COMP3000 Computing Project 2024/2025

# KANO: A Mood Analysis Voice Journal

### Links

Source code:[*https://github.com/Esther-Skillman/KANO-A-Mood-Analysis-Voice-Journal*](https://github.com/Esther-Skillman/KANO-A-Mood-Analysis-Voice-Journal)

Contents

[KANO: A Mood Analysis Voice Journal 1](#_Toc180693647)

[Links 1](#_Toc180693648)

[Project Vision 2](#_Toc180693649)

[Technological Architecture 3](#_Toc180693650)

[Mobile Application UI 3](#_Toc180693651)

[Back-End Server 3](#_Toc180693652)

[AI Processing 3](#_Toc180693653)

[Languages and Technologies 3](#_Toc180693654)

[Development Environment 3](#_Toc180693655)

[Repository Layout 3](#_Toc180693656)

[Test Environment 3](#_Toc180693657)

[Risk Management 3](#_Toc180693658)

[Assessment and Mitigation 3](#_Toc180693659)

[Risk Matrix 4](#_Toc180693660)

[Initial Product Backlog 4](#_Toc180693661)

[High Level Plan 4](#_Toc180693662)

[WBS Diagram 4](#_Toc180693663)

[Gantt Chart 4](#_Toc180693664)

[Project Management Tools 4](#_Toc180693665)

[Project Vision 4](#_Toc180693666)

[Risk Plan 5](#_Toc180693667)

[Proposed Gantt chart 6](#_Toc180693668)

[Keywords 6](#_Toc180693669)

[References 6](#_Toc180693670)

# Project Vision

**FOR** those who have an interest in their own emotional wellbeing, journaling and embrace new technologies involving AI. With these core characteristics, the intended clients’ demographics would be largely age 16-25 with their increasing usage of AI. Additionally, they would be expected to already be existing members of the journaling community, allowing for an easy integration of their current journaling habits to the niche that is voice journalism with mood analysis. Furthermore, the culture would be English speaking and the same as the one represented in the primary dataset(s) used. Other demographics such as gender, economic and social class are intended to not impact the client target audience, but this could be revised later during research and development.

**WHOSE** problem is addressed is an individuals’ inability to conduct an unbiased reflection of one of their many emotional expressions, in this case, speech. According to Whilst the common method of reflection for today is writing, these entries can be written over, once, twice or even three times with technology, obscuring a person’s true emotions to how they think they should feel as opposed to what they actually feel. Voice entries help remove this obscurity, as individuals can’t ‘rewrite’ their speaking as easily, providing a raw entry of sorts but likewise adds a significant key benefit of listening to not only what they say but HOW they say it. Whilst humans are able to discern the emotions of speech with minimal effort, such as anger, happiness or fear, this can fall short when they’re the ones analysing themselves. The process can become bias, thrown behind the lens of their own perspective and this is when technology can intervene - providing an unbiased report to aid an individual in their own reflection.

**THE PRODUCT** KANO which stands for ‘Keep A New Outlook’ will be a mood analysis voice journal that will use AI to discern the emotions of a person’s voice entry.

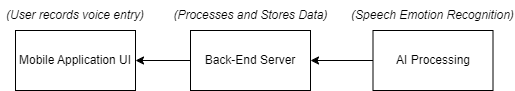
**IT IS A** mobile application that will initially address android mobiles, but compatibility with android and iPhone will be explored during development. Server-side options will be researched and reviewed further into development, dependant on many factors such as computational power, availability of server technologies and greatest suitability for mobile compatibility.

**THAT** which is the fundamental premise is a user will record themselves speaking and the AI will conduct a live analysis after they’ve finished, giving numerical and qualitative feedback on what emotion(s) the user displayed during the voice entry (see Fig 0.1). This premise works to achieve the overall goal of the application being a tool for a user’s mental wellbeing, as they’re able to reflect on an unbiased and numerical picture of the emotion(s) they portray. This will enable them to approach their mental health with a more objective viewpoint and keep a record of the frequency and trends of their emotions over time.

# Technological Architecture

## Initial Architecture

The architecture of the KANO project will involve three key layers: the user interface (UI), the backend server for computational tasks, and AI processing for Speech Emotion Recognition (SER). (Ref Fig)



### Mobile Application UI

The mobile application will act as the front end, allowing users to record voice entries. These entries will then be sent to the backend for processing with an SER model (AI).

### Back-End Server

The backend server will primarily be responsible for interacting with the UI, storing user data and sending the AI voice data to return a qualitative and quantitative analysis to then send to the UI.

### AI Processing

Research and development within the project will reveal how the AI processing will interact with the back-end server to the point they may either be completely independent or merged entities. Treating AI processing as a separate entity for now, this will be where the voice data is processed on a pre-trained SER model to gather a mood analysis and is then sent back to the UI.

## Languages and Technologies

**AI and Machine Learning**

Python will be used for the SER (Speech Emotion Recognition) model development, likely utilising libraries like TensorFlow and PyTorch for deep learning and machine learning tasks.

**Mobile Application**

JavaScript, which may be subject to change, will likely be used within the Android Studio environment for front-end development. The app will be initially developed for Android, with a possible future expansion to iOS if project management goes as planned.

**Backend**

Python will be used on the server side to handle the AI processing.

**Data Storage**

This is largely undecided as it is heavily dependent on the outcomes of development, although options that may be explored is a one option could be SQL Query for SQL databases, whilst technologies may include cloud-based storage.

**Dataset**

Kaggle datasets have been and will be used for initial training and model evaluation, allowing myself as a developer to familiarise myself with key concepts of SER with smaller datasets such as CREMA-D, RAVDESS, SAVEE and TESS. The actual dataset(s) used will be explored with the help of my tutor to establish a suitable set to train a SER model.

## Development Environment

## Repository Layout

*Students need to use GitHub. Ensure you add your supervisor to your repository and edit the readme.md file so that your project title, vision and allocated supervisor are clearly noted. Make sure the name of the repository reflects your name.*

## Test Environment

## Risk Management

### Assessment and Mitigation

*This is the stage where you must also give some consideration as to the potential for things to go wrong. Please read around the topic matter and identify what pertinent risks there might be to your project. Once you have identified what could go wrong, you need to consider the likelihood of that happening and what you will do about it. Any risks that have a high likelihood and a high impact MUST have actions identified to prevent them should the events happen.*

### Risk Matrix

# Initial Product Backlog

# High Level Plan

## WBS Diagram

## Gantt Chart

# Project Management Tools

*You should use appropriate project management tools such as a planner and a diary. Being able to use a diary and control your time is an essential part of being a professional.*

***02 Project Initiation***

*This deliverable is the output from the culmination of your sprint zero. Sprint zero should be spent getting yourself ready for delivering the project. You should identify the point and purpose of your project. This will require you to think of a title and work up a project Vision. Please do not consider that your project title and project vision are immutable – they can change as you develop your ideas. However, you need to know at the start roughly where you want to develop your ideas. By ensuring you start with a coherent project title and project vision, you can ensure you start in an organised fashion.*

*Sprint zero should also be about deciding which languages and technologies to use, setting up your development environment, sorting out your version-controlled repository, identifying your test environment, identifying the things that could go wrong and creating your initial product backlog. The product backlog will be an evolving, living item that changes and evolves as the project progresses. However, you need to ensure that there is enough in the product backlog to begin with. You should also include a high-level plan to show roughly when you intend to complete certain things – a recommended format would be a Gantt chart. Please refer to the documentation on the DLE around project management to help you with this.*

***02 Submission details***

*Use the 02.1 P1 template for your initial deliverable. Remember to delete the grey text and replace it with your own words. The grey text is there for guidance, not for submission. Save your document as a PDF prior to uploading. No other document format is accepted.*

### Project Vision

### Risk Plan

|  |  |  |
| --- | --- | --- |
| Risk Plan | | |
| Ref | Potential Risk | Mitigation Strategies |
| R1 | Unrealistic Time Estimates | Incremental development, tutor feedback, analysis of past work ethic/projects, multiple time estimation techniques, revision of schedule after MVP has been achieved |
| R2 | Personnel shortfalls | Tutor teaching and resources, training via online courses, university support e.g. writing cafe |
| R3 | Missed deadlines for development | Schedule contingency time for possible overruns, requirements revision, assess critical path, revise and extend schedule for similar developments |
| R4 | Personnel unexpected event (illness etc.) | Schedule potential sick days, assess impact to critical path, tutor meeting to evaluate next steps |
| R5 | Software doesn’t meet requirements | Prototyping, tutor feedback, consistent evaluation of requirements during development, software research |
| R6 | Software doesn’t meet user/client expectations | user testing and feedback (UAT), prototyping, alpha testing with MVP |
| R7 | Gold plating | Assess design time-cost, prototyping, requirements scrubbing |
| R8 | Development too technically challenging | Technical analysis with tutor feedback, technical training, requirements scrubbing |

Computational power and speed

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Risk Plan | | | | |
| Ref | Risks | Likelihood | Impact | Risk Exposure |
| R1 | Unrealistic Time Estimates | 5 | 4 | 20 |
| R2 | Personnel shortfalls | 4 | 3 | 12 |
| R3 | Missed deadlines for development | 4 | 5 | 20 |
| R4 | Personnel unexpected event (illness etc.) | 3 | 3 | 9 |
| R5 | Software doesn’t meet requirements | 3 | 4 | 12 |
| R6 | Software doesn’t meet user/client expectations | 2 | 4 | 8 |
| R7 | Gold plating | 4 | 2 | 8 |
| R8 | Development too technically challenging | 4 | 3 | 12 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Risk Assessment Matrix | | | | | |
|  | 1 | 2 | 3 | 4 | 5 |
| 1 |  |  |  |  |  |
| 2 |  |  |  | R7 |  |
| 3 |  |  | R4 | R2 R8 |  |
| 4 |  | R6 | R5 |  | R1 |
| 5 |  |  |  | R3 |  |

### Proposed Gantt chart

*Identify here your proposed work packages and how they will fit together over the time you have available. This is a high level estimate.*

### Keywords

voice, speech, SER, journal, dairy, mental health, healthcare, mood, analysis, AI, machine learning, deep learning, pattern recognition, mobile application, android

## References

1