

Documentation technique

Architecture :

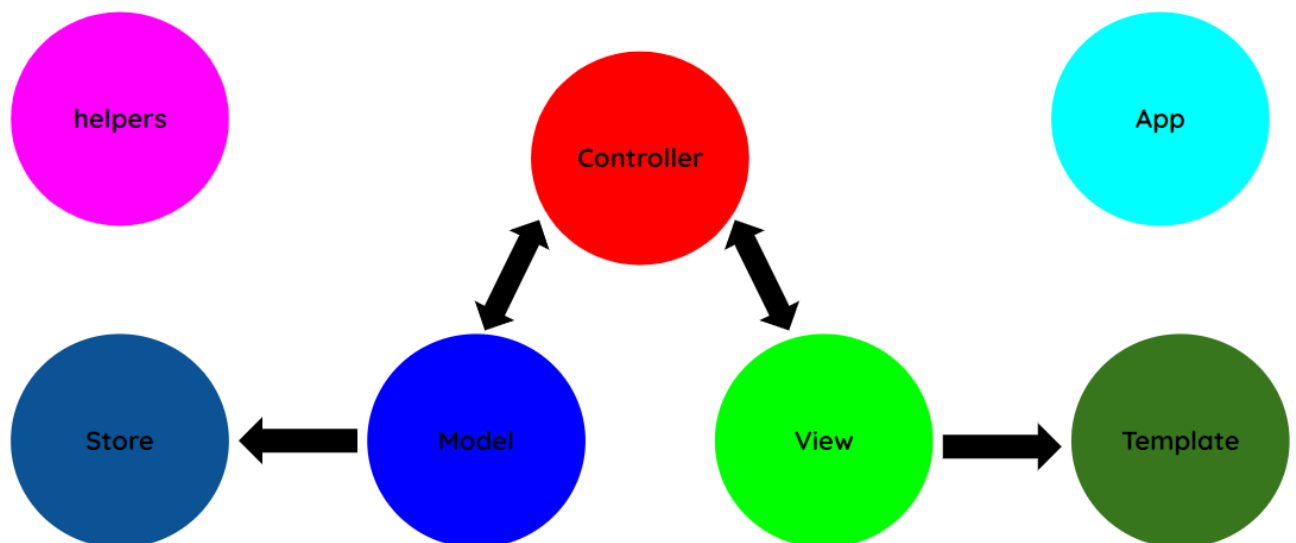
Notre application adopte une architecture MVC (Model View Controller).

Model (couche de données) : c'est ici que les données sont stockées pour l'application. Le model conserve la liste des todos et est responsable de toutes les actions effectuées sur chaque objet todo.

View (couche de présentation) : Cette partie de l'application à accès au dom et est responsable de la configuration des gestionnaires d'évènements (clic, double clics ...). La vue est également responsable de la présentation du code HTML. Dans l'application la vue sera chargée d'affiché la Todo list a l'utilisateur. Cependant, chaque fois qu'un utilisateur entre une nouveau todo via le champ de saisie, la vue utilise un répartiteur d'élément pour notifier le contrôleur, qui met alors le modèle à jour.

Controller (logique d'application) : le contrôleur est le liant entre le model et la vue. Le contrôleur traite et répond aux évènements définis par le modèle ou la vue. Il met à jour le modèle à chaque fois que l'utilisateur manipule la vue et peut également être utilisé pour mettre à jour la vue lorsque le modèle change.

On peut ici voir l'architecture de l'application.



Elle contient 4 fichiers en plus des fichiers MVC :

- *[store.js](#) qui est lié au model, il manipule les données du stockage local.*
- *[template.js](#) qui est lié à la vue, il fournit un pattern de mise en page d'un todo.*
- *[app.js](#) sert de base à l'application. Initialise les différents objets.*
- *[helpers.js](#) fournit des raccourcis utilisés dans les fichiers principaux.*

*Pour voir la documentation technique,
veuillez suivre le chemin :*

[projet-8_todo-list-app/out/index.html](#)

*Pour voir les tests effectués avec Jasmine,
veuillez suivre le chemin :*

[projet-8_todo-list-app/test/specRunner.html](#)