Article **Journal Computer Technology**
June 2023, Vol.7 No.18 23-31

Development of a web application for hotspot token vending machine administration

Desarrollo de aplicación web de administración para máquina expendedora de fichas hotspot

SAMPAYO-RODRÍGUEZ, Carmen Jeannette†*, GONZÁLEZ-AMBRIZ, Rosalba, LUNA-CARRASCO, Claudia Yadira and LUNA-TREJO, Cupertino

Tecnológico Nacional de México - Instituto Tecnológico Superior de Huauchinango, Maestría en Tecnologías de la Información, Ingeniería en Sistemas Computacionales, México.

ID 1st Author: *Carmen Jeannette, Sampayo-Rodriguez* / **ORC ID:** 0000-0001-8844-6055, **CVU CONAHCYT ID:** 951529

ID 1st Co-author: *Rosalba, González-Ambriz* / **ORC ID:** 0000-0001-5400-9754, **CVU CONAHCYT ID:** 368433

ID 2nd Co-author: *Claudia Yadira, Luna-Carrasco* / **ORC ID:** 0000-0002-4092-9987, **CVU CONAHCYT ID:** 368419

ID 3rd Co-author: *Cupertino, Luna-Trejo* / **ORC ID:** 0000-0001-5898-8486, **CVU CONAHCYT ID:** 904398

DOI: 10.35429/JCT.2022.18.8.22.30 Received: January 25, 2023; Accepted June 30, 2023

Abstract

This article describes the features of a web application to manage the availability and acquisition of wireless internet access credentials of vending machine, it followed the OOHMD agile software development methodology, for the implementation we used the client-server architecture, the JavaScript programming language, the JavaScript runtime environment of Node. js JavaScript runtime environment, Express web development framework for Node.js, Sequelize ORM, MySQL relational database management system, Passport authentication middleware, EJS open source template engine, Bootstrap CSS framework, JSON data interchange format and Visual Studio Code IDE. As a contribution, there is a web application that allows the synchronization of available tokens and the sale of a wireless internet access credential vending machine, which connects through the internet. The methodology used allowed to have an adequate development process to obtain a quality product that resulted in a web application that efficiently provides the available access credentials and manages the access credentials acquired from the vending machine.

Methodology, Application, Development, Synchronization, Credentials

Resumen

El presente artículo describe las característica de una aplicación web para administrar la disponibilidad y adquisición de credenciales de acceso a internet inalámbrico de máquina expendedora, se siguió la metodología ágil de desarrollo de software OOHMD, para la implementación se empleó la arquitectura cliente-servidor, el lenguaje de programación de JavaScript, el entorno de ejecución de JavaScript de Node.js, el framework de desarrollo web para Node.js de Express, el ORM de Sequelize, el sistema de gestión de base de datos relacional de MySQL, el middleware de autenticación de Passport, el motor de plantillas de código abierto de EJS, el framework de Bootstrap CSS, el formato de intercambio de datos JSON y el IDE de Visual Estudio Code. Como contribución se cuenta con una aplicación web que permite la sincronización de fichas disponibles y la venta de una máquina expendedora de credenciales de acceso a internet inalámbrico, que se conecta a través de internet. La metodología empleada permitió tener un proceso de desarrollo adecuado para obtener un producto de calidad que dio como resultado una aplicación web que brinda de manera eficiente las credenciales de acceso disponibles y administra las credenciales de acceso adquiridas de la máquina expendedora.

Metodología, Aplicación, Desarrollo, Sincronización, Credenciales

Citation: SAMPAYO-RODRÍGUEZ, Carmen Jeannette, GONZÁLEZ-AMBRIZ, Rosalba, LUNA-CARRASCO, Claudia Yadira and LUNA-TREJO, Cupertino. Development of a web application for hotspot token vending machine administration. Journal Computer Technology. 2023. 7-18:23-31.

* Correspondence to the author (E-mail: carmen.sr@huauchinango.tecnm.mx).
† Researcher contributing as first author.

Introduction

Object-Oriented Hypermedia Design and Hypermedia (OOHDM) is a model focused on the development of hypermedia applications, proposed by Daniel Schwabe and Gustavo Rossi with the objective of simplifying and facilitating the design of hypermedia applications, its stages are: information gathering and conceptual design, navigation design, abstract interface design and implementation. Web applications are tools that allow performing operations from a computer through the internet and according to them for the development of web applications, the most used methodology is the agile methodology and they observed that OOHDM meets the most optimal method (Ríos, Ordóñez, Segarra, & Zerda, 2017).

JavaScript is an interpreted programming language used primarily to create interactive web applications. It runs on the client side (web browser) and allows user interaction, manipulation of the DOM (Document Object Model) and the creation of animations and visual effects on the web page. It is also used on the server side (Node.js) to create web applications and APIs (Mozilla, s.f.)

Node.js is a JavaScript runtime environment based on Google Chrome's V8 JavaScript engine. It was created by Ryan Dahl in 2009 and allows developers to use JavaScript both client-side and server-side. Node.js is an open source server-side programming platform that uses a non-blocking, event-driven input/output model to provide fast, scalable performance (Kinsta, 2021).

Express is a web development framework for Node.js that allows you to create web applications and APIs quickly and easily. It was created in 2010 by TJ Holowaychuk and is currently one of the most popular and widely used frameworks in Node.js. Express provides a number of features and tools for developing web applications, such as routing, middleware, integration with other Node.js modules, error handling and RESTful API creation. In addition, it is highly customizable and allows developers to define their own application architecture. (Express, n.d.; Startechup, 2021).

Sequelize is a Node.js ORM (Object-Relational Mapping) that allows developers to interact with relational databases using JavaScript objects. Sequelize is compatible with different relational databases, such as PostgreSQL, MySQL, SQLite, among others (Sequelize, n.d.).

MySQL is an open source relational database management system (RDBMS) widely used in web applications. It was originally created by Michael Widenius and David Axmark in 1995, and has since been acquired by several companies. MySQL is compatible with multiple programming languages, such as PHP, Python, Java, Node.js, among others, and is used in a wide variety of applications, from small web applications to large enterprise systems. (MySQL, n.d.; IONOS, 2022).

Passport is an authentication middleware for Node.js that is used to authenticate requests in web applications. It provides a set of authentication strategies that allow developers to authenticate requests using different services, from local authentication to services such as Google, Facebook, Twitter, among others (Passport, n.d.).

EJS (Embedded JavaScript) is an open source template engine for JavaScript that runs on the server. It allows developers to create dynamic web pages by generating HTML through templates containing JavaScript code and HTML markup. (EJS, n.d.)

Bootstrap CSS is a popular framework, which includes a set of predefined styles, classes and components of HTML, CSS and JavaScript to help developers create web pages quickly and efficiently. It was originally created by Twitter and is now maintained by the open source community. It offers a wide variety of features, such as a responsive grid system, typography, forms, buttons, navigation, modals, alerts and much more. (RockContent, 2021).

JSON is the acronym for "JavaScript Object Notation". It is a lightweight data interchange format that is easy for humans and machines to read and write. It is mainly used to transfer structured data over the web (MDN Web Docs, n.d.).

ISSN 2531-2197

Visual Studio Code is an open source integrated development environment (IDE) developed by Microsoft that is used to write, debug and enhance software code. It is also highly customizable and supports a wide range of extensions and plug-ins that can help improve developer productivity. It supports several programming languages and is widely used by software developers and programmers around the world (Microsoft Corporation, 2021).

The article describes the methodology for the development of a web application that allows the administration, availability and token sales (credentials, username and password) of a token vending machine to connect to a wireless internet access point (hotspot) of a wireless internet service provider (wisp) that sells access credentials on a prepaid basis.

The importance of this web application is that it facilitates the management of available credentials and sales control for the vending machine from any location where internet access is available and eliminates the need to go on site to add more available credentials to the machine.

Although there are several solutions that could be carried out to have the web application and one of them prior to this article, the administration of credentials from google services using a google datasheet was presented. However, there was a need for a customized web application with a friendly and easy to use interface.

And as mentioned by Pandurman (2023) data from different sensor devices can be accepted directly and through network connections, which will be used in real time for user interfaces, text files and access to other systems through Representational State Transfer Application Programming Interface (REST API) services.

A solution was sought for the web application to manage the credentials and perform the cuts of tokens sold from various vending machines using a REST API.

The problem that is mainly addressed is that the vending machines will be placed in rural communities that are difficult to access or are located far from the city, and if the access credentials, which were originally sold in paper cards, had to be printed and taken to the locality when they ran out, it is now possible through the web application to load them and have them available when required, making it easier to determine the amount of credentials sold and the cash amount to be held at the time of withdrawal. In addition, it reduces the use of paper printing and allows local people to acquire them in self-service format at any time, 24 hours a day.
In the first section of the article you will find descriptions of the terms used as the methodology employed and the technology used for the development of the application, a brief description of the problem and objective, followed by the description of the activities carried out by stages of the methodology that was worked, results where the pages of the application are shown and the functionality for vending machine management is described, the acknowledgements to our funding source, the conclusions and finally the references or sources of consultation used.

Methodology to be developed

The methodology used for the development of the web application was the OOHDM.

The artifacts generated by each stage are presented below:

Stage 1. Requirements gathering and conceptual design

The functional requirements of the application were as follows:

RF01: The application should allow registering a system administrator and request username and password to log in.

RF02: The application should show general information on available tokens sold, total tokens held and show the profile of the most sold tokens, statistics by profile of available tokens, sold tokens and total tokens held.

RF03: The application must allow adding new cards under the command format: name=AA00001 password=276 limit-uptime=02:00:00 profile= "PLAN_TEST".

ISSN 2531-2197

RF04: The application must allow to perform search filters by cards and profiles.

RF05: The application must have a sales section and show the amount of total sales and by dispatching machines per profile.

RF06: The application must allow the following configurations: change the system user, change the password, add new profiles, edit existing profiles, activate or deactivate and delete them, add new vending machines, edit existing ones, deactivate or activate and delete them, delete the history of tokens sold by different search filters.

RF07: La aplicación debe permitir el cierre de sesión.

Figure 1 shows the use case diagram of the web application.

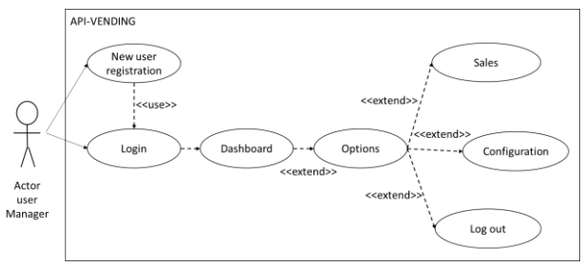


Figure 1 Diagram of use cases
Source: Own elaboration

Stage 2 Navigation design

Figure 2 shows the navigability map of the application.

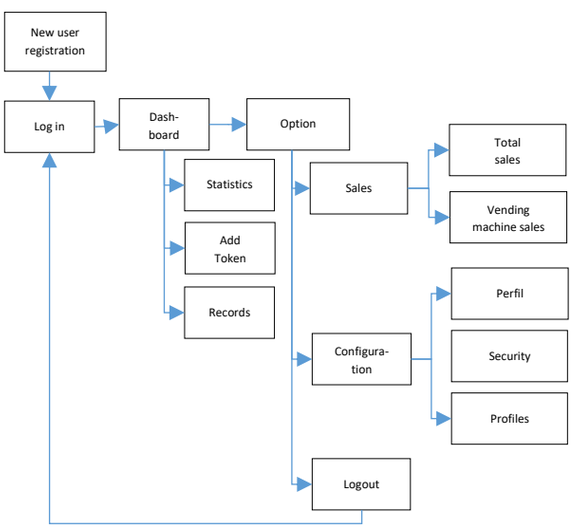
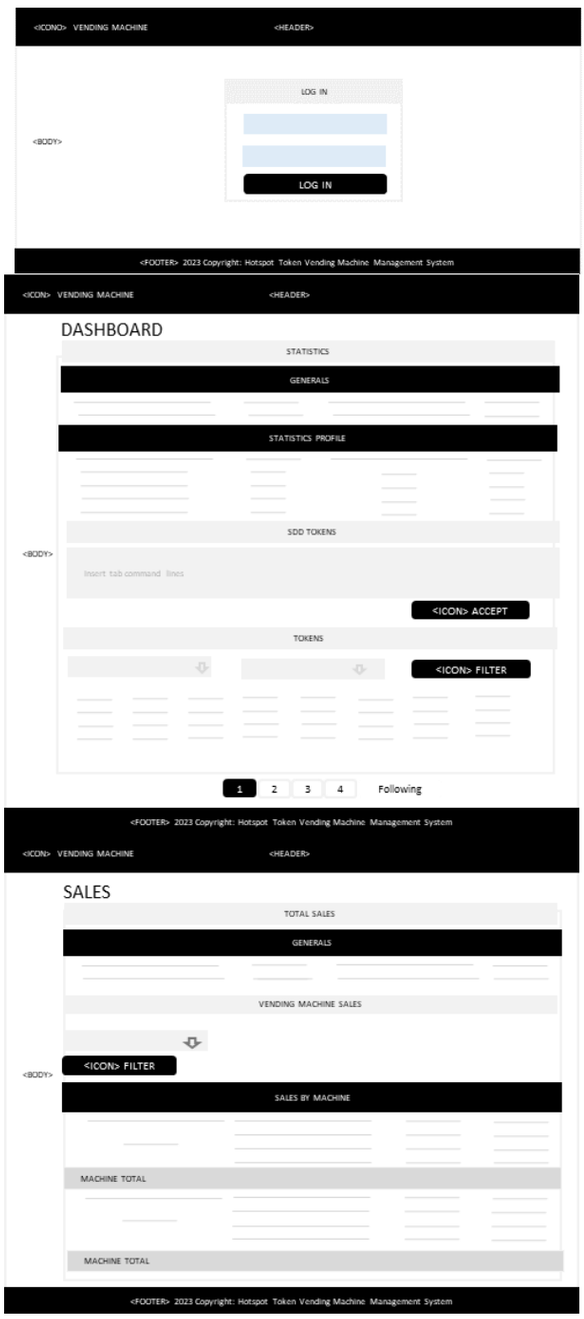


Figure 2 Navigability map
Source: Own elaboration

Stage 3 Abstract interface design

The Abstract Data View (ADV) model was used to design the abstract interface.



ISSN 2531-2197

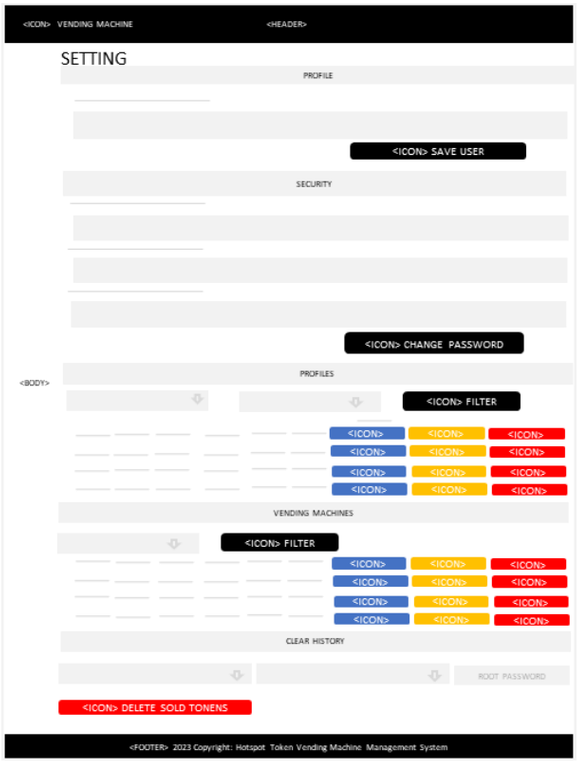


Figure 3 Abstract views
Source: Own elaboration

Stage 4. Implementation

The hotspot token vending machine web application follows a client-server architecture. The server side has been developed using the JavaScript programming language, through the Node.js runtime environment and the Express framework. In addition, the application has a connection to a MySQL database, accessed through the Sequelize ORM, which facilitates interaction with the stored data.

For user authentication, the system uses the Passport authentication middleware, which allows easy administration of system users and their sessions. As for the graphical interfaces of the application, they are generated by the server using the EJS template engine and sent to the user's web browser.

From the client side, there are two possible ways to interact with the system. The first is through a web browser, accessing the HTTP address where the system is hosted. In this way, the user can make requests to the server using the HTTP protocol and obtain the views and data generated by the system, allowing him to manage the tokens, the vending machines and the sales made.

The second way to interact with the system is through a vending machine. In order to carry out queries and sales of the tokens stored in the system, it is necessary to properly configure the machine. The interaction is done through the JSON format, by sending requests to the server and receiving responses.

Figure 4 shows the client-server architecture used for the application.

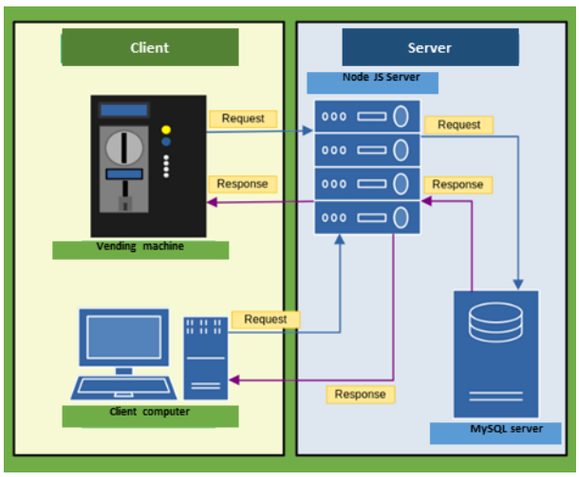


Figure 4 Client-server architecture. ***Own elaboration.***

Figure 5 shows the relational diagram of the classes, attributes and operations of the application.

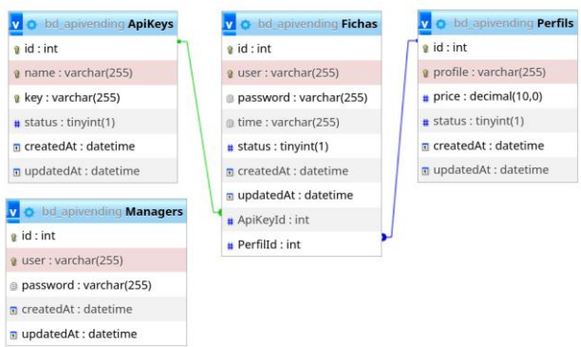


Figure 5 Relational diagram. *Own elaboration.*

Table 1 shows the routes of the web application.

Route	HTTP Method	Request	Response	Description
/	GET		HTML	System login page
/signin	GET		Redirect /	System login page
/signin	POST	user, password	HTML	System Login
/signup	GET		HTML	System registration page
/signup	POST	user, password, passwordConfirm	HTML	System registration
/logout	GET		Redirect /	Logout
/change-user	POST	user	Redirect /config	Change user name
/change-password	POST	password, newPassword, newPasswordConfirm	Redirect /config	Change user password
/config	GET		HTML	Configuration page
/dashboard?status=&profile=&vending=&page=	GET		HTML	Administration page
/addProfile	POST	profile, price	Redirect /config	Add new profile
/change-profile-status/id	GET		Redirect /config	Change profile status
/edit-profile/id	GET		HTML	Edit profile page
/edit-profile/id	POST	profile, price, status	Redirect /config	Edit a profile
/delete-profile/id	GET		Redirect /config	Delete a profile
/addKey	POST	name	Redirect /config	Add new vending machine
/change-key-status/id	GET		Redirect /config	Change vending machine status
/edit-vending/id	GET		HTML	Edit vending machine pagevending
/edit-vending/id	POST	name, key, status	Redirect /config	Edit a vending machine
/delete-vending/id	GET		Redirect /config	Delete vending machine
/fichas	POST	fichas	Redirect /dashboard	Add tokens
/clearHistory	POST	range	Redirect /config	Delete cards
/vending	POST	JSON { "key":"" }	JSON { "estatus": "Correcto", "result": [{ "perfil": "", "precio": 0, "disponibles": 0 }] }	Available profiles
/vending/price	POST	JSON { "key":"" }	JSON { "estatus": "Correcto", "result": { "user": "", "password": "", "precio": 0 } }	Sold hotspot profile

Table 1 Routes of the application
Source: Own elaboration

Results

Figure 6 shows the first login page to the application, where you must register a new user who will be the system administrator, enter name and password and password confirmation, then you can log in.

Figure 6 New user registration page
Source: Own elaboration

Figure 7 shows the login page.

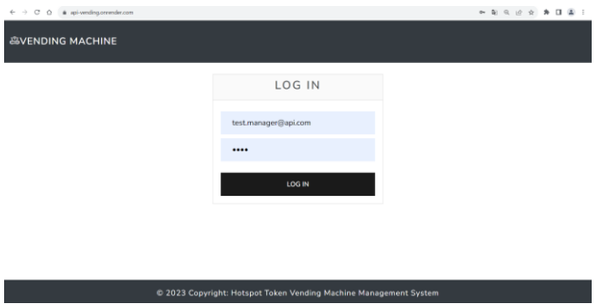


Figure 7 Login page
Source: Own elaboration

When logging in with the registered user and password you will enter the application and see the main page divided into 3 sections: statistics, add cards and cards, as shown in Figure 8.

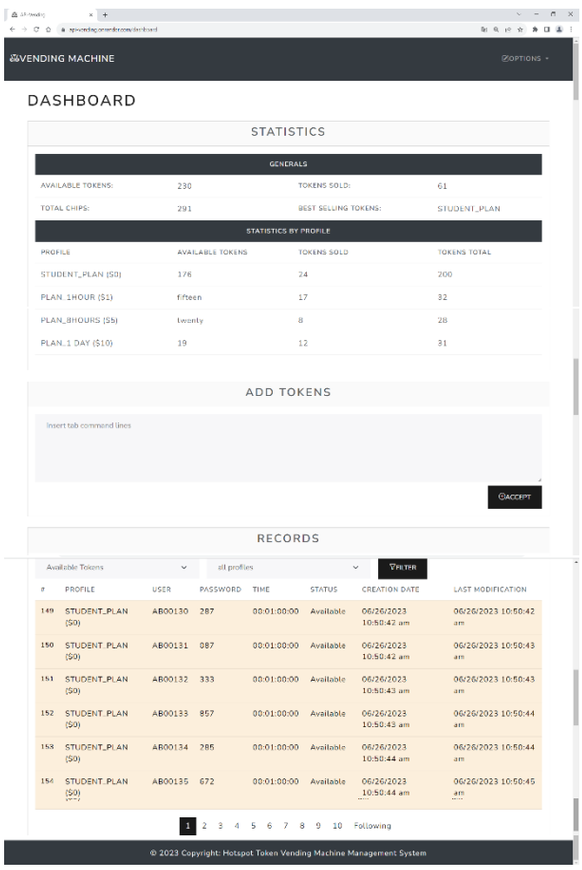


Figure 8 Dashboard page
Source: Own elaboration

1. Statistics: this section displays general statistics showing information on available, sold, total and most sold tokens; and statistics by profile showing available, sold and total tokens.

2. Add tokens: in this section the commands are entered with the mandatory data structure: name=AA00001 password=276 limit-uptime=02:00:00 profile= "PLAN_TEST", shown in Figure 9.

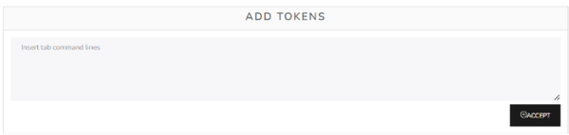


Figure 9 Add tab section of the statistics page
Source: Own elaboration

3. Cards: this section displays cards by applying any of the search filters: by available cards, sold cards and all cards of all profiles or by any other profile that is registered.

On the upper right side is the "Options" drop-down menu that allows you to go to the sales, configuration or logout page. As shown in Figure 10.

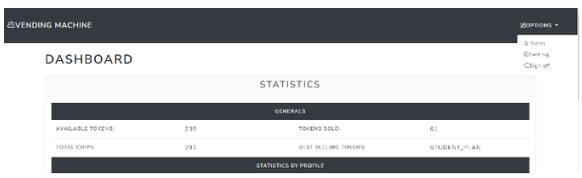


Figure 10 Options menu
Source: Own elaboration

Sales: this option shows the information on sales made and is divided into two sections: total sales and sales of the vending machine. As shown in Figure 11.

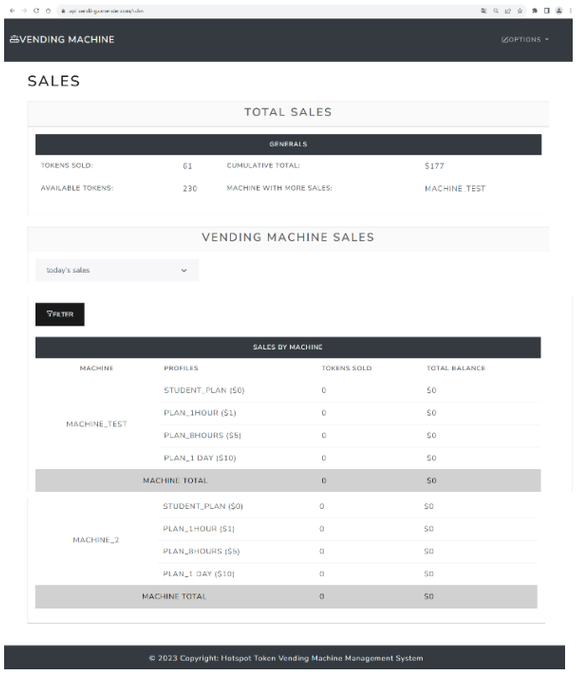


Figure 11 Sales page
Source: Own elaboration

The sales section shows general information on: tokens sold, tokens available, total accumulated amount and top selling machine.

The vending machine sales section displays machine information, profiles, tokens sold and accumulated balance by the selected sales search filters: today, yesterday, week, last week, month, last month, by custom date range and all sales.

2. Configuration: this option allows making the general configurations of the application for its operation and is divided into three sections: profile, security and profiles, vending machine and clear history. As shown in Figure 12.

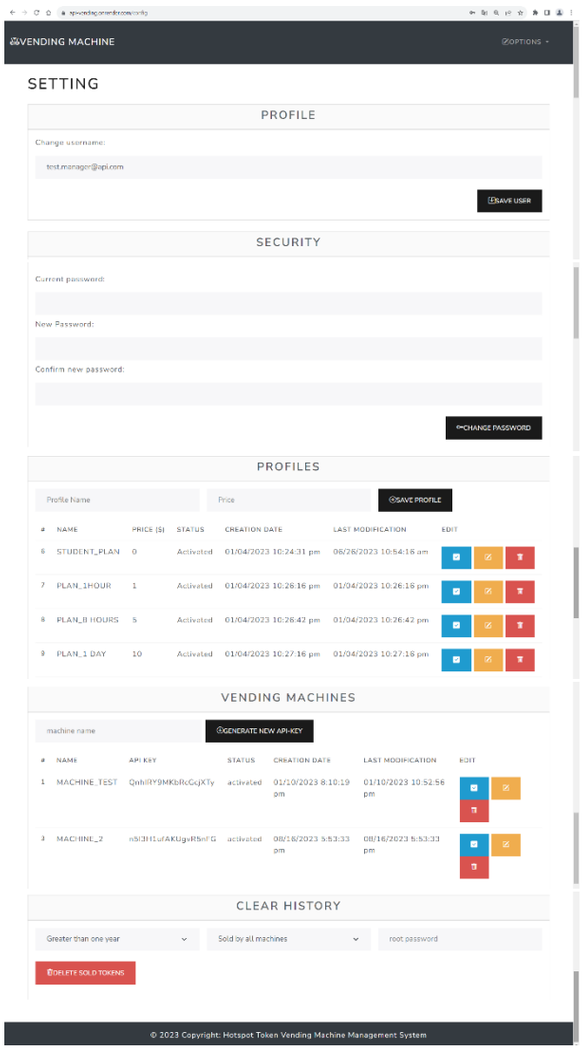


Figure 12 Configuration page
Source: Own elaboration

- Profile: this section allows you to change the name of the web application user.

- Security: this section allows you to change the password to enter the system.

- Profiles: this section allows to create and save the profiles of the cards: name, price, status, creation date, last modification. Once the profiles have been created, they can be edited: activate or deactivate the profile, edit the selected profile and delete the selected profile.

- Vending machine: this section allows to name the machine and generate a new api-key. Once created, it shows the machine information: name, api-key, status, creation date, last modification and the edit buttons to activate or deactivate the machine, edit and delete.

- Clear history: this section allows deleting sold cards by date: older than one year, six, three or one month, all sold, all cards - reset of sold for all machines or a specific registered machine. Such deletion of the history is with the confirmation of the password of the root administrator of the application.

- Logout: this option closes the web application by returning to the web application authentication page.

Acknowledgment

We thank the Tecnológico Nacional de México/Instituto Tecnológico Superior de Huachinango (TECNM/ITSH), the Master in Information Technologies, the Computer Systems Engineering program and the "Intelligent Computing" academic group with code ITESHUAU-CA-3 for the support provided.

Financing

This article has been funded by TECNM/ITSH [SPeI-SR-014 -2023].

Conclusions

The methodology used allowed to have an adequate development process to obtain a quality product that resulted in a web application that efficiently provides the available access credentials and manage the access credentials purchased from the vending machine.

There is a web application hosted on a free onreder.com server, which allows to manage the availability and sale of wireless internet access credentials of a vending machine.

The application works adequately, however one of the improvements that can be made is to upload it to a paid server to make the connection between the vending machine and the application more efficient in order to reduce the communication time with the server.

ISSN 2531-2197

References

EJS. (s.f.). *Embedded JavaScript templating.* https://ejs.co/

Express. (s.f.). *Fast, unopinionated, minimalist web framework for Node.js.* https://expressjs.com/

IONOS. (28 de noviembre de 2022). *¿Qué es MySQL?*. https://www.ionos.es/digitalguide/servidores/know-how/que-es-mysql/

Kinsta. (13 de mayo de 2021). *¿Qué es Node.js?*. https://kinsta.com/es/base-de-conocimiento/que-es-node-js/

MDN Web Docs. (s.f.). *JSON*. https://developer.mozilla.org/es/docs/Learn/JavaScript/Objects/JSON

Microsoft Corporation. (2021). *Visual Studio Code.* [Computer software]. https://code.visualstudio.com/

Mozilla. (s.f.). *JavaScript. Mozilla Developer Network.* https://developer.mozilla.org/es/docs/Web/JavaScript

MySQL. (s.f.). *MySQL: MySQL Community Server.* https://www.mysql.com/

Passport. (s.f.). *Simple, unobtrusive authentication for Node.js.* http://www.passportjs.org/

Panduman, YYF, Funabiki, N., Ito, S., Husna, R., Kuribayashi, M., Okayasu, M., ... &Sukaridhoto, S. (2023). Un Edge DeviceFramework en la Plataforma de Servidor de Aplicaciones IoT de SEMAR. Información, 14 (6), 312.

Ríos, J. R. M., Ordóñez, M. P. Z., Segarra, M. J. C., & Zerda, F. G. G. (2017). Estado del arte: Metodologías de desarrollo en aplicaciones web. *3c Tecnología: glosas de innovación aplicadas a la pyme*, 6(3), 54-71. https://dialnet.unirioja.es/servlet/articulo?codigo=6143045

RockContent. (22 septiembre de 2021). *¿Qué es Bootstrap?* https://rockcontent.com/es/blog/bootstrap/

Sequelize. (s.f.). *A promise-based ORM for Node.js.* https://sequelize.org/

Startechup. (15 de enero de 2021). *Express.js: qué es, para qué se usa y cuándo y dónde utilizarlo en el desarrollo de aplicaciones empresariales.* https://www.startechup.com/es/blog/express-js-what-it-is-used-for-and-when-where-to-use-it-for-your-enterprise-app-development/.

ISSN 2531-2197