C:\data\freeclinic\ADC_cover.tif

**Android Challenge I Qualifying Round**

April 14th 2008

Henry Koren - William Tai - Cameron Esfahani

Iman Sadeghi - Ryan Hertzog - Brian Visser

Seth Jaspan - Brent Longstaff – Joe Nasca

April 14th 2008

Table of Contents

[Application Overview 3](#_Toc195945468)

[Mission 3](#_Toc195945469)

[Project History 4](#_Toc195945470)

[UCSD Student Run Free Clinic Project 4](#_Toc195945471)

[Teams In Engineering Service 5](#_Toc195945472)

[Android Software Development Group 5](#_Toc195945473)

[Software Foundation 5](#_Toc195945474)

[Future Plans 5](#_Toc195945475)

[Licensing 5](#_Toc195945476)

[Commercialization 5](#_Toc195945477)

[Architectural Components 6](#_Toc195945478)

[Android Application 6](#_Toc195945479)

[Web Application 6](#_Toc195945480)

[Server Component 6](#_Toc195945481)

[Development Status’ 6](#_Toc195945482)

[Clinic Modules 7](#_Toc195945483)

[Log In 7](#_Toc195945484)

[Check In / Identification 7](#_Toc195945485)

[Enrollment 7](#_Toc195945486)

[Patient Information 7](#_Toc195945487)

[Visit Planner 7](#_Toc195945488)

[Visualization 7](#_Toc195945489)

[Clinic Visit 7](#_Toc195945490)

[Annotate 8](#_Toc195945491)

[Diagnose 8](#_Toc195945492)

[Treat 8](#_Toc195945493)

[Prescribe 8](#_Toc195945494)

[Refer 8](#_Toc195945495)

[Labs 8](#_Toc195945496)

[Pharmacy 8](#_Toc195945497)

[Prescription Filling 8](#_Toc195945498)

[Inventory Entry 8](#_Toc195945499)

[Inventory Audit 8](#_Toc195945500)

[Patient Assistance 8](#_Toc195945501)

[Reporting 9](#_Toc195945502)

[Security Features 9](#_Toc195945503)

[SSL Socket Communication 9](#_Toc195945504)

[Device Registration 9](#_Toc195945505)

[Remote Kill 9](#_Toc195945506)

[Device Tracking 9](#_Toc195945507)

[Patient Record Protection 10](#_Toc195945508)

[Technical Frameworks 10](#_Toc195945509)

[Clinic Item 10](#_Toc195945510)

[Clinic Connection 10](#_Toc195945511)

[Clinic Service 10](#_Toc195945512)

[Clinic Provider 10](#_Toc195945513)

[Clinic Activity 10](#_Toc195945514)

[Information Types 10](#_Toc195945515)

[Patients 10](#_Toc195945516)

[Messages 10](#_Toc195945517)

[Measurements 11](#_Toc195945518)

[Demographics 11](#_Toc195945519)

# Application Overview

## Mission

To create a software platform for doctors and medical workers to interact with the medical records of their patients. This software will be given freely to clinics which do not charge their patients for medical care, or licensed to clinics that operate for profit. The aim of the software is to improve the quality of patient care through better information. Areas of interest to the project will be diagnosis and treatment tools, chronic disease management, visualization of patient history, procedural checklists, and reporting on the efficacy of care.

As the medical community moves toward a greater need to modernize, they often face the task of moving their traditionally paper records system into a digital medium. While the technology to store the records exists, several problems have arisen from its implementation, one of the most substantial of which is doctors’ general wish to have ease of data entry. In the past, this has been accomplished by allowing them to make notes on paper files, write hand written prescriptions and record measurements by hand. If this practice continues into the digitalization of records, it will become necessary for someone later to take these records and enter them manually into a database. We believe that by providing doctors an easy to use digital medium by which to enter data, the data entry step can be eliminated and substantial man hours can be saved. Further, because a stage of data entry is skipped, there is less chance of errors in data and medical malpractice.

Our goal in this project is twofold:

1) To provide an efficient means by doctors and medical personnel to enter data

2) to provide tools to doctors that will allow them to increase the efficiency of their work by giving them immediate access to necessary patient and medical data without having to reference secondary sources.

The first point is addressed by the data entry system of the device which allows for easy entry of measurements such as temperature, heart rate, etc. We believe that in time and with appropriate development, we will be able to eventually receive data from instruments such and advanced hardware configurations of the hardware underlying android. Integration with sensor data sources will allow for the automatic entry of measurement data into the devices and further reduce the need manual input. In addition, we wanted to leave doctors the ability make “side notes” as they would with paper records by attaching a notes field to all elements of patient records.

The second point is addressed by the data made available by the service about each patient that the doctor is scheduled to see. We have developed a front end work station based system for registration of patients into the system and from here the data is sent to the mobile device of the doctor who will respond to that patient’s needs. Once the patient is added to the queue of the device, their medical history is also made available to the doctor so that they may quickly run through any necessary concerns before and during their exam. Past diagnosis, treatments, prescriptions and measurements are all visible on the device and past measurements can be graphed using the devices OpenGL software. The unique nature of having an always-on connection to the network means that the need to store information on the device itself is significantly reduced. This means that data can be frequently switched out and that confidential patient data can also be stored only in volatile memory where it is more difficult to recover if the device is stolen or lost.

## Project History

### UCSD Student Run Free Clinic Project

Started in 1997 this project led by Medical Professor Dr. Ellen Beck currently hosts free clinics at three different San Diego locations. In the effort to upgrade their information technology systems with medical record and computer aided disease management systems, the administrators of the program sought a partnership with the Jacob’s school of Engineering.

### Teams In Engineering Service

Started in September 2006 with the guidance of Associate Dean of Jacob’s School of Engineering Jeanne Ferrante the initial software design and development efforts was undertaken by an interdisciplinary team of Undergraduate students.

### Android Software Development Group

Started in November of 2007 by Computer Science Department student Henry Koren, this group of software Developers united to take the Google Challenge and produce an entry for their 10 million dollar competition. The group established their offices at UCSD’s California Institute for Information Technology and Telecommunications.

## Software Foundation

Ultimately we see the project coming to a point where it will be best served by detaching itself from UCSD and establishing itself as a 501C3 Non-profit organization. At this point all support from the school would stop. We anticipate that the project could reach this advanced stage within 3-5 years of active development.

## Future Plans

It is a goal of the project to formalize our position as a research Project within UCSD.

The project wishes to position itself to receive corporate sponsorships and donations. In order to make these donations possible it is essential that the donors be given some sort of guarantee that some portion of their money will be dedicated towards the furtherance of the project.

## Licensing

The project will be licensed under the GNU General Public License GPLV3.

For companies that wish to make closed-source commercial derivatives based on the project, licensing will be negotiated on a case by case basis.

## Commercialization

The potential of this project is not limited to academic non-profit pursuits. Ultimately this software will be found useful in commercial clinics as well as free ones. In order to get the maximum amount of impact, the project should be structured to encourage entrepreneurial efforts to provide commercial components and implementations.

There are some areas of a traditional for-profit medical practice which are not necessary components to our project. In an effort to encourage the commercialization of these components, the project will "cordon off" the development of these components so that no efforts are made to incorporate the features into the core functionality of the system. The following regions will be restricted from development:

**Biometric Identification** - Such as retina, fingerprint, bar code, or RFID systems

**Insurance Remittance** - Automation of the billing of insurance companies and/or Medicare.

**Remote Sensor Integration** - Retrieval of data through wired or wireless sampling. This includes Scales, Thermometers, blood glucose meters, pulse oximeters, blood pressure monitors and other sensors which can be used to sample patient data.

**Appointment System** - Since the free clinics are primarily first-come-first-serve, development of an appointment system will be licensed to an external commercial software provider.

**Clinical Pharmacology** - The compilation of drug information feeds for additional drug information will be provided by third-parties such as Epocrates.

A commercial license to provide preferred plug-in modules to the system will be granted to companies that make agreements to support the project.

# Architectural Components

## Android Application

The most useful features of the web application will be ported to work on Google's Android mobile handset platform. These devices will become available in Q2-2008.

## Web Application

This is the primary interface to the system. Certain aspects of the application, including patient check-in, enrollment, and administration will be accomplished with the existing PHP/ MySQL web application.

## Server Component

Both applications will interact with a Relational Database management system and business logic back-end that is hosted on a centralized server. The applications will connect to this server through an 128 bit SSL encrypted link.

The server will relay information to the client devices. The server will keep a database of messages updated on the client devices, however patient data will only be loaded onto the devices when a user is granted access to that record, and will be erased from the handset as soon as the user is done dealing with that patient.

## Development Status’

The web application whose initial planning began in September of 2006 is currently in pre-production testing Alpha development status. A partial release focusing on Pharmacy capabilities is planned for June of 2008.

The android application development began in November of 2007, it is currently being rapidly developed, and considered in pre-Alpha release status. Significant additions and testing will be required. A working Beta product should be to be developed before the end of 2008 as the android hardware market heats up.

# Clinic Modules

### Log In

The user enters a username and password which are verified by the server

## Check In / Identification

The user selects a patient checked in to the clinic or searches for a new patient to check in.

### Enrollment

A form containing patient information and demographics is filled out and a patient record is established

## Patient Information

This will display a visual representation of the patient’s record. Implement a graphing library using the OpenGL ES API that is included with Android to visualize a historical timeline that will superimpose vital signs, quality of well being, and other metrics with designators of events such as diagnoses and surgeries. This will be given an interface that allows you to zoom or horizontally scroll the historic data and future treatment plan with a whisk of your finger, or get details on a particular event by taping it’s icons.

### Visit Planner

The Visit Planner is a customizable, printable report that is part of the Registry module. It contains information about individual patients that are relevant for their assessment during a clinic visit by attending physicians and medical students. The information contained in it includes general vital information that are standard to all clinic patients, as well as disease-specific information that the patient may have (e.g. asthma, diabetes, hypertension, etc.). The form will be customizable to show or hide these specific when it is displayed or printed.

### Visualization

We are implementing a graphing library using OpenGL ES API that is included with Android to visualize a time line that will superimpose measurements and messages. Measurements will be normalized so that their range of values maximizes vertical real estate. You will be able to horizontally scroll or zoom the patient data. The user will be view messages or see measurements values by taping their screen elements.

## Clinic Visit

Here The user will be able to see current information for the patient. First and foremost user will be informed of the highest priority items that are to be addressed in their visit with the patient. Then, vital signs are taken, important messages are displayed, diagnoses are made, and scheduled treatments are administered. This will adapt to the patient based on the treatments for their diagnoses and other doctor recommendations. This will also allow Lab tests, Referrals, and Prescriptions to be made. From this screen the user can choose to view/alter, or add new kinds of information to the patient record, or visualize the patient's history.

### Annotate

Add a generic note to the patient record.

### Diagnose

Add a ICD9 diagnostic code to indicate a condition which the Doctor has assessed

### Treat

Record the occurrence of a specified treatment to the patient

### Prescribe

Create a drug recommendation to be forwarded to the pharmacist.

### Refer

Select a doctor to send the patient to.

### Labs

Internal and external labs start with a request and get returned with one or more measurement.

## Pharmacy

A system for maintaining drug inventories and prescribing patients pharmaceuticals. This will be integrated with patient assistance programs. The system currently includes dynamically updated feeds of drug information from the FDA and the office of national drug control (NDC).

### Prescription Filling

Create a prescription as a pharmacist, checking for drug interactions and allergies and managing the dosage and substitutions with other drugs.

### Inventory Entry

Addition of a bottle of drugs to the Inventory, assigning a custom bar code containing the Lot number and expiration date of the drug.

### Inventory Audit

Allow a query to the pharmacy inventory, showing the amount of drug stocked, the transactions within a certain date period, or the patient recipients of a particular lot number which may be recalled.

### Patient Assistance

Submit a request to a Drug Manufacturer for a supply of free drugs for an underserved patient. This will fill out a patient assistance form by superimposing patient information over the form and outputting a PDF which can be printed and mailed to the drug manufacturer.

Once received, the patient’s drugs are tagged and linked directly to them so no other patients are allowed to draw drugs from a certain patient’s designated supply.

## Reporting

Generating custom reports showing statistics about what kind of problems people had, and tracking the hopeful increase in the quality of well being (QWB) of the clinic patients.

# Security Features

## SSL Socket Communication

Communication between both the web browser and the Android handset are secured by a 128 bit SSL socket connection.

## Device Registration

* User must register device for access to data base.
* Registration is done via Short Message Service (SMS).
* Server registers device using the IMEI number and the phone number of the device.
* The registration process grants certain permissions of the device.
  + ACCESS\_ASSISTED\_GPS
  + ACCESS\_CELL\_ID
  + ACCESS\_GPS
  + ACCESS\_LOCATION
* Makes use of GSM libraries.
* The login activity gets the IMEI number and checks for valid registrations on the server.
* The server generates a SMS response that contains a random key which the client can then send back to the server to verify the authenticity of the phone number of the device connecting to the system.

## Remote Kill

In the event that a device is stolen, the owner can report it on-line and the phone will be flagged stolen. If the user logs on with a stolen device, the server will send it a kill signal which will accomplish the following:

* + - A remote service, will be activated
    - This will run in the background and not be seen by the user.
    - The local data on the handheld will be deleted.
    - Any local data maybe replaced with erroneous information to confound the data theft

## Device Tracking

* The device will send coordinates of this location to the server by way of the GPS API.
* Longitude, Latitude, and Altitude.
* The coordinates can be sent to an individual who can track the phone down.
* The person tracking the phone can make requests for updates for current location.

**Android Competition Note:**

Due to the lack of unique IMEI or phone numbers we were not able to include these functionalities with the demo. We will not be able to demonstrate this until the emulator is able to have unique identifiers or we have access to actual physical devices.

## Patient Record Protection

Strong user and group level permissions on opening a patient, in addition to challenge response security to require verification of certain pieces of patient information has yet to be implemented in the current release but will be added on as the core functionality to the application is further established.

# Technical Frameworks

## Clinic Item

## Clinic Connection

## Clinic Service

The service has the role of providing updated information to the device as it is requested or new patients are checked in.

## Clinic Provider

## Clinic Activity

# Information Types

## Patients

## Messages

Messages are part of clinic visit, registry, and pharmacy modules.

**Types of Messages:** Diagnosis (ICD9 Code), Referrals, Prescriptions, Notes, Lab Requests/Results, Patient Assistance Requests, Treatment Alerts. With each type of message comes a custom set of data which will be defined.

**Manual Creation:** A user can designate that a patient will have an alert added to their account. They will be able to search for past alert texts, or just type in custom text for a new alert

**Triggering:** An alert can be triggered based on a set of conditional logic statements being true.

**Inheritance:** A message can inherit the behavior of another message.

**Scheduling:** Periodic repetition for a specified time span.

**Status Tags:** Messages will be tagged with status that will change based on the progress of a message through the system. Example status codes: Open, Recommendation, Suspect, Due, Overdue, Pending, Closed.

Message data will be used In clinic visit, where lists can be generated that will show critical items at the highest priority. In the registry module, where reports can be generated based on the status codes of the patients that possess particular messages.

## Measurements

Measurements are values which can be of various types. Predominantly these measurements will be vital signs, such as blood pressure, heart rate & respiration. Some measurements may come as a result of lab tests, and certain diagnoses may cause the system to try to collect additional measurements (for examples, for diabetics the system will ask for blood glucose).

## Demographics

Static items such as age, race, social history, family history. Demographics are collected on the enrollment form and will be the basis of generating certain reports. These samples are used to generate 3D visualizations on the Android handset that will allow the Doctor to see trends in the figures over time.