

Android Project

EAD CA2



March 31, 2019

Anthony Neville & Ceiran Mccabe

X00127486 & X00132492

Contents

[App Introduction 2](#_Toc4941920)

[Visual Studio API 2](#_Toc4941921)

[Azure App Service & DB 2](#_Toc4941922)

[Azure DevOps 3](#_Toc4941923)

[GitHub 4](#_Toc4941924)

[Android Studio 4](#_Toc4941925)

[5](#_Toc4941926)

[5](#_Toc4941927)

[Swagger 6](#_Toc4941928)

[Tests 6](#_Toc4941929)

[Resources and Tutorials Used 6](#_Toc4941930)

[*API Development* 6](#_Toc4941931)

[*Android development* 7](#_Toc4941932)

[*Internationalizing the app* 7](#_Toc4941933)

[*Espresso Testing* 7](#_Toc4941934)

[Final Product 7](#_Toc4941935)

[Project Links 7](#_Toc4941936)

[Internationalize 7](#_Toc4941937)

[Snapshots 8](#_Toc4941938)

# App Introduction

For our EAD Android CA2 project we have decided to create a song database like the idea behind IMDb. We have chosen the best songs of all time from ten different genres and compiled them to make an eclectic list everyone can enjoy and relate to.

# Visual Studio API

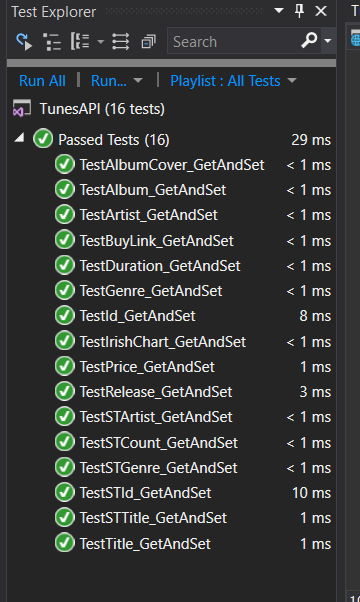
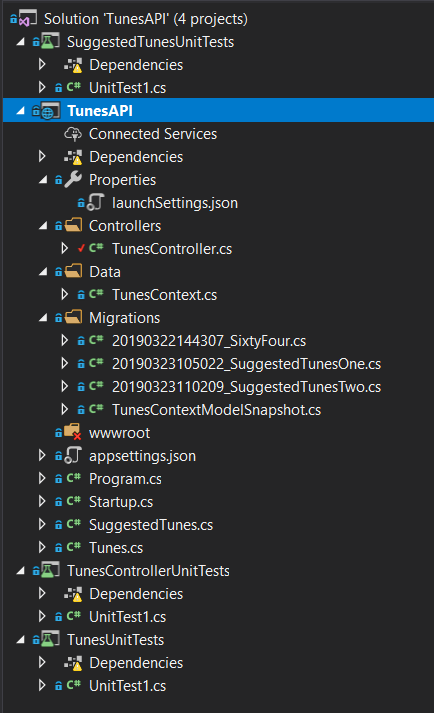
 We created our API instance using Entity Framework in C# on Visual Studio. Here we created a list of 60 songs which we persisted to an Azure SQL DB. We queried the DB on azure using API calls like GET, POST, PUT and DELETE to manipulate the data as wee needed. We also wrote some unit tests the likes of which we learned in EAD1 module to test some methods and variables.

Figure 2 Unit Tests

Figure 1 API Directory

# Azure App Service & DB

Figure 3 Azure App Service Details

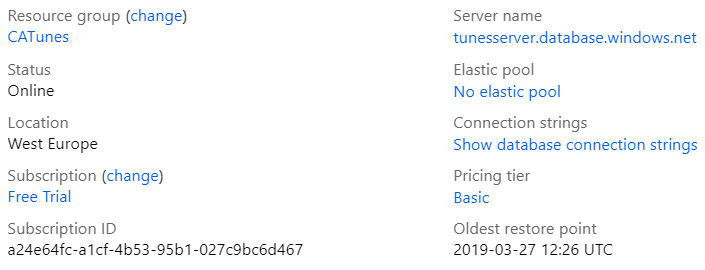
We hosted our API on Microsoft Azure and connected an SQL DB to our app service to store our app date. We used Ceiran’s account initially which was working and set up with an azure devops CI/CD pipeline for deployment. Unfortunately, his credits ran out and his account was disabled we set up a new app service with Anthony’s account which set us back a few hours but once we got that set up and connected to our C# API we were back in business.

Figure 4 Azure SQL DB and Sever Details

# Azure DevOps

Figure 7 Unit test run in CI Pipeline

Figure 6 Tasks in CI Pipeline

Figure 5 CI Pipeline

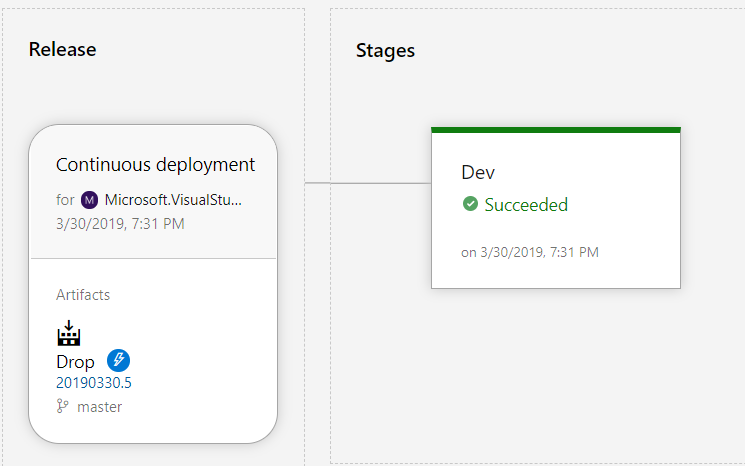


Figure 8 CD Pipeline Stages and deployment slots

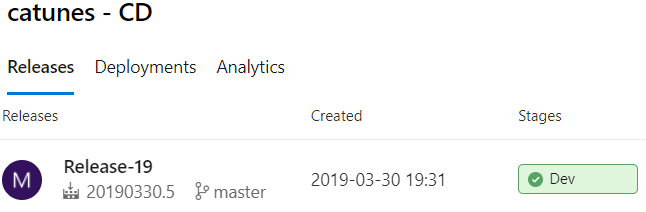
Once we had merged code to our master branch that was committed and pushed it to our GitHub account from our individual branches it azure devops would automatically build a Continuous integration pipeline for our project and then release our app in a Continuous delivery pipeline for deployment. Once both these succeeded our app was build and deployed on azure.

Figure 9 CD Pipeline

# GitHub

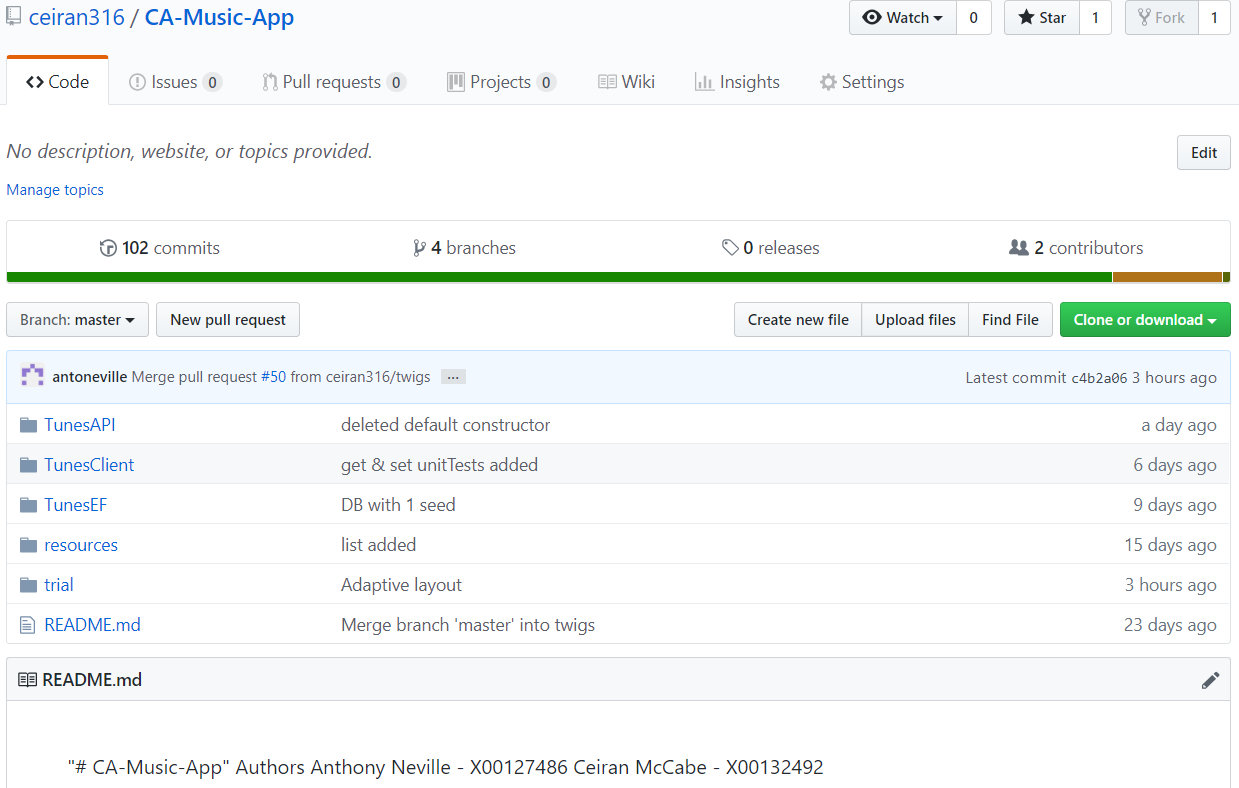


Figure 10 GitHub Repository

This is our git hub repository four our app which is connected to our Azure App Service. We had to make this private for security reasons due to our azure account login details being used in one of our files.

# Android Studio

Figure 1 Android Activity\_main.xml

# 

Figure 13 Example of adding language flags

Figure 12 Emulator Example

# 

Figure 14 Android repository and TunesAdapter.java class

# Swagger

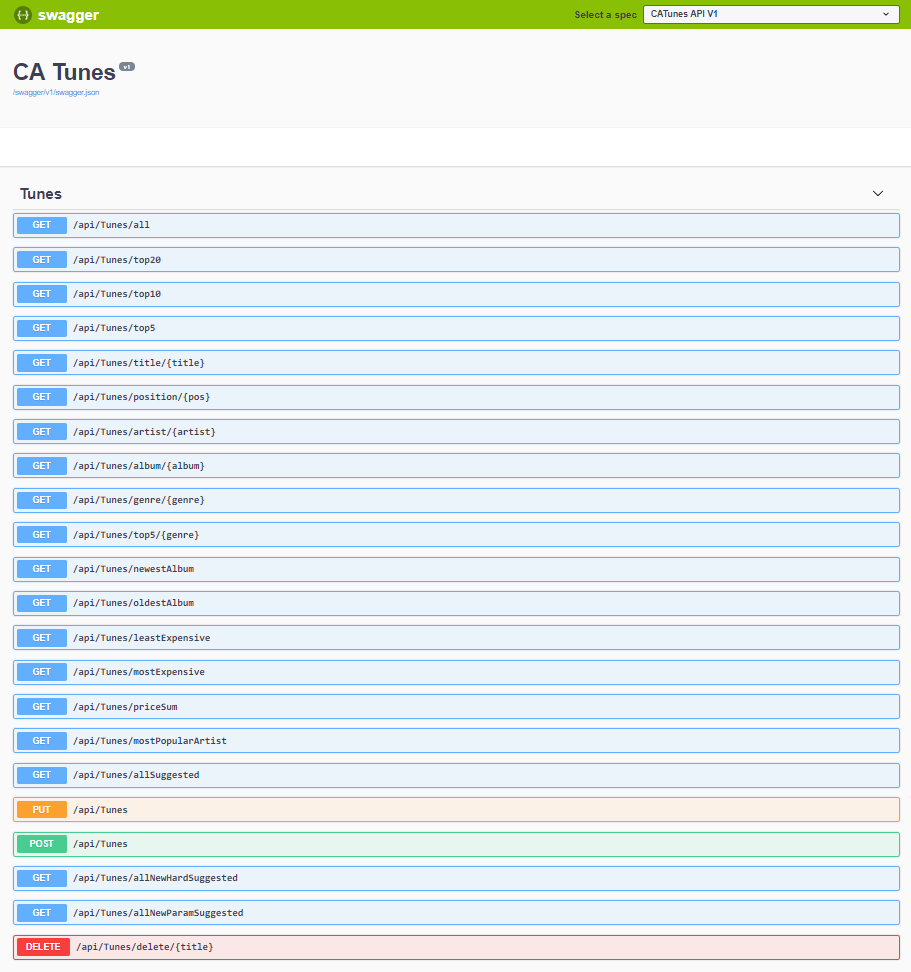


Figure 15 Swagger Displaying out REST API calls

# Tests

# Resources and Tutorials Used

## API Development

* <https://elearning.it-tallaght.ie/course/view.php?id=367>
* <https://docs.microsoft.com/en-us/visualstudio/data-tools/insert-new-records-into-a-database?view=vs-2017>
* <https://docs.microsoft.com/en-us/ef/ef6/fundamentals/working-with-dbcontext>

## Android development

* <https://www.udemy.com/the-complete-android-oreo-developer-course/>
* <https://developer.android.com/studio/intro>
* https://www.androidhive.info/2012/02/android-custom-listview-with-image-and-text/

## Internationalizing the app

* <https://developer.android.com/training/basics/supporting-devices/languages.html#java>
* <https://developer.android.com/guide/topics/resources/localization#creating-alternatives>

## Espresso Testing

* <https://www.pluralsight.com/guides/testing-in-android-with-espresso-part-1>
* <https://www.vogella.com/tutorials/AndroidTestingEspresso/article.html>

# Final Product

## Project Links

GitHub: <https://github.com/ceiran316/CA-Music-App>

Azure App Service: <https://portal.azure.com/#@ittd.onmicrosoft.com/resource/subscriptions/db4aae8c-4122-472f-a689-b96d7cdafc46/resourceGroups/catunes/providers/Microsoft.Web/sites/catunes/appServices>

Azure SQL DB: <https://portal.azure.com/#@antonevillehotmail.onmicrosoft.com/resource/subscriptions/a24e64fc-a1cf-4b53-95b1-027c9bc6d467/resourceGroups/CATunes/providers/Microsoft.Sql/servers/tunesserver/databases/tunesdb/overview>

Azure DevOps: <https://dev.azure.com/X00127486/CA-Music-App-2>

SwaggerUI: <https://catunes.azurewebsites.net/index.html>

Figure 16 App in English

## Internationalize

Figure 18 App in Russian

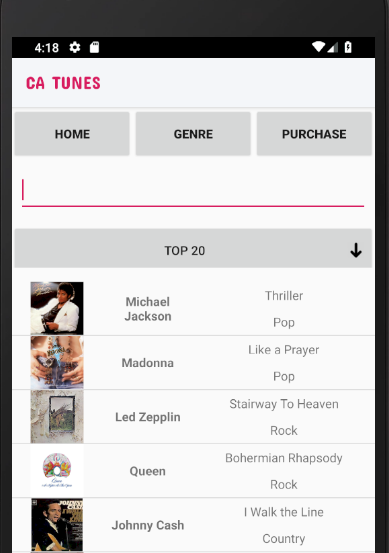


Figure 17 Languages setting changed to Russian

## Snapshots

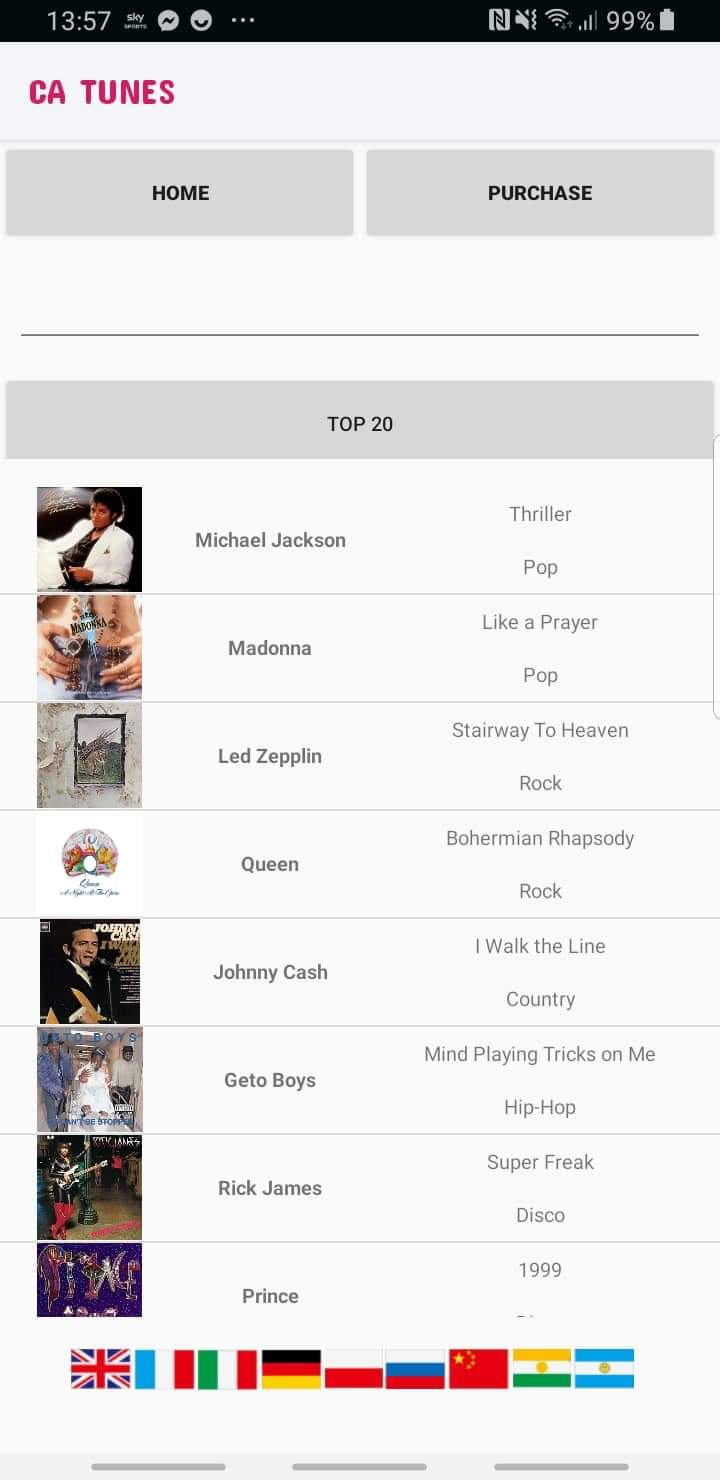
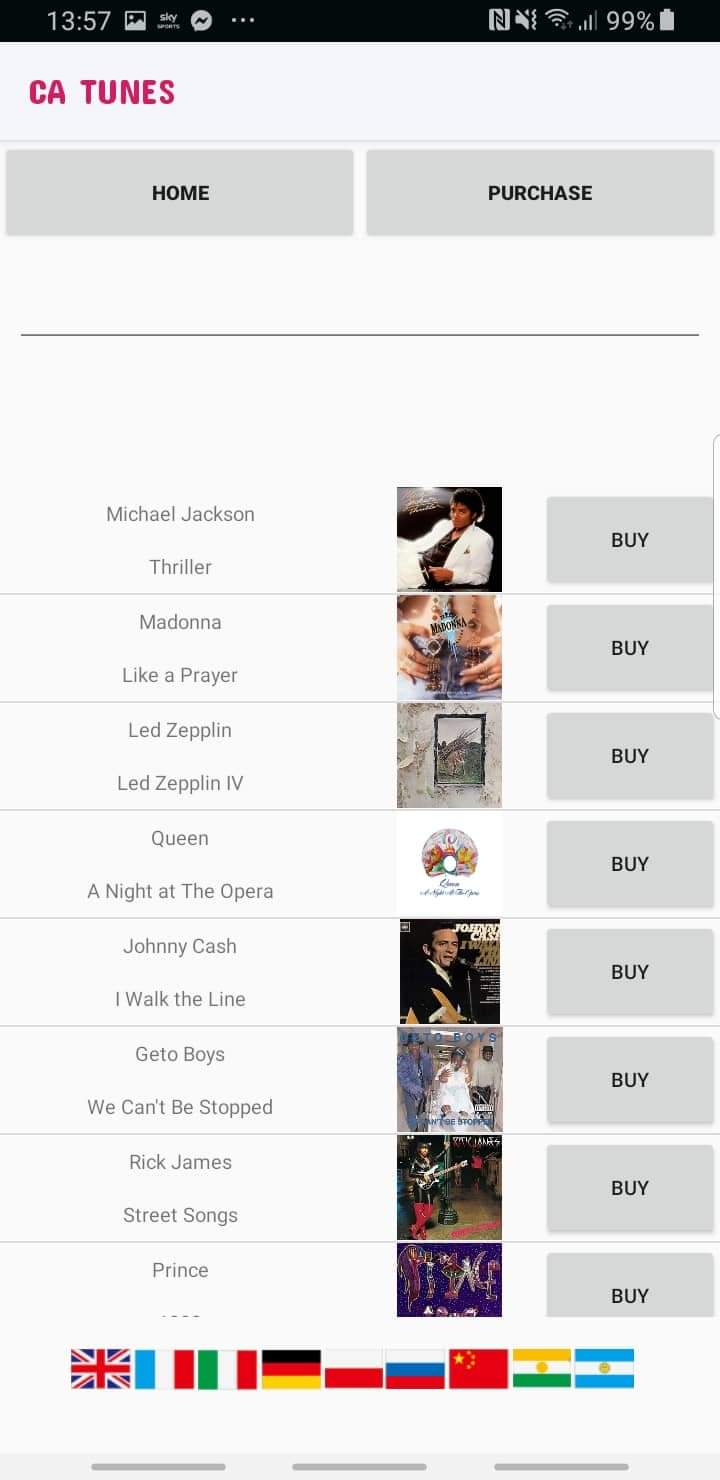


Figure 20 Purchase Screen

Figure 21 Thriller Buy Button Link

Figure 19 Home Screen