Community Gift Cards

API Specifications

November 12, 2018

Table of Contents

Revision History

Date	Version	Changes
February 24, 2017	1	Initial revision
February 25, 2017	1.1	Multiple new functions added.
March 2, 2018	1.2	Added Latitude/Longitude as fields to CreateLocation and UpdateLocation, as well as returned as part of GetLocations.
May 5, 2018	1.3	Added 5 custom fields that can be retrieved/updated via API only, for every location. ApiCustom1 … ApiCustom5. Also added capability to retrieve and update MarketPlace discounts for every location.
November 12, 2018	1.4	Added new FindCards function.

Introduction

The Community Gift Cards Application Progamming Interface (API) provides a mechanism whereby external systems are able to interact with the CommunityGiftCards platform in pre-described ways.

This allows for external systems to retrieve information from and/or update information to the platform.

The account owner has full control over what functionality is offered to each partner system, by granting (or not) access to the different “methods” described in this document.

With the use of an API, end-users of those external systems could have access to interact with CommunityGiftCards information, without having to leave that external system's presentation/control flow.

Connectivity / Limitations

The API endpoint is https://www.communitygiftcards.com/secure/cgcapihandler.php

The API expects a single POST variable, named “**xml**”, containing a string representing valid, well-formed XML.

All API function calls are monitored to ensure that you are following best practices and are not subjecting the CommunityGiftCards system to undue “flooding” of requests and traffic.

Should we deem that your system is generating more traffic than is acceptable, you will be contacted to determine the reason, and discussions will occur to determine steps forward for remediation.

The Fine Print

The data that is retrieved from the CommunityGiftCards system, is wholly and entirely owned by Community Gift Cards (algorithms and data relationships – whether explicitly stated or implied), and the account that is providing the access (locations, card balances, historical data, etc.).

No vendor should make any data available, in part or in whole, to any other party, unless specifically granted permission by Community Gift Cards and the account owner.

Functional Overview – Request format

The **xml** parameter should be of the following format:

```
<CGCApiRequest>
  <Username>USERNAME</Username>
  <Password>PASSWORD</Password>
  <ApiKey>APIKEY</ApiKey>
  <Method>METHOD_NAME</Method>
  … other XML tags with additional information
</CGCApiRequest>
```

USERNAME / PASSWORD / APIKEY represent the Username and Password created by the account owner, and the API Key generated by the system. If you have any questions or issues with authentication, please contact the account owner in question.

METHOD_NAME represents the functionality/method that you are trying to access. There are separate sections in this document for each method available.

Functional Overview – Response format

The API handler will return valid, well-formed XML. The **Status** tag should always be checked to verify the results of the API function call. It will either be “ERROR” (indicating an error) or “OK” (indicating success).

Below is an <u>example</u> of a response indicating an error has occurred due to invalid credentials being supplied:

```
<CGCApiResponse>
  <Status>ERROR</ReturnStatus>
  <ErrorCode>006</ErrorCode>
  <ErrorMessage>Invalid credentials provided or INACTIVE API Key</ErrorMessage>
</CGCApiResponse>
```

Below is an <u>example</u> of a successful response to a CreateLocation call:

```
<CGCApiResponse>
  <Status>OK</ReturnStatus>
  <LocationCode>XXX</LocationCode>
</CGCApiResponse>
```

LOCATION Functions

Method: GetLocationTags

This function call allows you to retrieve the list of location categories/tags for the account.

Example Request:

```
<CGCApiRequest>
  <Username>USERNAME</Username>
  <Password>PASSWORD</Password>
  <ApiKey>APIKEY</ApiKey>
  <Method>GetLocationTags</Method>
</CGCApiRequest>
```

Example Response:

```
<CGCApiResponse>
  <Status>OK</ReturnStatus>
  <LocationTagList>
    <LocationTag>Bed and Breakfast</LocationTag>
    <LocationTag>Winery</LocationTag>
    <LocationTag>Wine Tours</LocationTag>
    … more LocationTag fields …
  </LocationTagList>
</CGCApiResponse>
```

Method: GetLocations

This function call allows you to retrieve a list of locations for the account.

Example Request:

```
<CGCApiRequest>
  <Username>USERNAME</Username>
  <Password>PASSWORD</Password>
  <ApiKey>APIKEY</ApiKey>
  <Method>GetLocations</Method>
</CGCApiRequest>
```

Example Response:

```
<CGCApiResponse>
  <Status>OK</ReturnStatus>
  <LocationList>
    <Location>
      <Code>XYZ</Code>
      <Name>ABC Company</Name>
      … other fields
    </Location>
    … more Location collections
  </LocationList>
</CGCApiResponse>
```

Response fields:

Field	Guaranteed ?	Description
Code	Yes	Unique code (alpha-numeric) for the location.
Name	Yes	Indicates the name of the location.
Email	No	E-mail address associated with this location.
WebhookURL	No	The webhook URL associated with this location.
Tags	No	Comma-separated list of tags/categories for this location.
Latitude	No	Latitude for the location.
Longitude	No	Longitude for the location.
IsActive	Yes	TRUE indicates it's an active location, FALSE otherwise.
ApiCustom1 … ApiCustom5	No	Custom fields (alpha-numeric) that can only be retrieved/updated via the API – these are not displayed in any way in the CommunityGiftCards platform.
HasMarketplaceDiscount	Yes	Indicates whether this location has a discount that is advertised in the Marketplace. Either “TRUE” or “FALSE”.
DiscountPercent	Conditional	If HasMarketplaceDiscount is “TRUE”, this field is guaranteed. It will be an integer between 5 and 95, and will be divisible by 5 (ie. 5 % up to 95 %).
DiscountType	Conditional	If HasMarketplaceDiscount is “TRUE”, this field is guaranteed. It will be either “allitems”, “selecteditems” or “other”.
DiscountOtherText	Conditional	If HasMarketplaceDiscount is “TRUE”, and DiscountType is “other”, this field is guaranteed, and will be the text that appears in the Marketplace.
HasActiveDevices	Yes	TRUE indicates there is at least one active device registered in the system for this location, FALSE otherwise.
LastActivationEmail	No	If there has been a device activation e-mail sent to this location, this will be the latest date/time on which that e-mail was sent. Format is ISO8601.
NeedBankInformation	Yes	TRUE indicates that there isn't enough information in the system yet for this location to perform EFT/ACH transactions, FALSE otherwise.
LastDirectDepositEmail	No	If NeedBankInformation is TRUE, and there has been at least one e-mail sent to this location notifying them to upload a direct-deposit form into the system, this will be the latest

		date/time on which that e-mail was sent. Format is ISO8601.

Method: CreateLocation

This function call allows you to create a new location for the account.

Example Request:

```
<CGCApiRequest>
  <Username>USERNAME</Username>
  <Password>PASSWORD</Password>
  <ApiKey>APIKEY</ApiKey>
  <Method>CreateLocation</Method>
  <Location>
    <Name>ABC Company</Name>
    <Email>whatever@whatever.com</Email>
    <WebhookURL>http://yourdomain.com/yourwebhookurl</WebhookURL>
    <Tags>Winery,Wine Tours</Tags>
  </Location>
</CGCApiRequest>
```

Request fields:

Field	Required ?	Description
Name	Required	The name you provide must be UNIQUE for the account.
Email	Required	E-mail address for the location.
WebhookURL	Optional	If specified, must be a valid Webhook URL already defined in the system for the account. Please contact the account owner for the appropriate URL(s).
Tags	Optional	If specified, then every tag (comma-separated) must be already defined in the system for the the account – those can be retrieved by using the GetLocationTags method.
ApiCustom1 … ApiCustom5	Optional	Custom fields … alpha-numeric (255 characters max).
HasMarketplaceDiscount	Optional	Indicates whether this location has a discount that is advertised in the Marketplace. Either “TRUE” or “FALSE”.
DiscountPercent	Conditional	If HasMarketplaceDiscount is “TRUE”, this field is required and must be an integer between 5 and 95, divisible by 5.
DiscountType	Conditional	If HasMarketplaceDiscount is “TRUE”, this field is required and must be one of “allitems”, “selecteditems”, or “other”.
DiscountOtherText	Conditional	If HasMarketplaceDiscount is “TRUE” and DiscountType is “other”, this field is required.
Latitude	Optional	Latitude of the location.
Longitude	Optional	Longitude of the location. Note: if either Latitude or Longitude is provided, then they must **both** be provided. Both must be numeric values when provided.

Example Response:

```
<CGCApiResponse>
  <Status>OK</ReturnStatus>
  <LocationCode>XYZ</LocationCode>
</CGCApiResponse>
```

Use the LocationCode return value to uniquely identify the location in your system. You will need it if you plan to update the location, or to access other location-based methods via the API.

Method: UpdateLocation

This function call allows you to update certain fields about a location for the account.

Example Request:

```
<CGCApiRequest>
  <Username>USERNAME</Username>
  <Password>PASSWORD</Password>
  <ApiKey>APIKEY</ApiKey>
  <Method>UpdateLocation</Method>
  <Location>
    <Code>XYZ</Code>
    <Name>ABC Company</Name>
    <Email>whatever@whatever.com</Email>
    <WebhookURL>http://yourdomain.com/yourwebhookurl</WebhookURL>
    <Tags>Winery,Wine Tours</Tags>
    <IsActive>FALSE</IsActive>
  </Location>
</CGCApiRequest>
```

Request fields:

Field	Required ?	Description
Code	Required	Unique code for the location.
Name	Optional	If provided, must be a unique name for all locations on the account.
Email	Optional	Leave this field out completely if you don't want to change the e-mail address for the location … specify it otherwise. Note that there is no mechanism to "blank out" an e-mail address.
WebhookURL	Optional	If specified, must be a valid Webhook URL already defined in the system for the account. Please contact the account owner for the appropriate URL(s).
Tags	Optional	If specified, then every tag (comma-separated) must be already defined in the system for the the account – those can be retrieved by using the GetLocationTags method.
ApiCustom1 … ApiCustom5	Optional	Custom fields … alpha-numeric (255 characters max).
HasMarketplaceDiscount	Optional	Indicates whether this location has a discount that is advertised in the Marketplace. Either "TRUE" or "FALSE".
DiscountPercent	Conditional	If HasMarketplaceDiscount is "TRUE", this field is required and must be an integer between 5 and 95, divisible by 5.
DiscountType	Conditional	If HasMarketplaceDiscount is "TRUE", this field is required and must be one of "allitems", "selecteditems", or "other".
DiscountOtherText	Conditional	If HasMarketplaceDiscount is "TRUE" and DiscountType is "other", this field is required.
Latitude	Optional	Latitude of the property.
Longitude	Optional	Longitude of the property. Note: if either Latitude or Longitude is provided, then they must **both** be provided. Both must be numeric values when provided.
IsActive	Optional	Leave this field out if you don't want to change the active flag. If specified, must be either TRUE or FALSE to activate/in-activate the location.

Example Response:

```
<CGCApiResponse>
  <Status>OK</ReturnStatus>
</CGCApiResponse>
```

Method: SendDirectDepositEmail

This function call allows you to send an e-mail to a location, notifying them to upload their direct deposit form into the system.

Example Request:

```
<CGCApiRequest>
  <Username>USERNAME</Username>
  <Password>PASSWORD</Password>
  <ApiKey>APIKEY</ApiKey>
  <Method>SendDirectDepositEmail</Method>
  <Location>
    <Code>XYZ</Code>
  </Location>
</CGCApiRequest>
```

Request fields:

Field	Required ?	Description
Code	Required	Unique code for the location.

Example Response:

```
<CGCApiResponse>
  <Status>OK</ReturnStatus>
</CGCApiResponse>
```

Note: If the location already has banking information on file in the system, this function call will return an error. Use the NeedBankInformation field from the GetLocations call to determine if a location requires banking information to be provided.

Method: SendDeviceActivationEmail

This function call allows you to send an e-mail to a location, with instructions to activate their devices.

Example Request:

```
<CGCApiRequest>
  <Username>USERNAME</Username>
  <Password>PASSWORD</Password>
  <ApiKey>APIKEY</ApiKey>
  <Method>SendDeviceActivationEmail</Method>
  <Location>
    <Code>XYZ</Code>
  </Location>
</CGCApiRequest>
```

Request fields:

Field	Required ?	Description
Code	Required	Unique code for the location.

Example Response:

```
<CGCApiResponse>
  <Status>OK</ReturnStatus>
</CGCApiResponse>
```

CARD Functions

Method: GetCards

This function call allows you to retrieve a list of cards for the account.

Example Request:

```
<CGCApiRequest>
  <Username>USERNAME</Username>
  <Password>PASSWORD</Password>
  <ApiKey>APIKEY</ApiKey>
  <Method>GetCards</Method>
  <ActivityTimeRange>last30days</ActivityTimeRange>
  <CardStatus>OPEN</CardStatus>
</CGCApiRequest>
```

Request fields:

Field	Required ?	Description
ActivityTimeRange	Optional	Retrieve cards that have had any activity (bought, topped up, or redeemed) within the specified range. Valid values are “last30days”, “last60days”, “last90days”, “last120days”, “last180days”, “last360days”, or “forever”. If not specified, the default is “forever”.
CardStatus	Optional	Only retrieve those cards whose status is specified by this field. Valid values are: ** NEW – since bought (and/or topped up), no redemptions have occurred ** USED – there is still a balance left on the card, but there has been at least one redemption ** NOBALANCE – the card has been totally used up (the sum of the redemptions, equals the total of the initial value of the card plus any top-ups that have occurred) ** ONHOLD – the card is currently “on hold” If no value is specified, all cards with status of either “NEW” or “USED” will be returned (ie. Not on hold, and a balance exists)

Example Response:

```
<CGCApiResponse>
  <Status>OK</ReturnStatus>
  <CardList>
    <Card>
      <Code>XYZ</Code>
      <Balance>100.00</Balance>
      <CardStatus>NEW</CardStatus>
      <BoughtDateTime>ISO8601date</BoughtDateTime>
      <LastTopUpDateTime>ISO8601date</LastTopUpDateTime>
      <LastRedeemDateTime>ISO8601date</LastRedeemDateTime>
    </Card>
    … more Card collections
  </CardList>
</CGCApiResponse>
```

Note: If there have been no top-ups, LastTopUpDateTime will not be present.
Note: If there have been no redemptions, LastRedeemDateTime will not be present.

Method: GetCardSettings

This function call allows you to retrieve the standard settings in the system for this account, related to cards.

Example Request:

```
<CGCApiRequest>
  <Username>USERNAME</Username>
  <Password>PASSWORD</Password>
  <ApiKey>APIKEY</ApiKey>
  <Method>GetCardSettings</Method>
</CGCApiRequest>
```

Example Response:

```
<CGCApiResponse>
  <Status>OK</ReturnStatus>
  <CardSettings>
    <Currency>CAD</Currency>
    <AllowsPlasticCards>TRUE</AllowsPlasticCards>
    <StandardCardAmountList>
      <StandardCardAmount>100.00</StandardCardAmount>
      <StandardCardAmount>200.00</StandardCardAmount>
      … other amount tags …
    </StandardCardAmountList>
  </CardSettings>
</CGCApiResponse>
```

Response fields:

Field	Guaranteed ?	Description
Currency	Yes	Currency (either CAD or USD) associated with the account. This reflects the currency in which all amounts in the API are reflected for this account.
AllowsPlasticCards	Yes	“TRUE” if the account can ship plastic gift cards, “FALSE” otherwise. Use the GetPlasticCardPrice function to determine if plastic gift cards are available for a specific destination country code.
StandardCardAmountList	Yes	Reflects the list of “acceptable” monetary units that the account accepts for “buying” or “topping up” gift cards. Will always be in NNN.NN format, although the amounts will represent whole units. That is, the “cents” will always be “00”.

Method: CreateCard

This function call allows you to create a new gift card in the system.

Important: This function will NOT process any payments into any sort of payment gateway. It is assumed that the calling system will have processed any payments before-hand, or plans to process the payment once this function returns successfully. Careful attention should be paid to all of the formatting rules, so that the calling system can enforce similar logic before submitting the request, thereby minimizing the chance that a refund needs to be carried out.

Important: Note also that the amount specified, is assumed to be in the currency on the account, which can be retrieved by calling the GetCardSettings function. The amount must also match one of the "StandardCardAmount" values returned by the GetCardSettings function.

Note: This function WILL send an e-mail to the buyer of the card after the card is created. In the system, this new card will be flagged as a card that the buyer bought for him/herself.

Example Request:

```
<CGCApiRequest>
  <Username>USERNAME</Username>
  <Password>PASSWORD</Password>
  <ApiKey>APIKEY</ApiKey>
  <Method>CreateCard</Method>
  <NewCardAmount>100.00</NewCardAmount>
  <CampaignName>My campaign</CampaignName>
  <Email>whatever@whatever.com</Email>
  <Phone>999.999.9999</Phone>
  <FirstName>Chuck</FirstName>
  <LastName>Wagon</LastName>
  <Address1>123 Main Street</Address1>
  <Address2>Apartment 2B</Address2>
  <City>New York</City>
  <StateProvince>NY</StateProvince>
  <CountryCode>US</CountryCode>
  <ZIPPostal>10001</ZipPostal>
</CGCApiRequest>
```

Request fields:

Field	Required ?	Description
NewCardAmount	Required	Must be specified, and must match one of the StandardCardAmount values returned by the GetCardSettings function.
CampaignName	Optional	If you have many campaigns for gift cards, you can use this field to specify which campaign this particular card belongs to. Letters, digits, and spaces only.
Email	Required	Must be provided, and must be of valid format. Note that the system does not validate that the e-mail address actually exists. After the card is created, the system will send an e-mail to the buyer with the specifics of the purchased card. If the e-mail send fails, the system will try again at a later time (usually within a few minutes).
Phone	Optional	Free format text.
FirstName, LastName, Address1, Address2, City, StateProvince, CountryCode, ZIPPostal	See appendix A	Appendix A describes the formatting rules for the address fields.

Example Response:

```
<CGCApiResponse>
  <Status>OK</ReturnStatus>
  <NewCardCode>XYZ</NewCardCode>
</CGCApiResponse>
```

Method: TopUpCard

This function call allows you to add funds to an existing gift card in the system.

Important: This function will NOT process any payments into any sort of payment gateway. It is assumed that the calling system will have processed any payments before-hand, or plans to process the payment once this function returns successfully.

Important: Note also that the amount specified, is assumed to be in the currency on the account, which can be retrieved by calling the GetCardSettings function. The amount must also match one of the “StandardCardAmount” values returned by the GetCardSettings function.

Example Request:

```
<CGCApiRequest>
  <Username>USERNAME</Username>
  <Password>PASSWORD</Password>
  <ApiKey>APIKEY</ApiKey>
  <Method>TopUpCard</Method>
  <Card>
    <Code>XYZ</Code>
  </Card>
  <TopUpAmount>100.00</TopUpAmount>
</CGCApiRequest>
```

Example Response:

```
<CGCApiResponse>
  <Status>OK</ReturnStatus>
  <NewCardBalance>200.00</NewCardBalance>
</CGCApiResponse>
```

Method: FindCards

This function call allows you to determine the cards that are present in the system, for the specific card code “prefix” or “start”. Note that this function ensures that at least 8 characters of the card code are provided. This function returns all of the card codes that start with the code provided – this could result in multiple cards, so if the intent is to determine that a specific card exists, the caller should ask for more characters to be provided.

Example Request:

```
<CGCApiRequest>
  <Username>USERNAME</Username>
  <Password>PASSWORD</Password>
  <ApiKey>APIKEY</ApiKey>
  <Method>FindCards</Method>
  <Card>
    <Code>XYZ</Code>
  </Card>
</CGCApiRequest>
```

Example Response:

```
<CGCApiResponse>
  <Status>OK</ReturnStatus>
  <ValidCardCode>XYZ123</ValidCardCode>
  <ValidCardCode>XYZabc</ValidCardCode>
</CGCApiResponse>
```

Method: GetCardBalance

This function call allows you to determine the current balance for a specific gift card.

Important: Note that the amount returned, is expressed in the currency on the account, which can be retrieved by calling the GetCardSettings function.

Example Request:

```
<CGCApiRequest>
  <Username>USERNAME</Username>
  <Password>PASSWORD</Password>
  <ApiKey>APIKEY</ApiKey>
  <Method>GetCardBalance</Method>
  <Card>
    <Code>XYZ</Code>
  </Card>
</CGCApiRequest>
```

Example Response:

```
<CGCApiResponse>
  <Status>OK</ReturnStatus>
  <CardBalance>100.00</CardBalance>
</CGCApiResponse>
```

Method: RedeemCard

This function call allows you to decrease the balance on an existing gift card in the system, because a gift card holder has redeemed a portion (or all) of the remaining balance.

Important: Note that the amount specified, is assumed to be in the currency on the account, which can be retrieved by calling the GetCardSettings function.

Important: Note that there is no Location code specified here … this means that there will be no funds transfer to the location doing the redemption later. Also, if the card in question is a location-specific card, the API does not have a location to validate against and will assume that the caller is only doing redemptions appropriately.

The amount cannot be greater than the current balance on the card.

Example Request:

```
<CGCApiRequest>
  <Username>USERNAME</Username>
  <Password>PASSWORD</Password>
  <ApiKey>APIKEY</ApiKey>
  <Method>RedeemCard</Method>
  <Card>
    <Code>XYZ</Code>
  </Card>
  <RedeemAmount>6.78</RedeemAmount>
</CGCApiRequest>
```

Example Response:

```
<CGCApiResponse>
  <Status>OK</ReturnStatus>
  <NewCardBalance>93.22</NewCardBalance>
</CGCApiResponse>
```

Method: GetPlasticCardPrice

This function call allows you to determine, given a destination country, the price that should be charged to the customer for shipping a plastic gift card.

If the account does not offer plastic gift cards, an error message will be returned.

Also, if the account does not ship to the destination country in question, an error message will be returned.

Important: Note that the amount returned, is expressed in the currency on the account, which can be retrieved by calling the GetCardSettings function. Some accounts may not charge for plastic gift cards when shipped domestically, as an example, so the amount returned could in fact be zero (0.00) – this is a valid return amount.

Example Request:

```
<CGCApiRequest>
  <Username>USERNAME</Username>
  <Password>PASSWORD</Password>
  <ApiKey>APIKEY</ApiKey>
  <Method>GetPlasticCardPrice</Method>
  <DestinationCountryCode>US</DestinationCountryCode>
</CGCApiRequest>
```

Note: DestinationCountryCode must be a valid 2-character code, per ISO3166. See GetCountryCodes.

Example Response:

```
<CGCApiResponse>
  <Status>OK</ReturnStatus>
  <PlasticCardPrice>5.00</PlasticCardPrice>
</CGCApiResponse>
```

Method: GetPlasticCardStatus

This function call allows you to determine the status of the plastic gift card.

Example Request:

```
<CGCApiRequest>
  <Username>USERNAME</Username>
  <Password>PASSWORD</Password>
  <ApiKey>APIKEY</ApiKey>
  <Method>GetPlasticCardStatus</Method>
  <Card>
    <Code>XYZ</Code>
  </Card>
</CGCApiRequest>
```

Example Response:

```
<CGCApiResponse>
  <Status>OK</ReturnStatus>
  <QueuedDateTime>ISO8601date</QueuedDateTime>
  <PrintedDateTime>ISO8601date</PrintedDateTime>
  <ShippedDateTime>ISO8601date</ShippedDateTime>
</CGCApiResponse>
```

Response fields:

Field	Guaranteed ?	Description
QueuedDateTime	Yes	Date/Time (ISO8601 format) that the plastic gift card was ordered/queued for processing.
PrintedDateTime	No	If the plastic gift card was printed, the date/time (ISO8601 format) that the printing occurred.
ShippedDateTime	No	If the plastic gift card was shipped, the date/time (ISO8601 format) that the shipping occurred. Depending on the destination country, the actual arrival date of the plastic gift card will be some time after the date/time specified here.

Method: OrderPlasticCard

This function call allows you to place an order for a plastic gift card.

If the account does not offer plastic gift cards, an error message will be returned.

Also, if the account does not ship to the destination country in question, an error message will be returned.

If a plastic gift card has already been ordered for the card in question, an error is returned.

Important: Note that the amount provided (ShippingAmount), is assumed to be expressed in the currency on the account (which can be retrieved by calling the GetCardSettings function). It should match the amount that would be returned by the GetPlasticCardPrice function, or an error is returned.

Important: This function will NOT process any payments into any sort of payment gateway. It is assumed that the calling system will have processed any payments before-hand, or plans to process the payment once this function returns successfully. Careful attention should be paid to all of the formatting rules, so that the calling system can enforce similar logic before submitting the request, thereby minimizing the chance that a refund needs to be carried out.

Example Request:

```
<CGCApiRequest>
  <Username>USERNAME</Username>
  <Password>PASSWORD</Password>
  <ApiKey>APIKEY</ApiKey>
  <Method>OrderPlasticCard</Method>
  <Card>
    <Code>XYZ</Code>
  </Card>
  <ShippingAmount>5.00</ShippingAmount>
  <FirstName>Chuck</FirstName>
  <LastName>Wagon</LastName>
  <Address1>123 Main Street</Address1>
  <Address2>Apartment 2B</Address2>
  <City>New York</City>
  <StateProvince>NY</StateProvince>
  <CountryCode>US</CountryCode>
  <ZIPPostal>10001</ZipPostal>
</CGCApiRequest>
```

Request fields:

Field	Required ?	Description
ShippingAmount	Required	Must be specified, and must match the amount returned by GetPlasticCardPrice for the country code in question.
FirstName, LastName, Address1, Address2, City, StateProvince, CountryCode, ZIPPostal	See appendix A	Appendix A describes the formatting rules for the address fields. NOTE: If you have a different name (FirstName, LastName) than what was on the original CreateCard function, please note that the “cover letter” that may/may not be included in the mail with the plastic card, will in fact address the **buyer's** name (from CreateCard).

Example Response:

```
<CGCApiResponse>
  <Status>OK</ReturnStatus>
</CGCApiResponse>
```

MISCELLANEOUS Functions

Method: GetCountryCodes

This function is provided for your convenience, and provides you with the list of ISO3166 country codes. One of these country codes should be provided in any requests for CreateCard or OrderPlasticCard.

(the same list can be found here => http://en.wikipedia.org/wiki/ISO_3166-1_alpha-2)

Example Request:

```
<CGCApiRequest>
  <Username>USERNAME</Username>
  <Password>PASSWORD</Password>
  <ApiKey>APIKEY</ApiKey>
  <Method>GetCountryCodes</Method>
  <SortBy>name</SortBy>
</CGCApiRequest>
```

SortBy is either “name” (to sort by country name) or “code” (to sort by country code) – the default, if not provided, is “name”.

Example Response:

```
<CGCApiResponse>
  <Status>OK</ReturnStatus>
  <CountryList>
    <Country>
      <Code>CA</Code>
      <Name>Canada</Name>
    </Country>
    <Country>
      <Code>US</Code>
      <Name>United States</Name>
    </Country>
    … other Country collections ...
  </CountryList>
</CGCApiResponse>
```

Method: GetStatesUS

This function is provided for your convenience, and provides you with the list of state codes/names for the United States (country code = “US”). Whenever a method requires a country code and a state/province, and the country code selected is “US”, then the state/province should be the 2-character code (not the full name) as provided here.

Example Request:

```
<CGCApiRequest>
  <Username>USERNAME</Username>
  <Password>PASSWORD</Password>
  <ApiKey>APIKEY</ApiKey>
  <Method>GetStatesUS</Method>
</CGCApiRequest>
```

Note: The results will be sorted by state name.

Example Response:

```
<CGCApiResponse>
  <Status>OK</ReturnStatus>
  <StateList>
    <State>
      <Code>AL</Code>
      <Name>Alabama</Name>
    </State>
    <State>
      <Code>AK/Code>
      <Name>Alaska</Name>
    </State>
    … other State collections ...
  </StateList>
</CGCApiResponse>
```

Method: GetProvincesCA

This function is provided for your convenience, and provides you with the list of province codes/names for Canada (country code = “CA”). Whenever a method requires a country code and a state/province, and the country code selected is “CA”, then the state/province should be the 2-character code (not the full name) as provided here.

Example Request:

```
<CGCApiRequest>
  <Username>USERNAME</Username>
  <Password>PASSWORD</Password>
  <ApiKey>APIKEY</ApiKey>
  <Method>GetProvincesCA</Method>
</CGCApiRequest>
```

Note: The results will be sorted by province name.

Example Response:

```
<CGCApiResponse>
  <Status>OK</ReturnStatus>
  <ProvinceList>
    <Province>
      <Code>AB</Code>
      <Name>Alberta</Name>
    </Province>
    <Province>
      <Code>BC/Code>
      <Name>British Columbia</Name>
    </Province>
    … other Province collections ...
  </ProvinceList>
</CGCApiResponse>
```

Appendix A: Address Verification

There are 2 instances when address information is verified: when the CreateCard method is called, or when the OrderPlasticCard method is called. In either case, the field requirements are identified below:

Request fields:

Field	Required ?	Formatting Rules
FirstName	Required	Letters or spaces only (no digits).
LastName	Required	Letters, spaces, hyphens, and single-quotes only (no digits). Example: O'Brien-O'Leary is valid.
Address1	Required	Free format text.
Address2	Optional	Free format text.
City	Required	Free format text.
StateProvince	Required	Letters or spaces only (no digits). In addition, if CountryCode is one of "US" or "CA", then it must be a valid state/province code (not the full name) as returned by GetStatesUS or GetProvincesCA, respectively.
CountryCode	Required	Must be a country code as returned by GetCountryCodes.
ZIPPostal	Required	Letters, digits, spaces, or hyphens. In addition, if CountryCode is "US": - format is either nnnnn or nnnnn-nnnn (where "n" represents a digit) In addition, if CountryCode is "CA": - format is ananan or ana nan (with a space) (where "a" represents a letter, and "n" represents a digit) - letters D, F, I, O, Q, and U are not allowed - cannot start with W or Z

Note: While the system does check that the format of the provided fields is correct per the above rules, it does **not** check that the address is a valid physical address. That is, if you provide a fictitious city name, for example, and it meets the format checking rules above, the address verification/validation checks will pass.