



Mohini Banerjee



Shilika Gehlot



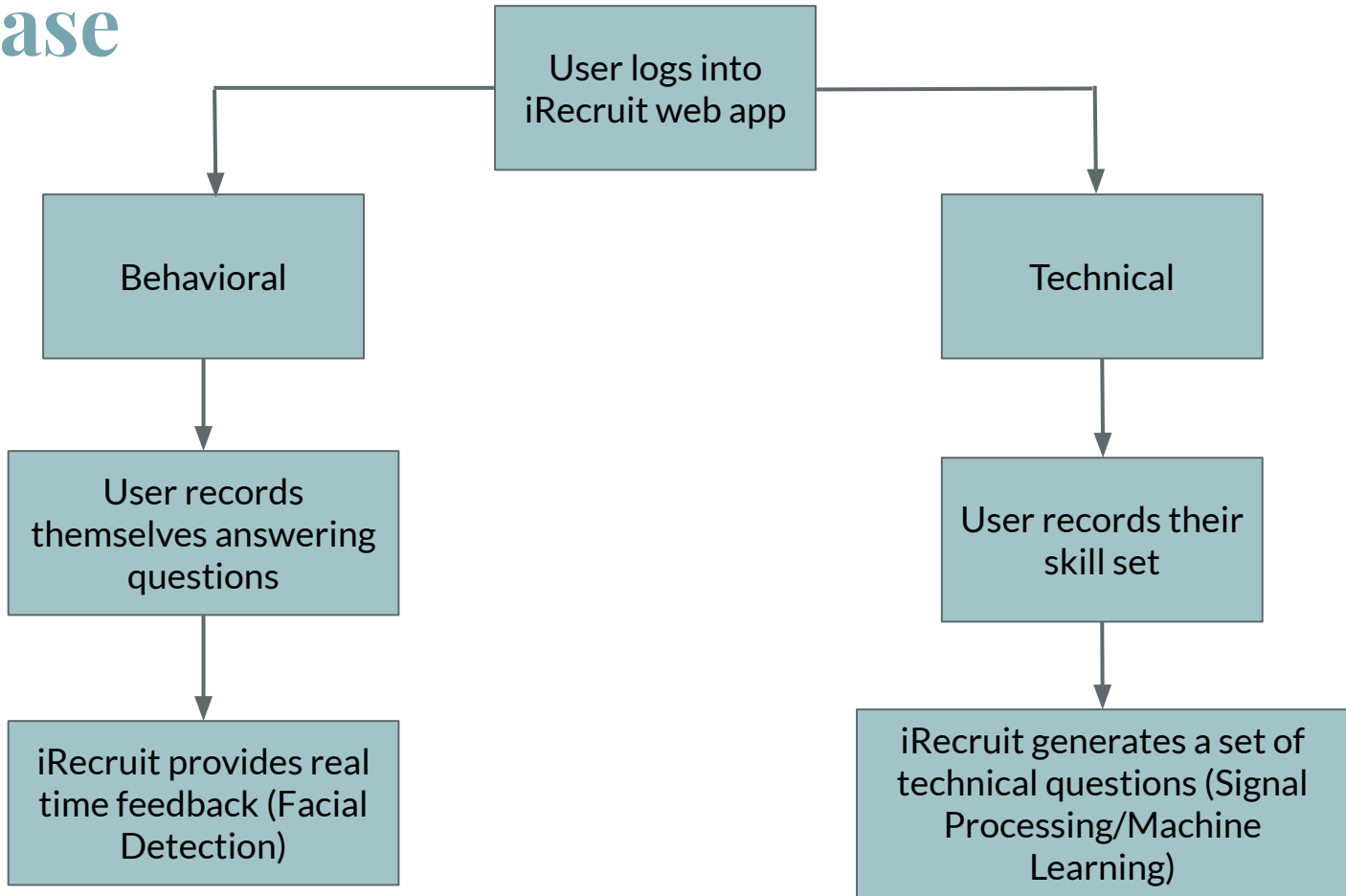
Jessica Meng

iRecruit | Team B2

Background

- Virtual interviews == new reality
- Current resources available: applications, books, and websites
- **Problem Area:** Lack of opportunity to practice a simulated, real-time interview
- **iRecruit:** Interview assistant capable of providing software engineering job-seekers a raw interviewing experience
- **Areas:** Software and Signals

Use Case



Requirements (Facial Detection)

Name	Requirement	Description
Initial Set-Up	≤ 5 seconds	User is given ~5 seconds to position themselves for us to learn facial features
Alerts	2 seconds	If user has subpar eye contact, posture, or screen alignment for 2 seconds, alert them
Accuracy	80%	Using OpenCV Haar Cascades, measured by correct detection of user's facial features

Requirements (Signal Processing)

Name	Requirement	Description
Read Audio Input	Represent audio file as a wave form	Take in sound waves when user spells out skills (letter-by-letter)
Sample and Analyze Audio Input	Ensure no critical information from waves is lost	Represent waves in finite form, determine high energy/frequency parts of waves
Neural Network Input	Successfully extract high energy parts of waves	Processed audio input needs to be fed into a neural network

Requirements (Machine Learning)

Name	Requirement	Description
Letter Determination	Alphabet	Need to determine letters from audio input (neural network)
Accuracy	65%	Taking into account complexity of speech processing part and time frame

Solution Approach

Web app
deployed via
AWS EC2



https://xxx.xxx.xx.x

☰ iRecruit



Name

Help

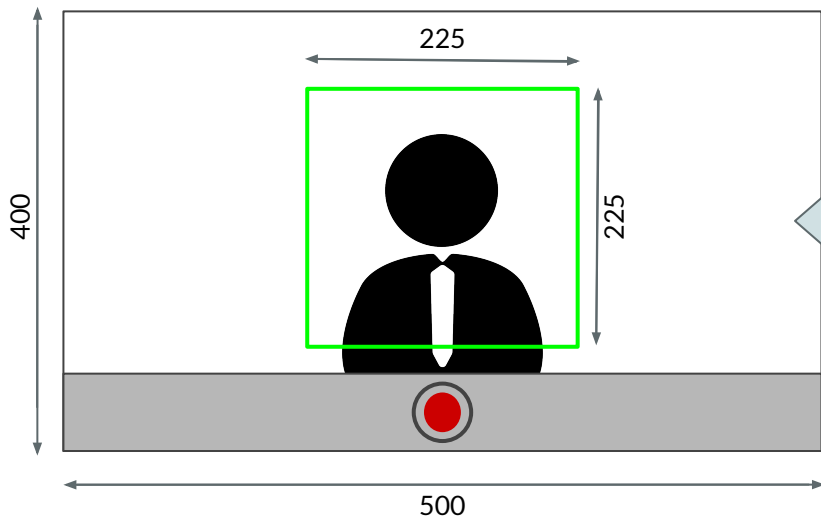
Behavioral

Technical

Q: _____

Qs stored in
fixed database

Previous video
recordings stored
in "Profile"



Save


Technologies:

- Backend (facial feature detection):
 - Python: OpenCV, NumPy, dlib
- Frontend:
 - Python: Django (web framework)
 - HTML/CSS
 - JavaScript (alerts)





Solution Approach

← → ↻ 🏠

☰ iRecruit  Name Help

Behavioral
Technical

Q: _____

Type answer here:

Input: Audio of user's interests and/or strengths

Previous technical questions stored in "Profile"

Post-Processing:

- Backend:
 - Python: SciPy, NumPy, pyplot
- Frontend:
 - Python: Django (web framework)
 - HTML/CSS
- Mathematical Tools:
 - Nyquist Theorem
 - Fourier Transform
 - Neural Network

Testing, Verification and Metrics

- Conduct tests for each implementation task
 - Verify on both backend (monitor data/pixel outputs) and frontend (visual)
- Keep track of actual vs. expected values
 - Facial Detection: # times system alerts user of subpar eye contact, posture, and screen alignment based on predetermined set-up
 - Speech Processing/Machine Learning: # letters from skill set audio file that are correctly captured
- True/false positives, true/false negatives

Tasks and Labor Divisions (Research)

- Facial Recognition (JM):
 - Determine location of eyes, nose, and mouth
 - Decide threshold for good eye contact, posture, and screen alignment
 - Research how to provide real-time feedback
- Signal Processing (SG):
 - Determine how to convert audio signal into integer matrix
 - Research best signal processing algorithms to use to sample the file
- Machine Learning (MB):
 - Determine how to use neural network to classify audio input into letters

Tasks and Labor Divisions (Implementation)

- Implement basic web app components (MB)
- Create behavioral and technical question databases (SG)
- Implement facial detection to locate facial features (JM)
- Set up alerts for subpar eye contact, posture, and screen alignment (JM)
- Implement signal processing to extract audio file features (SG)
- Program machine learning algorithm to determine the letters of processed audio file (MB)

Schedule

