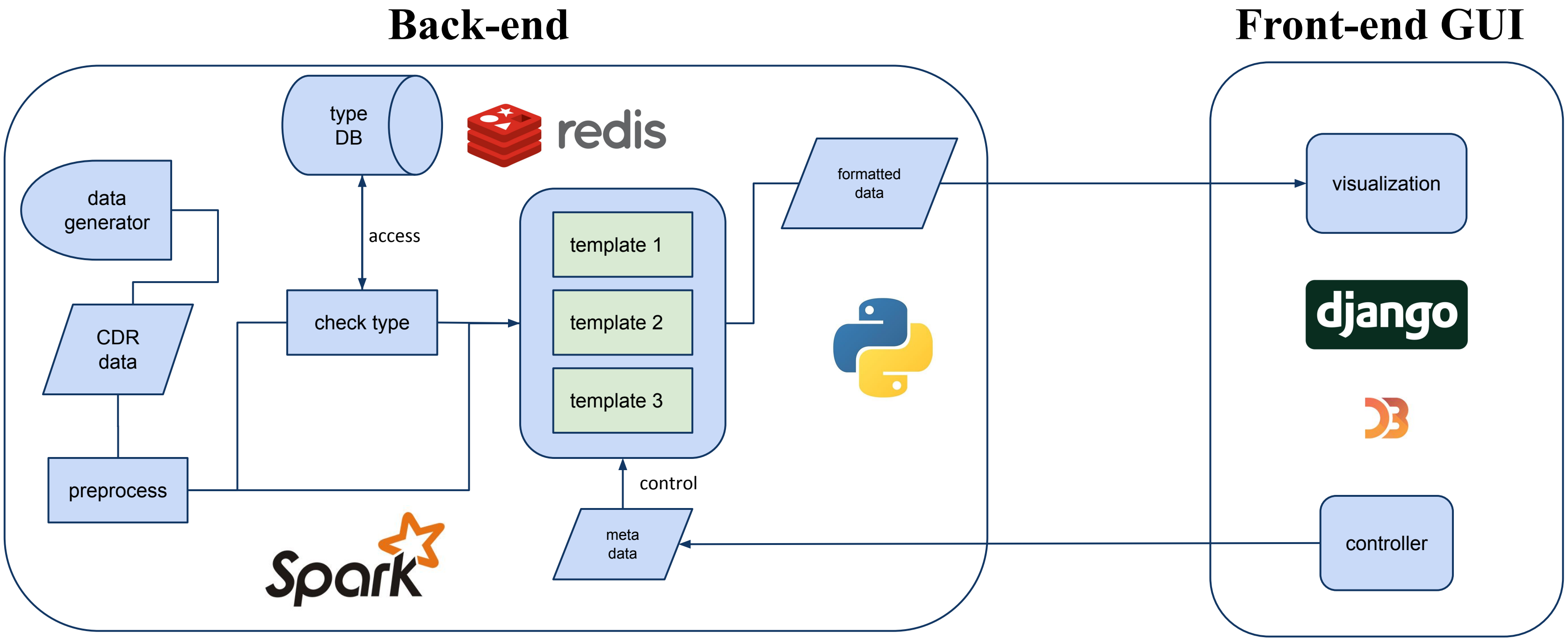
A Rule-based Marketing Platform to Manage Call Detail Record

Chong Hu (ch3467)
Wenjie Chen (wc2685)
Yanchen Liu (yl4189)
Jiajing Sun (js5504)

Problem Definition

- Develop rule-based marketing platform
  - Call Detail Record (CDR)
    - include **caller and callee** numbers + **time and duration** of the call
    - include **region** and **business types** (banking, housing, traveling, etc.)
  - **Match services** with calling profile of customers
  - **Modularization** : design components for data ingest, processing, egress as separate composite operators.
  - **Flow composition** : incorporate static, dynamic, nested composition portions

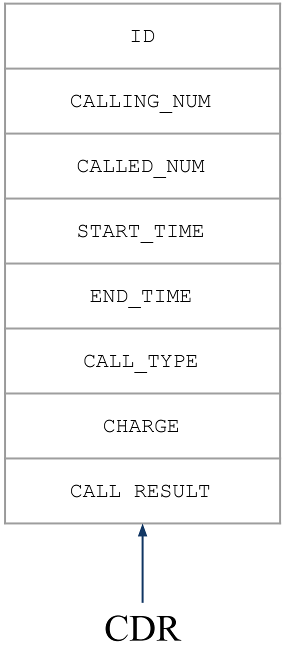
System Structure



Data Generator

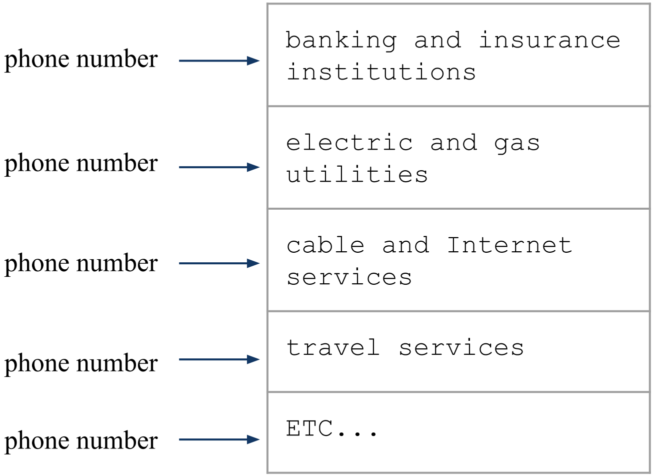
Call Detail Record (CDR)

Produced by data generator



Types of Business for phone number:

Stored in database



phone number
banking and insurance institutions
phone number
electric and gas utilities
phone number
cable and Internet services
phone number
travel services
phone number
ETC...

How was streaming used

How was streaming used

Our Solution

templates introduction

Final Results

Final Results

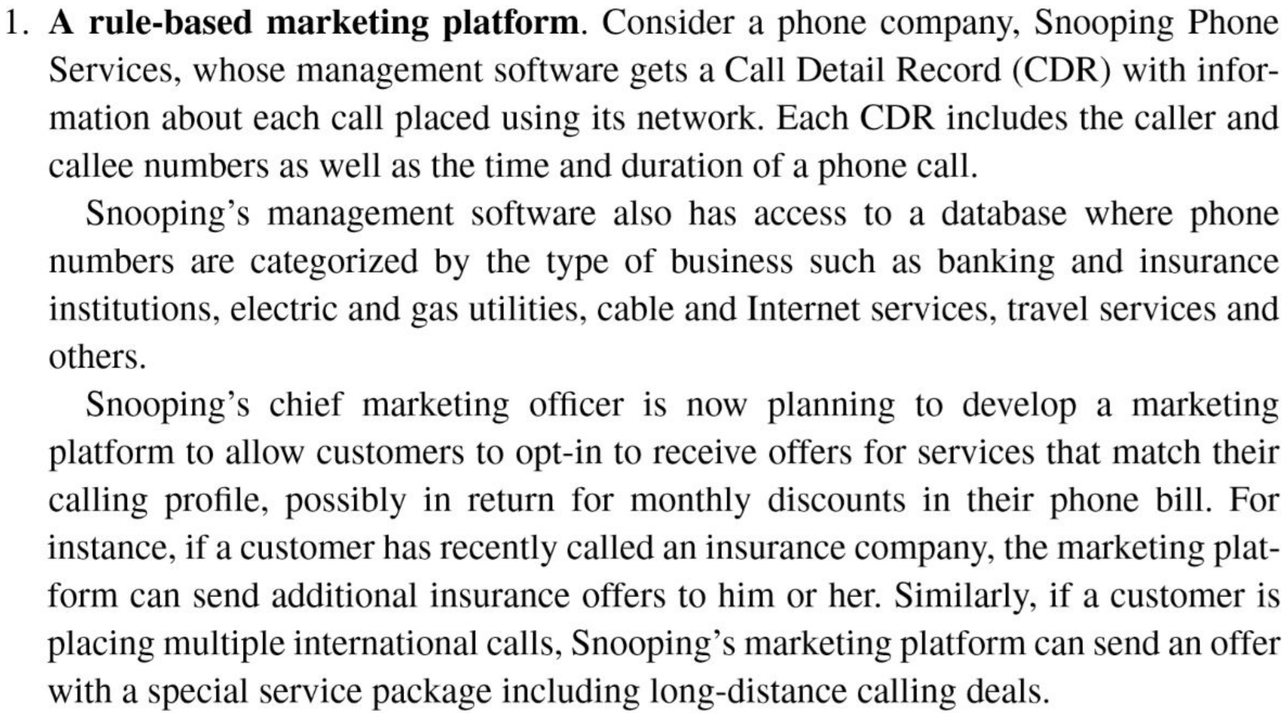
Thank you

Feedback from professor

For the demo to be effective - it will help to see the following:

1. The ability to generate CDRs with some distributions - where you pick the types of phone numbers (e.g. different types of businesses/residential/government), the frequency of calling etc.
2. **Live CDR processing**, so maybe you can show a **running counter** of the type of the CDR continuously - or more interestingly a graph with nodes being the types of phone numbers, and edges being the number of calls between them. Be creative here.
3. Interesting marketing rules - rules that measure some windowed properties of the call patterns to generate a promotion.
4. Tunability - I would like to be able to tune the workload generator from the GUI - so **change the distribution** (maybe you can have a few 2-3 settings) and see the effect of the promotions, i.e if there are more calls to credit card companies, generate promotion for new credit card offer. Think of types of companies that people call, and what types of promotions are possible. For the demo to be effective the result of the change should appear in the frontend at a **granularity of around 15-30 seconds** or so.
5. Similarly it would be nice to **add a new rule** to see what happens as a result.

Rule-based marketing platform

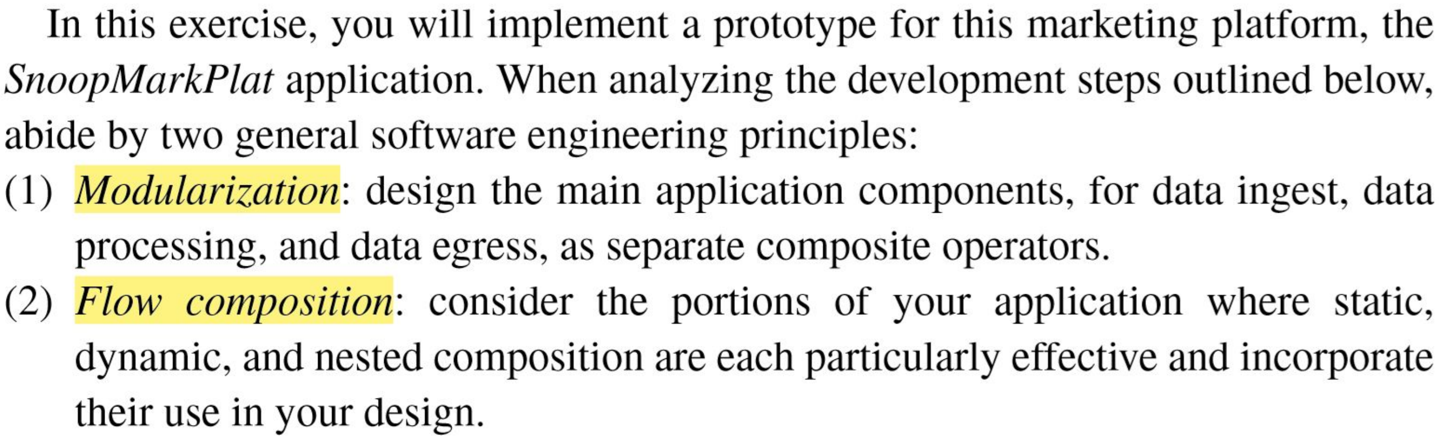


1. **A rule-based marketing platform**. Consider a phone company, Snooping Phone Services, whose management software gets a Call Detail Record (CDR) with information about each call placed using its network. Each CDR includes the caller and callee numbers as well as the time and duration of a phone call.

   Snooping's management software also has access to a database where phone numbers are categorized by the type of business such as banking and insurance institutions, electric and gas utilities, cable and Internet services, travel services and others.

   Snooping's chief marketing officer is now planning to develop a marketing platform to allow customers to opt-in to receive offers for services that match their calling profile, possibly in return for monthly discounts in their phone bill. For instance, if a customer has recently called an insurance company, the marketing platform can send additional insurance offers to him or her. Similarly, if a customer is placing multiple international calls, Snooping's marketing platform can send an offer with a special service package including long-distance calling deals.

Principles

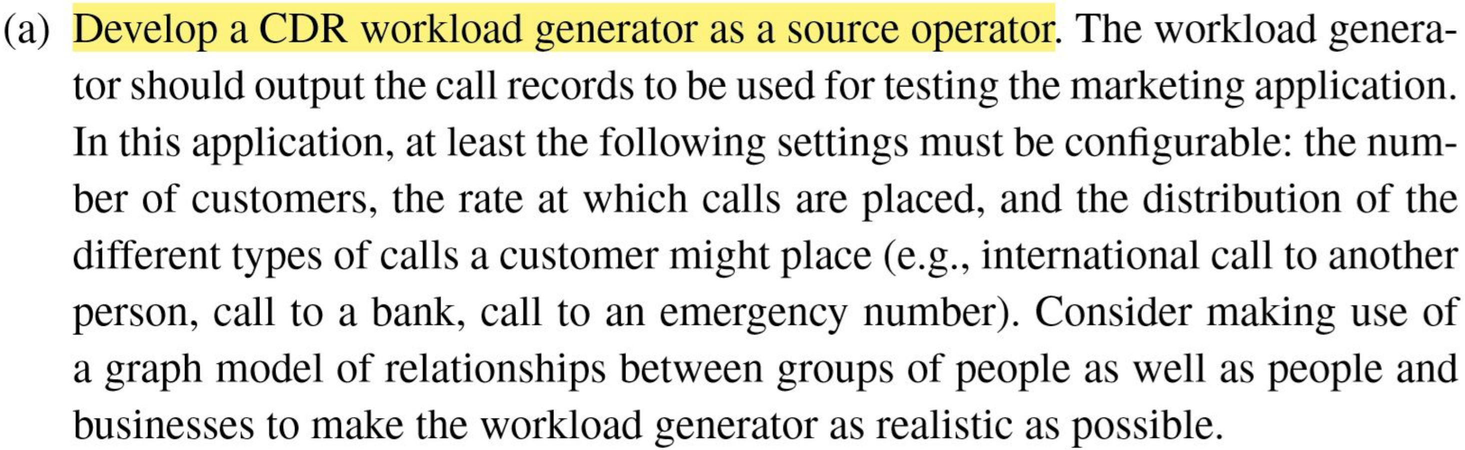


In this exercise, you will implement a prototype for this marketing platform, the *SnoopMarkPlat* application. When analyzing the development steps outlined below, abide by two general software engineering principles:

(1) *Modularization*: design the main application components, for data ingest, data processing, and data egress, as separate composite operators.

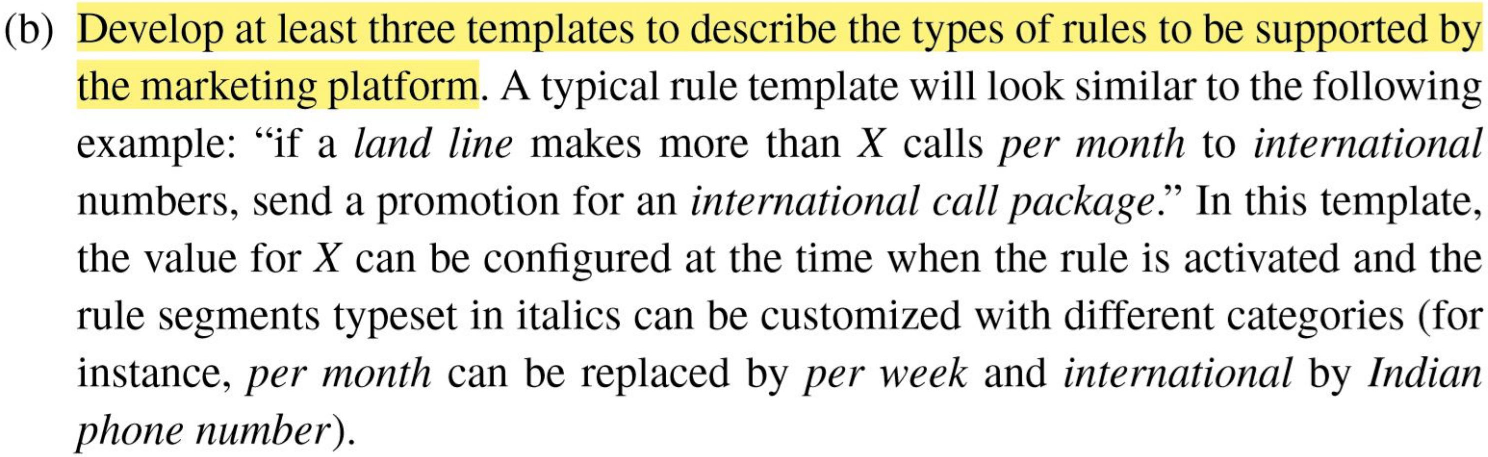
(2) *Flow composition*: consider the portions of your application where static, dynamic, and nested composition are each particularly effective and incorporate their use in your design.

Concrete Steps



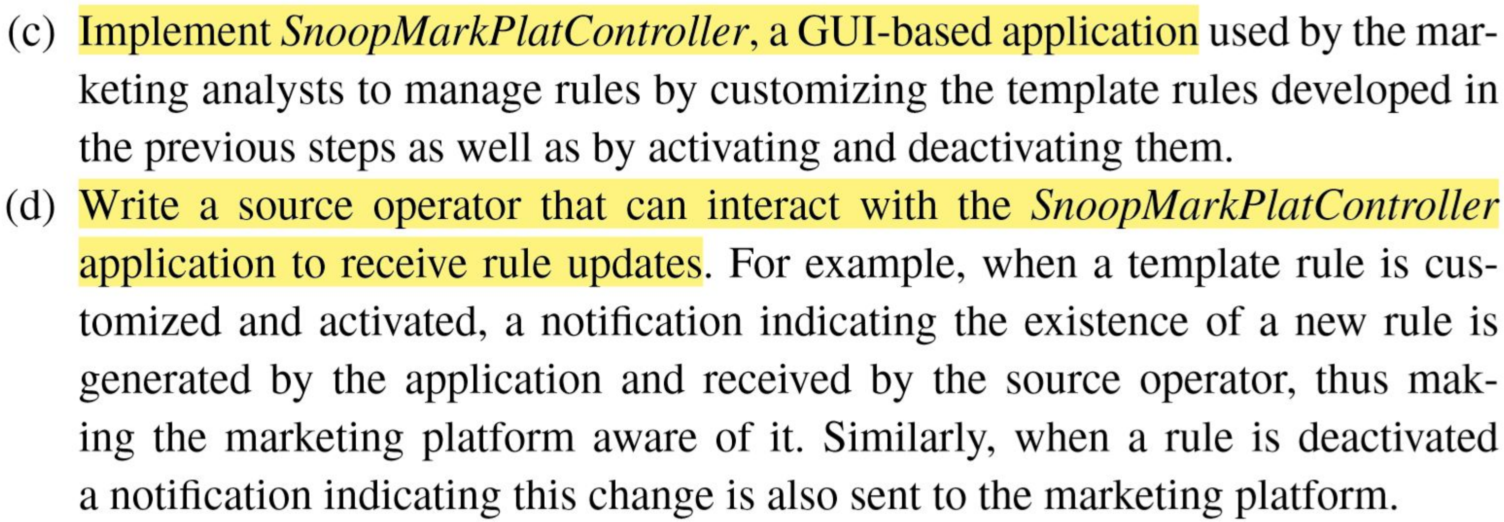
(a) Develop a CDR workload generator as a source operator. The workload generator should output the call records to be used for testing the marketing application. In this application, at least the following settings must be configurable: the number of customers, the rate at which calls are placed, and the distribution of the different types of calls a customer might place (e.g., international call to another person, call to a bank, call to an emergency number). Consider making use of a graph model of relationships between groups of people as well as people and businesses to make the workload generator as realistic as possible.

Concrete Steps



(b) Develop at least three templates to describe the types of rules to be supported by the marketing platform. A typical rule template will look similar to the following example: “if a *land line* makes more than *X* calls *per month* to *international* numbers, send a promotion for an *international call package*.” In this template, the value for *X* can be configured at the time when the rule is activated and the rule segments typeset in italics can be customized with different categories (for instance, *per month* can be replaced by *per week* and *international* by *Indian phone number*).

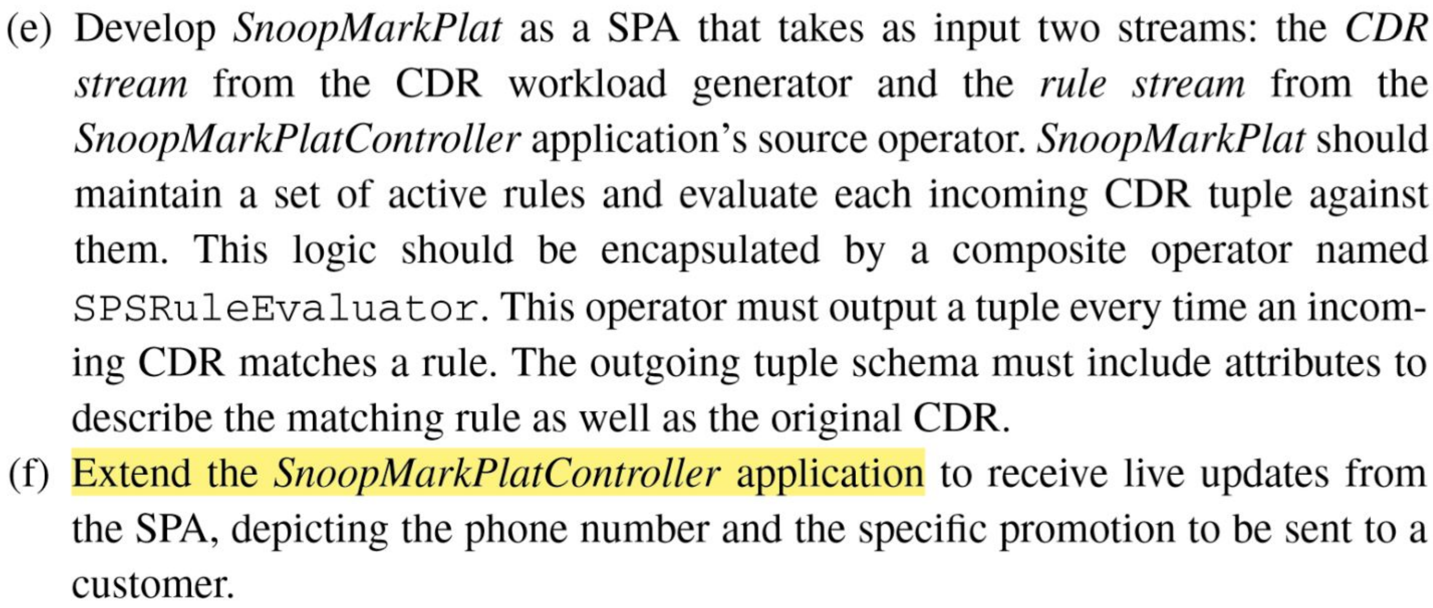
Concrete Steps



(c) Implement *SnoopMarkPlatController*, a GUI-based application used by the marketing analysts to manage rules by customizing the template rules developed in the previous steps as well as by activating and deactivating them.

(d) Write a source operator that can interact with the *SnoopMarkPlatController* application to receive rule updates. For example, when a template rule is customized and activated, a notification indicating the existence of a new rule is generated by the application and received by the source operator, thus making the marketing platform aware of it. Similarly, when a rule is deactivated a notification indicating this change is also sent to the marketing platform.

Concrete Steps



(e) Develop *SnoopMarkPlat* as a SPA that takes as input two streams: the *CDR stream* from the CDR workload generator and the *rule stream* from the *SnoopMarkPlatController* application's source operator. *SnoopMarkPlat* should maintain a set of active rules and evaluate each incoming CDR tuple against them. This logic should be encapsulated by a composite operator named `SPSRuleEvaluator`. This operator must output a tuple every time an incoming CDR matches a rule. The outgoing tuple schema must include attributes to describe the matching rule as well as the original CDR.

(f) Extend the *SnoopMarkPlatController* application to receive live updates from the SPA, depicting the phone number and the specific promotion to be sent to a customer.