| Name: | Tham Yeng Kitt (Joseph) |  | Date: | 05/06/2025 |
| --- | --- | --- | --- | --- |

| General. |
| --- |

1. What version of .NET have you worked with?

| **.NET Core / .NET Version** | **Years of Experience** |
| --- | --- |
| .Net Framework 4.7 | 5 years |
| 6.0 | 2 years+ |
| 8.0 | Less than a year |
|  |  |
|  |  |

2. What is the current .NET version that you are working on?

.Net Core 6.0 mainly

3. What sort of .NET technology have you worked on?

| **Name** | **VB/C#** | **Skill Level**  **(Basic/Intermediate/Advance)** | **Years of Experience** |
| --- | --- | --- | --- |
| ASP.NET Core – Web | C# | Intermediate | 2 years+ |
| .NET Core – Desktop | C# | Virtually Basic to None | Less than a year |
| ASP.NET - Webform | - | - | - |
| ASP.NET - MVC | C# | Intermediate | 3 years+ |
| Windows Form Application | VB6 | Basic - Intermediate | 1 year+ |
| Console Application | C# | Basic - Intermediate | 1 year |
| Windows Service | C# | Intermediate | 3 years |
| WCF Service | - | - | - |
| Web API 2.0 and above | C# | Basic - Intermediate | 1 year+ |
| LINQ to SQL | C# | Basic | Less than a year |
| Entity Framework | C# | Intermediate | 3 years+ |
| Blazor | C# | Intermediate | 2 years+ |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

*\* Please include the technology that are not listed here as well*

4. What are some of the revision source control system that you have used?

| **Name** | **Skill Level**  **(Basic/Intermediate/Advance)** | **Years of Experience** |
| --- | --- | --- |
| CVS | - | - |
| SVN | - | - |
| Git | Basic - Intermediate | 2 years+ |
| TFS | Basic | 3 years+ |
| SourceSafe | - | - |
|  |  |  |
|  |  |  |
|  |  |  |

*\* Please include the revision source control system that are not listed here as well*

5. What kind of database/caching systems (all the versions) that you have worked with?

| **Name** | **Release/Version** | **Skill Level**  **(Basic/Intermediate/Advance)** | **Years of Experience** |
| --- | --- | --- | --- |
| Microsoft SQL Server | 2008,2012,2016 | Basic - Intermediate | 5 years+ |
| MySQL |  |  |  |
| PostgreSQL | - | Basic | Less than a year |
| Oracle RDBMS |  |  |  |
| SQLite | Version 3 | Basic | Less than a year |
| DynamoDB |  |  |  |
| MongoDB |  |  |  |
| Redis |  |  |  |
| Amazon ElastiCache |  |  |  |
| Oracle Tom Cat | - | Basic | Less than a year |
| Sql Anywhere | Version 17 | Basic - Intermediate | 2 years+ |
|  |  |  |  |
|  |  |  |  |

*\* Please include the database/caching system that are not listed here as well*

6. What sort of client-side technologies have you worked on?

| **Name** | **Release/Version** | **Skill Level**  **(Basic/Intermediate/Advance)** | **Years of Experience** |
| --- | --- | --- | --- |
| AngularJS |  |  |  |
| Angular |  |  |  |
| Knockout |  |  |  |
| React |  |  |  |
| Bootstrap | 5 | Intermediate | 5 years+ |
| Foundation |  |  |  |
| VueJS |  |  |  |
| Blazor | .Net 6 and .Net 8 | Basic - Intermediate | 2 years + |

*\* Please include the other client-side technologies that are not listed here as well*

7. Do you have knowledge or have worked on the following?

| **Name** | **Skill Level**  **(Basic/Intermediate/Advance)** | **Years of Experience** |
| --- | --- | --- |
| Node.Js |  |  |
| jQuery | Intermediate | 3 years+ |
| JavaScript | Basic | 3 years+ |
| MS SQL - Transact SQL | Basic - Intermediate | 3 years+ |
| Amazon Web Service (AWS) |  |  |
| Microsoft Azure | Basic | Less than a year |
| Google Cloud |  |  |
| Alibaba Cloud |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

*\* Please include the knowledge that are not listed here as well*

8. What are some of the unit test frameworks/tools that you have used?

| **Name** | **Skill Level**  **(Basic/Intermediate/Advance)** |
| --- | --- |
| NUnit |  |
| MSTest |  |
| Selenium |  |
| Postman | Basic - Intermediate |
| Runscope |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

*\* Please include the unit test frameworks that are not listed here as well*

7. What kind of continuous integration systems (all the versions) that you have worked with?

| **Name** | **Release/Version** | **Skill Level**  **(Basic/Intermediate/Advance)** |
| --- | --- | --- |
| Jenkins |  |  |
| CodePipeline |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

*\* Please include the continuous integration systems that are not listed here as well*

**Code and submit the full solution with source codes.**

***This assessment contains 1 Tasks only with 3 basic and 2 advance questions.***

***Candidate must complete the basic sections.***

***Candidate able to complete the advance section as well will be ranked higher.***

***Code and submit the full solution with source codes.***

*Grading criteria:*

1. *Correctness*

*• Comprehension - understand the requirement and produce code accordingly.*

*• Bug free - the coding is done without logic bugs.*

*• Performant - able to execute within acceptable timeframe.*

1. *Style*

*• Code formatting - spaces, indentation, braces, new lines.*

*• Naming convention - variables, methods, classes are named meaningfully.*

*• Readability - clear and organized logic flow, easy to read and maintain by others.*

1. *Comprehensiveness*

*• Methods/Functions/Classes/Designs are flexible; cater for reusability and future extension.*

*• Has proper validation and able to handle case scenarios.*

*• Has unit test(s), if possible.*

# **Task:**

**Question 1 (Basic):**

Use **.NET 6.0 or later** to develop a REST API function which receives transaction information from **allowed Partner** to further process the transaction information. This request will return a success / failed result with result messages back to **allowed Partner**.

| **Partner No** | **Allowed Partner** | **Partner’s Password** |
| --- | --- | --- |
| FG-00001 | FAKEGOOGLE | FAKEPASSWORD1234 |
| FG-00002 | FAKEPEOPLE | FAKEPASSWORD4578 |

## API Specifications

| **Receiving Component** | [Domain]/api/submittrxmessage |
| --- | --- |
| **Purpose** | To receive transaction from partner for further processing |
| **Type of Interface** | HTTP POST to RESTful API (JSON) |
| **Frequency (or periodicity)** | On demand |
| **Triggering Event** | Triggered by partner site or application |

## Request Message

### **body**

This object is used to describe the transaction details of the vertical partner

| **No** | **Field Name** | **Type** | **Size** | **Req** | **Description** |
| --- | --- | --- | --- | --- | --- |
| **1** | partnerkey | String | 50 | M | The allowed partner's key |
| **2** | partnerrefno | String | 50 | M | Partner's reference number for this transaction  **Sample:**   * FG-00001 |
| **3** | partnerpassword | String | 50 | M | Encode to **Base64** format. |
| **4** | totalamount | Long | - | M | Total amount of payment include discount in the MYR only.  **Business Logic:**   * Only allow positive value * value in cents   Example: 1000 = MYR 10.00 |
| **5** | items | Array of **itemdetail** | - | O | Array of items purchased through this transaction |
| **6** | timestamp | String |  | M | String representation of the UTC time of the operation in ISO 8601 format ie. **2024-08-15T02:11:22.0000000Z** |
| **7** | sig |  |  | M | **Message Signature**. Sort the parameters alphabetically by their name, except for timestamp and sig.  sigtimestamp format are **yyyyMMddHHmmss**  Concatenate all parameter values into a single string in the specified order, encode it using **SHA-256** with **UTF-8** to produce a **lowercase hexadecimal hash**, and then convert the resulting hash to **Base64** using **UTF-8 encoding**. |

### **itemdetail definition (items)**

This object is used to describe the items information

| **No** | **Field Name** | **Type** | **Size** | **Req** | **Description** |
| --- | --- | --- | --- | --- | --- |
| **1** | partneritemref | String | 50 | M | Reference ID the partner uses for this item  **Business Logic:**   * Cannot be null or empty |
| **2** | name | String | 100 | M | Name of the item  **Business Logic:**   * Cannot be null or empty |
| **3** | qty | Integer | - | M | Quantity of the item bought |
| **4** | unitprice | Long | - | M | Price of one unit of the item in the currency of the transaction  **Business Logic:**   * Only allow positive value * Quantity must not exceed 5 * value in cents * Example: 1000 = MYR 10.00 |

## Response Message

### **body**

| **No** | **Field Name** | **Type** | **Size** | **Req** | **Description** |
| --- | --- | --- | --- | --- | --- |
| **1** | result | Integer | - | M | Whether the operation was successful or not.   * 1 is successful, * 0 if errors encountered. |
| **2** | totalamount | Long | - | O | Total amount of payment include discount in the MYR only.  **Business Logic:**   * Only allow positive value * value in cents   Example: 1000 = MYR 10.00 |
| **3** | totaldiscount | Long | - | O | Total discount of payment in the MYR only.  **Business Logic:**   * Only allow positive value * value in cents   Example: 1000 = MYR 10.00 |
| **4** | finalamount | Long | - | O | Final amount that customer paid in the MYR only.  **Business Logic:**   * Only allow positive value * value in cents   Example: 1000 = MYR 10.00 |
| **5** | resultmessage | String | - | O | Result message if the operation was Failure |

**Sample Request**

{

"partnerkey": "FAKEGOOGLE",

"partnerrefno": "FG-00001",

"partnerpassword": **"**RkFLRVBBU1NXT1JEMTIzNA==",

"totalamount": 1000,

"items": [

{

"partneritemref": "i-00001",

"name": "Pen",

"qty": 4,

"unitprice": 200

},

{

"partneritemref": "i-00002",

"name": "Ruler",

"qty": 2,

"unitprice": 100

}

],

"timestamp": "2024-08-15T02:11:22.0000000Z",

"sig": " MDE3ZTBkODg4ZDNhYzU0ZDBlZWRmNmU2NmUyOWRhZWU4Y2M1NzQ1OTIzZGRjYTc1ZGNjOTkwYzg2MWJlMDExMw=="

}

**Sample Response (Success)**

{"result":1,"totalamount":1000,"totaldiscount":0,"finalamount":1000}

**Sample Response (Failed)**

{"result":0,"resultmessage":"Access Denied!"}

-------------------------------------------------------------------------

To generate the signature output based on instruction:

1. **Message Signature Parameter Order**

timestamp + partnerkey + partnerrefno + totalamount + partnerpassword(encoded)

Example: 20240815021122FAKEGOOGLEFG-000011000RkFLRVBBU1NXT1JEMTIzNA==

1. **Apply SHA-256 hashing** **(UTF-8 encoded input)(lowercase hexadecimal hash output)**

017e0d888d3ac54d0eedf6e66e29daee8cc5745923ddca75dcc990c861be0113

1. **Convert the SHA-256 hash to Base64 (UTF-8 encoding)**

MDE3ZTBkODg4ZDNhYzU0ZDBlZWRmNmU2NmUyOWRhZWU4Y2M1NzQ1OTIzZGRjYTc1ZGNjOTkwYzg2MWJlMDExMw==

-------------------------------------------------------------------------

This API act as a middleware to validate the payload data only.

* 1. Ensure all fields name matches the API Specifications above
  2. All fields must be validated before executing business logic
     1. Example:
        1. Mandatory column cannot be NULL or not provided
        2. Positive value cannot accept **negative** value

**NOTE: Read the conditions carefully for each field**

-------------------------------------------------------------------------

**Question 2 (Basic):**

Apply the validation against the fields in the request and return them in error code. Below is an example error result messages we are looking for.

| **No** | **Error : Result Message** | **Description** |
| --- | --- | --- |
|  | Access Denied! | Unauthorized partner or Signature Mismatch |
|  | Invalid Total Amount. | Only applicable when **itemDetails** is provided.  The total value stated in **itemDetails** array not equal to value in **totalamount**. |
|  | Expired. | Provided timestamp exceed server time +-5min  Example: server time is 13 Dec 2021 15:04:02  The valid time will be +-5 Min of the server time.  Other than this will consider the API is expired. |
|  | **##ParamName##** is Required. | Mandatory Parameter is not provided  Example: partnerrefno is required. |
|  | **##Any Other Possible Messages##** | If you able detect more validation, feel free to add in and provide your own error messages. |

-------------------------------------------------------------------------

**Question 3 (Basic):**

In continuation of the **Question 1** API, you are tasked that calculates the final amount to be paid after applying several discount rules based on the **totalamount**. The business rules for applying discounts are as follows:

**Discount Rules:**

Base Discount:

If **totalamount** is less than MYR 200: No discount is applied.

If **totalamount** is between MYR 200 and MYR 500 (inclusive): Apply a 5% discount.

If **totalamount** is between MYR 501 and MYR 800 (inclusive): Apply a 7% discount.

If **totalAmount** is between MYR 801 and MYR 1200 (inclusive): Apply a 10% discount.

If **totalAmount** is greater than MYR 1200: Apply a 15% discount.

Conditional Discounts:

If **totalAmount** is a prime number above MYR 500: Apply an additional 8% discount.

If **totalAmount** ends in the digit 5 and is above MYR 900: Apply an additional 10% discount.

Cap on Maximum Discount:

The total discount (base + conditional) cannot exceed 20% of the **totalamount**.

**Sample calculation 1:**

**totalamount: 100000**

Base Discount:

Meets the condition of being within MYR 801 and MYR 1200: **10% discount**

Conditional Discounts:

None of the condition are met.

Cap on Maximum Discount:

The total discount is 10%, which is below the 20% threshold, so a 10% discount will be applied.

**Total Discount**: **RM1000 x 10% = RM100**

**Your Sample 1 Response will be**

{"result":1,"totalamount":100000,"totaldiscount":10000,"finalamount":90000}

**Sample calculation 2:**

**totalamount: 120500**

Base Discount:

Meets the condition of being greater than MYR 1200: **15% discount**

Conditional Discounts:

Meets the condition of ending in the digit 5 and being above MYR 900: **additional 10% discount**

Cap on Maximum Discount:

The total discount is 25%, which is above the 20% threshold, so a 20% discount will be applied.

**Total Discount**: **RM1205 x 20% = RM241**

**Your Sample 1 Response will be**

{"result":1,"totalamount":120500,"totaldiscount":24100,"finalamount":96400}

-------------------------------------------------------------------------

**Question 4 (Advance):**

In continuation of the **Basic** scenarios, saved all the request body, response body and possible logs in text file to ease future troubleshooting **using log4net.** Ensure that passwords in the log file are encrypted for security.

-------------------------------------------------------------------------

**Question 5 (Advance):**

Containerizing the application using **Docker** and provide Docker File.

-------------------------------------------------------------------------