Copilot

To create a personal development environment with Next.js, follow these steps:

Setting Up the Project

1. **Initialize Next.js Project**:

```
npx create-next-app@latest personal-dev-env
cd personal-dev-env
npm run dev
```

Implementing the Functionalities

1. **Diary with Notes and Canvas**:
    - Create a page for the diary.
    - Use a text area for notes and a canvas element for brainstorming.

```
// pages/diary.js
import { useState } from 'react';

export default function Diary() {
  const [notes, setNotes] = useState('');

  return (
    <div>
      <textarea value={notes} onChange={(e) => setNotes(e.target.value)} placeholder="Write your not
      <canvas id="brainstormCanvas" width="500" height="500"></canvas>
    </div>
  );
}
```

2. **GitHub Section**:
    - Fetch user repositories using GitHub API.

```
// pages/github.js
import { useEffect, useState } from 'react';

export default function GitHub() {
  const [repos, setRepos] = useState([]);

  useEffect(() => {
    fetch('https://api.github.com/users/YOUR_USERNAME/repos')
      .then((res) => res.json())
      .then((data) => setRepos(data));
  }, []);

  return (
    <div>
      {repos.map((repo) => (
        <a key={repo.id} href={repo.html_url} target="_blank" rel="noopener noreferrer">
          {repo.name}
        </a>
      ))}
    </div>
  );
}
```

3. **Password Manager**:
    - Use local storage to manage passwords.

```
// pages/password-manager.js
import { useState } from 'react';
```

```
export default function PasswordManager() {
  const [passwords, setPasswords] = useState([]);

  const addPassword = (site, password) => {
    const newPasswords = [...passwords, { site, password }];
    setPasswords(newPasswords);
    localStorage.setItem('passwords', JSON.stringify(newPasswords));
  };

  return (
    <div>
      <input type="text" placeholder="Site" id="site" />
      <input type="password" placeholder="Password" id="password" />
      <button onClick={() => addPassword(document.getElementById('site').value, document.getElementB
      <ul>
        {passwords.map((item, index) => (
          <li key={index}>{item.site}: {item.password}</li>
        ))}
      </ul>
    </div>
  );
}
```

4. **To-Do List**:
    - Implement a simple to-do list.

```
// pages/todo.js
import { useState } from 'react';

export default function ToDo() {
  const [tasks, setTasks] = useState([]);
  const [task, setTask] = useState('');

  const addTask = () => {
    setTasks([...tasks, task]);
    setTask('');
  };

  return (
    <div>
      <input type="text" value={task} onChange={(e) => setTask(e.target.value)} placeholder="New Tas
      <button onClick={addTask}>Add</button>
      <ul>
        {tasks.map((task, index) => (
          <li key={index}>{task}</li>
        ))}
      </ul>
    </div>
  );
}
```

5. **Social Section**:
    - List social media links.

```
// pages/social.js
export default function Social() {
  return (
    <div>
      <a href="https://twitter.com/YOUR_USERNAME" target="_blank" rel="noopener noreferrer">Twitter<
      <a href="https://linkedin.com/in/YOUR_USERNAME" target="_blank" rel="noopener noreferrer">Link
    </div>
  );
}
```

6. **Links Section**:
    - Save and display favorite links.

```
// pages/links.js
import { useState } from 'react';
```

```
export default function Links() {
  const [links, setLinks] = useState([]);
  const [link, setLink] = useState('');

  const addLink = () => {
    setLinks([...links, link]);
    setLink('');
  };

  return (
    <div>
      <input type="text" value={link} onChange={(e) => setLink(e.target.value)} placeholder="New Lir
      <button onClick={addLink}>Add</button>
      <ul>
        {links.map((link, index) => (
          <li key={index}><a href={link} target="_blank" rel="noopener noreferrer">{link}</a></li>
        ))}
      </ul>
    </div>
  );
}
```

7. **Document Section**:
    - Store and display important documents.

```
// pages/documents.js
import { useState } from 'react';

export default function Documents() {
  const [documents, setDocuments] = useState([]);
  const [document, setDocument] = useState('');

  const addDocument = () => {
    setDocuments([...documents, document]);
    setDocument('');
  };

  return (
    <div>
      <input type="text" value={document} onChange={(e) => setDocument(e.target.value)} placeholder=
      <button onClick={addDocument}>Add</button>
      <ul>
        {documents.map((doc, index) => (
          <li key={index}>{doc}</li>
        ))}
      </ul>
    </div>
  );
}
```

Additional Functionalities

1. **Clock and Date**:
    - Display current time and date.

```
// components/Clock.js
import { useEffect, useState } from 'react';

export default function Clock() {
  const [time, setTime] = useState(new Date());

  useEffect(() => {
    const timer = setInterval(() => setTime(new Date()), 1000);
    return () => clearInterval(timer);
  }, []);

  return (
    <div>
      {time.toLocaleTimeString()} - {time.toLocaleDateString()}
    </div>
```

```
  );
}
```

2. **Water Reminder**:
    - Set an interval to remind the user to drink water.

```
// components/WaterReminder.js
import { useEffect } from 'react';

export default function WaterReminder() {
  useEffect(() => {
    const reminder = setInterval(() => alert('Time to drink water!'), 1200000); // 20 minutes
    return () => clearInterval(reminder);
  }, []);

  return null;
}
```

3. **Pomodoro Clock**:
    - Implement a Pomodoro timer.

```
// components/Pomodoro.js
import { useState, useEffect } from 'react';

export default function Pomodoro() {
  const [seconds, setSeconds] = useState(1500); // 25 minutes
  const [isActive, setIsActive] = useState(false);

  useEffect(() => {
    let interval = null;
    if (isActive) {
      interval = setInterval(() => {
        setSeconds((seconds) => seconds - 1);
      }, 1000);
    } else if (!isActive && seconds !== 0) {
      clearInterval(interval);
    }
    return () => clearInterval(interval);
  }, [isActive, seconds]);

  return (
    <div>
      <div>{Math.floor(seconds / 60)}:{seconds % 60}</div>
      <button onClick={() => setIsActive(!isActive)}>{isActive ? 'Pause' : 'Start'}</button>
      <button onClick={() => setSeconds(1500)}>Reset</button>
    </div>
  );
}
```

4. **Change Background Image**:
    - Allow the user to change the background image.

```
// pages/index.js
import { useState } from 'react';

export default function Home() {
  const [bgImage, setBgImage] = useState('');

  return (
    <div style={{ backgroundImage: `url(${bgImage})`, height: '100vh' }}>
      <input type="text" placeholder="Enter image URL" onChange={(e) => setBgImage(e.target.value)}
    </div>
  );
}
```

Integrating Components

- Import and use these components in your main layout or specific pages as needed.

```
// pages/_app.js
import '../styles/globals.css';
import Clock from '../components/Clock';
import WaterReminder from '../components/WaterReminder';
import Pomodoro from '../components/Pomodoro';

function MyApp({ Component, pageProps }) {
  return (
    <div>
      <Clock />
      <WaterReminder />
      <Pomodoro />
      <Component {...pageProps} />
    </div>
  );
}

export default MyApp;
```

This setup will create a comprehensive personal development environment with Next.js, incorporating all the specified functionalities.