

StuView

Michael Peterson
Sergio Ramirez
Gavin Rouse
Corbin Staaben

Online Student Practice Interview System

- Allow students to practice technical and non-technical interviews
 - More convenient for all parties involved
 - Audio, video, and computer screen are recorded during the interview
 - Interview can be reviewed by professor or industry professional
 - Feedback from reviewer will be available to student
-

Two Types of Interviews

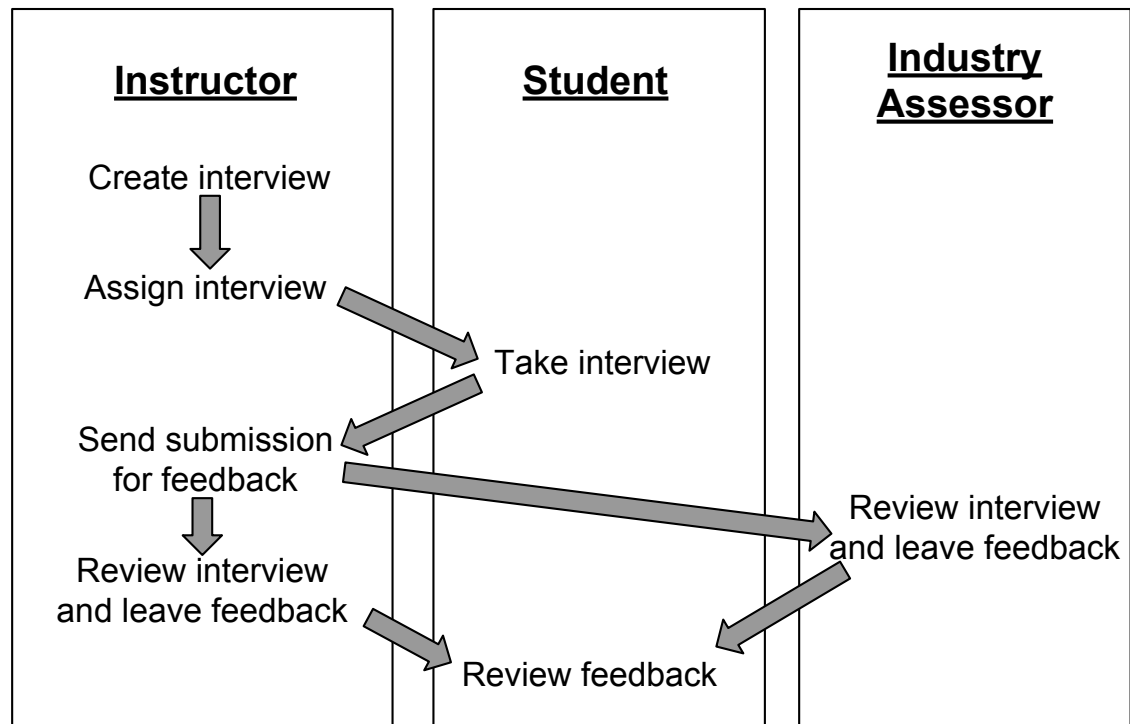
Practice Interview

- Students can take these as many times as they would like to get practice without review
- Only viewable to the student
- Available for download after interview is finished

Reviewed Interview

- Can only be taken once
 - After finishing, the interview is uploaded to vimeo and can be sent for review
 - Available for download by student after interview is finished
-

StuView Flow



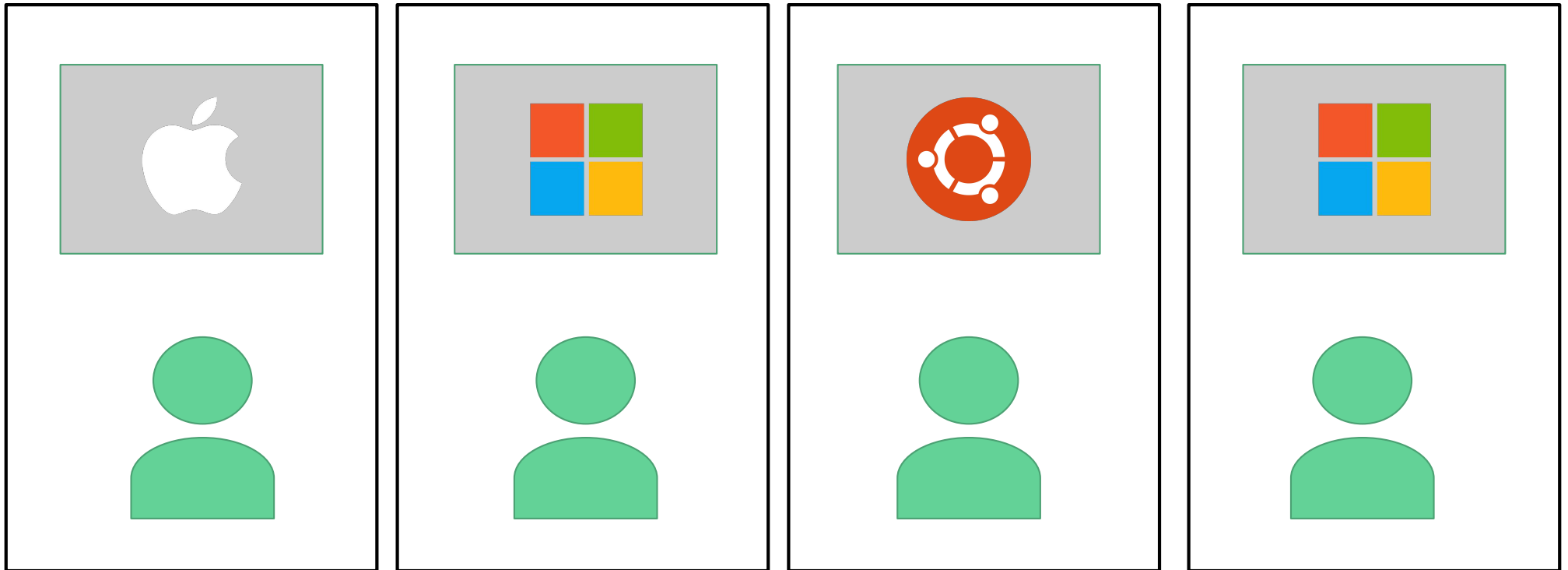
Where We Started

- Basic login functionality
 - User data stored in the Web page
 - No user validation or authentication
 - No interview functionality
 - Canvas and audio recording libraries incorrectly implemented
-

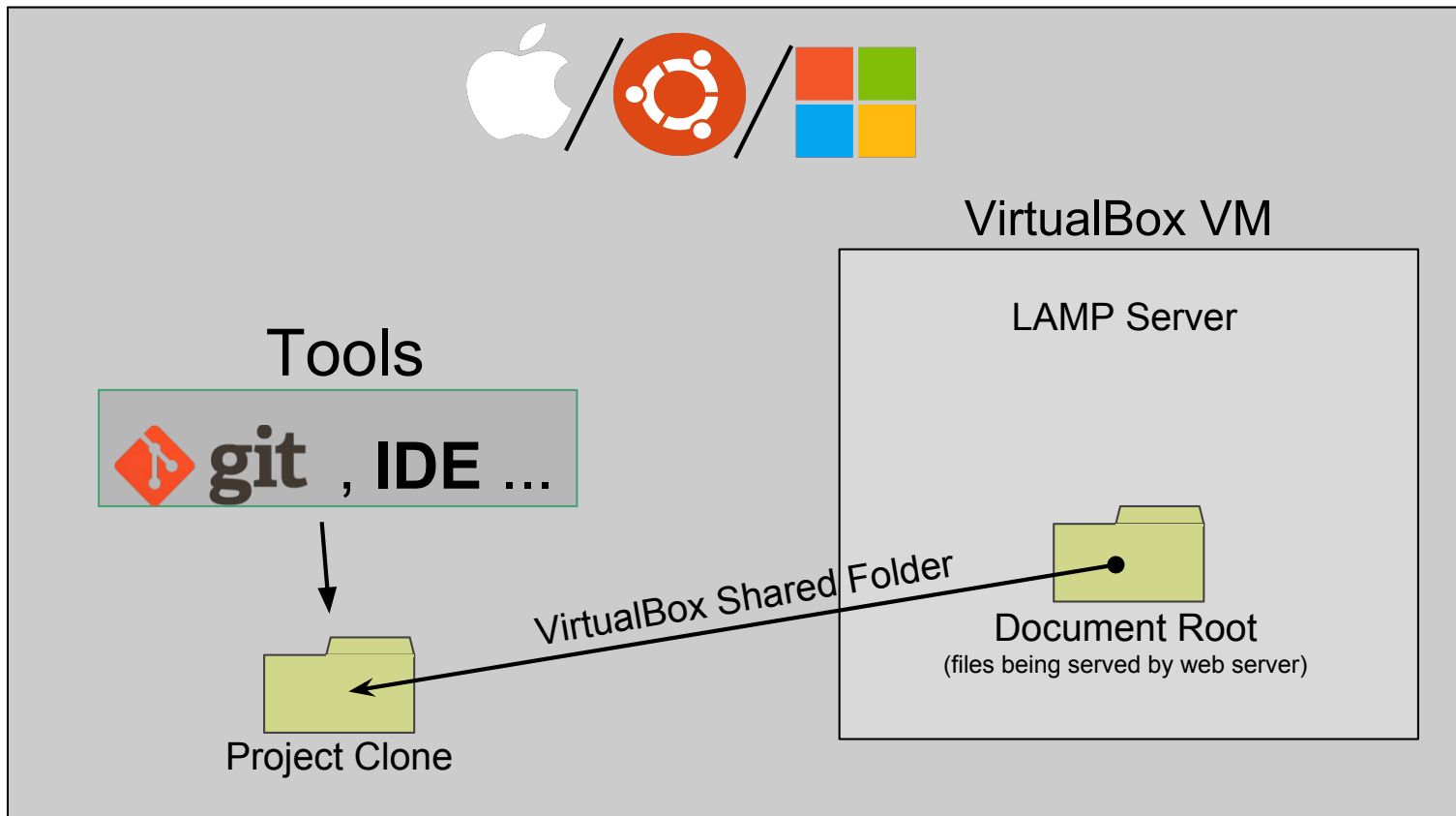
Our Goals

- Get the project running on all of our machines.
 - Identify what was broken and fix it.
 - Add Session management.
 - Get the interview recording/uploading working.
 - Make pages more intuitive.
-

Development Environment



Development Machine



Session Management

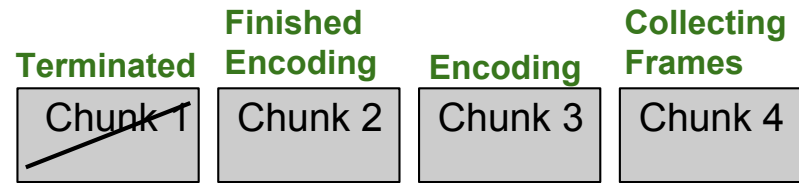
- User data stored in PHP sessions
 - Removed from client's browser and visibility
 - All database endpoints authenticate user's session
 - Randomly generated unique ID used to verify session
 - Manual testing
-

Record Interview

- Interviews are recorded as students take them
 - This includes recording input from the microphone as well as recording what's happening on the screen.
- In order to support full length interviews (over five minutes), the recording has to be split up into chunks. We used Web Workers (threads* in the browser) for taking care of the separate chunks.

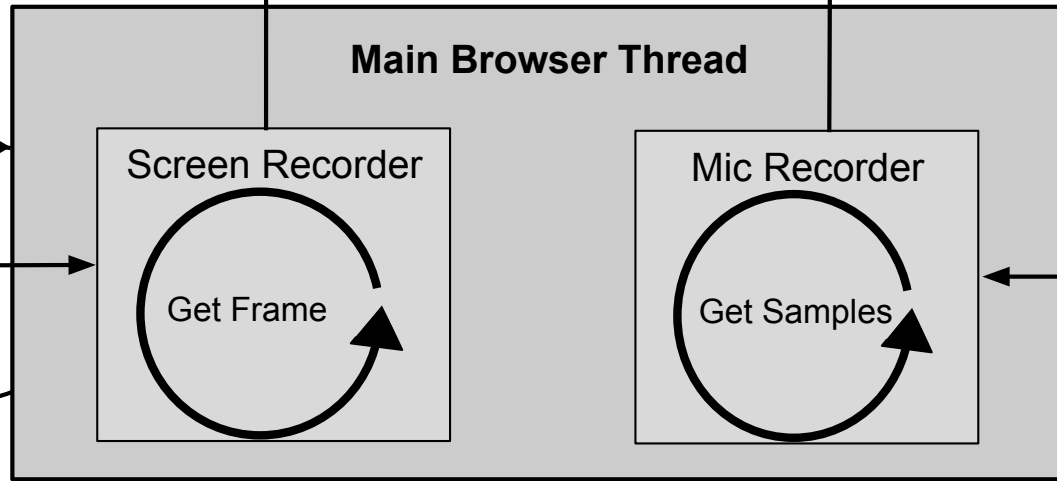
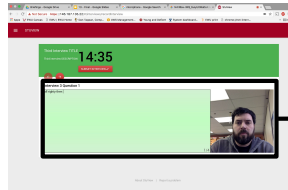
**A thread is a way of running different pieces of code at the same time*

Video Encoder Web Workers



Save Frame

Video Chunk

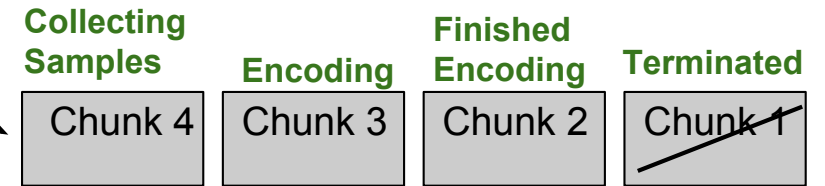


Video Chunk

Store Chunk

Web Server

Audio Encoder Web Workers



Save Samples

Audio Chunk



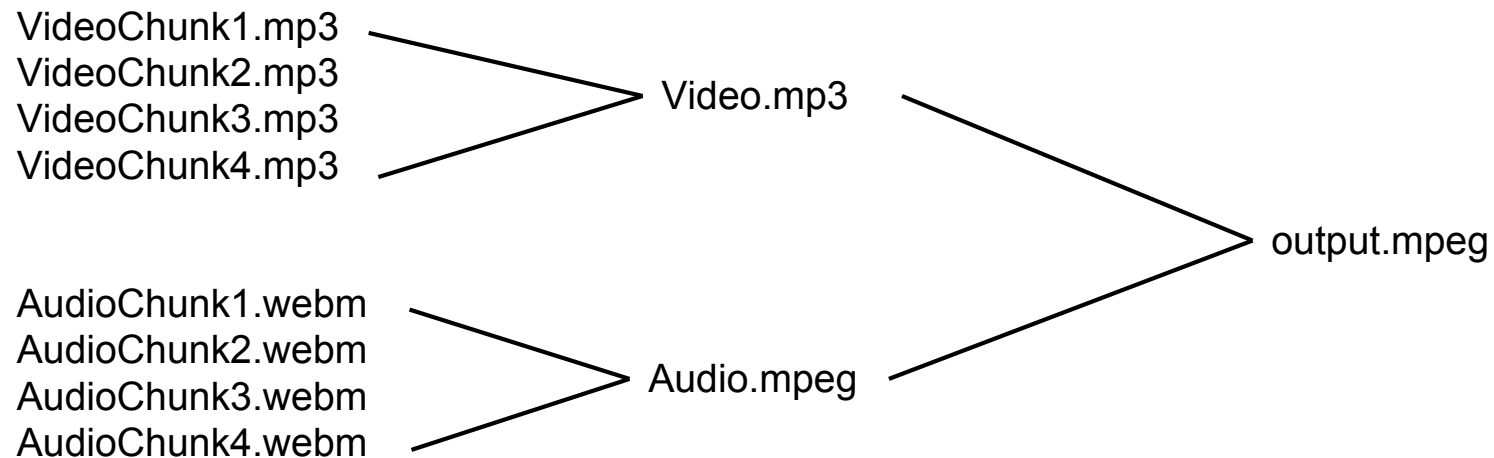
Audio Chunk

“front end”

“back end”

Putting the pieces together

The server uses ffmpeg to combine all of the chunks into a final video file



Video uploads



vs



Youtube

PROS

- Free, and unlimited space.

CONS

- Approximately 300 video uploads per day.
-

Vimeo

PROS

- Can request a much higher request limit.

CONS

- Paid service, and limited space (more space available for more money).
-

Vimeo

- Vimeo was the best choice to be able to handle 7k requests per day at a peak day.
- Even though Vimeo is a paid service it's the best choice for scalability.



Video Uploads

- We would still fall short in the number of requests we would need.
 - So we decided to store the video's temporarily in the server until requests and space become available.
 - From there it depends on the space available in the server.
-

Testing

- Postman was used for quick Vimeo API endpoint testing.
 - Programmatically created hundreds of videos to be able to test the code under high demand conditions.
 - Terminal was used primarily for testing, since this part of the code is only executed on the server side.
 - Tracking errors by logging them for future reference.
-

Video Uploads













- The code is fired up when a user submits an interview.
 - The code looks in our media folder for any videos, not just the one that was just created.
 - A sublist of videos is created based on the available requests/space in Vimeo.
 - The video uploading was written with nodejs, which is a way to write server side code using javascript.
-





















Instructor side

- Four pages on the instructor side
 - Create Interview
 - Manage Interviews
 - Interview Feedback
 - Manage Groups
 - Most of the work done on the instructor side was on the Interview Feedback page
 - The Manage Groups and Manage Interviews pages also required some work
-

Manage Groups

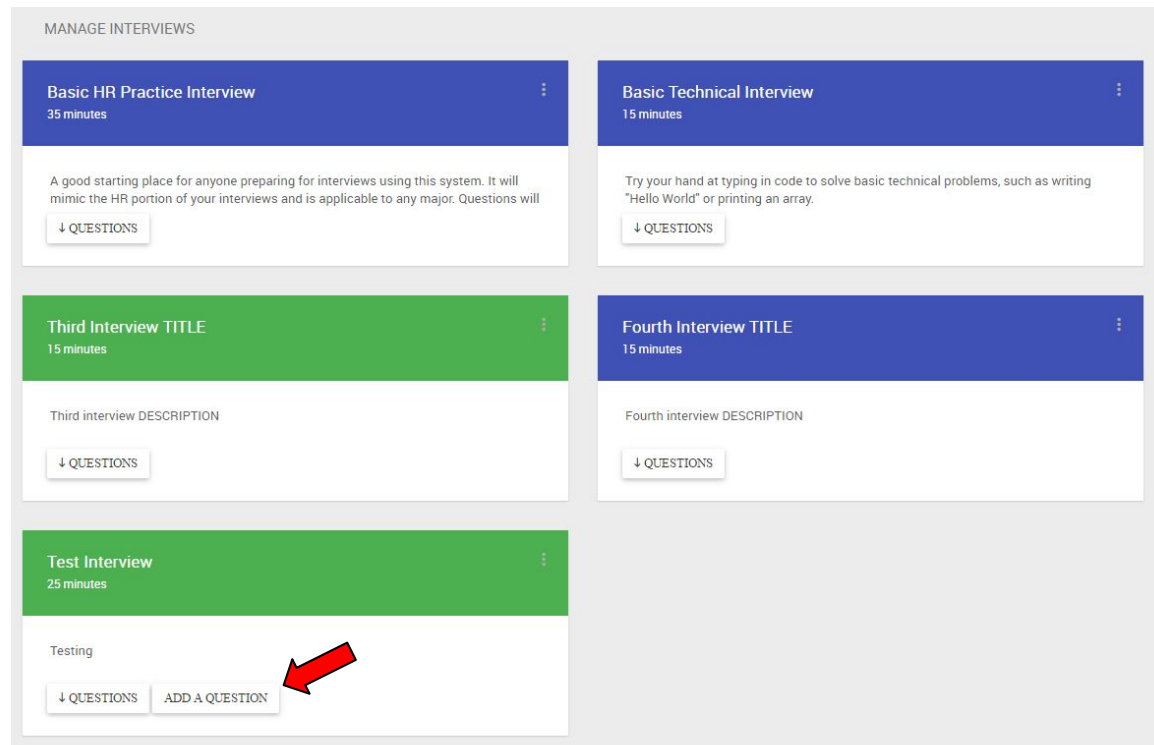
- Added ability to remove students from groups
- Added ability to remove groups

GROUPS				
Manage Groups				
New Group		ADD GROUP		
NAME	EDIT	ADD USERS	ADD	CURRENT USERS
test group		-- select a student --		Michael Peterson
test group 2		-- select a student --		Nicholas Witmer Michael Peterson
test group 3		-- select a student --		Shane Olson
test group 4		-- select a student --		
Senior Project		-- select a student --		Nicholas Witmer Michael Peterson Shane Olson Jake Flynn James Reisenauer
All Students		-- select a student --		Nicholas Witmer Gavin Rouse Michael Peterson Shane Olson Jake Flynn James Reisenauer

GROUPS						
Manage Groups						
New Group				ADD GROUP		
NAME	EDIT	REMOVE	ADD/REMOVE USERS	ADD	REMOVE	CURRENT USERS
Senior Project			-- select a student --			Nicholas Witmer, Shane Olson, Jake Flynn, James Reisenauer
test group			-- select a student --			
test group 2			-- select a student --			
test group 3			-- select a student --			
test group 4			-- select a student --			

Manage Interviews

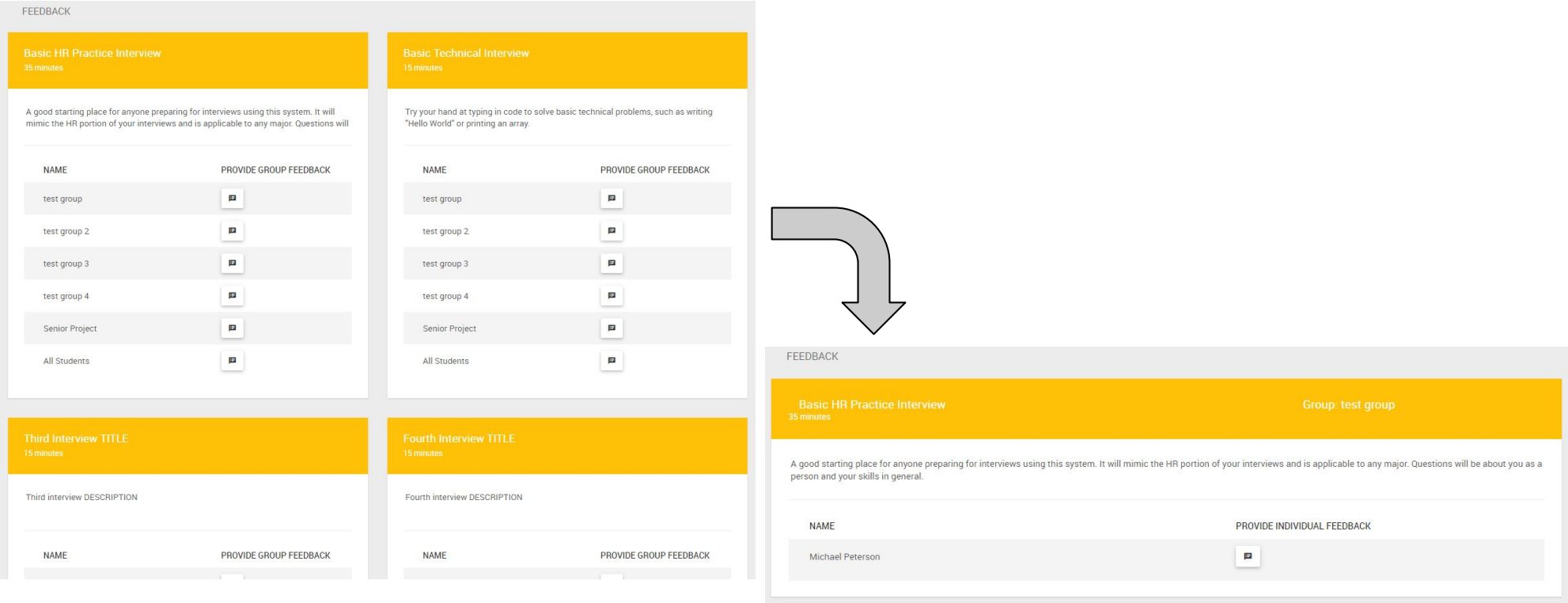
- Added ability to add questions after the initial interview creation
- Only adds questions to the end, they cannot be rearranged currently



Interview Feedback

- The process for choosing and sending submissions needed to be reworked
 - Added a table with all student interview submissions that is filterable and sortable
 - Instructors can select an interview submission and continue to a second page
 - The second page asks for an email to send a review request to
-

Interview Feedback - Before



Interview Feedback - After

FEEDBACK

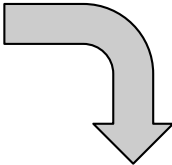
Interview Submissions

Q

Search

Student Name	Interview Title	Submission Date	Groups	Sent
Michael Peterson	Fourth Interview TITLE		test group, test group 2, Senior Project	✓
Nicholas Witmer	Third Interview TITLE		test group 2, Senior Project	✓
Nicholas Witmer	Third Interview TITLE		test group 2, Senior Project	✓
Nicholas Witmer	Third Interview TITLE		test group 2, Senior Project	✓

SHARE SUBMISSION



FEEDBACK

Share Submission

Student

Michael Peterson

Interview

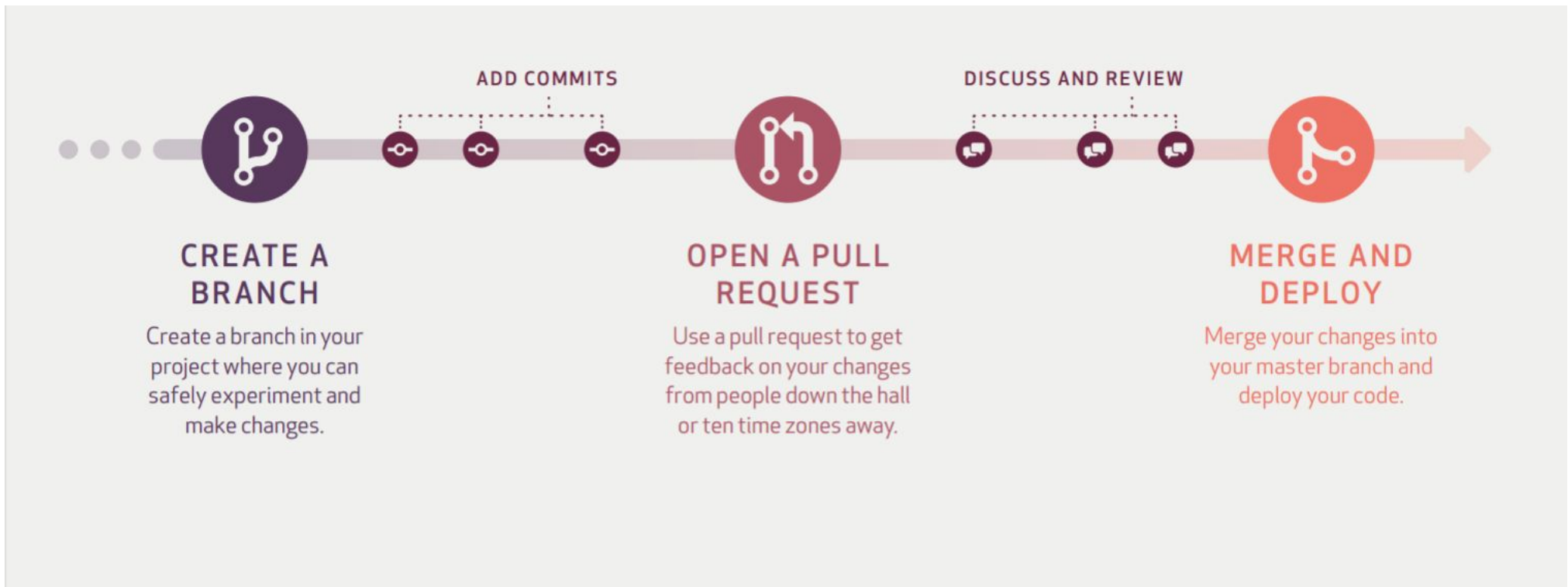
Fourth Interview TITLE

Assessor's Email

industryassessor123@company.com

SEND SUBMISSION

Workflow in GitHub



Source: <https://guides.github.com/introduction/flow/>

Sprints in JIRA

The screenshot displays the JIRA web interface for a sprint board. The top navigation bar includes the JIRA logo, a menu icon, and links to Dashboards, Projects, Issues, Boards, and a Create button. A search bar and user profile icon are on the right. The left sidebar contains navigation options: Backlog, Active sprints (selected), Releases, Reports, Issues, and Components. Below these are sections for Project Shortcuts (github.com, Add shortcut) and Source Code (Add source code). The main area is titled 'STUV Sprint 23' and shows a Kanban board with four columns: To Do, In Progress, In Review (Pull Request Pending), and Done. The board is organized into sections: Learning (4 issues), Implement Interview Video Upload (2 issues), Student Interview Front End Functionality (1 issue), and Instructor Functionality (1 issue). Three issues are visible: STUV-165 (Add video url to database), STUV-160 (Get full length videos working), and STUV-143 (Get the interview question flow working correctly). The right sidebar shows details for issue STUV-143, including its status (IN PROGRESS), priority (Medium), component (None), and assignee (Michael Peterson, Administrator). It also lists the reporter (Michael Peterson, Administrator) and the creation/update dates.

JIRA Dashboards ▾ Projects ▾ Issues ▾ Boards ▾ Create

Search 🔍

StuView
STUV board ▾

Backlog

Active sprints

Releases

Reports

Issues

Components

PROJECT SHORTCUTS

github.com

+ Add shortcut

SOURCE CODE

+ Add source code

Invite your team

Project settings

STUV Sprint 23

0 days remaining Complete sprint

QUICK FILTERS: Only My Issues Recently Updated

To Do	In Progress	In Review (Pull Request Pending)	Done
Learning 4 issues			
Implement Interview Video Upload 2 issues			
	<div>Add video url to database</div> <div>Implement Interview...</div> <div>STUV-165</div>		<div>Get full length videos working</div> <div>Implement Interview...</div> <div>STUV-160</div>
Student Interview Front End Functionality 1 issue			
	<div>Get the interview question flow working correctly</div> <div>Student Interview ...</div> <div>STUV-143</div>		
Instructor Functionality 1 issue			
			<div>Deleting an interview doesn't update the view.</div>

StuView / STUV-143

Get the interview question flow working correctly

Details

Status: **IN PROGRESS** (View workflow)

Priority: Medium

Component/s: None

Labels: None

Affects Version/s: None

Fix Version/s: None

Epic Link: Student Interview Front End Functionality

People

Reporter: Michael Peterson [Administrator]

Assignee: Michael Peterson [Administrator]

Dates

Created: 18/Jan/17 4:47 PM

Updated: 2 days ago

STUV Sprint 23






🕒 0 days remaining

STUV-160: 5 unique commits



StuView

[Show all files](#)

Author	Commit	Message	Date	Files
 f18a5ba <small>M</small>		Merge pull request #33 from EWU-CSCD488-W16/STUV-160-Get-full-length-videos-wor...	2 days ago	24 files
 370c8cf		STUV-160 Keep trying to upload chunks until it works	2 days ago	1 file
 d5fb4e2		STUV-160 * Fixed merge script sorting problem. * Changed the way the video is downlo...	2 days ago	8 files
 f2bc069 <small>M</small>		Merge branch 'develop' into STUV-160-Get-full-length-videos-working	3 days ago	26 files
 95e046f		STUV-160 Get full length videos working: * Web workers are being utilized on the brows...	3 days ago	19 files

[Close](#)



STUV-143

Team Meetings & Communication

- We met on average once a week.
- We used slack to communicate.

The screenshot shows a Slack workspace for 'StuView2.0'. The left sidebar contains a list of channels: #general, #github_messages (selected), #meetings, and #random. Below the channels are direct messages with slackbot, ramirser000 (you), cstaaben, gavin, and mikebean233. The main chat area is for the '#github_messages' channel, showing messages from the 'github' app and team members. The messages include notifications about new commits, pull request status changes, and new comments on pull requests.

StuView2.0 ramirser000

#github_messages ☆ | 4 | 0 | Add a topic

the readme. Today

github APP 7:09 PM
[StuView:STUV-143-Get-the-interview-question-flow-working-correctly] 1 new commit by Michael Peterson

[EWU-CSCD488-W16/StuView] Pull request closed: #36 Stuv 150 vimeo upload by ramirser000

[EWU-CSCD488-W16/StuView] Pull request submitted by ramirser000

#37 Stuv 150 vimeo upload

github APP 8:03 PM
[EWU-CSCD488-W16/StuView] New comment by mikebean233 on pull request #37: Stuv 150 vimeo upload

This file should probably be in project/StuView/storage/logs

github APP 8:17 PM
[StuView:STUV-143-Get-the-interview-question-flow-working-correctly] 1 new commit by Michael Peterson

new messages

Demo

Questions?
