What a Booking System Needs

Core Features

Calendar Integration

- Real-time calendar view (daily, weekly, monthly)
- Sync with Google Calendar, Apple Calendar, etc.
- Availability management (block/unblock time)

User Interface

- Clean, intuitive booking page
- Mobile and desktop responsive
- Branded/customizable design

Client Scheduling

- Select service/class/staff
- Choose date and time
- Optional: recurring bookings
- Confirmation notifications (email/text)

Staff & Resource Management

- Assign bookings to specific staff or rooms
- Manage staff availability
- Conflict resolution (avoid double-booking)

Payment Processing

- Integration with Stripe, Square, PayPal, etc.
- Accept deposits or full payments
- Offer promo codes or packages

Client Management

- Client profiles with booking history

- CRM tools (notes, preferences, etc.)
- Intake forms or waivers

Notifications & Reminders

- Automated email/SMS reminders
- Confirmations, cancellations, and reschedules
- Notifications for staff

Admin Features

Dashboard & Analytics

- Booking trends
- Revenue tracking
- No-show rates, client retention

Policies & Restrictions

- Cancellation/rescheduling windows
- Late/no-show fees
- Capacity limits (for classes/events)

Integration & Automation

- CRM (HubSpot, Mailchimp, etc.)
- Zapier or API access
- Membership or subscription systems

Security & Compliance

- GDPR / CCPA compliance
- Secure data handling
- Privacy policy & terms of use enforcement

Optional Add-ons

Waitlist system

Loyalty or rewards program

Multi-location support

Gift cards & promo management

Chat or messaging

Upsells (add-on products/services during booking)

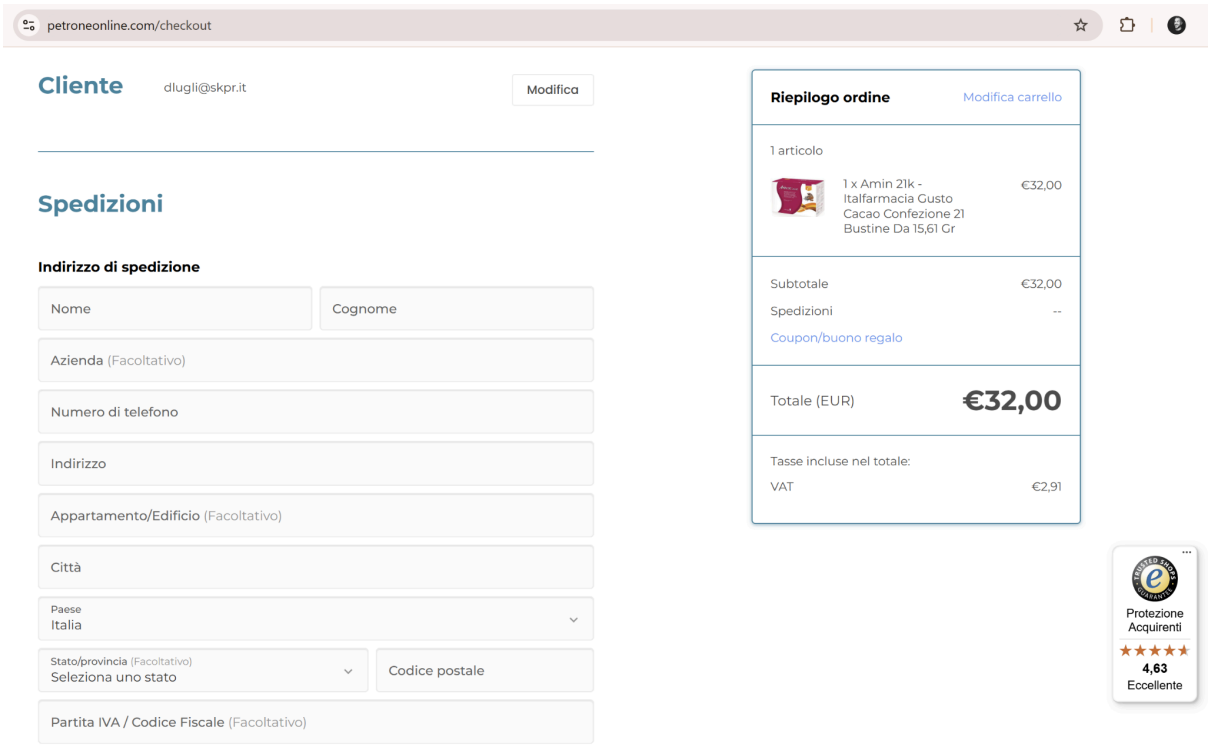
Preliminary Notes

The **Petroneonline CHECKOUT module** was custom-developed by an external company that we are no longer able to contact, so it is **not a native BigCommerce component**.
We can either **replace it** (with the required changes) or **modify the existing module**.

All field labels are currently in Italian — we can provide translations internally if needed.

The activity involves both **frontend** (fields, label formatting) and **backend** (data saving, data transfer to our server).

https://www.petroneonline.com/checkout



Three-Step Checkout with Dynamic Fields

STEP 1: Personal and Shipping Details

- First Name
- Last Name
- Email
- Phone Number
- Address
- Street Number
- ZIP Code
- City
- Province

Field suggestions (auto-complete or validation):

For fields like Address, Street Number, ZIP, City, Province,
we already have **GLS's technical documentation** for their address database.

Cart Contents

If the cart contains **medications, medical devices, or homeopathic products**:
→ show field:
"Enter your Codice Fiscale to receive a tax-deductible receipt (“scontrino parlante”) (optional)"

If the cart does **not** contain such products:
→ show the same field in **greyed-out style** with an **info icon**, showing on hover/tap:
"There are no deductible products in your cart."

Product type detection:

- **Medications**: MINSAN code starts with `0`
- **Homeopathics**: MINSAN code starts with `8`
- **Medical devices**: not identifiable via MINSAN — we may tag them using a **custom field**

STEP 2: Shipping Method

Upon reaching step 2, we want the **account to be automatically registered** in BigCommerce.

Checkbox:
"Use the provided address for shipping"

Courier selection with clear timing and pricing:
(Currently only GLS, but we want to add Amazon Shipping and In-post)

Option for delivery to a pickup point
(We already have all the necessary documentation to integrate GLS and In-post pickup points)

Additional notes for the courier (e.g., intercom info, delivery time slots, etc.)

STEP 3: Payment and Invoice (optional)

Payment methods (currently active):

- Credit/Debit Card
- PayPal
- Scalapay
- Cash on Delivery (add €5 surcharge, and display it in the cart total)
- Bank Transfer

Checkbox:
"I want to receive an invoice"

If selected, show the following fields:

- **VAT number** or **Codice Fiscale**
  → Validate **Italian CF** (e.g., `cf-validator` in JavaScript or `codicefiscale` in Python)
  → Validate **Italian VAT**:
    - 11 digits total
    - First 7 digits: taxpayer ID
    - Digits 8–10: Tax Office code (e.g., 001 = Turin, 003 = Milan)

    - Last digit: control digit (checksum via specific algorithm)
- Company Name
- Address
- Street Number
- ZIP Code
- City
- Province
- Certified Email (PEC) or SDI Code (at least one required)

Other Validations & Notes

Dynamic Functionality

- If the cart has no medications → do not show Codice Fiscale field
- If the customer requests an invoice → only show relevant fields
- Auto-validation of Codice Fiscale and VAT number
- Fully mobile-friendly, with clear instructions and validations

Additional UI Elements

- Display a **fixed order summary** on the right (or top on mobile)
- Clearly highlight: applied discounts, shipping cost, and total
- Clear final CTA: **"Confirm and Pay"**
- Use **clean design**, minimal text, intuitive icons, real-time field feedback (e.g., “Required field”, “Invalid email format”, etc.)

📩 For more information contact:

- **m.iuliano-guest@petroneonline.it**
- **dlugli76@gmail.com**