AngularJS + Firebase + d3 Application

SPECIFIC GOAL:
Integrate d3 charting with an AngularJS + Firebase CRUD app to create a simple data entry and analysis application.

LONG-TERM GOAL:
The data analysis is the centerpiece of a project management application that might also include messaging, file uploads, and user login. This application would be employed by companies (primarily nonprofits) who are receiving a government or private funding grant for a specific project and need some accountability for achieving project milestones. The app assists the company in seeing trends in real time and managing the project accordingly.

Ideally, the data collection and presentation will be replicable/modifiable, as each installation will collect different information and want to review it different ways. Each installation will require custom coding of the database and d3, but the overall framework should be easy to modify and adapt.

PHASE 1 DEVELOPMENT:
The goal of Phase 1 is to create a proof-of-concept application that can be run locally and used as a selling tool.

The CRUD component currently contains four fields for each record:

- Time/date of entry (currently non-functional)
- First Name
- Last Name
- Condition (option dropdown: Condition 1, 2, and 3)

The data component has three sections that will need to respond dynamically to new data entry (highlighted in blue)

- TOTALS: at the top of the app are places for running totals of each Condition plus overall total
- LINE CHART: a line chart reflecting the number of records added with each Condition, plus total entries on a day-by-day basis
- DONUT CHART: a donut chart reflecting percent of each Condition as a part of the total

Functional specs/issues include the following:

- If we need to start from scratch with a new Firebase, that's fine
- Need to determine ASAP if d3 will work, and if not, select another charting system
- CRUD app code can be modified, if necessary, for better performance / ease-of-use
- charts do not need to be styled beyond the basics
- data can update on refresh – does not need to be true real-time
- ideally, the line chart will have tooltip/hover functionality where the user can hover over a point in time and see the relevant details (e.g. Date: 9/13/2013 / Condition 1: 25)
- ideally, the line chart will also be responsive, as the proof-of-concept app will be built using the Bootstrap 3 responsive framework

PHASE 2 DEVELOPMENT (FUTURE WORK):

- integration of CRUD / d3 into admin template (included zip file)
- addition of persistent messaging / alert component (to be spec-ed later)

PHASE 3 DEVELOPMENT (ON SUCCESSFUL SALES):

- integration of login
- addition of file upload
- hosting / deployment considerations

QUESTIONS:

- what will happen in d3 as days go to months go to years?
- Timestamp vs Date Picker vs manual entry?
- File upload in AngularFire?
- Export JSON from within app to CSV?
- Master dashboard to pull from multiple installations?