

Mobile Application Store

Product API Design Document

Date: 10/9/2013

Author: Siddharth Singh

Reviewer(s): Hemant BAJPAI

Introduction

This document provides design details of Mobile Application store Product API.

Overview

Mobile application product API provides users, application developers and administrators to add and search contents in application store. It also enables administrators to add parameters related to contents. These parameters are countries and devices. Content can be downloaded in certain countries and content is compatible with certain set of devices.

Requirements

1. Product service provides API to add countries by sending country related parameters (country code, country name and exports status).
2. Product service provides API to add devices by sending device related parameters (device id, device name and manufacturer name).

3. Product service provides API to add content by sending content related parameters (content id, content name, content description, author's name, content rating, content categories, Countries in which content is available, devices content is compatible with, price, languages and image URL).
4. Product service provides API to search content in the store by sending parameters of zero or more search criteria (category list, text search, minimum rating, language list, country code, device id, content type list).

Use Cases

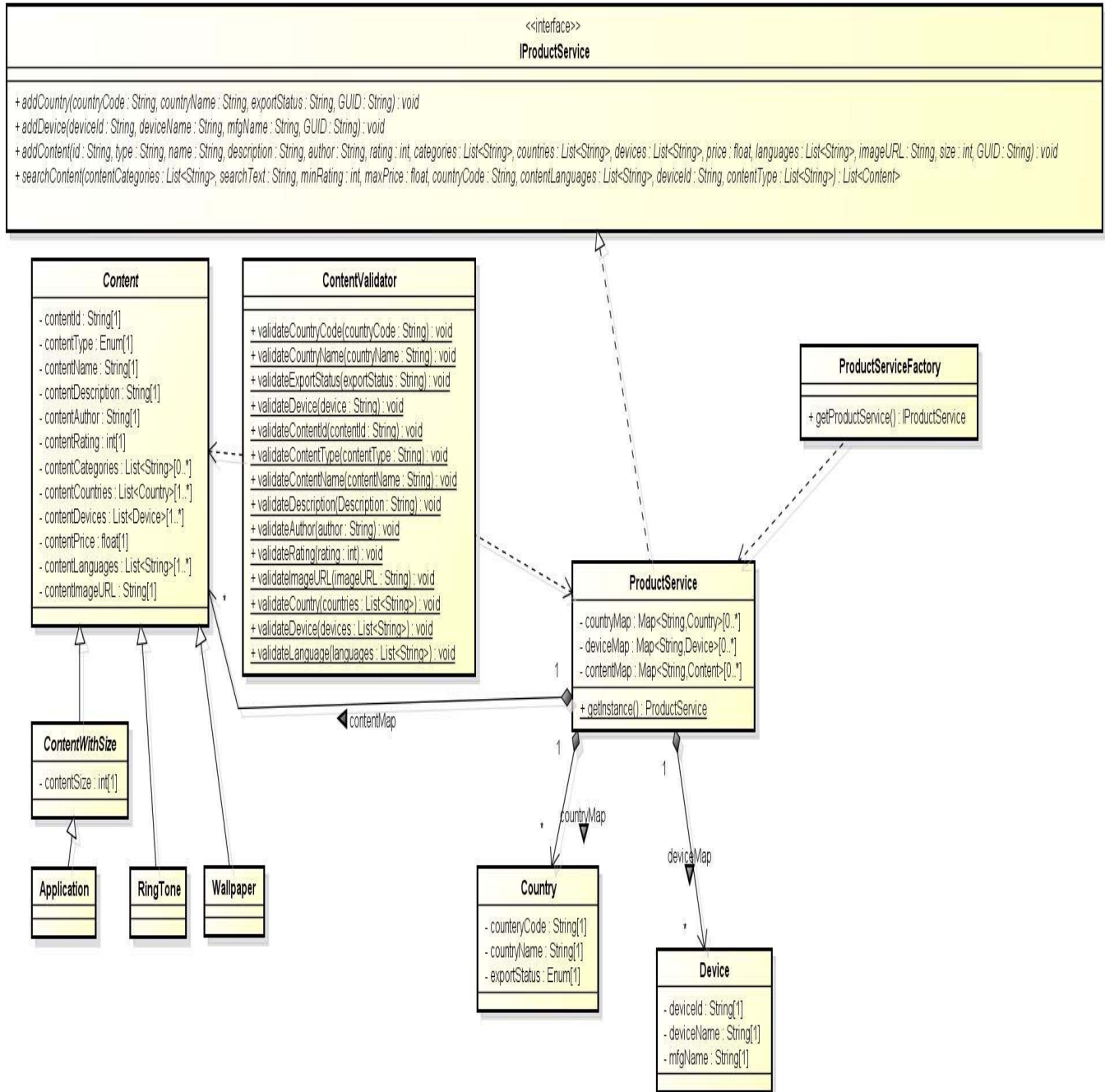
There are three types of users: Administrators, Developers and Consumers.

- Administrator can add countries and devices as well as add/modify and search contents.
- Developer can add, modify and search contents.
- Consumers can search contents.

Product API requires admin rights to add countries and devices and developer rights to add/modify contents. Admin also has developer rights.

Implementation

Class Diagram (Please see ProductAPIServiceClassDiagram.jpg or ProductAPIServiceClassDiagram.asta in the submission folder for the clear class diagram). The following class diagram defines the Product API implementation classes contained within package "cscie97.asn2.ecommerce.product". (Please see ProductAPIServiceClassDiagram.jpg or ProductAPIServiceClassDiagram.asta in submission folder for the clear class diagram).



Class Dictionary

This section specifies the class dictionary for the Product API. The classes should be defined within the package “cscie97.asn2.ecommerce.product”.

ProductServiceFactory

The public class which acts as a factory of Product service object. This class has a factory method which returns handle to product service singleton object.

IProductService <<interface>>

This is a public interface which declares APIs exposed by Product API service. ProductService implements and defines all the methods of this interface.

ProductService

This class implements all the methods of product service interface. This class has package visibility. This class composes a hash map for countries (with country code as key), a hash map for devices (device id as key) and a hash map for content objects with content id as key. Keys are stored in uppercase in order to make respective searches case insensitive. These composed objects are added in memory only if does not already exist. The content is searched in the data structure mentioned above given a search criteria.

Associations

Association Name	Type	Description
countryMap	Map<String, Country>	Stores country objects with country code as key. Keys are stored in uppercase letters.
deviceMap	Map<String, Device>	Stores device objects with device id as key. Keys are stored in uppercase letters.
contentMap	Map<String, Content>	Stores content objects with content id as key. Keys are stored in uppercase letters.

Methods

Method Name	Signature	Description
addCountry	(String countryCode, String countryName, String exportStatus, String GUID) throws ContentParameterException : void	Adds country to countryMap, validates properties before inserting and also checks if country is already in the map. Inserts only if properties are valid otherwise throws ContentParameterException. Does nothing if country already exists in the map.

addDevice	(String deviceId, String deviceName,String mfgName,String GUID) : void	Adds device to deviceMap, validates properties before inserting and also checks if device is already in the map. Inserts only if properties are valid otherwise throws ContentParameterExcep tion. Does nothing if device already exists in the map.
addContent	(String contentId,String contentType, String contentName,String contentDescription, String contentAuthor, int contentRating,List<String> contentCategories,List<String> contentCountries,List<String> contentDevices, float contentPrice,List<String> contentLanguages, String contentImageUrl,int contentSize,String GUID) throws ContentParameterException : void	Adds content to contentMap, validates properties before inserting and also checks if content is already in the map. Inserts only if properties are valid otherwise throws ContentParameterExcep tion. Does nothing if content already exists in the map.
searchContent	(List<String> contentCategories, String searchText, int minRating,float maxPrice, String countryCode, List<String> contentLanguages, String deviceId, List<String> contentType) : List<Content>	Search contents in the contentMap based on the search criteria given as arguments of this function. Returns list of contents which match the search criteria.

ContentValidator

This auxiliary class defines some static functions to validate properties of country, device and content.

Methods

Method Name	Signature	Description
validateCountryCode <<static>>	(String countryCode) throws ContentParameterException : void	Validates if country code is 2 letter word, throws ContentParameterException otherwise.
validateCountryName <<static>>	(String countryName) throws ContentParameterException : void	Validates if country name is not blank, throws ContentParameterException otherwise.
validateExportStatus <<static>>	(String exportStatus) throws ContentParameterException : void	Validates if export status is either open or closed. Throws ContentParameterException otherwise.
validateDevice <<static>>	(String deviceName) throws ContentParameterException : void	Validates if device name is not blank, throws ContentParameterException otherwise.
validateContentId <<static>>	(String contentId) throws ContentParameterException : void	Validates if content id is not blank, throws ContentParameterException otherwise.
validateContentType <<static>>	(String contenttype) throws ContentParameterException : void	Validates if content type is not blank, throws ContentParameterException otherwise.
validateContentName <<static>>	(String contentname) throws ContentParameterException : void	Validates if content name is not blank, throws ContentParameterException otherwise.

validateContentDescription <<static>>	(String contentDesc) throws ContentParameterException : void	Validates if content description is not blank, throws ContentParameterExc eption otherwise.
validateContentAuthor <<static>>	(String contentAuthor) throws ContentParameterException : void	Validates if content author is not blank, throws ContentParameterExc eption otherwise.
validateCountry <<static>>	(List<String> contentCountry) throws ContentParameterException : void	Validates if country list is not empty, throws ContentParameterExc eption otherwise.
validateContentDevices <<static>>	(List<String> devices throws ContentParameterException : void	Validates if device list is not empty, throws ContentParameterExc eption otherwise.
validateContentLanguages <<static>>	(List<String> languages throws ContentParameterException : void	Validates if language list is not empty, throws ContentParameterExc eption otherwise.
validateContentiamgeURL <<static>>	(String imageUrl) throws ContentParameterException : void	Validates if url is not blank, throws ContentParameterExc eption otherwise.
validateMinRating <<static>>	(String minrating) throws ContentParameterException : void	Validates if rating is between 0 and 5, throws ContentParameterExc eption otherwise.

Country

This class represents country objects and encapsulates all the properties of a country (code, name, export status).

Device

This class represents device objects and encapsulates all the properties of a device (id, name, manufacturer name).

Content <<abstract>>

This abstract class represents content objects and encapsulates all the properties of a content which are common to Application, Wallpaper and Ringtone.

ContentWithSize <<abstract>>

This abstract class is derived from Content and declares content size property. This is inherited by all types of contents which has a size.

Application

This is a concrete and specialized class inherited from ContentWithSize and represents Application objects.

RingTone

This is a concrete and specialized class inherited from Content and represents Ringtone objects.

Wallpaper

This is a concrete and specialized class inherited from Content and represents Wallpaper objects.

Implementation Details

ProductService class composes country, device and content map. Content keeps the list of countries and device references of the composed country and device objects.

Properties of country, device and content are validated before adding to the corresponding data structures. If properties are invalid then program throws ContentParameterException.

The policies of search for different search criteria are mentioned below.

1. Categories: If there is any common category between contents and categories in query.
2. Text Search: If name and description of content contains the search text. The comparison is case insensitive.
3. Minimum rating: Content's rating should be greater than or equal to given rating.
4. Maximum price: Content's price should be less than or equal to give price.
5. Languages: If there is any common language between content's languages and languages in query.
6. Country code: If content contains the country given in its list of countries.
7. Device Id: If content contains the device given in its list of devices.
8. Content types list: If content type is one the types in given list.

There is an "OR" relation between above mentioned criteria. Content is added to result set if any one of the above mentioned conditions are met.

Testing

TestDriver class contains main() and calls importCountries(), importDevices(), importContent() in order to import countries, devices,

contents from their csv files and calls `searchContents()` to run query file and search product contents.