a 12 ore senza zero iniziale (1 a 12)
%G	L'ora basata sul formato a 24 ore senza zero iniziale (0 a 23)
%i	Il minuto come numero con zero iniziale (00 a 59)
%s	Il secondo come numero con zero iniziale (00 a 59)
%a	Visualizza am (per gli orari da mezzanotte a mezzogiorno) e pm (per gli orari da mezzogiorno a mezzanotte)
%A	Visualizza AM (per gli orari da mezzanotte a mezzogiorno) e PM (per gli orari da mezzogiorno a mezzanotte)

Server - File di configurazione *appsettings.json*

LogServices

Indica le configurazioni che verranno utilizzate per il logging dei messaggi, la configurazione accetta più elementi (è un array)

> Type

Nome del servizio che verrà utilizzato (per ora si utilizza Elastic)

> Level

Livello minimo a partire dal quale i messaggi destinati ai servizi di logging verranno rilasciati.

Ci sono diversi livelli di messaggi:

- **DEBUG:**
in questo livello vengono segnati tempi di esecuzione, quanti elementi sono stati eseguiti etc. questo livello è il più basso e in quanto tale vengono segnati dei messaggi che hanno una bassa importanza.
- **INFO:**
in questo livello vengono segnalate quali chiamate sono state eseguite, in questo livello i messaggi sono tra i più frequenti perchè danno delle informazioni su cosa sta venendo eseguito
- **WARNING:**
i messaggi inoltrati in questo livello segnalano un problema, non bloccante, ovvero che nonostante il problema l'esecuzione terminerà in modo controllato, lato server.

Di solito i messaggi mostrati sono: elemento non trovato, elemento non previsto, sessione scaduta o inesistente.

▸ **ERROR:**

quando si presenta un messaggio error, viene indicato che l'esecuzione di una qualsiasi operazione è stata interrotta bruscamente / in modo non prevista dal sistema.

Questi e i messaggi Fatal rientrano nella categoria degli errori bloccanti.

▸ **FATAL:**

questi messaggi sono tra i più importanti che il sistema può generale, perchè indicano che si è presentata una situazione irreparabile e un riavvio del processo è obbligatorio per poter tornare ad una condizione stabile del sistema.

> **Sender**

Nome di chi sta mandando il messaggio, di base si utilizza il nome dell'istanza per riuscire a filtrare subito i messaggi da diverse macchine

> **Logger**

Usato per Elastic

> **BaseIndexName**

Base dell'indirizzo che devono seguire i messaggi

> **Elastic_url**

Url per l'endpoint di Elastic

ConnectionStrings

Configurazione per la connessione verso il database

> **sqlConnection**

Connessione verso sql, sono richiesti (separati da punti e virgola ;):

- Server: indirizzo del server di destinazione
- Database: nome del Database (che nel caso non esiste verrà creato)
- User: Nome utente (Id) e password (Password) scritte in chiaro separate da punto e virgola (;)
- TrustServerCertificate: se bisogna considerare affidabile il certificato del server (True o False)

ProductionPlan

Configurazione per il piano di produzione

> **durationMin**

Durata minima del piano di produzione espressa in minuti (default 30)

Session

Configurazione delle sessioni

> **durationMin**

Durata validità della sessione espressa in minuti (default 60)

RabbitMq

Configurazione dell'mqtt rabbitMq

> HostName

Nome dell'host di rabbit

> Port

Porta utilizzata da rabbit

> UserName

Nome utente (stesso nome utente utilizzato per entrare nel gestionale di rabbit)

> Password

Password utente (stessa password utilizzata per entrare nel gestionale di rabbit)

> VirtualHost

Nome dell'host virtuale che userà rabbit, è il nome verso cui i messaggi verranno instradati e tramite i quali si possono filtrare a chi appartengono determinate code, exchange etc.

Di solito si usa il nome del proprietario della macchina oppure il nome del cliente

> ClientName

Nome del client

Schedulazione



Sono richiesti i permessi di "gantt.scheduling"

Il sistema implementa un insieme di regole sofisticate per il posizionamento temporale delle fasi e utilizza indicatori visivi per segnalare potenziali problemi o violazioni dei vincoli. Queste regole assicurano che la pianificazione rispetti i vincoli temporali e le dipendenze tra le fasi, mentre gli indicatori visivi permettono una rapida identificazione delle situazioni critiche.

La schedulazione viene eseguita in ordine di della priorità assegnata ad esse, nel caso si avessero delle righe d'ordini all'interno di un ordine che presentano tutte la stessa priorità, verrà eseguita la schedulazione prima sulla riga d'ordine che ha scadenza più vicina, altrimenti verranno schedate in base alla data di creazione delle righe d'ordine

Vincoli di Posizionamento delle Fasi

Il posizionamento delle fasi segue regole precise che considerano diversi fattori temporali. Le fasi possono trovarsi in due stati principali: **bloccate** o **non bloccate**. Le fasi bloccate rappresentano punti fissi nella pianificazione, mentre quelle non bloccate vengono posizionate automaticamente dal sistema secondo

specifiche regole temporali.

Per le fasi non bloccate, il sistema determina il posizionamento considerando due vincoli fondamentali:

- La data odierna come primo punto possibile di inizio
- La data di arrivo del materiale, quando specificata, come ulteriore vincolo di partenza



Il sistema preferirà sempre seguire la realtà piuttosto che rispettare le scadenze. Perciò, se una riga d'ordine ha una data di consegna nel passato oppure la somma delle durate delle fasi risulta eccedere alla data di scadenza, il sistema non posizionerà mai la fase nel passato e nemmeno in anticipo rispetto alla data di consegna materiale, in quanto una fase non può essere eseguita se non dispone del materiale necessario. Ciò ovviamente non si applica alle fasi bloccate, in quanto non possono essere mosse dal sistema

Indicatori Visivi di Stato

Il sistema utilizza un sistema di colorazione per evidenziare situazioni che richiedono attenzione. Questo approccio permette una rapida identificazione visiva di potenziali problemi nella pianificazione:

1. Segnalazione Ritardi (Rosso)

Quando anche una sola fase del ciclo supera la data di consegna prevista, il sistema applica una colorazione rossa a tutte le fasi del ciclo. Questo segnale visivo immediato indica che l'intero ciclo è in ritardo nella consegna.

2. Vincoli Materiali (Giallo)

Se una qualsiasi fase viene posizionata in un periodo antecedente alla data di arrivo del materiale, tutte le fasi del ciclo vengono evidenziate in giallo. Questa colorazione segnala un potenziale problema di disponibilità dei materiali necessari.

E' possibile schedulare in due modi diversi:

1. **Schedulazione manuale:** la schedulazione manuale è a carico dell'utente, dove può prendere le fasi e spostarle singolarmente come meglio crede, utilizzando gli strumenti di auto sposta avanti-indietro, evita collisioni etc...
2. **Schedulazione automatica:** la schedulazione automatica permette di spostare tutte le fasi di un ordine o di tutti gli ordini, in base al ridosso delle scadenze oppure in anticipo rispetto alle scadenze. La schedulazione automatica si divide in:

Schedulazione ASAP

La schedulazione ASAP implementa una strategia di pianificazione immediata, mirando a iniziare la produzione il prima possibile. Quando l'utente attiva questa modalità attraverso l'apposito pulsante, il sistema:

- Identifica tutte le fasi non bloccate nelle commesse non evase
- Inizia la pianificazione dalla data odierna e subito dopo la data di consegna materiale (se presente)
- Posiziona le fasi rispettando la sequenza definita nel ciclo produttivo

Il processo di schedulazione ASAP segue un principio di avanzamento lineare nel tempo, dove:

- Le fasi vengono posizionate in sequenza, rispettando l'ordine definito nel ciclo

- Ogni fase viene collocata nel primo slot temporale disponibile
- Il sistema considera vincoli di risorse e calendari durante il posizionamento

Vantaggi schedulazione ASAP:

- Inizio immediato della produzione
- Identificazione precoce di potenziali ritardi
- Maggiore buffer temporale per gestire imprevisti

Schedulazione JIT

La schedulazione JIT rappresenta un approccio più sofisticato, che pianifica la produzione a ritroso partendo dalla data di consegna. Quando l'utente seleziona questa modalità, il sistema:

- Parte dalla data di consegna della commessa
- Posiziona le fasi procedendo all'indietro nel tempo
- Mantiene la sequenza corretta delle fasi del ciclo



Meccanismo di Fallback JIT → ASAP

La schedulazione JIT include un meccanismo di sicurezza che passa automaticamente alla modalità ASAP quando si verificano determinate condizioni critiche

- **Data di Scadenza nel Passato:**
Se la data di consegna è anteriore alla data odierna, il sistema passa automaticamente alla schedulazione ASAP, poiché non è possibile pianificare nel passato.
- **Vincoli di Fasi Bloccate:**
Quando una fase bloccata si trova dopo una fase mobile nella sequenza, il sistema passa a ASAP per garantire la coerenza temporale della produzione.
- **Insufficienza Temporale:**
Se la somma delle durate delle fasi supera il tempo disponibile fino alla data di consegna, il sistema adotta la strategia ASAP per garantire una pianificazione realizzabile.

Questo meccanismo di fallback assicura che:

- La pianificazione rimanga sempre realizzabile
- Non si creino sequenze temporalmente impossibili
- Si mantenga la coerenza del piano produttivo

Vantaggi schedulazione JIT:

- Minimizzazione delle scorte
- Ottimizzazione dei tempi di stoccaggio
- Migliore gestione dello spazio in magazzino

Capacità Infinita

La modalità di capacità infinita rappresenta un approccio speciale alla schedulazione che permette una pianificazione semplificata, focalizzata esclusivamente sulla sequenza temporale delle fasi senza considerare i vincoli di capacità delle risorse. Questa modalità offre una visione ideale della sequenza produttiva, utile in particolari scenari di pianificazione.

Quando viene attivato il flag "capacità infinita", il sistema modifica il suo comportamento di schedulazione in modo significativo. Il processo di pianificazione mantiene le regole fondamentali di sequenza e temporalità, ma ignora completamente i limiti di capacità delle risorse. Questo significa che le fasi verranno posizionate seguendo esclusivamente:

- Il rispetto della sequenza definita nel ciclo produttivo
- La data odierna come punto di partenza minimo
- La data di consegna del materiale come vincolo iniziale
- Lo stato di blocco delle fasi (bloccate vs non bloccate)

Il sistema **non** considera:

- La disponibilità effettiva delle risorse
- I limiti di capacità produttiva
- Le possibili sovrapposizioni di utilizzo delle risorse



Attenzione, utilizzare questo tipo di schedulazione avrà come risultato un posizionamento di fasi che:

Non riflette le reali limitazioni produttive;

Può generare piani non realizzabili nella pratica;

Richiede successivi aggiustamenti per l'implementazione reale;

Gestione delle Sequenze e Violazioni

Il sistema implementa una logica particolare per gestire situazioni dove la sequenza delle fasi viene compromessa. In particolare, quando si verifica una situazione dove esistono fasi non bloccate con sequenza minore (che dovrebbero quindi essere eseguite prima) e hanno una fase successiva invece bloccata (sempre in un periodo antecedente rispetto a quello odierno). In questo caso le fasi non bloccate verranno posizionate all'orario e al giorno corrente

Questo riposizionamento automatico serve a segnalare una violazione della sequenza naturale del ciclo produttivo. È un meccanismo di sicurezza che evidenzia situazioni dove la sequenza pianificata delle operazioni non è stata rispettata, permettendo agli operatori di intervenire e valutare le azioni correttive necessarie.

Importazione

Le importazioni, in base a quale elemento viene selezionato, hanno effetti a catena con gli altri elementi disponibili. Di seguito viene indicati quali elementi vengono importati in base a quale elemento viene importato in origine e in che ordine.

Cicli:

1. calendar
2. calendar entry
3. calendar time period
4. calendar override
5. calendar time period override
6. calendar global
7. calendar time period global
8. calendar resource
9. resource
10. cycle
11. jobcycle
12. jobcycle resource

Articoli:

1. calendar
2. calendar entry
3. calendar time period
4. calendar override
5. calendar time period override
6. calendar global
7. calendar time period global
8. calendar resource
9. resource
10. cycle
11. jobcycle
12. jobcycle resource
13. product
14. product cycle

Ordini:

1. calendar
2. calendar entry
3. calendar time period
4. calendar override
5. calendar time period override
6. calendar global
7. calendar time period global

8. calendar resource
9. resource
10. cycle
11. jobcycle
12. jobcycle resource
13. product
14. product cycle
15. order group
16. order
17. order row
18. job
19. schema colori

Risorse:

1. calendar
2. calendar entry
3. calendar time period
4. calendar override
5. calendar time period override
6. calendar global
7. calendar time period global
8. calendar resource
9. resource

Calendari:

1. calendar
2. calendar entry
3. calendar time period
4. calendar override
5. calendar time period override
6. calendar global

Calendari globali:

1. calendar global
2. calendar time period global

Schema Colori:

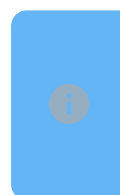
1. tag group

2. tag
3. tag color

Viste

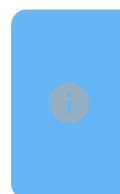
Qui di seguito sono indicate tutte le viste utilizzate all'interno di eTabula, sono obbligatorie per una corretta funzionalità del programma

Legenda



PrimaryKey

Questa proprietà è l'id della tabella, non può essere replicata in tutta la vista per evitare conflitti in fase di importazione



ForeignKey -> ***

Questa proprietà è l'id esterno di una tabella di riferimento, la tabella a cui fa riferimento è indicata a destra della freccia

ViewCalendar

Nome	Tipo	Note
OuterId_Calendar	nvarchar(500)	PrimaryKey
Name_Calendar	nvarchar(500)	Nome calendario
ValidFrom_Calendar	datetimeoffset	Inizio validità del calendario
ValidTo_Calendar	datetimeoffset	Fine validità del calendario

ViewCalendar_Resource

Nome	Tipo	Note
OuterId_CalendarResource	nvarchar(500)	PrimaryKey
OuterId_Resource	nvarchar(500)	ForeignKey -> ViewResource Chiave esterna della vista risorse
OuterId_Calendar	nvarchar(500)	ForeignKey -> ViewCalendar Chiave esterna della vista calendari

ViewCalendarEntry

Nome	Tipo	Note
OuterId_CalendarEntry	nvarchar(500)	PrimaryKey
OuterId_Calendar	nvarchar(500)	ForeignKey -> ViewCalendar Riferimento al calendario originale
DayOfWeek_CalendarEntry	int	Giorno della settimana di riferimento (0=domenica, 1=lunedì, 2=martedì, 3=mercoledì, 4=giovedì, 5=venerdì, 6=sabato)

ViewCalendarOverride

Nome	Tipo	Note
OuterId_CalendarOverride	nvarchar(500)	PrimaryKey
OuterId_Calendar	nvarchar(500)	ForeignKey -> ViewCalendar Riferimento al calendario originale dove applicare l'override del giorno
OverrideDate_CalendarOverride	date	Giorno dove impostare l'override (YYYY-MM-DD)

ViewGlobalCalendar

Nome	Tipo	Note
OuterId_GlobalCalendar	nvarchar(500)	PrimaryKey
OverrideDate_GlobalCalendar	date	Giorno dove impostare l'override globale (applicato su tutti i calendari) (YYYY-MM-DD)

ViewTimePeriod

Nome	Tipo	Note
OuterId_TimePeriod	nvarchar(500)	PrimaryKey
OuterId_CalendarEntry	nvarchar(500)	ForeignKey -> ViewCalendarEntry Calendar entry collegato

Nome	Tipo	Note
OuterId_CalendarOverride	nvarchar(500)	ForeignKey -> ViewCalendarOverride Calendar override collegato
OuterId_GlobalCalendar	nvarchar(500)	ForeignKey -> ViewGlobalCalendar Global calendar collegato
StartAt_TimePeriod	time	Inizio del periodo
FinishAt_TimePeriod	time	Fine del periodo
Capacity_TimePeriod	int	Capacità del periodo (quante fasi ci possono stare contemporaneamente)

ViewCycle

Nome	Tipo	Note
OuterId_Cycle	nvarchar(500)	PrimaryKey
Name_Cycle	nvarchar(500)	Nome del ciclo

ViewJobCycle

Nome	Tipo	Note
OuterId_JobCycle	nvarchar(500)	PrimaryKey
OuterId_StandardResource	nvarchar(500)	ForeignKey -> ViewResource Risorsa standard associata
OuterId_NextJobCycle	nvarchar(500)	ForeignKey -> ViewJobCycle Prossima fase ciclo associata (nullable)
OuterId_Cycle	nvarchar(500)	ForeignKey -> ViewCycle Ciclo associato
Name_JobCycle	nvarchar(500)	Nome della fase ciclo
StandardCycleTime_JobCycle	int	Tempo ciclo standard
Sequence_JobCycle	int	Numero di sequenza (univoco a parità di ciclo)

ViewJobCycle_Resource

Nome	Tipo	Note
OuterId_JobCycleResource	nvarchar(500)	PrimaryKey
OuterId_Resource	nvarchar(500)	ForeignKey -> ViewResource Risorsa associata
OuterId_JobCycle	nvarchar(500)	ForeignKey -> ViewJobCycle Fase ciclo associata
UnitTimeCycle_JobCycleResource	float	Tempo ciclo unitario in secondi per pezzo
SetupTime_JobCycleResource	float	Tempo di setup della macchina in secondi
SmallPillowTime_JobCycleResource	float	Tempo cuscinetto in secondi

ViewMes

Nome	Tipo	Note
OuterId_MES	nvarchar(500)	PrimaryKey
OuterId_Job	nvarchar(500)	ForeignKey -> ViewJob Chiave Job (alternativa a OrderRow e JobCycle)
OuterId_OrderRow	nvarchar(500)	ForeignKey -> ViewOrder Chiave riga ordine (alternativa a Job)
OuterId_JobCycle	nvarchar(500)	ForeignKey -> ViewJobCycle Chiave fase ciclo (alternativa a Job)
ResourceSync_MES	nvarchar(500)	ForeignKey -> ViewResource Chiave per cambio risorsa (opzionale)
NomeFase_MES	nvarchar(500)	Nuovo nome della fase (opzionale)
FaseStato_MES	int	Nuovo stato fase (0=Pronto, 1=Iniziato, 2=Finito)
DataInizioSync_MES	datetimeoffset	Nuova data inizio fase (opzionale)
DataFineSync_MES	datetimeoffset	Data fine fase (opzionale)
DurataSync_MES	int	Nuova durata fase (ignorata se date specificate)

<i>Nome</i>	<i>Tipo</i>	<i>Note</i>
Avanzamento_MES	float	Valore avanzamento fase (opzionale)
Tooltip_MES	nvarchar(500)	Commenti JSON formato [{"Nome Elemento":"Valore Elemento"}]
Tags_MES	nvarchar(500)	Colori JSON formato [{"Chiave gruppo":1,"VALORE":"Nome del Tag"}]

ViewOrder

<i>Nome</i>	<i>Tipo</i>	<i>Note</i>
OuterId_OrderRow	nvarchar(500)	PrimaryKey Chiave primaria riga ordine
OuterId_Order	nvarchar(500)	PrimaryKey Chiave primaria ordine
OuterId_Product	nvarchar(500)	ForeignKey -> ViewProduct Chiave prodotto associato
OuterId_OrderGroup	nvarchar(500)	ForeignKey -> ViewOrderGroup Chiave gruppo ordine (opzionale)
Name_OrderRow	nvarchar(500)	Nome riga ordine
Expire_OrderRow	datetimeoffset	Data di consegna
PreferredStartDate_OrderRow	datetimeoffset	Data inizio fasi (default=oggi)
MaterialArrivalDate_OrderRow	datetimeoffset	Data arrivo materiale (opzionale)
Quantity_OrderRow	int	Quantità commessa
Priority_OrderRow	int	Priorità schedulazione
MeasureUnit_OrderRow	nvarchar(500)	Unità di misura commessa
ExecuteScheduling_OrderRow	int	Tipo schedulazione (0=nessuna, 1=JIT, 2=ASAP)
InfinityCapacityScheduling_OrderRow	bit	Schedulazione a capacità infinita
Name_Order	nvarchar(500)	Nome ordine

ViewOrderGroup

<i>Nome</i>	<i>Tipo</i>	<i>Note</i>
OuterId_OrderGroup	nvarchar(500)	PrimaryKey
OuterId_OrderGroup_Parent	nvarchar(500)	ForeignKey -> ViewOrderGroup Gruppo ordine padre (opzionale)
Name_OrderGroup	nvarchar(500)	Nome gruppo ordine

ViewProduct

<i>Nome</i>	<i>Tipo</i>	<i>Note</i>
OuterId_Product	nvarchar(500)	PrimaryKey
Name_Product	nvarchar(500)	Nome prodotto
ProductCode_Product	nvarchar(500)	Codice del prodotto
Description_Product	nvarchar(500)	Descrizione del prodotto
MeasureUnit_Product	nvarchar(500)	Unità di misura del prodotto

ViewProduct_Cycle

<i>Nome</i>	<i>Tipo</i>	<i>Note</i>
OuterId_ProductCycle	nvarchar(500)	PrimaryKey
OuterId_Product	nvarchar(500)	ForeignKey -> ViewProduct Chiave esterna prodotti
OuterId_Cycle	nvarchar(500)	ForeignKey -> ViewCycle Chiave esterna cicli
IsStandard_Cycle	bit	Indica ciclo standard (un solo ciclo standard per prodotto)

ViewResource

<i>Nome</i>	<i>Tipo</i>	<i>Note</i>
OuterId_Resource	nvarchar(500)	PrimaryKey

Nome	Tipo	Note
Name_Resource	nvarchar(500)	Nome risorsa
Productivity_Resource	float	Produttività risorsa (percentuale, 0=0%, 1=100%)

ViewTag

Nome	Tipo	Note
OuterId_Tag	nvarchar(500)	PrimaryKey
OuterId_TagGroup	nvarchar(500)	ForeignKey -> ViewTagGroup Chiave esterna gruppo tag
Name_Tag	nvarchar(500)	Nome tag

ViewTagGroup

Nome	Tipo	Note
OuterId_TagGroup	nvarchar(500)	PrimaryKey
Name_TagGroup	nvarchar(500)	Nome gruppo tag
Key_TagGroup	nvarchar(500)	Chiave del gruppo

Basato su [Wiki.js](#)