**Ans: to the question no:- 01**

**What is Inertia js**:

It is called the modern monolith because it comes with a new approach to building classic server-driven web apps. For creating modern SPAs. It allows you to create fully client-side rendered, single-page apps, without any complexity. Inertia isn't a framework.

**Working procedure:** You can bulid morden application using Inertia with your server-side web frame work like Laravel, rails without using their template. Rather you can use entire frontend using react, Vue, svelte . Inertia works like a middle man between the server-side & client side with out using any api.

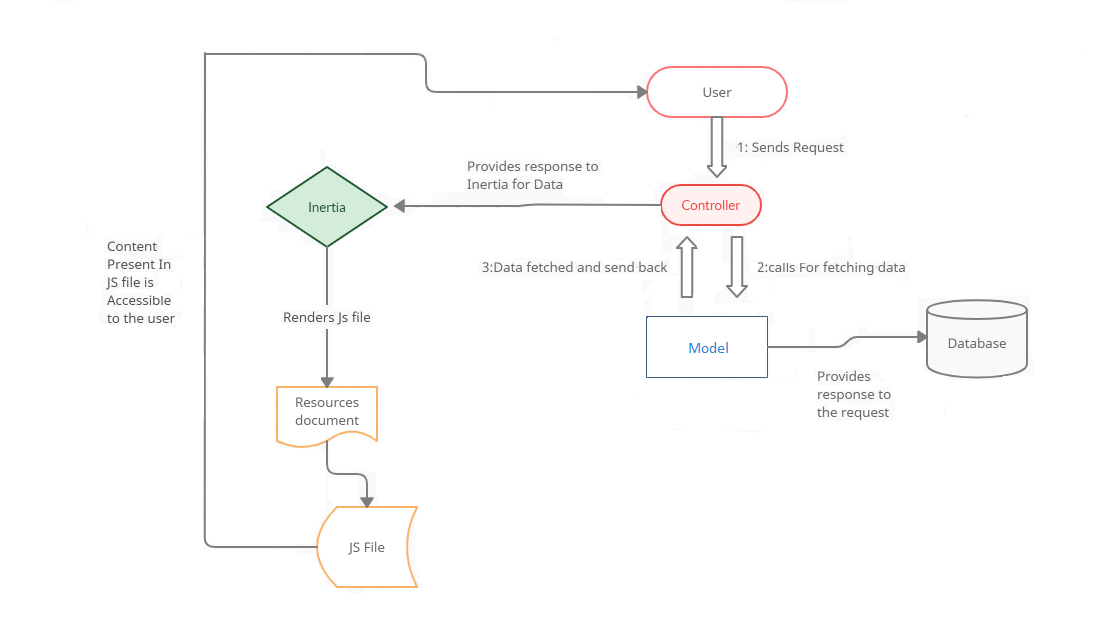


Fig: working procedure of inertia

**Ans: to the question no:- 02**

**Comparison of SSR and CSR:**

1. **SSR :**

SSR stands for server-side rendering. it is the ability of a web application to render the web page on the server instead of rendering it in the browser. when the page arrived on the client-side, it is fully rendered. it is because the server-side has fully rendered the page before it was sent by the server to the client. when the request is received on the server-side, it will compile everything, if the content of the page needed data from the database, the server will do that, then render the data into the fully rendered page and then send it to the client as the response. now, what if the client navigates to a different route? every time the client navigates to a different route, the server will do the work all over again.

1. **CSR:**

CSR stands for client-side rendering. overall, CSR is the opposite of SSR. If the SSR renders the page on the server-side, CSR renders the page on the client-side. when the request is received on the server, it will not render the page, instead, the server will send a single page that will be the skeleton of the page to the client. the server sends the page along with the JavaScript file. later, the js will turn the page into a fully rendered page. so where is the content? what if the page needs to take data from the database? Then, the api comes in. the client will make a request to the api to take the data and then render it to the page. Lastly, what if the client navigates to a different route? do the server will send the page again ? the server will not send the page again, instead, the client will re-render the page according to the route that client requested. so the page that is used, is always the same page as the first request.

**Difference between CSR and SSR:**

The main difference between CSR and SSR is where the page is rendered. SSR renders the page on the server-side and CSR renders the page on the client-side. Client-side manages the routing dynamically without refreshing the page every time the client requests a different route.

**Advantage of SSR:**

1. If SEO is your priority, typically when you are building a blog site and you want everyone who searching on google go to your website, then SSR is your choice.
2. If your website needs a faster initial loading.
3. If the content of your website doesn't need much user interaction.

**Advantage of CSR:**

* when SEO is not your priority.
* if your site has rich interactions.
* if you are building a web application.

**Disadvantages of Server-side Rendering :**

1. Increased server load.
2. More complexity
3. Delayed Interactivity
4. Increased bandwidth usage

**Disadvantages of client-side Rendering :**

1. Decreased server load.
2. Less complexity
3. Less Delayed Interactivity
4. Decreased bandwidth usage

**Ans: to the question no:- 03**

**Features:** There are many key features of Inertia js. They are:

**Ans: to the question no:- 05**