

Robotyc

Manuale utente del server

v 1.0

Sommario

Sommario	2
1 Introduzione	3
2 Una breve panoramica	4
2.1 Il Builder	4
2.1.1 Area di lavorazione	4
2.1.2 Progetti controller	4
2.1.3 Progetti di automazione	5
2.2 Il Controller.....	5
2.3 Il Player	5
3 Il server	6
3.1 La Dashboard	6
3.2 Repository	7
3.2.1 Caricamento di un nuovo progetto	8
3.2.2 Caricamento di un progetto modificato.....	10
3.3 Gestione siti e controllers	10
3.3.1 Configurazione del controller.....	11
3.3.2 Configurazione delle attività	11
3.4 Monitoraggio di tutti i siti	11
3.5 Monitoraggio sito	12
3.6 Monitoraggio controller	13
3.7 Monitoraggio attività	13
3.8 Monitoraggio esecuzioni	13
3.9 Monitoraggio batch.....	14
3.10 Monitoraggio task	15
4 Indice	16

1 Introduzione

Questo manuale ha lo scopo di descrivere il Robotyca Server che fa parte della suite Robotyca.

La suite è composta da diversi moduli software che elenchiamo e poi descriviamo brevemente perché funzionali alle spiegazioni contenute in questo manuale.

Pertanto, in questo manuale troverete un approfondimento relativo solo al server e solo alcuni concetti relativi agli altri moduli; in ogni caso sarà possibile approfondire i dettagli di questi moduli leggendo i manuali in linea distribuiti per ognuno di loro.

2 Una breve panoramica

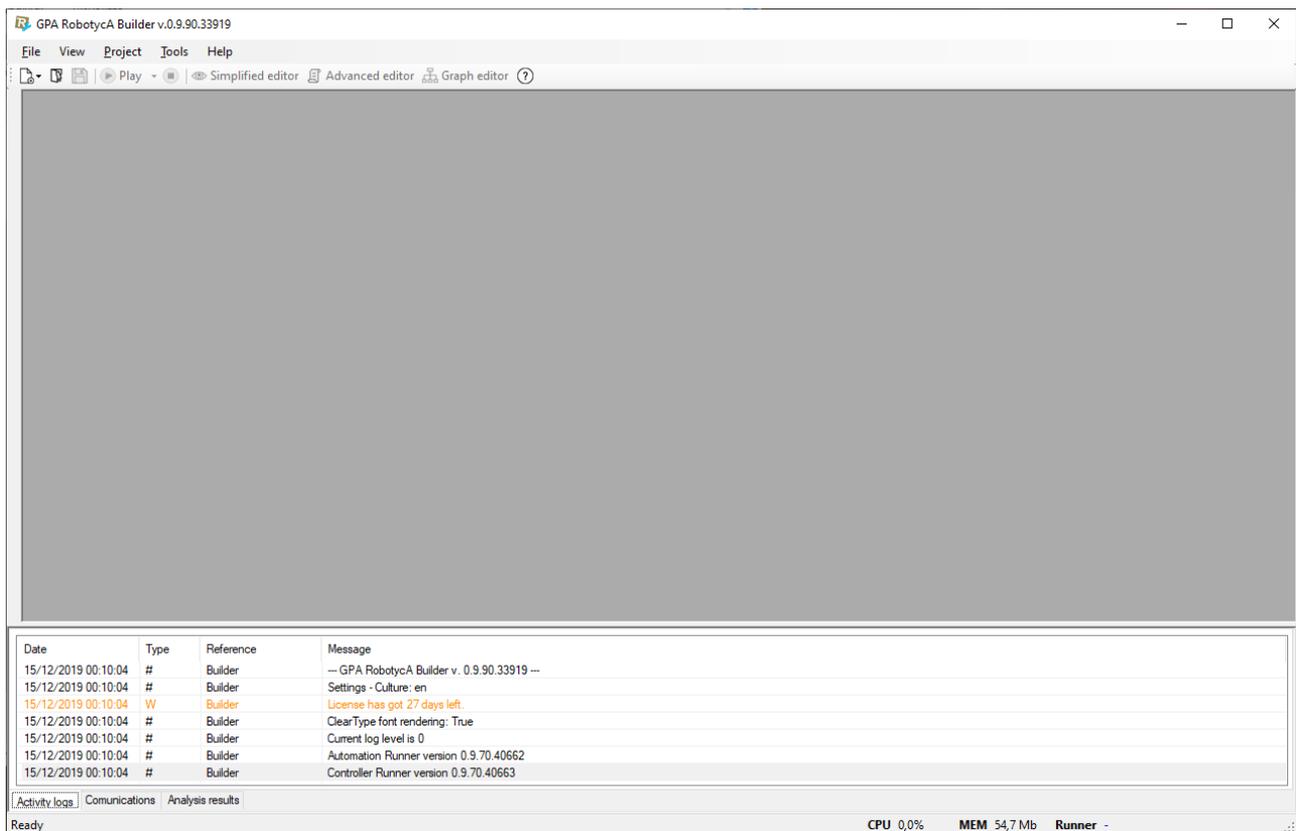
2.1 Il Builder

Il modulo è utilizzato per creare e quindi testare due tipi di progetti:

1. Le automazioni – in cui vengono eseguite in sequenza una serie di operazioni allo scopo di simulare l'interazione dell'utente con uno o più programmi o applicazioni web ed infine produrre il lavoro per cui l'automazione è stata sviluppata.
2. I progetti controller – il cui scopo invece è quello di prendere in ingresso delle informazioni, decidere quali azioni intraprendere e di quali automazioni servirsi e produrre infine l'output richiesto.

PER APPROFONDIMENTI FARE RIFERIMENTO AL MANUALE DEL BUILDER

2.1.1 Area di lavorazione



2.1.2 Progetti controller

I progetti destinati ai controller si compongono invece di tre fasi:

1. INPUT
2. RULE
3. OUTPUT

La prima fase si occupa di veicolare le informazioni estratte da sorgenti che possono essere file oppure dei servizi.

La seconda fase prenderà le informazioni raccolte e le utilizzerà all'interno delle regole (RULES). Esistono due tipi di regole:

1. Generiche
2. Automazioni

A differenza delle regole generiche che permettono semplicemente di direzionare le informazioni ingresso, le regole relative alle automazioni attivano quest'ultime sui player e ne catturano i risultati.

Infine, le fasi di tipo OUTPUT chiudono l'elaborazione producendo uscite di vario tipo, tra cui:

1. Report Pdf
2. Informazioni in ingresso a servizi web esterni
3. Aggiornamenti su base dati
4. Invio eMail

2.1.3 Progetti di automazione

Questi progetti definiscono le azioni che dovranno essere eseguite dai player.

2.2 Il Controller

Esso esegue sostanzialmente una serie di fasi e l'obiettivo è quello di calare le automazioni all'interno di processi più complessi.

Le fasi che costituiscono l'esecuzione effettuata dal controller sono descritte nel paragrafo 2.1.2.

PER ULTERIORI DETTAGLI FARE RIFERIMENTO AL MANUALE IN LINEA DEL CONTROLLER.

2.3 Il Player

Questo è il modulo che si occupa dell'esecuzione delle automazioni; ad ogni controller può essere assegnato uno o più player e ogni player va lanciato su una propria macchina (non è possibile avviare due player sulla stessa macchina).

PER ULTERIORI DETTAGLI FARE RIFERIMENTO AL MANUALE IN LINEA DEL PLAYER.

3 Il server

Il server ha lo scopo principale quello di distribuire i progetti ai controller organizzando il lavoro in:

1. Siti
2. Controller
3. Attività
4. Esecuzioni
5. Batch
6. Task

L'attività di distribuzione è possibile anche perché il server ha anche il ruolo di repository dei progetti. Infatti, dopo aver realizzato un progetto, è possibile direttamente dal Builder caricarlo sul server.

Il progetto appena caricato potrà essere associato ad un controller dove potrà essere utilizzato per eseguire un'attività.

L'attività non è nient'altro che un'istanza eseguibile di un progetto; ossia solo dopo averla definita sarà effettivamente disponibile al controller poter essere eseguita.

Ogni volta che il controller avvia un progetto viene creata un'esecuzione.

3.1 La Dashboard

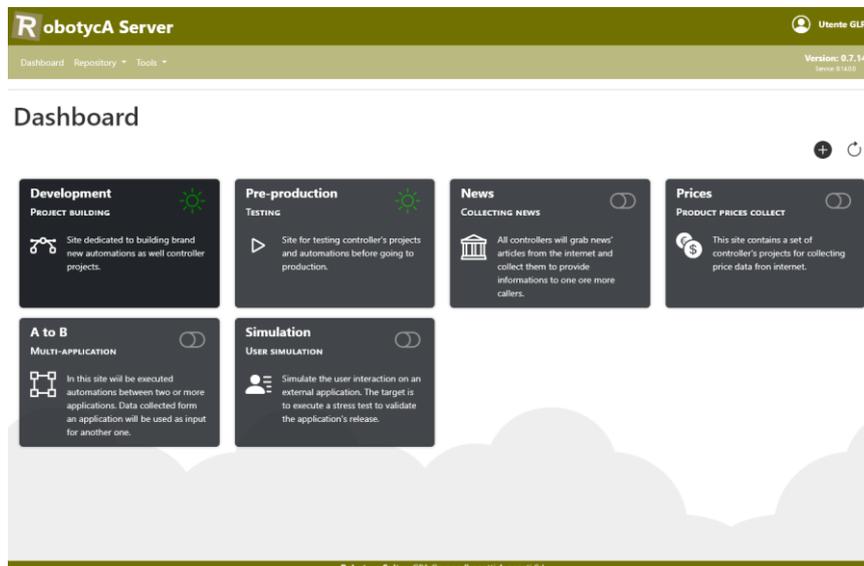


Figura 1 - La dashboard

Qui sopra possiamo identificare nella pagina della dashboard i siti. Ogni sito raccoglie uno o più controller che hanno un compito del medesimo tipo.

Come prima cosa nell'esempio troviamo un primo sito (denominato "Development") che è adibito alla costruzione e al test dei progetti e delle attività definite sui progetti; qui si avrà la possibilità di provare le macchine collegate e calibrarne il funzionamento.

Il secondo esempio "pre-production" è invece un sito destinato ad un test più massivo prima del passaggio in produzione.

Seguono poi degli esempi di siti di produzione.

L'architettura relativa al RobotycA Server può essere sintetizzata:

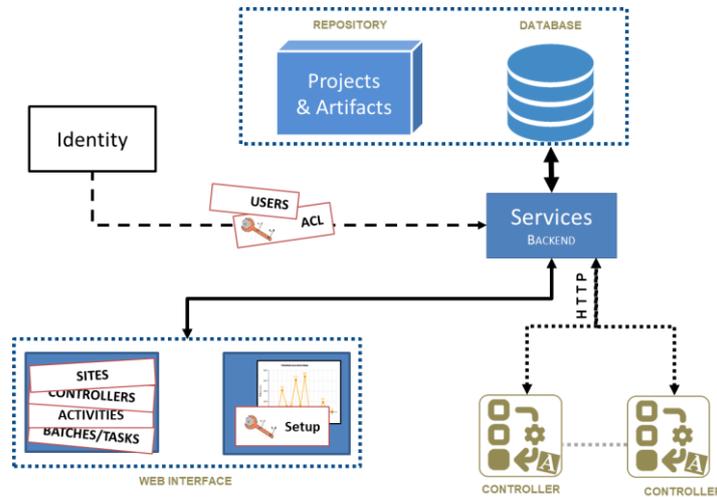


Figura 2 - Architettura generale

3.2 Repository

Il repository dei progetti è alimentato e gestito direttamente dal Builder. Insieme al progetto verranno caricati anche tutti gli artefatti necessari al funzionamento in modo che potranno essere scaricati dal controller che ne fa richiesta.

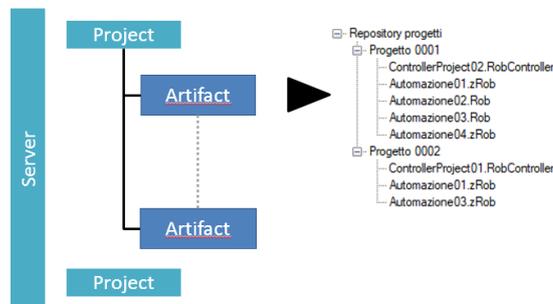


Figura 3 – Repository

Il caricamento dei progetti avviene secondo uno schema ben preciso che riportiamo e commentiamo di seguito:

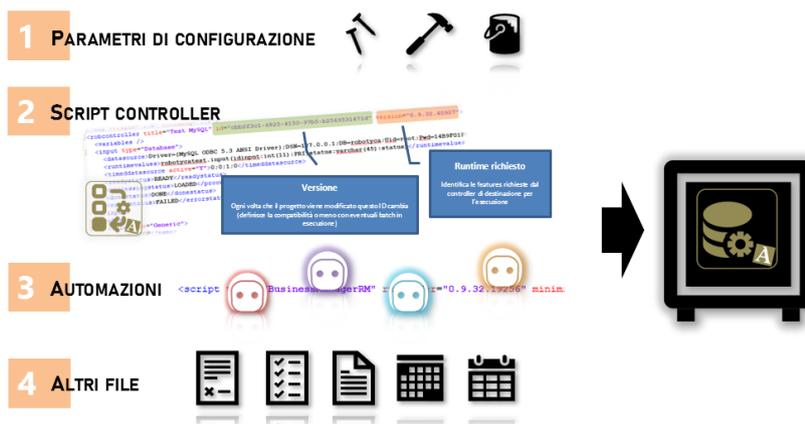


Figura 4 - Caricamento di un progetto

3.2.1 Caricamento di un nuovo progetto

Il caricamento di un nuovo progetto avviene partendo dalle seguenti operazioni:

1. Caricare il progetto nel builder¹
2. Aprire la finestra “Repository”
3. Avviare il caricamento sul server

La seconda parte della configurazione avviene direttamente dall’interfaccia web del server partendo dall’associazione del progetto ad un controller.

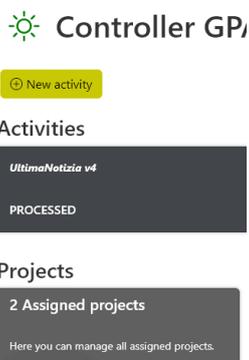


Figura 5 - Gestione dei progetti assegnati ad un controller

Controller Project list
This page shows all projects for the specified controller.
GPMATERBOOKD15

[Link a new project](#)

Id	Status	Description	Actions	Details
PRJ000028	NEW	UltimaNotizia v3 (da rimuovere)	Unlink	Edit
PRJ000029	NEW	UltimaNotizia v4	Unlink	Edit

Figura 6 - Associazione di un progetto

Associato il progetto, si passa alla creazione di una nuova attività che va associata al progetto e configurata.

¹ Qualora il progetto del controller venisse caricato e poi modificato; questo andrà salvato nuovamente prima di poter essere effettuato l’upload sul repository del server.

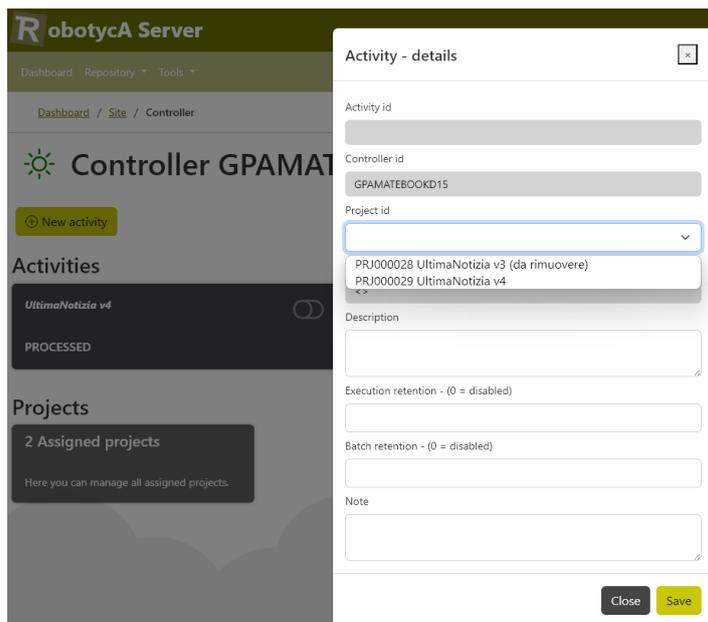


Figura 7 - Creazione di un'attività

La configurazione di un'attività consiste nella definizione delle variabili di progetto per lo specifico controller; questa operazione è necessaria nel caso in cui il progetto abbia bisogno di parametri di input specifici per il controller per cui l'attività è stata creata.

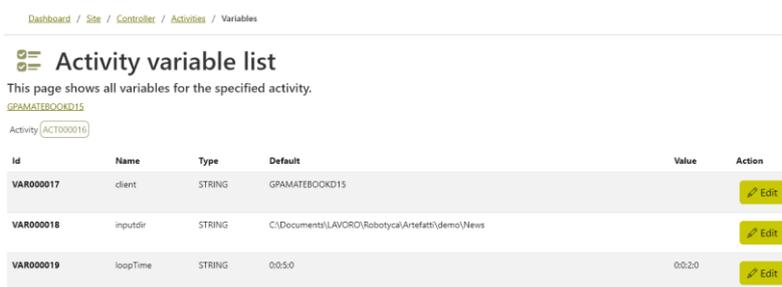


Figura 8 - Configurazione di un'attività

Nella pagina riportata nella figura precedente vediamo l'elenco delle variabili in ingresso relative al progetto associato ad un'attività. I valori possono essere cambiati ove necessario o lasciati con i default assegnati nel progetto.

Nell'esempio è stata cambiata la frequenza di attivazione della lavorazione (trattandosi di un progetto che viene avviato a cadenza regolare). I parametri *INPUTDIR* e *CLIENT* possono essere cambiati qualora la cartella dei file di input si trovasse in un percorso differente oppure se le macchine client su cui vengono eseguiti i player non fosse quella (o quelle separate da un punto e virgola) specificate.

Infine, per permettere ad un'applicazione esterna (detta "Caller") di usufruire dei dati catturati dall'automazione, questo dovrà essere definito.

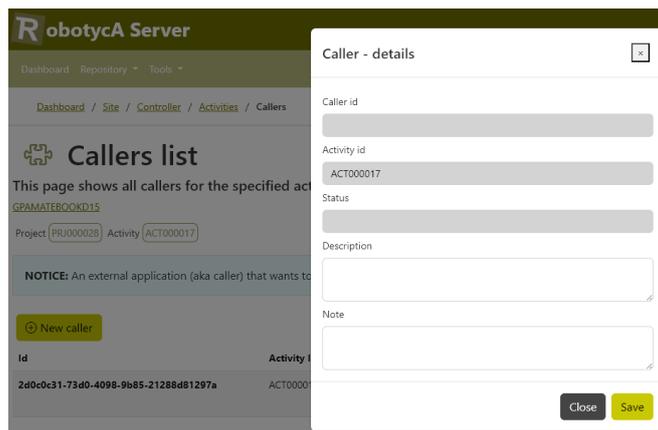


Figura 9 - Creazione di un caller

L'applicazione chiamante a sua volta dovrà effettuare una richiesta per estrarre i dati relativi all'ultima esecuzione specificato il GUID generato alla creazione del profilo del "caller".

Ora l'attività può essere resa "ACTIVE" e quindi utilizzata dal controller.

3.2.2 Caricamento di un progetto modificato

Il caricamento di una nuova versione di un progetto già caricato sul server deve avvenire come se fosse un nuovo progetto; questo garantisce la continuità della lavorazione che nel progetto già caricato potrebbe essere in atto.

Una volta caricato il progetto va creata una nuova attività e i caller che eventualmente sono associati andranno spostati (esiste una funzione di spostamento del caller già attivo che effettua una copia del profilo del caller impostando lo stato "UNACTIVE" per il vecchio caller mentre il nuovo diventa automaticamente "ACTIVE").

3.3 Gestione siti e controllers

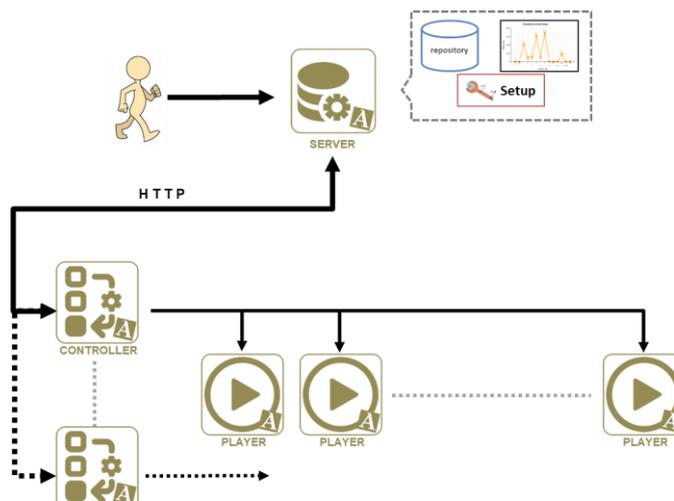


Figura 10 - Siti e controllers

Qui sopra è riportato un esempio di un sito contenente uno o più controller; a loro volta ogni controller potrà avere al suo servizio uno o più player.

3.3.1 Configurazione del controller

Il controller deve essere creato sulla base del nome della macchina; il Robotyca Controller, infatti, si autenticerà sul server specificando sia il sito di appartenenza che il nome macchina.

I controller potranno essere definiti ACTIVE oppure UNACTIVE; nel primo caso il controller potrà collegarsi con il server, nel secondo no.

Per ogni controller è possibile configurare la RETENTION ossia il numero di giorni massimo in cui una voce di log dovrà essere mantenuta sul server. Passato il numero di giorni definito la riga di log verrà rimossa dal server.

The screenshot shows a configuration form for a controller. It includes a status selector with 'ACTIVE' selected and 'UNACTIVE' as an option. Below are three input fields: 'Total connection requests received' (value: 23), 'Last connection timestamp' (value: 2023-07-26T09:25:29.684835+00:0), and 'Last disconnection timestamp'. There is also an 'Assigned token' field with a long alphanumeric string and a 'Log retention days - (0 = disabled)' field with the value 5.

Figura 11 - Parametri configurazione controller

Se la RETENTION non viene definita oppure è ZERO tutte le righe di log verranno mantenute.

3.3.2 Configurazione delle attività

Un'attività definisce i parametri di esecuzione di un progetto per un particolare controller e quando il controller è in esecuzione il suo stato viene aggiornato.

Per essere eseguita dal controller un'attività deve trovarsi nello stato ACTIVE (stato che può essere modificato dal client web a piacimento ma con alcune limitazioni).

SE IMPOSTANDO A **UNACTIVE** UN'ATTIVITÀ IN CUI È PRESENTE UN'ESECUZIONE IN ATTO QUESTA NON VERRÀ INTERROTTA MA CONTINUERÀ FINO A COMPLETAMENTO; LO STATO UNACTIVE IMPEDISCE AD UN CONTROLLER DI INIZIARE UNA NUOVA ESECUZIONE RELATIVA A QUELL'ATTIVITÀ. INFATTI, PER INTERROMPERE UN'ESECUZIONE, SI DOVRÀ INVIARE RICHIESTA DI INTERRUZIONE DIRETTAMENTE ALL'ESECUZIONE STESSA.

3.4 Monitoraggio di tutti i siti

Dopo la definizione delle attività i controller e i player (installati ed avviati sulle varie macchine) inizieranno a lavorare.

A questo punto potremo monitorarne il funzionamento e per questo ritorniamo alla pagina della dashboard.

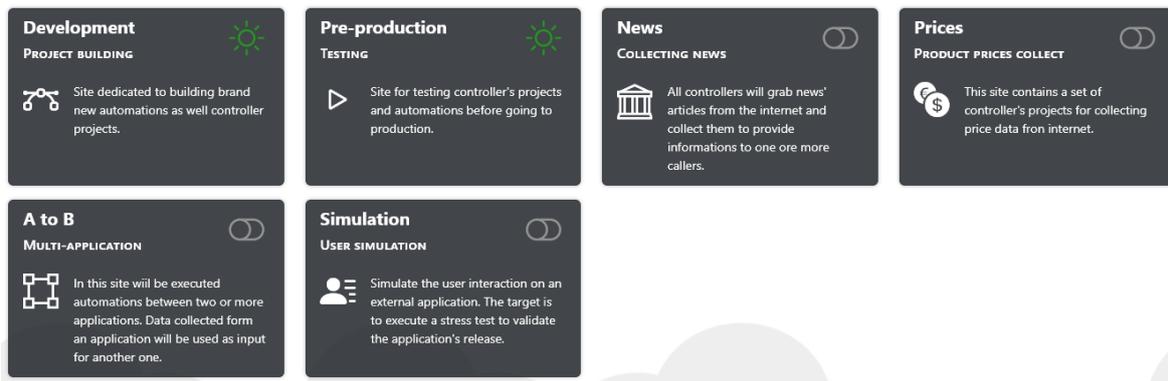


Figura 12 - Monitoraggio dalla dashboard

Direttamente dalla dashboard abbiamo le prime informazioni sui vari siti; infatti qui a seconda dello stato di lavorazione dei controller di ogni sito abbiamo una segnalazione di colore:

1. **VERDE** – quando ogni controller in esecuzione per un particolare sito ha solo esecuzioni terminate con successo
2. **GIALLO** – quando un controller del sito ha più esecuzioni terminate con successo che con errori
3. **ROSSO** – quando un controller del sito ha più esecuzioni terminate con errori che con successo
4. **GRIGIO** – quando tutti i controller non sono in funzione

Qualora il controller di un sito sia nella fase di esecuzione e non abbia ancora terminato la lavorazione corrente verrà visualizzata un’animazione che sta ad indicare il transitorio fino a lavorazione terminata (così come nel caso di un sito non attivo verrà segnalato con un’icona grigia).

Procedendo a selezionare un sito si passa al livello di dettaglio successivo del singolo sito.

3.5 Monitoraggio sito

Il sito contiene uno o più controller e qui abbiamo una rappresentazione per ogni controller dello stato di lavorazione.

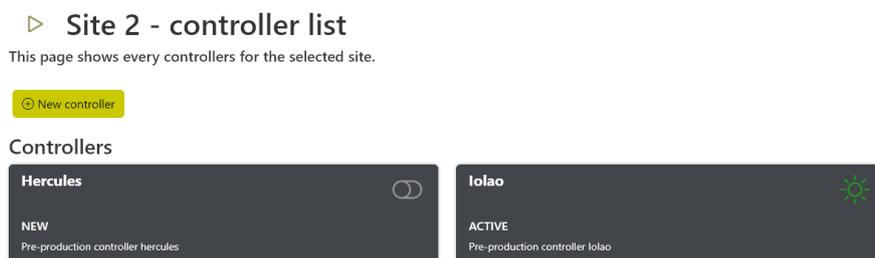


Figura 13 - Monitoraggio di un sito

Nell’esempio precedente vediamo i due controller del sito dove uno è non attivo mentre il secondo è attivo e non presenta anomalie.

Volendo scendere ancora più in dettaglio possiamo passare al livello del singolo controller.

3.6 Monitoraggio controller

Il controller può avere più attività ed in questa visualizzazione viene rappresentato lo stato di ogni attività definita.

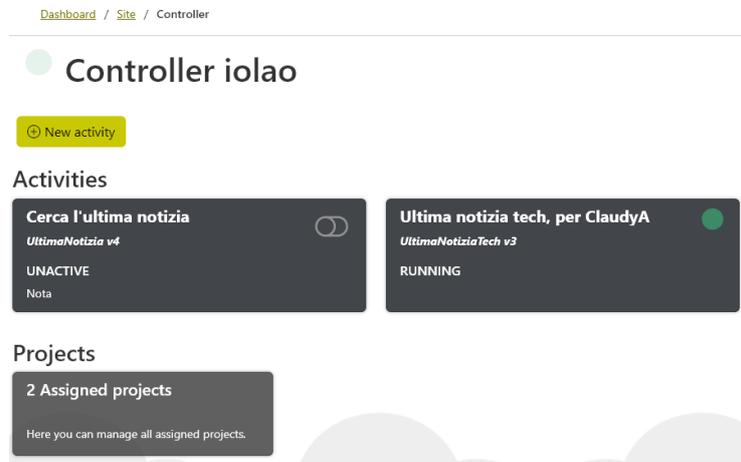


Figura 14 - Monitoraggio di un controller

In questo caso notiamo che la seconda attività è in esecuzione (il che è anche confermato dall’animazione dell’icona nel blocco della seconda attività).

Un’attività è in esecuzione fino al completamento della fase di output e fino a che non termina l’esecuzione del progetto oppure questo non si mette in attesa (nel caso in cui sia un progetto che prevede una pianificazione temporale).

In questa pagina, inoltre, si può accedere all’elenco dei progetti collegati al controller (dove si potrà anche collegare un nuovo progetto o scollegarne uno non più utilizzato).

3.7 Monitoraggio attività

La singola attività può avere più esecuzioni e qui possiamo accedere anche all’elenco delle esecuzioni:

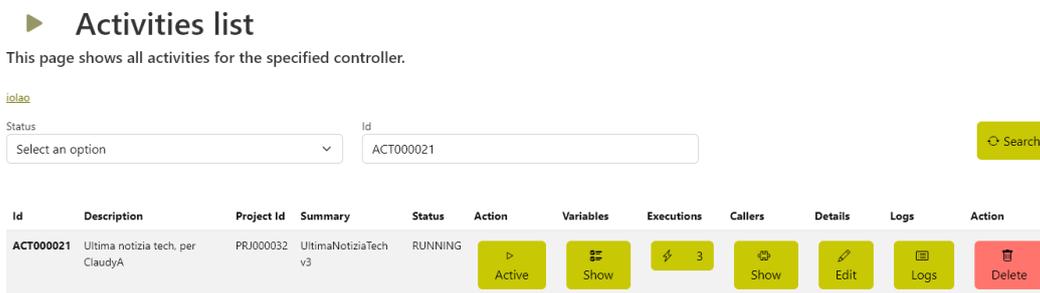


Figura 15 - Monitoraggio di un'attività

La pagina può essere utilizzata anche per monitorare tutte le attività basterà togliere il filtro sull’identificativo dell’attività per visualizzare tutte le attività relative al controller.

3.8 Monitoraggio esecuzioni

Ogni attività può avere da una a molte esecuzioni, una nuova esecuzione viene creata ogni volta che:

1. Si avvia il controller
2. Il controller carica un progetto (ad esempio dopo averne completato l'esecuzione o dopo che questa è stata interrotta)

Execution list
This page shows all execution for the specified project.

Project: PRJ000032 Activity: ACT000021

Status:

Id	Controller	Status	Start	Last update	Elapsed	Batches (ok/tot)	Actions
EXE000461	iolao	RUNNING	26/07/2023 11:44:02:826	1.8 min ago	-	12	<input type="button" value="Stop request"/> <input type="button" value="Batches"/> <input type="button" value="Details"/> <input type="button" value="Delete"/>
EXE000460	iolao	STOPPED	26/07/2023 11:25:34:186	-	16.3 min	1	<input type="button" value="Batches"/> <input type="button" value="Details"/> <input type="button" value="Delete"/>
EXE000427	iolao	STOPPED	25/07/2023 20:26:07:983	-	14.8 hr	10	<input type="button" value="Batches"/> <input type="button" value="Details"/> <input type="button" value="Delete"/>

3 elements Previous 1 Next

Figura 16 - Monitoraggio delle esecuzioni

Per ogni esecuzione troviamo indicati degli utili parametri:

1. Ultimo aggiornamento – è il conteggio del tempo trascorso dall'ultimo aggiornamento ricevuto dal controller (se questo tempo aumentasse oltre un certo limite potrebbe voler dire che ci sia un problema sulla macchina che ospita il controller).
2. Tempo impiegato (Elapsed) – una volta terminata l'esecuzione indica la durata dell'esecuzione.
3. Batches – indica il numero di batch completati con successo; qualora vi fosse stato un errore verranno indicati sia il numero di batch completati con successo che il numero totale di batch (la differenza saranno proprio i batch per i quali si è verificato un problema²).

Inoltre, da questa pagina è anche possibile richiedere l'arresto di un'esecuzione (STOP REQUEST³).

DURANTE LA FASE DI STOP REQUEST IL CONTROLLER CONTINUA A RICEVERE AGGIORNAMENTI DAI PLAYER E AD AGGIORNARE IL SERVER FINO A QUANDO NON VIENE APERTO UN NUOVO PROGETTO.

3.9 Monitoraggio batch

Ogni esecuzione produce un certo numero di batch e in questa pagina è possibile tener traccia dello stato di ciascuno di loro.

² Il parametro di RETENTION (qualora fosse configurato) non agisce sui batch in errore. Questi infatti verranno lasciati per permetterne l'analisi. Una volta non più necessari potranno essere rimossi e la RETENTION verrà di nuovo applicata.

³ Una richiesta di stop viene processata quando nell'esecuzione del progetto controller viene eseguita una nuova iterazione (se il progetto prevede più iterazioni), oppure quando il controller viene chiuso e riaperto.



Figura 17 - Monitoraggio di una esecuzione

Il grafico rappresenta i batch completati con successo, quelli in esecuzione e quelli terminati in errore.

3.10 Monitoraggio task

Scendendo ancora di più in dettaglio abbiamo la rappresentazione del singolo batch dove troviamo i task:



A seconda della struttura del progetto e/o delle informazioni presenti nell'input del progetto potranno essere creati da uno a più task ed ogni task produrrà un suo specifico output.

4 Indice

A	
architettura.....	7
associazione del progetto.....	8
attività.....	8
B	
Builder.....	4
C	
Caller.....	9
Controller.....	5
D	
Dashboard.....	6
M	
Monitoraggio attività.....	13
Monitoraggio batch.....	13
Monitoraggio controller.....	12
Monitoraggio di tutti i siti.....	11
Monitoraggio esecuzioni.....	13
Monitoraggio sito.....	11
Monitoraggio task.....	14
P	
Player.....	5
Progetti controller.....	4
Progetti di automazione.....	5
R	
Repository.....	7
Robotyca Server.....	3
V	
variabili in ingresso.....	9