**Risk Mitigation**

**Calendar:**

To implement the calendar into our program we will use the Calendar Provider which is a repository for a user's calendar events. Normally, to read or write calendar data, an application's manifest must include the proper permissions, described in User Permissions these include the READ\_CALENDAR permission in its manifest file as well as the WRITE\_CALENDAR permission to delete, insert or update calendar data. To make performing common operations easier, the Calendar Provider offers a set of intents which we will try to implement. These intents take users to the Calendar application to insert, view, and edit events. The user interacts with the Calendar application and then returns to the original application. To be able to hold the user’s data, through the Calendar Provider API, applications and sync adapters can get read/write access to the database tables that hold it which is something we will think about implementing as well since the user might have data with regards to the tasks they are setting for themselves. To directly insert, modify, and read events from the Calendar Provider, we will need the appropriate permissions. However, if we end up deciding that we are not building a full-fledged calendar application or sync adapter, requesting these permissions isn't necessary. We will instead use intents supported by Android's Calendar application to hand off read and write operations to that application. By designing your application to perform common operations through the Calendar, we will provide users with a consistent, robust user interface. In addition to that to perform an update of a calendar, we will provide the \_ID of the calendar either as an appended ID to the Uri (withAppendedId()) or as the first selection item. The selection should start with "\_id=?", and the first selectionArg should be the \_ID of the calendar. We will also do updates by encoding the ID in the URI. Calendars are designed to be primarily managed by a sync adapter, so we will should only insert new calendars as a sync adapter. The CalendarContract.Instances table holds the start and end time for occurrences of an event. Each row in this table represents a single event occurrence. The instances table is not writable and only provides a way to query event occurrences hence we will not forget to include this as well.