|  |  |  |
| --- | --- | --- |
| **Git Hub:** <https://github.com/nilesh507> | **Nilesh Bhoi** | [nilesh507@gmail.co](mailto:nilesh507@gmail.com)m |
| **LinkedIn:** [www.linkedin.com/in/nileshbhoi/](http://www.linkedin.com/in/nileshbhoi/) | | 315 4365644 |

**Education**

|  |  |  |
| --- | --- | --- |
| Syracuse University | **Master of Science** **Computer Science** | May 2024 |
| Indraprastha University | **Bachelor of Science in** **Information Technology** | August 2021 |

**Skills**

**Programming Languages:** Python, Java, JavaScript, Haskell, C/C++.

**Programming Skills:** Bootstrap, jQuery, Ajax, Node.js, Express.js, SQL, Oracle, MongoDB, Layouts & Partials, Passport(JWT, oauth2), CSS, SASS, Multer, NodeMailer, Socket.IO, React, Firebase, Redux, NumPy, Pandas, Matplotlib, RESTapi.

**Tools:** Linux, Eclipse, IntelliJ, VS Code, Git, WebStorm, PyCharm, Postman, Robo 3T, Firebase, Jupyter notebook, Spyder.

**Experience**

**Teaching Assistant,** Coding Ninjas, Delhi, IndiaMarch 2020 - July 2020 **(Java)**

* Acted as a mentor for individuals across India, regularly monitoring student performance.
* Providing assistance with any questions or doubts related to Data Structures and Algorithms.
* Total number of Doubts taken: 600 plus with rating: 4.7/5

**Coursework and related Projects**

* [**Minor Project: Bitcoin Price P**](https://ijtre.com/wp-content/uploads/2021/10/2020080320.pdf)**rediction (Python)**
* Employed deep learning techniques to predict the price of cryptocurrencies-(Bitcoin) to gain insights into future market trends.
* Analyzed the time series of bitcoin prices using advanced techniques such as Autoregressive-Moving-Average, Autoregressive Integrated-Moving Average (**ARIMA**), and Recurrent Neural Network (**RNN**) models to gain insights into price trends.
* [**Social Media Website**](https://github.com/nilesh507/Nilesh_Bhoi) **(JavaScript)**
* Developed a fully responsive website utilizing the Model-View-Controller **(MVC) architecture** pattern.
* Featuring three layers of security checks implemented using **PassportJS** for enhanced protection.
* Pages are dynamically created (using layouts and views) and the Website's API is secured with **JWT Authentication**.
* Enabled users to share photos with captions and added features such as commenting and following capabilities.
* Enabled users to make friends with other users through a ChatBox-chat feature, which was implemented using the **Socket.io**.
* [**Major Project: Steganography**](https://ijtre.com/wp-content/uploads/2021/09/2021081105.pdf)  **(Java)**
* Utilized a secure approach using spatial domain image steganography to encrypt a secret message within an image.
* Implemented a **Least Significant Bit (LSB)** algorithm for encryption within the application, utilizing Java programming language.
* Combined the cover image and encryption information into a stream of bits using BitStream, and vice versa for decryption.
* Utilized a **lossless compression technique** to encrypt and decrypt the cover image by breaking it up into individual bit-planes and embedding different levels of information within each plane.
* [**Movie-react-app**](https://github.com/nilesh507/Movie-App) **(React)**
* Developed a movie review application that allows users to search for and discover new films
* With add and remove functionality added to them in a personal favorites list.
* Utilized **RESTFUL API**-OMDB to retrieve movie information and data for the movies.
* [**Cart-react-app**](https://github.com/nilesh507/Cart) **(React)**
* Designed a cart module with randomized items with **CRUD functionality** appended to them on the **firebase database**.
* Implemented component update lifecycle to efficiently update the state of items within the app, which is hosted on Firebase.
* **OPERATING SYSTEM (C/C++)**
  + Designed the system to read and process multiple user programs and schedule them using different scheduling methods.
  + Implemented the ability to make **Unix system calls** such as read, write, yield, and exit within the application.
  + Emulated RAM functionality by implementing virtual address to physical address translation management in NACHOS.
* Built a ready queue and scheduler effectuated with the timer interrupt service routine in a **multi-programming** environment.
* Designed multiple scheduling algorithms for processes/threads with variable time quantum for performance comparison.
* **Structure Programming And Formal Methods (Haskell)**
* Built a property testing model for random/automatic **property testing** for the Haskell Programs.
* Plot a deep **automated theorem proofer** with the goal to express access control policies and their ramifications.
* Utilized an access control matrix to determine the accessibility of requests made and set appropriate access controls.
* Bell-La Padula security & Biba Integrity model were used to determine the grant of requests to ensure secure access controls.

**Training**

* [**Advance Web Development with React**](https://ninjasfiles.s3.amazonaws.com/certificate1039637c17f87f8e8eb78efd110912dbc4d4090.pdf)[4 months]
* [**Full Stack Front Web development with Node.js**](https://ninjasfiles.s3.amazonaws.com/certificate719289a0041f840f982eee2e76cd4b0ea95902.pdf)[4 months]
* [**Front End Web Development with Node.js**](https://ninjasfiles.s3.amazonaws.com/certificate719290dba9980b22b51305f2f3665a6706d5bb.pdf)[4 months]
* [**Java Foundation with Data Structures & Algorithms**](https://ninjasfiles.s3.amazonaws.com/certificate37598043dc02253543b43c307491d151258de8.pdf)[3 months]