**Development Workflow**

# Introduction

# Directory structure

Automate workflow using tool like

* Yeoman
* Bower
* Grunt

Using module pattern to organization code

test/ - test configuration.

src/ - all the source file for the application.

vendor/ - third-party libraries. Bower will install packages here. Anything added to this directory will need to be manually added to build.config.js and karma/karma-unit.js to be picked up by the build system.

.bowerrc - the Bower configuration file. This tells Bower to install components into the vendor/ directory.

bower.json - this is our project configuration for Bower and it contains the list of Bower dependencies we need.

Gruntfile.js - our build script;

package.json - metadata about the app, used by NPM and our build script. Our NPM dependencies are listed here.

# Prerequisites

Install Git

Install NodeJS: its node package manager and run the preconfigured local web-server and the test tools

Install Yeoman and Karma as global npm module (npm install –g yo karmar)

# Clone taskcenter.app

Clong takcenter.app located at Sanbox by using the following link: http://tiger.in.alcatel-lucent.com/svn/axs/taskcenter.app/trunk

# Install Dependencies

Npm install

We have preconfigured grunt to automatically run bower so we can simply do:

grunt

# Task Automation

Grunt-bower-task: using for grunt to automatically run bower

Grunt-ng-annotate

Grunt-contrib-stylus

# Run development web server

Grunt-server or grunt serve

http-server

now browse to the app at http://localhost:8080/app/index.html

# Structuring AngularJS code

Note: check out ng-annotate if you don’t want to worry about inject the parameter names to avoid script minification issues. It’ll handle doing that after the fact

# Testing

There are two kind of test in the application

Unit test: grunt test

End to End test:

# Debug

Using tool: Batarang debugging tool

# Deployment

grunt