



# FBLA Navigator 2.0 README

---

By Makenna Swartz, Daniel Simboli, Adil Ansari

Easton Area High School

Easton, PA

FBLA Navigator is a mobile application for Android and iOS devices. To login into the app, users must first fill out a form to join FBLA or authenticate using Google. Once in the app, users can view current events, competitive events, community service opportunities and fundraisers in the calendar or on the event pages. From both locations there are also options to sign up for upcoming events and meetings. On the "About FBLA" page there are links to state and national websites as well as a Q and A section. To connect with their chapter, users have access to links to social media, as well as a place to email their director directly from the application. Lists of members in the users chapter, as well as local, state, and national officer teams are also provided if the user wishes to contact them instead. Officers and directors may interact with the application by logging on to the website, [fbllamanager.me](http://fbllamanager.me), where they can create and post new events and messages, as well as track users and meeting attendance. FBLA Navigator is currently error free, however a bug reporting system is implemented if the user later discovers an issue.

## Features

---

- Designed for Android and iOS devices
- Options to login with email and password or through Google if user is already a member, or create an account to join FBLA
- Information about FBLA as well as links to important FBLA websites
- Q and A section of frequently asked questions
- A calendar full of current and competitive events
- A list of competitive events with links to their guidelines
- Options to sign up for current events and sign into meetings
- Ability for directors and officers to track meeting attendance from website
- Ability to email club director directly from application
- Links to local, state, and national social media accounts
- A feed of tweets from FBLA Twitter accounts
- Lists of local, state, and national officer teams
- List of members in the chapter as well as their contact information
- Ability to edit personal information as well as profile picture
- Officers and directors can create and post events and messages to the message board from the website
- Push notifications when a new post is created
- A bug reporting system

## Folder Layout

---

### /Compiled App

Contains our compiled, signed app in APK form for Android platforms

### /Documentation

Contains PDF overview of the app along with screenshots and explanations of all functionality

### /Source

Contains Visual Studio 2019 solution file

## How To Run

---

This mobile application was developed in C# using Visual Studio 2019 and the Xamarin Platform on Microsoft Windows. Contained within the competition submission is a folder named "CompiledApp" that contains a signed APK that was created for Android and iOS phones and emulators. Simply install the APK to your mobile device and run.

## Build Instructions

---

Visual Studio Requirements:

- Visual Studio 2019 Windows Community Edition or Visual Studio for Mac
- Android Platform 28 SDK (Pie) or iOS emulator

In order to build you will need Visual Studio or Visual Studio for Mac. Upon opening the Visual Studio solution it will immediately download all necessary packages from Nuget. You will need to execute a debug version of either the `FBLANavigator.Android` or `FBLANavigator.iOS` project either on a simulator or by connecting an Android mobile device that has Developer Options and Enable USB debugging turned on.

## Resources Used

---

Menu Icons from Icons8 - <https://icons8.com/>

Background Image from Pexels - <https://www.pexels.com/>

National Officer Photos from FBLA-PBL - <https://www.fbla-pbl.org/fbla/officers/>

## Software and Services Used

---

GitHub - <https://github.com/>

GitHub is an online source hosting service based around the Git version control system. We utilized Github to store source code revisions during this project.

ZenHub - <https://zenhub.com/>

ZenHub was used to create a product backlog, set goals and assign tasks to assure that we met the deadline and that all team members new their responsibilities.

GitKraken - <https://www.gitkraken.com/>

Gitkraken was utilized to manage code revisions, resolve merge conflicts, and test experimental branch features.

Instabug - <https://instabug.com/>

We utilize Instabug to provide comprehensive bug reporting and in-app feedback from our users during beta testing. Instabug automatically attaches steps to reproduce the bug, network request logs and view hierarchy inspections with each bug report. It also allows users to record videos demonstrating their problem.

OneSignal - <https://onesignal.com/>

One signal is used to send out push notifications.

Microsoft Visual Studio 2019

IDE for developing Xamarin.Forms applications in C#

Balsamiq - <https://balsamiq.com/>

Balsamiq was used to create wireframes and UI mockups so that we had a reference to work off of.

## Additional Software Components

---

Newtonsoft.Json by James Newton-King - <https://www.nuget.org/packages/Newtonsoft.Json/>

Json.NET is a popular high-performance JSON framework for .NET

Microsoft.AppCenter by Microsoft - <https://azure.microsoft.com/en-us/services/app-center/>

This package contains the basic functionalities that all App Center services use to communicate with the backend, including reports on analytics and crashes.

MonkeyCache by James Montemagno - <https://github.com/jamesmontemagno/monkey-cache>

A simple caching library to cache any data structure for a specific amount of time in any .NET application. Additionally, offers simple HTTP methods for caching web request data.

## Xamarin

Xamarin.Essentials by Microsoft - <https://www.nuget.org/packages/Xamarin.Essentials/>

Xamarin.Essentials: a kit of essential API's for your apps

Xamarin.Forms by Microsoft - <https://www.nuget.org/packages/Xamarin.Forms/>

Build native UIs for iOS, Android, UWP, macOS, Tizen and many more from a single, shared C# codebase

Xamarin.FFImageLoading by Daniel Luberda, Fabien Molinet - <https://www.nuget.org/packages/Xamarin.FFImageLoading/>

Xamarin Library to load images quickly and easily

## Syncfusion

Syncfusion Essential UI Kit for Xamarin by Syncfusion Inc. - <https://quizlet.com/447443583/easton-pa-fbla-study-set-flash-cards/>

This Essential UI Kit repository contains elegantly designed XAML templates for Xamarin.Forms apps. These templates are compatible with Android, iOS, and UWP platforms, and use the MVVM design pattern to provide trouble-free integration.

Syncfusion.Xamarin.Core by Syncfusion Inc. - <https://www.nuget.org/packages/Syncfusion.Xamarin.Core/>

This package contains common classes and interfaces that are used in other Syncfusion Xamarin UI controls

Syncfusion.Xamarin.SfBusyIndicator by Syncfusion Inc. - <https://www.nuget.org/packages/Syncfusion.Xamarin.SfBusyIndicator/>

The Syncfusion Busy Indicator for Xamarin.Forms control provides over 10 built-in animations that can be displayed within the application. It is used to indicate busy status during app loading, data processing, and more.

Syncfusion.Xamarin.SfSchedule by Syncfusion Inc. - <https://www.nuget.org/packages/Syncfusion.Xamarin.SfSchedule/>

Syncfusion Schedule for Xamarin.Forms is used to schedule and manage appointments through an intuitive user interface to efficiently plan and manage events or appointments.

Syncfusion.Xamarin.SfListView by Syncfusion Inc. - <https://www.nuget.org/packages/Syncfusion.Xamarin.SfListView/>

Syncfusion ListView for Xamarin.Forms is a feature rich list control that renders a set of data items with views or custom templates. It has many features like grouping, sorting, filtering, paging, swiping, multiple selection, dragging and dropping, and layout types. This control has also been optimized to work with large amounts of data.

Syncfusion.Xamarin.SfComboBox by Syncfusion Inc. - <https://www.nuget.org/packages/Syncfusion.Xamarin.SfComboBox/>

The Syncfusion Combo Box for Xamarin.Forms is used to select an item by typing a value or selecting a value from the list.

## References

---

- “FBLA-PBL.” FBLA-PBL, [www.fbla-pbl.org/](http://www.fbla-pbl.org/).
- “FBLA Competitive Events: Academic Competitions for High School Students.” FBLA, <https://www.fbla-pbl.org/fbla/competitive-events/>.
- “FBLA National Officers - FBLA-PBL.” FBLA, <https://www.fbla-pbl.org/fbla/officers/>.
- “Easton, PA FBLA Study Set.” Quizlet, 2019, <https://quizlet.com/447443583/easton-pa-fbla-study-set-flash-cards/>.
- BeckettMorsch. “BeckettMorsch/QuizzicalFBLA.” GitHub, <https://github.com/BeckettMorsch/QuizzicalFBLA/blob/master/README.md>.

## License

---

The MIT License (MIT)

Copyright (c) 2019

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.