Improving Management of Utilities Payment with Web – based Solution

Serban Popa*, Gabriela Ioana Prostean*

*“Politehnica” University, Faculty of Management in Production and Transport, Timisoara, România

popaserban@yahoo.com, gabriela.prostean@mpt.upt.ro

Abstract—The payment of the utilities: gas, electricity, water, mobile and fixed phones, internet, TV, insurance, local taxes, etc. benefits by multiple solutions. These solutions split in two categories: IT based (online) payment methods and outside payments methods, at different pay points. This research presents a solution, web-based and mobile based, for paying and managing the utilities that involves an online method which has a novel concept. This method is transparent to the users and involves almost zero time spent for payments and smaller costs generated by commissions than in other existing payment methods.

I. INTRODUCTION

There is a wide range of individuals, either person either company that has delays in time when paying utilities. This happens due to the administrative burden of these payments where usually a lot of time is needed to enter in possession of the invoices, analyze and pay in time each invoice.

A study made by a local bank, [1] presented in Table 1, shows the time effort of an individual for paying utilities at the utility provider office. A total time of 170 minutes for all utilities payments compared to the average time of 34 minutes for one payment, which is the case for our application is an certain example of the major benefit of our proposed solution. Alternative channels which offer the possibility to pay the invoices take less time but not lesser than our application. The alternative channels can be: internet banking, mobile banking, phone banking, direct debit. This time is shown in table 2, [1].

<table>
<thead>
<tr>
<th>Invoice type</th>
<th>Time consumed for utility payment (minutes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Electricity</td>
<td>30</td>
</tr>
<tr>
<td>Mobile/fixed telephony</td>
<td>20</td>
</tr>
<tr>
<td>TV cable and internet</td>
<td>30</td>
</tr>
<tr>
<td>Gas</td>
<td>45</td>
</tr>
<tr>
<td>Assurance policy</td>
<td>45</td>
</tr>
<tr>
<td>Total time (minutes)</td>
<td>170</td>
</tr>
<tr>
<td>Average time (minutes)</td>
<td>34</td>
</tr>
</tbody>
</table>

TABLE 2. TIME SPENT FOR UTILITY PAYMENT THROUGH ALTERNATIVE CHANNELS

<table>
<thead>
<tr>
<th>Invoice type</th>
<th>Time consumed for utility payment (minutes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Electricity</td>
<td>5</td>
</tr>
<tr>
<td>Mobile/fixed telephony</td>
<td>0</td>
</tr>
<tr>
<td>TV cable and internet</td>
<td>3</td>
</tr>
<tr>
<td>Gas</td>
<td>3</td>
</tr>
<tr>
<td>Assurance policy</td>
<td>4</td>
</tr>
<tr>
<td>Total time (minutes)</td>
<td>15</td>
</tr>
<tr>
<td>Average time (minutes)</td>
<td>3</td>
</tr>
</tbody>
</table>

II. RELATED WORK

In [2] a solution for domestic utility payment by using the mobile phone is described. The advantage of this solution is the reduction of the bank commissions for each payment.

A transaction barcode is read in [3] to collect the supplier identification information and verify the client information of the corresponding supplier upon receipt of a transaction request made by the customer. It is used for paying educational fee, insurance bill, utility bill, rental fee. The disadvantage is that it requires human resources during the process and it is not fully automated.

Electronic calendars, and especially internet-based calendars, are becoming more and more popular. The software application presented in [4] allows for the easy generation, management, and storage of lists of personal events using the internet-based application calendar.

Unlike the mentioned solutions, our solution involves zero time spent for paying the utilities and smallest costs possible for commissions. Also, our solution will be accessed via a mobile device, Android or IOS, making more efficient in managing and following the payments.

III. MANAGEMENT MODEL OF OUR SOLUTION

The management model used in developing our application is Agile Project Management. The Agile model is focused on business requirements split in more iterations which finally give a form of a finished product.
Agile splits the development process in smaller parts named stories. The main characteristics of the Agile model are:

- Allows a major change in the design during implementation without having a negative impact on the developing costs;
- Has a fixed iteration in time, usually lasts two weeks. This time period cannot be modified so a fixed result can be known at the final of each iteration;
- It is based on functional intermediary deliveries of the application. Each iteration will implement all the stories included in that iteration.

The Agile management methodology used is SCRUM. SCRUM values are:

- The individuals and the interaction have priority in front of the processes and the development instruments. Although these are useful, they will not bring any plus in the development progress if the team does not learn to communicate and cooperate in a constructive manner.
- The functional software delivery has priority in front of bushy documentation. The progress documentation is important but it is more important that the final result to meet the client expectations.
- The cooperation with the client has priority in front of the contract negotiation. The main idea is that a contract must not be signed for money but to solve the issue raised by the client.
- Answer the change following the plan. If the requests are changing, both the project plan and the design must be updated.

IV. OPERATIONAL MODEL OF OUR SOLUTION

The feature offered by the application to the end user is the handling of all monthly payments in one payment. The system generates one notification payment containing all monthly user payments that have the most appropriate maturity date. Once the notification payment is paid by the user the money are redirected to each supplier. This guarantees a very competitive additional cost of the payment process, additional cost given by the bank commissions. Also, it guarantees the lowest possible time spent on paying the utilities and other monthly or yearly payments. Payments that can be accomplished via this feature are: electricity, gas, TV, internet, mobile and home telephony, water, life assurance, local taxes, car taxes and assurance, etc. A user can be an individual or a company.

The system is currently web-based and it is developed mostly for desktop usage. Now, it is being developed for mobile telephones, Android and IOS based. The important feature that comes as a near future development is based on curl (Command line uniform resource locator). Curl is a free and open software that compiles and runs under a wide variety of operating systems. Curl is a command line tool for transferring data with URL syntax. Curl supports SSL certificates, HTTP POST, HTTP PUT, FTP uploading, HTTP form based upload, proxies, cookies, user and password authentication [7]. Curl allows the user to access each supplier account from our one application. By only a click, the user can see the status and cost control for each supplier. If he is in front of an ATM machine, he can
instantly check all the payment that has to be made and pay them if needed. In the first described feature there is a lack of knowledge from the application point of view. If a user made a payment, the application cannot know about this. But, buy using curl and checking immediately the cost control, the application knows if the payment was done or not and can add it or not to the notification payment that contains all the payments.

A next future development, that has been already started, is the personal management via a list of events that must be handled by each person. We all have a license card, an ID, a passport, different kind of assurance or documents that expire at one moment. The application will offer a full calendar which can be populated with personal information about the maturity dates of each document. It will be based on a notification system, by mail, by SMS or by having a personal consultant who calls at each needed time.

The financial foresights are generated by our application. The user can predict his monthly cash flow and the needed budget for the entire year.

V. FINANCIAL MODEL OF OUR SOLUTION

The financial schema is presented in figure 3. The steps in the right priority for the financial mechanism are as follows:

- The services suppliers release the monthly invoices for the user utilities. These invoices are transmitted to the application.
- The application receives the invoices and stores them in the invoice database. The application invoice analyzer parses and maps the invoices to each application user.
- The application builds a notification payment for each user containing all the payments that the user has to make in a time period. The notification payment is sent to the user.
- The user pays his notification payment or by loading his balance, the notification payment is automatically paid.
- The application collects the payments and redirects the money to each supplier.
- The sums are collected by the suppliers.

The major advantages in this process for the user are:

- He has only one bank commission if he decides to make the payment via a payment order;
- His time effort is as minimum as possible because he has to pay at the application cashier only one invoice instead of all supplier invoices.

The suppliers have also a big advantage in this process because there is another entity that notifies the users about the maturity dates of their invoices so the collection process has no costs and it happens on time.

Our checks shown in table 3 present a list of commissions which an individual should pay when he uses payments orders issued from a bank to each supplier. Our application commission is 8 RON, which is lower than 13 RON, the total sum of the commissions from Table 3.

<table>
<thead>
<tr>
<th>Utility/Service</th>
<th>Commission (RON)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Electricity</td>
<td>1</td>
</tr>
<tr>
<td>Gas</td>
<td>1</td>
</tr>
<tr>
<td>Sanitation</td>
<td>2.5</td>
</tr>
<tr>
<td>Mobile telephony</td>
<td>2.5</td>
</tr>
<tr>
<td>Fixed telephony</td>
<td>0</td>
</tr>
<tr>
<td>TV and Internet</td>
<td>1</td>
</tr>
<tr>
<td>Assurance for car or house</td>
<td>2.5</td>
</tr>
<tr>
<td>Private assurance</td>
<td>2.5</td>
</tr>
<tr>
<td>TOTAL</td>
<td>13</td>
</tr>
</tbody>
</table>

VI. TEST MODULE IMPLEMENTATION

Testing is very important and our implementation allows to easy create an object UserDAOTest by injecting the GenericDAO [8]. Dependency Injection (DI), which allows you to inject objects into a class, rather than relying on the class to create the object itself. The use of a factory class is one common way to implement DI. When a component creates a private instance of another class, it internalizes the initialization logic within the component. This initialization logic is rarely reusable outside of the creating component, and therefore must be duplicated for any other class that requires an instance of the created class. Dependency Injection aims to reduce the amount of boilerplate wiring and infrastructure code that you must write and it reduce coupling between components. [9]

```java
public class UserDAOTest {
    @Inject
    GenericDAO<User, Long> dao;
    @Before
    public void setUp() throws Exception {
        Injector injector = Guice.createInjector(new PersistenceModule());
    }
```
injector.injectMembers(this);
    dao.getEntityManager().getTransaction().begin();

    @After
    public void tearDown() throws Exception
    {
        dao.getEntityManager().getTransaction().rollback();
    }

    The tests allow an efficient management of the database for rollbacks. There is no need to implement transaction. begin() and transaction.commit() methods because the transactions management is done by the framework. Let’s consider a method M1() that has a transaction and calls a method M2() which also has a transaction. If the transaction form M2() fails then the transaction form M1() can be continued and the code does not have to throw an exception.

    @Test
    public void testFindsetUsernamet("user1");
    user.setPassword("password");
    user.setStatus(Boolean.TRUE);
    Long id = dao.insert(user);
    assertNotNull(dao.find(id));
}

VII. CONCLUSIONS

The paper described a novel and original solution for managing the utilities payments of an individual or a company. It is an informatics solution, online and mobile and it requires no time involved for the users and has the minimum cost possible comparing to all the other existing payment solutions.

REFERENCES

[3] J. Wee and J.S. Woe, Payment service for paying e.g. educational fee, insurance bill, utility bill, rental fee, involves requesting approval of transaction made by purchaser for merchant to finance company corresponding to financial institution card, Patent Number JP2000952764-W