

Mobile App Development with AppInventor2

Using AppInventor2 for teaching

Two of the aims of the Computer Science curriculum:

“Understand and apply fundamental principles and concepts of computer science, including abstraction, logic, algorithms, and data representation.”

“Be able to analyse problems in computational terms, and have repeated practical experience of writing computer programs in order to solve them.”

From: DfE National Curriculum Programme of Study for Computing

Why do we want to write mobile apps?

- Mobile touch-based devices are now the norm.
- Writing AppInventor2 apps is something you might want or need to do because:
 - It's fun to play with mobile comms technology
 - Mobile apps can be very engaging and personal
 - It meets many of the aims of the curriculum
 - It makes a good introduction to traditional languages

What's stopping us?

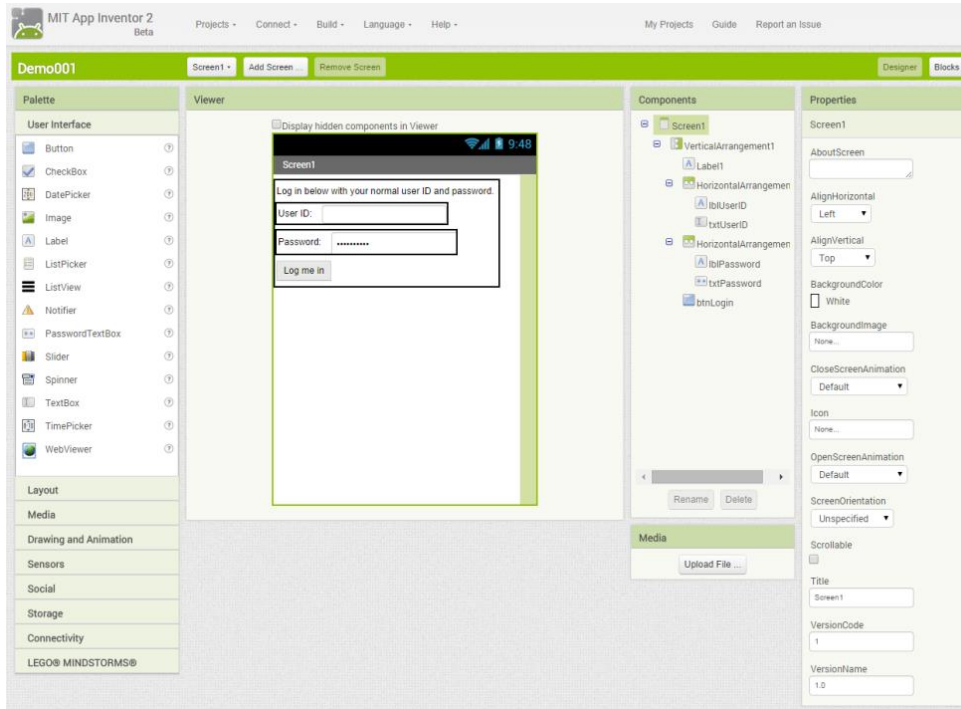
- Your time is the biggest cost and the toughest barrier to picking up new computing skills.
- There is also significant monetary cost when kitting out and maintaining labs.
- Android devices are popular, versatile and relatively inexpensive.
- AppInventor2 has a “low barrier to entry”.

About AppInventor2

- AppInventor2 is a web-based coding environment for Android apps only.
- There is no support for Apple iOS devices
- It has a user interface designer and a separate code “blocks” designer.
- The “blocks” system is similar to the Scratch approach.
- You will need a (free) Google account for each student.

Mobile App Development with AppInventor2

App Inventor 2 Form design interface

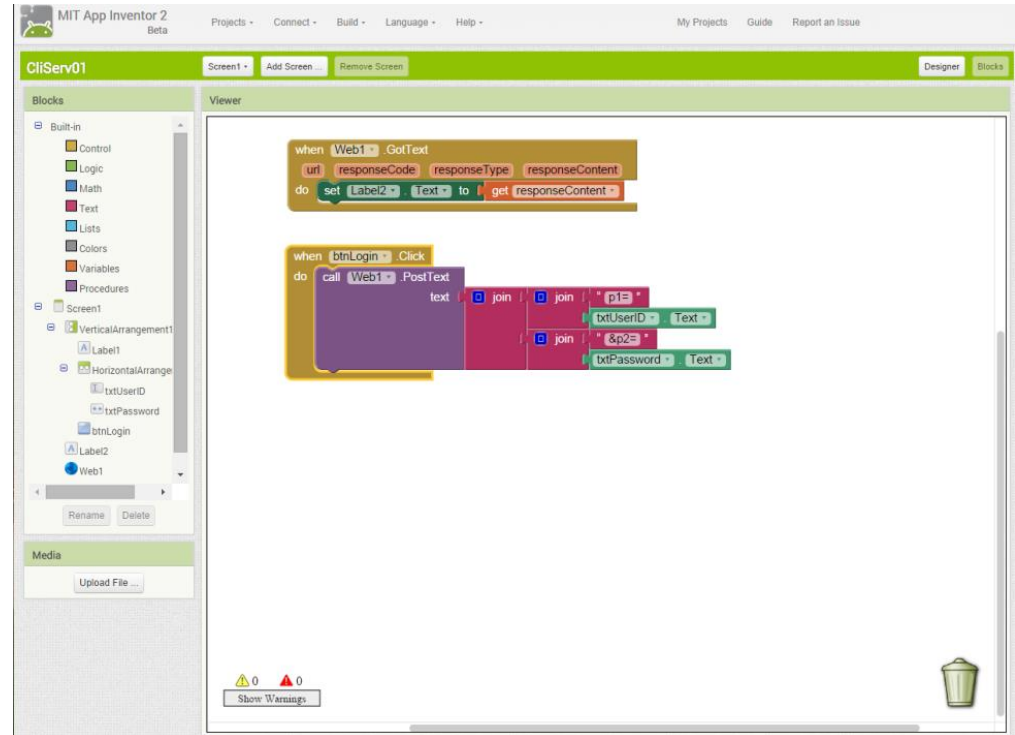


- The user interface builder is simple.
- It allows you to see all the possible user interface elements.
- Great for exploring.

Mobile App Development with AppInventor2

App Inventor 2 Blocks interface

- Code is event driven.
- Each routine is a colour-coded, visible block.
- The blocks fit like jigsaw puzzle pieces.



Mobile App Development with AppInventor2

Blocks in detail

```
when Web1 .GotText
  url  responseCode  responseType  responseContent
do
  set Label2 .Text to get responseContent
```

When a server replies, display the response in a label.

```
when btnLogin .Click
do
  call Web1 .PostText
  text
  join join " p1= "
  txtUserID .Text
  join " &p2= "
  txtPassword .Text
```

When a login button is pressed, send the username and password text field entries to a web server.
NB better ways to do this...

Getting and setting persistent data with TinyDB

Navigating in a simple 2-page app

Calling a PHP service to read an RSS feed

Using a pick list to dynamically view Google Maps locations

Using AppInventor-to-Javascript communication within a Web component

What we have seen

- We hope this has increased your interest in mobile development with free online tools.
- One of the big aims of this talk is to show you that “light-weight” development options still have plenty of power and are relevant to the curriculum.
- AppInventor2 is surprisingly flexible and a good choice for prototyping and some development.

App Inventor 2 - <http://ai2.appinventor.mit.edu/>

- Apache Cordova - <http://cordova.apache.org/>
- IOS development - <https://developer.apple.com/devcenter/ios/index.action>
- Android development - <http://developer.android.com/index.html>
- Get a Google account - <https://accounts.google.com/signup>

A quick demonstration?

- From nothing, we could:
 - Make a “speak the input” app
 - Build an app that makes a sound when clicked (think kittens here)
 - Make a form that takes text input and posts it to a server
 - Create social media feed viewers
 - Build animations or touch/tilt based games
 - See the AppInventor2 site for many ideas – Gallery is good!