

PAPI Overview

Probooker Application Programming Interface

1. The Interface

The Probooker Application Programming Interface (**PAPI**) interface is made up of web services developed in .NET. The functions can be consumed or accessed via a HTTP post.

Service descriptions and WSDL documents can be supplied on request after registration with Do IT Online Ltd.

2. The Functions

The following functions are available through the PAPI interface:

Accounting Web Service (pbAccount)

GetAccountDetail

Retrieve account detail for a given customer.

GetCCTypes

Retrieve list of all payment card types and associated charges.

ProcessSale

Process a sale with Probooker. Sales originated from another source other than EPOS can be forwarded to EPOS systems for stock control and reporting purposes.

Booking Procedures Web Service (pbBookProc)

AddOccupants

Add occupants to a pending booking.

AllowExtras

Allow occupants to purchase extras. Card payments can be supplied and an unsettled payment actioned against the card.

AmendBookingDates

Amend the booking dates of the supplied booking.

AmendOccupant

Amend details of the supplied occupant.

CancelBooking

Cancel supplied booking. Cancellation policy will be automatically applied.

CheckIn

Check In supplied booking.

GetAllBookingDetails

Retrieve details of all bookings for the supplied customer by booking reference.

GetAvailability

Retrieve availability for the supplied location and the appropriate pricing for the supplied dates and times.

GetBookSettings

Retrieve the settings for bookings. I.e. minimum booking period.

GetBookingDetail

Retrieve details of the supplied booking.

GetCurrentLocks

Retrieve a list of all resources currently locked by a booking process.

GetResourceOccupant

Retrieve details of occupant by the supplied booking reference or resource name.

LockPending

Lock a resource during the booking process.

ProcessBooking

Process a booking. Card payments can be automatically processed along with this method.

UnlockPending

Unload a resource locked by a booking process.

Customer Procedures Web Service (pbCustomer)

AmendCustomerDetail

Amend details of supplied customer.

GetCustomerDetail

Retrieve details of supplied customer.

SendUserPassword

Send customer their password.

UserLogin

Login customer using supplied credentials.

UserSignup

Signup customer using supplied credentials.

General Web Service (pbGeneral)

GetExtrasImage

Retrieve details of a product image in Byte format.

GetLocations

Retrieve list of locations and their associated ID's.

GetResourceCategories

Retrieve resource categories.

GetResourceExtras

Retrieve products available for sale via supplied interface.

GetResourceTypes

Retrieve list of resource types.

RegisterSys

Register third party API with Probooker and it's end point URL.

Messaging Web Service (pbMessage)

ResendConfirmation

Resend a booking confirmation by email for the supplied booking.

ResendRegistrationEmail

Resend email with registration details to supplied customer.

ResendSMSConfirmation

Resend a booking confirmation by text message for the supplied booking.

SendEmailMessage

Send an email message to the supplied customer.

SendGenesisMessage

Send message to all pagers

SendSMSMessage

Send an sms message to the supplied customer.

SendUserMessage

Send an internal Probooker message to the supplied user.

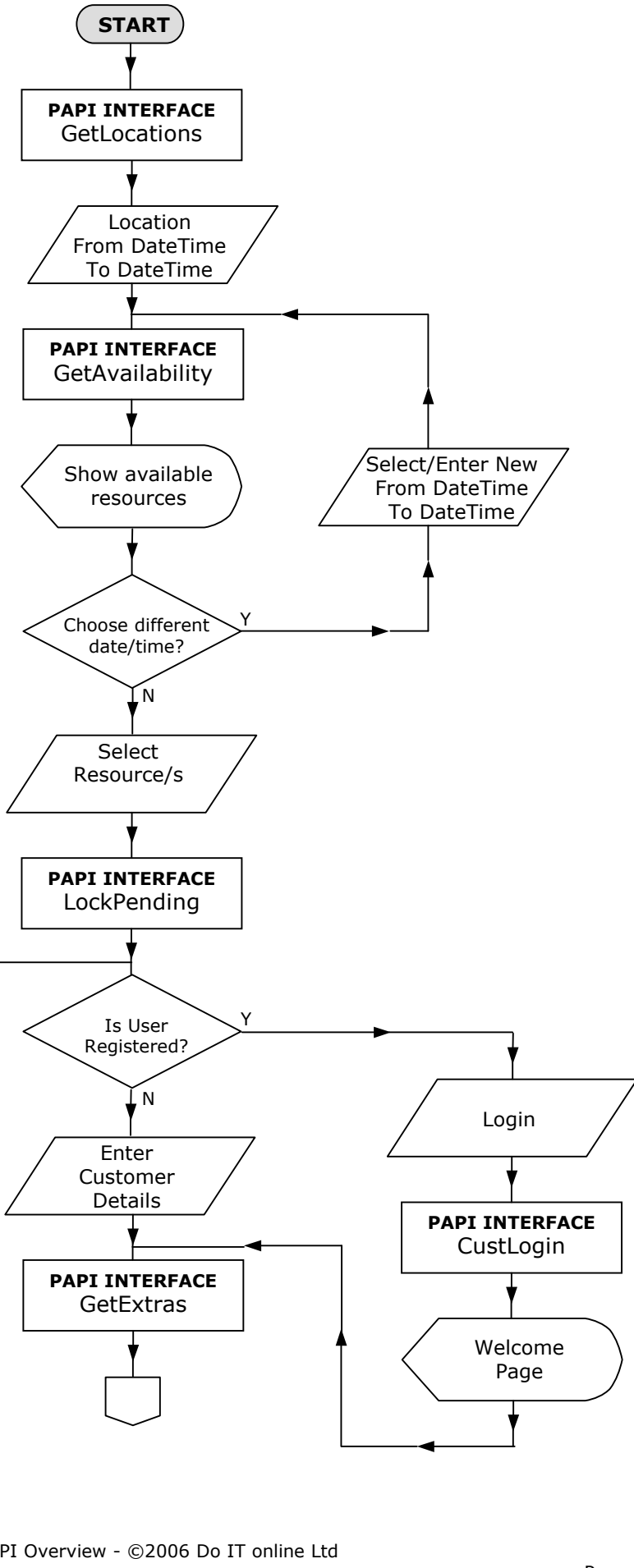
3. The Booking Process

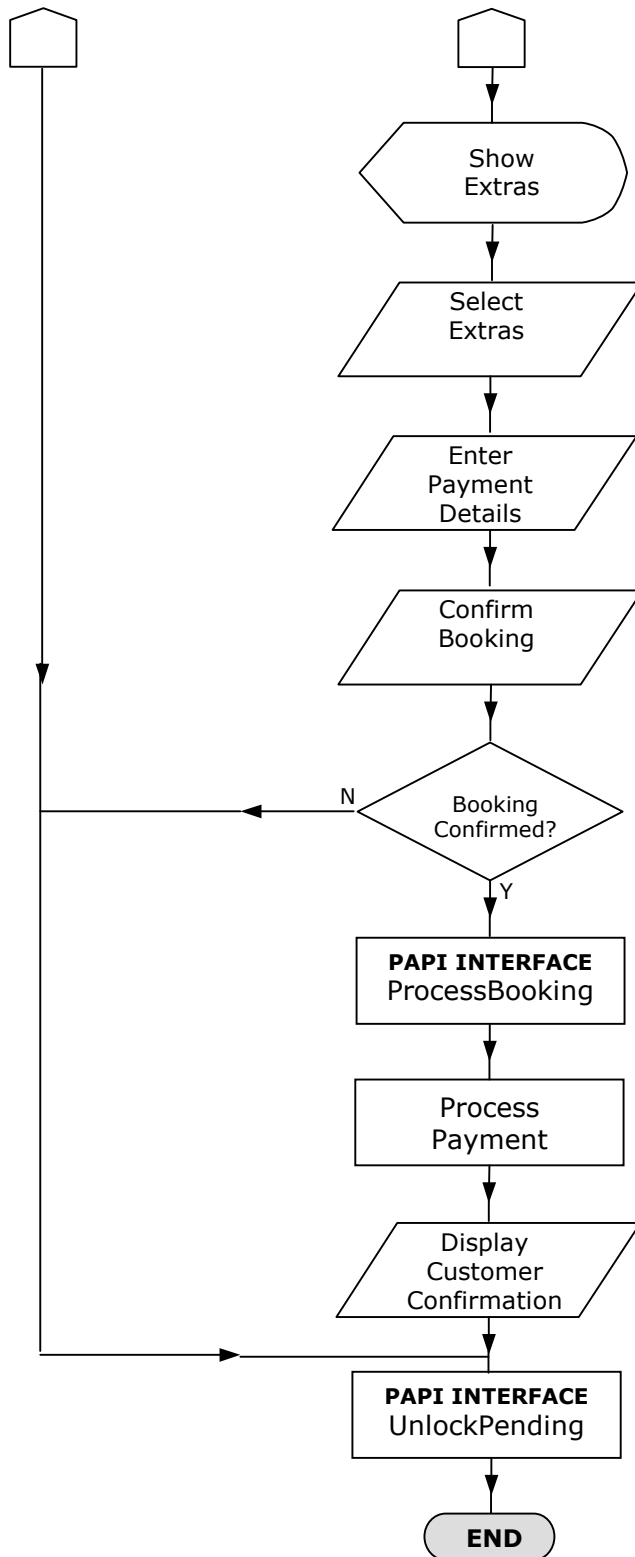
The following diagram illustrates the recommended booking process for integration with Probooker.

Online booking Process Overview

NOTES:

1. This process is for a new booking. An existing customer can have the opportunity to enter a booking reference and follow a different process which will allow amendments and pre-purchase of extras.
2. The reason for the customer to login after checking availability and pricing is to avoid the annoyance of time spent giving login details only to find no resources available. Alternatively, if customer identification affects pricing then this may appear earlier in the process.
3. Locks allow real-time availability to be managed. The LockPending function ensures resource/s matching requirements are locked. These locks are then later removed by the completion of the booking or by calling the UnlockPending function.





4. Example Function

The example shown below is the GetAvailability function.

Request

```

<GetAvailability xmlns="???">
  <APIUser>string</APIUser>
  <APIPass>string</APIPass>
  <strReqID>string</strReqID>
  <strLangID>string</strLangID>
  <lngLocationID>long</lngLocationID>

```

```
<dteArrival>dateTime</dteArrival>
<dteDeparture>dateTime</dteDeparture>
<strResType>string</strResType>
<intNoRequested>int</intNoRequested>
<lngBookRef>long</lngBookRef>
</GetAvailability>
```

Return

```
<?xml version="1.0" encoding="utf-8" ?>
<ResponseGetAvailability xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns="????">
  <ResponseHeader>
    <strReqID>1</strReqID>
    <strRequest>GetAvailability</strRequest>
    <intErrNo>0</intErrNo>
    <strErrNoDesc>OK</strErrNoDesc>
  </ResponseHeader>
  <booPrefAvail>true</booPrefAvail>
  <Resources>
    <ResourceDetail>
      <strResType>Standard</strResType>
      <dteArrival>2008-03-28T10:00:00</dteArrival>
      <dteDeparture>2008-03-28T16:00:00</dteDeparture>
      <decPrice>38</decPrice>
      <intNoAvailable>17</intNoAvailable>
      <intNoAccessible>1</intNoAccessible>
    </ResourceDetail>
    <ResourceDetail>
      <strResType>Premium</strResType>
      <dteArrival>2008-03-28T10:00:00</dteArrival>
      <dteDeparture>2008-03-28T16:00:00</dteDeparture>
      <decPrice>55</decPrice>
      <intNoAvailable>3</intNoAvailable>
      <intNoAccessible>2</intNoAccessible>
    </ResourceDetail>
  </Resources>
</ResponseGetAvailability>
```