Team D3 – Meal By Words

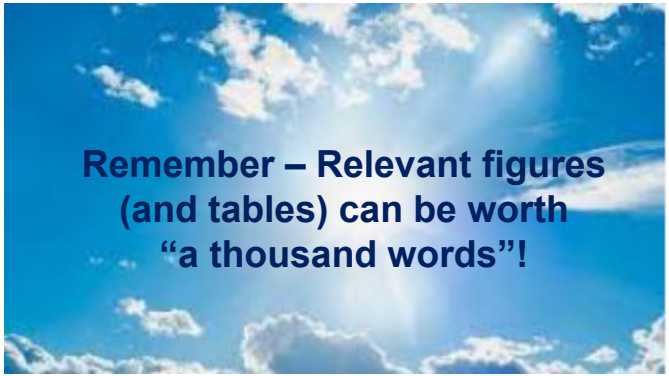
Nina Duan, Lisa Xiong, Shiyi Zhang

Add your 12 slides after this slide… [remember, 12 min talk + 3 min Q/A]

For more information about formatting or importing slides see:
https://gsuite.google.com/learning-center/products/slides/get-started/

Make sure to cover
(refer to the Proposal Presentation Guidance):

- Use Case
- Use-Case Requirements, especially Quantitative
- Technical Challenges
- Solution Approach
- Testing, Verification and Metrics
- Tasks and Division of Labor
- Schedule



Consider that this slide already works as a introduction slide so use your first slide wisely

Use Case

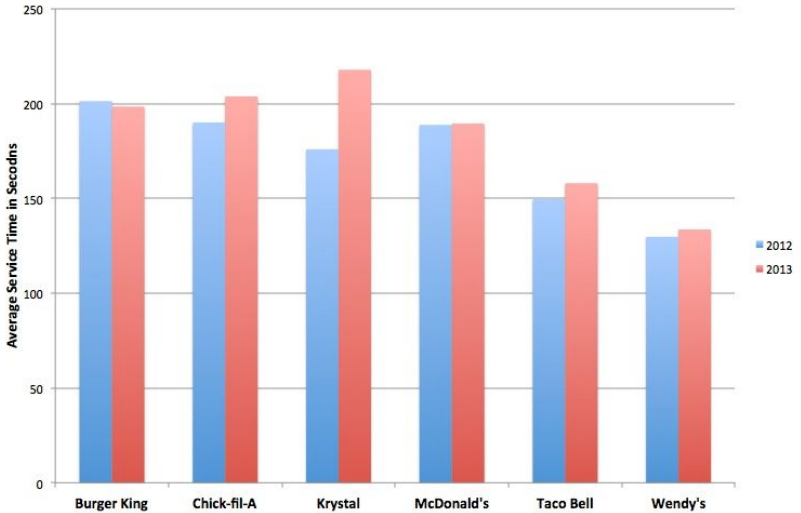
- Speech-operated ordering kiosk
- Areas: hardware, signal processing, software

- Primary customers: fast-food restaurants with fixed menus
- Secondary customers: fast-food restaurant customers

- Problems when ordering with human cashiers
  - Highly dependent on employee training
  - Inaccurate orders
  - Long wait times due to understaffing
  - Shuffling between kitchen and kiosk tires staff out

Use Case Requirements - General

- Should allow customer to verbally:
  - Order an item from the menu
  - Remove an existing order
  - Changing an existing order
  - Finish ordering and proceed to checkout
- Entire process should take <= 200 seconds on average
  - See chart for avg. drive-thru service time
- Should forward customers' orders directly to kitchen staff



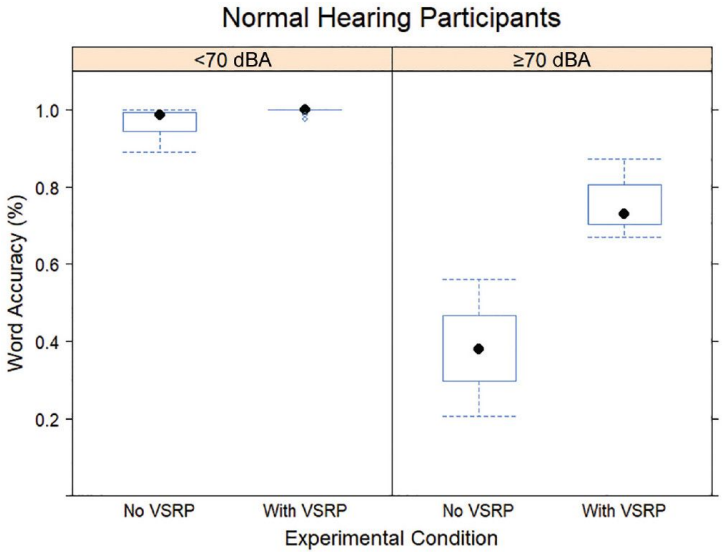
Research by QSR Magazine
(https://www.qsrmagazine.com/content/speed-service)

Use Case Requirements - Input Reception

- Signs on the ground guide customers to stand an arm's length (**0.3m ~ 0.4m**) away
- Detect approaching customer and wake system up
  - **Height**: <= 0.7m (2') to accommodate children and wheelchair-bound customers
  - **Distance range**: 0.3m (1') ~ 1.0m (3'3'')
  - Rather wake up unnecessarily than not waking up
    - 100% accuracy when customer exists (i.e. detect every approaching person)
    - 70% accuracy when no one is around (i.e. 10 empty trials, stay asleep for >= 7)
- Record customer speech
  - **Directional reception**: ~120° to accommodate customer movement
  - **Maximum distance**: 1.0m (3'3'')
  - **Frequency**: 80Hz ~ 260Hz (human voice)
  - **Volume**: 60dB ~ 80dB (normal conversation volume)

Use Case Requirements - Input Recognition

- Recognize customer order (name + quantity)
  - Noise level < 70dBA: 85% accuracy
  - Noise level >= 70dBA: 50% accuracy
  - 69.1dBA ~ 79.1dBA = a typical fast-food restaurant's noise level
  - Human speech perception in quiet environments is ~90%, but drops significantly when noise level rises above 70dBA
    - See chart ("No VSRP" experimental condition)
- Rather miss an order than order wrong item

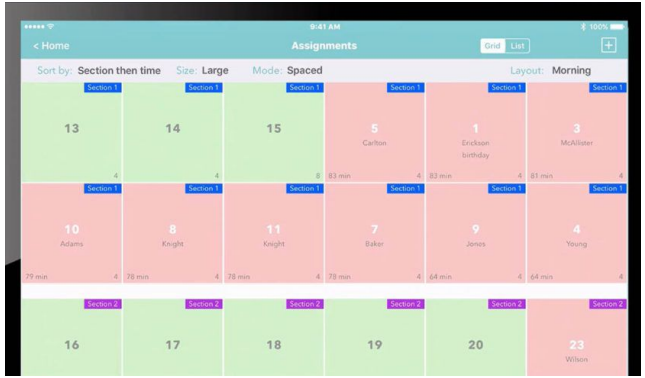


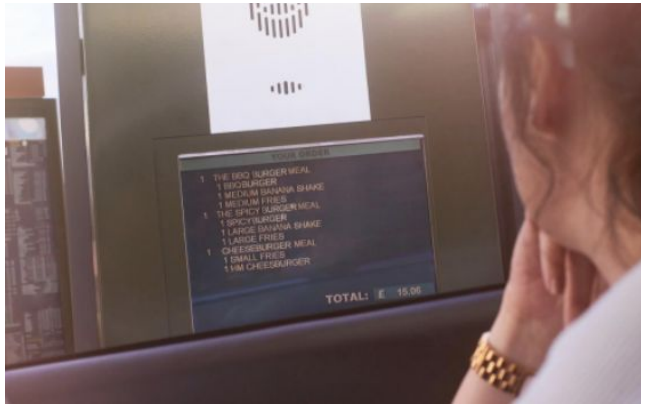
To, Wai Ming, and Andy Chung. "Noise in restaurants: levels and mathematical model." Noise & health vol. 16,73 (2014): 368-73. doi:10.4103/1463-1741.144412

Diagram: Raghavan, Arun M. et al. "Visual Speech Recognition: Improving Speech Perception in Noise through Artificial Intelligence." Otolaryngology-head and neck surgery 163.4 (2020): 771–777. Web.

Use Case Requirements - Input Processing & Utilization

- Customers’ side
  - System times out if customer doesn’t speak up
  - Terminates ordering process and proceeds to checkout
- Kitchen staff’s side
  - Displays ordered items in easily readable format
    - Should be able to read item type and quantity within 3 seconds
  - Groups items based on orders
  - Sequences orders based on order time (oldest to newest), color-code the most urgent ones
  - Allows staff to manually remove prepared orders
- Order delivered to kitchen staff (backend UI) within 1 second





Technical Challenge & Solution - Speech Recognition

- **Challenge**
  - 85% speech recognition accuracy in 70dBA noise
  - Identifying menu items and quantities correctly
- **Solution**
  - Directional microphone with professional sound shield
  - Build sentence segmentation algorithm to parse user input into individual sentences
  - Use Spacy tokenizer (parse words into tokens) to find alternative forms of menu item keywords
  - Identify edge cases and modify the natural language processing program accordingly
- **Risk Mitigation**
  - Build microphone holder that can extend to a position near customer's mouth
  - Allow kiosk usage only in quiet environments (<= 60dBA)
  - Only allow certain sentence structures when ordering
  - Don't allow full sentences at all – only item name and quantity (e.g. "1 cheeseburger")

Technical Challenge & Solution - Order Termination

- **Challenge**
  - Correctly identifying termination of ordering process
    - Timeout (customer leaves kiosk, system wrongly woken up, etc.)
    - Proceed to checkout

- **Solution**
  - For timeout
    - At 1-minute mark: display warning – “System will timeout in <countdown>”
    - At 2-minute mark: remove the current order
    - Any recognized speech can interrupt countdown and restart process
  - For checkout
    - Identify termination cues customers use at checkout (“that’s it”, “I’m done”, “all set”, etc.)
    - Find edge cases in user testing

- **Risk mitigation**: checkout button on screen OR fixed keyword for finishing ordering

Technical Challenge & Solution - Send Order to Staff

- **Challenge**
  - Deliver order to the kitchen’s end within 1 second

- **Solution**
  - Use noSQL key-value cloud database
    - AWS DynamoDB, Redis, MongoDB, etc.
    - No need for relational database: only storing simple, independent objects
    - Key-value database supports faster inserts and deletes

- **Risk mitigation**
  - Instead of using commercial database, send raw order information to staff end (represented by a laptop) and store as local data

Testing, Verification, and Metrics

Unit tests *** Each tested by 10 volunteers ***

- **Noise Tolerance**: ability to receive processable inputs at different background noise levels
  - Volunteer speaks at conversational level (~60dB)
  - Examine difference between microphone right in front of volunteer and microphone placed at kiosk
    - Quiet (<= 40dBA)
    - Normal (50dBA ~ 60dBA)
    - Noisy (>= 70dBA)
  - Use spectrogram to visualize frequency content of filtered audio
- **Recognition Test:** given filtered audio, ability to correctly translate sentence into text
- **Command Test:** given text string, ability to parse into sentences and find keywords
  - “I would like to order **one large fries** *(addition)* and **one large coke** *(addition)*”
  - “**Large fries instead of medium** *(change)* please.”
  - “**One large fries** *(addition)* please. Wait, **forget about the fries** *(deletion)*, just **large coke** *(addition)*.”

Testing, Verification, and Metrics (cont'd)

End-to-End Test

**** Tested by 10 volunteers ****

- What customers (volunteers) order → what kitchen staff (us) see
- Conducted at different background noise levels
  - Volunteer speaks at conversational level (~60dB)
  - Expect order accuracy:
    - Accuracy = (# of correct item entries staff see) / (total # of item entries customers say)
    - Quiet (<= 40dBA): 90%
    - Normal (50 - 60dBA): 85%
    - Noisy (>= 70dBA): 50%

Tasks and Division of Labor

Nina Duan

- Microphone setup
- Microcontroller & microphone integration
- Microcontroller & microphone programming
- Database selection & ramp-up
- Database setup
- Database & NLP integration

Lisa Xiong

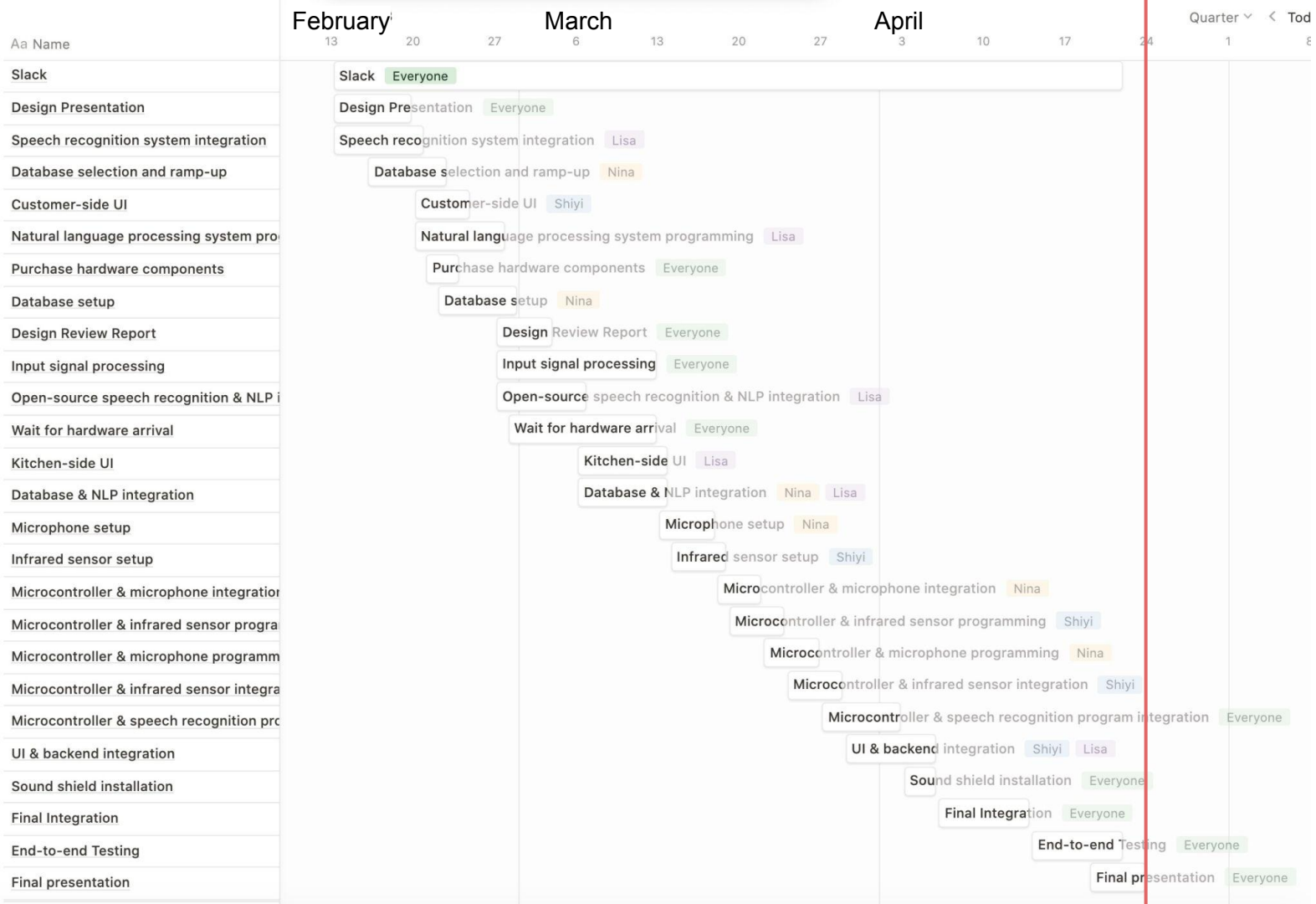
- Speech recognition system integration
- Natural language processing system programming
- Open-source speech recognition & NLP integration
- Kitchen-side UI
- Database & NLP integration
- UI & backend integration

Shiyi Zhang

- Infrared sensor setup
- Microcontroller & infrared sensor integration
- Microcontroller & infrared sensor programming
- Customer-side UI
- UI & backend integration

Everyone

- Purchase hardware components
- Sound shield installation
- Microcontroller & speech recognition program integration
- Input signal processing (noise removal, human voice isolation)


February
March
April
Quarter
Tod
Aa Name
13
20
27
6
13
20
27
3
10
17
24
1
Slack
Design Presentation
Speech recognition system integration
Database selection and ramp-up
Customer-side UI
Natural language processing system pro
Purchase hardware components
Database setup
Design Review Report
Input signal processing
Open-source speech recognition & NLP i
Wait for hardware arrival
Kitchen-side UI
Database & NLP integration
Microphone setup
Infrared sensor setup
Microcontroller & microphone integratior
Microcontroller & infrared sensor progra
Microcontroller & microphone programm
Microcontroller & infrared sensor integra
Microcontroller & speech recognition pro
UI & backend integration
Sound shield installation
Final Integration
End-to-end Testing
Final presentation
Slack Everyone
Design Presentation Everyone
Speech recognition system integration Lisa
Database selection and ramp-up Nina
Customer-side UI Shiyi
Natural language processing system programming Lisa
Purchase hardware components Everyone
Database setup Nina
Design Review Report Everyone
Input signal processing Everyone
Open-source speech recognition & NLP integration Lisa
Wait for hardware arrival Everyone
Kitchen-side UI Lisa
Database & NLP integration Nina Lisa
Microphone setup Nina
Infrared sensor setup Shiyi
Microcontroller & microphone integration Nina
Microcontroller & infrared sensor programming Shiyi
Microcontroller & microphone programming Nina
Microcontroller & infrared sensor integration Shiyi
Microcontroller & speech recognition program integration Everyone
UI & backend integration Shiyi Lisa
Sound shield installation Everyone
Final Integration Everyone
End-to-end Testing Everyone
Final presentation Everyone