**Feature Packages**

***@angular/core***: Critical runtime parts of the framework needed by every application. Includes all metadata decorators, Component, Directive, dependency injection, and the component lifecycle hooks.

***@angular/common***: The commonly needed services, pipes, and directives provided by the Angular team.

***@angular/compiler***: Angular's *Template Compiler*. It understands templates and can convert them to code that makes the application run and render. Typically you don’t interact with the compiler directly; rather, you use it indirectly via platform-browser-dynamic or the offline template compiler.

***@angular/platform-browser***: Everything DOM and browser related, especially the pieces that help render into the DOM. This package also includes the bootstrapStatic() method for bootstrapping applications for production builds that pre-compile templates offline.

***@angular/platform-browser-dynamic***: Includes [Providers](https://v2.angular.io/docs/ts/latest/api/core/index/Provider-type-alias.html) and a [bootstrap](https://v2.angular.io/docs/ts/latest/guide/ngmodule.html#bootstrap) method for applications that compile templates on the client. Don’t use offline compilation. Use this package for bootstrapping during development and for bootstrapping plunker samples.

***@angular/http***: Angular's HTTP client.

***@angular/router***: Component router.

***@angular/upgrade***: Set of utilities for upgrading AngularJS applications to Angular.

[***system.js***](https://github.com/systemjs/systemjs): A dynamic module loader compatible with the [ES2015 module](http://www.2ality.com/2014/09/es6-modules-final.html) specification. Other viable choices include the well-regarded [webpack](https://webpack.github.io/).

Your future applications are likely to require additional packages that provide HTML controls, themes, data access, and various utilities.

**Polyfill packages**

Angular requires certain [polyfills](https://en.wikipedia.org/wiki/Polyfill) in the application environment. Install these polyfills using the npm packages that Angular lists in the *peerDependencies*section of its package.json.

You must list these packages in the dependencies section of your own package.json.

For background on this requirement, see [Why peerDependencies?](https://v2.angular.io/docs/ts/latest/guide/npm-packages.html#why-peer-dependencies).

***core-js***: Patches the global context (window) with essential features of ES2015 (ES6). You may substitute an alternative polyfill that provides the same core APIs. When these APIs are implemented by the major browsers, this dependency will become unnecessary.

***rxjs***: A polyfill for the [Observables specification](https://github.com/zenparsing/es-observable) currently before the [TC39](http://www.ecma-international.org/memento/TC39.htm) committee that determines standards for the JavaScript language. You can pick a preferred version of *rxjs* (within a compatible version range) without waiting for Angular updates.

***zone.js***: A polyfill for the [Zone specification](https://gist.github.com/mhevery/63fdcdf7c65886051d55) currently before the [TC39](http://www.ecma-international.org/memento/TC39.htm) committee that determines standards for the JavaScript language. You can pick a preferred version of *zone.js* to use (within a compatible version range) without waiting for Angular updates.

[Zone.js](https://github.com/angular/zone.js) provides a mechanism, called zones, for encapsulating and intercepting asynchronous activities in the browser (e.g. setTimeout, , promises).

These zones are execution contexts that allow Angular to track the start and completion of asynchronous activities and perform tasks as required (e.g. change detection). Zone.js provides a global zone that can be forked and extended to further encapsulate/isolate asynchronous behaviour, which Angular does so in its **NgZone** service, by creating a fork and extending it with its own behaviours.

**NgZone** exposes a set of Observables that allow us to determine the current status, or *stability*, of Angular's zone.

* *onUnstable* – Notifies when code has entered and is executing within the Angular zone.
* *onMicrotaskEmpty* - Notifies when no more microtasks are queued for execution. *Angular subscribes to this internally to signal that it should run change detection.*
* *onStable* – Notifies when the last onMicroTaskEmpty has run, implying that all tasks have completed and change detection has occurred.
* *onError* – Notifies when an error has occurred. *Angular subscribes to this internally to send uncaught errors to its own error handler, i.e. the errors you see in your console prefixed with 'EXCEPTION:'.*

**Other helper libraries**

***angular-in-memory-web-api***: An Angular-supported library that simulates a remote server's web api without requiring an actual server or real HTTP calls. Good for demos, samples, and early stage development (before you even have a server). Read about it in the [HTTP Client](https://v2.angular.io/docs/ts/latest/guide/server-communication.html#in-mem-web-api) page.

***bootstrap***: [Bootstrap](http://getbootstrap.com/) is a popular HTML and CSS framework for designing responsive web apps. Some of the samples improve their appearance with *bootstrap*.

***devDependencies***

The packages listed in the *devDependencies* section of the package.json help you develop the application. You don't have to deploy them with the production application although there is no harm in doing so.

[***concurrently***](https://www.npmjs.com/package/concurrently): A utility to run multiple *npm* commands concurrently on OS/X, Windows, and Linux operating systems.

[***lite-server***](https://www.npmjs.com/package/lite-server): A light-weight, static file server, by [John Papa](http://johnpapa.net/) with excellent support for Angular apps that use routing.

[***typescript***](https://www.npmjs.com/package/typescript): the TypeScript language server, including the *tsc* TypeScript compiler.

***@types/\****: TypeScript definition files. Learn more about it in the [TypeScript Configuration](https://v2.angular.io/docs/ts/latest/guide/typescript-configuration.html#typings) guide.