eCommerce Model

Types

- Customer
    - ID
    - year of birth
    - gender
    - country
- Order
- Items
    - Type
    - Price

▼ Customer Generation

Attributes

- *ID* → “customer1”, “customer2”, …
- *Gender* → 50:50
- *YearOfBirth* → normal(μ: 30, σ: 10)

▼ *Country* → EU-Population distribution

EU Countries

Country	% of Population
Germany	18.81%
France	15.18%
Italy	13.12%
Spain	10.72%
Poland	8.20%
Romania	4.25%
Netherlands	3.97%

Country	% of Population
Belgium	2.62%
Czech Republic	2.41%
Sweden	2.35%
Portugal	2.33%
Greece	2.32%
Hungary	2.14%
Austria	2.03%
Bulgaria	1.44%
Denmark	1.32%
Finland	1.24%
Slovakia	1.21%
Ireland	1.16%
Croatia	0.86%
Lithuania	0.64%
Slovenia	0.47%
Latvia	0.42%
Estonia	0.30%
Cyprus	0.21%
Luxembourg	0.15%
Malta	0.13%

Registration

Assumption: at the beginning, there are no customers, then the customer base grows exponentially

- *GLOBAL_REGISTRATION_DELAY_LOWER_DAYS, GLOBAL_REGISTRATION_DELAY_UPPER_DAYS*
    - upper and lower limit on how long to wait until a new customer registers itself
- *GROWTH_COEFF*

    - chance of which a newly registered customer will “recruit” a second one (to spark exponential growth)

Login

- *LOCAL_LOGIN_DELAY_LOWER_DAYS, LOCAL_LOGIN_DELAY_UPPER_DAYS*
    - upper and lower limit on how long to wait until “the same” customer logs in again

Order

If the customer is logged in, there is a chance of *CHANCE_OF_ORDER* that he will create an order

(Currently, a customer places at most one order per “session”, i.e., login-logout)

▼ Add Items

▼ Item list

We have 20 different items with different price ranges. Per Order, we add between 1-10 items.

Type	Price Range
Sunscreen	$5 - $20
Umbrella	$10 - $40
Laptop	$300 - $3,000
Smartphone	$200 - $1,500
T-shirt	$10 - $30
Swimwear	$10 - $50
Jeans	$30 - $80
Coffee maker	$20 - $200
Headphones	$20 - $300
Sneakers	$30 - $150
Makeup	$10 - $50
Sofa	$300 - $2,000
Tennis racket	$20 - $200
Pet food	$10 - $50
Jacket	$70 - $200

Type	Price Range
Car battery	$50 - $150
Multivitamin	$5 - $20
Necklace	$20 - $100
Printer	$50 - $300
Tent	$50 - $300

The probability of picking an item is **evenly distributed**, but there are the following **association rules**:

- Younger people (<25) → electronics
    - with a chance of 50%, draw out of: Laptop, Smartphone, Headphones, Printer
- Older people (>35) → old people stuff
    - with a chance of 50%, draw out of: multivitamin, pet food, coffee maker
- Sunny (Spain, Portugal, Greece, Cyprus, Malta)
    - with a chance of 50% draw out of: Sunscreen, Swimwear
- Rainy (Ireland, Belgium, Luxembourg, Netherlands, Denmark)
    - with a chance of 50%, draw out of: Umbrella, Jacket

- Also account for combinations: Young-Sunny, Young-Rainy, Old-Sunny, Old-Rainy

Association rules within an order:
When swimwear is picked, there is a chance of *ASSOCIATION_STRENGTH*, that a sunscreen is picked as well

Between adding Items to an order, there is a **random delay** between 0-10 min

▼ Orders and Delivery

User places the order (same length of delay as when adding an item, after adding the last one), he is logged out

Payment

2 Cases:

- With a probability of *CHANCE_PAY_INSTANT,* the user pays directly after placing the order, then the payment is confirmed, then the invoice is sent.
- Alternatively, the user “pays via invoice”: after the invoice is sent, the user takes 1-14 days to pay, only after confirming the payment.
    - During the first 7 days, there is a chance of *CHANCE_CANCELATION* that the user cancels the order. Then, there are no further actions.

Only after confirming the payment, preparing the delivery is continued.

Delivery

With a chance of *CHANCE_CANCELATION*, the user selects a “fast delivery”, otherwise he chooses “normal delivery”.

Distribution of Order to packages:

- An order is decomposed into a random amount of packages, the items are assigned to the packages in a random order
    - suppose there are *itemsLeftToPackage* items that still need to be packaged
    - pick a *packageSize* randomly from [1, itemsLeftToPackage]
    - form a package of *packageSize* items that are randomly chosen from the items that are not packaged yet and continue with its delivery
    - continue until there are no items left to package
- Then the packages are shipped individually (fast-if chosen “fast delivery”, slower - otherwise)
    - give to carrier (fast/normal)
    - carrier confirms (fast/normal)
    - delivery (fast/normal)
    - delivered

▼ Time Block

Configuration:

- *STARTOFWORK:* Time of day that work starts
- *ENDOFWORK:* Time of day that work ends

- distribution of time on “target day” is variable and can be passed as a parameter
    - exemplary functions:
        - *drawTimeWeekdayEx()* - normally distributed with μ=20:15h, σ=2h
        - *drawTimeWeekdayEx()* - normally distributed with two peaks: μ_1 =12h, σ_1=3h, μ_2=18h, σ_2=3h

Functions

- *delayByDays(days, drawTimeWeekday, drawTimeWeekend)*
    - time is delayed by *days* days, then a target time is randomly chosen according to *drawTimeWeekday* or *drawTimeWeekend* depending on whether the target day is on a weekend or not
- *delayByWeekdays(days, drawTime)*
    - time is delayed by *days* working days, then a target time is randomly chosen according to *drawTime*
- *delayByWorkingHours(hours)*
    - time is delayed by *hours* working hours (i.e., hours that are not on a weekend and within [*STARTOFWORK*,*ENDOFWORK*])

▼ Logging

- Log “details”, e.g., customer gender, YoB, country, only once
- Otherwise, only log “IDs”, e.g. “delivered: package 2, customer1”

▼ ToDos & Problems

▼ ToDos

☐ Share Documentation

☑ ~~Time Block (”pretty” target times)~~

  ☑ ~~delayByDay~~

- ☑ ~~delayByWorkingDay~~
- ☑ ~~delayByWorkingHour~~

☑ ~~Exponential growth - I customer registration~~

☑ ~~Redo Relationships: (Customer, Order, [Items])~~

☑ ~~Association rules (without gender)~~

- ☑ ~~Sunscreen → Sun hours in country~~
- ☑ ~~Umbrella → Rainfall in country~~
- ☑ ~~younger people → electronics (Laptop, Smartphone, Headphones)~~
- ☑ ~~older people → multivitamin, necklace, coffee maker~~

☑ ~~Relation within Items in Order~~

- ☑ ~~If ordered swimming trunks also order sunscreen~~

☑ ~~Package type~~

☑ ~~invoice: 2 branches (pay instantly, vs. pay after receiving the invoice)~~

☑ ~~split order in one or more packages~~

☑ ~~more events special to packages: package items, give to carrier, carrier confirms (order, package involved), delivery (customer involved)~~

☑ ~~Long-term dependency: selected standard delivery / selected premium delivery → “fast delivery events” if premium delivery chosen~~

☑ ~~Simpler and Consistent logging~~

- □ Noise and Compliance Problems
  - □ noise ON-OFF parameter
    - □ skip transitions with low prob
    - □ order without login
    - □ …
  - □ Compliance
    - □ fast delivery although no fast delivery selected
    - □ duplicate shipment of packages
    - □ forget to ship packages

▼ **Problems:**

- Pick separate items for association rules “sunny” and “swimwear → sunscreen”, otherwise the relation may be hard to detect
- PackageToString - how to use stuff from “file” in “left column”?
- “Remaining” tokens in long-term dependencies, is that bad?