Personal Portfolio

Bradley Zaal
n9177493

Unit: IFB299

Release 1 Artefacts:

Learned the ruby programming language and how to use the rails frameworks by following two online tutorials.	The following Document highlights the tutorials followed: Artefact_1/Artefact_1.pdf	This study was vital for this project as no other group member had used Ruby before. Additionally, only one of the members in the group had sufficient experience in backend programming and one member with sufficient front end experience. These tutorials paved the way for the project and have been continually used to achieve tasks and fix issues. Evidence of Programming: • https://bitbucket.org/dashboard/repositories • https://github.com/freef49/IFB299-Music-School/tree/sign-up • https://github.com/freef49/IFB299-Music-School/tree/basic-login • https://github.com/freef49/IFB299-Music-School/tree/advanced-login • https://github.com/freef49/IFB299-Music-School/tree/modelling-users • https://github.com/freef49/IFB299-Music-School/tree/updating-users • https://github.com/freef49/IFB299-Music-School/tree/master
Created/Revised and Implemented the data model used for by the website for the first release.	The following Document highlights the design and implementation process: Artefact_2/Artefact_2.pdf	This was the foundation for release 1. With the database tables and ruby models completed, the rest of the MVC architecture could be completed. From this the forms, user signup/login and validation where created. Evidence of Programming: • https://github.com/freef49/IFB299-Music-School/tree/sign-up • https://github.com/freef49/IFB299-Music-School/tree/basic-login • https://github.com/freef49/IFB299-Music-School/tree/advanced-login • https://github.com/freef49/IFB299-Music-School/tree/modelling-users • https://github.com/freef49/IFB299-Music-School/tree/updating-users
Created and edited views used by the website. Including adding pictures, maps and forms. Routes and root route were created for all controllers.	The following Document highlights the design and implementation process for some features in the routes and views folder: Artefact_3/Artefact_3.pdf	The implementation of static pages allows users to interact with the website. The forms allow new users to signup, edit and login while the other static pages allow users to view general information about the Mika music school. Evidence of Programming: • https://bitbucket.org/brad_z/sample_app/branch/static-pages • https://bitbucket.org/brad_z/sample_app/branch/filling-in-layout • https://github.com/freef49/IFB299-Music-School/tree/master

Implementation of a module for the Users of the website, which includes creation, update and deletion operations (CRUD)	The following Document highlights certain design and implementation processes for the User model and controller: Artefact_4/Artefact_4.pdf	This implementation of (CRUD) for the users allowed users to be created, edited, deleted in and from the database. Without this, the core of release 1 could not be completed. With the completion of this module, the website was fully functional and a majority of release 1 tasks completed. Evidence of Programming: • https://github.com/freef49/IFB299-Music-School/tree/sign-up • https://github.com/freef49/IFB299-Music-School/tree/basic-login • https://github.com/freef49/IFB299-Music-School/tree/advanced-login • https://github.com/freef49/IFB299-Music-School/tree/modelling-users • https://github.com/freef49/IFB299-Music-School/tree/updating-users • https://github.com/freef49/IFB299-Music-School/tree/master
Developed automated test cases and fixtures for the website (including random database seeds).	The following Document highlights certain design and implementation processes for the automated testing and seeding: Artefact_5/Artefact_5.pdf	These test cases served multiple purposes towards this project. The main reason for these test cases were to see how new additions to the website were affecting the website. After each new addition, the tests were run to make sure nothing was broken and all assertions were completed. Additionally, these tests also behaved as a means to test the website and how closely they match the acceptance criteria. However, as website testing was also completed by another group member using the website, the test cases did not cover all acceptance criteria. Finally, the random seeds allowed other members to view how the website displayed users and also allowed other members to view the website without creating their own account every time. Test cases were changed continuously throughout the entire production of release 1. Evidence of Programming: • https://bitbucket.org/brad_z/sample_app/branch/static-pages • https://github.com/freef49/IFB299-Music-School/tree/sign-up • https://github.com/freef49/IFB299-Music-School/tree/basic-login • https://github.com/freef49/IFB299-Music-School/tree/advanced-login • https://github.com/freef49/IFB299-Music-School/tree/modelling-users • https://github.com/freef49/IFB299-Music-School/tree/updating-users • https://github.com/freef49/IFB299-Music-School/tree/commenting_code • https://github.com/freef49/IFB299-Music-School/tree/fixing_reg_form • https://github.com/freef49/IFB299-Music-School/tree/master

Release 2 Artefacts:

Refined data model planned as final data model design.	The following Document highlights the design process of the new database design being implemented in release 2: **Artefact_6/Artefact_6.pdf**	The revision of the database model was required in order to successfully complete sprints 3 and 4. The new design allows all members of the development team to see and revised the database design that will be used in release 2. The UML diagram also allowed the client team to easily view and understand how our teams wants to structure the data. This design is to be implemented into the website.
Developed the final database design including the generation of new tables and all migrations.	The following Document highlights the migrations, schema and relationships of the project as designed for release 2: **Artefact_7/Artefact_7.pdf**	With the tables and database elements completed, all the necessary controllers, views and forms can be completed for the project in order to complete release 2. The backend of the website is fundamental to the website architecture. With a concrete back end up and running, all remaining requirements can be added. Evidence of Programming: • https://github.com/freef49/IFB299-Music-School/tree/DatabaseDesign • https://github.com/freef49/IFB299-Music-School/tree/AddingReleationShips • https://github.com/freef49/IFB299-Music-School/tree/FreePlayRoomController
Implementation of modules for teacher availability, teacher skills, student preferences, student free play rooms and admin booking interview	The following Document highlights the CRUD operations of the modules as designed for release 2: **Artefact_8/Artefact_8.pdf**	These modules were the foundation of release 2. With these modules actively working, users can now add, edit and delete information about themselves and also book interviews. All modules implemented have index, new, edit and delete functionality. This completes most of the tasks required for release 2. With users now being able to edit and add things about themselves; class booking and timetabling can now be started. Evidence of Programming: • https://github.com/freef49/IFB299-Music-School/tree/AddingReleationShips • https://github.com/freef49/IFB299-Music-School/tree/FreePlayRoomController • https://github.com/freef49/IFB299-Music-School/tree/AddingTeacherAvailabilities • https://github.com/freef49/IFB299-Music-School/tree/AddingRegFormElements • https://github.com/freef49/IFB299-Music-School/tree/StylingFreePlayRoomAndAvailabilities • https://github.com/freef49/IFB299-Music-School/tree/TeacherBookInterview • https://github.com/freef49/IFB299-Music-School/tree/AddingPreferences • https://github.com/freef49/IFB299-Music-School/tree/AddingSkills

Developed the test data for the database tables (db:seed) based on the client feedback.	The following Document highlights the overhaul of database seed script and the randomly generated test data produced for release 2 (Noting that the generated data is always random and most likely will be different upon release if seeded again): **Artefact_9/Artefact_9.pdf**	Test data plays a critical role on the development and display of the project. The test data created, is almost completely random for both teachers and students. This randomness allows the clients to view and understand how the website will look and behave before actually deploying and letting real world people sign up. Additionally, this lets other group members see how the website reacts and what effects their changes will have on displaying the test data. Team members also don't have to worry about created sudo random entries to test their changes. Finally, the test data also acts as somewhat of website testing as during deployment of the seed, errors can be found if some seeds fail to validate. Evidence of Programming: • https://github.com/freef49/IFB299-Music-School/tree/AddingSeedData
Refined automated test cases, fixtures and before actions for the project to match new requirements.	The following Document highlights the additions added to the rails automated test fixtures, controllers and models. Also discussed is the addition of limiting controller actions to different users: **Artefact_10/Artefact_10.pdf**	Application testing plays an extremely significant role in application development. After the review of release 1, it was thoroughly discussed and established that our project testing was severely lacking and needed much attention in release 2. The test scripts added reflect the intended use and behaviour of fundamental processes expected of the project. Although other testing such a selenium will be taking place for release 2, in built rails test cases and assertions allow for future changes to be immediately tested with the 'rails test' command. This allows developers of the project to continuously make changes and review the effects in a more streamlined manor without requiring another member to have to re-test fundamental building blocks of the website. Evidence of Programming: • https://github.com/freef49/IFB299-Music-School/tree/AutomatedTesting