1) As we do not want to run our application using superuser rights, In the case of an error/bug, a process can bring down the entire system, as it has credentials to do anything. So we won't be able to listen on port 80. we can do is to set up an HTTP server/proxy to forward the requests. This can be nginx or Apache

2) 'use strict'
   It eliminates some silent errors and will throw them all the time.

   It allows the code to run faster because the interpreter knows the code it's optimized.

3)we can use https module instead of http module in order to communicate over a secure channel with the client.

4) JavaScript's object system is based on prototypes, class inheritance is implemented on top of prototypal inheritance, as a result, it creates the tightest coupling available in OO design.

 we can use object composition over class inheritance.

5) The library used in the software prototype.js is old may contain series vulnerabilities
We can use es6 class or modules based approach

6)We can use global function appRequire to require any modules as the code becomes big it will become difficult to maintain it can become clumsy and leads to error and performance