

## INTEGRAZIONE CUP - SERVIZI ESTERNI

### Specifica dei servizi esposti

Data	Versione	Autore	Descrizione
07/08/2020	5	M. Ubertone La Rocca	
23/09/2020	6	M. Ubertone La Rocca	Attivazione processi post prenotazione tramite la cancellazione del processo di prenotazione
21/10/2020	7	M. Ubertone La Rocca	Aggiornamenti: <ul style="list-style-type: none"> <li>aggiunto servizio Enti per recuperare gli enti attivi su cui l'utenza ha le autorizzazioni</li> <li>aggiunto servizio Sedi per recuperare le informazioni della sede richiesta</li> <li>aggiunto parametro in ricerca disponibilità: online=true</li> </ul>
26/11/2020	8	M. Ubertone La Rocca	Aggiornamenti: <ul style="list-style-type: none"> <li>inseriti filtri prestazioni erogabili sul servizio enti</li> <li>inseriti filtri prestazioni erogabili sul servizio sedi</li> </ul>
11/01/2021	9	M. Ubertone La Rocca	Aggiornamenti: aggiunto servizio Accettazione
15/04/2021	10	M. Ubertone La Rocca	Aggiornamenti: aggiunto servizio Lista Pre-Appuntamenti
29/04/2021	11	M. Ubertone La Rocca	Aggiornamenti: revisione servizio accettazione diretta
03/05/2021	12	M. Ubertone La Rocca	Aggiornamenti: conferma Pre-Appuntamento
03/05/2021	13	M. Ubertone La Rocca	Aggiornamenti: apertura posizione debitoria per accettazione diretta su PagoPa

A.	Servizi Generici	4
A.1.	Autenticazione	4
A.2.	Motivazioni	4
A.3.	Regimi di Erogazione	5
A.4.	Priorità	5
A.5.	Informazioni paziente	6
A.6.	Ricetta dematerializzata	6
A.7.	Enti	7
A.8.	Sede	8
A.9.	Prestazioni	9
A.10.	Liste di lavoro	10
A.11.	Profilo	10
A.12.	Tipologie di Accettazione	11
A.13.	Coperture Assicurative	11
A.14.	Listini	12
A.15.	Dottori	12
A.16.	Unità Eroganti	13
A.17.	Preventivi	14
A.18.	Documenti di Pagamento	14
A.19.	Sedi di Cassa	15
A.20.	Pagamenti	15
B.	Prenotazione	16
B.1.	Ricerca Prima Disponibilità	17
B.2.	Ricerca Disponibilità Successiva	19
B.3.	Lista Strutture e/o ambulatori	20
B.4.	Ricerca Disponibilità	21
B.5.	Conferma Prenotazione	22
B.6.	Annulla Prenotazione in corso	23
B.7.	Pre-prenotazione	24
C.	Consultazione appuntamenti	25
C.1.	Lista appuntamenti	25
C.2.	Annulla Appuntamento	26
C.3.	Sposta Appuntamento	27

C.4.	Ristampa ricevuta di prenotazione	28
D.	Pagamento	29
D.1.	Lista pendenze da pagare	29
D.2.	Paga	30
D.3.	Storico dei pagamenti	32
D.4.	Recupero Ricevuta Fiscale	33
E.	Accettazione	34
E.1.	Conferma Accettazione	38
E.2.	Aggiornamento Accettazione	39
E.3.	Ricerca Accettazione	40
F.	Pre-Appuntamenti	41
F.1	Lista Pre-Appuntamenti	41
F.2	Conferma Pre-Appuntamento	42

## A. Servizi Generici

Nel presente capitolo vengono riportati i servizi utili al recupero delle informazioni necessarie per le funzionalità principali, riportate nei capitoli successivi.

### A.1. AUTENTICAZIONE

L'autenticazione avviene tramite la basic-authentication negli header delle richieste. Il parametro Authorization all'interno dell'header deve essere valorizzato con la Basic Auth contenente username e password dell'utente codificati in base64.

### A.2. MOTIVAZIONI

Il servizio permette di recuperare le motivazioni presenti a sistema.

Servizio	Url	Metodo
Motivazioni	api/v1/uiapi/oms/reason	GET

Passando come parametri

Parametro	Tipo	Data	Descrizione
Authorization	header	string	Basic Auth con username e password codificati in base64
category	query	string	Categoria della motivazione (1=annullamento prenotazione, 3=spostamento prenotazione)

Viene restituito un oggetto contenente una lista delle motivazioni attive.

**Esempio Response (file Response\_Motivazioni.json)**

### A.3. REGIMI DI EROGAZIONE

Il servizio permette di recuperare la lista dei regimi di erogazione disponibili.

Servizio	Url	Metodo
Regimi di erogazione	api/v1/uiapi/oms/supplymode	GET

I parametri da passare sono:

Parametro	Valore	Tipo	Data	Descrizione
Authorization		header	string	Basic Auth con username e password codificati in base64
onlyEnable	Y	query	string	Regimi di erogazione attivi

Viene restituito un oggetto contenente una lista dei regimi di erogazione attivi.

**Esempio Response (file Response\_Regimi\_di\_erogazione.json)**

### A.4. PRIORITÀ

Il servizio permette di recuperare le priorità presenti a sistema.

Servizio	Url	Metodo
Priorità	api/v1/uiapi/oms/priority	GET

I parametri da passare sono:

Parametro	Valore	Tipo	Data	Descrizione
Authorization		header	string	Basic Auth con username e password codificati in base64
enabledOnly	Y	query	string	Priorità attive

Viene restituito un oggetto contenente una lista delle priorità attive.

**Esempio Response (file Response\_Priorità.json)**

## A.5. INFORMAZIONI PAZIENTE

Il servizio permette di recuperare i dati del paziente registrati a sistema effettuando una ricerca per codice fiscale.

Servizio	Url	Metodo
Pazienti	api/v1/uiapi/epr/patient	GET

I parametri da passare sono:

Parametro	Tipo	Data	Descrizione
Authorization	header	string	Basic Auth con username e password codificati in base64
fiscalCode	query	string	Codice fiscale del paziente

Viene restituito un oggetto contenente le informazioni utili del paziente.

### Esempio Response (file Response\_Informazioni\_paziente.json)

## A.6. RICETTA DEMATERIALIZZATA

Il servizio permette di recuperare i dati associati alla ricetta.

Servizio	Url	Metodo
Prescrizioni	api/v1/uiapi/oms/prescription/{nre}	GET

L'{nre} identifica il codice della ricetta per la quale effettuare le ricerca ed il recupero dei dati associati.

I parametri da passare sono:

Parametro	Tipo	Data	Descrizione
Authorization	header	string	Basic Auth con username e password codificati in base64
nre	path	string	Numero ricetta elettronica
fiscalCode	query	string	Codice fiscale del paziente
patientId	query	string	ID del paziente

Dove almeno uno tra fiscalCode o patientId deve essere valorizzato.

Un esempio di caso d'uso è definito dal seguente flusso:

1. Recupero informazioni paziente => Informazioni paziente (paragrafo A.5.)
2. Recupero informazioni della ricetta dematerializzata

Viene restituito un oggetto contenente le informazioni associate alla ricetta.

### Esempio Response (file Response\_Ricetta\_dematerializzata.json)

## A.7. ENTI

Il servizio permette di recuperare gli enti abilitati e autorizzati per l'utente secondo i filtri impostati.

Servizio	Url	Metodo
Enti	api/v1/uiapi/sys/authority	GET

I parametri da passare sono:

Parametro	Tipo	Valore	Data	Descrizione
Authorization	header		string	Basic Auth con username e password codificati in base64
onlyDisconnected	query	N	string	Filtra gli enti non disconnessi dalla gerarchia
onlyAuthorized	query	Y	string	Filtra gli enti per cui l'utente è abilitato
deliverServiceIds	query		Array[string]	Filtro per ricercare solo gli enti aventi sedi con unità eroganti in grado di erogare almeno una tra le prestazioni elencate. Valori ammessi: uno o più ID di prestazioni
allDeliverServiceIds	query		Array[string]	Filtro per ricercare solo gli enti aventi sedi con unità erogante in grado di erogare tutte le prestazioni elencate. Valori ammessi: uno o più ID di prestazioni

### Esempio Response (file Response\_Enti.json)

NB: per recuperare le informazioni di un ente conoscendone l'identificativo è necessario effettuare la stessa chiamata precedente (senza l'uso di query ad esclusione della view) passando nel path l'identificativo corrispondente, esempio: `api/v1/uiapi/sys/authority/{identificativo}`

## A.8. SEDE

Il servizio permette di recuperare sedi abilitati secondo i filtri impostati.

Servizio	Url	Metodo
Sedi	api/v1/uiapi/sys/site	GET

L'{id} è l'identificativo della sede richiesta. I parametri da passare sono:

Parametro	Tipo	Valore	Data	Descrizione
Authorization	header		string	Basic Auth con username e password codificati in base64
authorityIds	query		Array[string]	Uno o più ID di enti
servicelds	query		Array[string]	Filtro per ricercare solo le sedi con unità erogante in grado di erogare almeno una tra le prestazioni elencate. Valori ammessi: uno o più ID di prestazioni
allServicelds	query		Array[string]	Filtro per ricercare solo le sedi con unità erogante in grado di erogare tutte le prestazioni elencate. Valori ammessi: uno o più ID di prestazioni
view	query	normal	string	Criterio di verbosità

### Esempio Response (file Response\_Sede.json)

**NB:** per recuperare le informazioni di una sede conoscendone l'identificativo è necessario effettuare la stessa chiamata precedente (senza l'uso di query ad esclusione della view) passando nel path l'identificativo corrispondente, esempio:  
`api/v1/uiapi/sys/site/{identificativo}`



## A.9. PRESTAZIONI

Il servizio permette di recuperare prestazioni abilitate secondo i filtri impostati.

Servizio	Url	Metodo
Prestazioni	api/v1/uiapi/sys/healthcareservice	GET

I parametri da passare sono:

Parametro	Tipo	Data	Descrizione
Authorization	header	string	Basic Auth con username e password codificati in base64
name	query	string	Nome prestazione
code	query	string	Codice regionale prestazione
unitIds	query	Array[string]	Estrae le prestazioni associate a questa/e unità eroganti/refertanti
pattern	query	String	Filtra per Codice Mnemonico
doctorId	query	String	Filtro per ID dottore
onlyAdmissionAuthority	query	String	Filtra solo prestazioni per cui sia presente privilegio di accettazione. Si focalizza sulle prestazioni erogate dagli enti cui l'operatore è abilitato
view	query	String	Criterio di verbosità

Viene restituito un oggetto contenente la lista di informazioni associate alle prestazioni recuperate.

**Esempio Response (file Response\_Ricerca\_prestazioni.json)**

## A.10. LISTE DI LAVORO

Il servizio permette di recuperare la lista di lavoro secondo i filtri impostati.

Servizio	Url	Metodo
Gestione ordini	api/v1/uiapi/cpoe/order	GET

I parametri da passare sono:

Parametro	Tipo	Data	Valore	Descrizione
Authorization	header	string		Basic Auth con username e password codificati in base64
worklistTypeId	query	string	2	Filtro di ricerca per tipo lista di lavoro
fromScheduledDate	query	string		Filtra gli ordini che hanno una data erogazione successiva o uguale a questa data
toScheduledDate	query	string		Filtra gli ordini che hanno una data erogazione precedente o uguale a questa data
bookingStatusMask	query	string	1	Mask sullo status di prenotazione
executionStatusMask	query	string	7	Mask sullo status di erogazione
view	query	string	normal	Criterio di verbosità
onlyAuthorized	query	string	Y	Filtro per controllo abilitazioni visualizzazione appuntamenti
executionUnitIds	query	Array[string]		Filtra per elenco di unità eroganti
siteIds	query	Array[string]		Filtra per elenco di sedi

### Esempio Response (file Response\_Ricerca\_liste\_di\_lavoro.json)

## A.11. PROFILO

Il servizio permette di recuperare il profilo utente.

Servizio	Url	Metodo
Profilo	api/v1/uiapi/sys/userprofile	GET

Viene restituito un oggetto contenente una lista delle motivazioni attive.

### Esempio Response (file Response\_Profilo.json)

## A.12. TIPOLOGIE DI ACCETTAZIONE

Il servizio permette di recuperare tipologie di accettazione secondo i filtri impostati.

Servizio	Url	Metodo
Tipologie di accettazione	api/v1/uiapi/oms/admissiontype	GET

I parametri da passare sono:

Parametro	Tipo	Data	Valore	Descrizione
Authorization	header	string		Basic Auth con username e password codificati in base64
grantRead	query	string	Y	Filtro di ricerca per privilegio di visualizzazione
view	query	string	verbose	Criterio di verbosità

Viene restituito un oggetto contenente una lista delle tipologie di accettazione.

**Esempio Response (file Response\_Ricerca\_tipologie\_di\_accettazione.json)**

## A.13. COPERTURE ASSICURATIVE

Il servizio permette di recuperare coperture assicurative secondo i filtri impostati.

Servizio	Url	Metodo
Coperture Assicurative	api/v1/uiapi/pac/cover	GET

I parametri da passare sono:

Parametro	Tipo	Data	Valore	Descrizione
Authorization	header	string		Basic Auth con username e password codificati in base64
enable	query	string	Y	Filtro di ricerca per le strutture attive
view	query	string	normal	Criterio di verbosità
authorityIds	query	Array[string]		Uno o più ID di enti
subcategoryId	query	string		Filtro per la sottocategoria di fatturazione (vedere campo response pacSubcategoryId, paragrafo A.12)

Viene restituito un oggetto contenente una lista di coperture assicurative.

**Esempio Response (file Response\_Ricerca\_coperture\_assicurative.json)**

## A.14. LISTINI

Il servizio permette di recuperare listini secondo i filtri impostati.

Servizio	Url	Metodo
Listini	api/v1/uiapi/pac/pricelist	GET

I parametri da passare sono:

Parametro	Tipo	Data	Valore	Descrizione
Authorization	header	string		Basic Auth con username e password codificati in base64
coverIds	query	Array[string]		Uno o più ID di coperture assicurative
view	query	string	synthetic	Criterio di verbosità

Viene restituito un oggetto contenente una lista di listini.

### Esempio Response (file Response\_Ricerca\_listini.json)

## A.15. DOTTORI

Il servizio permette di recuperare dottori secondo i filtri impostati.

Servizio	Url	Metodo
Dottori	api/v1/uiapi/epr/doctor	GET

I parametri da passare sono:

Parametro	Tipo	Data	Valore	Descrizione
Authorization	header	string		Basic Auth con username e password codificati in base64
supplyCategoryIds		Array[string]		ID delle categorie di listino per cui filtrare: <ul style="list-style-type: none"><li>• 0 = SSN</li><li>• 1 = Solventi</li><li>• 2 = Libera Professione</li><li>• 3 = Convenzioni</li></ul>
coverAuthorityIds	query	Array[string]		Uno o più id di enti associati a coperture assicurative
view	query	string	normal	Criterio di verbosità
fiscalCode	query	string		Codice fiscale del medico

Viene restituito un oggetto contenente una lista di dottori.

### Esempio Response (file Response\_Ricerca\_dottori.json)

## A.16. UNITÀ EROGANTI

Il servizio permette di recuperare le unità eroganti secondo i filtri impostati.

Servizio	Url	Metodo
Sedi	api/v1/uiapi/sys/executionunit	GET

I parametri da passare sono:

Parametro	Tipo	Valore	Data	Descrizione
Authorization	header		string	Basic Auth con username e password codificati in base64
authorityIds	query		Array[string]	Uno o più ID di enti
servicelds	query		Array[string]	Filtro per ricercare solo le unità in grado di erogare almeno una tra le prestazioni elencate
allServicelds	query		Array[string]	Filtro per ricercare solo le unità in grado di erogare tutte le prestazioni elencate
careSupplyMode	query		String	Filtro singolo codice della modalità di erogazione: <ul style="list-style-type: none"><li>● 0 = SSN</li><li>● 1 = Solvente</li><li>● 2 = LP</li><li>● 3 = Convenzione</li><li>● 4 = Interni</li></ul>
onlyAuthorized	query	Y	String	Filtro per ricercare solo le unità eroganti per cui l'utente possiede le abilitazioni
view	query	normal	string	Criterio di verborosità

Viene restituito un oggetto contenente una lista di unità eroganti.

**Esempio Response (file Response\_Ricerca\_unità\_eroganti.json)**

## A.17. PREVENTIVI

Il servizio permette di recuperare preventivi abilitati i filtri impostati.

Servizio	Url	Metodo
Documenti di pagamento	api/v1/uiapi/pac/paymentpredocument	GET

I parametri da passare sono:

Parametro	Tipo	Data	Descrizione
Authorization	header	string	Basic Auth con username e password codificati in base64
admissionIds	query	string	L'Id di accettazione recuperato dalla pendenza da pagare (paragrafo D.1.)

Viene restituito un oggetto contenente una lista di preventivi.

**Esempio Response (file Response\_Ricerca\_Preventivi.json)**

## A.18. DOCUMENTI DI PAGAMENTO

Il servizio permette di recuperare documenti di pagamento secondo i filtri impostati.

Servizio	Url	Metodo
Documenti di pagamento	api/v1/uiapi/pac/paymentdocument	GET

I parametri da passare sono:

Parametro	Tipo	Data	Descrizione
Authorization	header	string	Basic Auth con username e password codificati in base64
admissionIds	query	string	L'Id di accettazione recuperato dalla pendenza da pagare (paragrafo D.1.)

Viene restituito un oggetto contenente una lista di documenti di pagamento.

**Esempio Response (file Response\_Documenti\_di\_pagamento.json)**

## A.19. SEDI DI CASSA

Il servizio permette di recuperare sedi di cassa secondo i filtri impostati.

Servizio	Url	Metodo
Documenti di pagamento	api/v1/uiapi/pac/paymentdocument	GET

I parametri da passare sono:

Parametro	Tipo	Data	Descrizione
Authorization	header	string	Basic Auth con username e password codificati in base64
admissionIds	query	string	L'Id di accettazione recuperato dalla pendenza da pagare (paragrafo D.1.)

Viene restituito un oggetto contenente una lista di sedi di cassa.

**Esempio Response (file Response\_Ricerca\_Sedi\_di\_Cassa.json)**

## A.20. PAGAMENTI

Il servizio permette di recuperare pagamenti secondo i filtri impostati.

Servizio	Url	Metodo
Pagamenti	api/v1/uiapi/pac/payment	GET

I parametri da passare sono:

Parametro	Tipo	Data	Descrizione
Authorization	header	string	Basic Auth con username e password codificati in base64
documentId	query	string	L'Id del documento.

Viene restituito un oggetto contenente una lista di pagamenti.

**Esempio Response (file Response\_Ricerca\_Pagamenti.json)**

## B. Prenotazione

La prenotazione comprende una serie di casi d'uso che contribuiscono alla realizzazione del flusso complessivo di prenotazione.

### Creazione Processi di prenotazione

Il servizio per la prenotazione necessita di un "idTransazione" (processId) da utilizzare come chiave all'interno del flusso di prenotazione stesso. Per ottenere tale chiave è necessario chiamare il servizio per l'apertura di un c.d. processo di prenotazione:

Servizio	Url	Metodo
Processi di prenotazione	api/v1/uiapi/oms/bpx	POST

Con i seguenti parametri all'interno del request body:

Parametro	Tipo	Descrizione
Authorization	header	Basic Auth con username e password codificati in base64
personId	body	ID persona (paragrafo A.5)

La response di tale servizio fornisce un identificativo univoco (tag "id") da utilizzare all'interno del flusso di prenotazione.

**Esempio Response (file Request\_Processo\_di\_prenotazione\_(recupero\_processId).json)**

**Esempio Response (file Response\_Processo\_di\_prenotazione\_(recupero\_processId).json)**

### Cancellazione Processi di prenotazione

Il servizio di cancellazione del processo di prenotazione fa sì che il sistema effettui tutte le operazioni necessarie a:

- chiusura della sessione in corso
- rilasciare tutti i locks della sessione
- avviare tutti i processi asincroni dopo la chiusura del processo di prenotazione (es. integrazione con PagoPA)
- creazione della stampa del promemoria e salvataggio sul repository

Servizio	Url	Metodo
Processi di prenotazione	api/v1/uiapi/oms/bpx/{processId}	DELETE

Con i seguenti parametri:

Parametro	Tipo	Descrizione
Authorization	header	Basic Auth con username e password codificati in base64
processId	path	ID del processo di prenotazione (Prenotazione - Creazione Processi di prenotazione)

La response di tale servizio restituisce come status 204.



## B.1. RICERCA PRIMA DISPONIBILITÀ

Una volta ottenuto l'idTransazione (processId) è possibile procedere alla ricerca di prima disponibilità per ogni prestazione all'interno dell'impegnativa, utilizzando il seguente servizio:

Servizio	Url	Metodo
Disponibilità	api/v1/uiapi/oms/availability	GET

I parametri elencati servono al servizio per cercare la prima disponibilità offerta per ogni prestazione contenuta all'interno dell'impegnativa.

Parametro	Valore	Tipo	Data	Descrizione
Authorization		header	string	Basic Auth con username e password codificati in base64
supplyModeCategoryIds		query	Array[string]	Categoria del regime di erogazione (paragrafo A.3.)
supplyModelIds		query	Array[string]	ID dei regimi di erogazione (paragrafo A.3.)
processId		query	string	L'ID del processo di prenotazione (paragrafo B Processi di prenotazione)
orders		query	Array[string]	Elenco delle prestazioni (orders) in formato stringa. Ogni stringa dovrà prevedere i seguenti valori separati da un pipe: <ul style="list-style-type: none"><li>• id di correlazione (obbligatorio); viene imputato dal client per accoppiare due o più orders per una gestione condivisa</li><li>• id dell'ordine (requestId) (obbligatorio); viene imputato dal client e verrà restituito nel risultato della ricerca per permettere al client di ricondurre una determinata disponibilità ad uno degli elementi di questa list. Dovrà essere riportato anche in fase di prenotazione della disponibilità ritornata</li><li>• id della prestazione (obbligatorio)</li><li>• priorità (obbligatorio)</li><li>• flag accesso diretto - Y o N (obbligatorio)</li><li>• id del medico (opzionale)</li></ul> es. 1 A ECG B serve per rappresentare un ECG con quantità 1 e priorità Breve
correlationStrategies		query	Array[string]	Elenco delle strategie di correlazione in formato stringa. Ogni stringa dovrà prevedere i seguenti valori separati da un pipe: <ul style="list-style-type: none"><li>• id di correlazione (obbligatorio); viene imputato dal client per accoppiare due o più orders per una gestione condivisa</li></ul>

				<ul style="list-style-type: none"> <li>• nome della strategia (obbligatorio); valore NRE</li> <li>• valore della strategia (opzionale)</li> </ul> es. 2 NRE 120101000000134 serve per indicare che gli order con il correlationId = 2 appartengono alla stessa impegnativa il cui numero è 120101000000134
firstBy	absolute	query	string	Prima disponibilità assoluta
personId		query	string	ID dell'individuo per cui cercare la disponibilità (paragrafo A.5.)
online	true	query	string	Filtra per agende e prestazioni prenotabili in autonomia da cittadino.
weekDays???		query		fisso a 127 per caso tamponi
fromDate???		query		
toDate???		query		
siteIds???		query		lista site id dei driveln
accessMode???		query	string	valori 0 o 1 (a piedi o in macchina)

Un esempio di caso d'uso è definito dal seguente flusso:

1. Recupero prestazioni della ricetta => Ricetta dematerializzata (paragrafo A.6.)
2. Creazione di un processo di prenotazione => (Prenotazione - Creazione Processi di prenotazione)
3. Ricerca Prima Disponibilità per le prestazioni presenti in ricetta

Viene restituito un oggetto contenente la prima disponibilità per l'appuntamento.

**Esempio Response (file Response\_Ricerca\_Prima\_Disponibilità.json)**

## B.2. RICERCA DISPONIBILITÀ SUCCESSIVA

Il servizio permette di richiedere al CUP la disponibilità successiva per le prestazioni proposte.

Servizio	Url	Metodo
Disponibilità	api/v1/uiapi/oms/availability	GET

I parametri elencati servono al servizio per applicare i filtri opportuni per la ricerca richiesta.

Parametro	Valore	Tipo	Data	Descrizione
Authorization		header	string	Basic Auth con username e password codificati in base64
supplyModeCategoryIds		query	Array[string]	Categoria del regime di erogazione (paragrafo A.3.)
supplyModelIds		query	Array[string]	ID dei regimi di erogazione (paragrafo A.3.)
orders		query	Array[string]	Elenco delle prestazioni
correlationStrategies		query	Array[string]	Elenco delle strategie di correlazione
firstBy	date	query	string	Prima disponibilità per giorno
processId		query	string	L'ID del processo di prenotazione (paragrafo B Processi di prenotazione)
personId		query	string	ID dell'individuo per cui cercare la disponibilità (paragrafo A.5.)
online	true	query	string	Filtra per agende e prestazioni prenotabili in autonomia da cittadino.

Un esempio di caso d'uso è definito dal seguente flusso:

1. Recupero prestazioni della ricetta => Ricetta dematerializzata (paragrafo A.6.)
2. Creazione di un processo di prenotazione => (Prenotazione - Creazione Processi di prenotazione)
3. Ricerca Disponibilità Successiva per le prestazioni presenti in ricetta

Viene restituito un oggetto contenente una lista di disponibilità per l'appuntamento sulla base dei filtri utilizzati.

### Esempio Response (file Response\_Ricerca\_Disponibilità\_Successiva.json)

### B.3. LISTA STRUTTURE E/O AMBULATORI

Attraverso l'utilizzo di questo servizio è possibile ottenere la lista delle strutture e/o ambulatori che erogano tutte le prestazioni contenute nella prescrizione (NRE).

Servizio	Url	Metodo
Unità Eroganti	api/v1/uiapi/sys/executionunit/	GET

I parametri da passare sono:

Parametro	Tipo	Data	Descrizione
Authorization	header	string	Basic Auth con username e password codificati in base64
servicelds	query	Array[string]	Filtro per ricercare solo le unità in grado di erogare almeno una tra le prestazioni elencate. Valori ammessi: uno o più ID di prestazioni
allServicelds	query	Array[string]	Filtro per ricercare solo le unità in grado di erogare tutte le prestazioni elencate. Valori ammessi: uno o più ID di prestazioni
authoritylds	query	Array[string]	Uno o più ID di enti
hospitallds	query	Array[string]	Uno o più ID di presidi
sitelds	query	Array[string]	Uno o più ID di sedi
type	query	integer	Filtra per il Tipo di unità erogante: <ul style="list-style-type: none"><li>● 0 = Struttura di tipo reparto misto (ordinari-diurni)</li><li>● 1 = Struttura di tipo ambulatorio</li><li>● 2 = Struttura di tipo laboratorio</li><li>● 3 = Struttura di tipo radiologia</li><li>● 4 = Struttura di tipo pronto soccorso</li><li>● 5 = Ufficio Amministrativo</li><li>● 6 = Reparto di Interni</li><li>● 7 = Reparto di Day Hospital</li><li>● 8 = Reparto assistenziale</li><li>● 9 = Centro di responsabilità</li><li>● 10 = Centro PRE-RICOVERO</li><li>● 11 = Reparto di Week Surgery</li><li>● 11 = Centro Microbiologia</li><li>● 12 = Struttura HOSPICE</li><li>● 13 = Struttura DAY HOSPICE.</li></ul>

Un esempio di caso d'uso è definito dal seguente flusso:

1. Recupero prestazioni della ricetta => Ricetta dematerializzata (paragrafo A.6.)
2. Ricerca Strutture e/o ambulatori che erogano tutte le prestazioni in ricetta

Viene restituito un oggetto contenente una lista di strutture e/o ambulatori.

**Esempio Response (file Response\_Lista\_strutture\_e\_o\_ambulatori.json)**

## B.4. RICERCA DISPONIBILITÀ

In fine è possibile cercare la prima disponibilità utile impostando una serie di filtri relativi alla data, fascia oraria e struttura sanitaria desiderata e/o ambulatorio. NB: Tale ricerca può restituire un insieme vuoto.

Servizio	Url	Metodo
Disponibilità	api/v1/uiapi/oms/availability	GET

I parametri da passare sono:

Parametro	Valore	Tipo	Data	Descrizione
Authorization		header	string	Basic Auth con username e password codificati in base64
supplyModeCategoryIds		query	Array[string]	Categoria del regime di erogazione (paragrafo A.3.)
processId		query	string	L'ID del processo di prenotazione (paragrafo B Processi di prenotazione)
supplyModelIds		query	Array[string]	ID dei regimi di erogazione (paragrafo A.3.)
personId		query	string	ID dell'individuo per cui cercare la disponibilità (paragrafo A.5.)
orders		query	Array[string]	Elenco delle prestazioni
correlationStrategies		query	Array[string]	Elenco delle strategie di correlazione
firstBy	absolute	query	string	Prima disponibilità assoluta
fromDate		query	string	La data minima di disponibilità in formato ISO-8601 yyyy-mm-dd
toDate		query	string	La data massima di disponibilità in formato ISO-8601 yyyy-mm-dd
fromTime		query	string	L'ora minima di disponibilità in formato ISO-8601 hh:mm:ss
toTime		query	string	L'ora massima di disponibilità in formato ISO-8601 hh:mm:ss
unitIds		query	Array[string]	ID delle unità eroganti in cui cercare la disponibilità (paragrafo B.3.)
online	true	query	string	Filtra per agende e prestazioni prenotabili in autonomia da cittadino.

Un esempio di caso d'uso è definito dal seguente flusso:

1. Recupero prestazioni della ricetta => Ricetta dematerializzata (paragrafo A.6.)
2. Ricerca strutture e/o ambulatori => Lista Strutture e/o ambulatori (paragrafo B.3)
3. Creazione di un processo di prenotazione => (Prenotazione - Creazione Processi di prenotazione)
4. Ricerca Disponibilità per le prestazioni presenti in ricetta

Viene restituito un oggetto contenente una lista di disponibilità per l'appuntamento sulla base dei filtri utilizzati.

## Esempio Response (file Response\_Ricerca\_Disponibilità.json)

### B.5. CONFERMA PRENOTAZIONE

Una volta effettuata la ricerca per disponibilità, è necessario confermare la prenotazione mediante il servizio esposto. Questo servizio ha il compito di registrare a sistema la prenotazione.

Servizio	Url	Metodo
Prenotazione	api/v1/uiapi/oms/booking	POST

I parametri da passare sono:

Parametro	Tipo	Descrizione
Authorization	header	Basic Auth con username e password codificati in base64
supplyMode	body	Il regime di erogazione
service	body	La prima prestazione
extraServices	body	Le altre prestazioni raggruppabili
diary	body	Agenda che ha fornito la disponibilità per l'appuntamento
person	body	La persona/paziente oggetto della prenotazione
processId	body	L'ID del processo di prenotazione (paragrafo B Processi di prenotazione)
requestId	body	ID usato in ricerca disponibilità per identificare un determinato elemento nel carrello di ricerca disponibilità
startTime	body	Data/ora inizio appuntamento
prescription	body	L'impegnativa associata all'appuntamento

**NB: è possibile identificare prestazioni raggruppabili attraverso la ricerca disponibilità; prestazioni aventi stesso slot disponibile su di un'agenda (diary e date uguali) risultano raggruppabili, ossia è possibile fare un'unica chiamata di pre-prenotazione e prenotazione utilizzando il parametro extraServices.**

**NB: i parametri service e extraServices saranno deprecati per prossima sostituzione con campo unico "services" (Array[string])**

Un esempio di caso d'uso è definito dal seguente flusso:

1. Recupero disponibilità per le prestazioni presenti in ricetta => Ricerca Disponibilità (una di quelle definite nei paragrafi B.1., B.2. e B.4.)
2. Possibilità di effettuare una pre-prenotazione (paragrafo B.7.)
3. Selezione e conferma dell'appuntamento
4. Cancellazione di un processo di prenotazione => (Prenotazione - Cancellazione Processi di prenotazione)

Viene restituito un oggetto contenente le informazioni dell'appuntamento appena inserito.

**Esempio Request (file Request\_Conferma\_Prenotazione.json)**

**Esempio Request (file Request\_Conferma\_Prenotazione (extraServices).json)**

**Esempio Response (file Response\_Conferma\_Prenotazione.json)**

**Esempio Response (file Response\_Conferma\_Prenotazione (extraServices).json)**

## B.6. ANNULLA PRENOTAZIONE IN CORSO

I lock temporanei per le prenotazioni non portate a termine vengono rilasciati dopo un tempo prefissato con lo scopo di rendere nuovamente prenotabile lo slot. Eventualmente è possibile rilasciare la risorsa mediante il servizio:

Servizio	Url	Metodo
Processi di prenotazione	api/v1/uiapi/oms/bpx/{processId}	DELETE

In questo modo è possibile annullare una prenotazione in corso, rilasciando appunto le risorse bloccate per la transazione di prenotazione ({processId}).

Parametro	Tipo	Data	Descrizione
Authorization	header	string	Basic Auth con username e password codificati in base64
processId	path	string	Id processo di prenotazione

Un esempio di caso d'uso è definito dal seguente flusso:

1. Recupero prestazioni della ricetta => Ricetta dematerializzata (paragrafo A.6.)
2. Creazione di un processo di prenotazione => (Prenotazione - Creazione Processi di prenotazione)
3. Recupero disponibilità per le prestazioni presenti in ricetta => Ricerca Disponibilità (una di quelle definite nei paragrafi B.1., B.2. e B.4.)
4. Cancellazione del processId per il processo di prenotazione in corso

## B.7. PRE-PRENOTAZIONE

E' possibile impostare un lock temporaneo sulla disponibilità fornita utilizzando il seguente servizio

Servizio	Url	Metodo
Pre-prenotazione	api/v1/uiapi/oms/bpx/{processId}/prebooking	POST

I parametri da passare sono:

Parametro	Tipo	Descrizione
Authorization	header	Basic Auth con username e password codificati in base64
processId	path	Id processo di prenotazione
date	body	Data lock
diary	body	Agenda che ha fornito la disponibilità per l'appuntamento
personId	body	ID dell'individuo per cui cercare la disponibilità (paragrafo A.5.)
processId	body	Id processo di prenotazione
service	body	La prestazione
extraServices	body	Le altre prestazioni prestazioni raggruppabili
requestId	body	ID usato in ricerca disponibilità per identificare un determinato elemento nel carrello di ricerca disponibilità
nre	body	Numero ricetta
dateOfCompletion	body	Data di compilazione
priorityClass	body	Priorità (paragrafo A.4.)
reservationId	body	L'Id processo di prenotazione
supplyMode	body	Il regime di erogazione (paragrafo A.3.)

Un esempio di caso d'uso è definito dal seguente flusso:

1. Recupero disponibilità per le prestazioni presenti in ricetta => Ricerca Disponibilità (una di quelle definite nei paragrafi B.1., B.2. e B.4.)
2. Lock temporaneo su una delle disponibilità recuperate

**Esempio Request (file Request\_Pre\_prenotazione.json)**

**Esempio Request (file Request\_Pre\_prenotazione (extraServices).json)**



## C. Consultazione Appuntamenti

La consultazione degli appuntamenti contiene una serie di casi d'uso che hanno a che fare con gli appuntamenti già prenotati.

### C.1. LISTA APPUNTAMENTI

Il servizio consente di recuperare la lista degli appuntamenti associati ad un utente.

Servizio	Url	Metodo
Prenotazione	api/v1/uiapi/oms/booking	GET

I parametri da passare sono:

Parametro	Valore	Tipo	Data	Descrizione
Authorization		header	string	Basic Auth con username e password codificati in base64
patientId		query	string	L'ID del paziente (paragrafo A.5.; patientId = personId)
fromDate		query	string	Filtra gli appuntamenti che sono successivi o uguali a questa data
bookingListTypeId	0	query	string	Filtra i campi ritornati in base al tipo di lista di lavoro (impostare a 2 per tampone???)
view	verbose	query	string	Criterio di verbosità
nre???				
statusMask???				

Un esempio di caso d'uso è definito dal seguente flusso:

1. Recupero appuntamenti paziente

Viene restituito un oggetto contenente la lista degli appuntamenti sulla base dei filtri impostati.

**NB: per recuperare le informazioni di un appuntamento conoscendone l'identificativo è necessario effettuare la stessa chiamata precedente (senza l'uso di query) passando nel path l'identificativo corrispondente, esempio: `api/v1/uiapi/oms/booking/{identificativo}`**

**Esempio Response (file Response\_Lista\_Appuntamenti.json)**

## C.2. ANNULLA APPUNTAMENTO

Servizio che permette l'annullamento dell'intero blocco di appuntamenti associati all'idPrenotazione recuperato attraverso la lista appuntamenti.

Servizio	Url	Metodo
Prenotazione	api/v1/uiapi/oms/booking/{idPrenotazione}	PATCH

I parametri da passare sono:

Parametro	Valore	Tipo	Data	Descrizione
Authorization		header	string	Basic Auth con username e password codificati in base64
Content-Type	application/merge-patch+json	header	string	Il tipo di dati multimediali contenuti nell'oggetto request
idPrenotazione		path	string	Id prenotazione
status	{"id" : 2}	body	string	Lo stato dell'appuntamento (id=2 stato cancellato)
_meta		body	string	Metadati per l'annullamento della prenotazione

Viene restituito un oggetto contenente le informazioni dell'appuntamento cancellato.

Il tag "reasonId" si deve compilare con uno dei valori recuperati (tag "id") attraverso il servizio motivazioni (paragrafo A.2.).

Un esempio di caso d'uso è definito dal seguente flusso:

1. Recupero appuntamenti paziente => Lista Appuntamenti (paragrafo C.1.)
2. Annullamento dell'intero blocco di appuntamenti associati alla prenotazione indicata

**Esempio Request (file Request\_Annulla\_appuntamento.json)**

**Esempio Response (file Response\_Annulla\_appuntamento.json)**

### C.3. SPOSTA APPUNTAMENTO

Servizio per lo spostamento di un appuntamento:

Servizio	Url	Metodo
Prenotazione	api/v1/uiapi/oms/booking	POST

I parametri da passare sono:

Parametro	Tipo	Descrizione
Authorization	header	Basic Auth con username e password codificati in base64
supplyMode	body	Il regime di erogazione
service	body	La prestazione
diary	body	Agenda che ha fornito la disponibilità per l'appuntamento
person	body	La persona/paziente oggetto della prenotazione
processId	body	L'ID del processo di prenotazione (paragrafo B Processi di prenotazione)
requestId	body	ID usato in ricerca disponibilità per identificare un determinato elemento nel carrello di ricerca disponibilità
startTime	body	Data/ora inizio appuntamento
_meta	body	Contiene le informazioni della prenotazione da spostare

Viene restituito un oggetto contenente le informazioni dell'appuntamento spostato.

I tag da utilizzare quali movedBookingReason e movedBookingNote si devono compilare con uno dei valori recuperati (rispettivamente tag "id" e "name") attraverso il servizio motivazioni (paragrafo A.2.).

Un esempio di caso d'uso è definito dal seguente flusso:

1. Recupero appuntamenti paziente => Lista Appuntamenti (paragrafo C.1.)
2. Recupero disponibilità per le prestazioni presenti associate all'appuntamento => Ricerca Disponibilità (una di quelle definite nei paragrafi B.1., B.2. e B.4. senza il parametro correlationStrategies)
3. Spostamento dell'appuntamento desiderato

**Esempio Request (file Request\_Sposta\_appuntamento.json)**

**Esempio Response (file Response\_Sposta\_appuntamento.json)**

#### C.4. RISTAMPA RICEVUTA DI PRENOTAZIONE

Il servizio consente di recuperare la ricevuta relativa ad una prenotazione, sulla base dell'identificativo del processo di prenotazione ({processId}).

Servizio	Url	Metodo
Stampa	api/v1/uiapi/oms/bpx/{processId}.PDF	GET

I parametri da passare sono:

Parametro	Tipo	Data	Descrizione
Authorization	header	string	Basic Auth con username e password codificati in base64
processId	path	string	Id del processo di prenotazione

Un esempio di caso d'uso è definito dal seguente flusso:

1. Recupero appuntamenti paziente => Lista Appuntamenti (paragrafo C.1.)
2. Stampa promemoria appuntamento desiderato

## D. Pagamento

Il pagamento di una prestazione comprende una serie di casi d'uso che contribuiscono alla realizzazione del flusso complessivo di pagamento della prestazione.

### D.1. LISTA PENDENZE DA PAGARE

Il servizio consente di recuperare la lista di pendenze di pagamento collegato all'id del paziente collegato.

Servizio	Url	Metodo
Accettazione	api/v1/uiapi/oms/admission	GET

I parametri da passare sono:

Parametro	Valore	Tipo	Data	Descrizione
Authorization		header	string	Basic Auth con username e password codificati in base64
view	normal	query	string	Criterio di verbosità
patientId		query	string	L'id del paziente
takeOnlyPayed	N	query	string	Ricerca prescrizioni non fatturate

Un esempio di caso d'uso è definito dal seguente flusso:

1. Recupero id paziente => Informazioni paziente (paragrafo A.4.)
2. Ricerca pendenze da pagare => Lista pendenze da pagare

Viene restituito un oggetto una lista delle pendenze ancora da pagare da parte del paziente.

**Esempio Response (file Response Lista\_pendenze\_da\_pagare.json)**

## D.2. PAGA

Per poter registrare un pagamento di una delle pendenze di pagamento ottenute mediante il servizio “Lista pendenze da pagare” (paragrafo D.1.) è necessario utilizzare in serie e ordine i seguenti servizi.

### *Processo di cassa*

Questo servizio permette di creare/avviare un nuovo processo di cassa.

Servizio	Url	Metodo
Processo di cassa	api/v1/uiapi/pac/cpx	POST

I parametri da passare sono:

Parametro	Tipo	Descrizione
Authorization	header	Basic Auth con username e password codificati in base64
ieo-authority-id	header	L'Id dell'ente associato alla cassa
ieo-site-id	header	L'Id della sede di cassa
cashSite	body	L'Id della sede di cassa a cui il processo di cassa afferisce

Se esiste già un processo di cassa aperto per la stessa cassa operatore, verrà restituito un messaggio http 409.

Viene restituito un oggetto con le informazioni del processo di cassa creato o già presente (in questo caso le informazioni del processo di cassa sono all'interno dell'oggetto “meta”).

**Esempio Request (file Request\_Processo\_di\_cassa.json)**

**Esempio Response (file Response\_Processo\_di\_cassa\_creazione.json)**

**Esempio Response (file Response\_Processo\_di\_cassa\_presente.json)**

## Transazioni di cassa

Una volta recuperate le informazioni associate ai documenti di pagamento è possibile procedere all'inserimento del pagamento stesso tramite il seguente servizio:

Servizio	Url	Metodo
Transazioni di cassa	api/v1/uiapi/pac/cash/{cashId}/transaction	POST

I parametri da passare sono:

Parametro	Valore	Tipo	Descrizione
Authorization		header	Basic Auth con username e password codificati in base64
ieo-authority-id		header	L'Id dell'ente recuperato dalla pendenza da pagare (paragrafo D.1.)
ieo-cash-process-id		header	L'Id del processo di cassa recuperato dal servizio precedente
ieo-site-id		header	L'Id della sede recuperato dalla pendenza da pagare (paragrafo D.1.)
cashId		path	Id della cassa
document		body	L'Id del documento di pagamento (paragrafo D.2. Documenti di pagamento)
deposit	N	body	La transazione non è un acconto
transactionType	C	body	Tipo di transazione (C=Pagamento)
processId		body	L'Id del processo di cassa (paragrafo D.2. Processo di cassa)
amount	null	body	Ammontare della transazione
payments		body	Informazioni relative al pagamento quali il totale pagato ("amout") o la tipologia di pagamento ("mode").

Viene restituito un oggetto con una lista di documenti di pagamento associati alla pendenza da pagare indicata dall'Id di accettazione.

Un esempio di caso d'uso è definito dal seguente flusso:

1. Ricerca pendenze da pagare => Lista pendenze da pagare (paragrafo D.1.)
2. Creo/recupero processo di cassa => paragrafo D.2. Processo di cassa
3. Aggiorno processo di accettazione => paragrafo D.2. Processo di accettazione
4. Recupero documenti di pagamento => paragrafo D.2. Documenti di pagamento
5. Esegui la transazione di cassa => paragrafo D.2. Transazioni di cassa

**Esempio Response (file Response\_Transazioni\_di\_cassa.json)**

**Esempio Request (file Request\_Transazioni\_di\_cassa.json)**

### D.3. STORICO DEI PAGAMENTI

Il servizio consente di recuperare lo storico dei pagamenti collegato all'id del paziente collegato ed a partire dalla data indicata.

Servizio	Url	Metodo
Accettazione	api/v1/uiapi/oms/admission	GET

I parametri da passare sono:

Parametro	Valore	Tipo	Data	Descrizione
Authorization		header	string	Basic Auth con username e password codificati in base64
patientId		query	string	L'Id del paziente (paragrafo A.5.)
takeOnlyPayed	Y	query	string	Ricerca prescrizioni fatturate
docStartTime		query	string	Data inizio ricerca documento

Un esempio di caso d'uso è definito dal seguente flusso:

1. Ricerca storico dei pagamenti

Viene restituito un oggetto una lista dei pagamenti del paziente a partire da una certa data.

**Esempio Response (file Response\_Storico\_dei\_pagamenti.json)**



#### D.4. RECUPERO RICEVUTA FISCALE

E' possibile recuperare i documenti dei pagamenti richiamando il seguente servizio:

Servizio	Url	Metodo
Ricevuta Fiscale	api/v1/uiapi/pac/paymentdocument/{documentId}.PDF	GET

Dove {documentId} è l'Id del documento recuperato tramite la ricerca dello storico dei pagamenti (paragrafo D.3.).

Parametro	Tipo	Data	Descrizione
Authorization	header	string	Basic Auth con username e password codificati in base64
documentId	path	string	Id del documento

Un esempio di caso d'uso è definito dal seguente flusso:

1. Ricerca storico dei pagamenti => Storico dei pagamenti
2. Recupero ricevuta fiscale della prescrizione che mi interessa

Viene restituito il documento o ricevuta fiscale della prescrizione scelta.

## E. Accettazione

L'accettazione comprende una serie di casi d'uso che contribuiscono alla realizzazione del flusso complessivo di accettazione.

### *Processo di accettazione*

Tramite questo servizio è possibile creare un processo di accettazione.

Servizio	Url	Metodo
Processo di accettazione	api/v1/uiapi/oms/apx	POST

I parametri da passare sono:

Parametro	Tipo	Descrizione
Authorization	header	Basic Auth con username e password codificati in base64
ieo-authority-id	header	L'Id dell'ente recuperato dalla pendenza da pagare (paragrafo D.1.)
ieo-cash-process-id	header	L'Id del processo di cassa recuperato dal servizio precedente
ieo-site-id	header	L'Id della sede recuperato dalla pendenza da pagare (paragrafo D.1.)
personId	body	L'id del paziente
admissions	body	Elenco di accettazioni che fanno parte di questo processo di accettazione

Viene restituito un oggetto con le informazioni del processo di accettazione creato.

**Esempio Request (file Request\_Processo\_di\_accettazione.json)**

**Esempio Response (file Response\_Processo\_di\_accettazione.json)**

## Processo di accettazione - abilitazione al pagamento

Tramite questo servizio è possibile aggiornare in maniera atomica alcune informazioni del processo di accettazione.

Servizio	Url	Metodo
Processo di accettazione	api/v1/uiapi/oms/apx/{processId}	PATCH

I parametri da passare sono:

Parametro	Valore	Tipo	Data	Descrizione
Authorization		header	string	Basic Auth con username e password codificati in base64
Content-Type	application/json-patch+json	header	string	Il tipo di dati multimediali contenuti nell'oggetto request
ieo-authority-id		header	string	L'Id dell'ente recuperato dalla pendenza da pagare (paragrafo D.1.)
ieo-cash-process-id		header	string	L'Id del processo di cassa recuperato dal servizio precedente
ieo-site-id		header	string	L'Id della sede recuperato dalla pendenza da pagare (paragrafo D.1.)
processId		path	string	L'Id del processo di accettazione creato precedentemente
	<pre>[{"op": "replace", "path": "/status.id", "value": "P"}, {"op": "replace", "path": "/predocument", "value": true}]</pre>	body	string	Lista di operazioni

Cambiare lo status del processo di accettazione a "Payable" farà scattare tutte le logiche di fatturazione legate alla generazione dei documenti contabili relativi alle accettazioni del processo in questione.

**NB: dalla release 21.03 in questo step verranno creati i preventivi e non più i documenti contabili; quest'ultimi verranno creati solo all'atto del pagamento. In questo caso è necessario passare tutte e 2 le option.**

Viene restituito un oggetto con le informazioni del processo di accettazione aggiornato.

**Esempio Response (file Request\_Processo\_di\_accettazione - abilitazione al pagamento.json)**

**Esempio Response (file Response\_Processo\_di\_accettazione - abilitazione al pagamento.json)**

## Processo di accettazione - apertura posizione debitoria su PagoPa

Tramite questo servizio è possibile aggiornare in maniera atomica alcune informazioni del processo di accettazione.

Servizio	Url	Metodo
Processo di accettazione	api/v1/uiapi/oms/apx/{processId}	PATCH

I parametri da passare sono:

Parametro	Valore	Tipo	Data	Descrizione
Authorization		header	string	Basic Auth con username e password codificati in base64
Content-Type	application/json-patch+json	header	string	Il tipo di dati multimediali contenuti nell'oggetto request
ieo-authority-id		header	string	L'Id dell'ente recuperato dalla pendenza da pagare (paragrafo D.1.)
ieo-cash-process-id		header	string	L'Id del processo di cassa recuperato dal servizio precedente
ieo-site-id		header	string	L'Id della sede recuperato dalla pendenza da pagare (paragrafo D.1.)
processId		path	string	L'Id del processo di accettazione creato precedentemente
	[{"op": "replace", "path": "/pagopa"}]	body	string	Lista di operazioni

Viene restituito un oggetto con le informazioni del processo di accettazione aggiornato.

**Esempio Response (file Request\_Processo\_di\_accettazione - apertura posizione debitoria su PagoPa.json)**

**Esempio Response (file Response\_Processo\_di\_accettazione - apertura posizione debitoria su PagoPa.json)**

### Processo di accettazione - stampa avviso di pagamento

Tramite questo servizio è possibile recuperare l'avviso di pagamento del processo di accettazione.

Servizio	Url	Metodo
Processo di accettazione	api/v1/uiapi/oms/apx/{processId}.PDF	GET

I parametri da passare sono:

Parametro	Valore	Tipo	Data	Descrizione
Authorization		header	string	Basic Auth con username e password codificati in base64
processId		path	string	L'Id del processo di accettazione creato precedentemente
print	paymentdocument	query	string	Stampa avviso di pagamento

Viene restituito il documento di avviso di pagamento del processo di accettazione indicato.

## E.1. CONFERMA ACCETTAZIONE

Una volta recuperate le informazioni da inserire nell'accettazione, è possibile registrare quest'ultima mediante il seguente servizio..

Servizio	Url	Metodo
Accettazione	api/v1/uiapi/oms/admission	POST

I parametri da passare sono:

Parametro	Valore	Tipo	Descrizione
Authorization		header	Basic Auth con username e password codificati in base64
id	new	body	Identificativo accettazione
processId		body	L'id del processo di accettazione a cui è agganciata questa accettazione
patient		body	Il paziente
supplyModeCategory		body	Il regime di erogazione
prescription		body	La ricetta (cartacea o elettronica)
type		body	Tipo accettazione
cover		body	La copertura assicurativa/convenzione
doctor		body	Il medico erogatore
site		body	Sede richiedente
executionUnit		body	L'unità che erogherà/ha erogato la prestazione
directEntry	true	body	Flag che indica se l'accettazione è in accesso diretto e quindi non richiede i dati dell'impegnativa
details		body	I record di dettaglio dell'accettazione

Un esempio di caso d'uso è definito dal seguente flusso:

1. Recupero informazioni paziente => Informazioni Paziente (paragrafo A.5)
2. Creazione processo di accettazione => Accettazione (paragrafo E.Processo di accettazione)
3. Recupero tipologia di accettazione => Tipologie di Accettazione (paragrafo A.12)
4. Recupero copertura assicurativa => Coperture (paragrafo A.13)
5. Recupero listino => Listini (paragrafo A.14)
6. Recupero dottore => Dottori (paragrafo A.15)
7. Recupero prestazioni => Prestazioni (paragrafo A.9)
8. Conferma accettazione

Viene restituito un oggetto contenente le informazioni dell'accettazione appena inserita.

**Esempio Request (file Request\_Conferma\_Accettazione.json)**

**Esempio Response (file Response\_Conferma\_Accettazione.json)**

## E.2. AGGIORNAMENTO ACCETTAZIONE

Tramite questo servizio è possibile aggiornare in maniera atomica alcune informazioni dell'accettazione.

Servizio	Url	Metodo
Accettazione	api/v1/uiapi/oms/admission/{admissionId}	PATCH

I parametri da passare sono:

Parametro	Valore	Tipo	Data	Descrizione
Authorization		header	string	Basic Auth con username e password codificati in base64
Content-Type	application/json-patch+json	header	string	Il tipo di dati multimediali contenuti nell'oggetto request
admissionId		path	string	ID accettazione creata precedentemente
	<pre>[{"op": "replace",   "path": "/directAdmissionReasonId",   "value": "801"},  {"op": "replace",   "path": "/status.id",   "value": "TSV"}]</pre>	body	string	Lista di operazioni

`{"op": "replace", "path": "/directAdmissionReasonId", "value": "801"}`: aggiorna la causale di un'accettazione indicandola come ACCETTAZIONE DIRETTA.

`{"op": "replace", "path": "/status.id", "value": "TSV"}`: aggiorna lo stato di un'accettazione come DA EROGARE.

Viene restituito un oggetto con le informazioni dell'accettazione aggiornate.

**Esempio Response (file Request\_Accettazione\_Diretta\_Causale.json)**

**Esempio Response (file Response\_Accettazione\_Diretta\_Causale.json)**

**Esempio Response (file Request\_Accettazione\_Diretta\_Stato.json)**

**Esempio Response (file Response\_Accettazione\_Diretta\_Stato.json)**

## E.3. RICERCA ACCETTAZIONE

E' possibile recuperare i documenti dei pagamenti richiamando il seguente servizio:

Servizio	Url	Metodo
Accettazione	api/v1/uiapi/oms/admission/{admissionId}	GET

Dove {admissionId} è l'Id dell'accettazione.

Parametro	Tipo	Data	Descrizione
Authorization	header	string	Basic Auth con username e password codificati in base64
admissionId	path	string	Id dell'accettazione

Viene restituita l'accettazione cercata.



## F. Pre-Appuntamenti

Elenco di servizi per il recupero e la conferma dei Pre-Appuntamenti.

### F.1. LISTA PRE-APPUNTAMENTI

E' possibile recuperare i Pre-Appuntamenti per un paziente richiamando il seguente servizio:

Servizio	Url	Metodo
Lista Pre-Appuntamenti	api/v1/uiapi/cpoe/order	GET

I parametri elencati servono al servizio per applicare i filtri opportuni per la ricerca richiesta.

Parametro	Valore	Tipo	Data	Descrizione
Authorization		header	string	Basic Auth con username e password codificati in base64
worklistTypeid	2	query	string	Filtro di ricerca per tipo lista di lavoro
fromScheduledDate		query	string	Filtra gli ordini che hanno una data erogazione successiva o uguale a questa data
toScheduledDate		query	string	Filtra gli ordini che hanno una data erogazione precedente o uguale a questa data
patientId		query	string	L'ID del paziente (paragrafo A.5.; patientId = personId)
bookingStatusMask	128	query	string	Mask sullo status di prenotazione
onlyAuthorized	Y	query	string	Filtro per controllo abilitazioni visualizzazione appuntamenti
preAppointment	Y	query	string	Ricerca solo Pre-Appuntamenti
view	verbose	query	string	Criterio di verbosità

Viene restituito un oggetto contenente una lista di Pre-Appuntamenti sulla base dei filtri utilizzati.

**NB: i Pre-Appuntamenti scaduti sono quelli con il campo "expireDate" minore della data odierna**

**Esempio Response (file Response\_Ricerca\_Lista\_Pre\_Appuntamenti.json)**

## F.2. CONFERMA PRE-APPUNTAMENTO

Un esempio di caso d'uso è definito dal seguente flusso:

1. Ricerca Pre-Appuntamenti Paziente => Lista Pre-Appuntamenti (paragrafo F.1.)
2. Recupero prestazioni della ricetta => Ricetta dematerializzata (paragrafo A.6.)
3. Recupero Pre-Appuntamento da confermare => Lista Appuntamenti (paragrafo C.1. con ricerca per identificativo); **Esempio response (file Response\_Recupero\_Appuntamento.json)**
4. Creazione di un processo di prenotazione => (Prenotazione - Creazione Processi di prenotazione)
5. Lock pre-prenotazione => Pre-Prenotazione (paragrafo B.7.)
6. Conferma Pre-Appuntamento => Conferma Prenotazione (paragrafo B.5.)
7. Cancellazione processo di prenotazione => (Prenotazione - Cancellazione Processi di prenotazione)

Viene restituito un oggetto contenente le informazioni dell'appuntamento appena confermato.

**Esempio Response (file Response\_Conferma\_Pre\_Appuntamento.json)**

**Esempio Response (file Response\_Conferma\_Pre\_Appuntamento.json)**