|  |
| --- |
| ***Author: Andrea Baccolini: x18147518@student.ncirl.ie*** |



|  |
| --- |
| **Higher Diploma in Science and Web technologies – National College of Ireland** |
| Project name: rockonnect |
| Interim progress report |
| *Student: Andrea Baccolini [student no: 18147518]*  Date: Sunday, June 30, 2019 |

**Recipient Organization list**

|  |  |  |
| --- | --- | --- |
| **Name** | **Title** | **Version** |
| Liam McCabe | Supervisor |  |
|  |  |  |
|  |  |  |

**Table of Contents**

[1 Introduction 3](#_Toc12720584)

[2 Abstract 3](#_Toc12720585)

[3 Project overview 3](#_Toc12720586)

[3.1 Project expected deliverables 4](#_Toc12720587)

[3.2 Project actual targets met 5](#_Toc12720588)

[3.3 Deliverables yet to be achieved 11](#_Toc12720589)

[4 Main problems encountered 12](#_Toc12720590)

[4.1 Major issues 12](#_Toc12720591)

[4.2 Minor issues 12](#_Toc12720592)

[5 Expected stumbling blocks 12](#_Toc12720593)

[6 Focus on rubric 13](#_Toc12720594)

[7 Summary of project progression 13](#_Toc12720595)

[8 Graphical representation 14](#_Toc12720596)

[9 Appendix A: list of figures 16](#_Toc12720597)

[10 Appendix B: Acronyms 17](#_Toc12720598)

[11 Bibliography 17](#_Toc12720599)

# Introduction

This document describes the project starting from its overview, then going on listing the deliverables and after that analysing which of them have been met at the time of the report and which are to be met yet and why. The document also describes the issues and problems encountered during the project development and finally provides a conclusion on the plan for the remaining milestones to complete.

# Abstract

In the final semester of the “Higher Diploma in Science and Web Technologies”, the students are required to develop a “project” which, in this case, consists in the development of a web application focused on a subject of students’ choice. In addition to that, the project requires the production of all the documentation necessary to design, implement, test and deliver the application as a final working solution. For this purpose, this document is set to report the progress of the mentioned activities, the encountered issues and plan to overcome them, from the perspective of a potential consumer of the web application.

# Project overview

The project is called “rockonnect”. The project owner is Andrea Baccolini, student number 18147518 at the “National College of Ireland”.

The project consists in a web application called rockonnect. The main purpose of rockonnect is to give an online space to classic rock lovers, where they can share experience on everything that related to the classic rock; from music, to concerts, to album releases, to the musical instruments used during the classic rock ages in Europe and US [from early 80s late 90s, (Hickey, 2014)]. To achieve that purpose, the web application provides two macro pieces of functionality described below.

The first piece is the “social media” functionality: with this web application, users can register to service, browse the profile of the members and connect to each other.

The second piece of functionality is the “discussion forum”: with this tool, users can create discussions about their own experience, read other people threads and share learn from each other.

In order to achieve the scope, the web application is designed to contain elements of client-side scripting, server-side languages and data storage functionalities to present consistent data to the users over time.

## Project expected deliverables

The project is going to be developed in four different stages or phases, each of them has specific deliverables. Please refer to the project plan, (Baccolini, 2019) for reference. The phases are described below.

*First phase*: research, upfront documentation and project setup. This is the initial phase of the project, and it consisted in the following deliverables:

* Declaration cover sheet
* Project proposal document
* Project requirement specifications document
* Interim report generation and submission
* Development environment setup:
  + for this purpose Visual Studio code has been chosen as development environment because of the following reasons: it can manage a huge variety of programming languages and technologies well: this is very important for the progress of the application and its roadmap features; it is well structure for web development and web designed; for example, VS Code allows to visualize well the adopted colours for a webpage during client side design giving a very good experience; it is very easy to use and intuitive [compared to others IDE]: for the purpose of the project was the best solution to adopt in relation to the short timeframe for providing a working web application.

*Second phase: Sprint 1*. This phase comprises the initial development of the server side and the CRUD part. The deliverables are:

* Project analysis and design implementation document
* Prototyping of the web application: this is the initial part of the code, where the team is starting to design the framework of the application and the framework of the code.
* Server-side development part 1: CRUD application development with external database. This part involves the creation of the server application using: node.js, express for the actual web server, mongoDB for the database/data storage part]
* User registration and login development and testing: this part involves the usage of express passport for user authentication, bcrypt for password encryption/decryption, and additional libraries for user interface messaging.

*Third phase: Sprint 2*. The second Sprint, or phase three of the project, has the following deliverables:

* Server-side development part 2: this part involves the generation of the remaining part of the server engine to allow the users to view the threads and create new threads. The threads are stored into another table in the existing MongDB collection in mongodb cloud.
* Client-side web development: this part involves the development of the “views” using “ejs” or “pug” for an easy representation of them, such as registration, login, logout, discussion threads generation and view.
* Testing of the web application features

*Fourth phase: Sprint 3* is the final phase of the project and the main deliverables are the following:

* Video and presentation generation of the web application
* Project final report generation
* Wrap up, tuning of the code, and code submission
* Final report [including test report] and final declaration submission

## Project actual targets met

The project started on May 25, 2019, right after the first online class where the supervisor has given the guidelines to the students.

Because the first important delivery was the “Project Proposal Document”, the first week was focused on finding the right subject for an application and delivering a consistent proposal, that has been accepted.

A second achievement is the designing of the “Project requirement Specifications document”, handed on-time as per project plan on June 15, 2019. This is an important achievement for the success of the project because the document defines the architecture, features and the requirements [in the form of functional clear use cases and non-functional] of the webapp. Without this document, there would not be any clear understanding of what to develop, what are the priorities and what is the required user experience [UX].

Because of the first two achievements, it was also much easier to meet other targets, such as initial framework, connection to database [mongoDB cloud cluster], CRUD development for user registration and threads and user registration with password encryption: all of those are essential parts of the framework. Please refer to the following figures for reference.

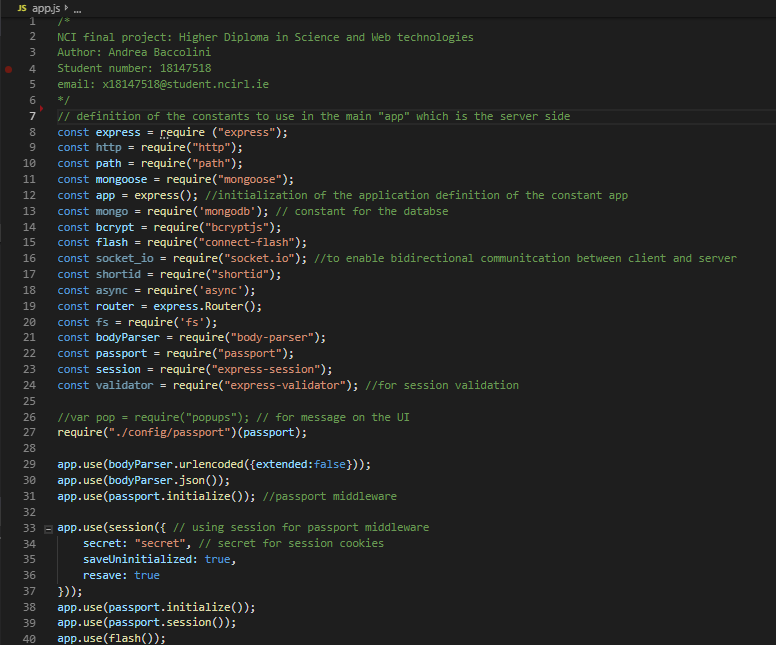


Figure 1 – app.js initial framework

*Figure 1* shows the basic framework of the application, the constants required for the app to work and the modules to use for the different functionalities required.

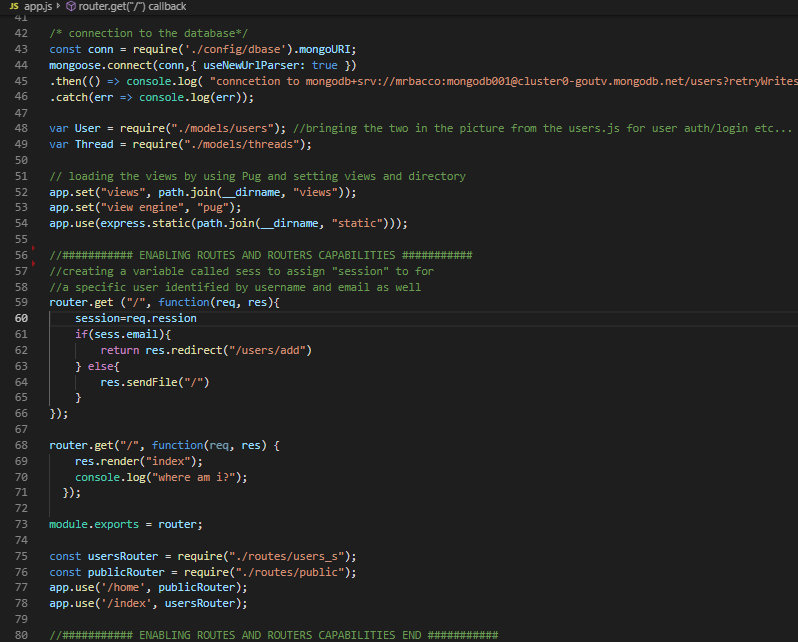


Figure 2 - Routes and database connection

*Figure 2* shows the connection to the database and the settings of the app including the basic routing functions to the index page and the users’deve page. This is needed for the application to allow only registered and then logged in users to view/edit items like users list or threads.

The example of the user registration is shown in *Figure 3*.

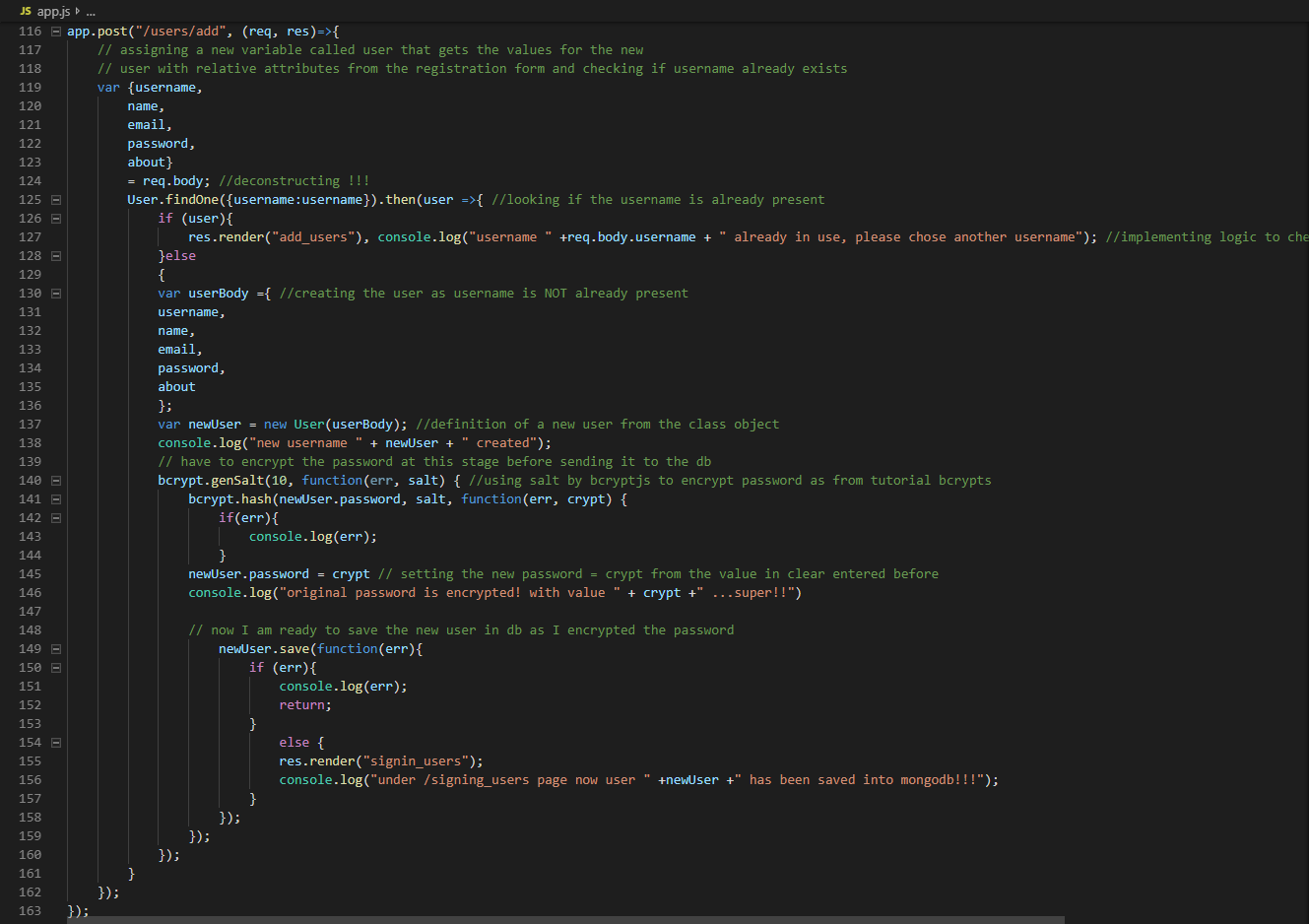


Figure 3 – User registration code

Username must be unique: to do so user registration logic performs an initial check on the username chosen by the user at the time of the registration, [please see lines 128 -131 in *Figure 3*]; if the username is already in use, then user is asked to choose another username, as it is, together with the id, the unique identifier of the record in the database. If username is not present yet, then the code is structured to encrypt the password selected by user at the time of registration and encrypt it using “bcrypt” function: please see lines 143-149 in *Figure 3*. If password encryption is successful, then the user can be saved in the database. Brcypt is also used for password decryption when user is logging in.

The login part has been partially accomplished, because it needs to be fully tested: the code for login is adapted from “passport” community (Passport, n.d.)framework. Please refer to *Figure 4* for a view of the login code which includes finding the users in the db, decrypting the password, comparing the password and, eventually, logging in the user [if all the previous steps are correctly accomplished].

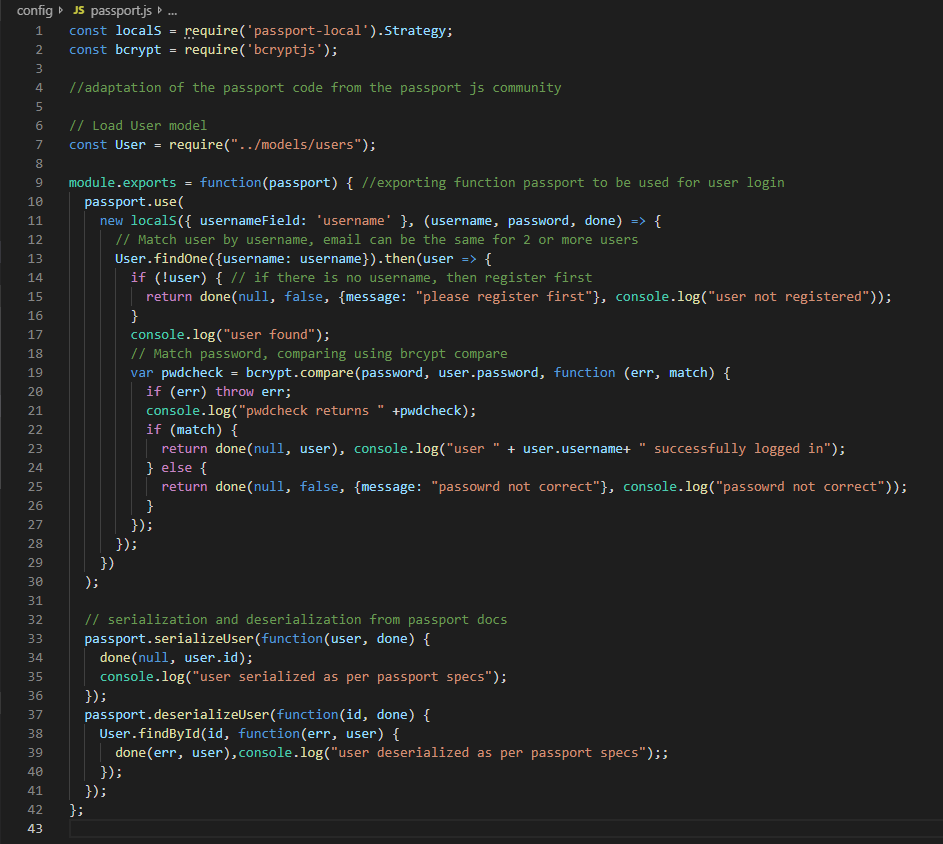


Figure 4 – User login code [ref. passport]

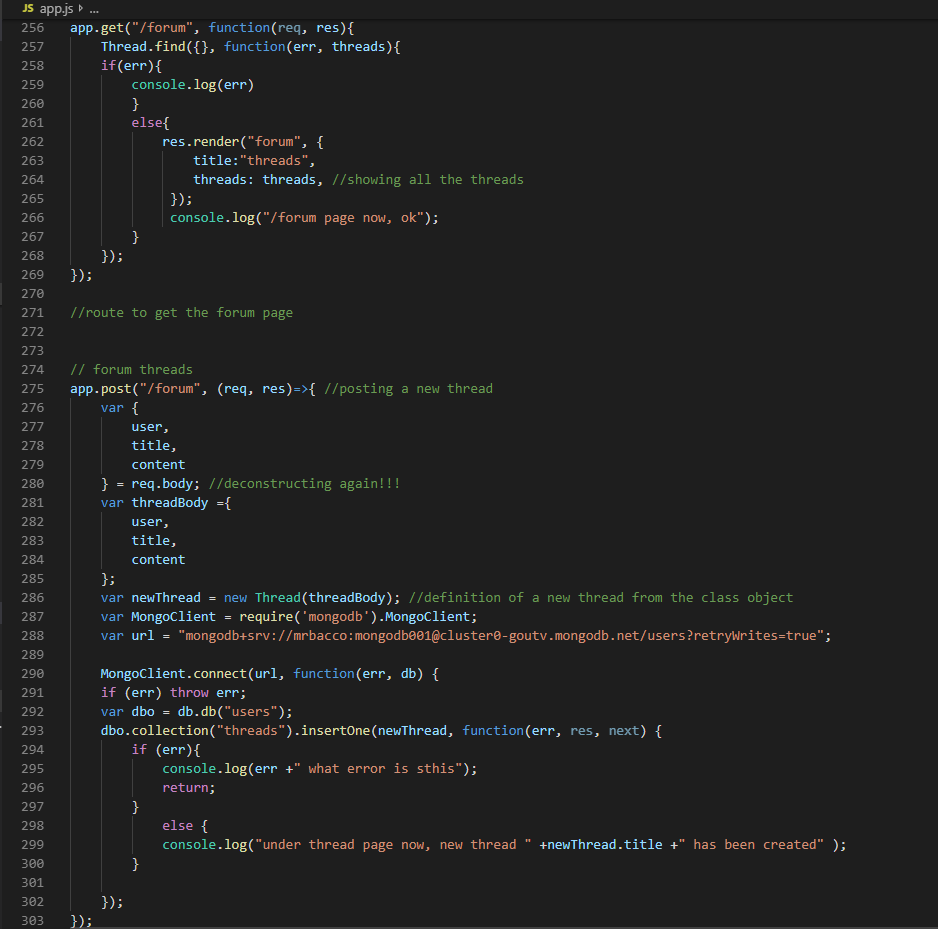


Figure 5 – Threads logic implementation

*Figure 5* shows the threads login implementation, very similar to the user creation because it leverages on the similar client-server architecture.

A snapshot of the actual achieved targets up to date, is shown in *Figure 6* [grey lines indicated the percentage of tasks that are achieved, vs blue lines “to be achieved”].



Figure 6 - Interim progress report status

## Deliverables yet to be achieved

As good part of the client-server framework is completed, but some deliverables are not yet achieved.

The login functionality, for example, is the main important yet to be achieved deliverable. Even if the integration with passport has been finished, the testing is not done. So deliverable is not met. The main reason for that is that the integration of the passport code for user login and authentication has resulted a bit more difficult than expected, so it required a bit more time and testing. Fortunately, this is the only deliverable not met up to date according to the original approved project plan [ (Baccolini, 2019)]. There is a plan to overcome this issue as explained in the following section.

The other yet to be achieved deliverables are the “Project Analysis and Design documentation” which is under development and needs to be delivered by July 13, 2019; the final client view part which entails subjecting some of the views [e.g. add users as friend, or like user A post etc … ] only to logged in users, and it is now relying on the prototype made of initial “pug” views; other feature to be implemented is the upload of dynamic images [avatars/user profile images, or images in the thread/forum page]: this is scheduled for Sprint 2 and is currently on time.

Final documentation such as presentation, video, and final report and testing report that will be completed end of July before the submission deadline for the code and final documentation, are yet to be delivered as they require proper progress in the development.

As a conclusion, beside the login functionality that needs to be completed on-time now, there are no others red flags up to date: a summary of the yet to be achieved tasks is shown in *Figure 7*.



Figure 7 - Deliverables to be achieve from project plan

# Main problems encountered

## Major issues

The biggest issue, developed into a red flag for the project, encountered so far is the is the login functionality using passport.js (Passport, n.d.) that resulted more difficult than expected. The development team has allocated more time for the testing since few deliverables of Setup phase have been accomplished before the expected delivery time. The login is currently under test, and there will be a release for the solution in Sprint 2.

## Minor issues

A minor issue encountered was creation of “pug” views. Although has presented problems during the implementation and the development, has been overcome now, thanks to some online tutorials on pug [ (Copes, 2019), (Pug, n.d.)].

Another issue which is overcome now, encountered during the view prototyping integration, was how to integrate and display images using “pug” on node.js/express.js. Also, this issue has been overcome by researching the problem (Gunda, 2018).

# Expected stumbling blocks

The major expected stumbling block is how to organize and render the “views” for registered/logged in users and not registered/not logged in users. As the webapp is supposed to provide a functionalities like users can add other users to their friends list only if logged in, and/or like other users’ threads etc …, this involves a series of controls that are new concept as far as development is concerned, therefore an adequate level of prior research is required to overcome this potential block.

There are few sources that have been identified to develop a networking like app (Dickey, 2015).

# Focus on rubric

In a scale from 1 to 5, where one is “bad” and five is “excellent”, the development team feels to give the following scores against the rubric.

Communication: good [4]. The development team is keeping close and continuous communication with the supervisor either via email or one to one after the weekly class: questions on direction have been asked and suggestions/feedback from the supervisor have been provided.

Writing and presentation results: fair to good [3.5]: so far the team has made the effort of documenting clearly and in details the development process and the project management following academic style, and even if the dissertation is not perfect, since the beginning and according to supervisor feedback the result were more than acceptable and very limited changes, if none, were suggested.

Complexity/Coding skills: fair [3]. The team is making a huge effort in trying to use sophisticated software development, in an environment very new and still to explore to the fullest.

Innovation: fair to good [3.5]: the concept is itself innovative and the team is trying to develop it with limited access to existing resources: for example, in the case of the login, the team has developed its own login system prior to integrate “passport”.

Technology: fair to good [3.5]. The team has so far developed a fair amount of code that exploits the used technology to good capabilities. The team is aware of the learning process that is underway with this project.

Completeness: fair [3]. The project is not yet ready for commercial implementation, although it is planned to be more than good by final deadline.

Testing/evaluation: good [4]. As the major testing was done on the prototype and the initial release, there is a fair bit of system testing that needs to be performed and documentation to assure the completeness of the application.

Project management: good to excellent [4.5]. The team is putting immense effort since the beginning in creating a detailed project, keeping track of the accomplished vs not yet accomplished tasks versus timelines as defined in the project plan please refer to *Figure 6* for an update on the accomplished tasks up-to-date. In addition, the team is trying to keep evidence of issues encountered and trail of solution adopted to overcome them.

# Summary of project progression

The team is making good headway with this project. Both the project proposal and project specifications have been reviewed and approved. The webapp coding has begun several weeks ago with prototyping and framework, and that was a good idea because it has given a head start on the actual development which has taken some pressure off the development team. In addition to that, the mastering of the chosen programming language [node.js] is getting better every day and has consistently improved since the very beginning. One other particularly rewarding aspect of the project so far, is the growing learning experience, that goes across from the generic software engineering, to documentation creation finishing to a more specific development, troubleshooting and debugging experience.

The Research part of the R&D has played a very important role to bring the team to the good progress as described in previous section. Internet has been, and will still be, the major resource of information, and it has been a great start of the journey so far for the team.

# Graphical representation

Following some screenshots of the actual application as it looks up to today.

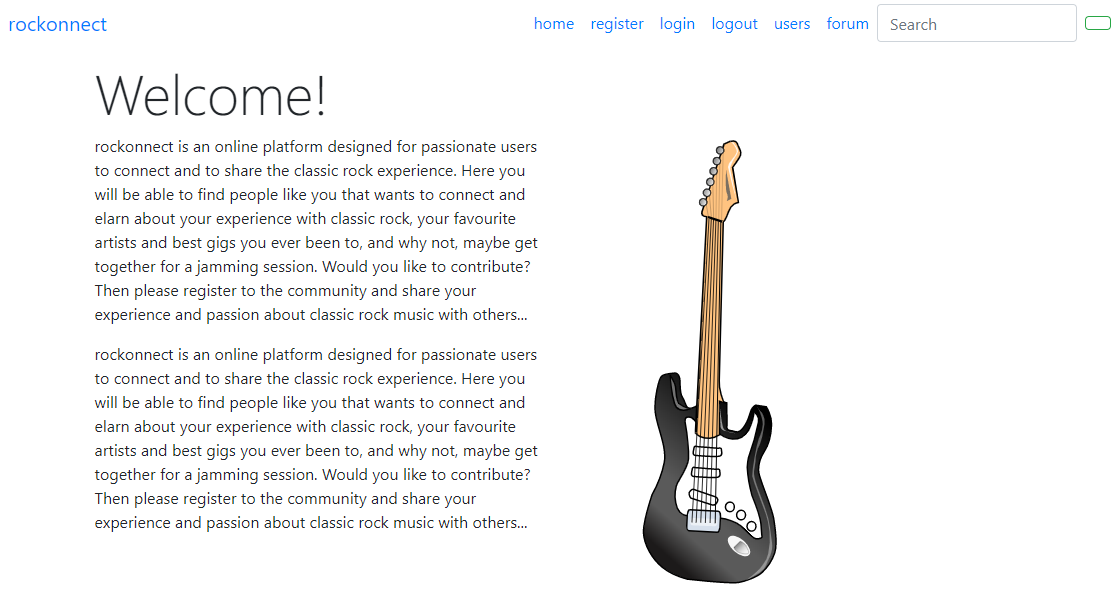


Figure 8 - Homepage

Figure 8shows the actual home page of the web application. Since most of the time is spent on the server side, the actual look & feel adopted is clean and plain and is an adapted (Bootstrap, 2019) template, which has a main menu bar on the top right part of the main page and a home page link on the top left right.

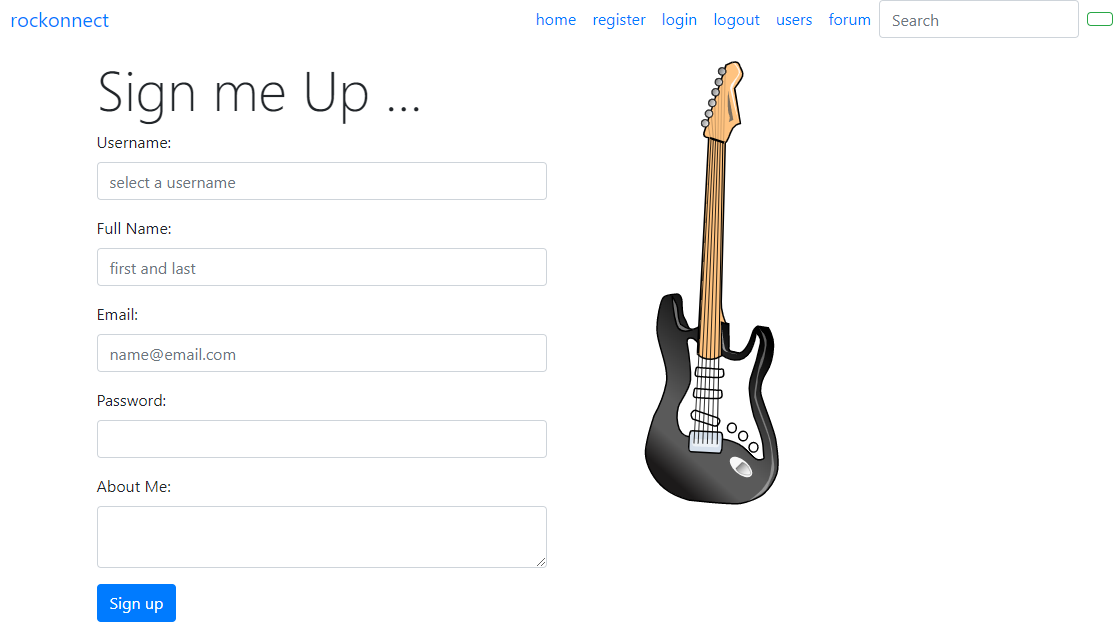


Figure 9 - Register page

Registration page, even thou incomplete cause it misses the upload of pictures, looks like *Figure 9* Login page, shown in *Figure 10*, is based on username and password [passport strategy, please refer to (Passport, n.d.)].

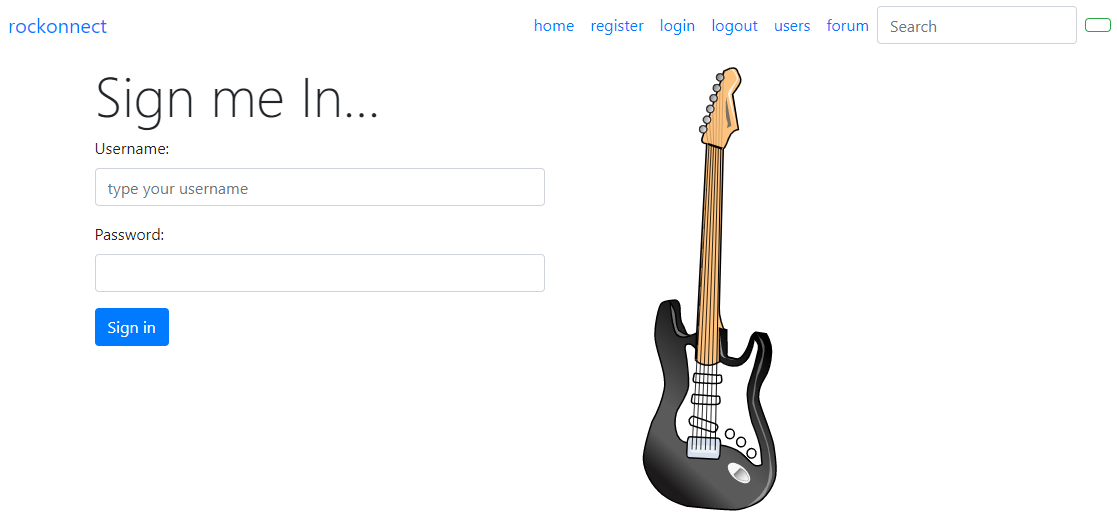


Figure 10 - Login page

*Figure* *11* shows the thread page as it appears for a login user [who can add new posts].

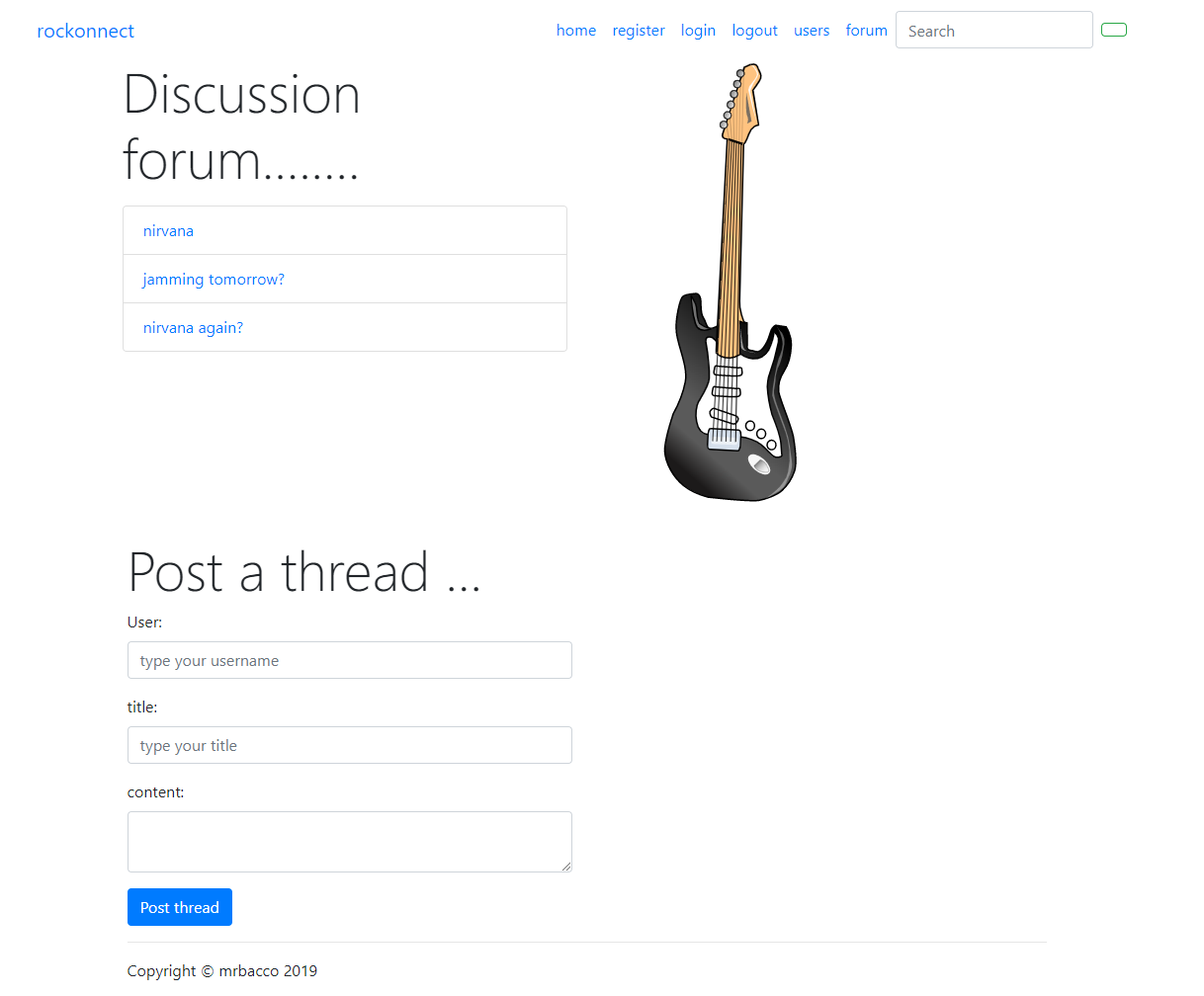


Figure 11 - Discussion page

# Appendix A: list of figures

[Figure 1 – app.js initial framework 6](#_Toc12720600)

[Figure 2 - Routes and database connection 7](#_Toc12720601)

[Figure 3 – User registration code 8](#_Toc12720602)

[Figure 4 – User login code [ref. passport] 9](#_Toc12720603)

[Figure 5 – Threads logic implementation 10](#_Toc12720604)

[Figure 6 - Interim progress report status 11](#_Toc12720605)

[Figure 7 - Deliverables to be achieve from project plan 12](#_Toc12720606)

[Figure 8 - Homepage 14](#_Toc12720607)

[Figure 9 - Register page 15](#_Toc12720608)

[Figure 10 - Login page 15](#_Toc12720609)

[Figure 11 - Discussion page 16](#_Toc12720610)

# Appendix B: Acronyms

|  |  |
| --- | --- |
| CRUD | Create Read Update Delete |
| IDE | Integrated Development Environment |
| R&D | Research and Development |
| webapp | Web Application |

# Bibliography

Baccolini, A., 2019. *Project\_Proposal\_ABACCOLINI\_summer\_2019.docx,* Dublin, Ireland: s.n.

Bootstrap, 2019. *Bootstraps themes.* [Online]   
Available at: https://getbootstrap.com/docs/4.3/examples/  
[Accessed 2019].

Copes, F., 2019. *The Pug Guide.* [Online]   
Available at: https://flaviocopes.com  
[Accessed 2019].

Dickey, J., 2015. *Write Modern Web Apps eith the MEAN stack.* s.l.:Peachpit press.

Erdeljac, A., 2018. *Learn how to handle authentication with Node using Passport.js.* [Online]   
Available at: https://www.freecodecamp.org/news/learn-how-to-handle-authentication-with-node-using-passport-js-4a56ed18e81e/  
[Accessed 2019].

Gunda, D., 2018. *All you need to know about uploading and displaying pictures using Node.js,Express.js,Pug/Jade.* [Online]   
Available at: https://medium.com/@deepika.gunda/all-you-need-to-know-about-uploading-and-displaying-pictures-using-node-js-express-js-pug-jade-d89fbeb19947  
[Accessed 2019].

Hickey, W., 2014. *Classic Rock Started With The Beatles And Ended With Nirvana.* [Online]   
Available at: https://fivethirtyeight.com/features/classic-rock-started-with-the-beatles-and-ended-with-nirvana/  
[Accessed 2019].

Hickey, W., 2014. *The Biggest One-Hit Wonders of Classic Rock.* [Online]   
Available at: https://fivethirtyeight.com/features/the-biggest-one-hit-wonders-of-classic-rock/  
[Accessed 2019].

Passport, n.d. *http://www.passportjs.org/docs/downloads/html/.* [Online]   
Available at: http://www.passportjs.org/  
[Accessed 2019].

Pug, C., n.d. *HTML to Pug converter.* [Online]   
Available at: https://html2jade.org/  
[Accessed 2019].

Pug, C., n.d. *Pug.* [Online]   
Available at: https://pugjs.org/api/getting-started.html  
[Accessed 2019].

Shahid, 2019. *How to Manage Session using Node and Express.* [Online]   
Available at: https://codeforgeek.com/manage-session-using-node-js-express-4/  
[Accessed 2019].

Stackoverflow, 2019. *Stackoverflow.* [Online]   
Available at: https://stackoverflow.com  
[Accessed 2019].

W3School, 1999-2019. *W3Schools learn node.js.* [Online]   
Available at: https://www.w3schools.com/nodejs/default.asp  
[Accessed 2019].