Template	Parent Class (if any): List the parent class if applicable Subclasses (if any): List all the subclasses separated by a comma
Responsibilities:	Collaborators:

LoginPage	Parent Class: App.js Subclasses: LoginForm.js
Responsibilities: Handles the UI for the login page	Collaborators: Rahul, David

LoginForm	Parent Class: LoginPage.js
Responsibilities: Handles logging in users to the system (once integrated with the DB)	Collaborators: Rahul, David

SignupPage	Parent Class: App.js Subclasses: SignupForm.js
Responsibilities: Handles the UI for the signup page	Collaborators: Rahul, David

SignupForm	Parent Class: SignupPage.js
Responsibilities: Handles creating new users and storing the information in the DB	Collaborators: Rahul, David

ForgotPasswordPage	Parent Class: App.js Subclasses: ForgotPasswordForm.js

Responsibilities: Handles the UI for the forget password page	Collaborators: Rahul, David

ForgotPasswordForm	Parent Class: ForgotPasswordForm.js
Responsibilities: Handles accessing the user who requests a password reset	Collaborators: Rahul, David

Graph.js	Parent Class: N/A Subclasses: N/A
Responsibilities: Handles and servers the schema for the graphql + neo4j api.	Collaborators: Daksh

Timer Page	Parent Class: App.js Subclasses: TimerPage.js
Responsibilities: Handles timer	Collaborators: Shun Chit

Timer	Parent Class: TimerPage.js Subclasses: Timer.js
Responsibilities: Create a format for the timer for pomodoro	Collaborators: Shun Chit

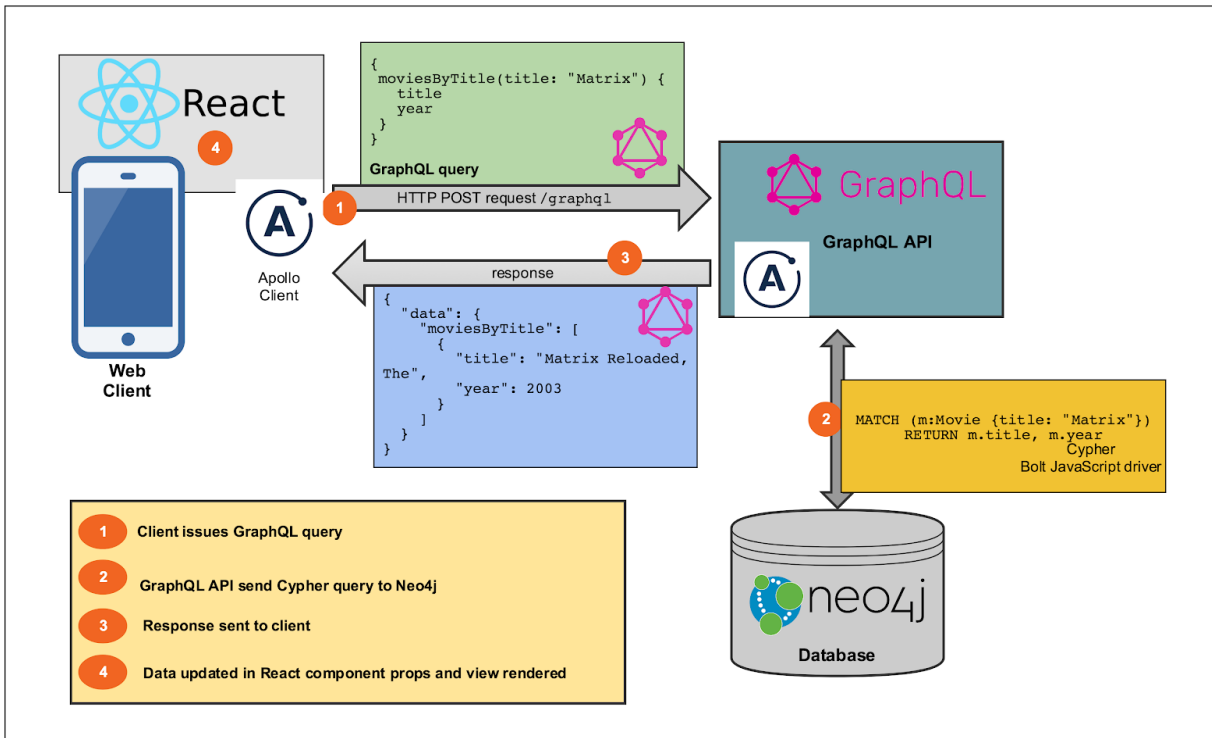|	
Set	Parent Class: Timer.js Subclasses: Set.js
Responsibilities: Handles converting time, start time, pause time	Collaborators: Shun Chit

SettingContext	Parent Class: Timer.js Subclasses: SettingContext.js
Responsibilities: Handles the setting for the timer	Collaborators: Shun Chit

Main Interface	Parent Class: App.js Subclasses: N/A
Responsibilities: Allows for the changing of pages and tabs, and calendar view and notes	Collaborators: Manjoy malhotra

Tasks Page	Parent Class: App.js
Responsibilities: Lists out the tasks the user still has to do, and can add more	Collaborators: Peter Ghobrial

Journal	Parent Class: App.js
Responsibilities: Creates journal posts so the user can write and create documents	Collaborators: Yuvraaj

Software Architecture Diagram

This image is taken from
https://grandstack.io/docs/getting-started-neo4j-graphql.

The architecture relies on a neo4j database being hosted on the same or a separate server. We also rely on node.js being installed. Node.js is then used to install react and other react libraries.