Contents

[Introduction 1](#_Toc448163405)

[Overview 1](#_Toc1266578712)

[Background 1](#_Toc1845706560)

[Review of competitive products looking at common core features and advanced features 1](#_Toc64984375)

[Core Functions 2](#_Toc158118698)

[Advanced Functions 2](#_Toc558887388)

[Data Protection 3](#_Toc580515327)

[Implementation 3](#_Toc317178249)

[User Interface 3](#_Toc1576253042)

[Technology Stack 3](#_Toc934215174)

[Data Organisation 3](#_Toc2017215064)

[Hosting 3](#_Toc822897085)

[Services 3](#_Toc1229947674)

[Scalability 4](#_Toc1807136024)

[Tracking and Statistics 5](#_Toc1178278203)

[Conclusion 5](#_Toc1926476360)

[References 5](#_Toc453764974)

# Introduction

Do towards end of development.

Describe problem

State scenario with details of selected cultural festival.

A local film festival has put out a tender for a new website to serve their needs they require a website to advertise the festival as well as directing users to the appropriate places to be able to fins out information about locations, what’s on, as well as being signposted to where to book tickets.

Background of scenario and client requirements

The festival is based in the Glasgow area and uses existing cinema facilities such as the GFT theatre, the Grosvenor, the everyman cinema as well as certain Cineworld and Odeon screens. The client is looking for a website which informs its users of what’s on, where and when, a section with background information on the festival and previous contributors, images or photos of previous festivals, the ability to search the available events. Initial prototype to show the basic functionality as a proof of concept with further advanced functionality included where time is available or indicated in the report.

Highlight any key points.

# Overview

## Background

Data sources app will use – public, private, licensed APIs

Mapping service such as Google Maps, HERE maps, or Mapbox.

Database/API to serve images and information to display to users

User inputs

A contact form for more information about the festival.

Login feature to track history.

## Review of competitive products looking at common core features and advanced features

<https://glasgowfilm.org/glasgow-film-festival>

<https://www.edfilmfest.org.uk/visiting>

<https://whatson.bfi.org.uk/lff/Online/default.asp?menu_id=D90AD2E1-12B8-4B8D-97A2-DB9A5A4BC4A4>

For a local film festival, we reviewed 3 different festival websites, the Glasgow film festival, Edinburgh International film festival and the BFI London film festival with a view to comparing the common features across all three then assessing what would be regarded as the core features to make a functioning website and what would be an advanced feature to provide more value to users of the website. This will help us in our development of a prototype proof of concept and guide any future developments.

Across the 3 websites the common features are background/history of the festival with an About us section, A means to donate to support the festival, a section containing the latest news, A newsletter signup form.

Each of the websites has a means of logging in or signing up, a ticket purchasing section or signposting to a third-party site or other means of booking tickets.

Development time and cost

Does the proposed app meet the brief

## Core Functions

Determine requirements of the app –

In an initial launch of the app there will be certain functionalities required to allow the clients to get the best use of the website at launch.

Display schedule – showings with locations and film information.

history of event - articles

photo gallery

key people involved

what’s on

Search events

Login functionality for an administrator to add festival events.

What data is needed

Profile information

## Advanced Functions

Advertise for volunteers

Apply for sponsors

Apply to take part?

tickets (or links to buy?)

## Data Protection

GDPR considerations for users alongside warnings about cookies used to track users will need to be considered

# Implementation

## User Interface

## Technology Stack

Mongo, Express, React and Node.JS

## Data Organisation

Db tables and diagrams

# Hosting

## Services

To deploy a project, we will require a hosting service to deploy the prototype onto to make the project available to show the client. There is an option to build and run a client’s own server to deploy the project and serve customers but that would involve added cost and technical specialists to run. Using a cloud-based solution is a more cost-effective solution to project deployment in which costs start at a free tier for small projects rising as more resources are used. This allows a project to grow naturally as well as any related costs rising alongside.

A hosting service such as Heroku would be a suitable solution as they provide low-cost solutions to deploy an application. Heroku is a Platform as a Service which is based on a managed container system and was created to provide an application focused approach (Heroku, n.d.). Heroku can use either their own CLI or GitHub integration to allow for quick uploading and updating of any code changes to which are made to an application this leads to a quick turnaround for any updates to be published to the production server once tested.

Heroku also provides many tools to manage the troubleshooting of issues one of these tools, Heroku OpEx can alert a developer to issues that arise quickly allowing for a quick response in identifying and solving problems. Using Heroku OpEx a developer has access to Logs to receive data regarding any errors or events, statistics such as the memory and data usage of a dyno or how quickly the dyno is serving HTTP requests. All information which can be used to inform the best way to use the platform. This dashboard also allows for Autoscaling of the dynos to quickly respond to increases in traffic to ensure the application does not struggle with unexpected traffic.

## Scalability

Any website or application which experiences any kind of significant growth will need to be able to scale to meet that demand. As this project is a data driven application it will be important to consider a scaling solution which considers the security and the integrity of the data which is being used. There are two main types of scaling to consider Vertical and Horizontal.

Vertical scaling is where a hosting service adds more computing power to the existing application to increase the computing power such as the CPU, RAM and DISK to deal with increasing workload. This, however, is restricted to the limits of the server which it is hosted on and as such there may be a risk of having downtime or a service outage if the demand reaches higher than can be currently coped with.

Horizontal scaling is where more machines are added to the resources available to the application and processes may be split among these extra machines to balance the user connections to the application and share the load.

Heroku uses a container system to run an application and initially this will be deployed on one web dyno (Heroku container) to serve the web app. As this project uses Node.JS which is JavaScript based technology and is therefore a single-threaded language, an API called Cluster is required to run multiple processes at the same time. This can be enabled in the CLI and will allow the app to be able to scale through the use of multiple dynos (Heroku, 2022). A developer is able to choose from the many solutions offered and can upgrade or downgrade a dyno quickly to be able to respond to the traffic an application is receiving.

This offers the flexibility of horizontal and vertical scaling and provides a good solution using the best method to serve an application. Setting up an application to run on multiple dynos also offers a level of protection against an outage as they are more likely to be run on different servers, meaning that if one goes down the application would still be available to users. This is one of the main benefits of using a cloud-based solution to serve a web application.

## Tracking and Statistics

Heroku offers a dashboard as an interface which provides support for a developer to keep informed of the existing web application including the applications use of the platform’s resources such as any add-ons and the dynos which the app is using at any time. The dashboard also allows a developer to view logs of

# Conclusion

# References