



HANOVER DOCUMENT

With Guide to Future developer and Test Report



Project name – Pristine Web

SOLO WEB PROJECT

BY BISHAL KUMAR SHAH - 30400939

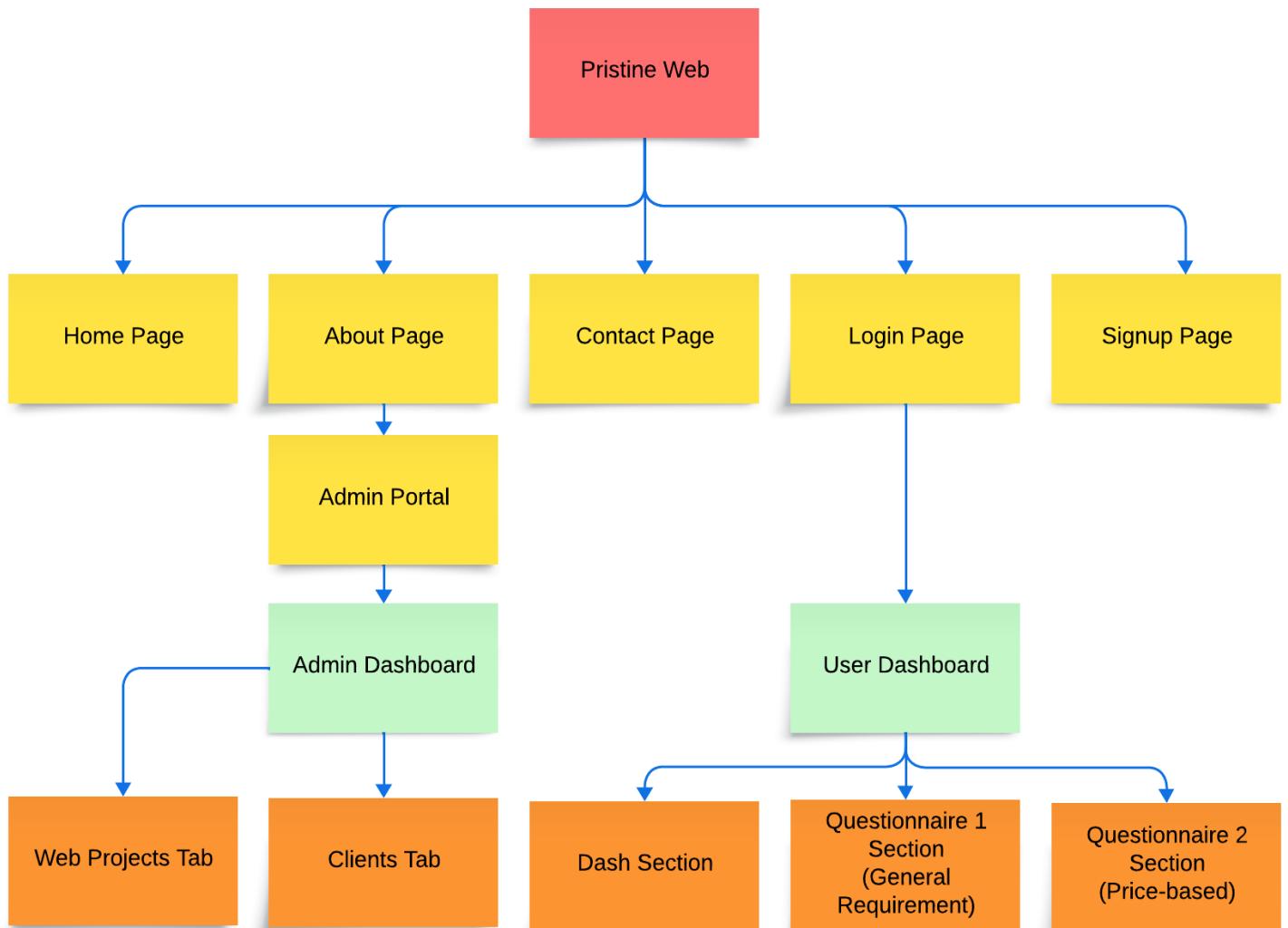
Table of Contents

Website sitemap	3
Installation Guide	4
Setting up environment in Windows.....	4
Setup database password	5
Guide to key information and future development.....	6
Updating the questionnaire and price:.....	6
Receiving the copy of PDF document by admin	8
Receiving the customer message from contact page	8
Source code structure	9
Web Stack.....	10
Database Scheme.....	10
User Dashboard	10
Comments and Testing breakpoints.....	11
User data handling	12
Product.....	13
Questionnaire 1.....	13
Questionnaire 2.....	15
Questionnaire Review Section	19
User Dashboard	20
Email API Integration.....	21
Integrated template for Invoice and Questionnaire PDF	22
Integrated API calls from PDF generator RESTful Endpoint	23
Integrated with SendGrid.....	24
Admin Portal	25
Database creation and integration.....	27
Homepage.....	28
Contact page	29
Secured access	29
Deployment Testing	30
Questionnaire Design Document.....	31
Well-structured and commented codebase	32
5 Sprint boards with feature tracking and test cases for future development.....	33
About 45 Commits for version control.....	34
Test Summary Report.....	35

Questionnaire Data Validation	35
Data Validation Summary.....	36
Test cases for the questionnaire	37
Questionnaire Navigation logic flow and expected behaviour	38
Sign-up Data Validation.....	39
Test Cases for submit_result.php Actions.....	39
Report Summary for submit_result.php	41

Website sitemap

Walkthrough Video - [Link](#)



Installation Guide

Installation Video - [Link](#)

Recommended to install in XAMPP local server initially. The project was developed and tested in XAMPP server with the following specifications:

- PHP version 8.2.4
- Apache Web Server version 2.4.56
- MySQL (MariaDB) version 8.2.4

Apache Version	Apache/2.4.56 (Win64) OpenSSL/1.1.1t PHP/8.2.4
----------------	------------------------------------------------

Setting up environment in Windows

Steps

1. Install XAMPP 3.3.0 from [here](#).
2. Copy the source code to the Install_Directory/ Xampp/htdocs/
3. Enable Apache and MySQL in the Xampp Control Panel



4. Go to <http://localhost/phpmyadmin/> in the web browser (Chrome, Firefox, etc.)
 5. Go to 'Import' tab
-
- A screenshot of the phpMyAdmin 'Import' tab. At the top, there is a navigation bar with several tabs: Databases, SQL, Status, User accounts, Export, Import (which is currently selected), Settings, Replication, and More. Below the tabs, there is a form for selecting a file to import, with a 'Choose File' button and a 'Import' button at the bottom.

6. Choose File to import from

Source_Code_Directory/database/Database_Schema_with_Example_Data.sql and click 'Import' from the bottom of the page.

All done. By default, the database should be connected using default password.

Setup database password

To setup the password in order to connect the website to database,

1. Update the User Name, Password and permissions to database from User Accounts tab

The screenshot shows the MySQL Workbench interface with the 'User accounts' tab selected. Below it, the 'User accounts overview' tab is active. A table lists five user accounts:

User name	Host name	Password	Global privileges	User group	Grant	Action
Any	%	No	USAGE		No	Edit privileges Export Lock
pma	localhost	No	USAGE		No	Edit privileges Export Lock
root	127.0.0.1	No	ALL PRIVILEGES		Yes	Edit privileges Export Lock
root	::1	No	ALL PRIVILEGES		Yes	Edit privileges Export Lock
root	localhost	No	ALL PRIVILEGES		Yes	Edit privileges Export Lock

- 2.

- a. Source code will contain the following code in multiple files.

```
$servername = "localhost";
$username = "root";
$password = "";
$database = "pristine_web_itech_db";
```

- b. Similarly, Update the username and password for following files in the database:
 - i. admin_login_signup.php
 - ii. login.php
 - iii. signup.php
 - iv. admin_dashboard_1
 - v. admin_dashboard_2
 - vi. submit_result.php
 - vii. website_card.php

After updating the credential in these files, the password will be successful update and database will be linked to the website.

Guide to key information and future development

Updating the questionnaire and price:

The content of the price-based questionnaire is located in the file:

source_code/dashboard/user_dashboard1.php

The following properties can be found the file inside the questionnaire2JSON object which can be used to change the question details:

- title
- description
- value
- text

```
<script>
  const questionnaire2Json = {
    logoPosition: "right",
    completedHtml: "<h3>Thank you for completing the questionnai",
    completedBeforeHtml: "<h3>Our records show that you have com
  pages: [
    {
      name: "page1",
      elements: [
        {
          type: "radiogroup",
          name: "question1",
          useDisplayValuesInDynamicTexts: false,
          title: "Website Type",
         isRequired: true,
          description: "What type of website is it?",
          choices: [
            {
              value: "1500",
              text: "Business",
            },
            {
              value: "500",
              text: "Personal ",
            },
            {
              value: "3000",
              text: "E-commerce",
            },
          ],
        },
      ],
    },
  ],
}
```

A Question for reference

1. Website Type *

What type of website is it?

- Business
- Personal
- E-commerce

Property	Details
title	Question title
description	Question description
value	Price of choices in the question
text	Choices of the questions

You can change these properties and save the file to update the questionnaires.

Note: Do not use the exact same ‘value’ property for the choices within a single question, as ‘value’ property must be unique. As the ‘value’ property is used to calculate the total price, there can be instances where prices can be same for different choices within a question, in which case, these values can be modified to vary in 3 decimal points. For instance,

- value of choice 1: 300.001
- value of choice 2: 300.002
- value of choice 3: 300.003

The total prices calculated will still be accurate as the last 2 decimal points will be bypassed through a Function to produce the total price.

Receiving the copy of PDF document by admin

The email address should be changed to receive the copy of the pdf by the admin. The variable \$to containing the array of the email address can be updated. To do so,

- Locate the submit_result.php in the ‘dashboard’ directory and replace the email address ‘bishal.shah2344@gmail.com’ with ‘admin_email@address.com’ as shown in the screenshot below.

```
//Construct the email content
$to = array($js_email, 'bishal.shah2344@gmail.com'); // Replace with the actual email addresses to send the copy
$subject = "Invoice and Website details from Prestine Web";
$body = "Hi " . $first_name . ",\n";
$body .= "Thank You for completing the questionnaire.\n";
$body .= "Your website details and invoice are attached.";

// SendGrid API configuration
$sendgrid_api_key = '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'; // Replace with your SendGrid API key
$url = 'https://api.sendgrid.com/v3/mail/send';
```

- Then save the file to start receiving the email address to specified email address.
- Additionally, the variables \$subject and \$body can be replaced to update the messages that the customer receives from after completion of the questionnaire.

Receiving the customer message from contact page

Similar to steps above, the customer message from contact page can be receive by replacing the email address variable in the file source_code/send_email.php containing \$to variable. The screenshot is shown below:

```
// Construct the email content
$to = "bishal.shah2344@gmail.com"; // Replace with the actual client email address
$subject = "Message to PrestineWeb from ". $first_name;
$body = "First Name: " . $first_name . "\n";
$body .= "Last Name: " . $last_name . "\n";
$body .= "Phone No.: " . $phone . "\n";
$body .= "Country: " . $country . "\n";
$body .= "Email: " . $email . "\n";
$body .= "Message:\n" . $message . "\n\n";
$body .= "Received via Pristine Softsol Website - ITECH eCommerce project";
```

The “Bishal.shah2344@gmail.com” can be updated with the actual admin email address to receive the message from the customer.

Source code structure

```
└── css
    ├── # contact_style.css
    ├── # dashboard_style.css
    ├── # home_style.css
    ├── # login_style.css
    └── # signup_style.css
└── dashboard
    └── JS
        ├── JS input_logic.js
        ├── # input_style.css
        ├── JS jquery.min.js
        ├── JS questionnaire1CssVariables.js
        ├── admin_dashboard_1.php
        ├── admin_dashboard_2.php
        ├── submit_result.php
        ├── user_dashboard_1.php
        ├── user_dashboard_2.php
        └── website_card.php
> dashboard_v0.1
└── database
    └── Database_Schema_with_Ex... U
> images
└── about.html
├── admin_dashboard.php
├── admin_login_signup.php
├── admin_logout.php
└── contact.html
├── dashboard.php
└── index.html
├── login.php
└── logout.php
├── README
└── send_email.php
├── session.php
└── signup.php
```

Web Stack

The components of web stack used for this project are:

- HTML, CSS and Javascript for front-end
- PHP for back-end
- MySQL for database
- Apache for web server
- SendGrid as mail server using API integration
- Jinja2 as PDF templating language
- APItemplate as PDF rendering service using API integration

Database Scheme

Database schema can be found in the file:

[**Source_Code_Directory/database/Database_Schema_with_Example_Data.sql**](#)

It also includes the test data for further references.

User Dashboard

Page for User Dashboard is rendered in two sections using <?php include > tag from folder containing files:

- user_dashboard_1.php
- user_dashboard_2.php

These files contain questionnaire1 section for requirement gathering questions and questionnaire2 section for price-based questions respectively. Each of these files are independently rendered using input_logic.js script. Therefore, overall themes and design will not affect one another.

Comments and Testing breakpoints

There are helpful comments and testing breakpoints throughout the codebase.

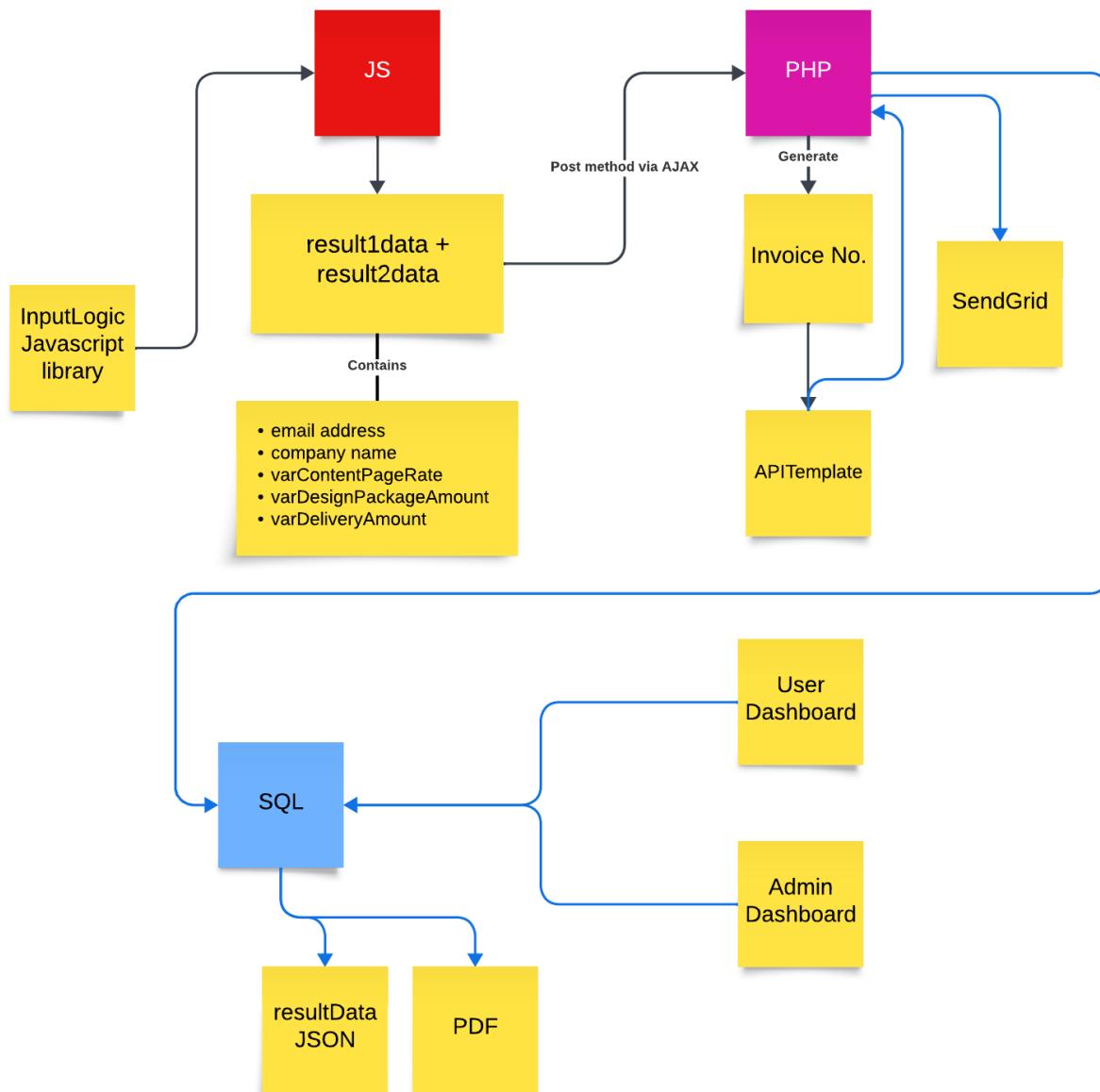
```
1  <?php
2  include '../session.php';
3
4  $resultData = $_POST['resultData'];
5  $userID = $_SESSION['user_id'];
6
7
8
9  //For testing - create a file and save resultData to it
10 /*
11  $file = fopen("resultData.txt", "w");
12  fwrite($file, $resultData);
13  fclose($file);
14 */
15
16 //save file to database
17
18 $servername = "localhost";
19 $username = "root";
20 $password = "";
21 $database = "pristine_web_itech_db";
22
23 // Establish database connection
24 $conn = mysqli_connect($servername, $username, $password, $database);
25 // Prepare the SQL statement with a parameter placeholder
26 $query = "INSERT INTO customer_data (user_id, user_data_json) VALUES (?, ?)";
27 $stmt = mysqli_prepare($conn, $query); // Prepare the statement
28 if ($stmt) {
29     mysqli_stmt_bind_param($stmt, "ss", $userID, $resultData); // Bind the parameters
30     mysqli_stmt_execute($stmt); // Execute the statement
31     // Get the invoice ID
32     $invoiceID = mysqli_insert_id($conn);
33     if (mysqli_stmt_error($stmt)) {
34         echo "Error: " . mysqli_stmt_error($stmt); // Check for errors
35     } else {
36         // Log the invoice ID
37         echo "<script>console.log('Saved to database successfully. Invoice ID: $invoiceID');</script>";
38     }
39     mysqli_stmt_close($stmt); // Close the statement
40 } else {
41     echo "Error preparing SQL statement: " . mysqli_error($conn);
42 }
```

You can use these breakpoints to test if the data is sent till the respective lines in the execution stage. Most of the output will be displayed to the console in web browser. Some breakpoints are made to directly output to the html page of the DOM model.

User data handling

The captured user's data are sent to submit_result.php from user_dashboard_2.php using AJAX Jquery function. After which, submit_result.php takes the following actions:

- Logs into the database
- Stores the captured result data in customer_data table along with the user_id.
- Retrieves the invoice_id from the auto-generated database field.
- Merges it to the Json data and make an API call to PDF Generator RESTful endpoint.
 - o PDF Generator endpoint render the data into the pdf.
 - o And sends the PDF URL as a HTTP response.
- Downloads the PDF and encodes it to base64 format.
- Makes the API request to SendGrid Cloud environment with PDF attached.
 - o SendGrid sends the email to Admin and User



Product

Questionnaire 1

Welcome to Your Dashboard

localhost/Red_Project_Pristine_Web/dashboard.php

Home Service About Contact Bishal

Are you looking for a specific style or type of website?

Other Design Portfolio
 Web store Restaurant
 Cleaning Company Plumbing Service
 Car Workshop English Pub
 Construction Company Dry Cleaning
 Business Consulting Flower Shop

What design do you prefer? *

Modern 

Corporate 

Clean 

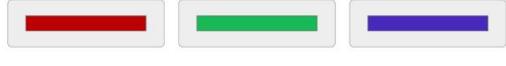
Artistic 

Tone and Atmosphere

Light Dark

Select the general colour of website for colour scheme

Primary Secondary Tertiary

Colour 

What is your target audience?

	The Perfect Audience is	A Great Audience is	A Good Audience is	Another Target is
Suburb	Richmond	Ringwood	Hawthorne	Nunawading
City	Melbourne	Melbourne	Melbourne	Melbourne
Country	Australia	Australia	Australia	Australia

Targeted audience demographic

Gender:

Male Female Other

Lifestyle:

Food Lover, Student

Age:

15 to 60

Socio economic / Income:

All groups

Likes / Dislikes:

Likes Healthy Food

Add your Contact details

Company name: *

Alex's Souvlakia Restaurant

Phone: *

04234567890

Address:

4 Frank St. Richmond 3131 VIC

Add description and history of the company

It is a souvlaki restaurant and we prioritize in serving healthy food. We have two branches in Richmond and Ringwood Melbourne Victoria.

Add additional information to let us know exactly how can your website be perfect

I would like the website to be responsive especially on Mobile devices as most customers will order using mobile phones.

Your email address *

Bishal.shah2344@gmail.com

Let's Configure

Questionnaire 2

[Home](#)[Service](#)[About](#)[Contact](#)[Bishal](#)

1. Website Type *

What type of website is it?

 Business Personal E-commerce

2. Pages *

What pages do you need?

 Home page Contact page Gallery page About page FAQ page Contact + About page in one

3. Content Page

These are informational pages such as Articles, News, Blogs, or Content for specific topics.

Custom Page Name

Description

Menu page

It will include menu items and images.



Restaurant History Page

It will include the history and a few images.

[Add Page](#)

4. Legal Information pages

 Terms & Conditions page Privacy Policy page

5. Content Management

Do you need CMS to manage the content of the website?

 CMS admin dashboard page[Next](#)

\$3500.00

[Home](#)[Service](#)[About](#)[Contact](#)[Bishal](#)

6. Account Pages

Do you need user account feature?

[No](#)[Yes](#) Login page and Sign up page Login + Sign up in one page Login/signup directly using pop-up

7. Login pop-up feature

Add Login pop-up feature from homepage

[No](#)[Yes](#)

8. Express Login

Option to fast Login from other account

 Google account Facebook account

9. Interactive Map

Do you need an Interactive map on the Contact page?

[No](#)[Yes](#)

10. Booking System

Do you need Booking system to book appointments, tickets, calls, etc.?

[No](#)[Yes](#)[Previous](#)[Next](#)**\$4270.00**

[Home](#)[Service](#)[About](#)[Contact](#)[Bishal](#)

Payment gateway

11. Do you need Payment Gateway?

No Yes

Payment accepting and checkout methods

- | | |
|-----------------------------------------------------------|-----------------------------------------------|
| <input checked="" type="checkbox"/> Credit and debit card | <input checked="" type="checkbox"/> Paypal |
| <input checked="" type="checkbox"/> Google Pay | <input checked="" type="checkbox"/> Apple Pay |

BNPL company

Buy now, Pay later for Loan and installments payment service

- AfterPay
 ZipPay
 Klarna

12. Express Checkout

- Paypal Express Checkout
 Stripe Express Checkout

13. Blog

Do you want to add a blogging feature to the site?

No Yes

14. Social Media Live Feed

Do you want to include social media live feed on your page?

No Yes

[Previous](#)

[Next](#)

\$5570.01

15. **Newsletter feature**

Do you want to add newsletter function to the website?

No

Yes

16. **Logo**

Do you want the logo created?

No

Yes

17. **Design Package ***

Tier 1 - Standard

Tier 2 - Gold (+15%)

Tier 3 - Platinum (30%)

Tier 1: Standard

**Templates,
Exceptional Quality**

- Sleek and Modern Design
- Mobile-Responsive
- Optimized for Speed and Performance

Tier 2: Original

Designs for a Unique Identity

- Custom Design Concepts
- Interactive and Engaging Elements
- Personalized User Experience

Tier 3: Premium

Ground-Up Designs for Unmatched Excellence

- Bespoke, Tailored Design
- Cutting-Edge Technology
- Unlimited Customization

[More Information](#)

Hide

Show

18. **SSL Certification**

Do you need an SSL certificate? (Price based on level of security, bought from Certificate Authority)

DV Certificate - For sites, such as blogs or small business websites

OV certificate - For sites, such as business websites with forms and lead capture capabilities

EV certificate - For the highest level of security, capable of handling sensitive information

19. **Website delivery date ***

When do you need the website created?

3 months - Standard delivery

2 months - Express delivery (+30%)

4 weeks - Expedited service (+50%)

Total

\$8,790.61

[Previous](#)

[Preview](#)

\$8790.61

Questionnaire Review Section



Home **Service** **About** **Contact** **Bishal** 

1. Website Type *

What type of website is it?

Business
 Personal
 E-commerce

2. Pages *

What pages do you need?

Home page Contact page
 Gallery page About page
 FAQ page Contact + About page in one

3. Content Page

These are informational pages such as Articles, News, Blogs, or Content for specific topics.

Custom Page Name	Description
Menu page	It will include menu items and images.
Restaurant History Page	It will include the history and a few

4. Legal Information pages

Terms & Conditions page
 Privacy Policy page

5. Content Management

Do you need CMS to manage the content of the website?

CMS admin dashboard page

Edit

6. Account Pages

Do you need user account feature?

No Yes

Login page and Sign up page
 Login + Sign up in one page
 Login/signup directly using pop-up

7. Login pop-up feature

Add Login pop-up feature from homepage

No Yes

8. Express Login

Option to fast Login from other account

Google account
 Facebook account

9. Interactive Map

Do you need an Interactive map on the Contact page?

No Yes

10. Booking System

Do you need Booking system to book appointments, tickets, calls, etc.?

No Yes

Edit

Payment gateway

11. Do you need Payment Gateway?

No Yes

Payment accepting and checkout methods

Credit and debit card Paypal
 Google Pay Apple Pay
BNPL company
Buy now, Pay later for Loan and installments payment service

AfterPay
 ZipPay
 Klarna

12. Express Checkout

Paypal Express Checkout
 Stripe Express Checkout

13. Blog

Do you want to add a blogging feature to the site?

No Yes

14. Social Media Live Feed

Do you want to include social media live feed on your page?

No Yes

Edit

15. Newsletter feature

Do you want to add newsletter function to the website?

No Yes

16. Logo

Do you want the logo created?

No Yes

17. Design Package *

Tier 1 - Standard Tier 2 - Gold (+15%) Tier 3 - Platinum (30%)

Tier 1: Standard Templates, Exceptional Quality <ul style="list-style-type: none"> • Sleek and Modern Design • Mobile-Responsive • Optimized for Speed and Performance 	Tier 2: Original Designs for a Unique Identity <ul style="list-style-type: none"> • Custom Design Concepts • Interactive and Engaging Elements • Personalized User Experience 	Tier 3: Premium Ground-Up Designs for Unmatched Excellence <ul style="list-style-type: none"> • Bespoke, Tailored Design • Cutting-Edge Technology • Unlimited Customization
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

More Information

Hide Show

18. SSL Certification

Do you need an SSL certificate? (Price based on level of security, bought from Certificate Authority)

DV Certificate - For sites, such as blogs or small business websites
 OV certificate - For sites, such as business websites with forms and lead capture capabilities
 EV certificate - For the highest level of security, capable of handling sensitive information

19. Website delivery date *

When do you need the website created?

3 months - Standard delivery
 2 months - Express delivery (+10%)
 4 weeks - Expedited service (+50%)

Total
\$8,790.61

Edit

Complete

User Dashboard: Web Project View Screen

The screenshot shows a web browser window with a yellow header bar. The address bar displays "localhost/Red_Project_Pristine_Web/dashboard.php". The main content area features a navigation bar with links for Home, Service, About, Contact, and a user profile for Bishal. Below this is a section titled "My Website" containing two lists of website features. At the bottom are "Download", "View", and "New" buttons. A modal window is open, displaying an invoice for "Pristine Web" with an invoice number of 10009. The invoice details include the recipient as "Alex's Souvlaki Restaurant" and a table of items and their totals. Below the modal, there are Q&A sections for contact information and pricing model.

Welcome to Your Dashboard

localhost/Red_Project_Pristine_Web/dashboard.php

Home Service About Contact Bishal

My Website

- Business Type
- Modern Design
- Restaurant
- Tier 2 - Gold (+15%)
- 2 months - Express delivery (+30%)

- Home page
- Gallery page
- FAQ page
- Contact page
- About page

Download View New

9d85f0e9-5375-4240... 1 / 6 75% + : 1/6

INVOICE

Pristine Web

Invoice No.: 10009

E-MAIL:
Bishal.shah2344@gmail.com

BILL TO Alex's Souvlaki Restaurant

ITEM NAME	TOTAL
Website Type	
Business	\$1500.00
Pages	
Home page	\$300.00
Gallery page	\$300.00
FAQ page	\$100.00
Contact page	\$200.00
About page	\$200.00
Content Page	
Menu page	\$200
Restaurant History Page	\$200

Q: How can I contact you?
A: You can reach us at pristine@redTeam.com or by phone at +61234567890.

Q: What is your pricing model?
A: Our pricing is flexible and tailored to meet individual project requirements. Please contact us for details.

Email API Integration

The screenshot shows a Gmail inbox with a single message highlighted. The message subject is "Invoice and Website details from Pristine Web". The message body contains an attached PDF file named "Invoice_Website_Details.pdf". The PDF document is an invoice from "Pristine Web" with Invoice No.: 10007. The invoice details various website features and their costs. The left sidebar of the Gmail interface shows the user's inbox, starred, snoozed, important, sent, drafts, categories, social, updates, forums, promotions, and more. Labels are also visible.

INVOICE

Pristine Web

Invoice No.: 10007

E-MAIL:
bishal.shah2344@gmail.com

BILL TO Prestige Vehicle Ltd.

ITEM NAME	TOTAL			
Website Type	\$1500.00			
Pages				
Home page	\$300.00			
Gallery page	\$300.00			
FAQ page	\$100.00			
About page	\$200.00			
Contact page	\$200.00			
Content Page				
Car show page	\$200			
Staff roster page	\$200			
Legal Information pages				
Privacy Policy page	\$300.00			
Content Management				
CMS admin dashboard page	\$200.00			
Account Pages				
Login + Sign up in one page	\$300.00			
Login pop-up feature	\$75.00			
Express Login				
Google account	\$80.00			
Facebook account	\$80.00			
Interactive Map				
Interactive Map	\$90.00			
Booking System				
Booking System	\$0.00			
Payment accepting and checkout methods				
Google Pay	\$100.00			
Credit and debit card	\$200.00			
Paypal	\$100.00			
BNPL company				
ZipPay	\$100.00			
Express Checkout				
Paypal Express Checkout	\$100.00			
Other features				
Blog	\$300.00			
Design Package				
Tier 3 - Platinum (30%)	\$1555.5			
SSL Certification				
DV Certificate - For sites, such as blogs or small business websites	\$60.00			
Website delivery date	Page 1 / 6	-	+	
				\$2022.15

Integrated template for Invoice and Questionnaire PDF using Jinja2 programming language

{% set Q16AnsArray= question16.answer.split(' ') %}	
ITEM NAME	TOTAL
{question1.question}	{% if question1.answer%}
{{question1.answer}}	{ "\$.2f" format(question1.value)}
{%endif%}	{% if question2.answer%}
{question2.question}	
{% for answer in Q2AnsArray %}	
{{answer}}	{ "\$.2f" format(question2.value[loop.index0])}
{% endfor %}{%endif%}	{% if question3.answer%}
{question3.question}	
{% for answer in question3.value%}	
{{answer.column1}}	\${calcValues.varContentPageRate}
{% endfor %}{%endif%}	{% if question5.answer%}
{question5.question}	
{% for answer in Q5AnsArray %}	
{{answer}}	{ "\$.2f" format(question5.value[loop.index0])}
{% endfor %}{%endif%}	{% if question6.answer%}
{question6.question}	
{{question6.answer}}	{ "\$.2f" format(question6.value[0])}
{%endif%}	{% if question7.answer%}
{question7.question}	
{{question8.value}}	{ "\$.2f" format(question7.value)}
{%endif%}	{% if question9.answer%}
{question9.question}	
{%endif%}	{% if question10.answer%}
{question10.question}	
{% for answer in Q10AnsArray %}	
{{answer}}	{ "\$.2f" format(question10.value[loop.index0])}
{% endfor %}{%endif%}	{% if question11.answer%}
{question11.question}	
{{question11.question}}	{ "\$.2f" format(question11.value)}
{%endif%}	{% if question12.answer%}
{question12.question}	
{{question12.question}}	{ "\$.2f" format(question12.value)}
{%endif%}	{% if question14.answer%}
{question14.question}	
{% for answer in Q14AnsArray %}	
{{answer}}	{ "\$.2f" format(question14.value[loop.index0])}
{% endfor %}{%endif%}	{% if question15.answer%}
{question15.question}	
{% for answer in Q15AnsArray %}	
{{answer}}	{ "\$.2f" format(question15.value[loop.index0])}
{% endfor %}{%endif%}	{% if question16.answer%}
{question16.question}	
{% for answer in Q16AnsArray %}	
{{answer}}	{ "\$.2f" format(question16.value[loop.index0])}
{% endfor %}{%endif%}	{% if question17.value%}
Other features	
{{question17.question}}	{ "\$.2f" format(question17.value)}
{%endif%}	{% if question18.value%}
{{question18.question}}	{ "\$.2f" format(question18.value)}
{%endif%}	{% if question19.value%}
{{question19.question}}	{ "\$.2f" format(question19.value)}
{%endif%}	{% if question20.answer%}

DIV » SPAN

346 WORDS POWERED BY TINY

BILL TO Alex's Souvlaki Restaurant

ITEM NAME	TOTAL
Website Type	
Business	\$1500.00
Pages	
Home page	\$300.00
Gallery page	\$300.00
FAQ page	\$100.00
Contact page	\$200.00
About page	\$200.00
Content Page	
Menu page	\$200
Restaurant History Page	\$200
Legal Information pages	
Privacy Policy page	\$300.00
Content Management	
CMS admin dashboard page	\$200.00
Account Pages	
Login + Sign up in one page	\$300.00
Login pop-up feature	
	\$0.00
Express Login	
Google account	\$80.00
Interactive Map	
Interactive Map	\$90.00
Booking System	
Booking System	\$300.00
Payment accepting and checkout methods	
Credit and debit card	\$200.00
Google Pay	\$100.00
Apple Pay	\$100.00
Paypal	\$100.00
BNPL company	
ZipPay	\$100.00
Express Checkout	
Paypal Express Checkout	\$100.00
Other features	
Blog	\$300.00
Social Media Live Feed	\$300.00
Newsletter feature	\$200.00
Design Package	
Tier 2 - Gold (+15%)	\$882
SSL Certification	
OV certificate - For sites, such as business websites with forms and lead capture capabilities	\$110.00
Website delivery date	
2 months - Express delivery (+30%)	\$2028.6
Total	\$8790.61

THANK YOU FOR YOUR BUSINESS!

Pristine Web

Zoom 100%

Integrated API calls from PDF generator RESTful Endpoint

```
{% set Q15AnsArray= question15.answer.split(' ') %}
{% set Q16AnsArray= question16.answer.split(' ') %}



| ITEM NAME                          | TOTAL                                                    |
|------------------------------------|----------------------------------------------------------|
|                                    | {% if question1.answer%}                                 |
| {{question1.question}}             |                                                          |
| {{question1.answer}}               | <b>[\$%.2f  format(question1.value)]}}</b>               |
| {%endif%}                          | {% if question2.answer%}                                 |
| {{question2.question}}             |                                                          |
| {% for answer in Q2AnsArray %}     |                                                          |
| {{answer}}                         | <b>[\$%.2f  format(question2.value[loop.index0])]}]</b>  |
| {% endfor %}{%endif%}              | {% if question3.answer%}                                 |
| {{question3.question}}             |                                                          |
| {% for answer in question3.value%} |                                                          |
| {{answer.column1}}                 | <b>\$[[calcValues.varContentPageRate]]</b>               |
| {% endfor %}{%endif%}              | {% if question5.answer%}                                 |
| {{question5.question}}             |                                                          |
| {% for answer in Q5AnsArray %}     |                                                          |
| {{answer}}                         | <b>[\$%.2f  format(question5.value[loop.index0])]}]</b>  |
| {% endfor %}{%endif%}              | {% if question6.answer%}                                 |
| {{question6.question}}             |                                                          |
| {{question6.answer}}               | <b>[\$%.2f  format(question6.value[0])]}]</b>            |
| {%endif%}                          | {% if question7.answer%}                                 |
| {{question7.question}}             |                                                          |
| {{question8.value}}                | <b>[\$%.2f  format(question7.value)]}}</b>               |
| {%endif%}                          | {% if question9.answer%}                                 |
| {{question9.question}}             | <b>[\$%.2f  format(question9.value)]}}</b>               |
| {%endif%}                          | {% if question10.answer%}                                |
| {{question10.question}}            |                                                          |
| {% for answer in Q10AnsArray %}    |                                                          |
| {{answer}}                         | <b>[\$%.2f  format(question10.value[loop.index0])]}]</b> |
| {% endfor %}{%endif%}              | {% if question11.answer%}                                |
| {{question11.question}}            |                                                          |
| {{question11.question}}            | <b>[\$%.2f  format(question11.value)]}}</b>              |
| {%endif%}                          | {% if question12.answer%}                                |
| {{question12.question}}            |                                                          |
| {{question12.question}}            | <b>[\$%.2f  format(question12.value)]}}</b>              |
| {%endif%}                          | {% if question14.answer%}                                |
| {{question14.question}}            |                                                          |
| {% for answer in Q14AnsArray %}    |                                                          |
| {{answer}}                         | <b>[\$%.2f  format(question14.value[loop.index0])]}]</b> |
| {% endfor %}{%endif%}              | {% if question15.answer%}                                |
| {{question15.question}}            |                                                          |
| {% for answer in Q15AnsArray %}    |                                                          |
| {{answer}}                         | <b>[\$%.2f  format(question15.value[loop.index0])]}]</b> |
| {% endfor %}{%endif%}              | {% if question16.answer%}                                |
| {{question16.question}}            |                                                          |
| {% for answer in Q16AnsArray %}    |                                                          |
| {{answer}}                         | <b>[\$%.2f  format(question16.value[loop.index0])]}]</b> |
| {% endfor %}{%endif%}              | {% if question17.value%}                                 |
| <b>Other features</b>              |                                                          |
| {{question17.question}}            | <b>[\$%.2f  format(question17.value)]}}</b>              |
| {%endif%}                          | {% if question18.value%}                                 |
| {{question18.question}}            | <b>[\$%.2f  format(question18.value)]}}</b>              |
| {%endif%}                          | {% if question19.value%}                                 |
| {{question19.question}}            | <b>[\$%.2f  format(question19.value)]}}</b>              |
| {%endif%}                          | {% if question20.answer%}                                |

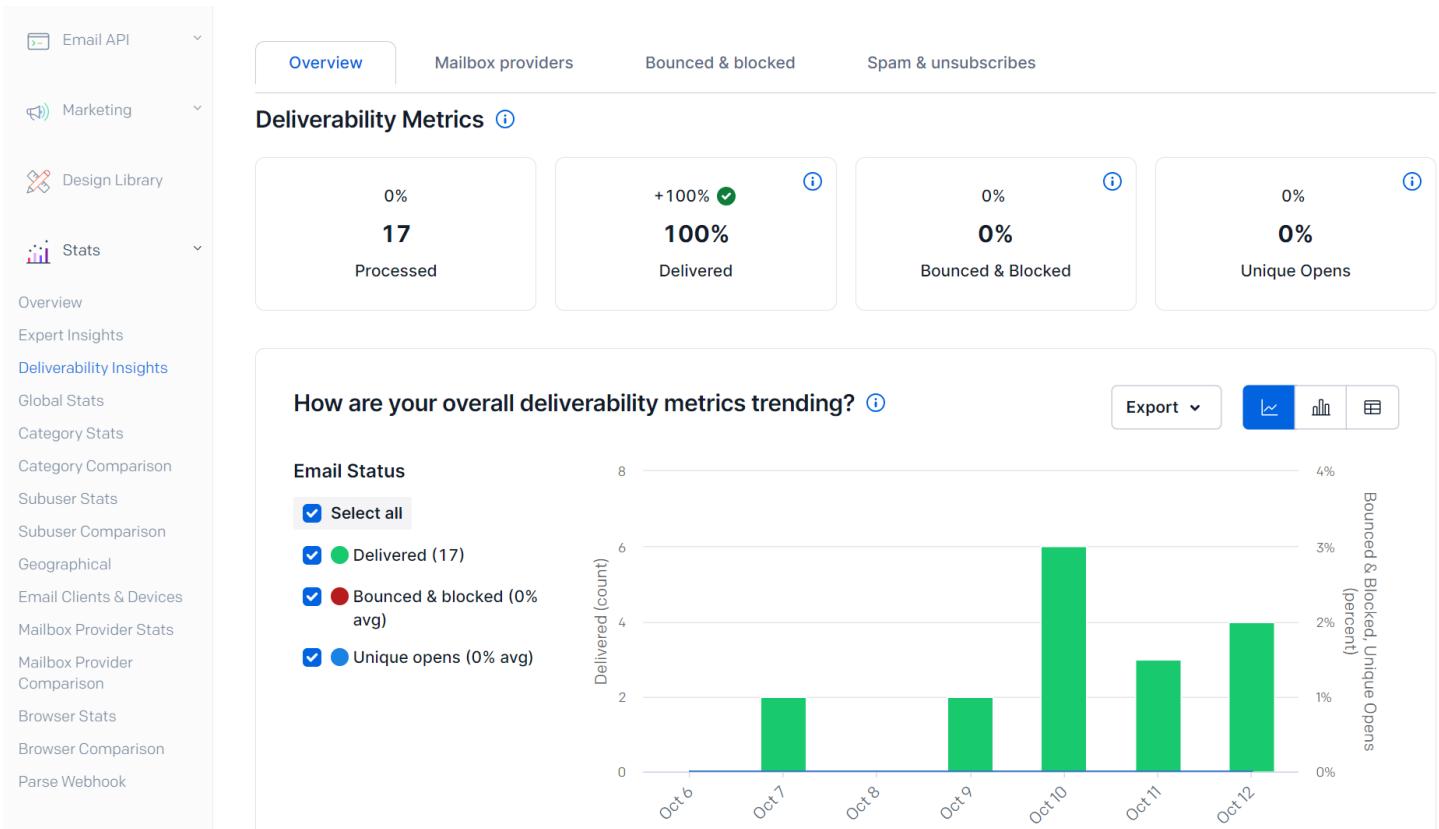

```

DIV > SPAN

346 WORDS POWERED BY TINY

```
1 {
2     "invoice_id": 10003,
3     "question1": {
4         "name": "question1",
5         "question": "Website Type",
6         "value": 1500,
7         "answer": "Business"
8     },
9     "question2": {
10        "name": "question2",
11        "question": "Pages",
12        "value": [300,300,100,200,200],
13        "answer": "Home page, Gallery page, FAQ page, Contact page, About page"
14    },
15    "question3": {
16        "name": "question3",
17        "question": "Content Page",
18        "value": [
19            {
20                "column1": "Menu page",
21                "column2": "It will include menu items and images."
22            },
23            {
24                "column1": "Restaurant History Page",
25                "column2": "It will include the history and a few images."
26            }
27        ],
28        "answer": [
29            {
30                "Custom Page Name": "Menu page",
31                "Description": "It will include menu items and images."
32            },
33            {
34                "Custom Page Name": "Restaurant History Page",
35                "Description": "It will include the history and a few images."
36            }
37        ]
38    },
39    "question4": {
40        "name": "question4",
41        "question": "eCommerce pages",
42        "value": "",
43        "answer": ""
44    },
45    "question5": {
46        "name": "question5",
47        "question": "Legal Information pages",
48        "value": [300],
49        "answer": "Privacy Policy page"
50    },
51    "question6": {
52        "name": "question6",
53        "question": "Other pages"
54    }
55}
```

Integrated with SendGrid Cloud environment as mail server using API calls



Admin Portal

The screenshot shows a web browser window with a yellow header bar. The title bar says "Sign Up" and the address bar shows "localhost/Red_Project_Pristine_Web/admin_login_signup.php". The main content area has a white background with a red header bar containing the text "Client". Below the header, there is a navigation menu with links: Home, Service, About, Contact, Admin, and Client. The "Admin" link is highlighted in red. The main content area is titled "Admin Portal". It contains two sections: "Sign Up" on the left and "Login" on the right. The "Sign Up" section has fields for First Name, Last Name, Date of Birth, Country Location, Email, and Password, each with a red border. A red "Sign Up" button is at the bottom. The "Login" section has fields for Email and Password, each with a red border. A red "Login" button is at the bottom.

Sign Up

First Name:

Last Name:

Date of Birth:

Country Location:

Email:

Password:

Sign Up

Login

Email:

Password:

Login

Admin Panel: Web Projects Page

The screenshot shows a web browser window with a yellow header bar. The address bar displays "localhost/Red_Project_Pristine.Web/admin_dashboard.php". The main content area is titled "Web Projects". On the left, there's a sidebar with a logo and navigation links: "Home", "Service", "About", "Contact", "Admin" (with a profile picture), and a "Menu" section containing "Projects" (which is selected), "Clients", and "Edit". The main content area displays three user profiles in cards:

- User: Bishal**
 - Business Type
 - Modern Design
 - Restaurant
 - Tier 2 - Gold (+15%)
 - 2 months - Express delivery (+30%)
 - Home page
 - Gallery page
 - FAQ page
 - Contact page
 - About page[View](#)
- User: David**
 - Ecommerce Type
 - Artistic Design
 - Electronic Store
 - Tier 3 - Platinum (30%)
 - 1 Month Expedited Delivery
 - Gallery page
 - Home page
 - Contact page
 - T&C page
 - eCommerce Pages[View](#)
- User: Alex**
 - Personal Type
 - Modern Design
 - Personal Portfolio
 - Tier 1 - Standard
 - 3 Months Standard Delivery
 - Gallery page
 - Home page
 - Contact page
 - T&C page[View](#)

Admin Panel: Customers Page

Clients

User ID	First Name	Last Name	Date of Birth	Country	Email
1	Bishal Kumar	Shah	1256-10-21	Australia	shah.bishal9876@gmail.com
2	Alex	TestLastName	1995-10-31	Nepal	abc@123.com
3	Bishal	Shah	1998-10-21	Nepal	bishal.shah2344@gmail.com
4	David	Tamang	1990-10-21	Australia	asdf@gmail.com
5	Craig	Carter	1997-12-12	Australia	test@gmail.com
6	Michelle	Smith	1994-10-19	Australia	Michelle5@gmail.com

Database creation and integration

admins

admin_id	first_name	last_name	dob	country	email	password
25	Bishal Kumar	Shah	1997-10-21	Australia	Bishal.shah2344@gmail.com	\$2y\$10\$saWJ6VzsFQFZjYPDTk4t
27	Bishal Kumar	Shah	1998-02-21	Australia	shah.bishal9876@gmail.com	\$2y\$10\$ShJiR0p8lGqH1oe50gPobu

customer_data

invoice_id	user_id	user_data_json	pdf_url
10001	2	{"question1": {"name": "question1"}, ...}	https://pub-cdn.apitemplate.io/2023/10/87d62879-e2...
10002	4	{"question1": {"name": "question1"}, ...}	https://pub-cdn.apitemplate.io/2023/10/87d62879-e2...
10009	3	{"question1": {"name": "question1"}, "question": "Websi... https://pub-cdn.apitemplate.io/2023/10/9d85f0e9-53...	
10010	3	{"question1": {"name": "question1"}, "question": "Websi... https://pub-cdn.apitemplate.io/2023/10/59d2c215-fe...	

users

user_id	first_name	last_name	dob	country	email	password
1	Bishal Kumar	Shah	1256-10-21	Australia	shah.bishal9876@gmail.com	\$2y\$10\$HjRtxG9Tdt3yUQpf
2	Alex	TestLastName	1995-10-31	Nepal	abc@123.com	\$2y\$10\$5jZRE13wm59FqK7
3	Bishal	Shah	1998-10-21	Nepal	bishal.shah2344@gmail.com	\$2y\$10\$VQOs3yOSgk688U;
4	David	Tamang	1990-10-21	Australia	asdf@gmail.com	\$2y\$10\$JN7G9IRFUJbsv9bc
5	Craig	Carter	1997-12-12	Australia	test@gmail.com	\$2y\$10\$IDafuEaIX.aOzVA3F
6	Michelle	Smith	1994-10-19	Australia	Michelle5@gmail.com	sdfsafasfsdfgdfghdrge

Homepage



[Home](#) [Service](#) [About](#) [Contact](#) [Log in](#)

#Sale information

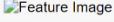
Create Your Perfect Website

We will help you create your website for small business, eCommerce or personal purpose, with our unique user experience. Our user experience is designed to fulfill all your website requirements.

Get started today

[Let's go](#)

Why Choose Pristine?



Personalized Approach
We understand that each local business is distinct, with specific needs and goals. Our innovative process starts by asking you a series of carefully crafted questions to understand your business inside out.

Seamless Experience
No technical expertise required! Our user-friendly platform guides you through the entire website-building process, step-by-step. You'll be amazed at how enjoyable and stress-free it can be.

Stunning Designs
Impress your customers with visually appealing and modern designs. Choose from a wide selection of templates tailored for plumbing, electrician, tinting, and many other local business niches.

Blog and E-commerce Ready
Whether you want to share industry insights through a blog or expand your reach with an online store, we've got you covered. Our websites seamlessly integrate with blogs and e-commerce functionalities.

Local SEO Optimization
Reach your local audience effectively with our SEO optimization. Your website will be designed to rank higher in local search results, helping potential customers find you easily.



How It Works

Questionnaire
Tell us about your business, your goals, and your target audience through our comprehensive questionnaire.

Customization
Choose from a variety of industry-specific web elements and customize your website's look and feel to match your brand identity.

Content Creation
Don't worry about writing content from scratch. We offer content writing and SEO optimization based on your questionnaire and suggestions.

Review & Launch
Take a final look at your masterpiece, provide feedback, and when you're ready, let's Launch the site!

Contact page

The screenshot shows a contact form titled "Contact Us". The form fields include "First Name" (John), "Last Name" (Doe), "Phone No." (0426879007), "Country" (Australia), "Email" (Data.Validation@works.com), and a "Message" box containing "This is a test message.". A "Submit" button is at the bottom. The top navigation bar has links for Home, Service, About, Contact, and a red "Log in" button.

Secured access

Secured user access and password storage for signup and login using Bcrypt Hashing algorithm

The screenshot shows the "users" table in the "user_account" database. The table contains the following data:

	id	first_name	last_name	dob	country	email	password
<input type="checkbox"/>	24	Bishal	Kumar	Shah	1154-10-21	Australia	sTeshah.bishal9876@gmail.com
<input type="checkbox"/>	26	Bishal	Kumar	Shah	4040-12-24	Australia	shasadash.bishal9876@gmail.com
<input type="checkbox"/>	27	Bishal	Kumar	Shah	2000-05-23	Australia	shah.bishal9876@gmail.com
<input type="checkbox"/>	28	Bradley		Shah	2000-02-23	Australia	Bishal.shah2344@gmail.com
<input type="checkbox"/>	29	alex		chan	1997-11-21	Nepal	abcd@gmail.com
<input type="checkbox"/>	30	Bishal	Kumar	Shah	2023-08-17	Australia	shah.bishal9876@gmail.com
<input type="checkbox"/>	31	John	Doe	Shah	1996-10-21	Nepal	john.doe123@gmail.com

A note on the right side of the table reads: "Bcrypt hashing algorithm for security".

Deployment Testing

Deployment of statics web pages in Azure Web app service for deployment testing.

A PristineWeb - Microsoft Azure x +

portal.azure.com/#@federationuniversity.onmicrosoft.com/resource/subscriptions/d4ee94c0-d422-4cd6-8...

Microsoft Azure Search resources, services, and docs (G+) bishalkumars@students... FEDERATION UNIVERSITY AUSTRALIA

Home > PristineWeb Web App

Search Browse Stop Swap Restart Delete Refresh Download publish profile Reset publish profile Share to mobile ...

Overview Activity log Access control (IAM) Tags Diagnose and solve problems Microsoft Defender for Cloud Events (preview)

Deployment Deployment slots Deployment Center

Settings Configuration Authentication

Essentials

Resource group (move) : Resource_Group	Default domain : pristineweb.azurewebsites.net
Status : Running	App Service Plan : ASP-PristineWeb-b0d2 (B1: 1)
Location (move) : Australia Southeast	Operating System : Linux
Subscription (move) : Azure for Students	Health Check : Not Configured
Subscription ID : d4ee94c0-d422-4cd6-8b28-fe3f51eb1ed3	GitHub Project : https://github.com/Bishal23441/RedTeamProject
Tags (edit) : Student-Research : ITECH3009	

Properties Monitoring Logs Capabilities Notifications Recommendations

Web app

Name : PristineWeb
Publishing model : Code
Runtime Stack : PHP - 8.1

JSON View

A PristineWeb - Microsoft Azure x +

pristineweb.azurewebsites.net

pristineweb.azurewebsites.net Connection is secure Cookies and site data Site settings About this page

#Sale information

Create Your Perfect Website

We will help you create your website for small business, eCommerce or personal purpose, with our unique user experience. Our user experience is designed to fulfill all your website requirements.

Get started today

Let's go



Questionnaire Design Document – Attached

This is a design document of main section of price-based web component and questionnaire which includes:

- Questionnaire structure
- Researched Price Estimation
- Question logic
- Example cases

Screenshot

Questionnaire with estimated price:

What type of website is it?

- Business - \$1500
- Personal - \$500
- E-commerce - \$3000

What pages do you need?

- Home page (Mandatory) - \$300
- Contact page - \$200
- About page - \$200
- Contact + About page in one - \$300
- Gallery page - \$300
- Content pages (Informational page such as Articles, news, blogs or any specific topic information) - \$200
 - o Name + Description input box
- Content Management System - \$200
 - o Admin dashboard to add contents in website
 - o Personalized user guide and training included.
- Account pages
 - o Login/Sign up pages: - \$300
 - 1. Login page and Sign up page
 - 2. Login + Sign up in one page
 - 3. Login/signup directly using pop-up from any page of website
 - (One selection is mandatory between option 1,2 and 3)
 - o Login pop-up from homepage (applies for option 1 or 2 only) - \$75
 - o Option to fast Login from other account
 - 1. Google account - \$80
 - 2. Facebook account - \$80

Well-structured and commented codebase

The screenshot shows a code editor interface with the following details:

- File Menu:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Search Bar:** Red_Project_Pristine_Web.
- Explorer:** Shows the project structure under "RED_PROJECT_PRISTINE_WEB".
 - css:** contact_style.css, dashboard_style.css, home_style.css, login_style.css, signup_style.css.
 - dashboard:** input_logic.js, input_style.css, jquery.min.js, questionnaire1CssVariables.js.
 - JS:** admin_dashboard_1.php, admin_dashboard_2.php, submit_result.php (selected), user_dashboard_1.php, user_dashboard_2.php, website_card.php.
 - HTML:** dashboard_v0.1, images, about.html, admin.dashboard.php, admin_login_signup.php, admin_logout.php, contact.html, dashboard.php, index.html, login.php, logout.php.
 - Others:** README, send_email.php, session.php, signup.php.
- Editor Area:** Displays the content of "submit_result.php". The code is well-commented and follows best practices for database interaction and security.
- Bottom Status Bar:** Shows the current branch (master), file count (0), and other status indicators.

```
<?php
include '../session.php';

$resultData = $_POST['resultData'];
$userID = $_SESSION['user_id'];

//For testing - create a file and save resultData to it
/*
$file = fopen("resultData.txt", "w");
fwrite($file, $resultData);
fclose($file);
*/

//save file to database
$servername = "localhost";
$username = "root";
$password = "";
$database = "pristine_web_iTech_db";

// Establish database connection
$conn = mysqli_connect($servername, $username, $password, $database);
// Prepare the SQL statement with a parameter placeholder
$query = "INSERT INTO customer_data (user_id, user_data_json) VALUES (?, ?)";
$stmt = mysqli_prepare($conn, $query); // Prepare the statement
if ($stmt) {
    mysqli_stmt_bind_param($stmt, "ss", $userID, $resultData); // Bind the parameters
    mysqli_stmt_execute($stmt); // Execute the statement
    // Get the invoice ID
    $invoiceID = mysqli_insert_id($conn);
    if (mysqli_stmt_error($stmt)) {
        echo "Error: " . mysqli_stmt_error($stmt); // Check for errors
    } else {
        // Log the invoice ID
        echo "<script>console.log('Saved to database successfully. Invoice ID: $invoiceID');</script>";
    }
    mysqli_stmt_close($stmt); // Close the statement
} else {
    echo "Error preparing SQL statement: " . mysqli_error($conn);
}

//add invoice id to resultData
$resultData = json_decode($resultData, true);
$resultData['invoice_id'] = $invoiceID;
//retrieve email from json
$js_email = $resultData['elements'][15]['answer'];
$resultData = json_encode($resultData);
```

5 Sprint boards with feature tracking and test cases for future development

The image shows a digital interface for managing a software development project. It features five horizontal boards, each representing a different stage of the development process:

- Backlog:** Contains user stories and requirements. One story is labeled "As a Admin, I need to access the blueprint document containing design, structure and price estimation of the questionnaire so that I can evaluate, review and give feedback on the design and development of User dashboard and refer in future development." It includes priority levels (Urgent, Large) and a user story button.
- Todo:** Contains two draft tasks: "Complete Sprint 4 artefacts and status report" and "Complete individual contribution report".
- In progress:** Contains two draft tasks: "Complete Sprint 4 artefacts and status report" and "Complete individual contribution report".
- In review:** Contains no visible items.
- Done:** Contains several completed tasks, each with a checkmark icon. Examples include "Research and study on programming the admin panel and selection of appropriate back-end platform", "Learn PHP advanced concept for database CRUD operations", and "Create blueprint document of question design and structure". These tasks also have priority levels (Urgent, Large) and a todo task button.

A detailed view of a single issue card for a user story titled "Use PHP code on the server-side to sent message via email in contact page." #68. The card includes the following details:

- Project:** project-repo-red_ecommerce-store-for-websites
- Status:** Open (Bishal23441 opened 2 weeks ago)
- Description:** As a user, I want to access the website from the internet using cloud provider such as Azure so that I can easily navigate through the platform and utilize the services and functions provided by the web project.
- Table:** A table showing field mappings:

Field	Data type	Pass Test data	Fail Test data
First name	String	John	-
Last name	String	Doe	-
Phone No.	6-10 digit no.	0426587125	12345
Country	String	Australia	-
Email	String@String format	abc@gmail.com	Abcgmail.com
Message	String	Hi, how are you?	-
- Test status:** passed
- Buttons:** A smiley face icon and a close button.
- Right sidebar:** Includes fields for Status (Done), Linked pull requests (No linked pull requests), Repository (project-repo-red_ecommerce-stor...), Priority (Medium), Size (Choose an option...), and a list of actions: Open in new tab, Copy link, Copy link in project, Archive, and Delete from project.

About 45 Commits for version control and feature tracking to help in future development

Commits

⌚ 45 commits

master

Commits on Oct 13, 2023

- Update README
Bishal23441 committed 2 hours ago Verified e3e529b 🔗
- Bug fixed - \$db_pdf_url, email copy added, Tested
Bishal23441 committed 2 hours ago 🔗 cfc9bd55 🔗
- Admin_dashboard_v2
Bishal23441 committed 15 hours ago 🔗 4d2c54f 🔗

Commits on Oct 12, 2023

- Till User Dashboard
Bishal23441 committed yesterday 🔗 9369fab 🔗
- pdf attachment api update
Bishal23441 committed yesterday 🔗 a347384 🔗

Commits on Oct 11, 2023

- pdf creation completion
Bishal23441 committed 2 days ago 🔗 91441bb 🔗
- Till pdf creation major update v2
Bishal23441 committed 3 days ago 🔗 077da26 🔗

Commits on Oct 7, 2023

- Major_PDF_Integration_JinJa2
Bishal23441 committed last week 🔗 6bd5c6b 🔗

Commits on Sep 29, 2023

- v3.1.1
Bishal23441 committed 2 weeks ago 🔗 82a2865 🔗

Test Summary Report

This test summary report provides an overview of the testing efforts and results of the website. The purpose of this testing was primarily to evaluate the functionality, usability, and performance of the questionnaires and JSON data validation.

Questionnaire Data Validation

The content of the price-based questionnaire is located in the file:

source_code/dashboard/user_dashboard1.php

The following properties can be found the file inside the questionnaire2JSON object which can be used to change the question details:

- title
- description
- value
- text

```
<script>
const questionnaire2Json = {
    logoPosition: "right",
    completedHtml: "<h3>Thank you for completing the questionnaire</h3>",
    completedBeforeHtml: "<h3>Our records show that you have completed the questionnaire</h3>",
    pages: [
        {
            name: "page1",
            elements: [
                {
                    type: "radiogroup",
                    name: "question1",
                    useDisplayValuesInDynamicTexts: false,
                    title: "Website Type",
                   isRequired: true,
                    description: "What type of website is it?",
                    choices: [
                        {
                            value: "1500",
                            text: "Business",
                        },
                        {
                            value: "500",
                            text: "Personal",
                        },
                        {
                            value: "3000",
                            text: "E-commerce",
                        },
                    ],
                },
            ],
        },
    ],
};
```

A Question for reference

1. Website Type *

What type of website is it?

- Business
- Personal
- E-commerce

Property	Details
<i>title</i>	Question title
<i>description</i>	Question description
<i>value</i>	Price of choices in the question
<i>text</i>	Choices of the questions

You can change these properties and save the file to update the questionnaires.

Data Validation Summary

Properties	Description	Summary
<i>name</i>	ID of the question	Must be unique
<i>title</i>	Title of the question	String data type
<i>isRequired</i>	Property to make question mandatory	Boolean type
<i>description</i>	Sub-text of the question	String data type
<i>choices</i>	Container for question's choices	Array of objects
<i>choices.value</i>	Price assigned to the choice	Must be unique within a question.
	Used to calculate total price	Float data type
<i>choices.text</i>	Choice of a question	String data type

Note: Do not use the exact same 'value' property for the choices within a single question, as 'value' property must be unique. As the 'value' property is used to calculate the total price, there can be instances where prices can be same for different choices within a question, in which case, these values can be modified to vary in 3 decimal points. For instance,

Property	Values
value of choice 1	300.001
value of choice 2	300.002
value of choice 3	300.003

The total prices calculated will still be accurate as the last 2 decimal points will be bypassed through a Function to produce the total price.

Test cases for the questionnaire

Test Case ID	Description	Steps	Expected Result	Actual Result	Status
TC-001	Validate multiple choice question type	1. Create a multiple choice question.	The question should have options to choose.	Pass/Fail	Pass
TC-002	Validate open-ended question type	1. Create an open-ended question.	The question should allow free text input.	Pass/Fail	Pass
TC-003	Validate conditional logic	1. Set up a question with conditional logic.	Based on the condition, the logic should work.	Pass/Fail	Pass
TC-004	Validate data storage	1. Submit a survey with various responses.	Data should be stored accurately.	Pass/Fail	Pass
TC-005	Validate error handling for invalid input	1. Input invalid data into a question.	Proper error messages should be displayed.	Pass/Fail	Pass
TC-006	Evaluate performance under normal load	1. Load a survey form with a typical number of simultaneous users.	The form should load within an acceptable time frame.	Pass/Fail	Pass
TC-007	Evaluate performance under high load	1. Load a survey form with a large number of simultaneous users.	The form should handle the load without crashing or significant performance degradation.	Pass/Fail	Pass

Questionnaire Navigation logic flow and expected behaviour

Step	User Action	Expected Behavior	Pass Data
1	Start the questionnaire	User is presented with the first question asking about the website type.	-
2	Choose "Business" as the website type	The questionnaire proceeds to the "Pages" section.	Business
3	Select "Home page", "Contact page", "About page"	The questionnaire proceeds to the "Content Page" section.	Home, Contact, About
4	Choose "Add Page" in the "Content Page" section	The questionnaire adds a content page and allows the user to enter custom page names and descriptions.	Add Page, Content Page
5	Choose "CMS admin dashboard page" for "Content Management"	The questionnaire proceeds to the "Account Pages" section.	CMS admin dashboard page
6	Select "Yes" for "Account Pages"	The questionnaire proceeds to the "Express Login" section.	Yes
7	Select "Google account" for "Express Login"	The questionnaire proceeds to the "Interactive Map" section.	Google account
8	Select "Yes" for "Interactive Map"	The questionnaire proceeds to the "Booking System" section.	Yes
9	Choose "Credit and debit card", " <u>Paypal</u> " for "Payment accepting and checkout methods"	The questionnaire proceeds to the "Express Checkout" section.	Credit and debit card, <u>Paypal</u>
10	Select " <u>Paypal</u> Express Checkout" for "Express Checkout"	The questionnaire proceeds to the "SSL Certification" section.	<u>Paypal</u> Express Checkout
11	Select "Yes" for "Newsletter feature"	The questionnaire proceeds to the "Design Package" section.	Yes
12	Choose "Tier 1 (Standard)" for "Design Package"	The questionnaire proceeds to the "SSL Certification" section.	Tier 1 (Standard)
13	Choose "DV Certificate" for "SSL Certification"	The questionnaire proceeds to the "Website delivery date" section.	DV Certificate
14	Choose "2 months - Express delivery (+30%)" for "Website delivery date"	The questionnaire is complete and provides the total price based on the selections.	2 months - Express delivery (+30%)

Sign-up Data Validation

These are the fields that are accepted by the forms while signing up in both end-user and admin cases.

Field	Data type	Pass Test data	Fail Test data
First name	String	John	-
Last name	String	Doe	-
Phone No.	<u>6-10 digit no.</u>	0426587125	12345
Country	String	Australia	-
Email	<u>String@String</u> format	abc@gmail.com	Abcgmail.com
Message	String	Hi, how are you?	-

Test Cases for submit_result.php Actions

Test Case 1: Database Login

Description: Verify that the **submit_result.php** successfully logs into the database.

Steps:

1. Execute the **submit_result.php** script.

Expected Result:

- The script should establish a successful database connection.

Actual Result:

- The script successfully connected to the database.

Status: Pass

Test Case 2: Data Storage in customer_data Table

Description: Ensure that the captured result data is stored in the **customer_data** table along with the **user_id**.

Steps:

1. Capture sample result data.
2. Execute the **submit_result.php** script.

Expected Result:

- The script should store the captured result data in the **customer_data** table associated with the **user_id**.

Actual Result:

- The script successfully stored the result data in the **customer_data** table.

Status: Pass

Test Case 3: Retrieval of Invoice ID

Description: Verify that the script retrieves the **invoice_id** from the auto-generated database field.

Steps:

1. Execute the **submit_result.php** script after storing result data.

Expected Result:

- The script should retrieve the auto-generated invoice_id from the database.

Actual Result:

- The script successfully retrieved the invoice_id.

Status: Pass

Test Case 4: API Call to PDF Generator Endpoint

Description: Ensure that the script correctly makes an API call to the PDF Generator RESTful endpoint and receives a PDF URL in response.

Steps:

1. Execute the **submit_result.php** script after retrieving the invoice_id.

Expected Result:

- The script should successfully call the PDF Generator endpoint and receive a PDF URL in the HTTP response.

Actual Result:

- The script made a successful API call and received the PDF URL.

Status: Pass

Test Case 5: PDF Download and Encoding to Base64

Description: Verify that the script downloads the PDF from the provided URL and encodes it to base64 format.

Steps:

1. Execute the **submit_result.php** script after receiving the PDF URL.

Expected Result:

- The script should download the PDF and encode it to base64 format.

Actual Result:

- The script successfully downloaded the PDF and encoded it to base64 format.

Status: Pass

Test Case 6: API Request to SendGrid with PDF Attachment

Description: Ensure that the script makes an API request to SendGrid with the PDF attached and sends emails to Admin and User.

Steps:

1. Execute the **submit_result.php** script after encoding the PDF.

Expected Result:

- The script should make an API request to SendGrid with the PDF attached and send emails to Admin and User.

Actual Result:

- The script successfully made the API request to SendGrid and sent the emails with the PDF attachment.

Status: Pass

Report Summary for submit_result.php

The testing primarily focused on validating the actions executed by the **submit_result.php** script, including database login, data storage, retrieval of invoice ID, API call to PDF Generator endpoint, PDF download and encoding, and API request to SendGrid.

All the test cases were executed successfully, and the **submit_result.php** script performed the intended actions as described.

The **submit_result.php** script has been found to execute the specified actions accurately and efficiently, ensuring successful storage of captured result data, retrieval of invoice ID, API calls to relevant endpoints, PDF generation, and email sending via SendGrid.

Recommendations

- Continue monitoring the integration with external APIs to ensure consistent functionality.
- Implement a proper error handling to handle potential failures during database access, API calls, and PDF generation during feature modification.