The process of creating a website using HTML, JavaScript, CSS, and Python Flask is a multifaceted and stimulating experience that requires a great deal of technical prowess and creativity. These technologies are incredibly powerful and versatile, providing me with the ability to craft websites that are dynamic, interactive, and capable of handling complex tasks.

In this in-depth report, we will explore the step-by-step process of me journey in creating a website utilizing these technologies taught by our professor. The first and most crucial step is to plan and design my website, which involves defining your website's purpose, identifying your target audience, and deciding on the content I wish to showcase. Additionally, it is essential to create a wireframe or mockup of my website's layout and design to give you a clear vision of what I hope to achieve.

Conception: with my love for invertebrates, especially beetles. And with the lack of online stores in Hong Kong, I decide to conjure one for the sole purpose of providing a website for customers to visit learn and purchase beetles and related caring products. Not only will they be able to buy products, but also at the same time learn more about the habits and biology of beetles.

Once you have established a clear plan and design, I can begin creating my website using HTML. HTML markup, which is the foundation of any website, consists of tags and attributes that define the structure and layout of your website. To create HTML markup, you can utilize a text editor such as Notepad, Sublime Text, or Visual Studio Code. Begin by creating an HTML file and adding the necessary tags and attributes to structure your content.

Here’s one of the shop blocks in the html

|  |
| --- |
| <h2>Darwin's beetle</h2> |
|  | <p>A strange beetle with slim, saw-like mandibles</p> |
|  | <img src="stactic/product6.jpg" alt="Product 6"> |
|  | <p>Price: $550.00</p> |
|  | <button class="CART">Add to Cart</button> |

CSS is used to style and design my website's layout and appearance. It provides you with the ability to control colours, fonts, sizes, and positioning of my website's elements. To add CSS styling,I created a separate style tag in the HTML file using the <style> tag. Then, use CSS selectors to target specific HTML elements and apply styles to them. The design I have chosen is a black header with a light green section to create contrast and a feel of visiting the jungle.

JavaScript is used to create interactive and dynamic elements on your website. It allows me to add animations, pop-ups, and other interactive elements that make my website more engaging and user-friendly. To add JavaScript interactivity, create a separate JavaScript file and link it to my HTML file using the <script> tag. Then, use JavaScript functions and events to create interactive elements.

Python Flask is a micro web framework that allows me to create web applications using Python. It is lightweight and user-friendly, making it an ideal choice for creating dynamic websites. To create a Python Flask application, I needed to install Flask and create a new Flask application. Then, I can create routes and views that handle requests and responses from my website's users. For example, in the auth.py file, the line of code @auth.route('/login', methods=['GET', 'POST'] ) establishes a connection to the login page, the code I used to define login

|  |
| --- |
| def login(): |
|  | if request.method == 'POST': |
|  | email = request.form.get('email') |
|  | password = request.form.get('password') |
|  |  |
|  | user = User.query.filter\_by(email=email).first() |
|  | if user: |
|  | if check\_password\_hash(user.password, password): |
|  | flash('login successful', category='success') |
|  | login\_user(user, remember=True) |
|  | return redirect(url\_for('views.home')) |
|  | else: |
|  | flash('Password are incorrect, please try again', category='error') |
|  | else: |
|  | flash('This email does not exist', category='error') |

From what i learn from the weeks of lectures, connecting website’s to the Flask application requires me to create a form that allows users to submit data to the server. Such a using the <form> tag in HTML to create a form and the POST method to send data to the server. Then, I would use Python Flask to handle the form data and perform any necessary actions, such as storing the data in a database like the internal database I used in the project called the sqlalchemy, in which I’ve used to store the data of registered users, and using denominators like {endblock} and {block content} to enable the use of python in html. Here is a dB model for our dB

|  |
| --- |
| class User(db.Model, UserMixin): |
|  |  |
|  | id = db.Column(db.Integer, primary\_key=True) |
|  |  |
|  | email = db.Column(db.String(150), unique=True) |
|  |  |
|  | password = db.Column(db.String(150)) |
|  |  |
|  | firstName = db.Column(db.String(150)) |
|  |  |
|  | notes = db.relationship('Note') |

Before launching your website, it is essential to test it thoroughly to ensure that it is functioning correctly. Test all links, forms, and other features to ensure that they are working correctly. should also test your website's load time and mobile-friendliness. Once I have tested the website, it is time to launch it. Launching your website involves publishing it to the internet and making it accessible to my audience. With this I use the virtual machine Ubuntu to run my program.

Creating a website is not a one-time task; it requires ongoing maintenance and updates to keep it running smoothly. making regularly update my website's content, fix any broken links, and ensure that it is secure. Furthermore, I would monitor my website's performance, such as its traffic, bounce rate, and conversion rate, to determine how well it is performing. Use analytics tools like Google Analytics to track your website's performance and make data-driven decisions to improve it in the future.

In conclusion, creating a website using HTML, JavaScript, CSS, and Python Flask is a complex process that requires a great deal of technical knowledge and creativity. By following the steps outlined in this report, i was able to can create a website that is visually appealing, easy to navigate, and engaging for my audience. It taught me that whether I’m building a personal blog, e-commerce site, or portfolio, creating a website using these technologies is an essential step in establishing my online presence and reaching other beetle loving fellows.