

INTERNSHIP PROJECT REPORT

Project report submitted in partial fulfillment of the requirement for the degree
of Bachelor of Technology

in

Computer Science and Engineering/Information Technology

By

(Tushar Gupta (181217))

Under the

supervision of

Dr. Jagpreet Sidhu

to



Department of Computer Science & Engineering and Information Technology
**Jaypee University of Information Technology Waknaghat, Solan-173234,
Himachal Pradesh**

Certificate

Candidate's Declaration

I hereby declare that the work presented in this report entitled "SHIPPING SERVICES" in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology in Computer Science and Engineering/Information Technology** submitted in the department of Computer Science & Engineering and Information Technology, Jaypee University of Information Technology Waknaghat is an authentic record of my own work carried out over a period from Feb 2022 to May 2022 under the supervision of **Dr Jagpreet Sidhu** (Assistant Professor(SG) CSE).

The matter embodied in the report has not been submitted for the award of any other degree or diploma.

(Student Signature)

Tushar

Gupta,181217

This is to certify that the above statement made by the candidate is true to the best of my knowledge.

(College Supervisor Signature)

Dr Jagpreet Sidhu

Assistant

Professor(SG)

CSE

Dated:



(Company Supervisor
Signature)

Ashna Arora

Team Lead

Web

Development

Dated:

ACKNOWLEDGMENT

I take this opportunity to express my sincere thanks and deep gratitude to all those people who extended their wholehearted cooperation and have helped me in completing this internship successfully.

First of all, I would like to thank Miss Ashna Arora, who mentored me, guided me and challenged me. I also thank my family and friends who greatly supported Me during the course of the Internship.

Last but not the least, I would like to thank our founders for considering me a part of the organization and providing such a great Platform to learn and enhance my skills.

Tushar Gupta

181217

Table Of Content

Serial No	Title	Page Number
0	Abstract	
1	Introduction <ol style="list-style-type: none"> 1. Introduction of Project 2. Objective 3. Methodology <ol style="list-style-type: none"> a. MERN b. ADA c. Internationalization 4. Organization <ol style="list-style-type: none"> a. Paxcom b. Paymentus 	1-25
2	Literature Survey	26-27
3	System Development <ol style="list-style-type: none"> 1. Components <ol style="list-style-type: none"> a. Main Output Window b. Side Panel Component c. Country Setting Panel 	28-33
4	Performance Analysis <ol style="list-style-type: none"> 1. Analysis of System Developed 	34-39
5	Conclusion	40
6	References	41

List Of Figures

Figure No	Title
1.1	MERN Diagram
1.2	3-Tier Diagram
1.3	Wave Evaluation Tool used in personal sample Project
1.4	Internationalization Applied on Main Project
1.5	Provider Function used in Internationalization
1.6	GERMAN Locale
1.7	FRENCH Locale
1.8	ENGLISH Locale
1.9	Translate Function
1.10	Translate Function Used in JSX
1.11	Company Overview
2.0	Website data
3.1	Data Flow Diagram
4.1	Dashboard
4.2	Side Panel
4.3	Country Setting Panel
4.4	Internationalization Output

4.5	Preferences Overview
4.6	Notifications
4.7	PDF Invoices
4.8	Pay By Text
4.9	My Accounts
4.10	My Plans
4.11	Manage Users
4.12	Pay By Text Enrolled Accounts

ABSTRACT

This project involves moving a payment center to the Paymentus platform. Similar to what we have with the Customer Portal payment center that will need to be adjusted to be able to support business categories, land payments and custom UI required for certain SS data.

The parent system is different from the payment center and website in all matters relating to shipping and use. Users, accounts, companies, programs and invoices created in the SS system.

A different system although it is under the SS domain. Authorized user in the SSO system of the SS at the payment center. The application focuses on payment-related actions and data delivery of the above-mentioned entries. At the highest level the payment center is responsible for managing the organizations of these organizations in the payment system. Any actions such as payments are referred to these organizations.

The goal is to deviate from the company model - The current model raises concerns about merging accounts that may be subject to multiple controllers in a company that provides full access to the accounts / programs and invoices at that Payment Center.

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

This project involves moving the payment center to a Paymentus platform. Similar to what we have with the Customer Portal payment center that will need to be modified to be able to support business categories, land payments and custom UI required for certain SS data.

The following are the key objectives of the high-level project SS looks forward to achieving as it moves into the new field of Electronic Billing, Payment and Presentment (EBPP):

- Electronic Invoice / Bill Introduction Bill
- Customer Payment Options
- Customer payment channels
- Business Intelligence and Reporting

SS.COM - The parent system is separate from the payment center and website in all matters relating to shipping and use. Users, accounts, companies, programs and invoices created in the SS system.

SS Payment Center - A separate system although it is under the SS domain. An authorized user in the SSO system of the SS at the payment center. The app focuses on actions related to payment and data delivery of the above-mentioned entries. At the highest level the payment center is responsible for managing the organizations of these organizations in the context of payment. Any actions such as payments are referred to these organizations.

Company - Today's Payment Center is organized with a "company" perspective where each company must have at least one user setting as a "Administrator" role with an option for any combination of user roles.

There is no guarantee that any accounts can be merged as long as the user based on his or her role can verify invoices to be added to his or her Payment Center company.

The goal is to deviate from the company model - The current model raises concerns about merging accounts that may be subject to multiple controllers at a company that provides full access to the accounts / programs and invoices at that Payment Center.

1.2 OBJECTIVE

These are the main drivers of this startup / partnership with Paymentus

- Improve self-help skills throughout the paid customer journey
- Strive to provide the right customer information with the right solution that suits the diverse needs of our customers.
- Provide greater clarity in the SS payment process

1.3 METHODOLOGY

MERN Stack

What is the MERN Stack, and how does it work?

The MERN Stack is a set of strong and dependable technologies for building important online applications with backend, frontend, and database components. JavaScript is a scripting language that may be used to create stack-based online applications quickly and easily. MERN Stack is a JavaScript solution for creating dynamic apps and web pages that is simple to use.



Fig 1.1 MERN Diagram

The MERN Stack is made up of four primary components or technologies:

MongoDB (Database) is a NoSQL (Non-Structured Query Language) database that is mostly used for document preservation.

The letter E stands for Express, a programming language that is frequently used to create Nodes.

web framework in js

R stands for React, which is mostly used to construct client-side JavaScript frameworks.

The letter N stands for js, which is a commonly used programming language for creating a JavaScript web server.

Each of these four technologies is critical in providing a framework from the developer's perspective. Even in the construction of online applications, these four technologies play a significant role.

Prior to the MERN stack, formerly known as the MEAN stack, the MERN Stack is one of the MEAN stamps, where MEAN also contains four components or means four different technologies, namely, M MongoDB, 'E' Express, 'A' for Angular.js, and 'N' for Node, here if you notice, you can point to MEAN, 'A', that is, Angular.js replaced by 'R,' Small applications, on the other hand, require a long time to build. In contrast, each have distinct features.

We can see that MEAN's end-to-end JavaScript framework is Angular.js, whereas the MERN stack is based on React and its ecosystem. MERN Stack was created to make the upgrade process go as smoothly as possible.

If we examine the diagram carefully, we can see that Node and Express belong to the same application or category. Node.js is the most powerful and popular JavaScript server domain, while Express.js is the web server that comes in second. ME (RVA) N, regardless of the type, is an excellent way to deal with JSON and JavaScript all of the time.

Why should we use the MERN Stack to create mobile and web apps?

MERN Stack based on JavaScript uses all four technologies described above, MERN (MongoDB, Express.js, React.js, and Node.js), making it economical and inexpensive to invest in. The user will be able to obtain better outcomes or results.

SEO-friendly: SEO (Search Engine Optimization) refers to the ability of Google, Yahoo, and other search engines to efficiently and effectively search each page of a website, translate and connect material successfully to the search term, and readily direct you to their website. When MERN technology is used to build a website, it is always SEO friendly.

Better Performance: Better performance refers to a faster reaction time between the backend and frontend of a website, which enhances website speed and results in better performance, allowing users to access content more smoothly.

Enhances Security: Has a major influence on MERN-based application security. The numerous processes, techniques, or technologies used to safeguard web servers and various online applications, such as APIs (App User Interface), from Internet-based threats are referred to as his or her web application's security. In general, MERN stack-based apps may be readily integrated by secure hosting firms. Security technologies such as MongoDB and Node.js are also used to enhance or strengthen security.

Fast delivery: Any Web applications and mobile apps produced with MERN Stack are built quickly, allowing us to provide our clients with speedier service.

Fast Conversion: MERN stack technology provides fast conversion on mobile and online applications at the request of customers.

MERN's four technologies are all open source and free to use. This tool aids developers in resolving issues that may arise when working on open portfolios. As a result, it will benefit the developer in the long run.

MERN is simple and fast to transition between client and server since it is developed in only one language. Switching between client and server is also fairly simple.

What is the structure of a building and how does it work?

MERN features a 3-tier architecture system, which is made up of three layers:

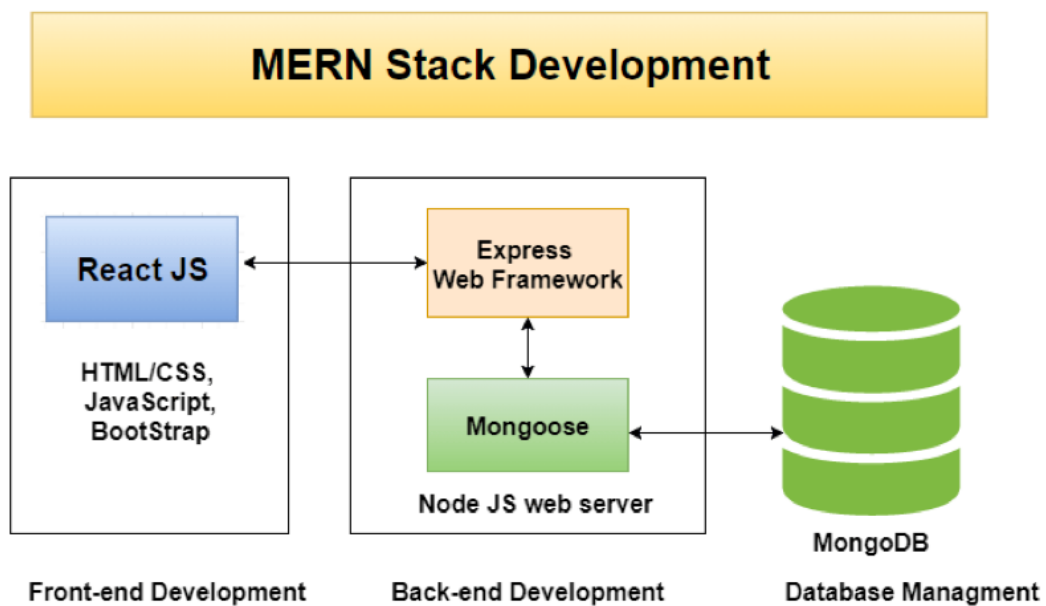


Fig 1.2 3-Tier Diagram

Following are the layers:

- As a first step, use the internet.
- As a middle-class server
- Background with a website

We already know that it is made up of four components: MongoDB, Express.js, React, and Node.js.

Let's learn more about these three groups now:

Web or Pre-section - React.js is in charge of the MERN stack's top level. It's one of the most extensively used JavaScript libraries for exposing the front end of Web applications. It is well-known for producing robust client applications. React allows you to create sophisticated working environments with just one component. It also links data on the backend server to such complex interactions. React is used to construct both mobile and online applications (React Native). React enables for code reuse and can easily support it, which has several advantages and saves a significant amount of work. Allows users to create big web apps with the ability to change page data without having to reload the page.

The server, also known as the middle-tier, is the layer below the top layer and is primarily governed by two components of the MERN stack: Express.js and Node.js. Because Express.js maintains a distinct Server framework that runs within the Node.js server, these two components manage you at the same time. One of the most popular approaches to enhance JavaScript Frameworks is to utilize Express.js. Allows developers to quickly and easily find reliable APIs (Application Programming Interfaces) and web servers. In Node.js HTTP (HyperText Transfer Protocol) objects, add helpful functionalities as well. On the other hand, Node.js plays a critical role in its own right. It's an open source server location that works alongside JavaScript code without requiring the use of a browser. Node.js makes extensive use of JavaScript, allowing computer users to easily create any network service or web or mobile application.

The database's primary function is to hold all of the data associated with your application, such as content, statistics, information, user profiles, and comments. Website as a backend category - It is one of the most important levels of the MERN Stack and is largely managed by MongoDB; the main role of the database is to store all the data related to your application, for example - content, statistics, information, user profiles, comments, and so on. It keeps the majority of the data for security reasons. Maintains a valid record that, in most cases, delivers data to the user when it is required. Data is saved on the website. It creates two or more copies of the data files so that if the system fails, the user may still get the identical information or data they requested. MongoDB is incompatible with a table-based website structure, as indicated by this message. On the other hand, it takes a whole new approach to data collection and storage. MongoDB is a well-known NoSQL (NoSQL or Non Structured Query Language) website, which is a text-based database. The phrase 'NoSQL' generally refers to a separate website that doesn't require a defined schema or related interaction tables

to hold the data it needs. MongoDB saves data in a unique structure that does not include similar tables, rows, or columns.

The MERN Stack's Most Important Components

Let's take a closer look at each of the four technologies or components that make up the MERN Stack:

I. MongoDB -

MongoDB is a well-known NoSQL (NoSQL or Non Structured Query Language) website, which is a text-based database.

The phrase 'NoSQL' generally refers to a separate website that doesn't require a defined schema or related interaction tables to hold the data it needs. MongoDB saves data in a unique structure that does not include similar tables, rows, or columns.

It indicates MongoDB isn't compatible with a table-based website structure. On the other hand, it takes a whole new approach to data collection and storage.

The data storage format is known as BSON, which stands for Binary JavaScript Object Notation; its binary structure includes codes for length and kind of information, allowing it to be rapidly identified.

When storing documents in clusters, MongoDB employs BSON.

Allows for the most adaptable and adaptable document format possible.

Because of its improved maintenance and identification procedures, it is significantly quicker than RDBMS.

Sophisticated joining methods aren't accessible in MongoDB, hence it can't handle complex transactions.

One of the key advantages of MongoDB is that it uses JavaScript to encrypt language.

There is no Schema for data that is kept in a separate document.

There is no idea of relationship or table structure in MongoDB, like there is in RDBMS (Related Data Management System), where tables have a specific relationship.

It also has a customizable document architecture that allows any engineer to write documents much more quickly.

One of the most popular forms of NoSQL databases is MongoDB. When we realise that a website operating on a single server will approach its measuring limit, it scales and gives exceptional performance.

MongoDB is a NoSQL database that expands by adding more servers and boosts productivity thanks to its flexible document architecture.

Other important features of MongoDB -

Schema-less MongoDB's ability to accommodate several types of scripts in the same cluster is one of its best features. Because of this unique capabilities, MongoDB provides a great deal of information flexibility. On the MongoDB website, a single collection can include several texts, each with its own numerical values, fields, and other information. One document does not need to relate to other documents in the same way that a relationship site does.

Due to this identifying function on the MongoDB website, it is simple to retrieve the relevant data from the database. Every data item in MongoDB has its own index, which is separated into primary and secondary indexes. Users will have an easier time obtaining data with this identifier since it saves them time. When data isn't listed, the site examines each page for a specific query, which is inefficient and time-consuming.

Document-Oriented: MongoDB stores all data in documents rather than tables like SQL. These papers also have their own object identification. When compared to RDBMS, data containing information is kept in fields, i.e. Instead of columns and rows, the data is organized into keyword pairs, making it more versatile and easier to download.

These sorts of data segments are dispersed uniformly among all shards on all portable servers. Due to the inclusion of good properties like identification, rating, duplication, and others, MongoDB has very high speed and data permanence when compared to other databases.

Multiplication and Availability - MongoDB improves data availability by duplicating data across numerous servers. In the end, providing data reconditioning or duplication safeguards the site from any hardware failure and prevents data loss in the future. All of this, I believe,

would be due to data unavailability if one server was inactive or in conflict owing to a mistake, and the data could not be quickly accessed from other live, immediate servers.

This MongoDB feature works in the same way as the SQL GROUPBY section. This GROUP BY clause performs a number of operations on the data it collects in order to produce distinct or computed variables.

Simple Local Setup:

- MongoDB's native settings for local setup are pretty simple. MongoDB may be quickly and easily implemented into a system.

II. Express

Express is a server-side JavaScript component that runs in js.

It is one of the most effective tools for creating JavaScript Frameworks.

It gives the developer a framework for building and managing reliable servers.

Express is a programming language that can be used to create and develop online and mobile apps rapidly and effortlessly.

As a result, Express is frequently used to offer server side logic for mobile and online apps.

Allows developers to quickly and easily find reliable APIs (Application Programming Interfaces) and web servers.

Express makes it simple to set up robust web servers for your application using a router and middleware.

Also add useful functions to Node.js HTTP (HyperText Transfer Protocol) objects.

It's a crucial component of MERN and MEAN Stack, and it's used to create faster, more secure, and more efficient web applications.

Other noteworthy aspects of Express include:

Express simplifies and accelerates the creation of Node.js online and mobile apps.

Express is placed in an extremely natural location. Express may be readily installed and configured on a computer with little effort.

Express makes it simple to connect to databases such as MongoDB.

Express allows you to specify your application's pathways using HTTP methods and URLs. The primary goal of the route is to describe the code that must be executed in response to each request that the server receives.

The most common way for routing is to use a sequence of URL patterns and the HTTP method associated with the application.

You may easily utilize the numerous intermediate modules provided at Express to do extra duties and tasks in each request and response.

A request is a message sent to the server asking for something, and a response is a message delivered by the server to the client containing the outcome of whatever the client asked for.

If you make a mistake and want to fix it quickly, you may use debugging middleware to accomplish it.

Middleware is used locally during the life cycle of a request or response in the form of code.

It's widely used to improve the performance or behavior of a web server.

REST APIs may also be built with Express (Representational State Transfer Application Programming Interface)

REST APIs, also known as RESTful APIs, conform to the REST architectural style and may be used to connect to RESTful web services. The versatility of the REST API is its main benefit; it obtains and processes data via HTTP queries.

The data flow in a website design may be simply done utilizing Express's two template engines, EJS and Jade.

Engineers may leverage Express's huge third-party add-on suite to give greater performance, aid raise safety, and improve speed.

It works well and is readily measured; it can be found almost anywhere and used in several programmes at the same time.

It has one wire and is Asynchronous.

It also has a large Node.js community.

III. React

MERN stack

React is one of the most widely used open source JavaScript libraries for web development.

Before utilizing react, there are several prerequisites that must be met, such as installing the newest Node packages on your machine. You should also be familiar with HTML, CSS, and JavaScript.

It's used to create user links, particularly in a one-page web interface.

There isn't a JavaScript framework here. It's basically a JavaScript library designed by Facebook to handle difficulties that we could have solved earlier with other libraries while developing online and mobile applications.

In mobile and online apps, React is also used to capture the viewing layer.

Allows us to design reusable UI (User Interface) components.

It was created by SE Jordan Walke, who worked at Facebook at the time.

React was first posted on the Facebook news server.

Allows developers to design big web apps with the ability to change page data without having to reload the page.

The main purpose of the response is that it only applies to user communication in the app, whether mobile or web.

Very fast, easy and measurable.

Other JavaScript libraries or frames are also used in conjunction with React.

There are numerous open source forums that make complex online and mobile apps simpler, such as Angular js in MVC, but React is the MEAN stack's replacement for Angular. Many developers now prefer the MERN stack to respond to a react since it is comparatively faster and is having more benefits than its predecessors.

Other important React features -

MERN stack

Simple to learn and use - One of the most appealing features of react is how simple it is for beginners to learn and use this advanced framework to build web and mobile applications.

When it is compared to Angular, anyone with a fundamental understanding of system may fastly understand React. Because Angular is known as a 'Direct Domain Language,' it is

assumed that it is tough to grasp. We'll need a brief understanding of CSS and HTML to get started with React JS.

React is one of the most basic JavaScript frameworks for open source online and web application development. It employs a phase-based strategy, plain and simple JavaScript, and a well-defined life cycle, making the response more easier and more straightforward. For the purpose of simple reading and the development of industry grade mobile and web apps.

It employs a basic syntax known as JSX, which allows readers or developers to mix HTML with JavaScript in order to create more effective online and mobile apps. However, you don't have to use JSX; you can just use ordinary JavaScript instead. However, Because of its simplicity and grammar, JSX is a significantly better choice.

Data Bonding - React uses the Flux application architecture to govern the flow of data into segments through a single control point known as a dispatcher. It employs a single data binding mechanism for self-contained components of huge React apps, which is simple to correct.

React is used to construct mobile applications (React Native) and online applications in the traditional way. React enables you to reuse code and just support it, which has a number of benefits and saves you time.. As a result, we may develop iOS, web apps, and Android applications at the same time.

Due to the stability of the data, React works quite quickly. We may forecast that stable data structures do not change, allowing you to compare direct object indicators instead of deep tree comparisons, as the name implies. The aforementioned cause has an effect on the reaction's performance and makes it speedier.

Testability - React is extremely simple to test; any apps we create in response, whether mobile or online applications, will be significantly easier to test when shared. In the respond, there are some global functions that are deemed regional functions by distinct reaction concepts, and we may easily be misled by the circumstance we communicate in the response perspective. We may also look at output and active activities, tasks, and events, among other things.

IV. Node.js

MERN stack

js is an open source server location that allows you to use JavaScript code without having to use a browser.

js is neither a planned nor a draught language.

Many background apps, such as network applications and mobile applications, are frequently built and upgraded using it.

Uber, PayPal, Netflix, and other major corporations utilize it mostly in production.

It may be used as a free ASCII text file archive by anyone.

It will run on a variety of operating systems, including Windows, Mac OS X, Linux, Unix, and others.

It's really simple to get started with and may be used for fast prototyping and development.

Provides consumers with extremely quick and extremely rapid services.

It's incredibly compatible and may be used to clean ASCII text files.

Runs JavaScript in the background, allowing the computer user to swiftly create any network service or web or mobile application.

This application is a complete solution for any ASCII text file library.

It may have a non-blocking or asynchronous environment.

.

Other important React features -

Easy Scalability: Because it employs a single series event architecture, js is incredibly scalable. Because of the event process, the server normally replies in a non-blocking manner. It also allows the server to scale far larger than conventional servers, which only produce a certain number of threads to process applications. Node.js is based on a single-series application that may deliver services to numerous apps.

Acceleration: Because Node.js' event loop handles all associated processes, it functions as a speedy suite, and all tasks, Network communication, reading and writing on a webpage, and programming a file are among the tasks that are accomplished quickly. It is based on Google's V8 engine.

Because js employs JavaScript to run code for web-based projects and other online and mobile applications, it is simple to read and correct. You can simply construct and execute the

software on Node.js and explore as much as you like if you are successful in advanced programming and have a decent JavaScript command; it relies on your strength.

Real-time web apps: JavaScript is essential in real-time web apps.. You may also utilize PHP to create a mobile phone or web application, albeit it will take the same amount of time as Node.js. However, due to its quick synchronization, Node.js is the greatest solution for building game apps and chat systems.

Advantage of Caching: js enables temporary storage for one module. You don't always need to re-use the same lines of code because they're already saved in the Node.js archive.

Data Distribution: In Node.js, hypertext transfer protocol (HTTP) requests and response units were considered as 2 separate events. They are streams of information, so once you have edited the file during upload, it will slow down and create as soon as the information is provided in a transfer style. You may also stream music and video files at breakneck speeds.

The biggest criticism of Node.js was its JavaScript, which frequently had many spaghetti process scripts. Coffee Script and Typescript were created to overcome these issues, but they became a hurdle for others who were more concerned with code standards. With the regular release and adoption of ES6, classes are incorporated into the framework, and the code appears identical to C#, Java, and SWIFT.

Event-Driven and Asynchronization- All Apis of Node.js asynchronous library site, i.e., non-blocking. It suggests that Node.js-based servers should not wait for a related API to return information. When the server is live, it moves on to the next API, and the Node.js Events notification system aids the server in getting feedback on earlier API selections.

Business Support: Node.js is used to develop apps by several well-known firms such as PayPal, Wal-Mart, Microsoft, and Google. Because Node.js is based on JavaScript, most businesses merge their front and back offices.

MERN Stack's Advantages

There are many benefits to MERN Stack, some of which are listed below -

It includes MVC (Model View Controller) elements for creating a smooth online or mobile application; the main goal of this structure is to separate the display's subtleties from the business logic.

From advanced development to backend development, it covers every facet of web programming using JavaScript.

It is a community-supported open source framework that is frequently used to construct web-based or mobile apps.

In comparison to MEAN Stack, it is extremely quick and efficient, and it is best suited for tiny applications, whereas MEAN Stack is best suited for large applications.

ADA COMPLIANCE

The Americans with Impairments Act (ADA) Standards for Accessible Design mandate that all electronic and information technologies (such as websites) be accessible to individuals with disabilities. This is distinct from 508 compliance.

The ADA was created to safeguard people with disabilities from being discriminated against. This guarantees that companies, employers, and others are accountable to a set of standards in order to avoid disability discrimination and ensure that settings are accessible.

The legislation is divided into numerous titles, each of which prohibits and defines discrimination in different ways. Employment, state and municipal governments, public transit, public places, and public telecommunications are just a few examples.

All of the WCAG's principles are listed in an easy-to-follow checklist. There are three levels to this checklist:

Level A: Create a website that can be accessed by a limited number of people.

Level AA: Create a website that practically everyone can utilize.

Level AAA: Create a website that is accessible to all people.

General ADA Rules:

- Add alt tag to images as alt text can be read by screen readers, and helps visually impaired people to understand the purpose of the images.

- Anchor tags should have a role either set to link or button as per their functionality. At certain times, we use anchor tags (`<a />`) as buttons also, so it is advisable to define their roles.
- Color contrast ratio should fulfill ADA guidelines - <https://www.w3.org/WAI/WCAG21/Understanding/contrast-minimum.html>
- Some users prefer to use applications in High Contrast Mode and colors in this case too need to be taken care of.
- Tables should have a proper caption that announces their purpose. If the table header is visible, then that serves the purpose provided it is read by the screen-reader otherwise you can add a screen-reader label for the same.
- Column headers, if sortable, should have aria-sort property set according to the sorted state.
- On opening modal windows, make sure focus always stays inside the control. On pressing Tab or Tab+Shift keys, focus should not move outside the modal window.
- Modal windows should announce the proper caption or purpose of the dialog on opening so the user is aware which window is opened.
- Close button in the modal window should have a descriptive label. Eg. Close payment dialog button.
- Radio buttons should have a proper grouping label that is announced when focus moves to that particular group.
- Non-visual keyboard users will be confused with the purpose of the elements if a proper leading question or group heading is not announced. Non-visual users will not understand how these elements relate to each other. With proper grouping, the user can quickly understand their relationship to each other and understand their purpose.
- Inputs should be associated with proper labels using `for(htmlFor in React)` and `id` properties - <https://www.w3.org/WAI/tutorials/forms/labels/>
- Proper hierarchy levels should be followed by headings.
- <https://www.w3.org/WAI/tutorials/page-structure/headings/>
- <http://pauljadam.com/guides/headings.html>
- If any error occurs in the application, then it should be announced by the reader. Set role as alert to let the user know that some error has occurred.

- Same goes for success status, for example if a form is submitted successfully, then the user should be notified about it.
- Make sure focusable elements have tabindex set as 0. Buttons and links have tabindex set by default so they don't need to be defined explicitly.
- For elements like drop-down or input, if some value is required, then add "required" attribute to those inputs so the screen reader would announce it as required to non-visual users.
- Buttons should be descriptive enough for non-visual readers to understand its purpose. Suppose if the button name is "change" or "edit" and for display purposes, it is not possible to add more content, then add an aria label to them which describes their purpose. For eg. "Change the settings"
- Any help popovers should announce the content when they get the focus so the user understands the requirement or knows the purpose of the help popover.
- If links open in a new tab, then add a title or aria-label indicating some informative message like “opens in new tab”.

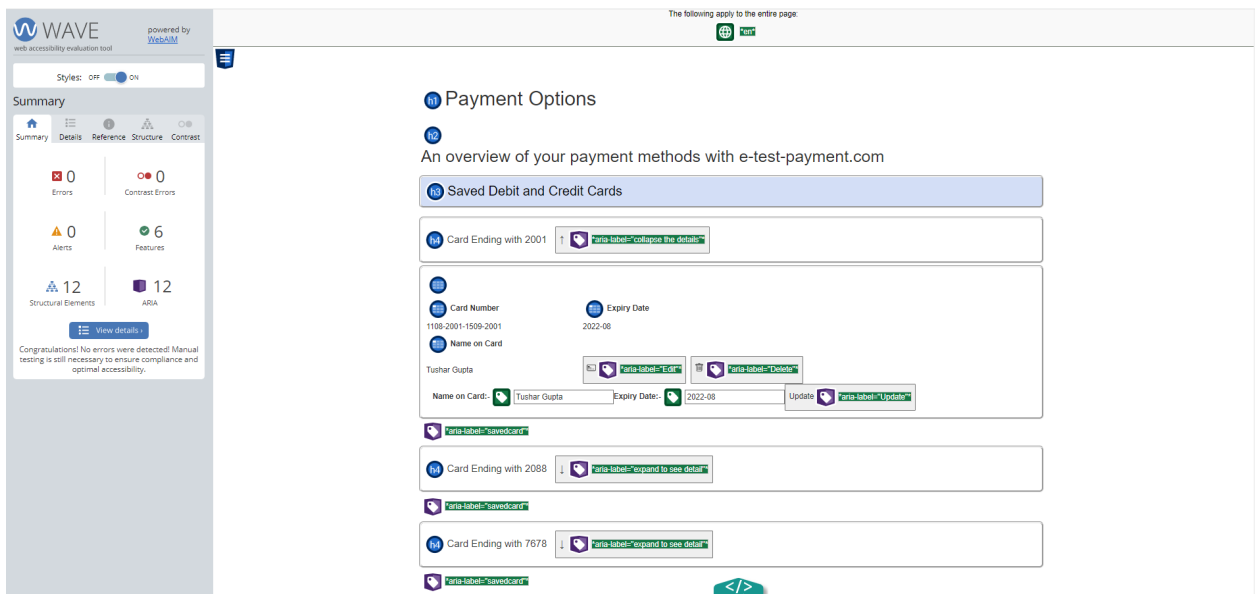


Fig 1.3 Wave Evaluation Tool used in personal sample Project

INTERNATIONALIZATION

intl react

Yahoo's React Intl is an open source project that's also a component of Format.js, a global JavaScript library based on the Intl Javascript Intl API.

The Internationalization API in JavaScript

The Internationalization API is rendered in JavaScript with the Intl object as the default built-in object.

This API is used and built upon by React Intl. React Intl will continue to perform its magic as long as the browser supports these APIs.

Note that Safari is the only browser that does not support these APIs. In the example project below, we'll utilise polyfill to solve this problem.

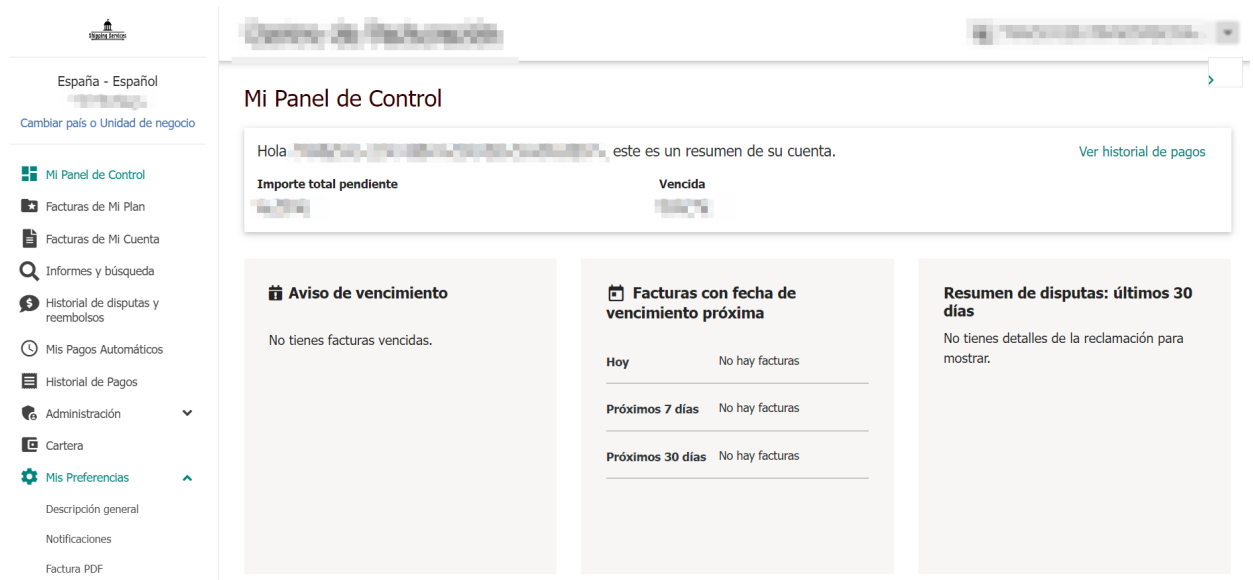


Fig 1.4 Internationalization Applied on Main Project

Many modules

React Intl's package includes ES6, CommonJS, and UMD modules. As a result, bulk sites like Webpack, Browserify, and Rollup benefit greatly from it.

Data about the location is being loaded.

This location data is used by React Intl to support the quantity and formatting of the associated time. For each given place, location data describes the following:

```

sampleapp1 > src > i18nProvider > JS Provider.js > children
1  import React, { Fragment } from 'react';
2  import PropTypes from 'prop-types';
3  import { IntlProvider } from 'react-intl';
4
5  import { LOCALES } from './constants';
6  import messages from './messages';
7  debugger;
8  const Provider = ({ children, locale }) => (
9    <IntlProvider
10     textComponent={Fragment}
11     locale={locale}
12     messages={messages[locale]}
13   >
14     {children}
15   </IntlProvider>
16 );
17
18 Provider.displayName = 'I18nProvider';
19
20 Provider.propTypes = {
21   children: PropTypes.oneOfType([
22     PropTypes.arrayOf(PropTypes.node),
23     PropTypes.node,
24   ]).isRequired,
25   locale: PropTypes.oneOf(Object.values(LOCALES)),
26 };
27
28 Provider.defaultProps = {
29   locale: LOCALES.ENGLISH,
30 };
31
32 export default Provider;
33

```

Fig 1.5 Provider Function used in Internationalization

Patterns related to formatting and division, times, time zones, numbers and currency values

```

import { LOCALES } from '../constants';

export default {
  [LOCALES.GERMAN]: {
    'edit-file': 'Bearbeiten Sie (file) und speichern Sie es, um es neu zu laden.',
    'hello': 'Hallo',
    'Payment Options': 'Zahlungsmöglichkeiten',
    'An overview of your payment methods with e-test-payment.com': 'Eine Übersicht Ihrer Zahlungsmethoden mit e-test-payment.com',
    'Saved Debit and Credit Cards': 'Gespeicherte Debit- und Kreditkarten',
    'Add New Card': 'Neue Karte hinzufügen',
    'Show Previously Added Cards': 'Zuvor hinzugefügte Karten anzeigen',
    'Name on Card': 'Name auf der Karte',
    'Card No ': 'Karten-Nr',
    'Expiry Date': 'Verfallsdatum',
    'Add New': 'Neue hinzufügen',
    'Card Ending with': 'Karte endet mit',
    'Update': 'Aktualisieren',
    'Card Number': 'Kartennummer'
  },
},
};

```

Fig 1.6 GERMAN Locale

```

import { LOCALES } from '../constants';

export default {
  [LOCALES.FRENCH]: {
    'edit-file': 'Modifiez {fichier} et enregistrez pour recharger.',
    'hello': 'Bonjour',
    'Payment Options': 'Options de paiement',
    'An overview of your payment methods with e-test-payment.com': 'Un aperçu de vos moyens de paiement avec e-test-paiement.com',
    'Saved Debit and Credit Cards': 'Cartes de débit et de crédit enregistrées',
    'Add New Card': 'Ajouter une nouvelle carte',
    'Show Previously Added Cards': 'Afficher les cartes ajoutées précédemment',
    'Name on Card': 'Nom sur la carte',
    'Card No ': 'Carte No ',
    'Expiry Date': 'Date d expiration',
    'Add New': 'Ajouter nouveau',
    'Card Ending with': 'Carte se terminant par',
    'Update': 'Mettre à jour',
    'Card Number': 'Numéro de carte'
  },
};

```

Fig 1.7 FRENCH Locale

```

import { LOCALES } from '../constants';

export default {
  [LOCALES.ENGLISH]: {
    'edit-file': 'Edit {file} and save to reload.',
    'hello': 'Hello',
    'demo-placeholder': 'Placeholder sample',
    'Payment Options': 'Payment Options',
    'An overview of your payment methods with e-test-payment.com': 'An overview of your payment methods with e-test-payment.com',
    'Saved Debit and Credit Cards': 'Saved Debit and Credit Cards',
    'Add New Card ': 'Add New Card ',
    'Show Previously Added Cards': 'Show Previously Added Cards',
    'Name on Card': 'Name on Card',
    'Card No ': 'Card No ',
    'Expiry Date': 'Expiry Date',
    'Add New': 'Add New',
    'Card Ending with': 'Card Ending with',
    'Update': 'Update',
    'Card Number': 'Card Number'
  },
};

```

Fig 1.8 ENGLISH Locale

Interpretation of names of currencies, times, months, midweek, etc.

Language knowledge and script knowledge (multiple cases, characters used, gender list, uppercase, typography, etc.)

Country information (currency, calendar preferences, weekly settings, phone codes, etc.)

```

import React from 'react';
import { FormattedMessage } from 'react-intl';

const translate = (id, value = {}) => <FormattedMessage id={id} values={{ ...value }} />;

export default translate;

```

Fig 1.9 Translate Function

```
<h1>{translate('Payment Options')}</h1>

<h2><p>{translate('An overview of your payment methods with e-test-payment.com')}</p></h2>
<div class="cards2"><h3 class="saved">{translate('Saved Debit and Credit Cards')}</h3></div>
</div>
```

Fig 1.10 Translate Function Used in JSX

Formatting Data using React Components vs. API

The library provides two ways to format the character unit, numbers, and dates: React components or API.

Response feature:

API:

I take the first step where possible, using idiomatic-React components to format data with a critical API.

The advantage of this method is that it allows a) integration of components with other components, b) allows rich text and characterization of character units, c) provides prop type warnings for formatting options, and d) appropriate ComponentUpdate tools to avoid expensive formatting tasks. .

Of course, there are times when your only option is to use the API (Example: thread transfer as prop, HTML feature name, etc.), so it still works fine.

React Intl Library allows React internal installation directly, with non-shelf components and an API that can handle everything from a series of formats, dates, and numbers, to quantity.

Engineers call this practice an international practice (often abbreviated as “18n,” because there are 18 characters between ‘I’ and ‘n’ in the word Internationalization.)

1.4 ORGANIZATION

DESCRIPTION OF INDUSTRY

PAXCOM

PAXCOM provides cloud-based multichannel process management services for Commerce (hereafter referred to as “Services”).

We are a team of 400+ Ecommerce enthusiasts, who love to use our technology and knowledge to facilitate Digital Commerce of products in all global markets.

Some of the world's leading brands like PepsiCo, Mondelez, Britannia, Lenovo, Wipro, Abbott trust our data, statistics, consultation and marketing solutions and implement their ecommerce-driven marketing strategy.

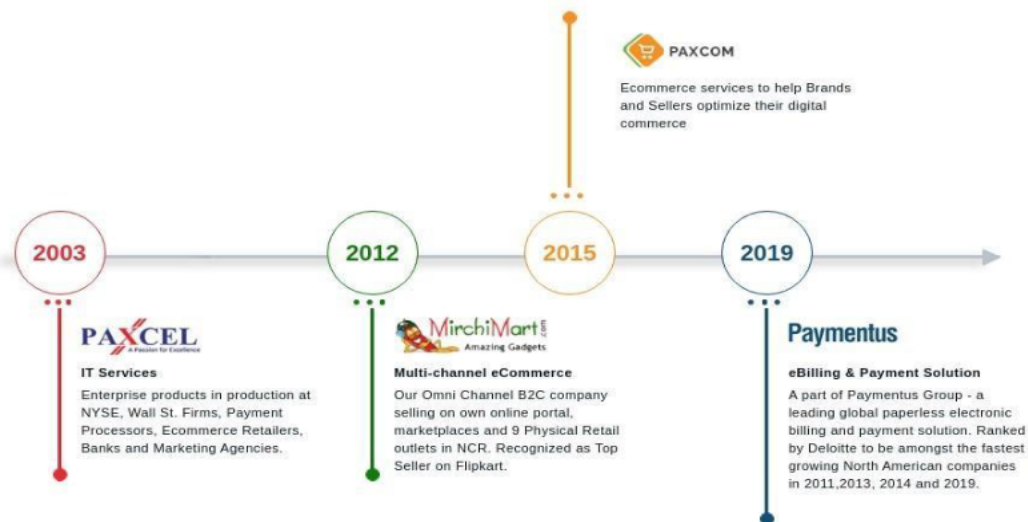


Fig 1.11 Company Overview

Our Principles

1. Honesty

Nothing is more important to Paxcomer than honesty and integrity in our work, and in the people we work with. We value authenticity and are committed to delivering real results.

2. Customer attraction

Our customers have made a significant contribution to what we are today. We work with our customers as partners who care about their success, and we work hard to meet their needs and gain their trust by making them happy.

3. Long-distance travel

When it comes to innovating, discovering new opportunities, simplifying things, and continuing to learn and improve, we think a lot! We have never learned to ignore, either to our customers or to ourselves.

4. Speed

Paxcomers believe in anticipation, planning, preparation, and action rather than simply responding to circumstances.

5. Collaboration

We all work together to achieve the same goals as a large family. We respect and trust each other, and we recognize the importance of the individual's role in our success.

6. Build a Paxcom logo

Every Paxcomer is a product representative and Paxcom agent. In all that we do, each one of us integrates and reflects the values we stand for, and we strive to advance and strengthen the brand.

7. Have fun

We believe the work can be fun and rewarding. As a result, we strive to create a work environment where Paxcomers can be fun and productive at the same time.

PAYMENTUS

Paxcom is part of the Paymentus team - the world's leading electronic payment and payment solutions.

Paymentus (NYSE: PAY) is a leading provider of cloud-based payment technology solutions and solutions. We present our next-generation product list using a state-of-the-art technology stack for more than 1,300 bill payers across North America. Our omni-channel platform provides consumers with an easy-to-use, flexible and secure electronic payment billing experience with their favorite payment channel and brand.

Paymentus's proprietary Instant Payment Network [™], or IPN, expands our reach by connecting IPN partner forums with tens of thousands of debt collectors in our integrated payment, billing, and reconciliation capabilities.

Paymentus offers taxpayers' costs for all sizes in a different vertical industry, including services, financial services, insurance, government, communications and health care.

In 2004, Paymentus was born out of a desire to improve the way debt is repaid. Vision, innovation and exemplary service have propelled Paymentus to be the leading electronic payment solution and payment solution in the market, resulting in 1,700 customers, including some of North America's largest taxpayers.

We know that in order to keep our solutions current and relevant, we need people with knowledge, driving and the ability to promote the most exciting customer experience. Our highly dedicated, intelligent staff have transformed the vision into a secure Customer Partnership, established by SAAS and Payment Forum; one that enables direct billing organizations to provide integrated customer information and improve the acceptance of electronic payment and payment services services.

Recognized by Deloitte as one of North America's fastest-growing companies for 4 years, Paymentus consistently strives to develop better, faster, more secure, less expensive payment and payment platforms. We are always looking for the highest number of our customers, in both solutions and service.

That is what has led to our amazing growth over the past decade. We succeed when our clients are successful. They thrive when their customer relationships are improved and,

subsequently, their customers participate in these cost-effective electrical services at higher prices.

PAYMENTUS DIFFERENCES: A PAYMENT TECH THAT INCREASES YOUR COMPLIANCE, EFFECTIVENESS, KEY AND CUSTOMER CUSTOMER SKILLS

1. MOST FUNNY

We believe in configuration, not customization. Our enhanced, law-based engine solves your business needs and payment rules without customization.

2. ACKNOWLEDGMENTS

Our cloud-based platform is accessible through a host of APIs, iFrames and fully customized solutions that offer 360 degrees of control over your user experience.

3. NEXT

As a compliant provider of PCI Level I, we help businesses reduce and eliminate the burden of PCI.

4. SAFE AND RELIABLE

Our system is designed for 100% uptime and high security. Multi-layer detection and blocking system, multi-level authentication, encryption and token processing ensures the reliability and security of the transaction.

5. VERY CONNECTED

We cover more than 350 key programs, including the CIS, accounting and ERP systems, which facilitate seamless payments, reporting and reconciliation to key financial systems and operations.

6. BUILT FOR THE FUTURE

It is built from the ground up. On the basis of a single code and no version, we are able to innovate constantly and stay ahead of ever-changing needs.

CHAPTER 2

LITERATURE SURVEY

In today's fast-paced business environment, it's critical to be able to anticipate and efficiently comprehend the wants of consumers. If our clients could access our business online and have our products or services in their hands, it might have a big influence on their everyday lives, creating an environment for conducting business online and serving customers on a broad scale. For the second time, buying or doing business online has become a lifeline for the younger generation. Shipping web application that provides a variety of items and allows users to send to any location in the globe using our website. This project enables the delivery of a wide range of items, whether bought online for personal or business use. This initiative provides a means for company owners to get accessibility. Several technologies are taught and acknowledged when you meditate to construct a shipping web application. React.js, MongoDB, Node.js, and Express.js are among the technologies available. This is a project to make various goods more accessible and to create a web application in which the client receives a fully functional web application as well as an understanding of the technology used to create such an application.

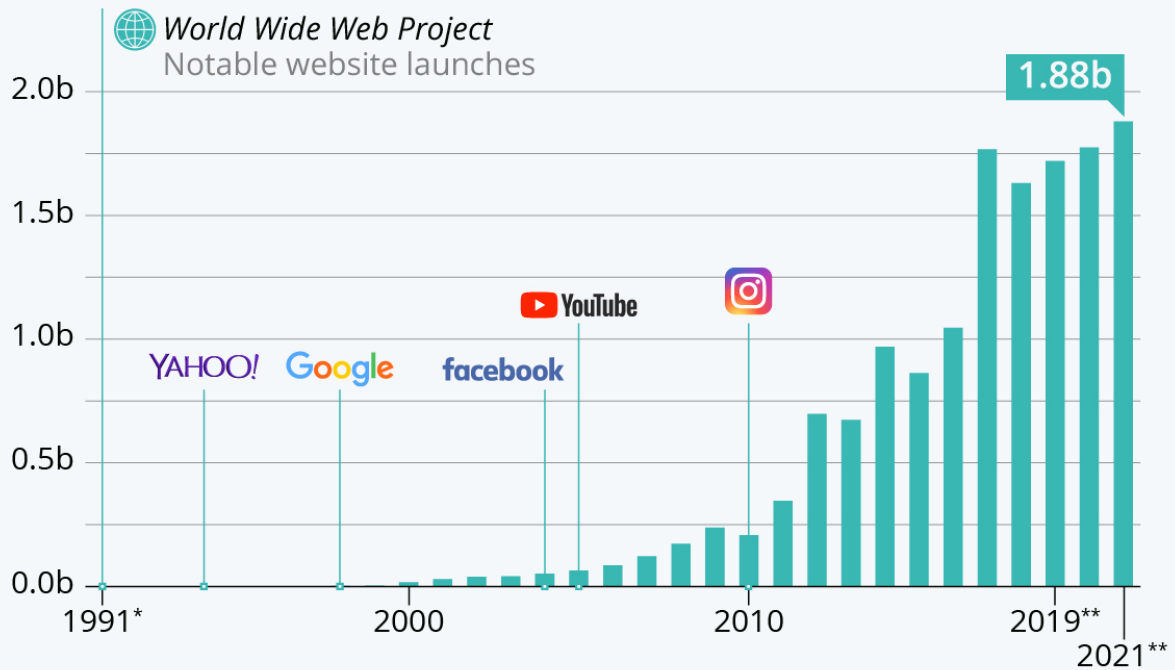
Reference Terms — ReactJS, NodeJS, MongoDB, ExpressJS

Web design relates to the user interface of a web page and is a global language in the visible world. The design's major goal is to place the content front and centre so that consumers may simply access and utilise web material. Web design has evolved significantly as a result of various technological advancements and styles, moving from a first-generation web page that displayed content using a simple text webpage to a second-generation web page with lots of images and bright colours to create memorable web pages, and finally to modern web pages. Web design that is simple and intuitive. The design of a web page should constantly be current, and the information should be updated as well.

Create a web application for the MERN stack you're using. I created a web application with Mongoose and MongoDB data using the MERN stack. Simulator tools and Chrome developer tools were utilised throughout the test.

How Many Websites Are There?

Number of websites online from 1991 to 2021



statista

Fig 2.0 Website Data

CHAPTER 3

SYSTEM DEVELOPMENT

3.1 Components

3.1.1 The main output window

The first component in the system development part is the parent component which contains all child components that are sub parts of the final product that is expected.

The parent component contains a header component that contains the name of the company/client and user id with which the user has logged in .

The parent component contains a dashboard for the application that contains the Welcome message for the user logged in with its user id displaying over the screen.

The dashboard for the user contains the total amount outstanding and past due amount as a first sub component with view payment history and Pay invoices method.

In the Payment history method the data for payment history for the logged user is fetched from the Database using MongoDB and Mongoose in which data for payment is saved for previous payments.

The Pay Invoices button triggers an onClick event that fetches the amount from the database that is needed to be paid by the user.

The second sub component of dashboard component contains 3 tiles as sub comp i.e,

1. Past Due Notice
2. Invoices Due Soon
3. Dispute Summary- Last 30 days

1.Past Due Notice:- This tile contains the invoices that are due after the notice time with the amount displayed. This tile also contains pay now that clears the due amount that is fetched from the user account .

2. Invoices Due Soon:- This tile contains the invoices that will get due soon with a range of days specified.

3.Dispute Summary:- This tile contains the invoices that have some dispute in clearance

Parent component also contains a Side panel that contains the logo of the company as the first part of the side panel.

Second sub comp in side panel contains an option to change locale and type of merchant from which state can get updated and rendered comp will show the data that is accessible to that particular user based on the type of merchant and language used.

3.1.2 Side panel Components

3.1.2.1 Administration

Roles govern the actions that the user may perform.

Existing 6 roles are there:

- . View Only : view invoices of their assigned accounts and update their email preferences.
- . View and Adjust:View invoices and submit disputes
- . View and Pay:View invoices and make payments
- . View, Pay, and Adjust: view invoices, set up payments and submit disputes
- . Plan Manager: In addition to View, Pay and Adjust access, ability to assign their plan's accounts to new or existing users. Plan Managers may provide View Only through View, Pay and Adjust access
- . Administrator: his user has full access to all accounts, plans, and functions of the SS application including the ability to add / remove users, manage access assignments, and cancel or re-enroll accounts

The first user by default is assigned an administrator role. The user can be an admin for one account but may not be administrator for another account.

This component further consists of following sub-components:

- I. **My Accounts:** A user may add accounts (that could be associated with a different BUs)

An account has two key entities BU and country that determine which application features and invoice types the user may have access to.

The process associating an account or a plan with a user is called enrollment. Enrollment involves account invoice authentication to ensure proper authorization for association.

An account can be associated with multiple users.

An account can't be a part of more than one country and BU.

- II. **My Plans :** Plan is an entity that has multiple accounts associated with it. Plans allow for rolled up views and actions on a group of accounts. An account can be associated with multiple plans. The association of plan and accounts happens in ss.com. Adding / removing accounts to a plan happens outside of the application.

Enrollment process in the application associates users with plans.

A payment against an invoice may be made at an account level or at a plan level depending on the setting 'Account Pay' and ' Plan Pay' respectively.

- III. **Manage Users:** A user logged in as a customer should be able to view and manage user's

Access

Roles

Add/Delete users

The admin should be able to assign roles to another user.

Add User

An admin should be able to add another user to the application. The user must have a ss.com email id, to be added to the application. The user may be provided one of the roles. More than one user can have admin roles.

Manage Access

An Admin should be able to allow users access to accounts. The user's role will govern the actions allowed.

Delete User

An admin should be able to delete a user from the application. Removal of the user should result in a constellation of autopay.

View/Add/Modify account assignment

An admin should be able to view the account assigned to the user.

Admins should be able to add accounts assigned

Admin should be able to unassign an account.

Unassignment should delete all automatic payments associated with the user.

IV. Pay By Text : Payment reminder text will be valid up to the invoice due date for a given account and will not be allowed to make the payment via text after the due date. If a customer initiates payment via text after invoice due date, an error text message will be sent to the customer. However, customers will be able to pay the invoice in the application even after the invoice due date.

System must check for full payment received in the application if the user submits the payment by text for a given account. If full payment is received, then customers will not be allowed to make the payment via text for that given account and pay by text reference number. In the event if the customer initiates payment via text, then an error text message will be sent.

If the payment fails or declines while processing the payment via text, then an error message will be sent. and no retry option will be provided.

On successful processing of payment via text, a confirmation text message will be sent and the message will include account Number, payment reference number and invoice amount.

An account can have multiple invoices and each of these invoices can have different due dates, hence the user can receive multiple payment reminder text messages for the

same account number based on the applicable invoice due date with a unique pay by text reference number. In this illustration below, the user will receive 2 payment text messages.

3.1.2.2 Preferences

This component consists of details of user (userName and userId) and following three features:

1. Notifications (choose when to be notified about your billing activity)

In this component, the user can select preferred language for email and enter required details to notify himself like the number of days before he wants to get notified.

He can also selected whether he wants to notified when:

- . invoice is ready
- . a dispute status changes
- . payment is past due
- . payment card is expiring
- . payment failed or rejected
- . payment is successful
- . automatic payment is created
- . automatic payment is deleted
- . new user is added and many more

2. Pdf Invoice (Get password-protected account invoices emailed to you)

Here users can create a password for a PDF invoice.

To protect information and privacy, users have to enter a password to protect PDF. Users will be required to enter this password each time the user accesses PDF invoices.

This password can be changed at any time but will be applied to new PDFs that are sent to the user. Older PDFs will still use the password they were set with at the time.

3. Pay by Text (Text us to make your payments)

Payment reminder text will be valid up to the invoice due date for a given account and will not be allowed to make the payment via text after the due date. If a customer initiates payment via text after invoice due date, an error text message will be sent to the customer. However, customers will be able to pay the invoice in the application even after the invoice due date.

System must check for full payment received in the application if the user submits the payment by text for a given account. If full payment is received, then customers will not be allowed to make the payment via text for that given account and pay by text reference number. In the event if the customer initiates payment via text, then an error text message will be sent.

If the payment fails or declines while processing the payment via text, then an error message will be sent and no retry option will be provided.

On successful processing of payment via text, a confirmation text message will be sent and the message will include account Number, payment reference number and invoice amount.

An account can have multiple invoices and each of these invoices can have different due dates, hence the user can receive multiple payment reminder text messages for the same account number based on the applicable invoice due date with a unique pay by text reference number. In this illustration below, the user will receive 2 payment text messages.

3.1.3 Country Setting Panel

Country Setting Panel consists of options to change locale/ language/ region for particular user by using the feature/ library of react internationalization intl.

By changing the country setting many features are enabled and disabled depending upon the selected region.

Country setting panel updates the state and changes the locale by using `<FormattedMessage/>` function and uses the json file depending on the locale selected.

CHAPTER 4

PERFORMANCE ANALYSIS

4.1 Analysis of System Developed

The screenshot displays a user dashboard for 'Shipping Services'. On the left, a sidebar contains navigation links: 'United States - English SS Package', 'Change Country or Business Unit', 'My Dashboard', 'My Plan Invoices', 'My Account Invoices', 'Reporting & Search', 'Dispute History & Refunds', 'My Automatic Payments', 'Payment History', 'Administration', 'Wallet', 'My Preferences', and 'Overview'. The main content area is titled 'Shipping Services' and includes a breadcrumb trail: 'My Dashboard > My Preferences > Overview'. Below this, the 'Overview' section shows the user's name and ID. Three cards are visible: 'Notifications' (Choose when to be notified about your billing activity), 'PDF Invoice' (Get password-protected account invoices emailed to you), and 'Pay by Text' (Text us to make your payments). Each card has an 'Edit' link.

Fig 4.1 Dashboard

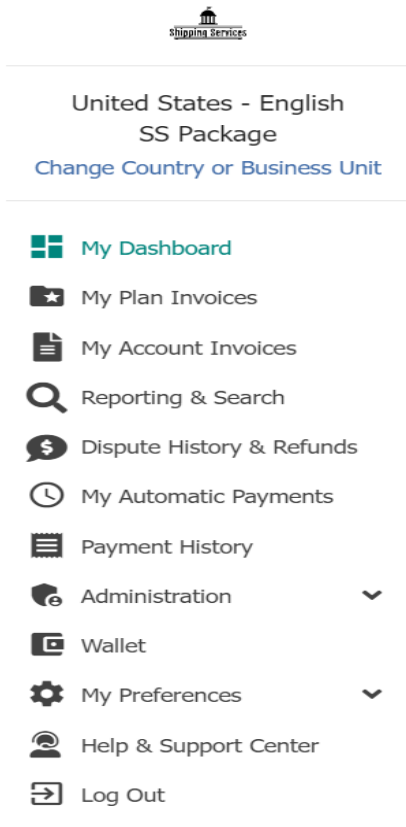


Fig 4.2 Side Panel

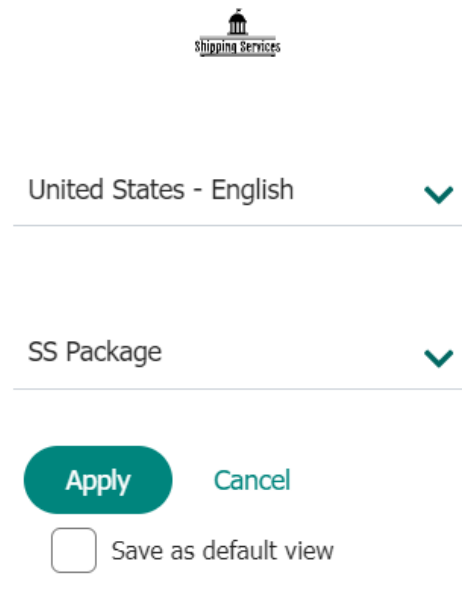


Fig 4.3 Country Setting Panel



Fig 4.4 Internationalization Output

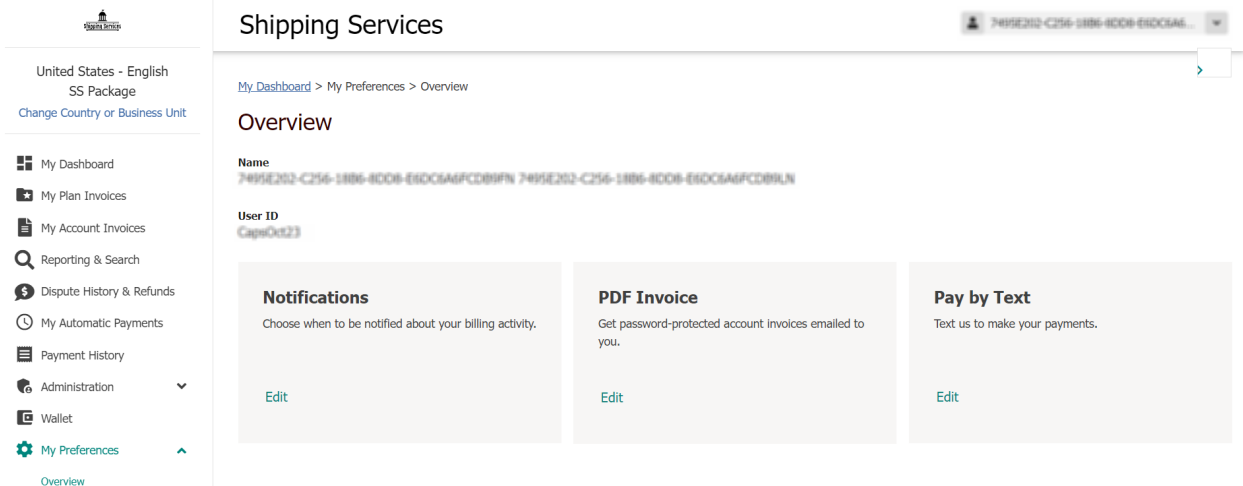


Fig 4.5 Preferences Overview

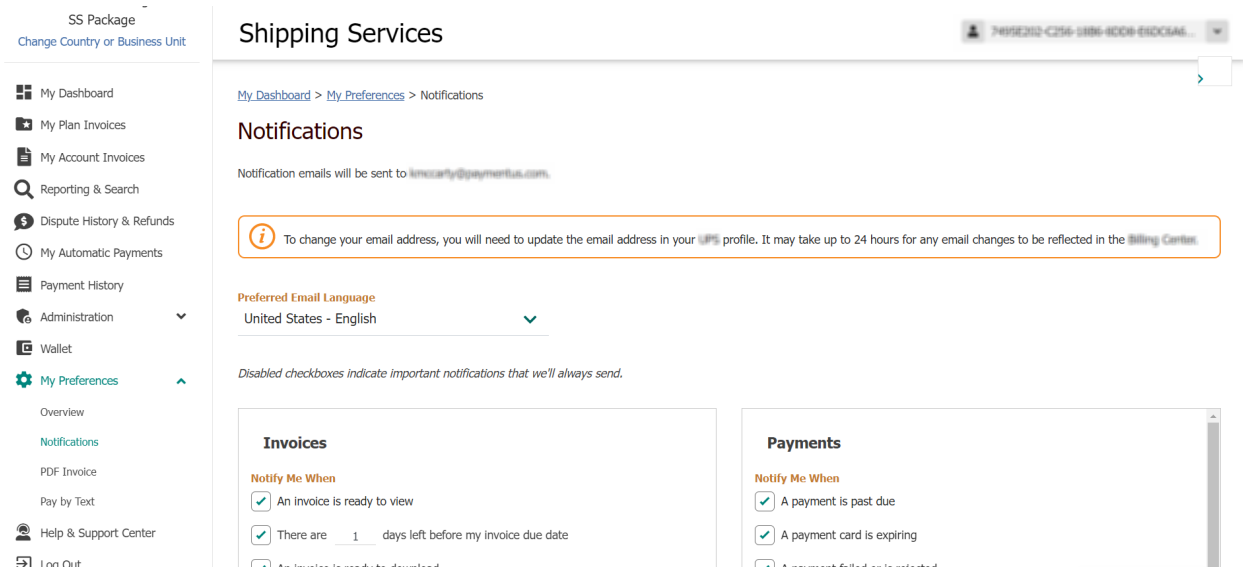


Fig 4.6 Notifications

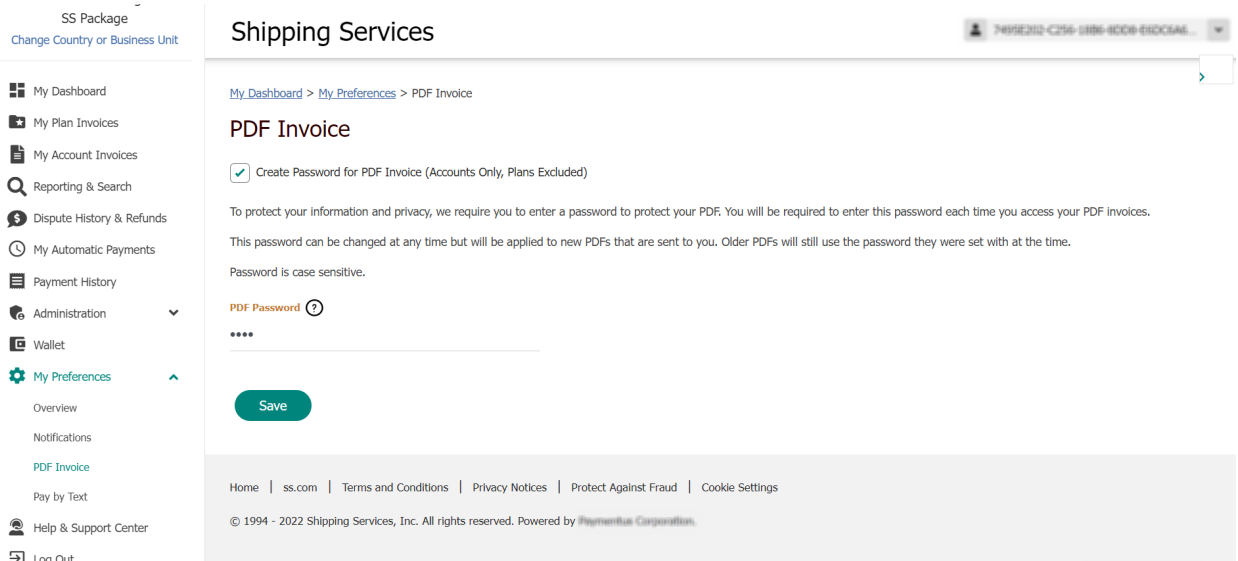


Fig 4.7 PDF Invoices

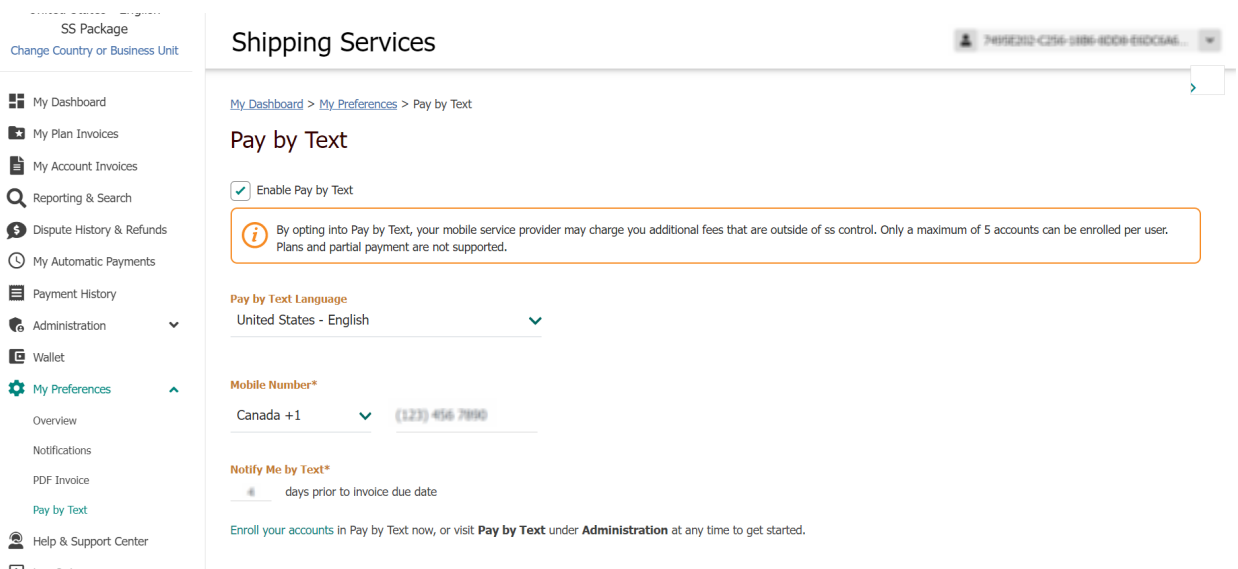


Fig 4.8 Pay By Text

Shipping Services

Administration > My Accounts

My Accounts

Search [] Add Account

Account Number	Account Name	Country or Territory	Business Unit	Actions
[]				⋮
[]				⋮
[]				⋮
[]				⋮
[]				⋮
[]				⋮
[]				⋮
[]				⋮
[]				⋮
[]				⋮

Fig 4.9 My Accounts

Shipping Services

Administration > My Plans

My Plans

Search [] Add Plan

Plan Number	Plan Name	Country or Territory	Plan Type	Actions	
[]	LU PLAN	LU PAYMENTUS CHP7	Luxembourg	Holding Plan	⋮
[]	BE PLAN	BE PAYMENTUS CHP8	Belgium	Holding Plan	⋮
[]	SE PLAN	SE PAYMENTUS CHP9	Sweden	Holding Plan	⋮

Show 10 | Showing 1 to 3 of 3 entries

Home | ss.com | Terms and Conditions | Privacy Notices | Protect Against Fraud | Cookie Settings

© 1994 - 2022 Shipping Services, Inc. All rights reserved. Powered by Paymentus Corporation.

Fig 4.10 My Plans

Shipping Services

United States - English
SS Package
Change Country or Business Unit

- My Dashboard
- My Plan Invoices
- My Account Invoices
- Reporting & Search
- Dispute History & Refunds
- My Automatic Payments
- Payment History
- Administration**
 - My Accounts
 - My Plans
 - Manage Users
 - Pay by Text
- Wallet
- My Preferences

Shipping Services

Administration > Manage Users

Manage Users

7495E2D2-C256-58B6-8D08-EDDC5AFCDBA8

Search [] Add New User

<input type="checkbox"/>	Name	Role	Last Modified Date	Last Modified By	Actions
<input type="checkbox"/>	7495E2D2-C256-58B6-8D08-EDDC5AFCDBA8	Administrator	Oct 24, 2020, 1:28:40 AM	KA McCarly	
<input type="checkbox"/>	Ashra Arora	Administrator	Dec 9, 2021, 9:31:51 PM	KA McCarly	⋮
<input type="checkbox"/>	mailorder21@Name	View, Pay, and Dispute	Feb 24, 2022, 1:20:12 PM	KA McCarly	⋮
<input type="checkbox"/>	testuser@Name	View Only	Jan 7, 2022, 5:03:02 PM	Katherine McCarly	⋮
<input type="checkbox"/>	testuser@Name	View, Pay, and Dispute	Feb 24, 2022, 1:24:17 PM	KA McCarly	⋮

Show 10 | Showing 1 to 5 of 5 entries

Fig 4.11 Manage Users

Shipping Services

United States - English
SS Package
Change Country or Business Unit

- My Dashboard
- My Plan Invoices
- My Account Invoices
- Reporting & Search
- Dispute History & Refunds
- My Automatic Payments
- Payment History
- Administration**
 - My Accounts
 - My Plans
 - Manage Users
 - Pay by Text**
 - Wallet
 - My Preferences

Shipping Services

Administration > Pay by Text

Pay by Text

7495E2D2-C256-58B6-8D08-EDDC5AFCDBA8

Search [] Enroll Account in Pay by Text

Account Number	Account Name	Payment Method	Actions
Account 185088W	RELUY 0401	PREPAIDTEST	⋮
Account 335222		*****9246 - Visa (Debit)	⋮
Account 6277777		TestBCE3	⋮
Account 158670	Ashra Sethi		⋮

Show 10 | Showing 1 to 4 of 4 entries

Home | ss.com | Terms and Conditions | Privacy Notices | Protect Against Fraud | Cookie Settings

© 1994 - 2022 Shipping Services, Inc. All rights reserved. Powered by Paymentus Corporation.

Fig 4.12 Pay By Text Enrolled Accounts

CHAPTER 5

CONCLUSION

You may examine invoices, modify payment options, and pay your existing shipping bills using the SS Payment Center.

In a single dashboard, users may observe the estimated invoice and credit deadline, as well as regulatory functions and necessary alerts.

The invoice may be paid without logging in. Create an automated payment system and modify your Payment Center wallet by adding or removing payment kinds.

You can acquire what you want with a searchable, structured table view of all your credits.

View and print payment invoices or SS Billing Data files connected to a registered shipping account using the View Bill tool.

REFERENCES

- <https://www.statista.com/chart/19058/number-of-websites-online/>
- <https://www.freecodecamp.org/news/internationalization-in-react-7264738274a0/>
- <https://www.ijert.org/research/performance-optimization-using-mern-stack-on-web-application-IJERTV10IS060239.pdf>
- <https://www.testpros.com/articles/what-is-ada-compliance-and-what-does-it-mean-for-your-website/#:~:text=ADA%20compliance%20is%20short%20for,often%20confused%20with%20508%20compliance.>
- <https://paxcom.ai/>
- <https://www.paymentus.com/>
- <https://www.javatpoint.com/mern-stack>