- Built on top of nodejs HTTP facilities.
- Allows dispatching HTTP requests based on a modular routing framework.
- Allows inserting *middleware* into the applications request-response cycle.
- Allows plugging in multiple **template engines** to generate dynamic content.

- Routing allows specifying that a **handler** function (single or multiple) run when the URL for a web request matches a particular route.
- Matching route can be a fixed string or a pattern or a regular expression.
- Handler function takes a `req` and `res` argument to access the incoming `Request` and outgoing `Response`. Optionally, it can also take a `next` argument; calling `next()` allows chaining to the next set of handlers for a route which matches the request URL.

- Allows inserting arbitrary code into request-response cycle. Can make changes to the request and response objects, end the cycle, call the next middleware.
- Can be used for logging, caching, cookies and many other services.
- Set up similarly to routes except that it is specified using use().

Provide a web service for maintaining users. Provides following methods.

GET /users Return information for all current users.

GET /users/*userId* Return information for existing user *userId*.

POST /users Create new user specified using JSON request body.

PUT /users/*userId* Replace information for existing user *userId* by JSON request body.

PATCH /users/*userId* Update information for existing user *userId* by JSON request body.

DELETE /users/*userId* Delete information for existing user *userId*.

Modification of earlier user-store. Changes include:

- Deal with only a single user rather than a list of users.
- No default id (defaulting to configured git email).
- Read method changes to allow any filter (including none).
- Services provides are create, delete, load, read, replace and update.

- Top-level index.js. Start web server using:

      $ ./index.js 1234 simpsons.json

  to run on port 1234 loading data from `simpsons.json`.
- Web services user-ws.js.
- Underlying store user-store.js.