Looking for Communicative Sinergy Between Generators and Consumers of Açaí Pulp Production Waste with Software Prototyping

Procurando uma sinergia comunicativa entre geradores e consumidores de Açaí Resíduos de produção de celulose com Protótipo de software

DOI:10.34117/bjdv7n6-291

Recebimento dos originais: 07/05/2021
Aceitação para publicação: 14/06/2021

Jefferson Junior Alves da Silva
Master's Degree Student in Software Engineering
Federal University of Pará
R. Augusto Corrêa, 01, Guamá 66075-110, Belém/PA/Brazil
jefferson.alves.silva@icen.ufpa.br

Ítalo Flexa Di Paolo
Master in Electrical Engineering
Pará State University
R. Pedro Porpino da Silva, no. 1181, São José 68745-000, Castanhal/PA/Brazil
itflexa@uepa.br

Wanderson Alexandre da Silva Quinto
Master in Electrical Engineering
Pará State University
Rua Mato Grosso, 137, Setor Alto Paraná 68550-000, Redenção/PA/Brazil
w.quinto@uepa.br

Adriano Cesar Calandrini Braga
Doctor in Food Science and Technology
Pará State University
R. Pedro Porpino da Silva, no. 1181, São José 68745-000, Castanhal/PA/Brazil
accbraga@uepa.br

Denilson Ricardo de Lucena Nunes
Doctor in Production Engineering
Pará State University
R. Pedro Porpino da Silva, no. 1181, São José 68745-000, Castanhal/PA/Brazil
denilson.nunes@uepa.br

André Cristiano Silva Melo
Doctor in Production Engineering
Pará State University
Tv. Dr. Enéas Pinheiro, no. 2626, Marco 66095-015, Belém/PA/Brazil
acsmelo@uepa.br
ABSTRACT
In Northern Brazil, many açaí producers have difficulties and are even unaware of methods that can be used to treat the waste from their production. There are many reports on how this waste is disposed of and several methods are not advisable due to poor compliance with Federal Law no. 12,305/2010, which established the National Solid Waste Policy. On the other hand, some industrial segments or links of large, medium, and small size are interested in this waste for use in different branches of activity; however, there is no computational tool that guarantees integration between such segments, which prevents waste from flowing. These processes are linked to reverse logistics and revaluation with waste recycling. In this regard, the main objective of our work is to promote an experiment platform to serve as “evolutionary prototyping” that meets the development of software in an agile, incremental, iterative approach and synergistically integrates the links of the production chain, increasing assertiveness in açaí solid waste management throughout Pará.

Keywords: Reverse Logistics, Prototyping, Agile Methods, Açaí Solid Waste.

RESUMO
No norte do Brasil, muitos produtores de açaí têm dificuldades e até mesmo desconhecem métodos que podem ser utilizados para tratar os resíduos de sua produção. Há muitos relatórios sobre como esses resíduos são descartados e vários métodos não são aconselháveis devido ao fraco cumprimento da Lei Federal nº 12.305/2010, que estabeleceu a Política Nacional de Resíduos Sólidos. Por outro lado, alguns segmentos ou elos industriais de grande, médio e pequeno porte estão interessados nestes resíduos para uso em diferentes ramos de atividade; entretanto, não há nenhuma ferramenta computacional que garanta a integração entre tais segmentos, o que impede o fluxo de resíduos. Estes processos estão ligados à logística reversa e à revalorização com a reciclagem dos resíduos. Neste sentido, o principal objetivo de nosso trabalho é promover uma plataforma experimental para servir como "protótipo evolutivo" que atenda ao desenvolvimento de software de forma ágil, incremental, iterativa e que integre sinergicamente os elos da cadeia produtiva, aumentando a assertividade na gestão de resíduos sólidos do açaí em todo o Pará.

Palavras-chave: Logística Reversa, Prototipagem, Métodos Ágeis, Resíduos Sólidos do Açaí.

1 INTRODUCTION
According to reports from the Brazilian Association of Cleaning and Special Waste – ABRELPE (2019), of the waste collected, in general, in the Northern region, 64.7% or 8,456 tons per day are inappropriately destined for controlled dumps and landfills. However, açaí stone waste is still dumped in large volumes on public spaces by informal producers, causing damage, as illustrated in Figure 1. This demonstrates the absence of selective collection by companies working for the government, since the Federal Solid Waste Law states that the parties responsible for collection and disposal are
the generators themselves, as a result of their commercial activity. Only a few producers pay for the transportation of this waste to appropriate locations, which otherwise can end up clandestinely in landfills.

Figure 1. Açaí stone disposal in public space.

![Açaí stone disposal in public space](image1.jpg)

Author: Ivan Duarte (DIAS, 2019).

The Municipal Secretariat of Sanitation of Belém – SESAN (DIAS, 2019) estimates that a daily waste volume of approximately 350 tons of açaí stone is produced in the city of Belém and its districts. Reduced capacity to store them—such as in Small Volume Receiving Units (URPV) administered by SESAN—is noticeable, as shown in Figure 2, in addition to açaí having great importance for the culture and regional economy (SILVA et al., 2020).

Figure 2. Locations of waste disposal at URPV.

![Locations of waste disposal at URPV](image2.jpg)

Source: Gutierrez et al. (2017).
This large amount of waste generated from açai pulp production can also serve as input for several industries, such as food, energy generation, pharmaceuticals, and others (MELO et al., 2016). And information technology can be useful in creating synergy between the agents involved in these processes, without relying on intermediaries such as the government itself (GUTIERREZ et al., 2017).

This work discusses the process of building a prototype for a computerized information system, which focuses on the reuse of solid waste from the açai stone by interested parties in the production and consumption chains, and which serves as a support tool for specification requirements that represents what the end users and those involved in the project can evaluate and propose during processes of iterative development cycles. In addition, this prototype will be used as an evaluation proposal for future assertive contributions. For this purpose, our specific objectives are:

- to execute a software prototyping process;
- to perform acceptance tests of the prototype in a complete testing process.

2 MATERIALS AND METHODS

This study is linked to the research project approved in Announcement 43/2016 from the Amazonia Foundation for the Support of Study and Research (FAPESPA) entitled “Structuring Reverse Channels for the Production Chain of Açai: A Proposal Guided by the National Solid Waste Policy”, which is still running with support from the Study and Research Group on Information Systems and Knowledge (GSIC) and the Management of Logistic Systems and Productive Systems for Regional Development at the State University of Pará (UEPA), with coordinated actions between the courses of Production Engineering, Food Technology, and Technology in Systems Analysis and Development at UEPA. It contributes to a model of reverse channels and reverse logistics focused on waste generated from the production of açai pulp or wine in the capital Belém, in the municipalities of the metropolitan region of Belém, and in Castanhal, and also takes into consideration the guidelines imposed by the National Solid Waste Policy (NSWP) through Federal Decree no. 7,404/2010, which regulates Federal Law no. 12,305/2010 (MELO et al., 2016). According to the Environmental Prosecution Department and the Public Prosecutor’s Office of the State of Pará, there is no specific legislation for the logistics of açai waste which promotes the creation and strengthening of interests around production chains focused on reuse. For this reason, our work intends to be a vector that
stimulates and favors the structuring of revaluation channels for the reduction of açaí stone waste.

Sabbagh (2013) explains that the manifesto for agile software development defines a structure which “moves” easily, quickly, and “lightly”, in relation to “traditional methods.” In this sense, we developed a practical view of the actions, as shown in Figure 3.

Figure 3. Project activity flow using Scrum.

During the first team meetings, issues dealing with the project’s Roadmaps (goals map), product vision, and preliminaries were discussed. Starting with unconnected ideas, the Product Backlog artifact was created with 28 items also known as “user stories”, each one with a degree of complexity and ordered by priority. Next came the Sprint Backlog for “grooming” (refinement) and arranging the items in a progress chart called Kanban, followed by the definition of preparation up to the execution of the work with Sprint Planning. Since the Product Backlog is an emerging initial process, in a character of relevance and gradual detailing, it was necessary to follow refinement techniques in more detail, as shown in Figure 4—which demonstrates other aggregated techniques such as estimates, story points, planning poker, a project manager called Product Owner, a development team (Dev Team), and a Scrum Master.
Based on meetings for the gathering of suggestions, a graphic interface sketch was developed using the Balsamiq tool. In this step, means of interaction were demonstrated, simulating an actual system environment as presented in Figures 5 to 7.

Figure 5. Wireframe for the project’s home screen.

Source: The authors.
The Product Backlog with 28 records contains columns: ID for the identifier; a summary DESCRIPTION of what the listed activity or story represents, or its functionalities; a degree of IMPORTANCE in relation to the customer’s perspective, measured on a scale from 10 to 150—the higher the score, the more important the story; P/S INITIAL ESTIMATE (People per Story), focusing on verifying the number of points, or the measure of effort that the team will deploy to execute an activity, along with its
complexity, and which will serve as base to calculate how many days/hours/people will be needed to deliver the story, a unit that is linked to the ideal “person/days ratio” (for example: “in how many days will we present a finished, demonstrable, and tested implementation?” If the answer is, “with 3 people in a room it will take approximately 4 days,” then the initial estimate is 12 points per story); NOTES, with comments on the functionality (optional); and finally ACCEPTED, with the mutual decision that an item will be forwarded to delivery. Sprint Planning, as shown in Figure 8—containing all the necessary planning for the execution of each activity—, was implemented using an electronic spreadsheet. In this element, we can highlight the definition of each story, the week it was placed on, who will execute it, its beginning and end, the number of days assigned, the status of progress or postponement, and acceptance criteria.

Figure 8. Sprint Planning artifact.

Source: The authors (2018).

To assist in the planning and management of delivery and progress control, two very important artifacts were used as well, one of which is the Burndown follow-up graph, also in an electronic spreadsheet format, as shown in Figure 9.
Figure 9. Burndown graphical artifact (Progress).

Source: The authors (2018).

The other artifact is called a Kanban Board and displays the items that have been selected for development. In Figure 10 we can see the phases (Items, To Do, Doing, Done) that each item can go through.

Figure 10. Kanban Board artifact (Monitoring).

Source: The authors (2018).

Figure 11 shows the proposed life cycle for the process model implemented in the prototype. There are clearly two distinct stages to be followed, namely: (1) use of the Scrum framework in an iterative and incremental cycle, also known as evolutionary, which intends to develop artifacts specifying the goals; and (2) creation of a prototype until confirmation of product completion. Within the evolutionary development stage there are connections from one process to the other. Due to its incremental nature, only after one process is completed can we move on
to the next. In the prototyping stage, we also see an incremental sequence; however, a few dependencies are associated with some processes, as indicated by dashed lines.

Figure 11. Diagram for the Prototype’s Development Cycle.

![Diagram for the Prototype's Development Cycle]

Source: The authors.

Developing the prototype required a preliminary artifact to organize the project structure and define important tools for its elaboration. For this purpose, a component diagram was established for the project, defining an application architecture, and organizing interacting components and flow mapping. For this prototype, the MVC architecture standard (Model, View, Controller) was chosen, which basically divides the project into layers and displays the flow between them. For example, a request from the view layer passes through an intelligent controller layer and triggers an action at the model layer responsible for accessing a data repository in a database. Regarding business layers, the Design Pattern, and the DAO (Data Access Object) standards were implemented, to ensure separation and distinction of classes or files for both layers, reducing dependencies as much as possible. The prototype was designed to be a hybrid web application (accessed by multiple operating systems, which guarantees a wider range
of use) adapted to be executed by a “world” of devices. With this in mind, the tools used to develop the prototype were HTML 5, PHP 7, JavaScript, and MariaDB 10.

In the prototype, we wrote code using the Google Maps API to geolocate addresses that were visited and flagged in a mobile device—which, connected to the Internet, flagged the points, and defined the types of flags, as shown in Figure 12. These locations are registered in the database so they can be referenced on the map, as shown in Figure 13, with the prototype interface.

Figure 12. Mapping of waste production locations by the field teams.

Source: The authors.

Figure 13. Points referenced in the map after registration on the prototype.

Source: The Authors.
The prototype contains 38 files and 4 folders, organized as follows:

1. css (files defining the style of web pages)
2. fonts (files for the fonts used in web pages)
3. img (files for all images of the application)
4. js (files that execute programming on the client side of the navigation)

Code files obey the structure [performed_operation or inclusion_name]_[module_name]_[layer_name].php.

For example: record_negotiation_contoller.php, where the layer name is in parenthesis because it can be omitted.

3 RESULTS

We developed the prototype to use it as an experiment tool with all agents involved in the reverse channels of açaí solid waste, rendering an experience as close as possible to a final system acceptable for the customers—namely, the links that make up the production and consumption chains of açaí waste.

As a hybrid application, it will be accessible on any device connected to a network and it will preserve its interface standard. The application was developed under the client/server principle, therefore it can be made available on closed local networks (intranets) or hosted on a server with internet access; in the later, we noticed that the application behaves responsively. A screen for changing user registration is shown in Figure 14.
Figure 14. Screen for changing user data in the prototype executed on mobile devices.

Source: The authors.

Teixeira (2014) and Mazza (2012) explain that these resources facilitate prototype evaluation, allowing for better user experience in new contexts through responsive design, whereby screens adapt to several devices. Collection of catalogued information by a web server—to check which browser is used, its type and operating system—takes place through HTTP headers, which includes User-Agents:

What people do is create huge databases of User-Agents. Each browser is tested for all types of important functionalities and is associated with its User-Agent. Then, when a request arrives at the server, you take the User-Agent and search this database for browser characteristics (LOPES, 2013, p. 121).

According to Guerra (2012), we need to understand the standards adopted in an API (Application Programming Interface), which can be located on a web-based platform. In order to solidify understanding and disseminate the use of APIs, companies like Google provide full documentation that facilitates implementing map resources in a web project or mobile application. JavaScript APIs give us the power to implement different mechanisms for a device or an internet browser, and they can be platforms complete with resources in web format (CASSIO, 2014).

The prototype was developed based on an actual model of the system and the business in question; therefore, it can be used as if it were the final project or very close to it, enabling assessment for future improvements. This prototype, like any system
model, features menu items that trigger the necessary actions to start its functionalities. Basically, interaction in this prototype occurs while understanding how to operate waste negotiation and handling process flows. For example, it is possible to view on the map the location of waste producers and consumers, and to list open, pending, and canceled deals, among others, as shown in figure 15.

![Figure 15. Negotiation panel.](image)

**Source:** The authors.

Deal options available in item “My Business”—which displays business interactions and their status—are examples of process flows developed in this prototype. In the process, still within operational flow, each open deal can be confirmed by triggering the “Confirm Deal” button event, which further gives us the idea of an actual system focused on this activity. In the notifications section, triggering the event from the “Unread” menu item will display the notification panel, as shown in Figure 16, with the last notification sent by anyone to the user in question. Then, it is possible to confirm having read the notification when triggering the button event “Mark as Read”, which redirects the message to the “Read” menu. If a notification has already been marked, the button is disabled.
Along with process flows, we defined business rules as the need arose. For example, clicking on a given point on the map will open a deal according to your profile. Deals can only be opened with profiles different from that of the user initiating the deal, as the business rule is comprised of a user with a “Producer” profile and another user with a “Consumer” profile. This is to say that deals between users with the same profile are not accepted, as it is not possible to sell or buy waste to or from yourself.

The negotiation flow consists of opening a deal with a registered user and waiting for the user’s feedback, as exemplified by Figure 17.
Trading takes place externally, by exchanging e-mail, or internally, via the notifications panel. If the negotiation is successful—meaning that, outside the prototype, there was an agreement between the parties and the amount negotiated between them was delivered without any issues of bad conduct or divergent information—, then the closure procedure can be executed by the user who opened the deal: he selects menu item “MY DEALS” and chooses “Pending”, after which he clicks on the “Confirm Deal” button, as shown in Figure 18. When this stage is completed, the prototype will execute the qualification and ranking procedure for that occurrence by assigning a score to confirmed deals, which will appear in the map panel under “SCORES AND RANKING”. In the event of a cancellation, the corresponding notification will be displayed to the user.

**Figure 18. Deal being closed by the user who opened it.**

A user may be negatively qualified by these scores and rank poorly; other users will be able to see if his dealings were good or bad. This process is very important because it provides centralized control over user interaction in the prototype, enabling the generation of management reports with information contained in the database—which is recommended procedure in the management model regarding operations with the tool.

### 4 CONCLUSIONS

With significant overhead in planning and implementing the prototype, we expect that users can generate business with more experience by using a tool already as close as possible to the final system. Prototyping helps in the discovery of new features and supports an interactive and intuitive interface structure.
In the context of an experimental tool, this prototype used fundamentals of Software Engineering combined with a process model based on prototyping. Deliveries were established in increments with agile methods and a Scrum framework, mirrored in a contract between customer and supplier which determines an iterative approach for specification, development, and delivery.

In this scenario of agile methods and prototyping, the customer can request changes according to his needs—even changes in requirements, for quick deliveries in subsequent increments with the guarantee of quality and simplicity, keeping people involved beyond mere processes. Tests for this prototype are back-to-back, undergoing the flow of test data, prototype of a system in parallel with the application system, comparison of results, and reporting of differences. In this work, we propose the development of an application system based on the prototype with a minimum of tests during the application system stage, since the adoption of a prototype minimizes usage errors, validates proposed requirements, and enables a much greater range of acceptance. Prototype usage was tested in a meeting with the project stakeholders.

Through project artifacts used to create the prototype, data mapped by the project teams for registration, alteration, exclusion, negotiation processes, notification, qualification, and ranking were deployed in the database. The proposal was presented in a meeting with the goal of defining the acceptance of the final product, guaranteeing the fulfillment of the prototype’s objectives within stages that involve documentation, implementation, and tests.

One of this work’s strengths are the technologies involved, all of which fit the current market standard and guarantee flexibility and maintenance; versions of frameworks and tools (and their continuance over time) are one of the major concerns in systems design. This prototype was designed to suffer minimum impact when faced with alteration in libraries specific to its functioning, such as the Google Maps API, Bootstrap, and HTML 5 components. This enabled the use of web standards with screens adapted to various devices, pleasant visual environment, and resources required for working with maps. We also chose databases that guarantee portability, compatibility, performance, stability, and ease of use, such as Oracle’s MySQL.

With regard to weaknesses, we can point out the prototype version, which is a web system that works in an internet browser, instead of a mobile application; if it were the latter, portability would be increased by hosting in an application store and native operation in a smartphone. Another point of weakness is the lack of integration with a
Google tool titled My Business; this tool integrates information from establishments flagged on the map, without the need, for example, to create a database of these establishments, or even components to read the data. The last point of weakness is the need to send and receive payments in banking transactions through the prototype similarly to an e-commerce project.

Regarding future work, we intend to apply this project, to broaden the prototype’s coverage considering varied application scenarios, to integrate with other supporting technologies and programming languages. We would also like to experiment with this prototype on other types of waste management and handling, such as mango, buriti, babaçu, and cupuaçu stones, as well as with electronic and other types of waste that represent abundant raw materials—and are always in the cross-eyes for reuse and social benefits.
REFERENCES


