# Minneapolis Marauders IT Health - Macbrained Hackathon App

A collaboration between Jeff Wilson, Nick Olmsted, and James Trousdale



GitHub Address:

<https://github.com/jeffdageek/MBHackathon>

**Problem Statement:**

Currently there is no way for an employee to proactively monitor their state of IT security compliance or as we are calling it their “IT Health”. A user would need disparate tools to see their own state of backup and would need to contact IT to allow them to see if they are properly encrypted or have all necessary software updates installed. Due to this, a majority of users do not know whether their devices are secured. If they are not, they have no clear path to remediate the issue without generating a ticket for IT assistance. In addition to not knowing their state of compliance, in the event of a theft or loss of device the end users have a non-uniform way to report this to IT where minutes can make all of the difference in confirming that the content is wiped from the device.

**Solution:**

IT Health empowers your users with knowledge. IT Health is app that centralizes big pieces of security compliance in an easy to read and fun way. Allow the user to solve their own problem of not being backed up or not having their computer encrypted by following a guide on their iOS device. Also provide a mechanism to report a stolen or lost device as quickly as possible to allow IT to act quickly.

**Application:**

Code is documented to give idea of main workflow of the application and areas that need improvement or correction in future versions. Currently, application lacks the web calls and parsing of the result due to time limitations. However, classes exist and data is statically programmed to give an idea of workflow.

**Deployment:**

App requires knowledge of user’s JSS Server and CrashPlan server. Future versions would allow for this to be a MDM configurable setting due to it’s likely being deployed from an MDM. However, if this setting is not set, future versions will need to prompt the end user for this information if the information is missing or incorrect.

**Extensions:**

App attempts to conform to the Model-View-Controller-Store methodology. The Store is a class stub that’s currently in the app, the data is provided (and in future versions: cached) from the webservices.

**API calls:**

Below we will describe the API calls and responses that would be used within the IT Health application.

* User Name
* Devices and Backup percentage
* Encryption percentage
* Software up-to-date percentage

User Name

API Call:

<https://www.crashplan.com/api/User>

Response:

{"metadata":{"timestamp":"2014-07-11T09:02:00.145-05:00","params":{}},"data":{"totalCount":1,"users":[{"userId":1476110,"userUid":"ce6597f722790048","status":"Active","username":"nick.olmsted+mauraders@code42.com","email":"nick.olmsted+mauraders@code42.com","**firstName**":"MN","**lastName**":"Mauraders","quotaInBytes":-1,"orgId":110129,"orgUid":"642507809774108673","orgName":"MacBrained Hackathon","active":true,"blocked":false,"emailPromo":true,"invited":false,"orgType":"ENTERPRISE","usernameIsAnEmail":true,"creationDate":"2014-06-25T11:09:14.775-05:00","modificationDate":"2014-06-25T11:09:38.625-05:00","passwordReset":false}]}}

Parse response for:

* firstName
* lastName

User’s Devices and Backup Percentage

API Call:

<https://www.crashplan.com/api/Computer?active=true&incBackupUsage=true>

Response:

{"metadata":{"timestamp":"2014-07-11T09:19:32.718-05:00","params":{"incBackupUsage":"true","active":"true"}},"data":{"computers":[{"computerId":3368546,"**name**":"ubuntu","guid":"642515379838517505","type":"COMPUTER","status":"Active","active":true,"blocked":false,"alertState":2,"alertStates":["CriticalConnectionAlert"],"userId":1476110,"orgId":110129,"parentComputerId":null,"lastConnected":"2014-06-25T12:42:53.099-05:00","osName":"Linux","osVersion":"3.11.0-15-generic","osArch":"amd64","address":"172.16.73.132:4242","remoteAddress":"162.222.47.218","javaVersion":"1.7.0\_45","modelInfo":null,"timeZone":"America/Los\_Angeles","version":1388556100363,"productVersion":"3.6.3","creationDate":"2014-06-25T12:18:34.165-05:00","modificationDate":"2014-07-03T03:52:22.715-05:00","loginDate":"2014-06-25T12:18:38.332-05:00","backupUsage":[{"targetComputerId":14,"targetComputerParentId":null,"targetComputerGuid":"43","targetComputerName":"PROe Cloud, US","targetComputerOsName":null,"targetComputerType":"SERVER","selectedFiles":3,"selectedBytes":73,"todoFiles":0,"todoBytes":0,"archiveBytes":4522561,"billableBytes":4522561,"sendRateAverage":34319323,"completionRateAverage":0,"lastBackup":"2014-06-25T12:19:09.468-05:00","lastCompletedBackup":"2014-06-25T12:19:09.468-05:00","lastConnected":"2014-06-25T12:20:46.828-05:00","lastMaintenanceDate":"2014-06-25T12:18:34.216-05:00","lastCompactDate":null,"modificationDate":"2014-07-03T03:52:23.059-05:00","creationDate":"2014-06-25T12:18:34.216-05:00","using":true,"alertState":16,"alertStates":["CriticalBackupAlert"],"**percentComplete**":100.0,"storePointId":2900,"storePointName":"agb-msp-2","serverId":160021697,"serverName":"agb-msp","serverHostName":"https://agb-msp.crashplan.com:4285","isProvider":false}]}]}}

Parse response for:

* Loop through response by “computers”
* Return “name” for device name
* Return “percentComplete” for backup percentage

User Devices:

API call:

[https://your/jss/address/JSSResource/get/users/name/{name}](https://your/jss/address/JSSResource/get/users/name/%7Bname%7D)

Response:

The XML reply is a full list of the user’s information and the ids of any device associated to that user.

Parse response for:

* Loop through response by <computers> and look for <computer>
* Return <id> and associate <name> to devices in top of user’s console e.g. MacBook Air.
* Return “percentComplete” for backup percentage

Encryption (FileVault):

API Call:

[https://your/jss/address/JSSResource/computers/id/{id}](https://your/jss/address/JSSResource/computers/id/%7Bid%7D)

Response:

The XML reply is a full computer id report that includes all of the inventory data and extension attributes for a particular device.

Parse response for:

* Loop through response by <device> and look for <partition><type><boot>
* Return <filevault2\_status> Display Encrypted 100% for <encrypted> for all other states return state to match i.e. Not Encrypted for <Not Encrypted>
* Return numeric result as percentage only for <filevault2\_percent> if status is <Encrypting>

Software Up to date:

API Call:

[https://your/jss/address/JSSResource/computers/id/{id}](https://your/jss/address/JSSResource/computers/id/%7Bid%7D)

Response:

The XML reply is a full computer id report that includes all of the inventory data and extension attributes for a particular device.

Parse response for:

* Loop through response by <device> and look for <software><available\_software\_updates/>
* Return numerical value of software updates available e.g. 2 if there are two updates that need to be installed report 2 available

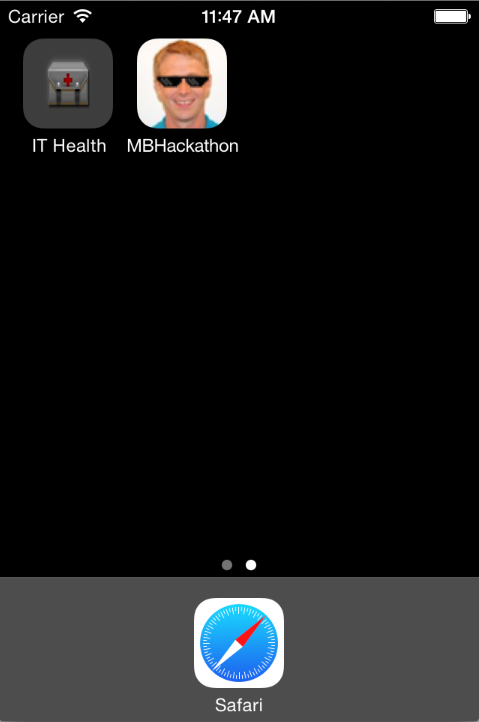
**Use Cases:**

Below are two common use cases we see our users taking advantage of the IT Health app.

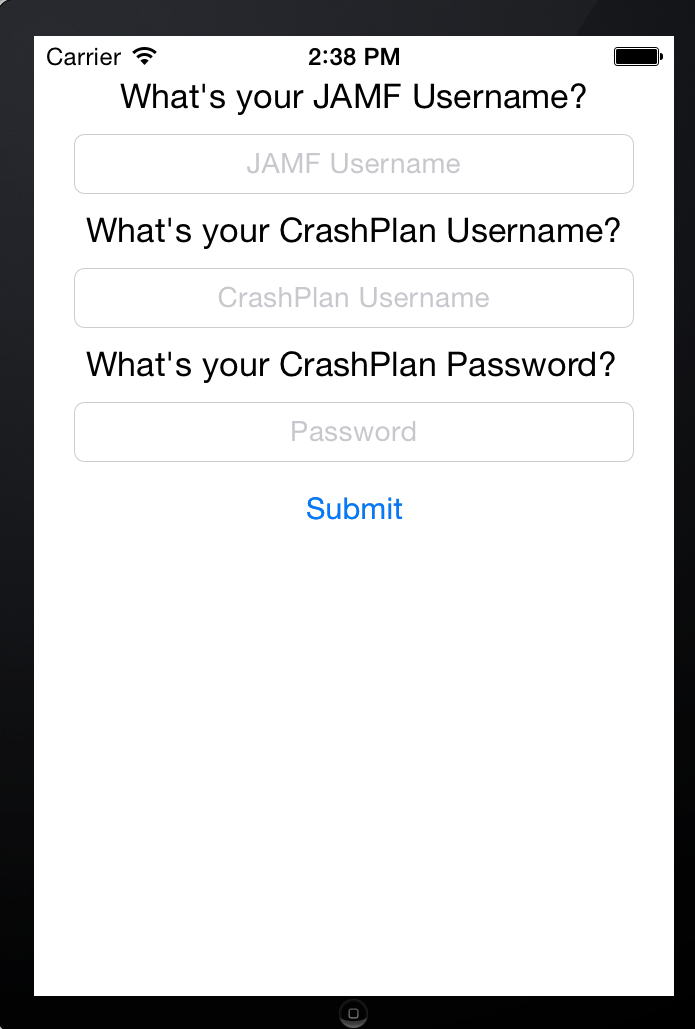
1. User wants to check their “IT Health”.
2. User had their laptop lost/stolen and wants to report the device to IT.

**Use Case 1: User wants to know where they stand with IT compliance**

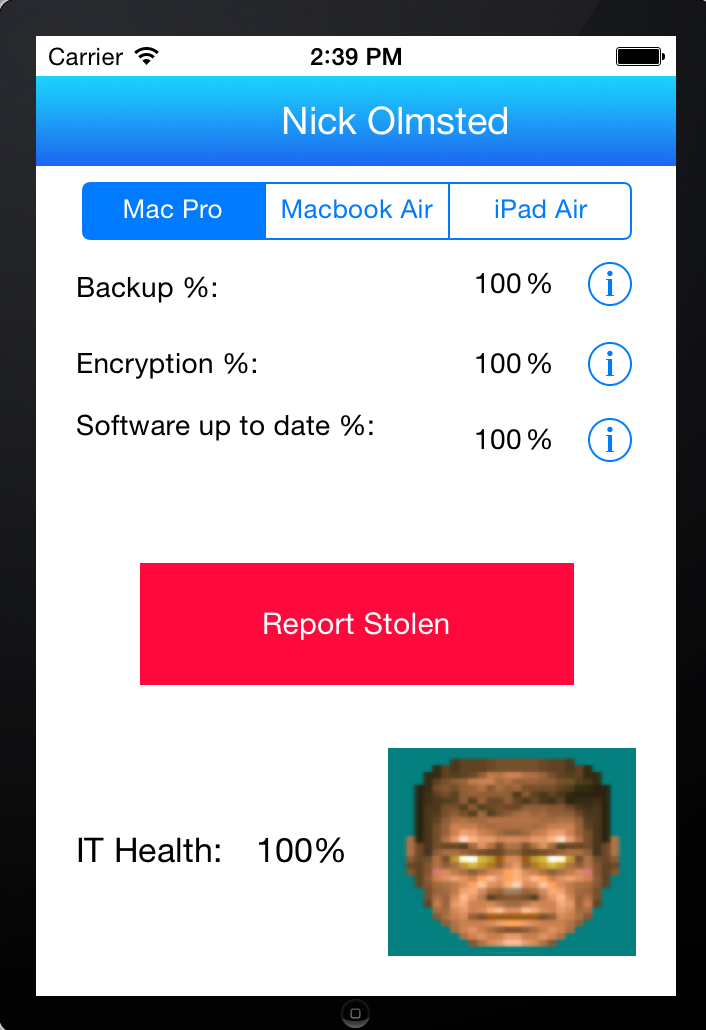
1) User opens the IT Health app on their iPhone device.



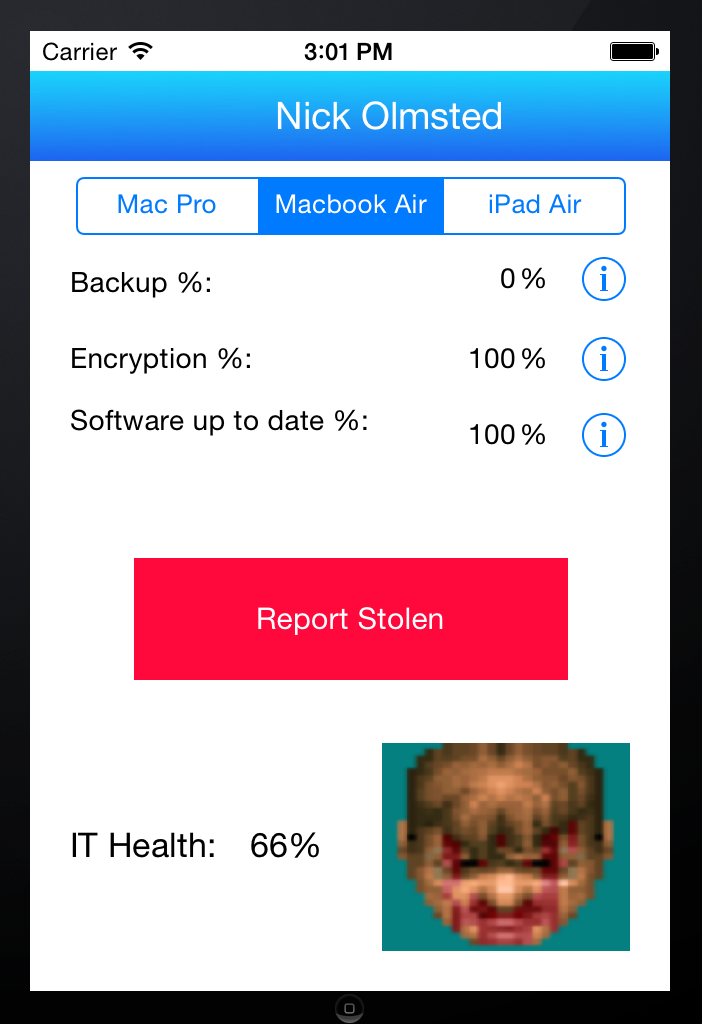
2) Logs in to the application with their JAMF and CrashPlan usernames (if needed).



3) Views their IT Compliance status for the Mac Pro device. The “Doom” picture changes to reflect the user’s IT Health. In this case the MacBook Pro has a IT Health of 100%.



4) Selects their second device (Macbook Air) and views their IT compliance status. Since the user is at 66% IT Health we display the “damaged Doom” picture. This indicates that the user needs to perform an action to increase their IT Health to 100%.

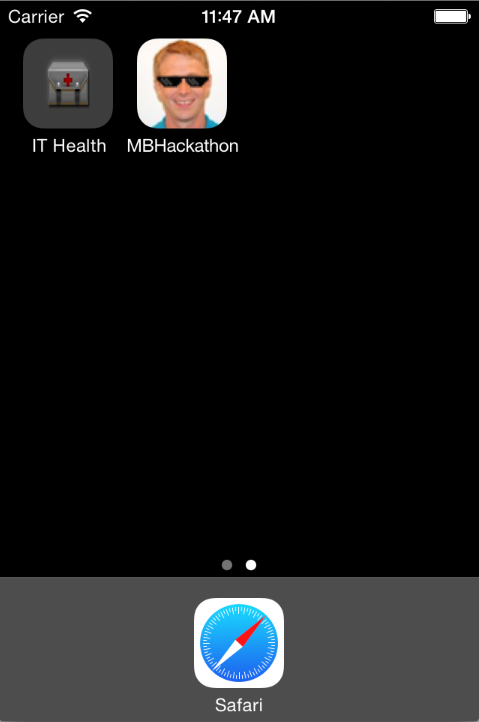


5) Clicks on the information icon for Backup % and opens a CrashPlan or company-specific support article on how to start their backup (feature not implemented yet)

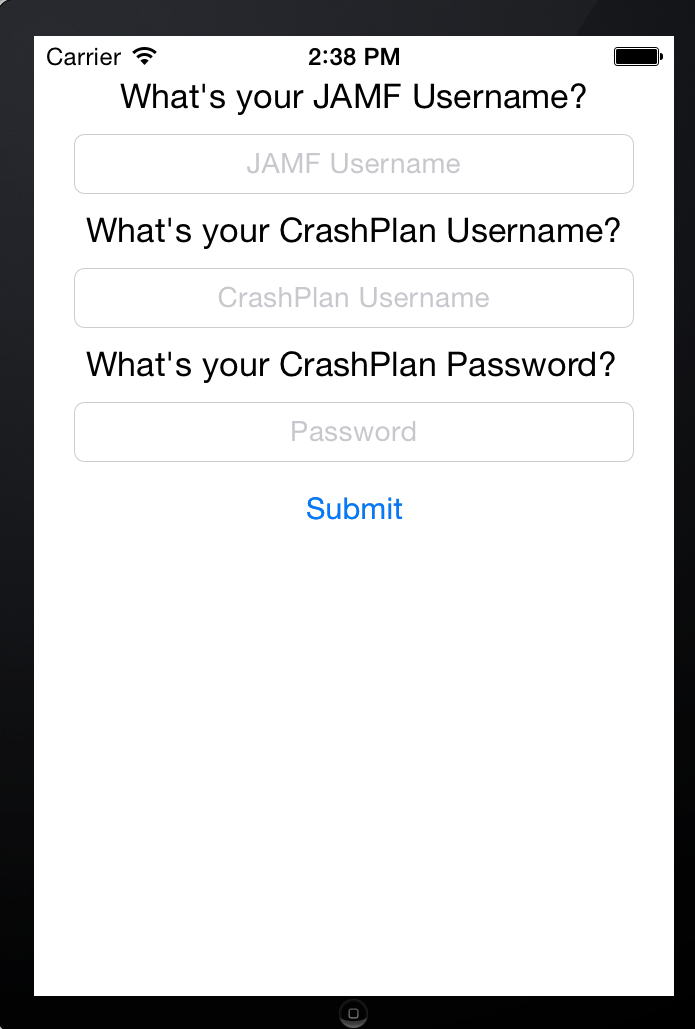
[CrashPlan or company-specific support article opens with default mobile browser]

**Use Case 2: User had their laptop lost/stolen**

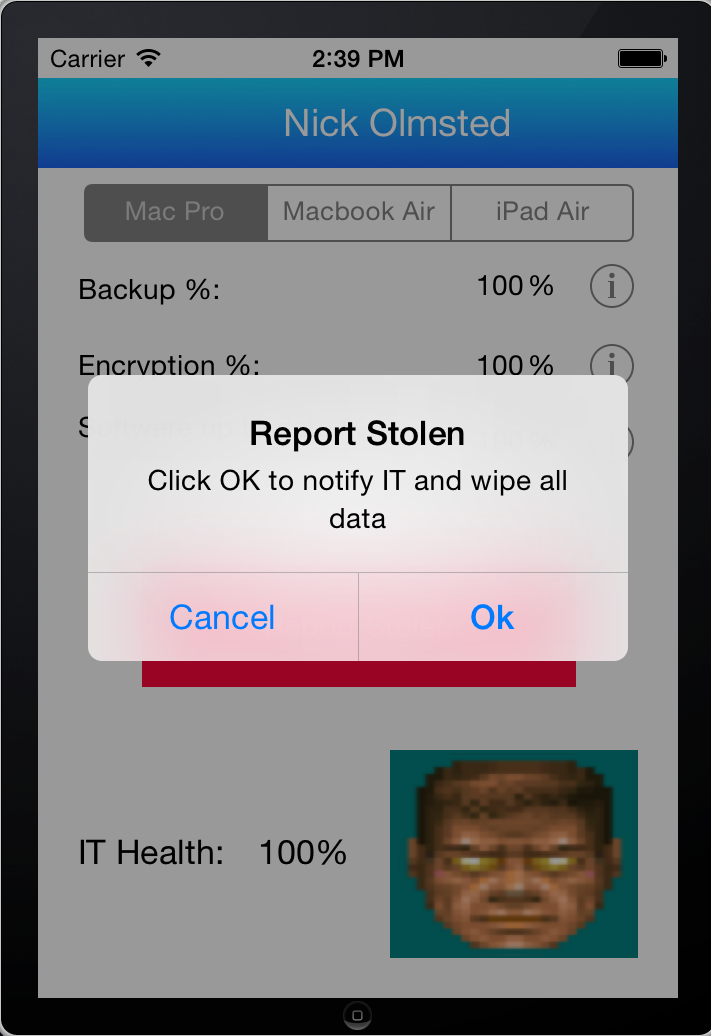
1) User opens the IT Compliance app on their iPhone.



2) Logs in to the application with their JAMF and CrashPlan usernames.



3) Clicks on the Report Stolen button and confirms to send IT an email and remotely wipe their stolen device.



**References:**

Additional info on the CrashPlan API:

<https://www.crashplan.com/apidocviewer/>

<https://support.code42.com/Administrator/3.6_And_4.0/Monitoring_And_Managing/API>