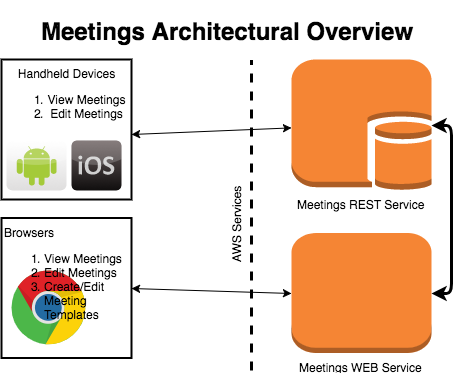
*LDS Meetings Architectural Design*

This document gives a brief overview of the architectural design of the LDS Meetings software. The following diagram illustrates the basic design.



# Overview

The overall architecture is pretty simple. There is a REST service that runs on AWS that talks to a database. The database stored all of the information about templates, meeting instances, and meeting instance data. Handheld devices such as iPhone, Android phone and tablet will talk to the REST service to retrieve information for displaying meetings and editing meeting data.

There will also be a WEB service running in AWS that will talk to the REST service to retrieve information. The regular browser will communicate with the WEB service to display meetings, edit meeting data, and create and edit meeting templates.

## Heading 2

### To easily apply any text formatting you see in this outline with just a tap, on the Home tab of the ribbon, check out Styles.

### For example, this paragraph uses Heading 3 style.

# REST Service

The REST service is the hub of all activity. It will provide end-points for all information related to meetings. The following subsections define the general areas of information the REST service will be providing.

All of the information for meeting templates, etc. will be stored in the database.

**FUTURE: This whole system of meetings may be mutated to work with any company in which case the information may be stored in that company’s cloud service.**

## Templates

### A template is a general outline of a meeting. Any time a meeting follows the same format from meeting to meeting it should be turned into a template that defines the general layout and input parameters. The template will include things like background, images, person conducting, speakers, etc.

### Here are some possible end points for templates.

### Retrieve all templates

### Retrieve, Update, Delete a template

## Meeting Instances

### A meeting instance is a specific meeting. It will consist of the date and time of the meeting, meeting type, and the template being used.

### Here are some possible end points for meetings.

### Retrieve, Update, Delete a meeting

### Retrieve a list of meetings (date range, meeting type, etc.)

## Meeting Data

## Meeting data is the data that fills in the template for meeting instances. Once a meeting is created and a template chosen then the meeting data can be filled in for the template. Here are some examples of meeting data that would need to be filled in:

## Speakers

## Date / Time

## Location

## Conducting

## Callings / releasings

## Hymns

## Possible end points for meeting data are:

### Add / Delete data

# Web service

The web service is responsible for driving the browser. The web site will have much the same functionality of the handheld devices with the advantage of using a full screen and creating templates. It will also include admin functions exclusive to the administrators of the entire system.

The web service will talk to the REST service in much the same way as the hand-held devices do. It will retrieve all data and update data for the meeting.

# Hand-held devices

Applications will be written to work on iOS and Android devices that will communicate with the REST service to retrieve and view information about meetings. It will also allow the user to edit meeting information depending on their access level to the meeting. Access levels are Administrator, limited administrator, viewer. Here are definitions:

* Administrator – this person has full access to all aspects of the meeting, able to create, change, and view a meeting. The admin defines who has access to the meeting (everybody, relief society, priesthood, etc.)
* Limited Admin – This person has access to update certain parts of a meeting such as hymns.
* Viewer – This person can view the meeting and access certain links presented in the meeting such as contact info.

# access

Access to the meeting system will be controlled by login access. For LDS Meetings, access will be controlled through login to the LDS account. This login will provide information about the user such as calling in the ward. The calling in the ward will determine access levels. For example, the bishopric members will automatically be administrators. They will also have the ability to designate others as administrators. Viewers of meetings will not need to log in to view unrestricted meetings. For restricted meetings a login will be required to determine what group the view falls into.

With hand-held devices the device location and time can be used to determine the meeting they are in and automatically bring it up for them if they are not logged in. This is accomplished by setting location information for a meeting which can be the ward, stake, or an address. If a viewer is visiting another ward the system will be able to look up any meetings that match the users location and the time.

# Templates

Templates were mentioned earlier. They are key to the meeting system. This section contains information about templates.

The idea behind templates is pretty simple. Meeting formats do not change that much from meeting to meeting. By creating a template for a meeting the user creates an instance of a meeting and assigns a template to it. Then the user can just fill in the blanks for the meeting.

The system will come with stock templates for meetings such as sacrament, stake conference, etc. The user will be able to create a custom template using the stock template as a starting point. For example, if a ward likes a specific picture of the savior for their sacrament meeting, rather than using the stock template and filling in the picture for each new meeting, they will create a custom template with the picture already filled in.

Templates will be saved in xml format. The specifics of the xml will be defined in another detail design document.