**E-COMMERCE APPLICATION ON IBM CLOUD FOUNDRY**



**SUBMITTED BY**

SANDIP MONDAL

SRINITHESH

SURYA

SADURDEVA

**PHASE 3 SUBMISSION DOCUMENT**

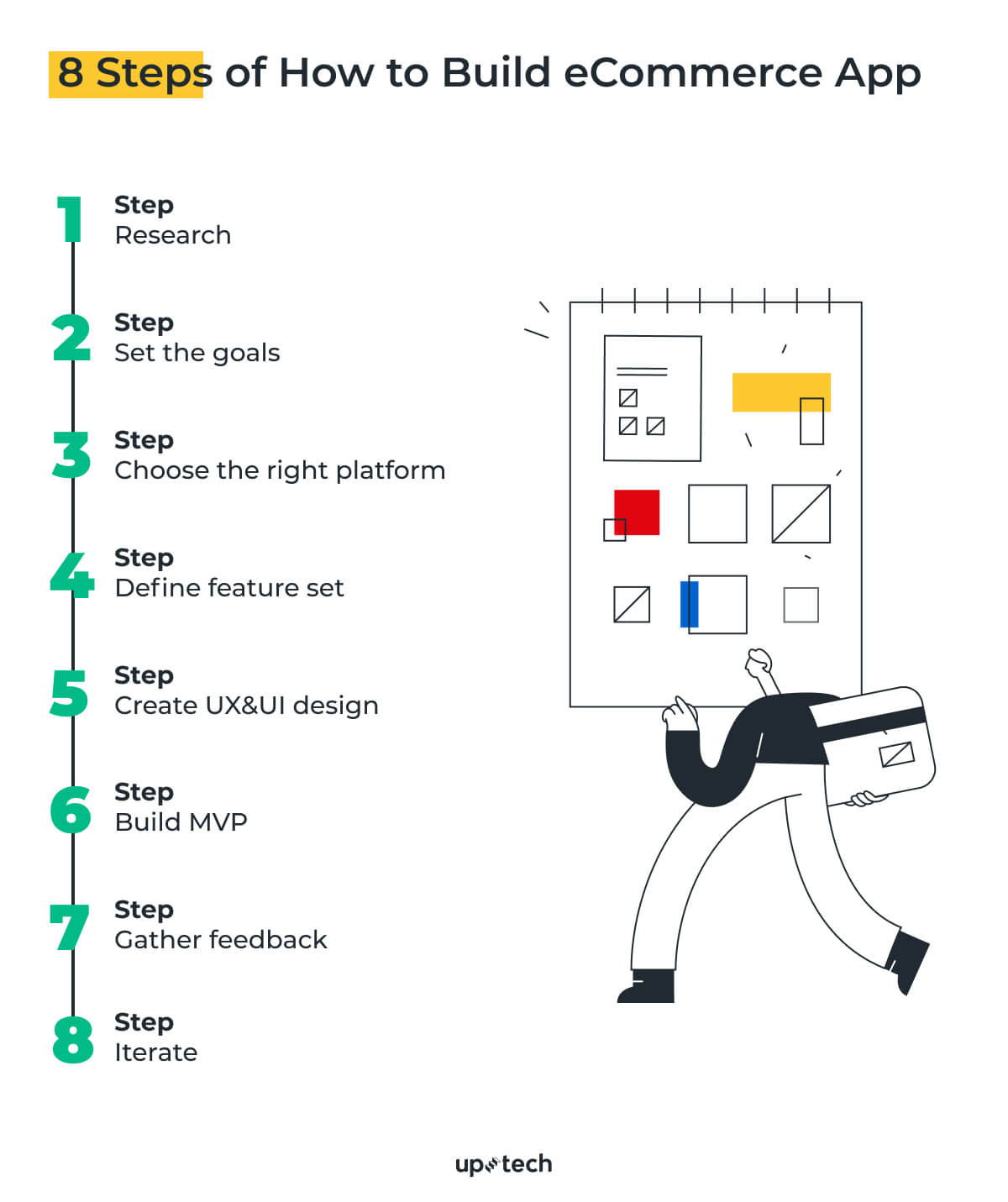
**Phase 3:** Development part1

**Topic:** Design the platform layout and create a database to store product information.

**Problem Statement**:

In this part you will need to understand the problem statement and create a document on what have you understood and how will you proceed ahead with solving the problem. Please think on a design and present in form of a document.

**Steps to create E-Commerce Application**

****

**Design e-commerce platform layout and create database to store product information coding**

1. **Planning and Design**:

* Start by creating wireframes or mockups of your e-commerce platform's layout. Tools like Adobe XD, Sketch, Figma , or even simple pen and paper can be used for this.
* Plan the user interface (UI) design, including navigation, product listings, product pages, shopping cart, and checkout process.

1. **Front-End Development:**

* Develop the front-end using web technologies such as HTML, CSS, and JavaScript. You can use popular frameworks like React, Angular, or Vue.js for this purpose.
* Implement the layout and user interface design in your front-end code.

**3. Back-End Development:**

* Choose a programming language and framework for your back end. Common choices include Python with Django, Ruby on Rails, or Node.js with Express.
* Create an API for your front end to communicate with the back end.
* Implement user authentication, including registration and login.

**4.Database Design:**

* Choose a database system, such as MySQL, PostgreSQL, or NoSQL databases like MongoDB, based on your requirements.
* Design the database schema to store product information, user data, orders, and other necessary information.

**5. Database Implementation:**

* Write SQL or NoSQL queries to create tables and define relationships.
* Populate the database with sample data for testing and development.

**6. Product Information:**

* Create a table to store product details, including attributes like name, description, price, stock, and images.
* Implement an admin panel or a data entry system to add, update, and delete products.

**7. Connect Front-End to Back-End:**

* Use API calls to retrieve and display product information on your e-commerce platform.
* Implement shopping cart functionality to allow users to add and manage products.

**8. Security and Payment Processing:**

* Implement secure payment processing using a payment gateway such as Stripe or PayPal.
* Ensure that your platform follows security best practices to protect user data.

**9.Testing:**

* Test your e-commerce platform thoroughly, including functionality, user experience, and security.
* Fix any bugs or issues that arise during testing.

**10. Deployment:**

* Choose a hosting environment for your application, such as AWS, Heroku, or a dedicated server.
* Deploy your front-end and back-end code, along with the database.

**11. Monitoring and Maintenance:**

* Set up monitoring tools to track the performance of your platform.
* Plan for regular maintenance and updates to keep your platform secure and up to date.

**Create a New Cloud Foundry Application:**

* Use the IBM Cloud CLI to create a new Cloud Foundry application.
* This will allocate the necessary resources for your app.

CODE: **ibmcloud cf create-service**

**Set Up Your Development Environment:**

* Create a new project directory on your local machine.
* Initialize a version control system (e.g., Git) to track your code changes.

**Code Your E-commerce Application:**

* Write the code for your e-commerce application using your chosen programming language and framework.
* Implement features such as product listings, product detail pages, shopping cart functionality, and user authentication.

**Database Integration:**

* Connect your application to a database service on IBM Cloud, such as IBM Db2 or a NoSQL database like Cloudant.
* Configure your application to use this database for storing product information, user data, and order history.

**Payment Integration:**

* Integrate a payment gateway (e.g., Stripe or PayPal) to handle online transactions securely.

**Testing:**

* Thoroughly test your application, including both functionality and security.
* Ensure that user data is handled securely and that payment transactions work as expected.

**Deploy to IBM Cloud Foundry:**

* Use the IBM Cloud CLI to deploy your application to the IBM Cloud Foundry.

CODE : ibmcloud cf push <your-app-name>

**Configure Environment Variables:**

* Set environment variables for your application, including any secrets or configuration values.

CODE : ibmcloud cf set-env <your-app-name> ENV\_VARIABLE\_NAME ENV\_VARIABLE\_VALUE

ibmcloud cf restage <your-app-name>

**Monitoring and Scaling:**

* Set up monitoring tools to track the performance of your application.
* Configure auto-scaling rules to handle traffic fluctuations.

**Monitoring and Scaling:**

* Set up monitoring tools to track the performance of your application.
* Configure auto-scaling rules to handle traffic fluctuations.

**Secure Your Application:**

* Implement security best practices, such as using HTTPS, securing APIs, and protecting user data.