Karta Dimoti Mobile Application

General Description:

Karta Dimoti mobile application was designed for all residents of every municipality in Greece. Each person has his own QR code where he can be scanned by partner businesses and services and in the future collect points from them. He can also see a multitude of businesses and services in his city and be informed about announcements and events that concern him. Finally, the application also works from the employer's point of view, who can check his business or the service to which he is subcontracted and scan the clients himself.

File Structure:

90% of the source code can be found in the src folder. There you will found 9 seperated folders:

* “api”
  + Client.js : has the base url for the api calls.
  + Auth.js : has the apis you call for user authentication.
* “auth” : Here you can find the screens and the functions for user authentication.
  + LoginScreen.js : Login Screen for the user.
  + PinScreen.js : OTP Screen for user confirmation via email.
* “Business” : Here you can find of the files that are needed for the business version of the application.
  + BusinessProfileAddNewImage.js : The screen to change the business profile photo.
  + BusinessProfileScreenNavigator.js : The Stack Navigator for the business profile.
  + BusinessDetails.js : Show the details of every business partner.
  + BusinessListNavigator.js : Stack Navigator for parteners list and their details.
  + BussinessNavigator.js : The main navigator of the business version of the app.
  + HistoryScreen.js : Screen to show the users that have been scanned by the business.
  + HomeNavigatorBussiness.js : Stack navigator for business homepage.
  + HomeScreenBussiness.js : Homescreen of the business version.
  + ProfileScreenBussiness.js : Business profile screen.
  + ScannerScreen.js : Scanner screen for the business.
* “components” : All the components that have been user from the application
  + BusinessScreenImageComponent.js : Component for changing the business profile screen
  + BussinessComponent.js : Component used to show each partner business in a list.
  + EventComponent.js : Component used to show each registered event in a list.
  + MessageHistoryComponent.js : Component used to show each push notification in a list.
  + RenderModalButtons\_Service.js : Component to populate the modal for switching to service version.
  + RenderModalButtons.js : Component to populate the modal for switching to business version.
  + UserImageComponent.js : Component for changing the user profile screen.
* “Globals”
  + MyContext.js : Initialize the context for the application.
* “History”
  + MessageHistoryScreen.js : Screen to show the list of push notifications sent.
* “NewsAndAnnounc” : Here you will find all the files for news and announc stack.
  + EventsDetailsScreen.js : Screen to display the details of an event.
  + EventsNavigator.js : Stack navigator for events and their details.
  + EventsScreen.js : Screen to show the list of events.
  + NewsDetailsScreen.js : Screen to display the details of an announcement.
  + NewsNavigator.js : Stack navigator for news and their details.
  + NewsScreen.js : Screen to show the list of news.
* “Services” : Here you can find of the files that are needed for the service version of the application.
  + HistoryScreen.js : Screen to show the users that have been scanned from the service.
  + HomeNavigatorServices.js : Stack navigator for the screens of the service homepage.
  + HomeScreenService.js : Homescreen for service.
  + ProfileScreenService.js : Profile screen for service.
  + ScannerScreenService.js : Scanner screen for service.
  + ServiceDetails.js : Screen to show the details of each service.
  + ServicesListNavigator.js : Stack navigator for the service list and their details.
  + ServicesNavigator.js : The main navigator of the business version of the app.
* “User” : Here you can find of the files that are needed for the user version of the application.
  + BussinessScreen.js : Global screen to show the list of the partner businesses.
  + ErrorScreen.js : Error Screen. Currently no use.
  + HistoryNavigator.js : Top tab navigator for user history.
  + HistoryScreenBusiness.js : A screen listing the businesses that have scanned the user.
  + HistoryScreenServices.js : A screen listing the services that have scanned the user.
  + HomeNavigator.js : Stack navigator for the screens of the user homepage.
  + HomeScreen.js : Homescreen for the user and for the whole application.
  + NewsLetterScreen.js : Tab navigator for the newsletter and events screens.
  + NotActiveUser.js : NotActiveUserScreen. Currently no use.
  + ProfileScreen.js : Profile screen for the user.
  + ProfileScreenAddNewImage.js : Screen to change the profile photo of the user.
  + ProfileScreenNavigator.js : Stack navigator for user profile.
  + ReportProblemAddPhotoScreen.js : Screen to upload an image for reporting a problem.
  + ReportProblemNavigator.js : Stack navigator for the report problem action.
  + ReportProblemScreen.js : Screen to report a problem.
  + ScannerScreenHome.js : Screen for user scanner.
  + ServicesScreen.js : Global screen to show the list of the registered services.
  + UserNavigator.js : The main navigator of the user version of the app.

The other 10% is :

* “.env” : Here are the environmental variables of the application such as BASE\_URL.
* “App.js” : The main navigator for the application. Includes the login screen, otp screen and the 3 bodies for the 3 versions of the applications. All of them wrapped by the appropriate providers.

Installation Guide:

* Node Version : v18.16.1
* Expo Version : ~48.0.15

Application can be installed in both android and ios devices. Given that the repository exists in personal computer :

For Android Devices

The command is this : *eas build --platform android --local .*

This will produce an .abb file which must be signed with a keystore and a java bundletool.

After rename the file from .apks to .zip extract it. Take the universal.apk and install it directly to the device.