Cardiff School of Computer Science and Informatics

Coursework Assessment Pro-forma

Module Code: CM6311
Module Title: Commercial Frameworks, Languages and Tools
Lecturer: Dr Kathryn Jones
Assessment Title: Coding Task
Assessment Number: 1
Date Set: 1st November 2019 (Week56)
Submission Date and Time: Friday 8th November at 9:30am (Week 6)
Return Date: 6th December 2019 (Autumn Semester Week 10)

This assignment is worth 50% of the total marks available for this module. If coursework is submitted late (and where there are no extenuating circumstances):

1. If the assessment is submitted no later than 24 hours after the deadline, the mark for the assessment will be capped at the minimum pass mark;
2. If the assessment is submitted more than 24 hours after the deadline, a mark of 0 will be given for the assessment.

Your submission must include the official Coursework Submission Cover sheet, which can be found here:

https://docs.cs.cf.ac.uk/downloads/coursework/Coversheet.pdf

Submission Instructions

The module leader clearly defines how the assessment should be submitted. For example, should students submit a single PDF or multiple files. The coursework specification will specify the number of files expected along with the permitted filetypes.

Description	Type of submission	Name of submission
Coversheet	One PDF (.pdf) file for each team member.	[student number].pdf
Application	downloadable link to GitLab project (100%).	[sappo].zip

Any code submitted will be run on or from your NSA provided Windows Laptop and must be submitted as stipulated in the instructions above. Any deviation from the submission instructions above (including the number and types of files submitted) will result in a mark of zero for the assessment.

Staff reserve the right to invite students to a meeting to discuss coursework submissions

Assignment

This coursework is not about making a "Big Code" application, in fact spending time over engineering your application could mean you neglect software quality considerations. Instead, this assessment is about engineering an application that demonstrates you can meet the requirements, write quality software and identify any paradigms used. The scenario below provides the context for the application and a list of requirements:

Scenario & Requirements – Citizen Science in the Brazilian Rainforest

There is a community of school children in the Brazilian Atlantic Rainforest that are participating in a Citizen Science Project. The project aims to help them engage in science by learning more about the indigenous species of frog that live in the forest around them. As part of the project, experiments have been conducted around the potential to set up an *ad hoc* mobile communication network in the forest. If this is possible the children will be able to connect to the network and communicate via their smart phones. When on a frog finding expeditions into the forest this will allow the children to communicate with each other things like species of frog and geotag the location, etc. The children and community would like a bespoke web application that allows them to message the group and record information on the frogs they find. The children would like to call the application "Sappo" (portuguese for frog) and would also like their logos to be incorporated into the design.

Based on the above scenario the minimum viable product (MVP) for the application is as follows:

1) The application should allow broadcast communication of messages to all connected users.
2) The application should allow messages to be date and timestamped
3) The application should allow messages to be associated with a username
4) The application should allow the following additional data to be sent in the message: geographic coordinates, temperature, abundance and species identity.
5) The application should incorporate a logos into the application interface.
6) The application should incorporate the name of the application into the interface.

If the MVP is exceeded there are three more requirements that should be considered:

7) Allow picture and/or audio to be sent to the group.
8) A way of persisting all messages at the received at the server.
9) Keep a local count of species of frog found.

Any additional features or innovations you would like to add can also be incorporated and should be included and justified in the ReadMe.md.

ReadMe.MD

You should add a ReadMe.md file to your project and it must include:

- how to build/start/run/compile the project
- details of any assumptions you have made
- details of any aspects you have mocked
- explanation of any paradigms used
- justification for styling of application

Quality solutions will also provide an example(s) of how testing can be used to ensure quality of your solution.

Technology Stack

The application should be built using React.js on the frontend and Node.js on the backend. You are permitted to use any other commercial frameworks, languages, tools and libraries that you believe necessary to your solution, but you must justify use of these in your ReadMe file. For example, using Bootstrap just to simply add one heading or button is not a good enough reason to use it in the first place.

You are permitted to do one of the following to make your project:

- Use create-react-app on your local machine to create a starter project yourself.
- Produce a starter project using your preferred method and build tool.

The project should either be downloadable/cloneable from GitLab.

Learning Outcomes Assessed

1. Recognise, identify and characterize a range of engineering paradigms.
2. Develop commercial software using an alternative framework, language, or tool.
~~3. Recognize the advantages and disadvantages of alternative frameworks, languages, and tools for commercial projects.~~
~~4. Reflect on the advantages and disadvantages of adoption of frameworks, languages, and tools for commercial projects.~~

Skills Addressed

- Using commercial development, build and test tools.
- Using third-party and/or open-source frameworks, languages, and tools.
- Leveraging freely available software tools and online resources.
- Planning/managing your own workload.

Criteria for assessment

Credit will be awarded against the following criteria.

Classification	Criteria
1st (70 -100%)	• Project has excellent/outstanding, originally thought out, functionality that meets and goes beyond the MVP with some innovations evident. • Specified technology and libraries have been used beyond what was expected. With concepts attempted, demonstrating knowledge beyond coverage in the lectures. • The project builds and starts according to the available instructions with no issues. • Justification is provided for choice of method to create a project and innovations in build and usability are evident. • Project has very well styled components with a strong justification of why that approach was taken. • There is an excellent testing strategy and several examples of how you might test a function/component/view. • The project has excellent organisation, and code quality is of a very high standard. There are aspects of the code that are beautifully crafted.
2.1 (60% – 69%)	• Project has good well thought out functionality that meets the MVP and, in some way, goes beyond. • Specified technology and libraries have been used as expected. Concepts attempted, at an advanced level, that were covered in the lectures. • The project builds and starts according to the instructions. • Justification is provided for choice of method to bootstrap the project and any boilerplate. • Project has nicely styled components and a supporting justification of approach has been provided. • There is a well thought out test strategy and at least one relevant example of how you might test a function/component/view. • The project has been well thought out in terms of organisation, and code quality.
2.2 (50% - 59%)	• Project meets the MVP. • Specified technology and libraries have been used. Concepts attempted that were covered in the lectures. • There are a few issues with either the instructions or building and starting the project accordingly. Some justification has been provided regarding the approach to bootstrapping the project and any boilerplate used. • Project has some styled components little to no justification of approach has been provided. • There is some strategy and a non-relevant example of how you could test a function/component. • The code has some organisation and some attention has been paid to code conventions. • There are some appropriate comments.
3rd (40% - 59%)	• Project has basic functionality and in some ways the solution works well enough. • Specified technology and libraries have been used. Concepts attempted that were covered in the lectures. • There are a few issues with either the instructions or building and starting the project accordingly. • Some basic justification has been provided regarding the approach to bootstrapping the project and any boilerplate used. • Project has some styled components and limited justification of approach has been provided. • There is some strategy and a non-relevant example of how you could test a function/component. • The code has some organisation and some attention has been paid to code conventions.
Fail (< 40%)	• Project has very limited functionality very few of the requirements have been implemented and as such solution is limited in how it works. Specified technology and libraries have not been used as expected. Very few concepts attempted that were covered in the lectures. • The project does not build or start and there are no instructions. • There is no justification in why the chosen approach to bootstrapping and use of any boilerplate code have been taken. • Project has no styled components beyond default style provided via any boilerplate. • There is no test strategy provided and no example of how you might test a function of component with in the project. • The code is not well organised and doesn't adhere to relevant conventions of the technology.

Feedback and suggestion for future learning

Generalised cohort feedback will be provided via Learning Central by November 22nd. Individual feedback on your coursework will address the stated criteria and will be returned along with marks on 6th December 2019 via Learning Central, this will be useful for the module's second assessment and the Large Team Project module. Further opportunities for feedback are provided by appointment only.