Telecom Plan Optimization - Documentation

Overview

This module provides a system to evaluate multiple telecom plans against a user's usage profile and OTT (Over-The-Top media) service requirements. It calculates the estimated monthly cost ("quote") for each eligible plan, considering voice, SMS, data consumption, and OTT subscriptions included in the plan. It then recommends the most cost-effective plan meeting the customer's OTT needs.

Problem Statement

Given diverse prepaid telecom plans with varying prices, validity periods, data/voice/SMS quotas, and bundled OTT subscriptions, a user wants to find:

- Which plans meet their OTT platform requirements
- The estimated monthly cost for their particular usage pattern (voice minutes, SMS count, data used)
- Which plan offers the best value (lowest total cost) considering OTT requirements and usage

The system must normalize plans with different validity durations to a common 30-day equivalent for consistent comparison, prorate included quotas, calculate over-usage charges, and apply any unlimited allowances.

Core Concepts

OTT Services

- Supported OTT platforms: Netflix, Hotstar, Prime, and Spotify
- Users specify their required OTT services, which plans must cover fully to be eligible

User Usage Profile

- Voice call duration in minutes
- Number of SMS messages sent
- Data consumed in megabytes
- Required OTT subscriptions

System Components

Data Classes

- OTTRequirement: Contains boolean flags for each OTT service required by the user. Includes helper to get the set of required OTT names.
- Usage: Encapsulates the user's actual usage data and OTT requirements.
- PlanQuote: Contains detailed cost breakdown for a single plan, including rental (normalized to 30 days), overage costs for data/voice/SMS, and the total monthly cost. Provides a formatted summary string.

Abstract Plan Class (`Plan`)

- Defines common interface and state for telecom plans: name, price, validity, and OTT bundles.
- Contains methods to check OTT compatibility, compute prorated 30-day rental, and determine unlimited allowances for data, voice, SMS.
- Subclasses implement the `quote` method which, given a `Usage`, calculates the total cost.

Concrete Plan Implementations

- Each telecom plan is represented as a subclass of `Plan` overriding `quote` and unlimited feature queries as appropriate.
- Plans differ in rental price, data/voice/SMS quotas, OTT bundles, rates for over-usage, and validity duration.
- Some plans offer unlimited data, unlimited voice, or unlimited SMS, impacting cost calculations.

Pricing & Overages

- Rental price is prorated to a standard 30-day period regardless of actual plan validity.
- Data overage cost is computed in 10MB blocks at a fixed rate.
- Voice and SMS overages are calculated based on prorated quotas and predefined rates.
- Unlimited features result in zero overage costs for the respective service.

Plan Optimization (`PlanOptimizer`)

- Aggregates a set of available plans.
- Filters eligible plans based on OTT subscription requirements.
- Computes cost quotes for eligible plans based on usage.
- Recommends the plan with the lowest total cost, with tie-breakers favoring plans with more OTT services and more unlimited features.

Usage Example

1. Define a `Usage` object with voice minutes, SMS count, data usage in MB, and OTT requirements.
2. Instantiate `PlanOptimizer` with a predefined list of plans.
3. Call `eligible_plans` to get plans meeting OTT needs.
4. Call `quotes_for` to get pricing breakdown for those plans.
5. Call `recommend` to get the best plan and print detailed cost breakdowns.

Extensibility

- New plans can be added by subclassing `Plan` and implementing the `quote` method accordingly.
- Additional OTT platforms can be included by extending the `OTT` class and updating the `OTTRequirement`.
- Pricing rules and overage rates can be easily adjusted or extended in individual plan classes.
- The system is organized to facilitate adding new unlimited feature types or discount schemes.

Assumptions & Notes

- Data usage is measured in megabytes and prorated according to validity.
- Voice and SMS quotas and rates are also prorated to a 30-day standard to enable fair comparison.
- Overage blocks for data are charged per 10MB increments, rounded up.
- Unlimited plans have zero overage charges for respective service types.

- OTT requirements are strictly enforced — plans must include all required OTT subscriptions to qualify.

Example Output (Based on Sample Usage)

Needs:

```python
if __name__=="__main__":
    need = Usage(
        voice_minutes = 650,
        sms_count=300,
        data_mb = 8*GB_TO_MB,

ott_requirement=OTTRequirement(netflix=False,prime=False,hotstar=False,spotify=False)
    )
```

Ouput:
Per plan 30day breakdown:

Plan: Basic Lite

   Rental(30d): Rs266.79

   Data over: Rs0.0

   Voice extra: Rs407.14

Sms extra: Rs60.0

----

total:Rs733.93

Plan: Saver 30

   Rental(30d): Rs499.0

   Data over: Rs0.0

   Voice extra: Rs262.5

Sms extra: Rs40.0

----

total:Rs801.5

Plan: Unlimited Talk 30

   Rental(30d): Rs650.0

   Data over: Rs215.6

   Voice extra: Rs0.0

Sms extra: Rs0.0

----

total:Rs865.6

Plan: Data Max 20

   Rental(30d): Rs1123.5

   Data over: Rs0.0

   Voice extra: Rs375.0

Sms extra: Rs0.0

----

total:Rs1498.5

Plan: Student Stream 56

   Rental(30d): Rs435.0

   Data over: Rs0.0

   Voice extra: Rs262.5

Sms extra: Rs20.0

----

total:Rs717.5

Plan: Premium Ultra 30

   Rental(30d): Rs2999.0

   Data over: Rs0.0

   Voice extra: Rs0.0

Sms extra: Rs0.0

----

total:Rs2999.0

Plan: Test 9th Plan

   Rental(30d): Rs299.0

   Data over: Rs0.0

   Voice extra: Rs0.0

Sms extra: Rs0.0

----

total:Rs299.0

Plan: Family Share 30

   Rental(30d): Rs535.71

   Data over: Rs0.0

   Voice extra: Rs0.0

Sms extra: Rs0.0

----

total:Rs535.71

Plan: Data Max Plus 30

   Rental(30d): Rs1499.0

   Data over: Rs0.0

   Voice extra: Rs262.5

Sms extra: Rs20.0

----

total:Rs1781.5

Recommended:Test 9th Plan - Total Rs299.0

Reason: Cheapest among plans that meet the OTT Requirement(s)Bundled Otts: prime

PS C:\Users\johangeorge.e\Desktop\Python test>