Povzetek

Zavarovalnice sklepajo zavarovanja za različna zavarovalna področja kot so na primer življenjska zavarovanja, premoženjska zavarovanja, zdravstvena zavarovanja itd. Vsa ta zavarovanja imajo različne zavarovalne pogoje, trajanja, izstopne pogoje, dinamike plačevanja (mesečno, polletno, letno...) in posledično tudi različna pravila opominjanja. Želja naročnika je čim bolj poenotiti in avtomatizirati proces opominjanja strank za vsa zavarovalna področja.

Naročnik je predstavil pogoje opominjanja, izrazil je željo, da bi bili pogoji zapisani v šifrantih, da jih lahko sam ureja. Poleg pogojev, ki bodo zapisani v šifrantih, so tu še tako imenovani sistemski pogoji, ki opredeljujejo, kaj sploh je odprta zapadla terjatev. Naročnik je želel tudi vpogled v kandidate za opomine, pri tem je poudaril, da želi vedeti tudi zakaj nekatere postavke, ki ustrezajo sistemskim pogojem, niso kandidati za opomin – torej pri katerem pogoju iz šifrantov račun izpade iz kandidatov. Naslednja zahteva je bila možnost izločitve postavke iz seznama kandidatov za opomin. Ko je stranka s seznamom kandidatov zadovoljna, lahko iz njega izdela opomine in obvestila o nastanku opomina.

Pogoje smo združili v smiselne skupine, vsako skupino smo predstavili z bazno tabelo. Naročniku smo omogočili vpogled v bazo s pomočjo orodja Oracle Forms.

Kandidate, opomine in obvestila smo prav tako predstavili z baznimi tabelami ter izdelali obrazce, preko katerih naročniku omogočimo vpogled v obdelave kandidatov, opominov in obvestil.

Abstract

Insurance companies offer different types of insurance products, like for example life insurance, property insurance, health care insurance and many more. Each of these have different insurance policy, duration, opt–out policy, payment interval (monthly, every six months, once per year...) and as such different reminder rules. Our client would like to have a unified and automated process of reminding it's clients about outstanding claims in all insurance fields.

The requirements for functionalities regarding the reminding process were specified by the client. They need to have the ability to set and change the rules by themselves. In addition to the custom rules, there are some system rules, defining what is an open outstanding claim. The client wants an overview of potential candidates for reminders. They also requested a feature which will help them determine why some invoices that meet the system rules are not candidates for the reminders and which conditions are the cause of such exclusions. An option to disable reminder generation for specific candidates must be provided. Once the outstanding claim list meets their expectations they should be able to generate the reminders.

The reminder rules were combined into groups. Each group as well as candidates, reminders and notifications were stored in database tables. The user interface was implemented using Oracle Forms, by which the client is able to access candidate lists as well as control reminder and notification processing.

Math. Subj. Class. (2010): 68N15, 68N19, 68P05, 68P10, 68P20
Ključne besede: zapadla neplačana postavka, pogoji opominjanja, podatkovna baza, Oracle, OAS, vnosni obrazci
Keywords: overdue receives, conditions of reminders, database, Oracle, OAS, forms

Viri

[1] *What is Linux?* [Elektronski vir].

Dosegljivo: https://www.linux.com/what–is–linux. [Dostopano: 16.6.2016].

[2] *What is Apache?* [Elektronski vir].

Dosegljivo: http://wiki.apache.org/httpd/FAQ#What_is_Apache.3F.
[Dostopano: 16.6.2016].

[3] *Hypertext Transfer Protocol* [Elektronski vir].

Dosegljivo: https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol.
[Dostopano: 16.6.2016].

[4] *Oracle Designer* [Elektronski vir].

Dosegljivo: https://en.wikipedia.org/wiki/Oracle_Designer. [Dostopano: 16.6.2016].

[5] *Computer–aided software engineering* [Elektronski vir].

Dosegljivo: https://en.wikipedia.org/wiki/Computer–aided_software_engineering.
[Dostopano: 16.6.2016].

[6] *Entity–relationship model* [Elektronski vir].

Dosegljivo: https://en.wikipedia.org/wiki/Entity%E2%80%93relationship_model.
[Dostopano: 16.6.2016].

[7] *Toad (software)* [Elektronski vir].

Dosegljivo: https://en.wikipedia.org/wiki/Toad_%28software%29.
[Dostopano: 16.6.2016].

[8] *SQL* [Elektronski vir].

Dosegljivo: https://en.wikipedia.org/wiki/SQL. [Dostopano: 16.6.2016].

[9] *Oracle Forms* [Elektronski vir].

Dosegljivo: https://en.wikipedia.org/wiki/Oracle_Forms. [Dostopano: 16.6.2016].

[10] *Oracle Repots* [Elektronski vir].

Dosegljivo: https://en.wikipedia.org/wiki/Oracle_Reports. [Dostopano: 16.6.2016].

[11] *Java (programming language)* [Elektronski vir].

Dosegljivo: https://en.wikipedia.org/wiki/Java_(programming_language).
[Dostopano: 16.6.2016].

[12] *HTML* [Elektronski vir].

Dosegljivo: https://sl.wikipedia.org/wiki/HTML. [Dostopano: 16.6.2016].

[13] *Oracle Application Server* [Elektronski vir].

Dosegljivo: https://en.wikipedia.org/wiki/Oracle_Application_Server.
[Dostopano: 16.6.2016].

[14] *Foreign key* [Elektronski vir].

Dosegljivo: https://en.wikipedia.org/wiki/Foreign_key. [Dostopano: 16.6.2016].

[15] *Unique key* [Elektronski vir].

Dosegljivo: https://en.wikipedia.org/wiki/Unique_key. [Dostopano: 16.6.2016].

[16] *Synonym (database)* [Elektronski vir].

Dosegljivo: https://en.wikipedia.org/wiki/Synonym_%28database%29.
[Dostopano: 16.6.2016].

[17] *Grant* [Elektronski vir].

Dosegljivo:
http://docs.oracle.com/cd/B19306_01/server.102/b14200/statements_9013.htm.
[Dostopano: 16.6.2016].

[18] *Relational database* [Elektronski vir].

Dosegljivo: https://en.wikipedia.org/wiki/Relational_database. [Dostopano: 16.6.2016].

[19] *Database Schema* [Elektronski vir].

Dosegljivo: http://www.tutorialspoint.com/dbms/dbms_data_schemas.htm
[Dostopano: 16.6.2016]