IDE-IT FrontEnd

Puja Ramanathan, Alyssa Ricketts, Rachel Zigman

Motivation

Importance

- IDEs provide powerful tools for developers
  - Code insight
  - Debugging
  - Resource Management

Problem

- Only a small number of these functionalities get used
- Low discoverability of existing IDE tools and shortcuts

Accessibility

Awareness

Configuration

Current Approaches

MouseFeed

User clicks menu item:

Informs of hotkey shortcuts ✔

Requires user already knows feature exists ✖

Features Trainer

Daily lessons:

Informs of hotkey shortcuts ✔

Irrelevant Interrupts flow ✖

Tips

Tips of the day:

Introduces new features to user ✔

Irrelevant Bothersome ✖
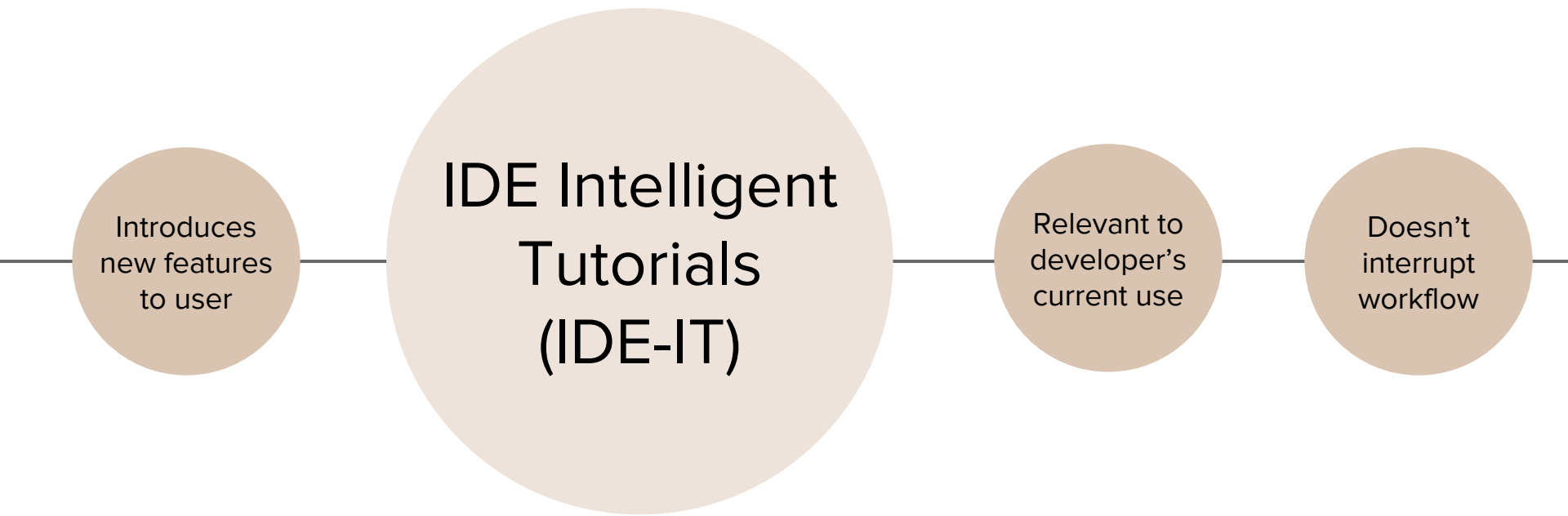
Web Search
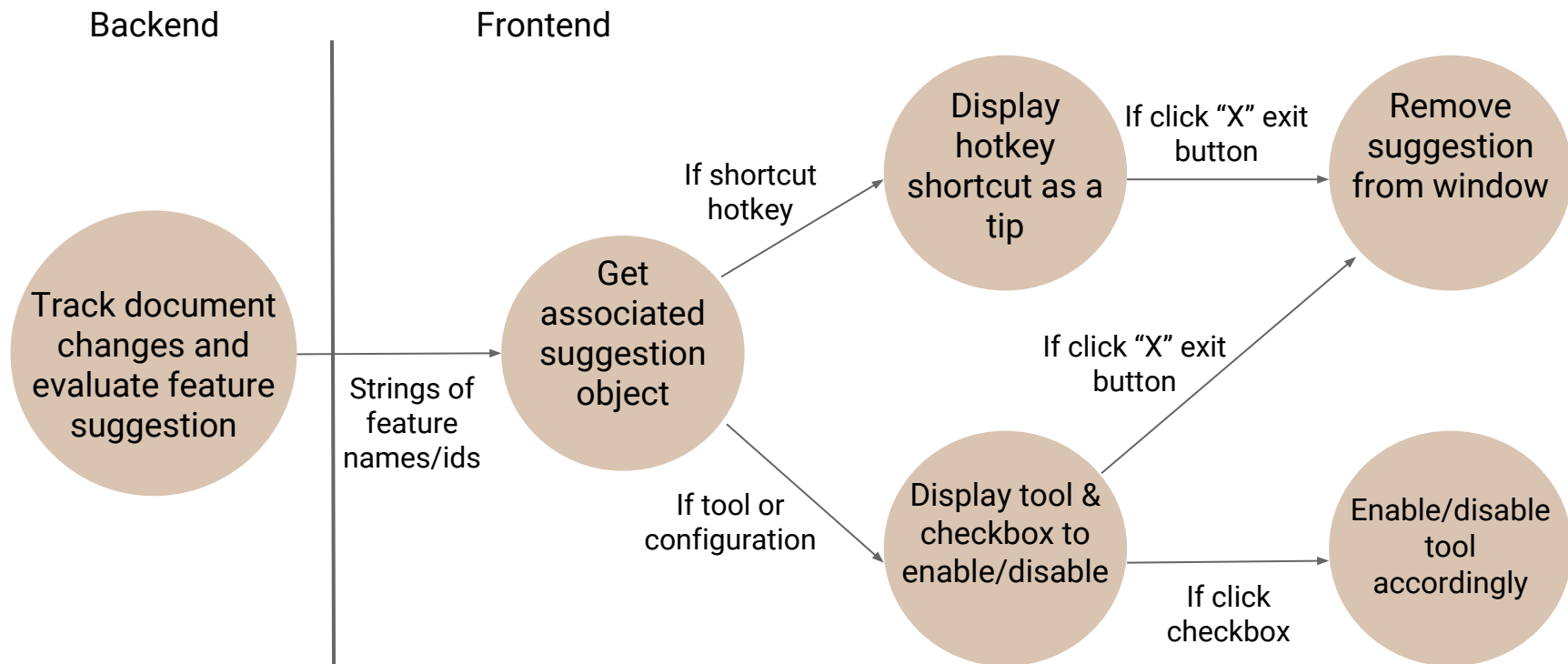
Videos, forums, articles:

Introduces new features to user ✔

User search Interrupts flow ✖

Our Approach

Quick Access

Package Explorer ⌗

- ▶ cse332.datastructures.worklists
- ▶ cse332.exceptions
- ▶ > cse332.interfaces.misc
- ▶ cse332.interfaces.trie
- ▶ cse332.interfaces.worklists
- ▶ cse332.misc
- ▶ cse332.sorts
- ▶ cse332.types
- ▶ > datastructures.dictionaries
- ▶ datastructures.worklists
- ▶ experiment.BSTvsAVL
- ▶ experiment.ChainingHashTable
- ▶ experiment.GeneralPurposeDictionary
- ▶ experiment.HashFunctions
- ▶ p2.clients
- ▶ p2.sorts
- ▶ p2.wordcorrector
- ▶ p2.wordsuggestor
- ▶ p2.writeup
- ▼ tests.gitlab.ckpt1
  - ▶ CircularArrayComparatorTests.java
  - ▶ Ckpt1Tests.java
  - ▶ MoveToFrontListTests.java
  - ▶ NGramToNextChoicesMapTests.java
- ▼ tests.gitlab.ckpt2
  - ▶ AVLTreeTests.java
  - ▶ CircularArrayHashCodeTests.java
  - ▶ Ckpt2Tests.java
  - ▶ HashTableTests.java
  - ▶ HeapSortTests.java
  - ▶ QuickSortTests.java
  - ▶ TopKSortTests.java
- ▶ JRE System Library [Java SE 8 [1.8.0_11]

Tabs: *AVLTree.java ⌗ | MinFourHeap.java | *Dictionary.java | BinarySearchTree.java

```java
1  package datastructures.dictionaries;
2
3  import cse332.datastructures.trees.BinarySearchTree;
4
5  /**
6   * Interface for an AVL tree: a self-balancing Binary Search Tree
7   * where the difference between heights of left and right subtrees
8   * cannot be more than one for all nodes.
9   */
10
11 // @SuppressWarnings("hiding")
12 public class AVLTree<K extends Comparable<K>, V> extends BinarySearchTree<K, \
13     private int size;
14     private int testVal;
15
16     public AVLTree() {
17         super();
18         this.size = 0;
19     }
20
21     /**
22      * Inner class to represent a node in the tree.
23      */
24     public class AVLNode extends BSTNode {
25         public int height; // Height field
26
```

Task List ⌗

Find ▶ All ▶ Activate...

Suggestions:

💡 Try using "CMD + /" to comment several lines ✕

💡 Try using "CTRL + D" to delete a row ✕

☑ Enable compiler warnings for hidden fields ✕

☐ Enable content assist ✕

☑ Enable auto-complete ✕

Problems | Javadoc | Declaration | Console ⌗ | Search | Git Staging

```
<terminated> Ckpt2Tests [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_111.jdk/Contents/Home/bin/
AVLTree                                          ✗ Failed Test(s)
    testTreeWith5Items                             ✓ Passed
    testHugeTree                                   ✗ Failed
    checkStructure                                 ✓ Passed
HashTable                                        ✓ Passed Test(s)
    testHugeHashTable                              ✓ Passed
CircularArrayHashCode                            ✓ Passed Test(s)
    equality                                       ✓ Passed
    ineq1                                          ✓ Passed
    ineq2                                          ✓ Passed
    ineq3                                          ✓ Passed
```

Outline ⌗ | Visualizer | Debug | Saros

- datastructures.dictionaries
- ▼ > AVLTree<K extends Comparable<K>, V>
  - size : int
  - testVal : int
  - AVLTree()
  - ▼ AVLNode
    - height : int
    - AVLNode(K, V)
  - insert(K, V) : V
  - insertHelper(AVLNode, K, V) : AVLNode
  - updateHeight(AVLNode) : void
  - getBalance(AVLNode) : int

Writable | Smart Insert | 17 : 17

Research Questions

Hypotheses

- Suggestions will be non-intrusive and not interrupt workflow
- Users will positively respond to suggestions

Methodology

- Recruit ~10 UW CSE students
- Ask them to perform specific tasks
    - I.e. comment multiple lines
- Survey: Scale 1-10

Information and Metrics

- User response to interface
    - Window placement
    - Clarity of message
    - Distractibility
    - Did they use the suggestion?

Tie to Real-World Problem

- Annoyance of "daily tips"
- Interruption of workflow
- Timing of suggestions
- Accessibility of configuration

Preliminary Results

❖ Learned how to create an initial working Eclipse plug-in

❖ Creating the window and checkboxes was simple to set up

❖ Challenge of enabling or disabling configurations through the code

❖ Understand the relationship between the backend and frontend teams

Related Work

API Explorer

- Extension to **content assist** in Eclipse IDE
- Recommends methods or types
- Users could easily choose relevant API elements from recommendations

Key Promoter X

- Plug-in to learn **IntelliJ shortcuts**
- Works on buttons clicked within the IDE
- For buttons without shortcuts, plug-in prompts user to create one

Thank you!

Questions?