

# Angular Workflow with Gulp.js

Cihad Horuzoğlu



# Cihad Horuzoğlu

**Front-end & Mobile Developer @ CicekSepeti.com**

**@cihadhoruzoglu**

**<http://cihadhoruzoglu.com>**

**Think your Work Flow**

# Work Flow



## Setup

- Download dependencies
- Download framework
- Download libraries



## Develop

- Non minified
- Separated files
- Test configuration
- Compiling
- Test code
- Watchers
- Live reload
- Linting



## Build

- Minified
- Merged files
- Live configuration
- Compiled
- Renamed
- Optimize performance
- Deployment

**Don't do it manually,  
Automate it!**





**Use front-end tools**







**What is the difference  
between Grunt & Gulp ?**

# Gulp vs Grunt

## Gulp

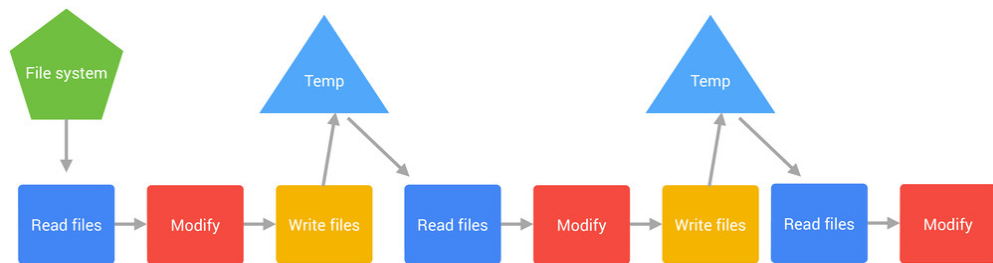
- Code over configuration
- Easy to read & use
- Based on streams

## Grunt

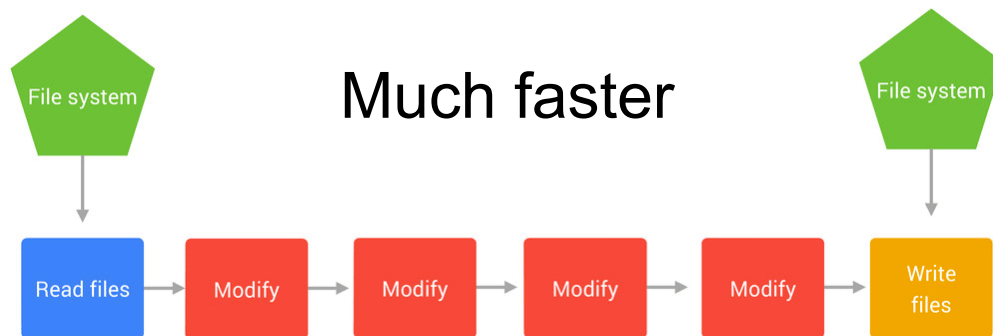
- Configuration over code
- Hard to read
- Based on files

**How these works?**

# Grunt



# Gulp



# The streaming build system



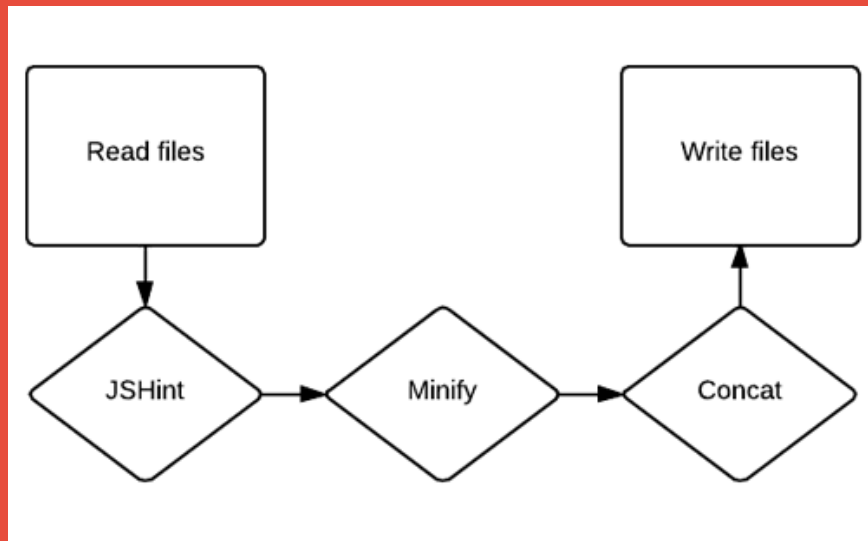






# Sample Stream

```
gulp.src('app/js/main.js')  
  .pipe(jshint())  
  .pipe(uglify())  
  .pipe(concat())  
  .pipe(gulp.dest('build'))
```



# Getting started with Gulp

# Using gulp

Install gulp

Create  
gulpfile.js

Install  
dependent  
packages

Write tasks to  
your gulp file

Run your gulp  
tasks

# Install Gulp

## Globally Install

```
$ npm install -g gulp
```

## Install as a dependency to your project

```
$ npm install gulp --save-dev
```

# Create gulpfile.js

```
var gulp = require(gulp);
```

```
gulp.task('default', function() {  
  // place code for your default task here  
});
```

# Run via Terminal

**\$** gulp *(runs default task)*

**\$** gulp minify concat *(runs multiple task)*


**\$** gulp watch *(runs watch task)*



# Gulp API

# gulp.task(name[, deps], fn)


```
gulp.task('js', function() {  
  gulp.src('app/js/main.js')  
    .pipe(uglify())  
    .pipe(concat())  
    .pipe(gulp.dest('build'))  
});
```



Registers a task name with  
a function

# gulp.src(globs[, options])

```
gulp.src('app/css/*.less')  
  .pipe(less())  
  .pipe(concat('main.css'))  
  .pipe(gulp.dest('public/css'))
```



Takes a file system glob and starts emitting files that match.

# gulp.dest(path[, options])

```
gulp.src('app/index.html')  
  .pipe(gulp.dest('public/html'))
```



Piped files are written to the file system

# gulp.watch(glob[, opts], tasks)

```
gulp.task('watch', function() {  
  gulp.watch(['./js/**/*.js'], ['js']);  
});
```



Run a function when glob  
change

**That's it.**

**Gulp has just 4 function.**



**Let's discover some plugins**

# Common Gulp Plugins

```
$ npm install
```

- `gulp-minify-css`
- `gulp-uglify`
- `gulp-concat`
- `gulp-ng-annotate`
- `gulp-ngdocs`
- `gulp-ng-html2js`

# gulp-minify-css

```
var minifyCSS = require('gulp-minify-css');
```

```
gulp.task('minify-css', function() {  
  gulp.src('./static/css/*.css')  
    .pipe(minifyCSS({keepBreaks: true}))  
    .pipe(gulp.dest('./dist/'))  
});
```

# gulp-uglify

```
var uglify = require('gulp-uglify');
```

```
gulp.task('compress', function() {  
  gulp.src('lib/*.js')  
    .pipe(uglify())  
    .pipe(gulp.dest('./dist/'))  
});
```

# gulp-concat

```
var concat = require('gulp-concat');
```

```
gulp.task('scripts', function() {  
  gulp.src('./lib/*.js')  
    .pipe(concat('all.js'))  
    .pipe(gulp.dest('./dist/'))  
});
```

**What about Angular  
plugins ?**



# gulp-ng-annotate

```
var ngAnnotate = require('gulp-ng-annotate');
```

```
gulp.task('default', function () {  
  gulp.src('src/app.js')  
    .pipe(ngAnnotate())  
    .pipe(gulp.dest('dist'));  
});
```

# gulp-ng-docs

```
var gulpDocs = require('gulp-ngdocs');
```

```
gulp.task('ngdocs', function () {  
  gulp.src('path/to/src/*.js')  
    .pipe(gulpDocs.process())  
    .pipe(gulp.dest('./docs'));  
});
```

# gulp-nghtml2js

```
var ngHtml2Js = require("gulp-ng-html2js");
```

```
gulp.task("./partials/*.html")  
  .pipe(ngHtml2Js({  
    moduleName: "MyAwesomePartials",  
    prefix: "/partials"  
  }))  
  .pipe(gulp.dest("./dist/partials"));
```

**Demo Time**

# Questions

# Thanks.

**Follow us**

**@cihadhoruzoglu**

**@techciceksepeti**