KDM

ADVICE REPORT TO STAKEHOLDERS

K2DM

MEMBERS

Donald
Kingsley
Dang Khoa
Matthew

TABLE OF CONTENTS

ABSTRACT

This advice report outlines the steps and considerations necessary to successfully transfer the Pixel Playground Knowledge Database website to our client. The primary objective of this advice report is to ensure a seamless transition, enabling the client to deploy and utilize the website without encountering any operational issues. Key recommendations involve the client setting up a MongoDB account and updating the configuration. The anticipated outcomes include a fully functional and independent deployment of the website by the client, marking the successful transfer of the project.

OBJECTIVES

The objectives of this advice report are as follows:
1. Facilitate Smooth Transition: To provide a clear and concise guide to assist the client in seamlessly taking over the Pixel Playground Knowledge Database website and ensure that the transition process is smooth, and that the client can readily manage and maintain the website.
2. Ensure Proper Configuration: To outline the steps necessary for the client to update and configure the project settings, including MongoDB account details, and environment variables. This will ensure that the website functions correctly.

METHODOLOGY

1. Interviews:
   Objective: Understanding the perspectives, expectations, and requirements of key stakeholders involved in the project, and end-users.
   Process: Conducted one-on-one interviews with project stakeholders to gather insights into their goals, preferences, and concerns.

2. Surveys:
   Objective: Collecting feedback from potential users to identify user preferences, and feature priorities.
   Process: Designed and distributed surveys to our potential users. The surveys covered questions such as user interface preferences, content accessibility, and feature preferences. Analyzed survey responses to inform design and development recommendations.

3. User Testing:
   Objective: Assessing the usability and functionality of the Pixel Playground Knowledge Database through direct user interaction.
   Process: Conducted user testing sessions with a diverse group of participants. Observed user interactions with the website, collected feedback on usability issues, and identified pain points. Iteratively refined the user interface and functionality based on user testing results.

4. Iterative Design and Development:
   Objective: Employing an iterative approach to design and development based on continuous feedback loops.
   Process: Implemented changes to the website design and functionality based on insights gained from interviews, user surveys, and testing. Regularly reviewed and refined project elements to align with user needs and expectations.

5. Feedback Analysis:
   Objective: Analyzing feedback from users and users to identify common themes, recurring issues, and areas for improvement.
Process: Compiled and analyzed feedback received during interviews, surveys, and user testing sessions.

CURRENT SITUATION ANALYSIS

STRENGTHS

Strengths of the current interactive product, the Pixel Playground Knowledge Database website, include:

1. Functional CMS: The website features a functional Content Management System (CMS) built with Node.js and Express. This allows for efficient management and organization of tutorial content, including content uploads, editing, and deletions.

2. MongoDB Atlas Integration: Integration with MongoDB Atlas enhances the handling of tutorial content uploads, editing, and deletions. This cloud-based database solution offers reliability, scalability, and ease of management for content storage.

3. Responsive Design: The website is responsive on both mobile and desktop devices. A responsive design ensures a seamless and visually appealing user experience across various screen sizes, enhancing accessibility and usability.

4. User Interaction: The CMS allows for user-friendly interactions, making it easier for administrators to update, edit, and delete tutorial content.

WEAKNESSES

1. Responsive Issues on Certain Screens: The website may experience responsive issues on specific screens, such as iPads. Addressing these issues is crucial to ensure a consistent and user-friendly experience across a variety of devices.

2. Hardcoded Credentials for Admin Section: The admin section responsible for adding new tutorials currently relies on hardcoded username and password credentials. This represents a security vulnerability and could be improved to enhance the overall security of the website.

OPPORTUNITIES

1. Gamification of Tutorials: Introduce gamification elements within tutorials to enhance user engagement and motivation. Implement features such as badges, points, or levels based on tutorial completion and active participation. This can create a more interactive and enjoyable learning experience.

2. Feedback Mechanism for Tutorials: Implement a feedback mechanism for each tutorial, allowing users to rate and review the content. Positive feedback can help highlight high-quality tutorials, while constructive criticism can guide improvements.

3. Booking System for the Pixel Playground Availability: Implement a booking system that allows users to check the availability of the Pixel Playground for use. This feature can streamline the process of reserving the space, ensuring a seamless experience for users interested in using the Pixel Playground.

THREATS

1. Security Vulnerabilities: The use of hardcoded usernames and passwords poses a security threat. A malicious person with knowledge of these credentials might be able to compromise the admin section, potentially leading to unauthorized modifications to the website content.

ADVICE AND RECOMMENDATIONS

OVERVIEW

The proposed advice aims to enhance the overall functionality, security, and user experience of the Pixel Playground project. The recommendations address identified weaknesses, leverage opportunities, and mitigate potential threats. The key focus areas include:

1. Enhancing Security Measures: Replace hardcoded usernames and passwords with an authentication system, such as user accounts with encrypted passwords. This will improve the security of the admin section, reducing the risk of unauthorized access.

2. Responsive Design Optimization: Address responsive design issues on devices like iPads, to ensure a seamless user experience across various screen sizes.

3. Feature Expansion - Booking System: Explore the opportunity to implement a booking system for users to check the availability of the Pixel Playground. This addition can enhance user engagement and provide a valuable feature for those interested in using the Pixel Playground.

4. Gamification of Tutorials: Introduce gamification elements to the tutorials, making the learning experience more interactive and engaging. Incorporate rewards, badges, or challenges to motivate users and sustain their interest in educational content.

5. Regular Updates and Maintenance: Establish a plan for regular updates and maintenance to keep the platform aligned with technological advancements, user expectations, and industry standards.

SPECIFIC RECOMMENDATIONS

1. Security: Implement an authentication system to replace hardcoded usernames and passwords, enhancing overall security. Utilize the following modules: passport, passport-local, bcryptjs, express-session, and connect-flash to establish a secure and customizable local authentication strategy. Ensure passwords are securely hashed and salted before storing in the database. This will significantly strengthen the protection of user credentials and reduce potential security vulnerabilities associated with hardcoded access credentials.

2. User Experience: Enhance responsiveness across all screens to ensure a seamless and enjoyable user experience. Conduct a thorough review of the application's display on various devices, addressing any responsiveness issues. Consider using more media queries to adapt the layout based on different screen sizes. This will result in an improved user interface, making the product more accessible and user-friendly across a diverse range of devices. Regular testing on different devices and screen sizes is recommended to maintain consistent responsiveness.

3. Budget and Financing: MongoDB, the database we used, will remain free if the dynamically stored and displayed content remains below 512MB. It's essential to monitor the content size regularly and plan for an upgrade if the database approaches this limit. Additionally, explore MongoDB's pricing models for larger storage requirements in the future.

IMPLEMENTATION PLAN

1. Security Enhancement:

Action Steps:
·Install necessary packages: npm install passport passport-local bcryptjs express-session sqlite3 connect-flash
·Implement Passport.js for local authentication.
·Integrate bcrypt for password hashing.
·Set up express-session for session management.
·Utilize connect-flash for displaying flash messages.

Resources Required:
·Developer familiar with Node.js, Express, and Passport.js.

2. User Experience Improvement:

Action Steps:
·Identify and resolve responsiveness issues on various screen sizes, including iPads.
·Optimize images and media for faster loading times.
·Conduct user testing to ensure a seamless experience.

Resources Required:
·Front-end developers, UX/UI designers.

3. Budget and Financing:

Action Steps:
·Regularly monitor MongoDB content size.
Review MongoDB pricing models for potential scaling.

CONCLUSION

In conclusion, this advice report outlines critical considerations and recommendations for the successful transfer and optimization of the Pixel Playground Knowledge Database website. The key points include:

1. Enhancing Security: The implementation of a proper authentication system is paramount to replacing hardcoded usernames and passwords. This measure significantly strengthens the overall security of the product, mitigating potential vulnerabilities.

2. Improving User Experience: By making the website more responsive across all screens, it will provide a seamless and enjoyable user experience. This will cater to a broader audience and improve overall user satisfaction.

3. Budget and Financial Planning: Notably, the MongoDB database, a core component of the project, will remain free if the content stored and displayed dynamically stays below 512MB.

USING THE WEBSITE: A STEP-BY-STEP GUIDE

STEP 1

Unzip the Project Files

STEP 2

Open the Zipped Folder in VS Code and Access Terminal
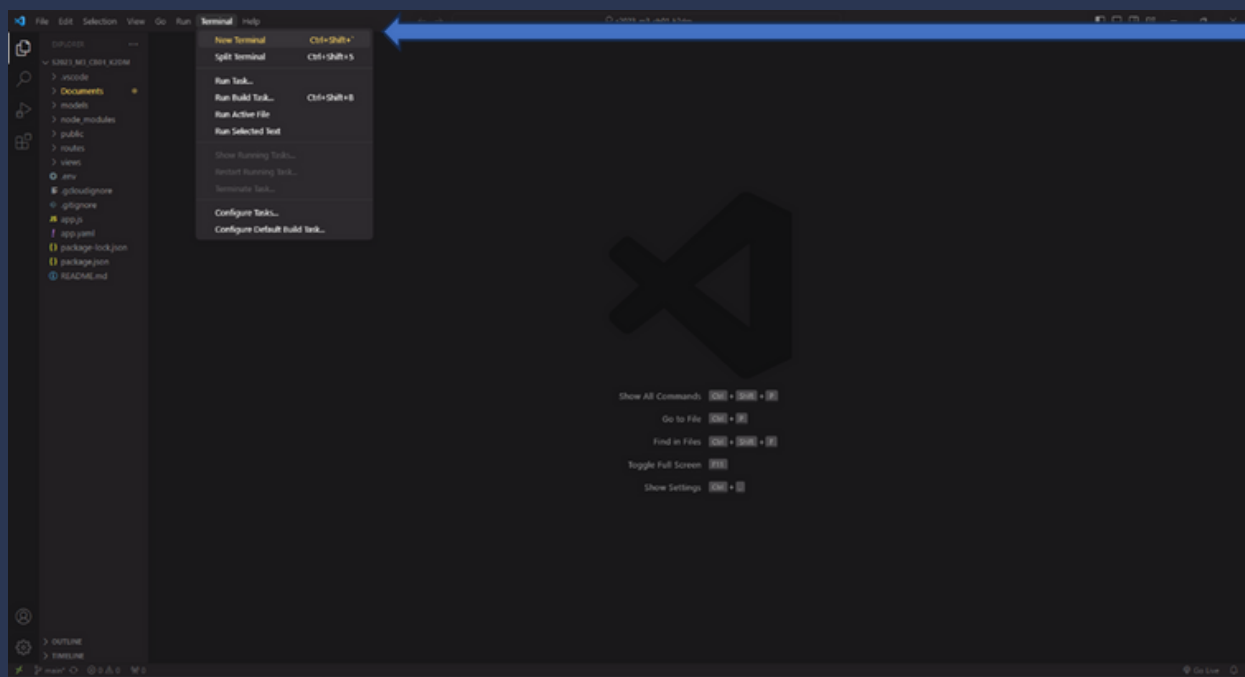
STEP 3

In the terminal, run the command 'node app.js' to start the application. This will start the server and connect to MongoDB Atlas.
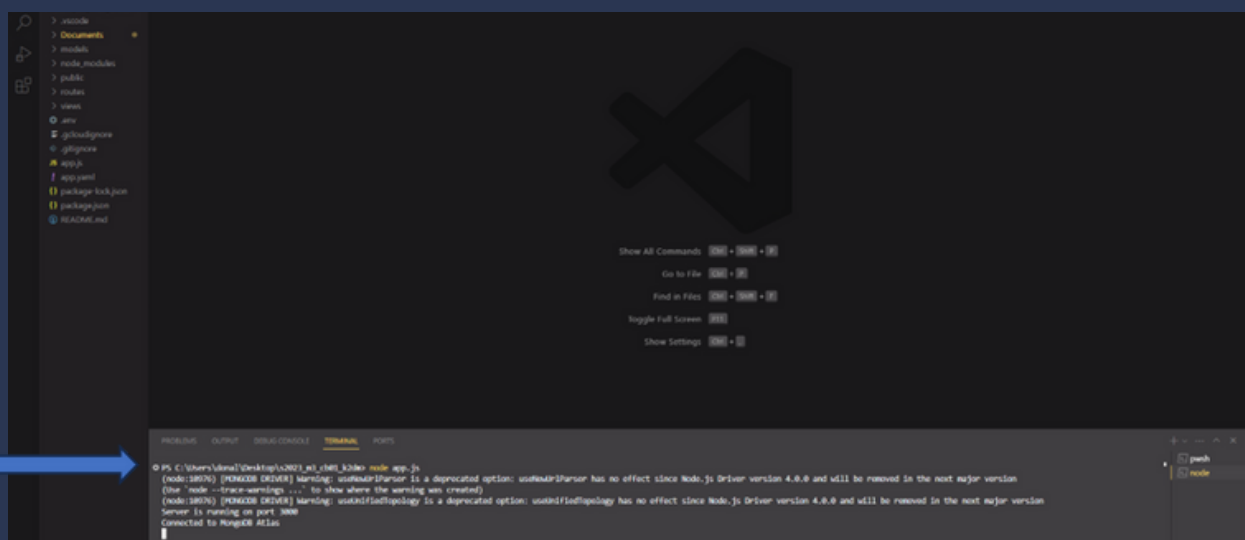
USING THE WEBSITE: A STEP-BY-STEP GUIDE

STEP 4

Open your browser and navigate to http://localhost:3000/
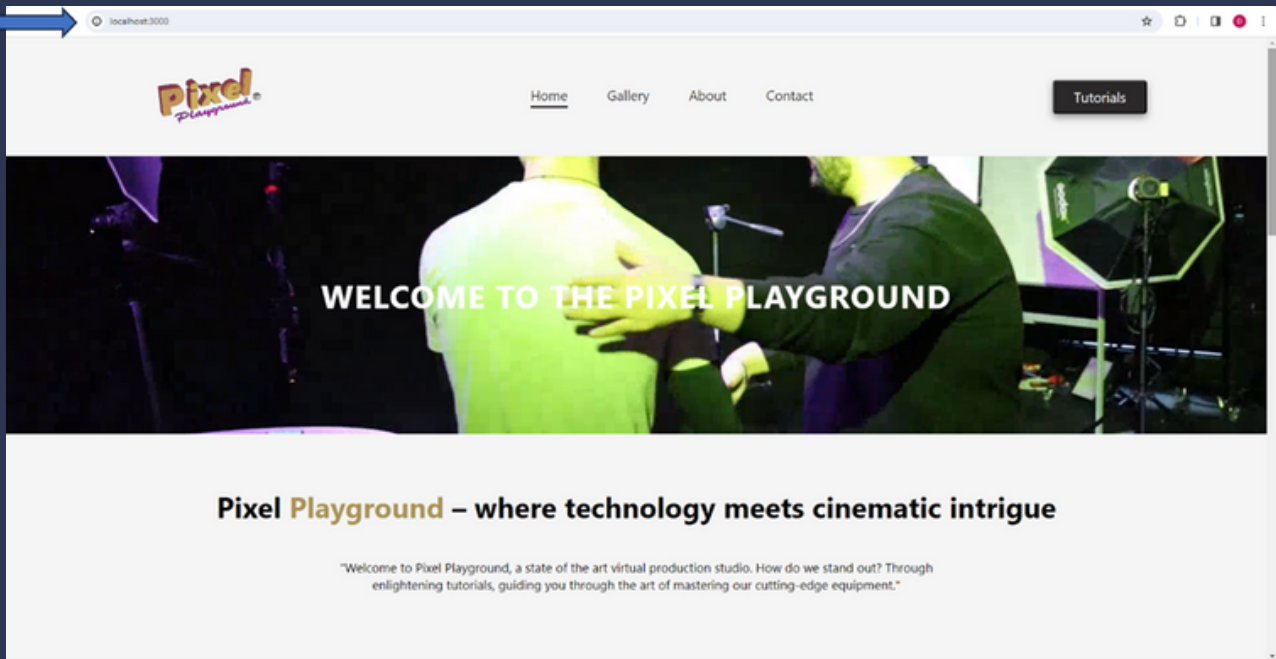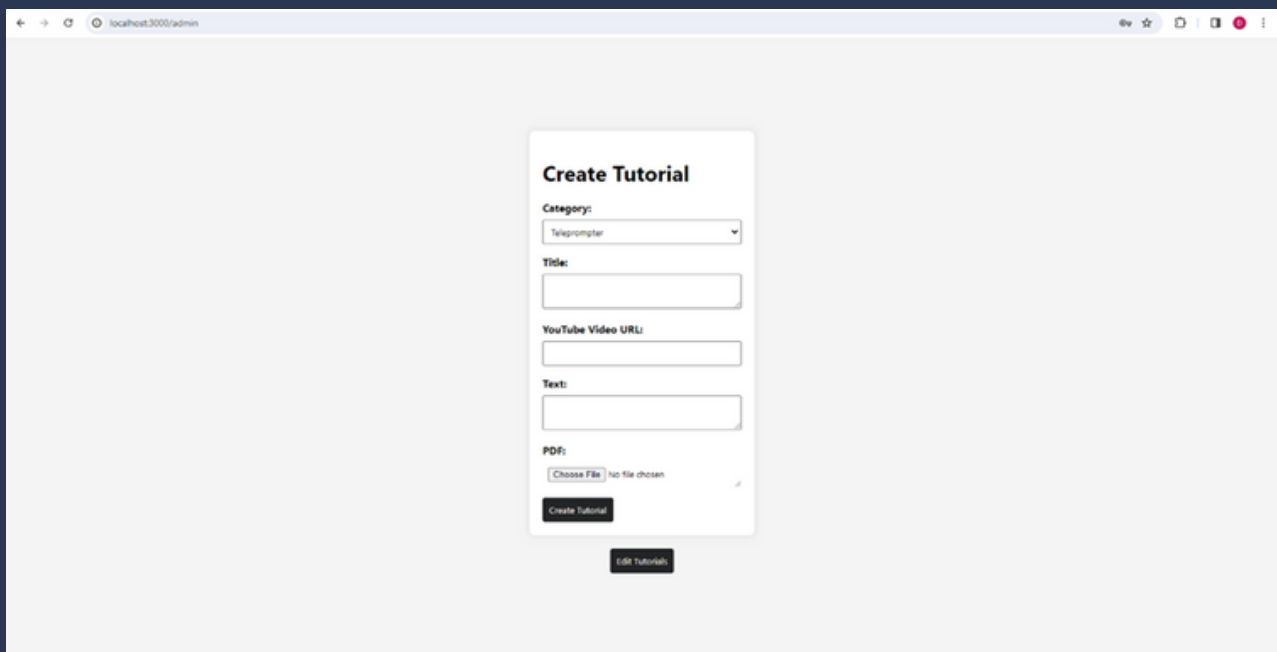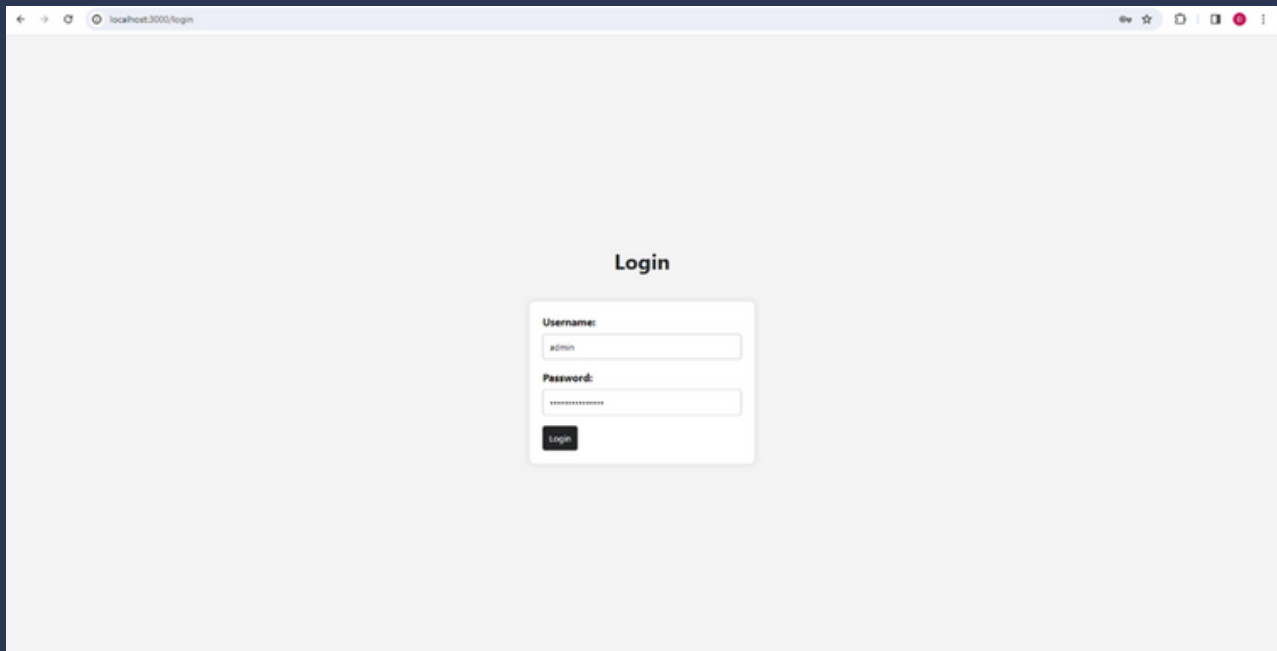

STEP 5

Access the admin panel to add new, edit and delete tutorials.

In your browser, navigate to http://localhost:3000/admin

USING THE WEBSITE: A STEP-BY-STEP GUIDE

STEP 6

Log in to the admin panel.

The admin username and password are provided in a separate admin.txt file in the project zip file.

USING THE WEBSITE: A STEP-BY-STEP GUIDE

STEP 7

Create a new post.

i.Select tutorial category from the dropdown menu.

ii.Input title of the tutorial.

iii.Navigate to the video of the tutorial on Youtube, click on share, click on embed and copy the Youtube ID of the video as highlighted in the screenshot below.
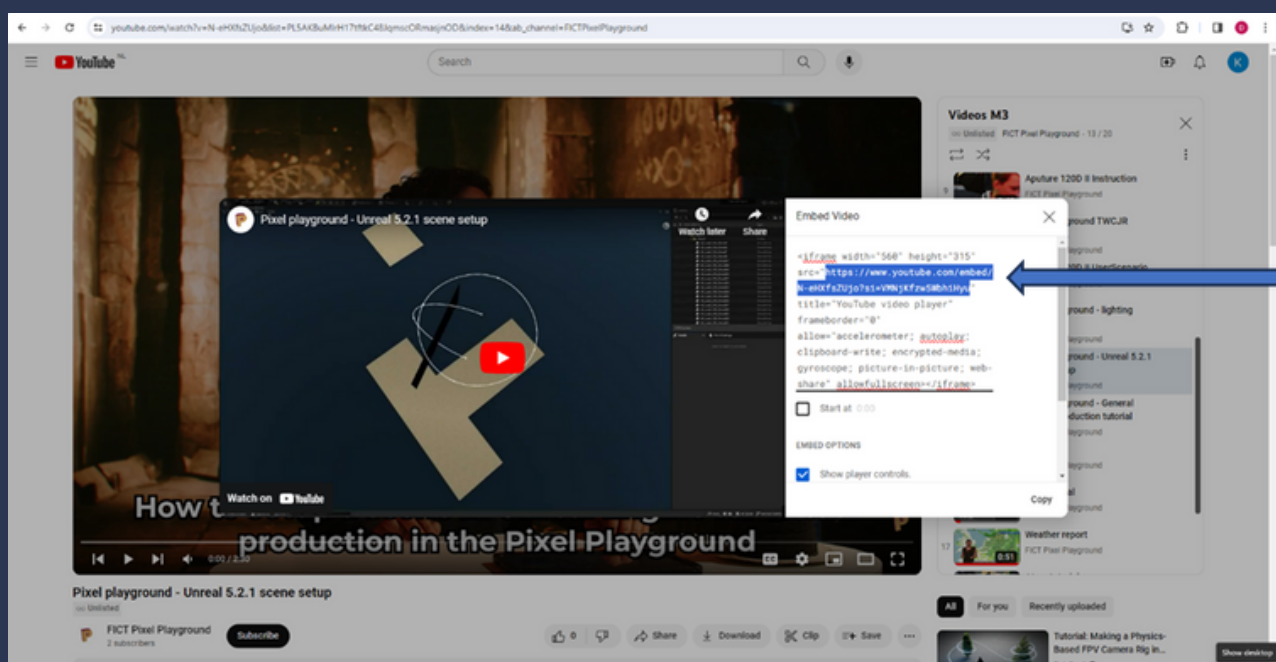
iv.Input a short description of the tutorial in the text field.

v.Upload pdf of the tutorial. (Please ensure to keep the file sizes moderate for efficient storage in MongoDB. Compress the PDF and consider the overall file size to enhance system performance).
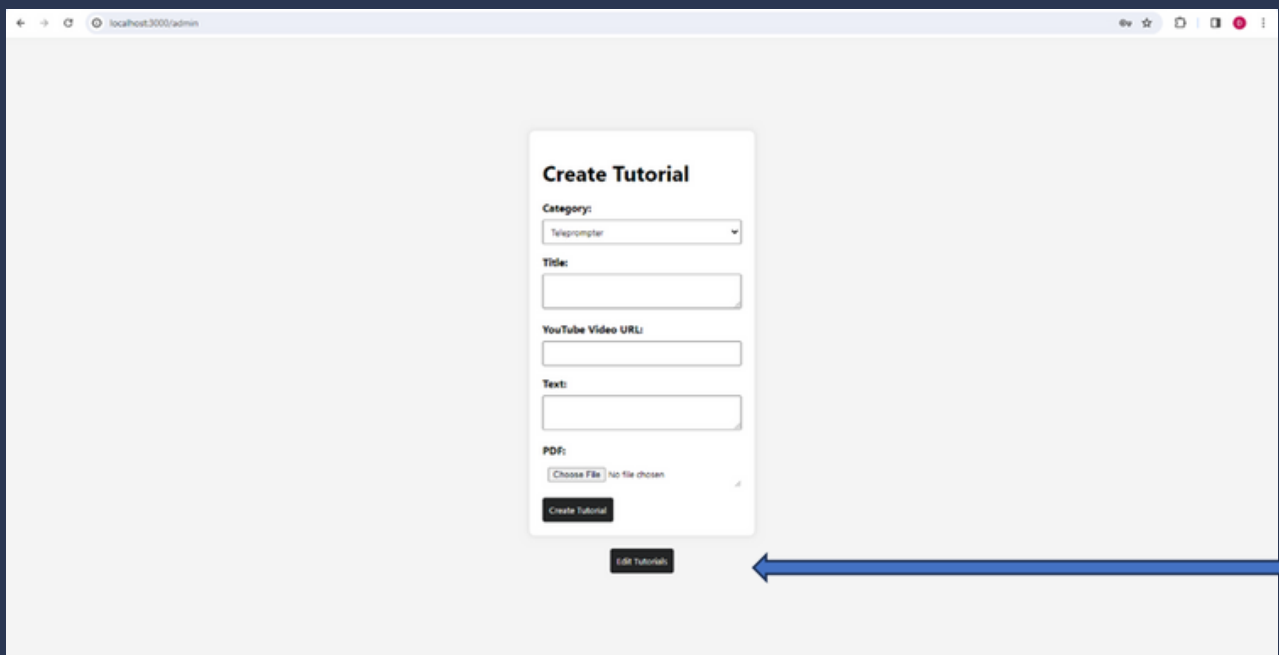
iv.Click 'Create Tutorial'.

USING THE WEBSITE: A STEP-BY-STEP GUIDE

STEP 8

Edit tutorials.

i. On the 'admin page', click 'Edit Tutorials'

ii. To edit a tutorial, click on 'Edit'

USING THE WEBSITE: A STEP-BY-STEP GUIDE

iii. This will open a new page containing the Title and Text of the tutorial. You can make changes to these fields and click 'Save changes'.
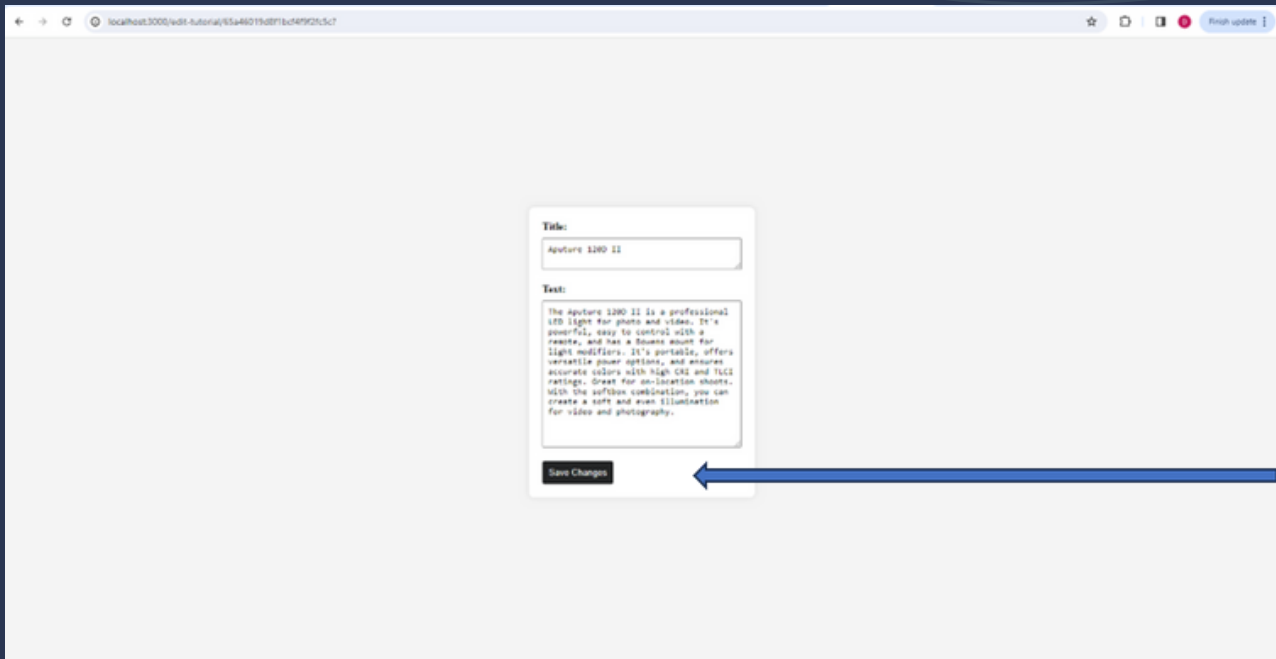
STEP 9

Delete tutorials.

To delete a tutorial, look for the tutorial under the category where you uploaded it and click on 'Delete'

NB: It's not possible to change/edit the Youtube video or PDF after tutorial has been created. To do this, you must delete the tutorial and upload it again.