K.S.Rangasamy College of Arts & Science
(Autonomous)
Ksr Kalvi Nagar,Tiruchengode - 637215.
Namakkal Dist. Tamil Nadu.

Ishare

Issue #112
December 2017

Department of computer sciecne
& Applications

Ishare

Editorial

We would like to wholeheartedly thank our honorable Chairman, Vice-Chairman, Executive Director and Principal for their continuous encouragement and constant support for bringing out the magazine. We profoundly thank our Head of the Department for encouraging and motivating us to lead the magazine a successful one right from the beginning. Ishare serves as a platform for updating and enhancing upcoming technologies in Information and Communication. We are grateful to all the contributors to this magazine so far. The magazine has been sent to almost 60 Institutions in and around Tamilnadu. So far we have received feedbacks and appreciations from various Institutions.

We would be very pleased to receive your feedbacks. Please send your feedbacks to ishare@ksrcas.edu

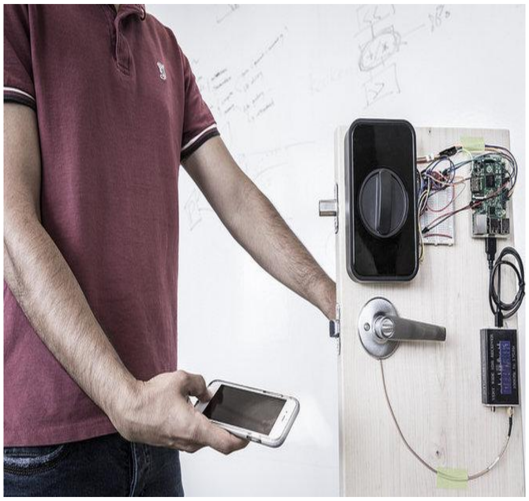
By,

Editorial Board

CONTENTS

Send Passwords Securely Through Your Body

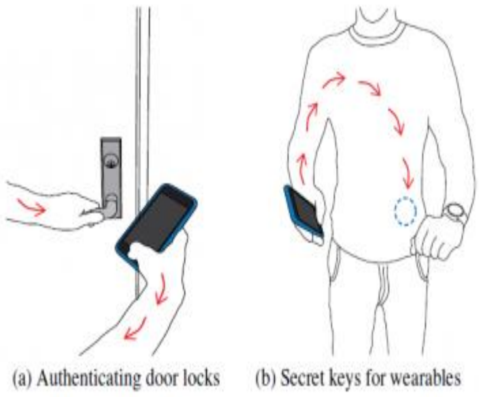
Sending a password or secret code over airborne radio waves like WiFi or Bluetooth means anyone can eavesdrop, making those transmissions vulnerable to hackers who can attempt to break the encrypted code.



Now, University of Washington computer scientists and electrical engineers have devised a way to send secure passwords through the human body — using benign, low-frequency transmissions generated by fingerprint sensors and touchpads on consumer devices.

“Fingerprint sensors have so far been used as an input device. What is cool is that we’ve shown for the first time that fingerprint sensors can be re-purposed to send out information that is confined to the body,” .

Potential applications for on-body transmissions include securely sending information to door locks, glucose sensors or other wearable medical devices.



These “on-body” transmissions offer a more secure way to transmit authenticating information between devices that touch parts of our body — such as a smart door lock or wearable medical device — and a phone or device that confirms our identity by asking us to type in a password.

This new technique, which leverages the signals already generated by fingerprint sensors on Smartphone’s and laptop touchpad’s to transmit data in new ways.

“Let’s say I want to open a door using an electronic smart lock, I can touch the doorknob and touch the fingerprint sensor on my phone and transmit my secret credentials through my body to open the door, without leaking that personal information over the air.”

The research team tested the technique on iPhone and other fingerprint sensors, as well as Lenovo laptop trackpads and the Adafruit capacitive touchpad. In tests with 10 different subjects, they were able to generate usable on-body transmissions on people of different heights, weights and body types. The system also worked when subjects were in motion — including while they walked and moved their arms.

“We showed that it works in different postures like standing, sitting and sleeping,” said co-lead author Vikram Iyer, a UW electrical engineering doctoral student. “We can also get a strong signal throughout your body. The receivers can be anywhere — on our leg, chest, hands — and still work.”

The research team from the UW's Networks and Mobile Systems Lab systematically analyzed Smartphone sensors to understand which of them generates low-frequency transmissions below 30 megahertz that travel well through the human body but don't propagate over the air.

The researchers found that fingerprint sensors and touchpad's generate signals in the 2 to 10 megahertz range and employ capacitive coupling to sense where our finger is in space, and to identify the ridges and valleys that form unique fingerprint patterns.

Normally, sensors use these signals to receive input about your finger. But the UW engineers devised a way to use these signals as output that corresponds to data contained in a password or access code. When entered on a Smartphone, data that authenticates your identity can travel securely through our body to a receiver embedded in a device that needs to confirm who we are.

Their process employs a sequence of finger scans to encode and transmit data. Performing a finger scan correlates to a 1-bit of digital data and not performing the scan correlates to a 0-bit.

The technology could also be useful for secure key transmissions to medical devices such as glucose monitors or insulin pumps, which seek to confirm someone's identity before sending or sharing data.

The team achieved bit rates of 50 bits per second on laptop touch pads and 25 bits per second with fingerprint sensors — fast enough to send a simple password or numerical code through the body and to a receiver within seconds.

This represents only a first step, the researchers say. Data can be transmitted through the body even faster if fingerprint sensor manufacturers provide more access to their software. The research was funded by the Intel Science and Technology Center for Pervasive Computing, a Google faculty award and the National Science Foundation.

'TELEPATHY'

Robot 'Telepathy' Could Make Self-Driving Cars Safer

- Are you nervous about entrusting your life to a self-driving car? What if you could telepathically communicate with the vehicle to instantaneously let it know if it makes a mistake?
- That is the ultimate promise of technology being developed by a team from Boston University and the Computer Science and Artificial Intelligence Laboratory (CSAIL) at the Massachusetts Institute of Technology.
- Tech uses brain signals to automatically correct a robot's errors.

- Using a so-called brain-computer interface (BCI) to communicate with a robot is not new, but most methods require people to train with the BCI and even learn to modulate their thoughts to help the machine understand, the researchers said
- By relying on brain signals called "error-related potentials" (ErrPs) that occur automatically when humans make a mistake or spot someone else making one, the researchers' approach allows even complete novices to control a robot with their minds.

Working with machines

This technology could offer an intuitive and instantaneous way of communicating with machines, for applications as diverse as supervising factory robots to controlling robotic prostheses, the researchers said.

When humans and robots work together, you basically have to learn the language of the robot, learn a new way to communicate with it, and adapt to its interface.

In this work, we were interested in seeing how you can have the robot adapt to us rather than the other way around.

The system uses EEG brain signals to detect if a person notices robots making a mistake.

In the study, the researchers described how they collected electroencephalography (EEG) data from volunteers as those individuals watched a common type of industrial humanoid robot, called Baxter, decide which of two objects to pick up.

This data was analyzed using machine-learning algorithms that can detect ErrPs in just 10 to 30 milliseconds

This means results could be fed back to the robot in real time, allowing it to correct its course midway, the researchers said.

Refining the system

The system's accuracy needs significant improvement, the team admitted. In real-time experiments, the bot performed only slightly better than 50/50, or chance, when classifying brain signals as ErrPs. That meant that nearly half the time it would fail to notices the correction from the observer.

And even in more leisurely, offline analysis, the system still got it right only roughly 65 percent of the time, the researchers said.

But when the machine missed an ErrP signal and failed to correct its course (or change course when there was no ErrP), the human observer typically produced a second, stronger ErrP.When we analyze that offline, we found that the performance boosts by a lot, as high as 86 percent, and we estimate we could get this upwards of 90 percent in the future.

So the next step is to actually detect those in real time as well and start moving closer towards our goal of actually controlling these robots accurately and reliably on the fly. Doing this will be tricky, though, because the system needs to be told when to look out for the ErrP signal, the researchers said. At present, this is done using a mechanical switch that gets activated when the robot's arm starts to move.

A secondary error won't be created until after the robot's arm is already moving, so this switch won't be able to signal to the system to look for an ErrP, the researchers said. This means the system will have to be redesigned to provide another prompt, they added.

The study is well-written, said Klaus-Robert Müller, a professor at the Technical University of Berlin, who was not involved with the new research but has also worked on BCIs that exploit these error signals. But, he said using ErrPs to control machines is not particularly new and he also raises concerns about the low ErrP classification ratesthe group achieved.

José del R. Millán, an associate professor of Switzerland, said he agrees that the performance of the group's ErrP decoder was low. But he thinks the approach they've taken is still "very promising," he added. Millán's group has used ErrP signals to teach a robotic arm the best way to move to a target location. In a 2015 study published in the journal Scientific Reports, Millán and his colleagues described how the arm in their work starts by making a random movement, which the human observer decides is either correct or incorrect.

Through a machine-learning approach called reinforcement learning, the error signals are used to fine-tune the robot's approach, enabling the bot to learn the best movement strategy for a specific target.

Millán said using ErrP to control robots could have broad applications in the future.

I see it in use for any complex human-machine interaction where most of the burden is on the machine side, because of its capacity to do tasks almost autonomously, and humans are simply supervising.

FUTURE INVENTIONS

Cell phone inventor Martin Cooper claims the communication devices used on Star Trek motivated him to invent cellular phones. Harry Potter has an invisibility cloak and the Romulans in Star Trek make their starships disappear using cloaking devic3es.

This fictional technology has inspired scientists to try and make invisibility a possibility. In fact, recent developments in nanotechnology could make invisible cloaking a reality in the near future.

We see things because light reflects off objects. Black objects absorb light and this absence of light is seen as darkness which also helps us to detect objects.

For an object to become invisible it cannot reflect or absorb light. If light could be bent around an object without any reflection or casting of shadows then we would not see it.

Ideally, we would only see what is directly behind the object. It sounds strange but scientists have created a new class of matter that can do this.

Metamaterials are nanostructures that have a negative refractive index, which means they can control how light waves are bent, reflected and absorbed.

But these meta-materials were limited in size. They couldn't be made larger than a few microns and only manipulated certain frequencies of light. So they could only make invisible what you couldn't see anyway.

However, a team of scientists at the University of Central Florida have discovered a method for creating "large-area fabrication" of these meta-materials for the full spectrum of light.

As described in their research published in the journal of Advanced Optical Materials, this means that any object covered in this fabric would become invisible.

Cyborg Beetle

Advances in microelectronics is making yesterday's science fiction into tomorrow's future inventions. Scientists at the University of California have implanted beetles with miniaturized electronics that allows the insects to be controlled.

Research presented at a conference in Italy demonstrated a new technology for creating "cyborg insects".

The flying beetles receive wireless signals from a mobile transmitter that controls the insect. Varying impulses trigger an appropriate response from the beetle so that it flies according to it's handler.

The insect can be made to take-off, land, hover, or follow a given flight path.

Previous research concentrated on moths but giant flower beetles were easier to operate and could carry larger payloads because of their size.

Further advances with nanobots and miniaturization will allow heat sensors and cameras to be embedded into the bugs. The insect could then be used for surveillance or search and rescue missions.

Undersea Resort

L Bruce Jones is a submarine inventor who designs and sells private luxury submarines. Following in the footsteps of his rocket scientist father, and his grandfather who invented self-elevating jack-up rigs and created the world's first containerized shipping company; Bruce has invented the world's first undersea resort.

Accessible by two elevators from the surface, the resort is being built on the seabed surrounding a private Fiji island in the South Pacific. The underwater facilities will include 24 luxury state rooms, a restaurant and bar, library, conference room, wedding chapel, underwater spa, and the ultra-luxury Nautilus Suite with stunning undersea views.

Electrical Clothing

One of the future inventions that could greatly impact our lives are nanoribbons.

Rubber films developed by engineers at Princeton University could power mobile devices and other electronic devices.

The silicone sheets are embedded with ceramic nanoribbons (piezoelectric ribbons) that generate electricity when flexed, converting mechanical energy to electrical energy.

Materials made of this material, such as shoes, would harvest electrical energy created from walking and power everything from an ipod to a pacemaker.

The nanoribbon strips are so narrow that 100 strips fit side-by-side in a space of a millimeter. The strips are then embedded into clear sheets of silicone rubber to create a chip.

These sheets could be woven into fabric and placed against any moving area on the body to create electricity.

For example, a vest made from this material could take advantage of breathing motions to generate energy.

Nanoribbons are highly efficient in converting about 80% of mechanical power into electricity.

Needle-less Injection:

This future invention is a device for delivering medication and vaccinations through the skin.

As an alternative to injecting a needle, micro-poration is a painless method of transferring medication (intraepidermal) into the body using laser technology.

A handheld laser creates micro pores in the epidermis of the skin for the transfer of molecules. It has a familiar comparison to the "needle-less" device used by Dr. "Bones" McCoy on Star Trek.
The popular sci-fi series has inspired more than a few new inventions including the "laser" and the cell phone.

This new micro-poration technology is painless to use and requires no supervision to administer. The interfaced controls regulate the dosage.

T-Shirt Gadgets

Scientists at Standford University are developing future inventions using e-Texiles.
This new class of flexible, stretchable and lightweight clothing would function as rechargeable batteries.

Cotton and polyester fabrics are dyed with an ink made from carbon nanotubes, which are electrically conductive carbon fibers that are 1/50,000th the width of a human hair.

The electronic properties of the fabric are maintained even when the clothing is washed. Researchers believe that the e-Textiles will eventually allow us to use our clothing as gadgets - talking to a friend through our shirt sleeve or surfing the web on a pant leg.

Tissue Regeneration

One of the future inventions in health care may be the development of a spray gun for regenerating skin tissue. The Armed Forces Institute of Regenerative Medicine (AFIRM), recently established by the U.S. Department of Defense, is funding research into the regrowth of bones, muscles, tendons, nerves and blood vessels.

The University of Pittsburgh's McGowan Institute for Regenerative Medicine will join a consortium of 30 institutes in regenerative research.

Last year, a pathologist at the McGowan Institute regrew the severed fingertips of two patients in their mid sixties. Existing scar tissue was removed by an enzyme. Scarring inhibits regrowth, however cells derived from pig bladders can override the scarring process and attract cells and proteins needed for growth.

Another researcher with Wayne Forest University, is developing an inkjet device with cartridges containing tissue cells with growth factors.

The device would dispense layers of tissue onto deep flesh wounds for healing and regrowth.

For surface wounds, a hand-held sprayer is in development, which sprays immature skin cells (called keratinocytes) onto the skin. Clinical trials with burn victims have been promising. The advantages of this technology compared to "grafting" is that it eliminates patchwork scarring and uses less skin.

"Conventional methods cannot return people to the way they were before" says biochemist Alan Russell, but it's within the grasp of science.

E-Paper

Among the future inventions in e-readers is this paper thin, flexible film that reads like a magazine or newspaper.

LG Display, a manufacturer of thin-film transistor liquid crystal display panels has developed an 11.5 inch flexible e-paper display.

The 0.3 mm thick metal foil substrate resembles a traditional newspaper but operates as an e-reader.

The GIP (Gate-in-Panel) technology promises to be the next generation in digital display technology competing in the e-book market. E-paper's ultra-thin, flexible and lightweight form reduces maintenance and power consumption costs but its scalable size encourages additional applications such as mounting to flat and curved surfaces like walls and pillars.

LaCie Safe

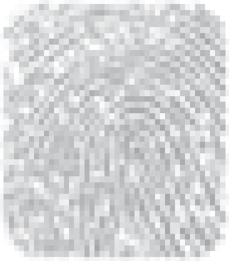
Storing your files on this mobile hard drive gives you multiple levels of protection because it uses advanced encryption and biometric authentication technology.

The Safe uses 128-bit AES encryption (Advanced Encryption Standard) which is the same standard used by governments to protect top secret information.

Encryption converts information that is readable into a mixture of unreadable characters. Decryption processes the encrypted unreadable characters back into a readable format.

The algorithm that encrypts and decrypts the information is known as a cipher. The cipher allows access to the readable information when you enter a password.

Most ciphers will use passwords that are four to eight characters in length, but a 128-bit AES cipher uses a 16 character password which is extremely difficult to hack.



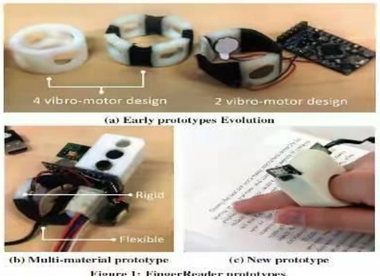
The AES cipher or "Rijndael" (pronounced Rein Dahl) is named after the Belgian inventors Joan Daemen and Vincent Rijmen. Biometric authentication is a technology that recognizes physical or behavioral characteristics such as fingerprints, palm geometry, retina patterns, voice and signature. Fingerprint recognition is the most popular because it's easier to use.

Wearable devices

In a wearable form-factor, it is possible to use the body as a directing and focusing mechanism, relying on proprioception or the sense of touch, which are of utmost importance for people with VI. Yi and Tian placed a camera on shade-glasses to recognize and synthesize text written on objects in front of them, and Hanif and Prevost's did the same while adding a handheld device for tactile cues. Mattar et al. are using a head-worn camera, while Ezaki et al. developed a shoulder-mountable camera paired with a PDA . Differing from these systems, we proposed using the finger as a guide , and supporting sequential acquisition of text rather than reading text blocks . This concept has inspired other researchers in the community

Assistive mobile text reading products

Mobile phone devices are very prolific in the community of blind users for their availability, connectivity and assistive operation modes, therefore many applications were built on top of them: the kNFB kReader1, Blindsight's Text Detective2, ABBYY's Text Grabber3, StandScan4, SayText5, ZoomReader6 and Prizmo7. Meijer's vOICe for Android project is an algorithm that translates a scene to sound; recently they introduced OCR capabilities and enabling usage of Google Glass8. ABiSee's EyePal ROL is a portable reading device, albeit quite large and



(a) Early prototypes Evolution

(b) Multi-material prototype

(c) New prototype

Figure 1: FingerReader prototypes

heavy9, to which OrCam’s recent assistive eyeglasses10 or the Intel Reader11 presents a more lightweight alternative.

Prototypes and products in all three categories, save for [23], follow the assumption that the goal is to consume an entire block of text at once, therefore requiring to image the text from a distance or use a special stand. In contrast, we focused on creating a smaller and less conspicuous device, allowing for intimate operation with the finger that will not seem strange to an outside onlooker, following the conclusions of Shinohara and Wobbrock [21]. Giving the option to read locally, skim over the text at will in a varying pace, while still being able to read it through, we sought to create a more liberating reading experience.

FINGERREADER: A WEARABLE READING DEVICE

FingerReader is an index-finger wearable device that supports the blind in reading printed text by scanning with the finger and hearing the words as synthesized speech. Our work features hardware and software that includes video processing algorithms and multiple output modalities, including tactile and auditory channels. The design of the FingerReader is a continuation of our work on finger wearable devices for seamless interaction, and inspired by the focus group sessions. Exploring the design concepts with blind users revealed the need to have a small, portable device that supports free movement, requires minimal setup and utilizes real-time, distinctive multimodal response. The finger-worn design keeps the camera in a fixed distance from the text and utilizes the inherent finger's sense of touch when scanning text on the surface. Additionally, the device provides a simple interface for users as it has no buttons, and affords to easily identify the side with the camera lens for proper orientation.

Hardware Details

The FingerReader hardware features tactile feedback via vibration motors, a dual-material case design inspired by the focus group sessions and a high-resolution mini video camera. Vibration motors are embedded in the ring to provide tactile feedback on which direction the user should move the camera via distinctive signals. Initially, two ring designs were explored: 4 motor and 2 motor. Early tests with blind users showed that in the 2 motor design signals were far easier to distinguish than with the 4 motor designs, as the 4 motors were too close together. This led to a new, multi-material design using a white resin-based material to make rubbery material for the flexible connections. The dual material design provides flexibility to the ring's fit as well as helps dampen the vibrations and reduce confusion for the user.

Your finger is scanned for minutia, which are the points on a fingerprint where a ridge ends or splits into two. An algorithm extracts the minutia points and creates a template image that is used for authentication.

Vein Identification

This latest technology invention is a biometric identification and security device known as PalmSecure.

It works by identifying the vein pattern in the palms of our hands. Similar to our fingerprints, vein patterns are unique to each individual. The purported advantages of this technology is that it is less expensive, easier to manage, and is more reliable than traditional methods of identification.

3D Printed Car

The latest technology inventions in 3d printing are rapidly changing how things are being made.

It's an emerging technology that is an alternative to the traditional tooling and machining processes used in manufacturing.

At the International Manufacturing Technology Show in Chicago, a little known Arizona-based car maker created a media sensation by manufacturing a car at the show.

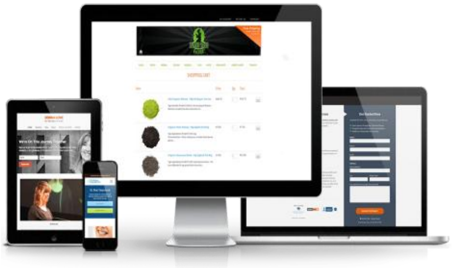
It was a full scale, fully functional car that was 3d printed in 44 hours and assembled in 2 days. The video below shows the car being made.

Top Principles of Effective Web Design

Name: M.Jayapal

Designation: Programmer

1. Purpose



Good web design always caters to the needs of the user. Are your web visitors looking for information, entertainment, some type of interaction, or to transact with your business? Each page of your website needs to have a clear purpose, and to fulfill a specific need for your website users in the most effective way possible.

2. Communication

People on the web tend to want information quickly, so it is important to communicate clearly, and make your information easy to read and digest. Some effective tactics to include in your web design include: organizing information using headlines and sub headlines, using bullet points instead of long windy sentences, and cutting the waffle.

3. Typefaces

In general, Sans Serif fonts such as Arial and Verdana are easier to read online (Sans Serif fonts are contemporary looking fonts without decorative finishes). The ideal font size for reading easily online is 16px and stick to a maximum of 3 typefaces in a maximum of 3 point sizes to keep your design streamlined.

4. Colors

A well thought out color palette can go a long way to enhance the user experience. Complementary colors create balance and harmony. Using contrasting colors for the text and background will make reading easier on the eye. Vibrant colors create emotion and should be used sparingly (e.g. for buttons and call to actions). Last but not least, white space/ negative space is very effective at giving your website a modern and uncluttered look.

5. Images

A picture can speak a thousand words, and choosing the right images for your website can help with brand positioning and connecting with your target audience. If you don't have high quality professional photos on hand, consider purchasing stock photos to lift the look of your website. Also consider using infographics, videos and graphics as these can be much more effective at communicating than even the most well written piece of text.

6. Navigation

Navigation is about how easy it is for people to take action and move around your website. Some tactics for effective navigation include a logical page hierarchy, using bread crumbs, designing clickable buttons, and following the 'three click rule' which means users will be able to find the information they are looking for within three clicks.

7. Grid based layouts

Placing content randomly on your web page can end up with a haphazard appearance that is messy. Grid based layouts arrange content into sections, columns and boxes that line up and feel balanced, which leads to a better looking website design.

8. "F" Pattern design

Eye tracking studies have identified that people scan computer screens in an "F" pattern. Most of what people see is in the top and left of the screen and the right side of the screen is rarely seen. Rather than trying to force the viewer's visual flow, effectively designed websites will work with a reader’s natural behaviour and display information in order of importance (left to right, and top to bottom).

9. Load time

Everybody hates a website that takes ages to load. Tips to make page load times more effective include optimizing image sizes (size and scale), combining code into a central CSS or JavaScript file (this reduces HTTP requests) and minify HTML, CSS, JavaScript (compressed to speed up their load time).

10: Mobile friendly

It is now commonplace to access websites from multiple devices with multiple screen sizes, so it is important to consider if your website is mobile friendly. If your website is not mobile friendly, you can either rebuild it in a responsive layout (this means your website will adjust to different screen widths) or you can build a dedicated mobile site (a separate website optimized specifically for mobile users).

EXCELLENT MOBILE APP BUILDERS

Over the last six months, I've been examining and testing a variety of mobile app builders and mobile back ends. In some cases, the app builders and back ends were part of a single product. In other cases, the app builders or back ends stood on their own.

In this roundup, I'll summarize seven products that are at least partially a mobile app builder. Some have IDEs that run locally on your computer; others give you a Web IDE that lives in the cloud. Some are aimed at enterprise development, others at individual developers or even students.

As we’ll see, they can have almost any level of complexity for the developer, ranging from drag-and-drop simple like EachScape, NSB/AppStudio, and Salesforce1, to providing an API for the developer to code against in Xcode or Eclipse, which is the way Appcelerator supports native SDK developers.

They can target mobile Web, mobile hybrid, or native apps for Android, iOS, and occasionally some of the less popular mobile device platforms, such as Windows Phone. They may integrate with one or more mobile security products. For instance, AnyPresence makes it easy to secure your app with Apperian.

They may be tied to an MBaaS (mobile back end as a service) platform or not. They may or may not be able to consume and modify data from systems of record. If they can, they may require the developer to write a RESTful interface, or they may take care of the connection themselves.

They might reduce the work required to support offline mobile operation with offline/online data synchronization and conflict resolution to checking a few boxes on a form, like Alpha Anywhere, or hand you a box of parts and an assembly diagram with pictures and instructions in Swedish -- sorry, that’s Ikea, but you know what I mean. In between those extremes, they may supply a framework that does part of the work, but leave out the rest and expect you to fill it in with code and forms.

If they support HTML5 apps, they might or might not support your favorite JavaScript framework. If they target native or hybrid apps, they might have their own online app building services, integrate with PhoneGap Build, or rely on you to build apps with the native SDKs on your development box.

They may be priced anywhere from $99 per developer to “low six figures per company per year.” In most cases, I’ve found the prices to be appropriate and the value to be good for the right audience, but a student can’t benefit from an enterprise-level app builder and MBaaS any more than an enterprise developer could get by with a simple app builder with no integration capabilities.

In short, the scope and complexity of these seven products vary widely, and no single product is ideal for everyone. With that in mind, I’ll try to emphasize what sort of developers and designers are most likely to enjoy and be productive with each app builder. Different strokes

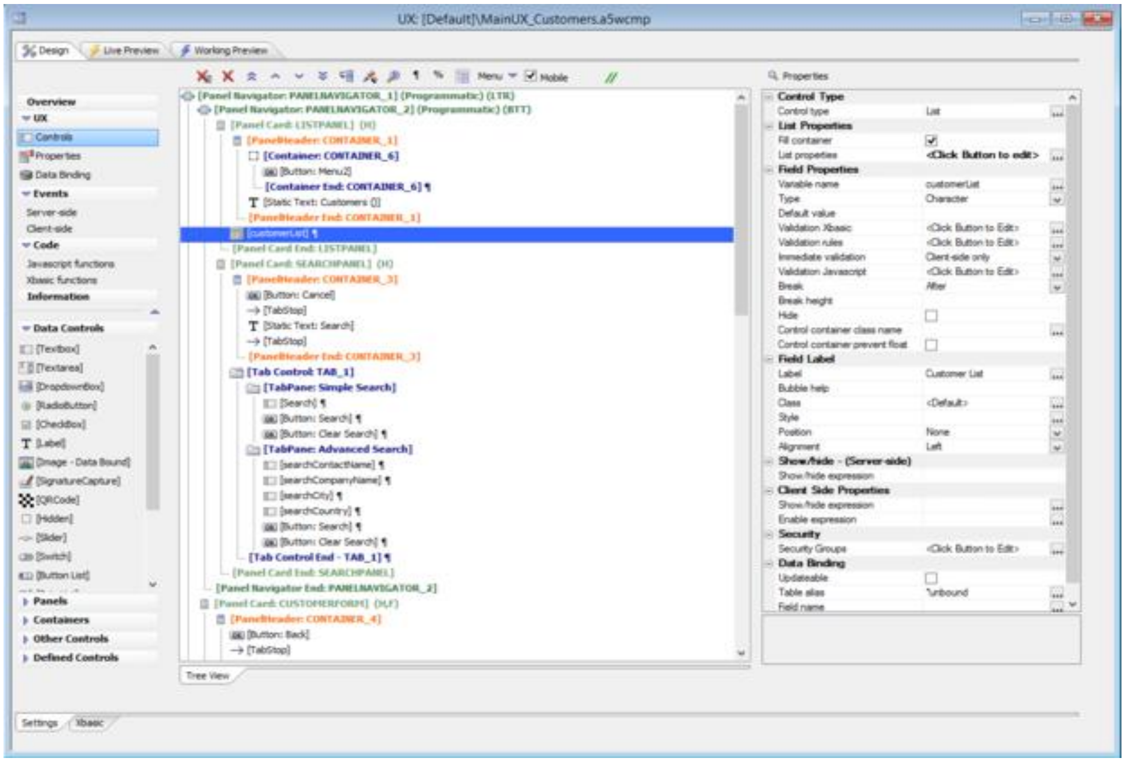
Alpha Anywhere

Alpha Anywhere is a database-oriented rapid development tool that allows developers and designers to create Web and hybrid mobile apps that work offline. It allows less-experienced developers to create sophisticated apps with a combination of configurable components, visual design tools, code-generation “genies,” and a small amount of coding in Xbasic or JavaScript.

The Alpha Anywhere IDE runs on Windows. The tool targets iOS, Android, Windows Phone, and other mobile and desktop clients with HTML5-compliant Web browsers.

Alpha Five was a Web and desktop application development tool with an application and Web server, a PDF report generator, and strong support for dozens of SQL and desktop databases. Alpha Five in turn grew out of Alpha Four, which was an easy-to-use dBase clone.

Alpha Anywhere is an extension of Alpha Five that brings mobile Web and hybrid mobile app development to Alpha Five’s bag of tricks, including the recent addition of support for offline operation, offline-online data synchronization, and intelligent data conflict resolution. The folks at Alpha Software have thought through most of the cases a mobile device might encounter in the field -- trying to consume, modify, and generate server data with unreliable network connectivity -- and have reduced the choices you need to make as a developer to a matter of checking a few boxes.



```
[Panel Navigator: PANELNAVIGATOR_1] (Programmatic) (LTR)
  [Panel Navigator: PANELNAVIGATOR_2] (Programmatic) (BTT)
    [Panel Card: LISTPANEL] (H)
      [PanelHeader: CONTAINER_1]
        [Container: CONTAINER_6]
          [Button: Menu2]
        [Container End: CONTAINER_6] ¶
        [Static Text: Customers {}]
      [PanelHeader End: CONTAINER_1]
      [customerList] ¶
    [Panel Card End: LISTPANEL]
    [Panel Card: SEARCHPANEL] (H)
      [PanelHeader: CONTAINER_3]
        [Button: Cancel]
        [TabStop]
        [Static Text: Search]
        [TabStop]
      [PanelHeader End: CONTAINER_3]
      [Tab Control: TAB_1]
        [TabPane: Simple Search]
          [Search] ¶
          [Button: Search] ¶
          [Button: Clear Search] ¶
        [TabPane: Advanced Search]
          [searchContactName] ¶
          [searchCompanyName] ¶
          [searchCity] ¶
          [searchCountry] ¶
          [Button: Search] ¶
          [Button: Clear Search] ¶
      [Tab Control End - TAB_1] ¶
    [Panel Card End: SEARCHPANEL]
  [Panel Navigator End: PANELNAVIGATOR_2]
  [Panel Card: CUSTOMERFORM] (H,F)
    [PanelHeader: CONTAINER_4]
      [Button: Back]
      [TabStop]
```

Control Type	
Control type	List
List Properties	
Fill container	☑
List properties	**<Click Button to edit>**
Field Properties	
Variable name	customerList
Type	Character
Default value	
Validation Xbasic	<Click Button to Edit>
Validation rules	<Click Button to Edit>
Immediate validation	Client-side only
Validation Javascript	<Click Button to Edit>
Break	After
Break height	
Hide	□
Control container class name	
Control container prevent float	□
Field Label	
Label	Customer List
Bubble help	
Class	<Default>
Style	
Position	None
Alignment	Left
Show/hide - (Server-side)	
Show/hide expression	
Client Side Properties	
Show/hide expression	
Enable expression	
Security	
Security Groups	<Click Button to Edit>
Data Binding	
Updateable	□
Table alias	"unbound
Field name	

To design a mobile UI with Alpha Anywhere, you typically drag panels, containers, and controls onto the tree view of a UX component, and set the properties of each element. This methodology is rapid, although not quite as intuitive as a WYSIWYG designer. You can see a working or live preview at any time, and you can see it simultaneously with the designer if you have enough screen space. The server-side Xbasic of an Alpha Web or mobile component generates HTML5 and JavaScript for rendering by a browser.

On the downside, the Alpha IDE and application server currently run only on Windows; Alpha's application server is proprietary; Alpha lacks a scalable cloud service; and Alpha lacks native mobile client support. Mitigating factors for the lack of native iOS and Android code generation are integration with PhoneGap, and the fact that the Alpha Anywhere mobile components and controls have been crafted to look and feel native.

Overall, I rate Alpha Anywhere very good as a Web, mobile Web, and mobile hybrid development system. The IDE is powerful and easy to use, although it has so many features that it's easy to forget where to find the one you need. The capabilities and integrations are very good.

Alpha has its own JavaScript client framework, which includes support for jQuery, but not for some newer frameworks such as Angular and Backbone. On the other hand, Alpha's templating language has a distinctly Angular feel to it, and you can't really complain about the framework when the system generates almost all of the code for you.

Companies that want to create mobile apps that use SQL and REST data sources will benefit from Alpha Anywhere, especially if ease of development and short time to market are important factors. Alpha's strong support for offline mobile operation, offline-online data synchronization, and intelligent data conflict resolution helps it to stand out in a world where those important issues are too often ignored.

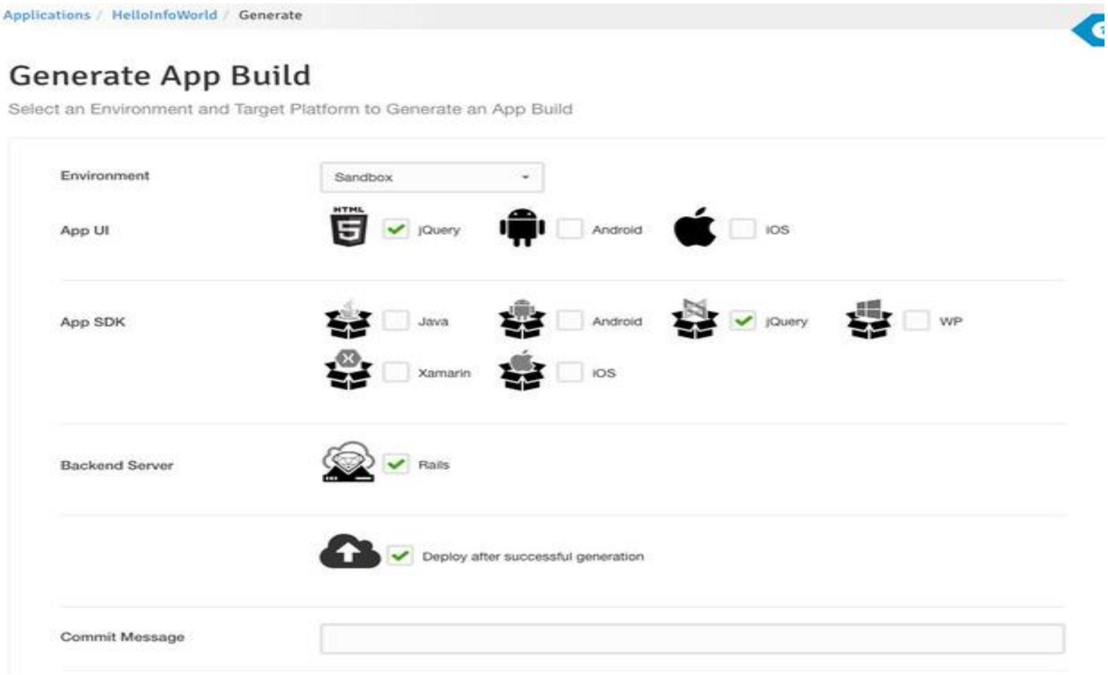
AnyPresence

Both online mobile app builder and back-end service, AnyPresence combines broad client support, useful code generation, and a rich set of options for data storage and enterprise integration. While I originally rated AnyPresence using the criteria of the MBaaS category (with a 9.1 and an Editor’s Choice), it is also an excellent app builder.

AnyPresence builds apps, back-end services, and API gateways. It has an online designer that not only generates back-end and mobile app code, but also customized mobile API code. All of the generated code can be downloaded, edited, and run on compatible platforms. To cite one of AnyPresence’s favorite customer examples, MasterCard has used AnyPresence to enable partners to easily build mobile apps against MasterCard's Open API services.

AnyPresence generates App UIs (or starter kits, if you wish) for jQuery, Android (XML layout), and iOS (Storyboard), and it generates App SDKs for Java, Android, HTML5, Windows Phone, Xamarin, and iOS. It generates back-end servers for Ruby on Rails and Node.js. The AnyPresence environment can generate deployments to Heroku (usually for a back end), to Amazon S3 (usually for HTML5 apps), and to native iOS and Android apps with or without Apperian security.

AnyPresence's app build selection screen. Note the wide assortment of SDKs that can be generated, and the small assortment of prototype app UIs that can be generated.



Applications / HelloInfoWorld / Generate

Generate App Build

Select an Environment and Target Platform to Generate an App Build

Environment: Sandbox

App UI: ☑ jQuery □ Android □ iOS

App SDK: □ Java □ Android ☑ jQuery □ WP □ Xamarin □ iOS

Backend Server: ☑ Rails

☑ Deploy after successful generation

Commit Message:

The AnyPresence design environment lives online and runs in most browsers. In addition to the interface designer, it has a dashboard; a settings screen; screens to create and monitor environments, deployments, and builds; screens to generate and deploy apps, back ends, and SDKs; screens to add and manage data sources and data objects; screens for authorization, roles, and authentication strategy; screens for stock and custom extensions; and a customizable set of themes.

What sets AnyPresence apart is the way the data model integrates throughout the design environment and into all the generated code. The only other app builder that comes close is Alpha Anywhere, which uses SQL databases for its back-end data store.

AnyPresence lacks its own monitoring service, but integrates with third-party services such as Airbrake and New Relic. AnyPresence pricing is high compared to many of its competitors, but offers more value for enterprises that need to integrate their existing systems with mobile applications. It is especially valuable for enterprises that wish to expose their APIs to partners who can in turn use them in their own mobile applications.

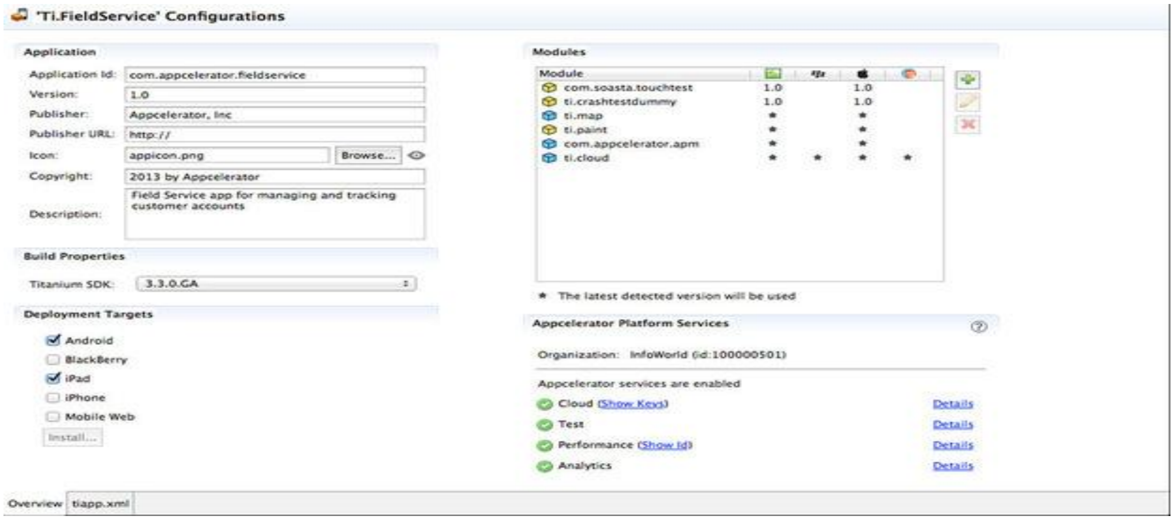
Appcelerator

Appcelerator Titanium has been a player in the mobile development space for several years, with a JavaScript-based development environment that compiles to native code for iOS, Android, and other targets. With the release of Appcelerator Studio 3.3 and Appcelerator Platform 2.0, the company added an MBaaS with about 25 APIs, Node.js support, and online analytics. Also, Appcelerator has published interfaces to its MBaaS that developers can add to apps built with native SDKs, although it hasn't yet supported native SDKs in its own Studio IDE.

As with AnyPresence, originally rated Appcelerator as an MBaaS. Of course, it is also a very good app builder with an excellent IDE.

Appcelerator has multiple frameworks on the client side and multiple API types for the cloud. At the base level on the client, Appcelerator offers the Titanium SDK, which provides an interface between JavaScript and native services. At a higher level, Appcelerator offers the Alloy Framework, which is based on the model-view-controller architecture and contains built-in support for Backbone.js and

Underscore.js. When you create a new client app from Studio, you'll typically generate one that uses Alloy.



'Ti.FieldService' Configurations

Application

Application Id: com.appcelerator.fieldservice
Version: 1.0
Publisher: Appcelerator, Inc
Publisher URL: http://
Icon: appicon.png
Copyright: 2013 by Appcelerator
Description: Field Service app for managing and tracking customer accounts

Build Properties

Titanium SDK: 3.3.0.GA

Deployment Targets

☑ Android
□ BlackBerry
☑ iPad
□ iPhone
□ Mobile Web

Modules

Module				
com.soasta.touchtest	1.0		1.0	
ti.crashtestdummy	1.0		1.0	
ti.map	*		*	
ti.paint	*		*	
com.appcelerator.apm	*		*	
ti.cloud	*	*	*	*

* The latest detected version will be used

Appcelerator Platform Services

Organization: InfoWorld (id:100000501)

Appcelerator services are enabled

- Cloud (Show Keys) Details
- Test Details
- Performance (Show Id) Details
- Analytics Details

On the cloud side, you can reach the Appcelerator Cloud Services using a REST API, via bindings to the Titanium SDK, via Node.ACS, and via native SDKs. The REST API will always work, though it's the least convenient option. You'll mostly want to use REST calls to reach new services that don't yet have bindings to the Titanium SDK.

Appcelerator can call REST and even SOAP services using HTTPClient and its built-in parsing routines. If you've set up a REST wrapper for a database query, you can get the JSON data into your app fairly easily. That wrapper might be implemented on Node.js or on another server, as in the case of a Web service extension to the database server.

A more serious MBaaS would already have tested, integrated modules set up to easily map the major databases to a form consumable by its apps, certainly for Oracle, SQL Server, MySQL, and PostgreSQL.

Appcelerator says it has a few enterprise connectors it sells on the MBaaS layer, such as for SAP and Salesforce.com. And one of the advantages of Node is the supply of community-developed modules for many other sources such as MySQL, SQL Server (which works on a Windows server with Node.js), PostgreSQL, and many NoSQL databases.

Similarly, Appcelerator can use a local SQLite database on a device, work with pair storage, cache in-memory, and detect when the device is online. However, it has no complete framework in place for handling intermittently connected apps, especially not conflict resolution. According to the company, most of its customers use Alloy models to handle some of this.

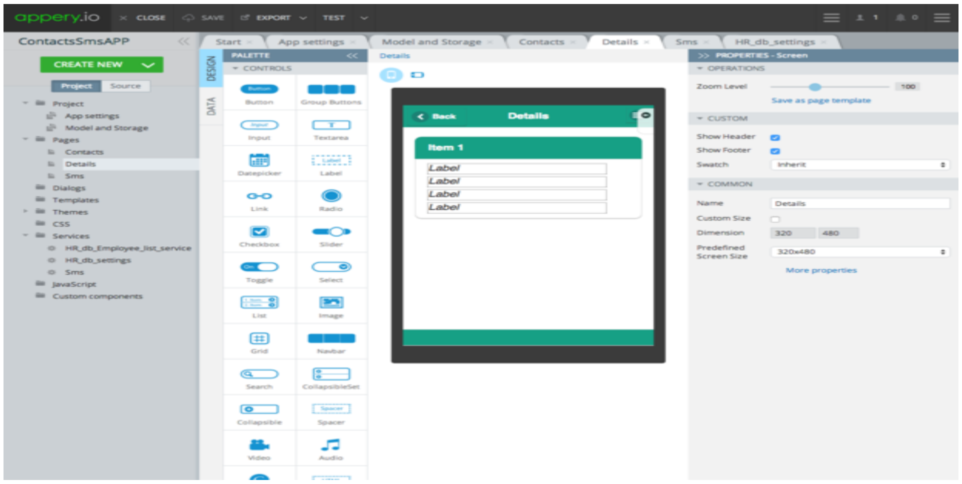
Appery.io

Appery.io is a rather capable cloud-based mobile Web and hybrid mobile development platform with online visual design and programming tools and integrated back-end services. You can think of it as a cross between an app builder and an MBaaS.

The Appery.io app builder generates HTML5, jQuery Mobile, AngularJS, Bootstrap, and Apache Cordova code, and the Appery.io build server generates iOS, Android, Windows Phone, and HTML5 apps. The Appery.io MBaaS provides hosting, a MongoDB database, push notifications, JavaScript server code, and a secure proxy. It allows

HTML hosting to its own cloud, to Heroku, and (manually) to third-party hosting providers.

The Appery.io app builder has tabs for the app settings, your model and storage, your pages as you create them, dialogs, templates, themes, CSS, whatever services you define, your JavaScript, and any custom components you define. The builder uses a WYSIWYG design metaphor with a palette of more than 25 controls, including those for external services such as Google Maps and Vimeo, and displays a property sheet for each item. You can switch from design view to source code view to see your generated HTML, CSS, JavaScript, and any device-specific code: Java for Android, Objective-C for iOS, and XAML backed by C# for Windows Phone.



Zoom Level 100

Show Header ☑
Show Footer ☑
Swatch Inherit

Name Details
Custom Size □
Dimension 320 480
Predefined Screen Size 320x480

Appery.io features a drag-and-drop page designer. The platform automatically generates source code that you can view online. When you are happy with the app, you can export it to mobile Web and mobile hybrid targets, including app binaries.

Appery.io can talk to essentially any REST APIs, whether or not the company has prebuilt the interface. Tying a prebuilt REST interface to a service is a matter of a few minutes; building the REST interface from scratch takes a little longer and requires knowing a little more, but it's not a big job.

You can test your HTML5 app as you go, both in your desktop browser and in your phone and tablet browsers; everything that doesn't depend on Cordova will work. To test your Cordova code (for example, to use native device capabilities or get push messages), you build your app, download it to your device, and run it there. For convenience, Appery.io will display QR codes for your HTML5 app and your binaries so that you can download them directly to your device. For even more convenience, you can install the Appery.io Native Test App shell on your device and point that at your code.

In general, the Appery.io app builder easy to learn and use. Appery.io has done a good job of designing its IDE so that mobile developers will not usually be surprised by what they get.

It's nice that Appery.io has its own cloud-based builder and build service. Coupled with the browser-based IDE, this means that mobile developers don't need to have multiple computers or multiple VMs to create native apps, and they don’t have to maintain multiple native SDKs and IDEs.

EachScape

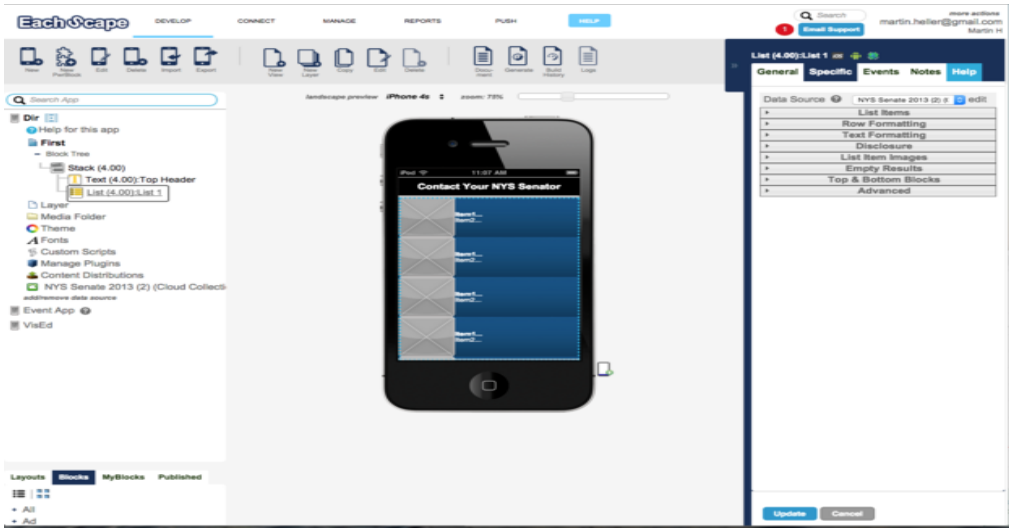
EachScape accomplishes the hat trick of generating iOS, Android, and Web apps from an online drag-and-drop designer. In addition, EachScape provides mobile back-end services for apps you build with its platform, Web preview for all apps, and an online build service.

The architecture that allows EachScape to build iOS, Android, and HTML5 apps from a drag-and-drop editor (the Cloud Studio) depends on blocks and modules, as well as layouts and actions. Under the hood, EachScape has implemented a set of classes in Objective-C for iOS, in Java for Android, and in CoffeeScript for Web apps that correspond to ads, buttons, containers, controls, data connectors, data input, HTML, images, maps, media, navigation, placeholders, RESTful remote queries, social networks, and text. Advanced developers can build new blocks and modules for EachScape to extend its capabilities, using its SDKs.

In EachScape Web Studio, you can drag and drop blocks and configure them to create an app.

The EachScape back-end services include Cloud Collections (explained below), data connectors, analytics, mobile ads, social media access, push notifications, location services, and billing. EachScape does not currently offer back-end services outside of platform subscriptions.

The EachScape Cloud Studio has a Microsoft Visual Basic/Borland Delphi kind of development paradigm. Drag a block onto a page of the app, position it visually, and configure its properties. View a Web preview, play with it, and iterate. Use a cloud data collection or another data source to populate the app with data.



When you're ready to try the app on a device or in a simulator, build the app online and check the targets you'd like from various ranges of Android versions, iOS 7 and 8, and HTML5. Once any target has been built on the EachScape cloud (which can take a few minutes, especially the first time you build an app for a given target) you can download the app for testing in a device or simulator. A QR code on the build history screen makes the download to a device painless.

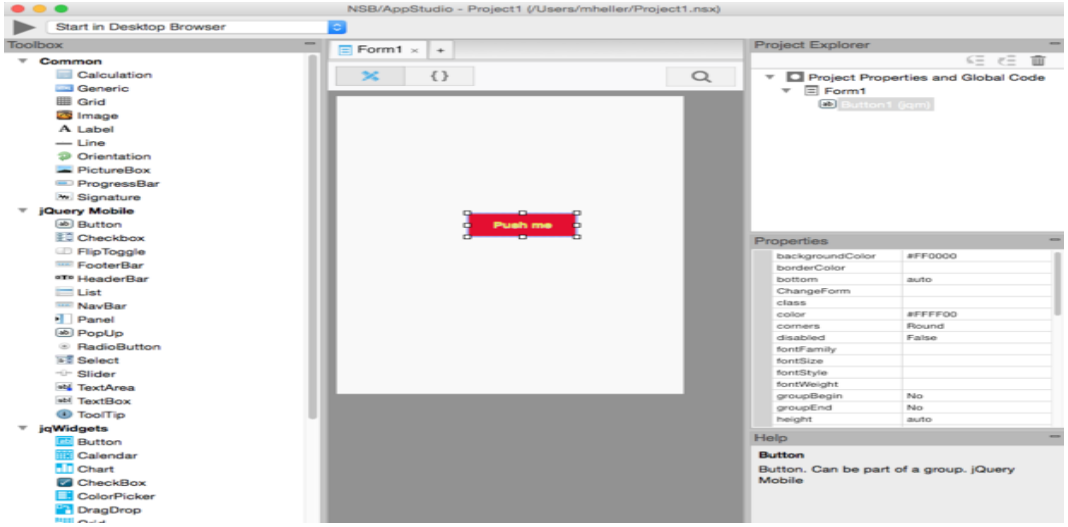
The Cloud Collections feature of EachScape is a little like the MongoDB implementation in most MBaaS platforms and a little like the CMS in WordPress. The EachScape data connector is essentially limited to RESTful XML, RSS, and JSON data sources. EachScape currently offers no tools of its own to create RESTful wrappers around systems of record. According to the company, those will be coming in Q2 2015.

NSB/AppStudio

NSB/AppStudio targets both mobile Web and mobile hybrid apps. The AppStudio IDE was written in JavaScript, HTML5, and WebKit, and it runs on Windows and Mac OS X. The combination of ease of learning, ease of use, royalty-free distribution, and low prices helps AppStudio bring mobile Web and hybrid development to the masses, in the spirit of Visual Basic and the early Borland visual programming products.

You can drag and drop your way to runnable mobile applications built from forms and controls, and write code either in NS Basic -- essentially VBScript with a few extensions -- or in JavaScript. At app publication or runtime, whether for local development or server deployment, any Basic script is transcompiled to JavaScript. You can display the JavaScript for any displayed form from the IDE.

In the AppStudio IDE we have a form designer, toolbox, project explorer, property sheet, and help windows, all familiar from Visual Basic and its heirs and imitators. The IDE doesn't do its own debugging, however -- that is handled by the browser or, in the case of PhoneGap apps, through weinre, a remote debugger for Web pages.



backgroundColor	#FF0000
borderColor	
bottom	auto
ChangeForm	
class	
color	#FFFF00
corners	Round
disabled	False
fontFamily	
fontSize	
fontStyle	
fontWeight	
groupBegin	No
groupEnd	No
height	auto

Button

Button. Can be part of a group. jQuery Mobile

NSB/AppStudio is a drag-and-drop IDE for mobile Web and mobile hybrid app development, very much in the spirit of Microsoft Visual Basic. Notice the familiar form designer, toolbox, project explorer, property sheet, and help windows.

Right-clicking on a control brings up a context-sensitive action menu that lets you create event handlers, add components, and adjust the layout. The selection of components is satisfying and includes almost 60 controls, ranging from simple labels to complex widgets and interfaces to financial services and social media.

AppStudio allows you to build both mobile Web apps and mobile hybrid apps; the latter is facilitated by integration with PhoneGap. AppStudio comes with more than 100 samples, ranging from “hello, world” to demonstrations of using all the included controls, about 30 Web services, and a dozen third-party JavaScript libraries.

The performance of AppStudioNSBasic mobile Web and hybrid apps is surprisingly good and basically identical to the performance of JavaScript from other mobile Web app builders.

If you are only beginning to play with mobile development and don’t have a programming background, NSB/AppStudio is a good place to start. Be aware, however, that it has limited functionality compared to full-featured mobile IDEs and MBaaS platforms, and specifically lacks native mobile app support and enterprise integrations.

Salesforce1

Over the last few years, Salesforce.com, the prominent SaaS platform for sales force automation and other business applications, has been building out its mobile strategy at multiple levels of developer difficulty, ranging from easy with minimal control to hard with complete control. In the simplest option, the Salesforce1 toolkit includes a Web-based drag-and-drop designer suitable for a business analyst. It allows the analyst to customize the app, control security and access, and streamline the process of working with records from a mobile device.

While using the Salesforce1 designer seems simple, it provides a lot of value. When you use it, you provision a custom schema in a cloud database as a service, with strong security, role-based permissions, and automatically exposed REST API endpoints. You get a mobile app that can access anything in it. Should you need to connect to Salesforce via

XML Web services, Salesforce can generate the appropriate WSDL for your custom schema.

At the next level of complexity, a Web developer who knows some HTML5, CSS3, and JavaScript can build pages for Salesforce1 in Visual Force with Mobile Packs or using Lightning components. At the highest level of complexity, a mobile developer can build native or hybrid applications against Salesforce data for iOS and Android with the Salesforce Mobile SDKs. Meanwhile, any of these can utilize the mobile back-end services provided by the Salesforce1 Platform.

The Salesforce1 setup screen is a new part of the Force.com home screen for developers. As you can see, it exposes a quick start wizard, ways to customize your app, ways to control security and access, and ways to enable working with records from mobile.

Salesforce provides free Salesforce1 native mobile shell apps for iOS and Android, which users can download from the appropriate store and use for viewing Salesforce1 Web content. These hybrid apps provide advantages over Web apps, such as retaining state after a context switch and supporting secure offline read access once the data has been loaded. A Salesforce1 Windows Phone app has been announced, albeit without a scheduled release date.

Just as Salesforce has mobile development methods for all levels of developer, it also has back-end development methods for all levels of developer. For novices and business analysts, the obvious choice is point-

and-click declarative Force.com logic. For developers familiar with Java, Apex code will be relatively easy to learn. For those who know the SQL Select statement, SOQL (Salesforce Object Query Language) will prove to be an easy way to query Salesforce objects. For full-text searches, there’s SOSL (Salesforce Object Search Language), which allows you to search Salesforce text, email, and phone fields for multiple objects simultaneously.

Finally, for even more flexibility, you can connect Salesforce to Heroku and do server programming in open source languages.

MAILING LIST – To Whom We Send

- **Mr.B.Murali, HOD of CS, PSG college of Arts and Science, Coimbatore- 14.**
- **Mr.P.Narendran, HOD of CS, Gobi Arts &Science College, Gobichettipalayam-53.**
- **Dr.PannirSelvam, HOD of CS, Erode Arts College (Autonomous), Erode - 09.**
- **Mr.S.SureshBabu, HOD of CS, Thiruvalluvar Government Arts College, Rasipuram.**
- **Dr.K.Thangavel, HOD of CS, Periyar University, Salem-11.**
- **Dr.P.Venkatesan, Principal, Vysya College of Arts and Science, Salem-03,**
- **Dr.P.Swaminathan, Dean, School of Computing, SASTRA University, Kumbakonam.**
- **Dr.S.K.Jayanthi, HOD of CS, Vellalar College for Women, Erode-9**
- **Dr.S.Krishnamoorthy, Dean, Anna University, Trichy-24.**
- **Dr. K. Rama, Deputy Adviser, NAAC, Bangalore.**
- **Dr.HannahInbarani, Asst Prof, Dept of CS, Periyar University, Salem-11.**

- Dr.R.Balasubramaniam, Prof& HOD of CS, ManonmaniamSundaranar University, Tirunelveli.
- Dr.P.Jaganathan, Director, Dept of MCA, PSNA Engineering College, Dindugal-22.
- Dr.D.Venkatesan, SeniorAsst. Prof, Dept. of CS, School of Computing, SASTRA University, Tanjore-01.
- Dr. D.I. George Amalarethinam, Director, Department of MCA, Jamal Mohamed College, Tiruchirapalli – 20.
- Mr. B. Rajesh Kanna, Assistant Professor in Elect &Comm, Annamalai University, Chidambaram.
- Dr.H.FaheemAhmed, Asst Prof & HOD of CS, Islamiah College, Vaniyambadi – 02
- Dr. S. Leela, Controller of Examination, Periyar University, Salem-11.
- Dr. M.Manivannan, The Registrar, Periyar University, Salem-11.
- Prof.Dr.C.Swaminathan, Vice Chancellor, Periyar University, Salem-11.
- Dr.T.Santhanam, Reader& HOD of CA, Dwaraka Doss Goverdhan Doss Vaishnav College, Chennai –06.
- Dr.PremavathyVijayan, Vice Chancellor, Avinashilingam University, Coimbatore.
- Dr.R.S.Rajesh, Reader, Computer Science and Engineering, ManonmaniamSundaranar University, Tirunelveli-12.
- Dr.L.Arockiam, Associate Professor, Dept of CS, St. Joseph College, Tiruchirapalli-620002

- **Mr.V.Saravanan, Associate Professor, Dept of CA, Hindustan College of Arts and Science, Coimbatore – 28.**
- **Dr.R.Ravichandran, Secretary, Dept of CS, KGISL Institute of Technology, Coimbatore-35.**
- **Dr. N.Sairam, Associate Dean, School of Computing, Sastra University, Tanjore – 01**
- **Dr.T.Senthikumar, Asst Prof, Amrita Institute of Technology, Coimbatore - 12**
- **Mr.S.T.Rajan, Sr. Lectr, Dept of CS, St. Josephs College, Trichy-02.**
- **Dr.R.AmalRaj, Prof. Dept Of CS, SriVasavi College, Erode – 16.**
- **Dr.R.Pugazendi, Assistant Professor, Dept. of CS, Government Arts and Science College, Salem-7.**