



Team BirdFeedr

Josh Gillham
Thomas Macari
Di Tran

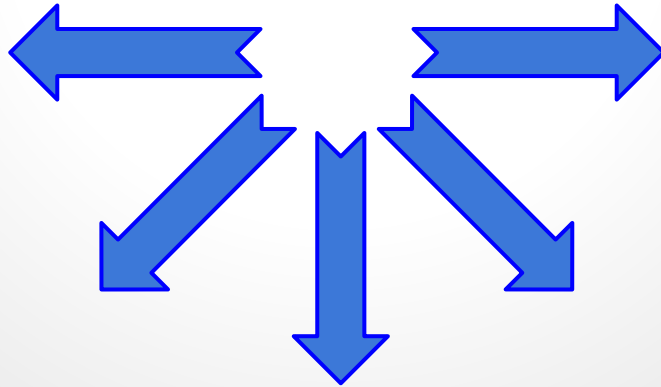
Wei Huang
Miguel Roman
Nate Witt

Process Setting

- Selection to pursue a BlueJ extension
 - Want to contribute to BlueJ community
 - Flexibility - low risk way to explore projects
 - Encouraged freedom and creativity

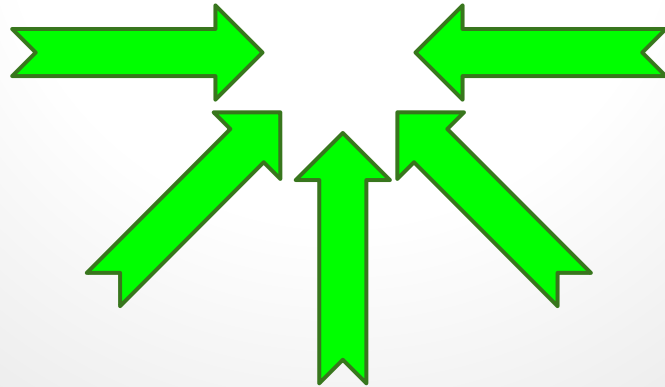
Process Explore BlueJ

- We decide to learn more about the target technology
- Split into groups



Process Regroup

- Placed the exploration projects on the shelf.
- Came back as group and shared what we learned.



Process Brainstorm

- We listed some extension ideas.
 - A chirp when your program compiles successfully.
 - Allow BlueJ to use GitHub as a version control tool



Process Creating a Vision

- We needed a direction so we created a vision to direct our group.



Process Vision Statement

“To inspire people to reach their computer science dreams by providing tools that help them overcome the barriers of programming.”

Process Choosing a Project

- We used our vision statement to weed out our list of old ideas.
- We interviewed students and teachers to develop our final idea.



Process Observations

- Help students learn how to program within BlueJ by providing programming exercises.
- Get instant feedback on those exercises.



Process



BlueJ TA Description

A BlueJ extension that allows students to practice Java problems and receive feedback.



BlueJ TA Features

- Select a practice problem
- See problem description
- Practice Java
- Receive feedback

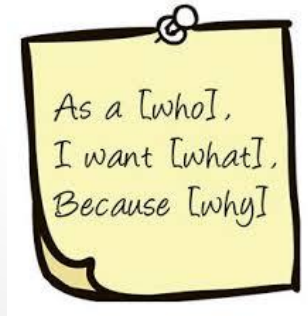


BlueJ TA Parallels with BlueJ

- Uses design elements consistent with BlueJ
 - Uses BlueJ's basic features to accommodate user coding and compilation.
 - Spawns new windows for information and exercises.
 - Extends menu options from BlueJ's menus.

Process User Stories

- We created user stories and they led us to features we need to focus on.



Process Modules

- Split the goal into modules that could be developed iteratively and simultaneously.
 - XML Reading
 - Graphical User Interface
 - Execute JUnit test



Process Integration

- Took components and integrated them together into a BlueJ extension.



Process Refactoring

- Our project was a mess but it worked and proved our idea.
- We needed to clean it, redesign it, and document it.



Process Verification and Validation

- End-to-end testing on Windows and OSX.
- Verified user stories.
- Logged issues.
- Wrapped up project.

Process

Questions?

BlueJ TA Demo

The presentation will pause and we will show you a quick demo.

BlueJ TA On startup

- Creates temp dir in BlueJ's config dir.
- Loads exercises from JAR.
- Creates a BlueJ menu option in "Tools."

BlueJ TA Setting preferences

- BlueJ sets and remembers extension directory in its preferences folder

BlueJ TA Picking an exercise

- Button within BlueJ's tools menu.
- Clicking launches a JavaFX GUI.
- GUI lists exercises from set directory.
- GUI pulls exercise summaries from XML files.

BlueJ TA Launching an exercise

- Creates a new BlueJ project.
- Locks the UI
- Populates the project with classes from the XML, including an JUnit class.
- Compiles the project.
- Transfers the JUnit class out of the project.
- Unlocks the UI.

BlueJ TA Exercise description

- Right clicking an exercise class pops up a button to view the description.
- Clicking this button opens a new window.
- This window displays relevant information about the exercise, like its description and desired output.

BlueJ TA Execution of an exercise

- Locks the UI.
- Transfers execution classes into the project
- Runs a JUnit test with these classes.
- Removes the execution artifacts.
- Unlocks UI.
- Passes results on to GUI.

BlueJ TA Canceling an execution

- Test are run using BlueJ's user VM
- Canceling a test is the same as canceling other code running in BlueJ.
- By resetting the users java VM.
- When this is done the test GUI is updated with a canceled message.

Team BirdFeedr



“To inspire people to reach their computer science dreams by providing tools that help them overcome the barriers of programming.”