A Tale of Tales

Introduction

This report presents the project plan for the COMP1004 module coursework. The Project shown below is a hypertext fiction game that explores the stories from different myths and legends in a way that aims to be fun and engaging.

Within this report the Software Development Lifecycle (SDLC) is discussed with an explanation of how this is applied to the current project. Following on from the project plan a design document is provided. This will be written in the format of a GDD.

Software Development Lifecycle

This section discusses the Software Development Lifecycle (SDLC) and describes how this is being used in this project.

In this scenario, we will be using the Agile Ideology of the SDLC to make the game. The Agile Ideology is where each individual aspect of the project is planned for, designed, and implemented separately and sequentially.

For example, I had to start by figuring out which stories I wanted to add to my game and how to link them to one another, which I then planned for and put into paragraphs in an html file. Afterwards, I would plan how I would go about implementing the links to each of the stories in a single page application and then add them with hopefully no issues.

Design Document

This section provides the key elements of the design documentation.

Project Vision

When most people think of the Norse god Thor, they will imagine Thor from the film series produced by Marvel and acted by Chris Hemsworth. This is one example of many where people's perception of something from a myth is vastly different to what was told in any correct version. My aim with this project is to create a fun game which correctly tells the stories from myths to inform people of the correct versions of different characters and relics.

Final Project Plan

Sprint 1 and 2:

Created an html file.

Things to do for next time:

Learn html and CSS.

Sprint 3:

Started coding in vs Code.

Created a paragraph on Mjollnir.

Things to do for next time:

Add more paragraphs.

Sprint 4:

Added paragraphs for all the chosen stories in Norse mythology in a random order.

Things to do for next time:

Implement some CSS.

Sprint 5:

Made the background dark grey and the words white.

Changed the paragraphs to be in order.

Added the first paragraph for Greek mythology.

Things to do for next time:

Hyperlinks.

Sprint 6:

Added hyperlinks to the Norse paragraphs.

Added a hyperlink at the top for each mythology.

Added paragraphs for all the chosen stories in Greek mythology in order.

Things to do for next time:

Japanese folklore.

More CSS.

Sprint 7:

Added the first paragraph for Japanese folklore.

Made a grid for the hyperlinks at the top.

Created a JavaScript file.

Things to do for next time:

JavaScript.

Sprint 8:

Added paragraphs for all the chosen stories in Japanese folklore in order.

Fixed an issue with the CSS grid.

Created a random number function for the links at the top of the page.

Things to do for next time:

Make the paragraphs appear from nothing.

Sprint 9:

Made paragraphs appear when clicking on a link that take the user to chosen paragraph.

Initial design completed.

Things to do for next time:

Make any improvements if any are needed.

Background

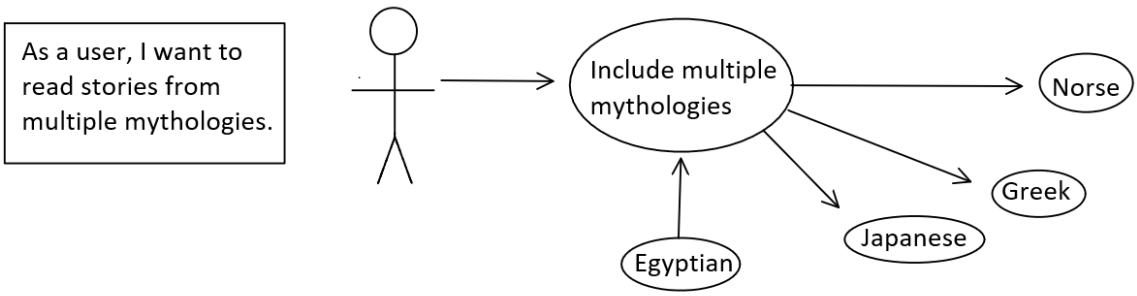
A Tale of Tales is a hypertext fiction game focusing on providing the user with multiple stories from various backgrounds in a unique way. The game's main way of playing has taken direct inspiration from the video game Immortality, where the player is tasked with exploring the making of three movies starring the same actress and what went wrong. How Immortality functions is rather simple, you are given a clip from one of the movies and can click on different items or people in the clip. Each clickable item transports you to a different clip to repeat the previous actions until you have collected every piece of information about the movies and can figure out what happened. A Tale of Tales functions similarly to this in the sense that in each mythology, you are given a few stories split up into different paragraphs and are intertwined within each other. There are certain words in each paragraph which will take you to different paragraphs until you have collected all sections of each of the stories. The aim of this gameplay system is to help readers stay engaged in the stories they are reading by making them find the next, or even previous, section of the story.

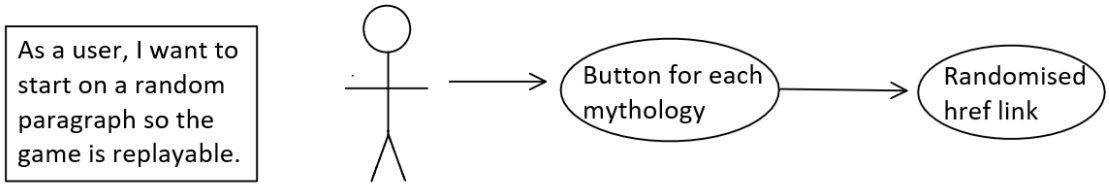
Consideration must be given to legal, social, and ethical issues relating to this project. Game assets, characters, sound effects and music for example either do not exist or are simply geometric shapes. In addition, all mythologies and their stories are public domain for anyone to use.

Finally, web accessibility is a key aspect to be considered when designing the interface. Good accessibility not only makes sense to allow a wider audience to be able to use the application, but it is also a legal requirement under the UK Equality Act of 2010 (Equality, 2010).

User stories and Associated Use Case Scenarios

This section examines the user stories identified so far and provides the accompanying use case scenarios.

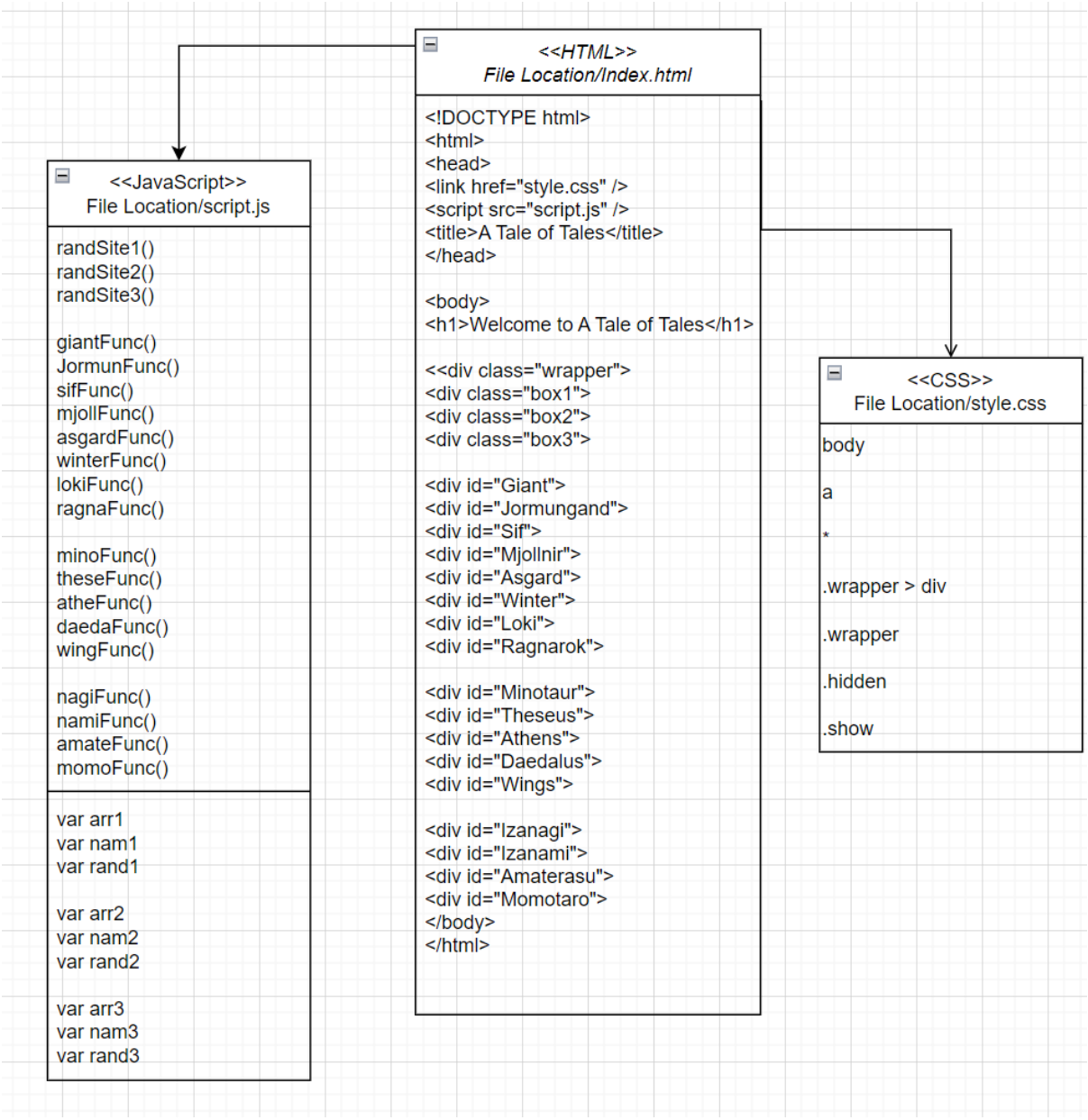




Name	Multiple mythologies
Short Description	Include multiple mythologies
Precondition	User wants to read a certain mythology
Post Condition	User plays the game related to chosen mythology
Error Situations	Mythology does not exist/is not included on website
System state in the event of an error	Mythology does not exist
Actors	User
Triggers	User must select a mythology that is on the website.
Standard Process	1. User selects a mythology. 2. Beginning paragraph for chosen mythology appears.

Class Diagram

My code does not contain any classes but I have created a diagram that is similar.

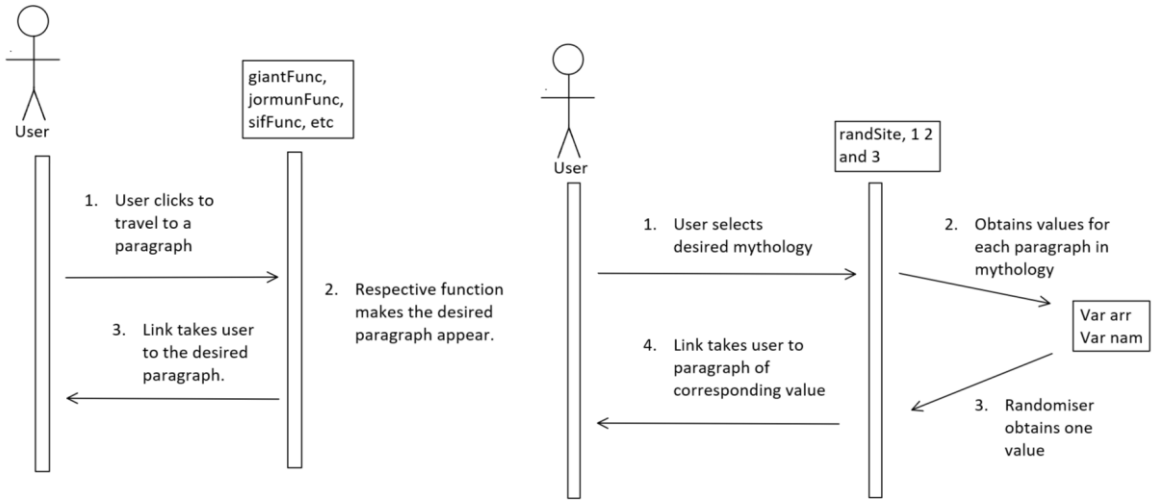


Architecture

This section discusses how the architecture for the single page application is envisaged.

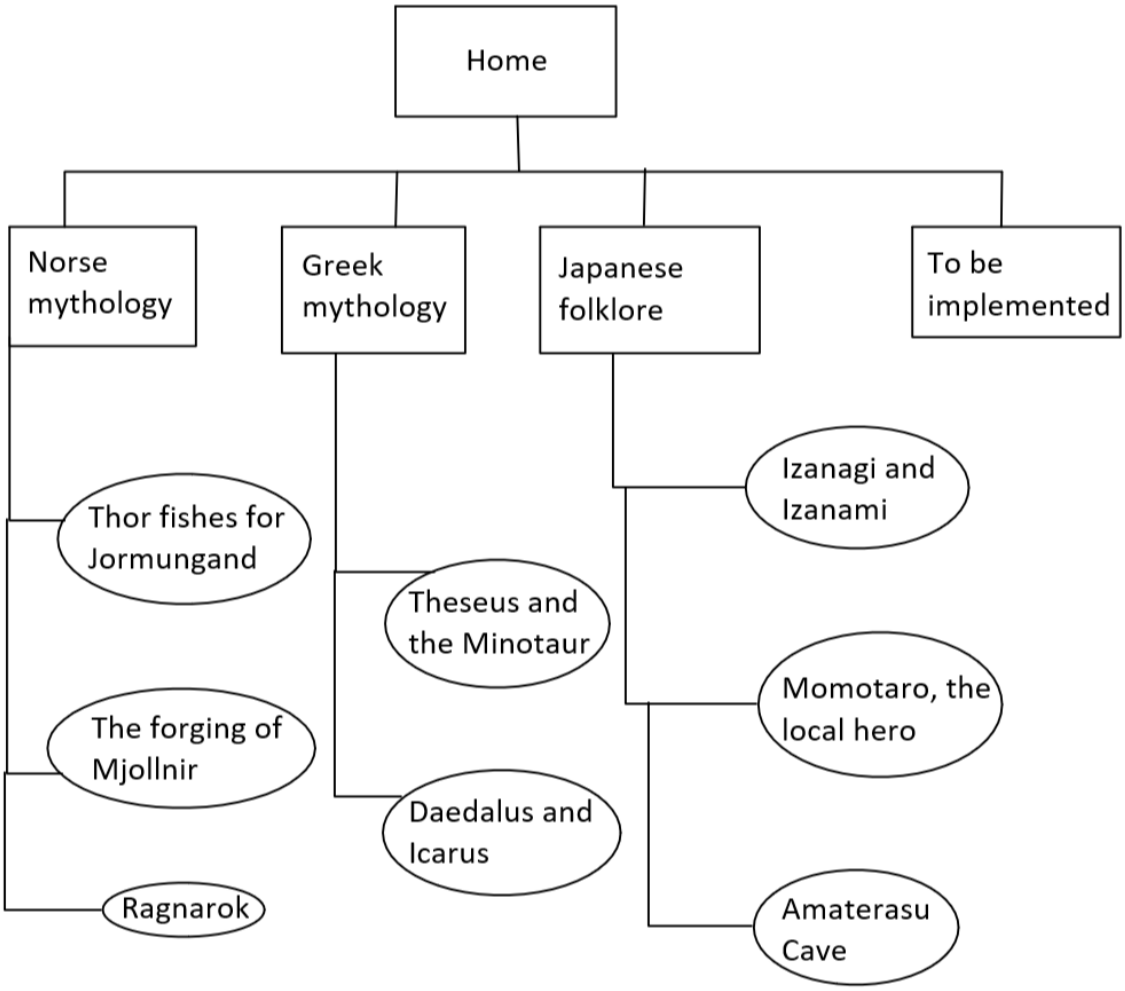
My code does not have any classes, nor does it have any functions that call other functions. Therefore, I see no point in creating a class diagram that contains nothing in it.

The sequence diagrams below show how the classes will interact to implement the given use case scenarios.

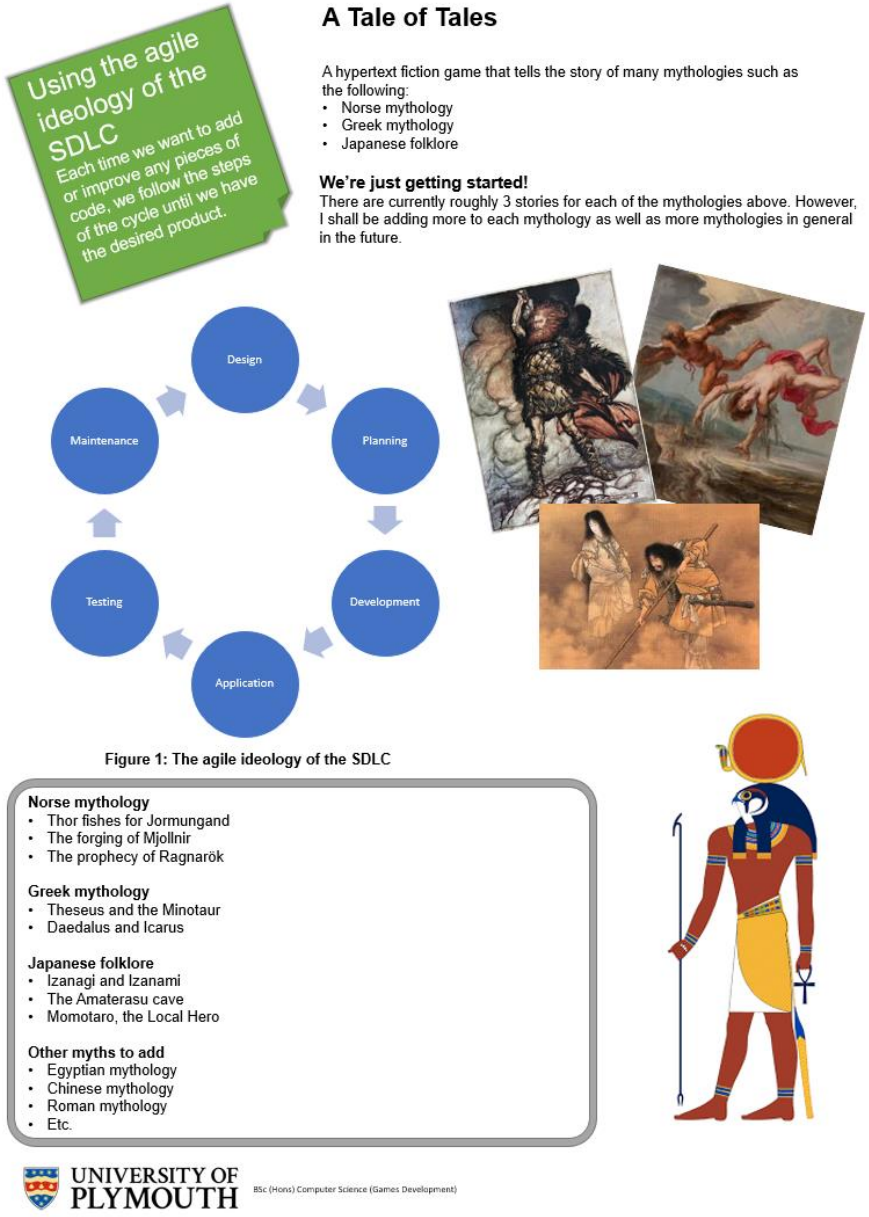


Sitemap

This section provides an outline of how the application is designed. Whilst this is a single page application, the sitemap indicates how the user will navigate through the topics.


Home
Norse mythology
Greek mythology
Japanese folklore
To be implemented
Thor fishes for Jormungand
The forging of Mjollnir
Ragnarok
Theseus and the Minotaur
Daedalus and Icarus
Izanagi and Izanami
Momotaro, the local hero
Amaterasu Cave

Poster



Noted issues and constraints.

While preparing for this section, I unfortunately was unaware of how to code in either HTML, CSS, or JavaScript prior and we only were assigned a singular lecture for each going over the bare bones of them all. Therefore, I had to spend a couple weeks learning each of the languages to a respectable level before even contemplating how to go about coding my website. I started off by using the guide on how to make a html file on the DLE. However, the guide was outdated and had to figure out how to start it. In the end, I was able to create the correct file and was able to run it in Visual Studio Code. I had one problem with the CSS in my code where I had put the background colour for the grid at the top of the page in the wrong section. This made it so that there was a square around the entirety of the grid instead of there being a square around each separate item in the grid. There were not many other issues that I had to overcome, yet there was an ever so slight problem with the time constraint, having to complete our project alongside the completion of the assignments from two other modules.

GitHub repo link

https://github.com/MichaelHooper36/COMP1004-Assessment

References

Equality, 2010, Equality Act, 2010 c 15, https://www.legislation.gov.uk/ukpga/2010/15/contents, date last accessed 15/04/2024.

-