

Development project specifications

TELCO SERVICE APPLICATIONS

A telco company offers pre-paid online services to web users. Two client applications using the same database need to be developed.

CONSUMER APPLICATION

The consumer application has a public Landing page with a form for login and a form for registration. Registration requires a **username** (which can be assumed as the unique identification parameter), a **password** and an **email**. Login leads to the Home page of the consumer application. Registration leads back to the landing page where the user can log in.

The user can log in before browsing the application or browse it without logging in. If the user has logged in, his/her username appears in the top right corner of all the application pages.

The Home page of the consumer application displays the service packages offered by the telco company.

A **service package** has an ID and a name (e.g., “Basic”, “Family”, “Business”, “All Inclusive”, etc). It comprises one or more services. Services are of four types: fixed phone, mobile phone, fixed internet, and mobile internet. The **mobile phone service** specifies the number of minutes and SMSs included in the package plus the fee for extra minutes and the fee for extra SMSs. The **fixed phone service** has no specific configuration parameters. The **mobile and fixed internet services** specify the number of Gigabytes included in the package and the fee for extra Gigabytes. A **service package** must be associated with one validity period. A **validity period** specifies the number of months (12, 24, or 36). Each validity period has a **different monthly fee** (e.g., 20€/month for 12 months, 18€/month for 24 months, and 15€ /month for 36 months). A package may be associated with one or more **optional products** (e.g., an SMS news feed, an internet TV channel, etc.). The validity period of an optional product is the same as the validity period that the user has chosen for the service package. An optional product **has a name and a monthly fee independent of the validity period duration**. The same optional product can be offered in different service packages.

From the Home page, the user can access a Buy Service page for purchasing a service package and thus creating a **service subscription**. The Buy Service page contains a form for purchasing a service package. The form allows the user to select one package from the list of available ones and choose the validity period duration and the optional products to buy together with the chosen service. The form also allows the user to select the **start date** of his/her subscription. After choosing the service packages, the validity period and (0 or more) optional products, the user can press a CONFIRM button. The application displays a CONFIRMATION page that summarizes the details of the chosen service package, the validity period, the optional products and the total price to be pre-paid: (monthly fee of service package * number of months) + (sum of monthly fees of options * number of months).

If the user has already logged in, the CONFIRMATION page displays a BUY button. If the user has not logged in, the CONFIRMATION page displays a link to the login page and a link to the REGISTRATION page. After either logging in or registering and immediately logging in, the CONFIRMATION page is redisplayed with all the confirmed details and the BUY button.

When the user presses the BUY button, **an order is created**. The order has an ID and a **date and hour of creation**. It is associated with the user and with the service package, its validity period and the

chosen optional products. It **also contains the total value** (as in the CONFIRMATION page) and the start date of the subscription. After creating the order, the application bills the customer by calling an external service. If the external service accepts the billing, the order is marked as valid and a service activation schedule is created for the user. A **service activation schedule** is a **record of the services and optional products** to activate for the user **with their date of activation and date of deactivation**.

If the external service rejects the billing, the order is put in the rejected status and the user is flagged as insolvent. When an insolvent user logs in, the home page also contains the list of rejected orders. The user can select one of such orders, access the CONFIRMATION page, press the BUY button and attempt the payment again. When the same user causes three failed payments, an alert is created in a dedicated auditing table, with the user Id, username, email, and the amount, date and time of the last rejection.

EMPLOYEE APPLICATION

The employee application allows the authorized employees of the telco company to log in. In the Home page, a form allows the creation of service packages, with all the needed data and the possible optional products associated with them. The same page lets the employee create optional products as well.

A Sales Report page allows the employee to inspect the essential data about the sales and about the users over the entire lifespan of the application:

- Number of total purchases per package.
- Number of total purchases per package and validity period.
- Total value of sales per package with and without the optional products.
- Average number of optional products sold together with each service package.
- List of insolvent users, suspended orders and alerts.
- Best seller optional product, i.e. the optional product with the greatest value of sales across all the sold service packages.

DESIGN DOCUMENTATION AND IMPLEMENTATION NOTES

- The call to the external service must be simulated with a function that returns true or false pseudo-randomly. For testing purposes, the demonstration should be able to show at least one case in which the service call fails and one case in which the service call succeeds.
- The aggregate data of the sales report must be computed by triggers that populate materialized view tables. The documentation must describe the SQL code of the view that would compute the aggregate data, the logical schema of the materialized view table(s) that store the aggregate data and the triggers that populate the content of the materialized view table(s).
- The project documentation should be realized in PDF or PPT format with the following parts:
 - Description of any extra hypothesis on the project specifications
 - Entity Relationship diagram
 - Relation model of the database in SQL or graphical format
 - Description of the views, materialized view tables and code of the materialization triggers
 - Description of the ORM
 - List of the application components
 - Examples of UML sequence diagrams of particularly significant interactions
- A PPT template of the documentation is provided in the project folder in Webeep
- Examples of the documentation are provided in the JPA exercise folder

ANSWERS TO QUESTIONS ASKED BY STUDENTS

1. To me it is not clear which is the unique parameter required for registration and the subsequent login. Can I assume that each email corresponds to one user to allow different users with the same username?
 - a. Username can be used as the unique identification parameter
2. According to "a public Landing page with a form for login and a form for registration," the registration form should be in the landing page. But here this says "back to the landing page." Is the Registration supposed to be done in the landing page or another page (the REGISTRATION page)?
 - a. The registration can be done in the landing page. After registering one is taken back to the same page, which also contains the login form. Then the user can login.
3. If the user can avoid logging in so can also an unlogged user see all details about packages?
 - a. Yes, an unlogged user can browse the content about the packages
4. If the response is affirmative, can I avoid putting login and registration forms in a different page and have packages home page and, for instance in a different column, login/registration forms all in one page?
 - a. Improvements in usability that do not alter the workflow are acceptable. But the evaluation of the project considers functional aspects only, because usability patterns are not addressed in the course and depend on many factors
5. Can I assume at most 4 services are included (at most one of each type)?
 - a. No, there could be multiple services of the same type, with different characteristics (e.g., more or less minutes of gigabytes, different extra fees, etc)
6. Can users buy a mobile phone service adding also Gigabytes? Can users buy a mobile or fixed internet service adding minutes and/or SMS?
 - a. The only configurations that can be purchased are those bundled in the offered packages.
7. How do we have to deal with extra fee? Should it be a counter of the minutes/giga etc already used and at the end of the available minutes we create another order or is it not necessary?
 - a. It is not necessary to monitor the actual consumption of minutes and gigabytes after the purchase of a package
8. There is no description for fixed phone service. What are the attributes to add to this service? Do we need to add only the number of minutes?
 - a. The fixed phone service has no specific configuration attributes.
9. Before, the text said that "service package must be associated with one validity period", now we can choose it. Is the validity period bound to packages or subscription?
 - a. A service package must be associated with one validity period, chosen from multiple ones available for the package. For example, service A can be sold at 20€/month for 12 months, 18€/month for 24 months, and 15€ /month for 36 months. The user selects the package A and also the duration of the subscription. A longer pre-paid duration implies a greater up-front disbursement but a long-term saving.
10. I understand that it is independent because the duration depends on the package. Since after it's said that the total price is pre-paid it's impossible to modify a package and its optional services during the validity period, right?
 - a. Yes, the package conditions remain fixed for the validity period and the purchase is pre-paid
11. Is it right to assume these 2 examples? - starting date 31/10 duration of 1 month results in ending date 30/11 - starting date 15/10 duration of 4 months results in ending date 15/2

- a. Yes, the various "Date" data types (Java, SQL) have methods for adding months to a date value.
- 12. I didn't understand if the validity period is associated with the service package or the specific subscription of the user.
 - a. A given service package is offered with different validity periods. The user can choose one of the offered validity periods among those associated with the package.
- 13. Also, for all the optional products offered by the company, for each service package only a subset of them is available, right? So the user can select optional products for a service package only from this subset, or am I missing something?
 - a. Yes, the interpretation is correct
- 14. Should this service activation schedule appear in any page? (the home page when the user is logged in?)
 - a. It is not requested, but can be useful for debugging and for the demo
- 15. *[If the external service accepts the billing, the order is marked as valid and a service activation schedule is created for the user. A service activation schedule is a record of the services and optional products to activate for the user with their date of activation and date of deactivation.]* Should this record contain all the services purchased by the user or only those that have not been activated yet?
 - a. It is sufficient to include only the service packages that are associated with the specific order.
- 16. If the insolvent user pays, is the insolvent flag removed?
 - a. You can remove the insolvent flag when the user no longer has any pending unpaid orders
- 17. Must the three failed payments be consecutive and about the same order or are they just three failed payments made by the same user?
 - a. just three failed payments made by the same user are enough to raise the alert
- 18. *[Total value of sales per package with and without the optional products]* It refers to: -the sum of total revenue from each package with and without optional services
 (total_revenue_package=x | total_revenue_package-optional_revenue_package=y)
 -the number of total sales of each package with and without optional services
 total_number_of_package_sold=x | total_number_of_package_sold -
 total_number_of_package_sold_without_optional= y)
 - a. It refers to two numbers: 1) the total revenues of the sale of package X not considering the value of the options; 2) the total revenues of the sale of package X including the value of the options sold together with the package X;
- 19. *[For testing purposes, the demonstration should be able to show at least one case in which the service call fails and one case in which the service call succeeds.]* Instead of a random return value, can we put two BUY buttons, one that will fail, and the other one that won't? This way we could test and clearly show all the scenarios in a predictable way.
 - a. Yes, this is a good idea