Jamie Javis

CS - 405

9/29/24

Enhancement Two: Algorithms and Data Structure

1. Briefly describe the artifact. What is it? When was it created?

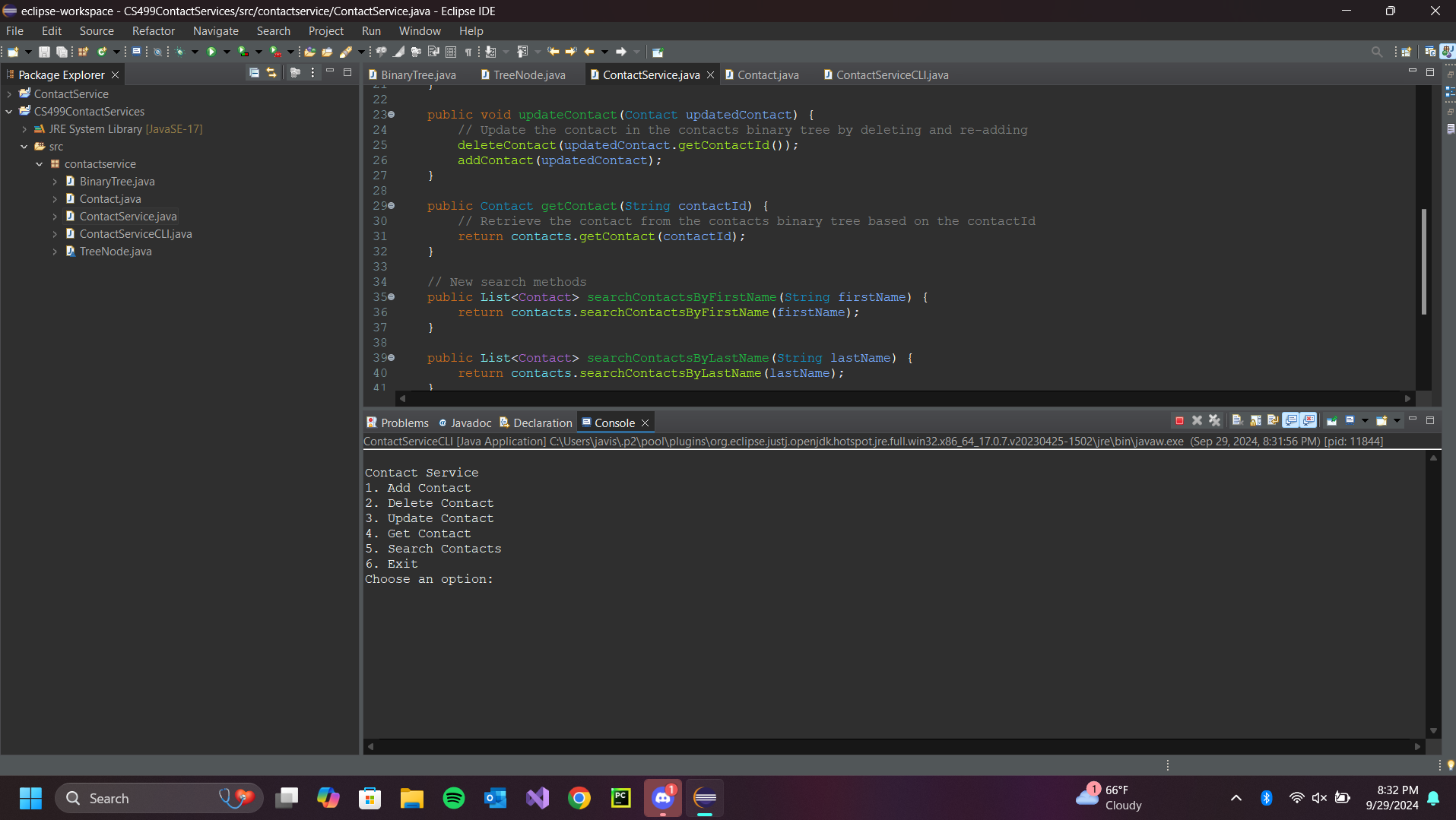
For Artifact Two, I used an artifact from CS320 that created a way to store contacts. The contact service uses in-memory data structures to support storing contacts (no database required). In addition, there is no user interface for this milestone project. The current version of this assignment used a Hashmap to store data for the contacts. This assignment was created over a year ago and needed an upgrade.

2. Justify the inclusion of the artifact in your ePortfolio. Why did you select this item? What specific components of the artifact showcase your skills and abilities in algorithms and data structure? How was the artifact improved?

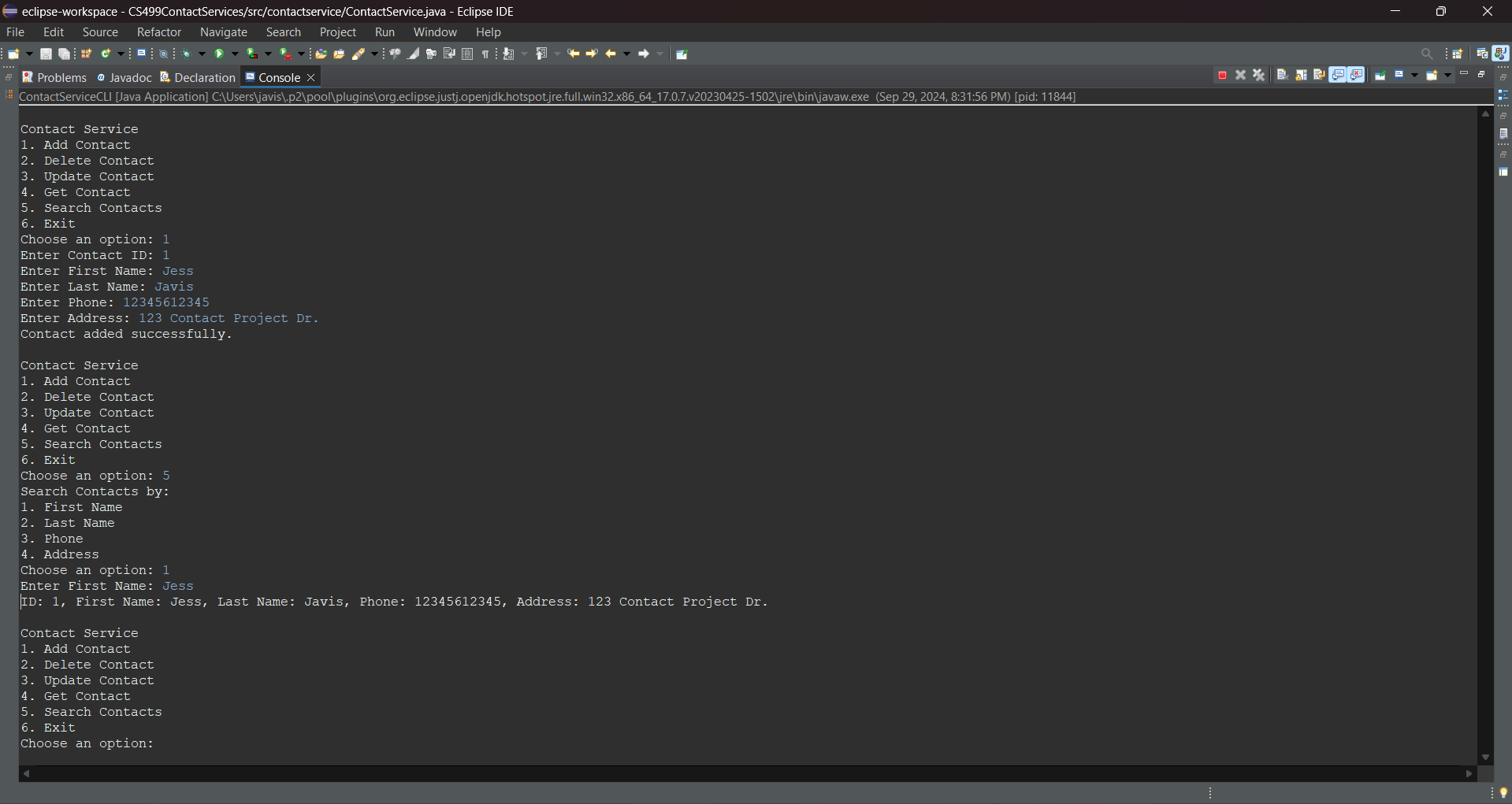
I converted the current data structure (hash map) into an AVL Tree in java. The tree is more compact and has a more valuable way of ordering data. maintaining a sorted order for the elements in an entry, perform ordered operations, and still require relatively efficient access, insertion, and deletion. By also updating this artifact, a UI element was added to help interact with the application. I was able to show my skills in problem-solving, analytical skills, design skills, and optimization skills. I was able to create a tree to optimize the space usage as well as a more secure way to organize the data input using an array and CLI. I was able to save on overhead memory while also making the implementation more complex for myself due to the tree set up. By designing this tree successfully, I showed my ability to utilize a more complex yet helpful way to store the data that could then be used in the future for larger projects. I was able to show my skills in optimization by converting the HashMap to a tree to optimize the allocated memory for the device used.

3. Did you meet the course outcomes you planned to meet with this enhancement in Module One? Do you have any updates to your outcome-coverage plans?

I did meet the planned enhancement for this course outcome.



Above is an image of the terminal that allows users to interact with the system.



Above is an image showing the success of adding a contact and being able to retrieve the contact.

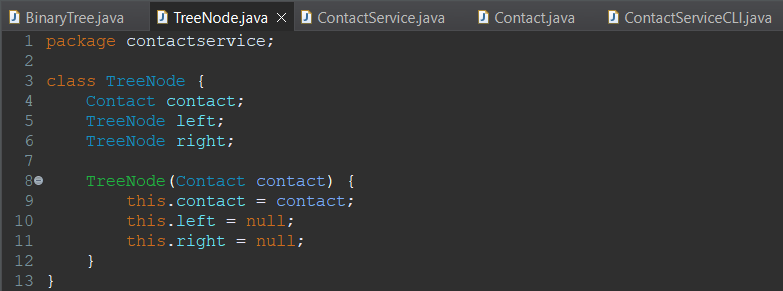


This image shows the successful deletion of the contact added

4. Reflect on the process of enhancing and modifying the artifact. What did you learn as you were creating it and improving it? What challenges did you face?

Implementing a binary tree for contact storage reinforced my understanding of data structures and algorithms. I revisited concepts like tree traversal, node insertion, and deletion, which are foundational in computer science. The process highlighted the importance of encapsulation and modularity. Creating separate classes for BinaryTree, ContactService, and ContactServiceCLI helped maintain a clean and organized codebase. Implementing search methods for different contact attributes (first name, last name, phone, address) required careful thought about how to traverse the tree and collect matching nodes efficiently. Designing the CLI for user interaction made me consider usability aspects, such as clear prompts and handling user input effectively. Ensuring a smooth and intuitive user experience was a key focus. As the project grew in complexity, maintaining code readability and organization became crucial. Ensuring that each class and method had a single responsibility helped manage this complexity. I only had trouble when trying to bring in my original project. The original project was built in a Linux environment, so the setup of the Eclipse IDE was more frustrating and required me to start a new project.

TreeNode:



BinaryTree:

