

# LibreDTE: Software Tools and Support for Electronic Invoicing in Chile

Esteban De La Fuente Rubio\* and Miguel Gutiérrez Gaitán<sup>†‡</sup>

\*SASCO SpA, Av. Daniel Barros Grez 191, Santa Cruz, O'Higgins, Chile

<sup>†</sup>Facultad de Ingeniería, Universidad Andres Bello, Santiago, Chile

<sup>‡</sup>CISTER/ISEP and CISTER/FEUP, Porto, Portugal

**Abstract**—This work introduces LibreDTE, an initiative providing software tools and support for electronic invoicing in Chile. LibreDTE is mainly built upon *libredte-lib*, a free software library that directly connects with the Chilean Internal Revenue Service, thus enabling automatic emissions of official e-invoices and other electronic tax documents. LibreDTE is either: (1) the first (and so far, the only one) Chilean framework for free-software-driven e-invoicing, and (2) an official software solution with featured e-business capabilities. In this paper, we describe both, the *community* (or free software) version, and the *official* (or commercial) version of LibreDTE. We focus on their primary building blocks and major technical differences, and we show, in a tutorial way, some of the key design considerations behind their common e-invoice generation. We also discuss some lessons learned from earlier implementations, as well as the latest (and promising) features incorporated within the official version.

**Index Terms**—B2B, Digital Transformation, DTE, e-invoicing, electronic billing, electronic tax document, free software.

## I. INTRODUCTION

Undoubtedly, electronic invoicing and electronic billing (or e-invoicing and e-billing, respectively) have become standard practices worldwide [1]. Benefits such as cost, time and carbon footprint reduction are crystal clear, especially when compared to the more traditional paper-based processes [2]–[4]. This is accompanied by the additional advantages in terms of the accuracy and scalability provided by software-based solutions, which make business processes more efficient. In fact, digitization on the management of commercial transactions has, in general, significantly improved the electronic handling of business-to-business (B2B), business-to-consumer (B2C) and business-to-government (B2G) digital operations. Benefits that are well-known by governments around the world, which, indeed, have been promoting its adoption for years.

More recently, with the mandatory regulations being imposed in some regions, including countries across Europe and Latin America, the adoption of some sort of electronic billing scheme has become massive. This has been intensified due to the COVID-19 pandemic, which has driven further legal impositions in areas where before e-commerce was still scarce [5]. All in all, the migration of companies and taxpayers to digital is happening at a large scale, thus creating new business opportunities for software-based services companies. This, not only from the perspective of the mandatory requirements each government may imposed, but beyond, e.g., leveraging on the potential of model-driven approaches to offer flexible

and maintainable software solutions that may overcome the constant changes in regulations worldwide [6].

In Chile, governmental efforts started in 2002 with Law No. 19,799, which stipulated the first set of regulations for handling and operating with Electronic Documents, Electronic Signatures and Certification of Signatures. This includes the so-called DTEs (or Electronic Tax Documents) of which the present work is concerned. However, it was only in 2017 (i.e., ~15 years later), when the adoption of electronic billing/invoicing become really predominant. Before this, the adoption was encouraged but voluntary, and thus it was a real need (and practice) mostly for e-commerce-based businesses.

In effect, it was Law No. 20,727, which triggered the greater digitization in Chile, stipulating the issuing of electronic invoices as mandatory for all companies. This started gradually with the larger companies in the 2014–2016 period, and it became massive only in 2017 when micro-businesses were forced to migrate to the current mandatory electronic billing scheme. More recently, in 2020, Law No. 21,210 on tax modernisation added further commercial activities into the regulatory framework (e.g., those falling within the B2C model), further expanding its scope and impact.

All these laws and related mandates, resolutions, decrees, etc., issued over the span of the last 20 years in Chile, have provided the legal basis and thus the market opportunities for an emerging group of software companies that nowadays design tailored solutions for e-billing processes, electronic tickets, and related electronic documents; often framed within the more common private and proprietary approach. This, of course, is in line with the global panorama, where the number of companies of this kind is vast. Yet, free-software-driven electronic billing and invoicing alternatives around the world are still scarce, and indeed, almost inexistent in some geographical regions or countries; as in the case of Chile.

In this work, we present LibreDTE, a free-software-driven initiative providing software tools and support for electronic invoicing in Chile. LibreDTE is mainly built upon *libredte-lib* [7], a PHP-based library that directly connects with the Chilean Internal Revenue Service, thus enabling automatic emissions of official e-invoices and other electronic tax documents. LibreDTE is either: (1) the first (and so far, the only one) Chilean framework for free-software-driven e-invoicing, and (2) an official software solution with featured e-business capabilities. Through this paper we introduce both, the *commu-*

nity (or free software) version, and the *official* (or commercial) version of LibreDTE, focusing on their primary building blocks and major technical differences.

The rest of this paper is organized as follows. Section II introduces some background on the basic Chilean DTE billing schema. Section III describes some key design considerations and underlying implementation aspects common to both LibreDTE versions. Section IV goes into more details on each of the versions, highlighting major technical differences and specifying their scope. Section VI concludes the paper.

## II. BACKGROUND: CHILEAN DTE BILLING SCHEMA

### A. DTE: A definition

In Chile, an Electronic Tax Documents or DTE (from the Spanish, *Documento Tributario Electrónico*) can be defined as any digital document that is exchanged electronically between different entities as tax receipts. A DTE refers, in general, to the XML (Extensible Markup Language) representation of an invoice or other tax document that is signed electronically, and thus, it is recognized for official tax purposes. A DTE can thus represent an *invoice* document, e.g., resulting from a classical B2B operation, or a *ticket*, e.g., derived from the local (Chilean) definition of a B2C transaction. Furthermore, it can also refer to others (more specific) types of tax documents that go beyond the discussions of this paper.

As an illustrative example, we can say that the XML generated as the DTE is equivalent to a paper document that is both stamped and signed. The stamp is here represented by an electronic authorization indicating that the document is official (or in other words, that can be officially issued by the company); while the signature, is the electronic method providing the authenticity component to the document.

### B. The DTE Billing Schema

The DTE concept may also refer to the whole electronic billing schema operating in Chile, i.e., the whole process required to complete a specific (electronic) commercial transaction. This involves: (i) the adoption of set requirements to be agreed by both the issuer(s) and the receiver(s), (ii) the management of the transactions between those entities, and (iii) the set of official communication performed with the Chilean Internal Revenue Service, also known as SII (from the Spanish, *Servicio de Impuestos Internos*). An illustration summarizing this schema is presented in Figure 1.

As previously mentioned, today, in Chile, it is mandatory to follow this billing schema when handling any B2B or B2C electronic transactions. This requirement can be satisfied, e.g., (i) by relying on the free (but very limited) software alternative provided by the SII, or (ii) by acquiring (and employing) one of the commercial software solutions provided by the numerous companies that has been registered to operate within the SII. Note that these companies only fulfill an administrative registration to operate, which is not a technical guarantee neither a certification of a proper (electronic) operation.

In this work, we focus on LibreDTE, which is both, (1) a free software alternative for issuing e-invoices following the

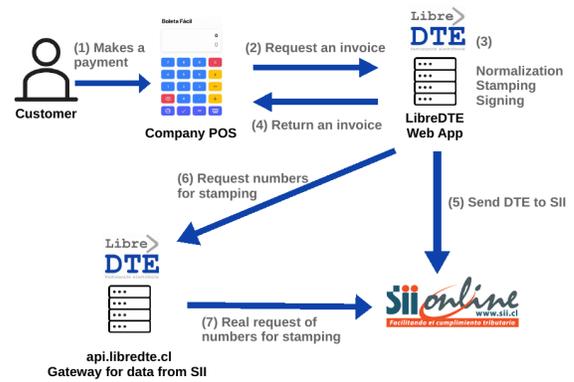


Fig. 1. The Chilean DTE billing schema with LibreDTE.

national billing schema, and (2) a commercial and featured software solution provided by SASCO SpA, one of the many registered companies to offer this kind of services in Chile. Note that the official version provides at least the same minimum functionalities than the community version, but it also includes featured and seamlessly integrated services for advanced and massive handling of e-commerce transactions. Both of these versions are based on *libredte-lib*, the free software library providing the underlying mechanisms for the e-invoice (or DTE) generation.

## III. LIBREDTE: GENERAL ASPECTS

LibreDTE [8], firstly released in October 2015, has experienced several changes since its conception, yet, always keeping the essence of its vision “offering a free-software DTE generation system for Chilean companies”. This sort of promise is mainly kept by the continuous update and maintenance of the core of the LibreDTE project, i.e., the PHP-based free software library *libredte-lib*. This library is responsible for providing the minimum building blocks for the underlying e-invoice (or DTE) generation; which is the essential feature of both LibreDTE versions. The implementation of the DTE generation, although trivial now, experimented several changes during the earlier versions, and thereby it was not a so straightforward design decision at that moment.

In this section, we reveal, in a tutorial way, some of the lessons learned from earlier implementations, and the reasons behind the design and implementations decisions that shaped LibreDTE into what is today.

### A. A PHP-based standard library for LibreDTE

This subsection presents *libredte-lib* [7], the main software library implementing the core functionalities for the DTE generation. These basic functionalities are enumerated as follows:

- 1) XML generation
- 2) Electronic signature
- 3) Sending the DTE to the SII

Within the next paragraphs, we will explain each of these functionalities in more detail, providing some of the intuitions behind its design and implementation decisions.

Note that other minor functionalities, e.g., the CSV and PDF generation, the different types of DTE generation, etc., are not covered within this paper but can be fully checked within their respective project websites.

**1) XML generation:** As mentioned before, a DTE is an official document representing a transaction that is required to be in an XML format. This allows both standard human-readable and machine-readable document encoding. While generating an XML is, in general, a straightforward task, there are, indeed, many different ways of producing equivalent results. Thus, we explored different options for the DTE document generation, and then we implemented within the LibreDTE core, the one which showed better performance.

For completeness, we will present here a few of the design and implementation alternatives evaluated. Note that for evaluation purposes, we have considered the case of simple XML file definition as the one presented as follows:

```
<root>
  <node1>A</node1>
  <node2>B</node2>
</root>
```

Given this, we present four of the options that although capable of producing an XML representation of a DTE, did not offer the best performance results.

**1.a) Concatenate.** This option considered the possibility of mixing strings with variables that then are added (or concatenated) within a variable as strings. This intricate option produced poor performance results since when a node needed to be edited, the access to the node was very costly (mainly in terms of memory and computation resources), thus not representing a scalable solution.

For completeness, we present this option as follows:

```
$xml = '<root>';
$xml .= '<node1>' . $value1 . '</node1>';
$xml .= '<node2>' . $value2 . '</node2>';
$xml .= '</root>';
```

**1.b) Template in variable.** This option is alike to the prior one presented, but instead of using a variable, we used a "replaceable" field within the string. However, this option showed similar performance issues than its predecessor.

**1.c) Template from a file.** This option is also similar to the first one, but now more comparable to option **1.b)**, specifically, having now the string within a file instead of a variable. However, this option didn't offer any performance improvement. In practice, it just moved the template to a file.

**1.d) Use a DomDocument for each type of XML.** This option showed better performance but still being not scalable enough since difficult to maintain. This because it requires to create manually (thus statically) each of the nodes, which does not match well with the possibility of having dynamic nodes.

This option is presented as follows:

```
$xml = new DomDocument();
$root = new DOMElement('root');
$xml->appendChild($root);
$root->appendChild(
    new DOMElement('node1', $value1)
);
$root->appendChild(
    new DOMElement('node2', $value2)
);
```

All these four possibilities were indeed feasible proof-of-concepts of the DTE generation, but they all shared the same issue, i.e., it was required to have static and well-known node definitions beforehand.

**1.e) Some notes on the preliminary definitions behind the proposed option.** In LibreDTE we opted for a more generic, dynamic and scalable DTE generation strategy, targeting the support for external entities (nodes) that, e.g., may not have been defined yet by the Chilean Internal Revenue Service. To this purpose, we proposed to include additional requirements to the XML generation scheme:

- **REQ1:** To enable the dynamic addition of new nodes to the XML, without modifying its source code generator.
- **REQ2:** To enable the seamless conversion from the JSON format to the XML format.

We note that this second requirement aroused because a JSON file can be easily read from any PHP array, and in turn, a PHP array can be saved as a JSON without many difficulties. In other words, this requirement allowed us to include a generic input to the software using JSON, which in PHP it is treated as an array. This can be interpreted too as the option of being able to create an XML from the following:

```
{
  "root": {
    "node1": "A",
    "node2": "B"
  }
}
```

Given this, we target a DTE generation option that is able to provide the following:

- A quick access to the data in PHP, both for writing and reading operations.
- An easy way to modify the array, i.e., adding or removing elements dynamically.
- An unrestricted definition for the indexes of the array, thus allowing to add new nodes when required, e.g., when defined by the Chilean Internal Revenue Service.

**1.f) Recursive search in PHP array.** This corresponds to the proposed (and implemented) solution. It consists in having

a recursive method able to search within the PHP array the required element, to then generate the XML from it.

```
$XML = new \sasco\LibreDTE\XML()->generate([
    'root' => [
        'nodo1' => $valor1,
        'nodo2' => $valor2,
    ]
]);
```

The full implementation of this method counts only 50 lines of code, and essentially, it allows to generate an XML file from a PHP array, which in turn, can come from data available in JSON. Similarly, it allows to convert an XML file to a PHP array, and then save it as a JSON file. This design option fulfils the two new requirements (REQ1 and REQ2) to work with the DTE data as defined within the LibreDTE framework.

Another relevant aspect of this option is the fact that data in LibreDTE is encoded in UTF-8. Yet, before to save the data, there is a class responsible to automatically convert it to the ISO8859-1 format, which is the encoding used by the SII. This is done with the aim that the underlying structure of the data matches exactly the same as the one provided officially by the SII. By doing this, we emphasize the fact that there is no proprietary structure, neither of files, formats or tags, having specific (proprietary) definitions; thereby, everything is done generically. This has the advantage that at any (eventual) modification that the SII could make, the XML format will be immediately supported by the LibreDTE XML generator, without the need for further adaptations. We believe this feature provides freedom to the programmer by not depending on methods, e.g. to add the company name or the money order (or any other node) into the XML.

Nevertheless, a possible limitation to this solution is the verification and validation of the data, which may be required by other methods, e.g., when verifying fields such as the national ID number (a.k.a. in Chile as RUT). In any case, in *libredte-lib* this is not prevented, but it is left to be done by the higher layers, and by using different mechanisms. In particular, LibreDTE supports this process by using:

- (i) First, what we call *normalization* process, i.e., a significant reduction in the amount of data to be provided when building a DTE.
- (ii) Second, a validation method relying on the original SII definition only, i.e., by using the format/schema that the SII defines, without any additional verification process.

This whole process allows the detection of common format errors, e.g., when improper tags or formats are used within the XML. Thus, by choosing this, LibreDTE does not really need an extra validation of the format of the data, because it is always enforcing to use the schema that the SII provides.

2) **Electronic Signature:** In general, the process of how to obtain an electronic signature is well-known and standard. For the DTE case, it is indeed similar, but we briefly summarize it here for completeness.

- First, the nodes where the data will be stored are created. These nodes start with the tag `Signature`.

- Then, the `DigestValue` is calculated. This is done using the SHA1 base64 hash of the tag to be signed.
- After that, the `SignatureValue` is obtained. This is the signature of the `SignedInfo` node that contains the `DigestValue` previously calculated.
- Finally, the `SignatureValue` is added to the node `Signature`, along other values associated with it.

While at first sight, this process may seem simple, it can be further simplified in practice, e.g., if using the CryptoSys's XML signature generation, as explained in [9]. Particularly, this way of generating the electronic signature has been adapted to be used within LibreDTE, and in specific, to work with PHP. We further note that this mechanism, as well as the previously explained XML generation, are fully available online, since part of the LibreDTE community version, which will be further explained in the next section.

As a result of this simplification, a software developer that wants to sign an XML file using LibreDTE can do it as simple as follows:

```
$DTE = new \sasco\LibreDTE\Sii\Dte($dte);
$DTE->timbrar($InvoicesNumbers);
$DTE->firmar($Signature);
```

where:

- `$DTE` is the PHP array with the DTE data.
- `$InvoicesNumbers` is an object with the data from the so-called CAF file, i.e. the list of invoices authorized by the SII.
- `$Signature` is the object with the electronic signature (or digital certificate).

3) **Sending the DTE to the SII:** This is the last step of the DTE generation process. It involves the sending process, and the validation and verification tasks from the SII. After the SII determines that everything is correct according to their requirements, the XML sent is accepted and thus will it becomes legally valid. On the contrary, if the XML is rejected, this means that the generated DTE has errors, thus, it is not legally valid, and must be replaced by a correct one. All in all, the XML sending process toward the SII is simple, but it requires an XML correctly generated and signed. Having that done, the following steps are then performed:

- 1) A random seed from the SII is obtained.
- 2) By using the seed, a token for the session is obtained.
- 3) After the token is obtained, the user can be authenticated at the SII, and then send the XML.
- 4) After that, a new token is obtained to check the status of the sent file at the SII.

A fully functional version of this process is available at the LibreDTE community website [8].

#### IV. LIBREDTE: COMMUNITY AND OFFICIAL VERSIONS

This section provides further details and technical differences between the two versions of LibreDTE. It starts with the LibreDTE community version, which indeed shares basic (or minimal) functioning with the official version. Then, it

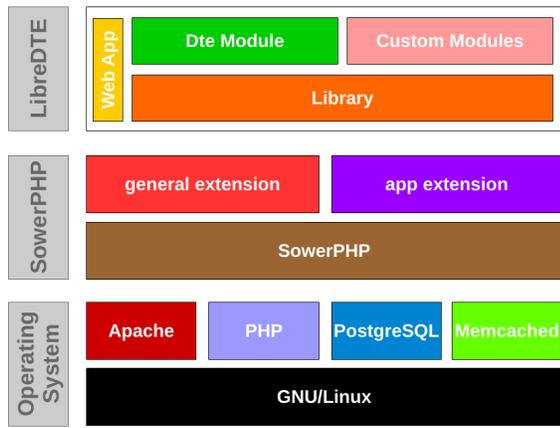


Fig. 2. Architectural Components of the LibreDTE Community Version

addresses the extra features and tools available within the official version.

### A. LibreDTE - Community Version

This version is intended to be used by end-users, i.e., the companies who need to issue the DTE. It is based on the core functionalities of *libredte-lib*, which, although available for end-users, is intended to be accessed by software developers. It is important to mention that the community version that the user installs is still subject to the license and legal terms and conditions available at the project website [10].

The objective of the LibreDTE community version is then to offer a web-based platform for all Chilean companies as free software. The general architecture of this version is presented in Figure 2, and further explained within the next subsections.

**1) Operating System:** LibreDTE runs on top of the GNU/Linux Kernel, where it uses PHP, Apache, PostgreSQL and Memcached as the most salient components. The two first are relevant for the already addressed *libredte-lib*, while the latter two are further explained next. In particular, PostgreSQL is used to directly query data in XML within the database. This, as contrary to the more common practice of storing XML in files. We opted for this since allowing us to provide faster filtering and extracting data capabilities using the same SQL query. Memcached, on the other hand, allows a quick and easy export of data from any table generated within the framework. This is done in a cache system which is used by default to store certain database queries and user's sessions.

**2) SowerPHP:** This is a specific framework that has been enlarged greatly thanks to LibreDTE. Thus, it is not a generic solution for any kind of project in PHP, but a framework with tailored adaptations for LibreDTE. It mainly provides the minimum functionalities that the community version requires. While some of its features are common in other PHP frameworks, e.g., session management, database, authentication, web services, networking, HTTP requests,

email, etc., its relevance lays on the key extensions defined specifically for LibreDTE. In particular, the so-called *general* and the *app* extensions. The first (**general**) is responsible for the more ordinary functionalities of the website, including PDF creation and format data exporting; while the second (**app**) is in charge of the provision of more specific functionalities associated with the web application, e.g., authentication and web-services. We note that this framework implements a model-view-controller (MVC) pattern.

**3) LibreDTE:** This is indeed the key component of the architecture. It can be divided into 3 major modules, namely, Library, DTE Module, and Web App. We briefly describe these modules as follows:

- (i) Library [7]: This refers to the core PHP-based library (*libredte-lib*) described in prior sections.
- (ii) DTE Module [11]: This is the primary module in the community version. It is responsible for the minimal implementations related to the electronic invoicing process.
- (iii) Web App [12]: This is the container module that allows interaction with the other components. It enables a functional web-based billing system to be provided to end-users.

We also note that this component includes all the necessary functionalities to operate with invoices according to the SII regulatory framework. Moreover, this version can be further customized by software developers, allowing, e.g., additional modules to be incorporated according to the user needs. As an example, a user can be interested in including an accounting module, or a more business-specific module. We exploit this situation to provide further features within the official version of LibreDTE.

### B. LibreDTE - Official Version

This is the commercial (or official) software solution that provides featured support and extended capabilities based on the LibreDTE core. It basically extends the community version by adding an extra layer called the *official extension*. This provides the version with exclusive functionalities beyond the DTE generation, e.g., including the support of different electronic payment systems, integration with popular e-commerce platforms using web-hooks, among others.

Its general architecture is alike to the one presented in Fig. 2, but now with the additional layer on the top of the LibreDTE component. The scope of this official extension is to offer highly automated (commercial) features that satisfy more complex user demands, e.g., the ones related to the handling and issuance of a large volume of DTE of invoices and tickets.

An important enabler of this massive management is the LibreDTE API (Application Programming Interface), which comes in the form of an extra platform that works as an intermediary between the LibreDTE layer and the SII website, e.g., for the purpose of data extraction, mainly through data scraping. This approach automates tedious operations that would otherwise have to be done manually, e.g., accessing purchase and sales registers, or stamping. Thus, reducing

significantly the time spent on the tasks associated with the SII website interaction. In addition, it is worth noting that the official version runs over Amazon Web Services (AWS), and thus makes use of several popular AWS technologies and services, e.g. Amazon EC2, Amazon RDS, ElastiCache, Lambda, among many others. This particular aspect, although relevant, will be a subject of further discussions in future work.

## V. RESULTS & DISCUSSION

This section outlines what we consider are relevant outcomes of LibreDTE within its 6-year period of history.

### A. LibreDTE - Community Version

LibreDTE is nowadays the most important free software project associated with electronic invoicing in Chile, currently having thousands of users. It has been downloaded thousands of times since its first release, counting more than 17,000 downloads only within the *Packagist* composer repository (which is one of the most popular ways of distributing its core). It is also used by numerous commercial DTE providers in Chile, thus having a nationwide presence, indirectly. For instance, if we consider *www.vepsi.cl* only, one of its popular providers, LibreDTE counts thousand of served companies.

### B. LibreDTE - Official Version

The official version of LibreDTE is one of the many commercial software alternatives used to provide DTE generation in Chile. LibreDTE is own by SASCO SpA, which - through his founder - is also the main contributor of the community version. During the first three years of its history, the evolution of this version of LibreDTE was relatively fast, despite the limited resources available. In fact, initially, software development was done exclusively by one programmer, and market positioning was done mostly thanks to *viva voce*. The proper marketing strategies came much later, and when the regulations imposed by the Chilean government already started to be mandatory for all companies. Moreover, the COVID-19 pandemic had an influence too, mainly because the market increased significantly due to the forced digitization that many businesses experienced. Fortunately, these two latter factors were crucial for the growth in sales during the period 2019-2020, when costumers adding to the main DTE services increased by %700. A similar situation is already happening in 2021 (up to October), where the number of new customers that moved to the so-called “*plus*” services is already exceeding (by far) the numbers of 2020.

We believe these results are not only the results of new regulations and the COVID-19 pandemic but also of the tailored and highly automated solutions SASCO is providing through its LibreDTE line of business. A proof of this is the good acceptance LibreDTE API is experiencing, which has been successful not only on the automation of queries but also as an enabler of the new services SASCO is offering, in particular, BHEXpress and ContaFi. The former is a platform for the issuance of electronic tickets for the so-called second category taxpayers, which became recently mandatory

in Chile (since July 2021). The second is a platform for the automatic generation of reports and statistics; both for DTEs and electronic tickets. These new services do not only expand SASCO’s business market but also enrich the Chilean e-business panorama, particularly, satisfying the needs that the Chilean Internal Revenue Service does not provide, e.g. the integration with popular e-commerce platforms.

## VI. SUMMARY & CONCLUSION

E-invoicing has emerged as the natural option for faster, cheaper, accurate and scalable handling of invoices of products, services, taxes, etc., all around the world, without the need for paper-based processes. While this concept has been around for years, its massive adoption is quite recent and often driven by mandatory impositions by governments, e.g. across countries in Latin America and Europe. In this paper, we have addressed the case of Chile, for which recent regulations has been stipulated. More concretely, we have presented LibreDTE, the first (and so far, the only one) free-software-driven project in Chile targeting the provision of free web-based services for automatic DTE generation. LibreDTE is both a community-based free-software framework and a commercial (official) set of software products offering featured functionalities to Chilean companies. In its more than 6 years of history, both versions have been quite successful in the e-business Chilean panorama, becoming at the same time, the most important free-software project for e-invoicing in Chile, and a software-based set of commercial products with rapid (and promising) growth in the last two years.

## ACKNOWLEDGMENT

This paper was partially supported by National Funds through FCT/MCTES (Portuguese Foundation for Science and Technology), within the CISTER Research Unit (UIDP/UIDB/04234/2020) and by FCT through the European Social Fund (ESF) and the Regional Operational Programme (ROP) Norte 2020, under grant 2020.06685.BD.

## REFERENCES

- [1] P. Cedillo, A. García, J. D. Cárdenas, and A. Bermeo, “A systematic literature review of electronic invoicing, platforms and notification systems,” in *2018 International Conference on eDemocracy & eGovernment (IcEdEg)*, pp. 150–157, IEEE, 2018.
- [2] K. Poel, W. Marneffe, and W. Vanlaer, “Assessing the electronic invoicing potential for private sector firms in Belgium,” *The International Journal of Digital Accounting Research*, vol. 16, no. 22, pp. 1–34, 2016.
- [3] M. Tenhunen and E. Penttinen, “Assessing the carbon footprint of paper vs. electronic invoicing,” in *Australasian Conference on Information Systems (ACIS), Brisbane, Australia, 1-3 December, 2010*, pp. paper–95, Australasian Conference on Information Systems (ACIS), 2010.
- [4] B. Hernandez-Ortega, “Key factors for the adoption and subsequent use of e-invoicing,” *Academia. Revista Latinoamericana de Administración*, no. 50, pp. 15–30, 2012.
- [5] H. Elnaggar, “E-invoicing in the middle east: the difficulties and opportunities,” *International Tax Review*, 2021.
- [6] P. Cedillo, A. García, and J. D. Cárdenas-D, “Towards an architecture for electronic notifications services on the cloud: A model driven approach,” in *2019 Sixth International Conference on eDemocracy & eGovernment (ICEDEG)*, pp. 76–83, IEEE, 2019.
- [7] <https://github.com/LibreDTE/libredte-lib>.
- [8] <https://github.com/LibreDTE/>.
- [9] <https://www.cryptosys.net/pki/xmlsig-ChileSII.html>.
- [10] <https://legal.libredte.cl/>.
- [11] <https://github.com/LibreDTE/libredte-modulo-Dte>.
- [12] <https://github.com/LibreDTE/libredte-webapp>.