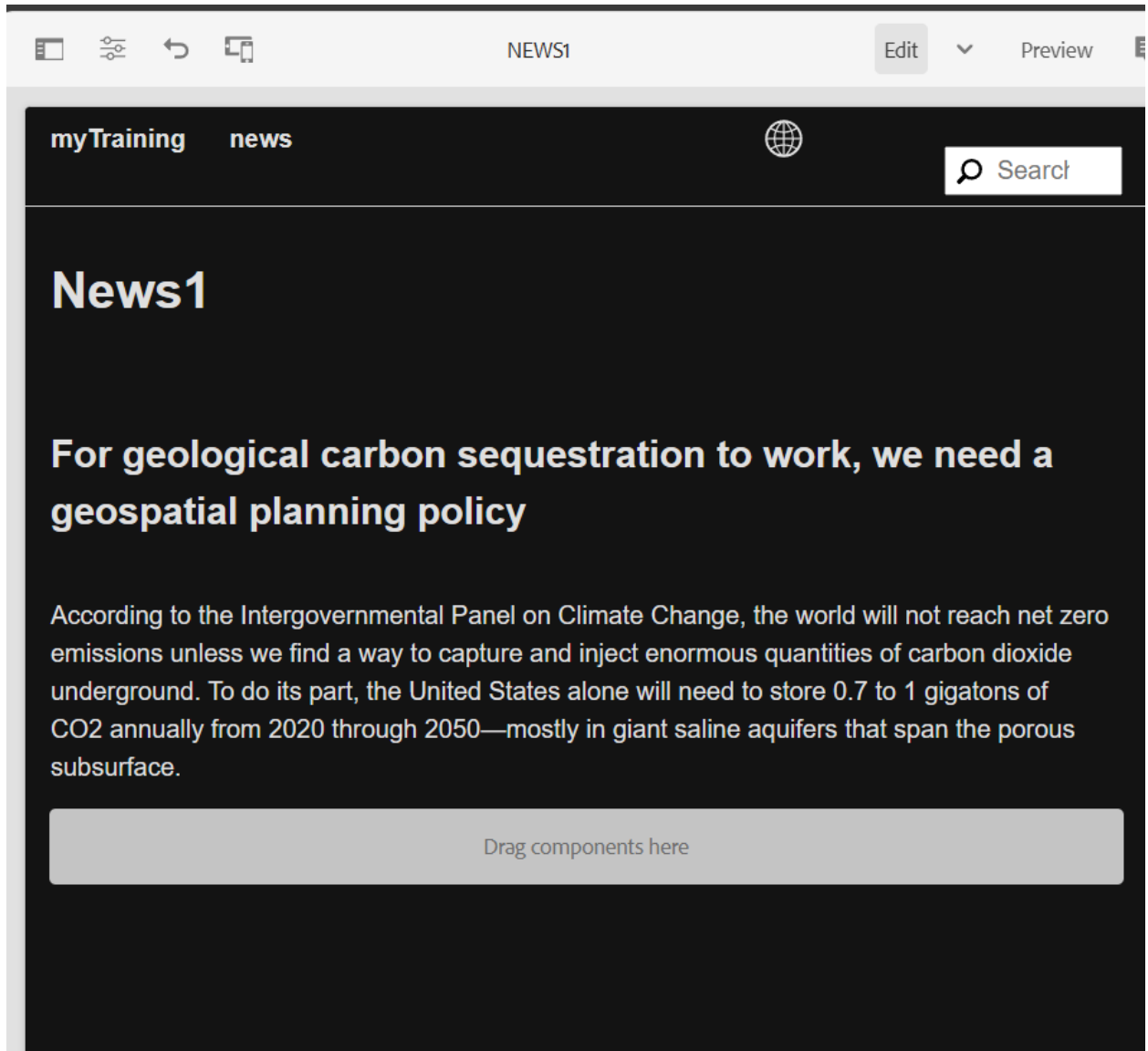


Assignment:

1. Create 5 news article pages under /content/us/en/news



# News2

## Climate change impacts have potentially big repercussions for kids' education

In the spring of 2022, the Calf Canyon/Hermit's Creek fire burned over 300,000 acres in New Mexico, including terrain near the Rio Gallinas School for Ecology and the Arts in Las Vegas, New Mexico. "If you were in the parking lot, you could see all this smoke coming up," says the school's director, Aja Currey. Las Vegas, New Mexico is a rural community—vastly different from its Nevada namesake—and children aged 5 to 14 travel from around the region to attend the school. Some students and staff evacuated from areas that burned and wondered if they'd have a home to come back to. Classes were in session for about two of the three months the fire raged, and children showed up when they could. But Currey estimates that around a quarter of the student body was absent at any given time. More recently, fires around Los Angeles have caused over half a million students to miss school.

[Read More](#)



## News3

### How do nanocrystals grow into myriad shapes? A model shows the way.

Scientists and engineers use nanocrystals in sensors, in catalysts, and for drug delivery. But how well they perform depends on their shape—and that can be difficult to predict. Researchers in Germany have now produced a model of how these crystals grow, which offers a theoretical framework of the atomic-level mechanisms involved.

[Read More](#)

Drag components here



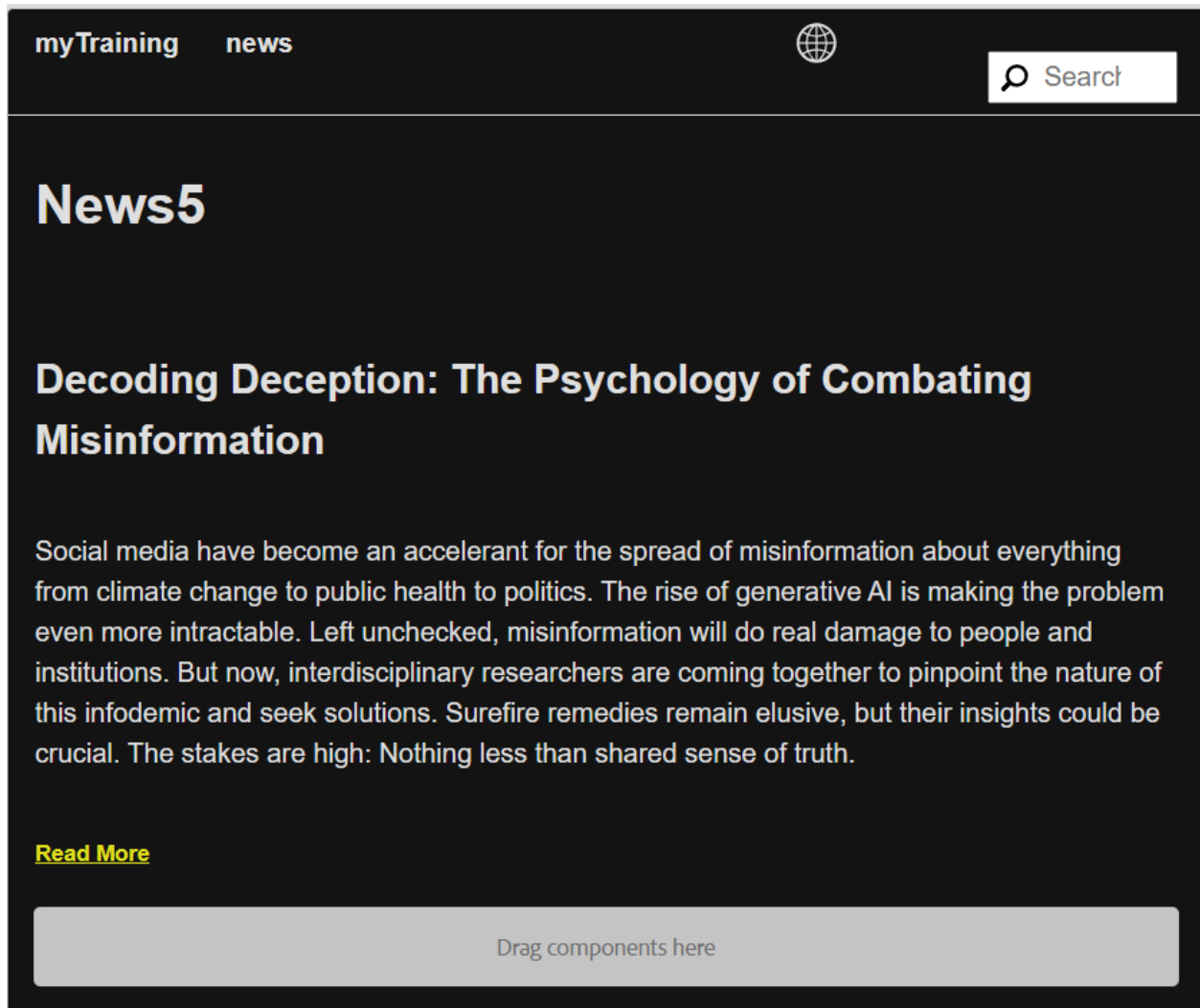
# News4

## Greener skies? The quest for sustainable flying

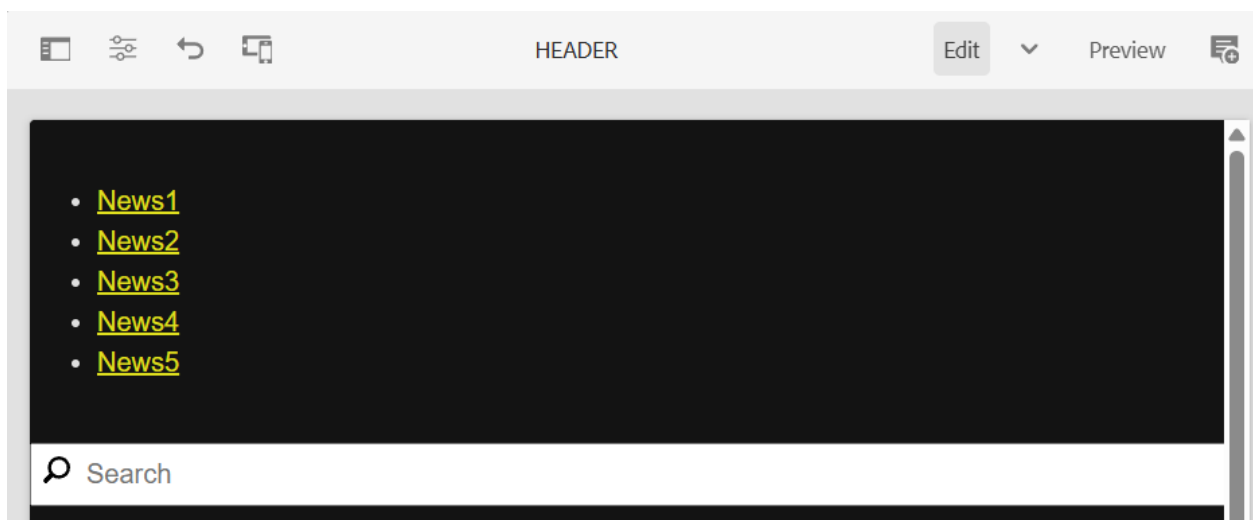
Flying long distances is routine for millions of people around the globe. Unfortunately, pushing a plane through the air at close to the speed of sound requires a tremendous amount of energy, and right now, only fossil fuels have the necessary punch. That, of course, makes flying a major contributor to climate change.

[Read More](#)

Drag components here

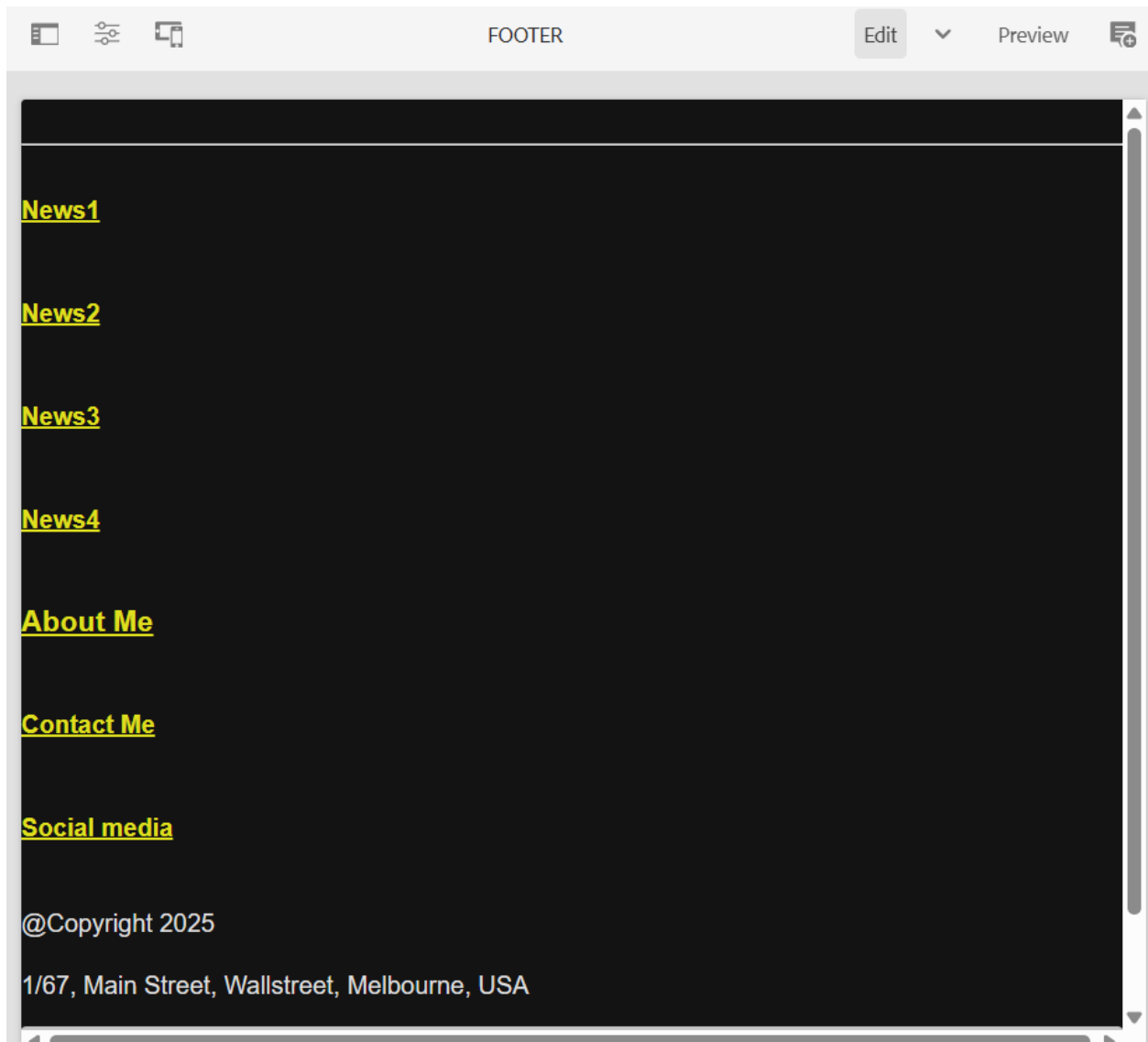


Create Header Experience fragment for header and use these page as menu (name should be news) and apart from that create contact us page and about me page



Create footer XF and it could have 4 sections:

1. News menu section-can have 4 news article you can use list component for this
2. About me section-can have some content and u can use text compoennt
3. contat us section-can have some content and u can use text compoennt
4. Social media section-you can use list component for this to provide som social media account references.



Create a custom service to print hello world and call this service from news component sling model and print this value in logs as well.

```
// com.example.core.services.HelloWorldService.java
package com.example.core.services;

public interface HelloWorldService {
    String getMessage();
}
```


```
// com.example.core.services.impl.HelloWorldServiceImpl.java
package com.example.core.services.impl;

import com.example.core.services.HelloWorldService;
import org.osgi.service.component.annotations.Component;
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;

@Component(service = HelloWorldService.class, immediate = true)
public class HelloWorldServiceImpl implements HelloWorldService {

    private static final Logger LOG = LoggerFactory.getLogger(HelloWorldServiceImpl.class);

    @Override
    public String getMessage() {
        String message = "Hello World from Custom Service!";
        LOG.info("HelloWorldService called: {}", message);
        return message;
    }
}
```



```
// com.example.core.models.NewsModel.java
package com.example.core.models;

import com.example.core.services.HelloWorldService;
import org.apache.sling.api.resource.Resource;
import org.apache.sling.models.annotations.DefaultInjectionStrategy;
import org.apache.sling.models.annotations.Model;
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;

import javax.annotation.PostConstruct;
import javax.inject.Inject;

@Model(adaptables = Resource.class,
      defaultInjectionStrategy = DefaultInjectionStrategy.OPTIONAL)
public class NewsModel {

    private static final Logger LOG = LoggerFactory.getLogger(NewsModel.class);

    @Inject
    private HelloWorldService helloWorldService;

    private String helloMessage;
```

```
@PostConstruct
protected void init() {
    if (helloWorldService != null) {
        helloMessage = helloWorldService.getMessage();
        LOG.info("NewsModel received message: {}", helloMessage);
    } else {
        LOG.warn("HelloWorldService is not available in NewsModel");
    }
}

public String getHelloMessage() {
    return helloMessage;
}
}
```



Create custom configurations where I can provide the 3rd party api for  
example(<https://jsonplaceholder.typicode.com/posts>)

```
// com.example.core.services.impl.ExternalApiServiceImpl.java
package com.example.core.services.impl;

import com.example.core.config.ExternalApiConfig;
import org.apache.commons.io.IOUtils;
import org.apache.http.client.methods.CloseableHttpResponse;
import org.apache.http.client.methods.HttpGet;
import org.apache.http.impl.client.CloseableHttpClient;
import org.apache.http.impl.client.HttpClients;
import org.osgi.service.component.annotations.*;
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;

import java.io.InputStream;
import java.nio.charset.StandardCharsets;

@Component(service = Runnable.class, immediate = true, configurationPolicy = ConfigurationPolicy.
@Designate(ocd = ExternalApiConfig.class)
public class ExternalApiServiceImpl implements Runnable {

    private static final Logger LOG = LoggerFactory.getLogger(ExternalApiServiceImpl.class);

    private String apiUrl;
```

```
@Activate
@Modified
protected void activate(ExternalApiConfig config) {
    this.apiUrl = config.apiUrl();
    LOG.info("Configured API URL: {}", apiUrl);
    run(); // trigger immediately
}

@Override
public void run() {
    try (CloseableHttpClient client = HttpClients.createDefault()) {
        HttpGet get = new HttpGet(apiUrl);
        try (CloseableHttpResponse response = client.execute(get);
            InputStream content = response.getEntity().getContent()) {
            String json = IOUtils.toString(content, StandardCharsets.UTF_8);
            LOG.info("Fetched API JSON: {}", json);
        }
    } catch (Exception e) {
        LOG.error("Failed to fetch data from external API", e);
    }
}
```