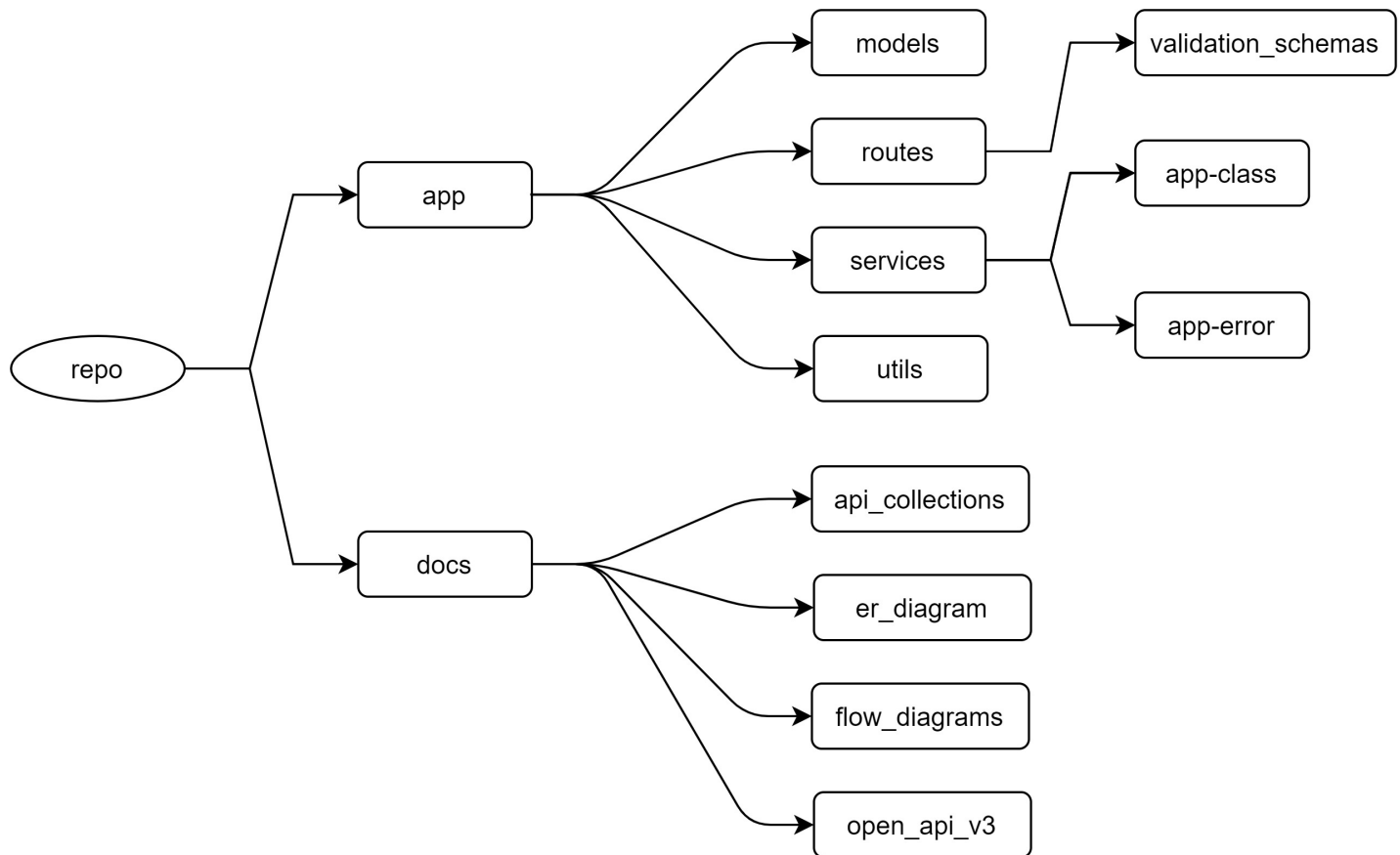


## Repository Organisation



- A repo consists of a lot more supporting files like app.js, package.json, network files and which the dev team doesn't work on.
  - The naming convention of
- Dev team will work on these two folders:
  - **app**: This folder consists of all the application logic and implementation. This **app** folder is divided into 4 sub-folders for easy organisation.
    - **models**: This folder contains data schemas and exposes database access methods like create, read, update, delete.
    - **routes**: This folder contains all the routes which can be used by other applications apart from this folder also contains another sub-folder "**validation\_schemas**" which has validation schemas of query\_params and payload in **open\_api\_v3** JSON format
    - **services**: This folder is where all the business logic and application logic is present. Each operation or feature is maintained as a separate package(folder). It contains two important folders:
      - **app-class**: This class should be imported in every service and then extends this class.
      - **app-error**: This class should be imported in every service and implements this class for an exception or to throw an exception
    - **utils**: This folder contains all the utility functions used in any module. For example like s3 put/get, logger, mongoose bootstrapper, bookshelf connector.
  - **docs**: This folder contains all the documents which are related to business process, features, routes schemas, API documentation.
    - **api\_collection**: This folder contains postman collection of api with examples.
    - **er\_diagram**: This folder contains all the database designs, relationship diagrams and attributes present in each of the collections

- **flow\_digrams:** This folder contains the all execution diagrams, BPMN diagrams , flow charts and sequence diagrams.
  - **NOTE:** What ever the diagrams put here should be named after the service name and the flow should be matched with code implimentation.
    - example if service name in "registration.js" then flow name should be "registration.flow\_diagram.jpeg" or "registration.bpmn.jpeg" or "registration.sequence\_diagram.jpeg"
  - **NOTE:** For each service in services folder must and should have atleast one flow diagram which illustrates the implimentation of code
- **open\_api\_v3:** This folder contains Swagger Specs of each API with Query Params, Payload and Response schemas.
  - **NOTE:** For every route in rotues folder must should have documentation in open\_api\_v3.