Company
Bhuvanlabs is an IT Consulting company.

Application

- The company has developed a series of applications and services.
- The Application and services are developed in ASP.Net Core.
- An application called “BhuvanlabsPolicyService” is deployed to Azure Web App Service.
- The “BhuvanlabsPolicyService” application needs to react to events from Azure Event Grid and perform required actions
- The “BhuvanlabsPolicyService” application must include the Event Grid Event ID field in all Application Insights telemetry.
- The “BhuvanlabsPolicyService” application must process all sign-in and sign-out events. The sign-out process must be processed as soon as possible.
- All applications must write logs to BLOB storage and logs must remain there for 15 days.
- A shared library named BhuvanlabsLib contains all common functionality for the ASP.NET Core web services and application.
- The company has an anomaly detection service that analyses all log information for anomalies.
- The anomaly detection service is implemented as a Machine Learning Model.
- If an anomaly is detected, an Azure Function that emails administrators is called by using an HTTP WebHook.

Code snippets for various code modules are displayed below

EventGridController.cs

```
public class EventGridController : Controller
{
    public static AsyncLocal<string> EventId = new AsyncLocal<string>();
    public IActionResult Process([FromBody]) string eventsJson
    {
        var events =JArray.Parse(eventsJson);
        foreach (var @event in events)
        {
            EventId.Value=@event["id"].ToString();
            if(@event["topic"].ToString().Contains("providers/Microsoft.Storage"))
            {
                sendToDetectionService(@event["data"]["url"].ToString());
            }
            {
                LogService(@event["subject"] ToString());
            }
       }
       return null;
       }
```

```
private async Task SendToAnomalyService(string uri)
{
    var content = GetLogData(url);
    var bhuvanlabs_scroreRequest = new
    {
        input = new Dictionary<string, List<IDictionary<string, string>>>()
        {
            "input1", new List<Dictionary<string, string>>()
            {
                new Dictionary<string, string>>()
                {
                    "logcontent", content
                }
            }
        }
    },
    GlobalParameters = new Dictionary<string, string>(){}
}
```