Mobile Application Development

LECTURE 17

Targetting multiple platforms in Ionic

- There are certain scenarios where we might need to run some code on a particular device.
- For e.g, I only want this code to run if the app is running on Android.
- Ionic provides us way to check which device is it currently and we can use that to run the platform specific code.

Targetting multiple platforms in Ionic

- Ionic provide Platform Service which we can

  use to check which platform we're currently on.

- This helps us write platform specific code.

- https://ionicframework.com/docs/angular/platform

```
import { Platform } from '@ionic/angular';

constructor(private platform: Platform) {}

 this.platform.ready().then(() => {
      if (this.platform.is('android')) {
           console.log('android');
      } else if (this.platform.is('ios')) {
           console.log('ios');
      } else {
           //fallback to browser APIs or
           console.log('The platform is not supported');
             }
      }
```

Handling Issues/Crashes in Production

- Suppose your customer tells you that the mobile app you created crashes on certain scenarios.
- Often the hard problem is figuring out what caused the failure. Why did app crash for you ? It's running perfectly for me.
- There is where Firebase Crashlytics (previously Fabric) comes in.

Crashlytics

- provides Real-time alerts.
- Provides crash analytics. Tells you which error caused how many crashes and on which devices.
- Very easy to setup with Ionic. Just add a cordova plugin, few lines of code and your crashlytics are up. https://ionicframework.com/docs/native/firebase-crashlytics