

# Main Page

From 3002S12021\_HackElite

## Contents

- 1 Team Information
  - 1.1 Project Manager
    - 1.1.1 Vaish Arjun
  - 1.2 QA Manager
    - 1.2.1 Kiran Mac Milin
  - 1.3 QA Engineer
    - 1.3.1 Roxas Abby Maurea Imus
  - 1.4 Lead Developer
    - 1.4.1 Asok Kumar Gaurav
  - 1.5 Front-End Developer
    - 1.5.1 Arya Shashwat
  - 1.6 Back-End Developer
    - 1.6.1 Chandna Divvij
  - 1.7 Release Engineer/Manager
    - 1.7.1 Koh Liang Jing
- 2 Project Proposal
  - 2.1 Executive Summary
  - 2.2 Statement of Problem
  - 2.3 Objectives
    - 2.3.1 Face Recognition
    - 2.3.2 Natural Language Processing (NLP)
  - 2.4 Technical Approach
    - 2.4.1 Plan of Action
    - 2.4.2 Customer Needs
    - 2.4.3 Target Specifications
    - 2.4.4 Technology Consideration
    - 2.4.5 System Architecture/Platform
  - 2.5 Project Management
    - 2.5.1 Deliverables
    - 2.5.2 Communication and Coordination with Sponsor
    - 2.5.3 Team Qualifications
  - 2.6 Conclusion
  - 2.7 References
  - 2.8 Appendix A: Résumés of Team Members
- 3 Use Case Model
  - 3.1 Use Case Diagram
  - 3.2 Use Case Descriptions
- 4 Backlog
  - 4.1 Meeting 1
  - 4.2 Meeting 2
  - 4.3 Meeting 3
  - 4.4 Meeting 4
  - 4.5 Meeting 5
- 5 System Requirement Specifications (SRS)
  - 5.1 Problem Statement
  - 5.2 Overview
    - 5.2.1 Background
    - 5.2.2 Overall Description
  - 5.3 Investigation & Analysis Methodology
    - 5.3.1 System Investigation
    - 5.3.2 Analysis Methodology
      - 5.3.2.1 Feasibility study and requirements elicitation
      - 5.3.2.2 System analysis and requirements specification
        - 5.3.2.2.1 Perform an analysis of the problem using object-oriented techniques
        - 5.3.2.2.2 Scope and Limitations
      - 5.3.2.3 Object-oriented design using UML
      - 5.3.2.4 Prototyping
  - 5.4 Constraints
    - 5.4.1 Scalability
    - 5.4.2 Open Source
    - 5.4.3 Development Costs
    - 5.4.4 Project Schedule
  - 5.5 Operational Requirements
    - 5.5.1 Help Desk Support
    - 5.5.2 Application Services and Technical support
    - 5.5.3 Administration Features
    - 5.5.4 Chatbot System independent of Emotion Detection
    - 5.5.5 System hardware fail over and routine back up
    - 5.5.6 Audit Trail
  - 5.6 Functional Requirements
    - 5.6.1 AI Chatbot System for Mental Wellbeing
      - 5.6.1.1 Registration/ Login
      - 5.6.1.2 Profile Settings
      - 5.6.1.3 Chatbot Interaction
      - 5.6.1.4 Wellness/Wellbeing Centres
  - 5.7 Input Requirements
    - 5.7.1 User identifier key and user access

- 5.7.2 Facial Recognition
- 5.8 Process Requirements
  - 5.8.1 Performance
  - 5.8.2 Responsiveness
  - 5.8.3 Usability
  - 5.8.4 Data Validation
  - 5.8.5 Data Security
  - 5.8.6 Data Repository
- 5.9 Output Requirements
  - 5.9.1 Confidentiality and Privacy
- 5.10 Hardware Requirements
  - 5.10.1 Network
  - 5.10.2 User Computers
- 5.11 Software Requirements
  - 5.11.1 Client Operating System
  - 5.11.2 Client Application
  - 5.11.3 Network System
  - 5.11.4 Software Tools and Libraries
  - 5.11.5 Backend Server
  - 5.11.6 Testing
  - 5.12 Deployment Requirements
- 6 Quality Plan
  - 6.1 Signature Page
  - 6.2 Document Change Record
  - 6.3 Purpose and Scope
    - 6.3.1 Purpose
    - 6.3.2 Scope
  - 6.4 Reference Documents
  - 6.5 Management
    - 6.5.1 Management Organisation
      - 6.5.1.1 Project Management
      - 6.5.1.2 Assurance Management
    - 6.5.2 Tasks
      - 6.5.2.1 Product Assessments
      - 6.5.2.2 Process Assessments
    - 6.5.3 Roles and Responsibilities
      - 6.5.3.1 QAM
      - 6.5.3.2 Software Quality Personnel
  - 6.6 Documents
    - 6.6.1 Purpose
    - 6.6.2 Minimum Document Requirements
  - 6.7 Standards, Practices, Conventions and Metrics
    - 6.7.1 Purpose
    - 6.7.2 Software Quality Program
      - 6.7.2.1 Functionality
      - 6.7.2.2 Maintainability
      - 6.7.2.3 Reliability
      - 6.7.2.4 Usability
    - 6.7.3 Standard Metrics
  - 6.8 Software Reviews
    - 6.8.1 Purpose
    - 6.8.2 Minimum Software Reviews
  - 6.9 Test
  - 6.10 Problem Reporting and Corrective Action
  - 6.11 Tools, Techniques and Methodologies
    - 6.11.1 Software Quality Tools
  - 6.12 Media Control
  - 6.13 Supplier Control
  - 6.14 Record Collection, Maintenance, and Retention
  - 6.15 Training
  - 6.16 Risk Management
  - 6.17 SQA Plan Change Procedure and History
- 7 Project Plan
  - 7.1 Revision History
  - 7.2 Introduction
    - 7.2.1 Project Overview
    - 7.2.2 Project Description and Scope
  - 7.3 Project Organization
    - 7.3.1 Team Structure
    - 7.3.2 Roles & Responsibilities
    - 7.3.3 Team Communication
  - 7.4 Process Definition
  - 7.5 Schedule
    - 7.5.1 Activity Dependencies and Schedule
    - 7.5.2 Work Breakdown Structure
    - 7.5.3 Work Packages
    - 7.5.4 Activity Dependencies
    - 7.5.5 Work Package Details
  - 7.6 Project Estimates
    - 7.6.1 Code Size Estimation using Function Points
      - 7.6.1.1 Unadjusted Function Points
      - 7.6.1.2 Adjusted Function Points
      - 7.6.1.3 Lines of Code
    - 7.6.2 Efforts, Duration and Team Size Estimation
      - 7.6.2.1 Distribution of Effort
      - 7.6.2.2 Cost Estimates
        - 7.6.2.2.1 Hardware

- 7.6.2.2.2 Software
- 7.6.2.2.3 Other Requirements
- 7.7 Product Checklist
- 7.8 Best Practice Checklist
- 7.9 Risk Management
  - 7.9.1 Technological Risk
  - 7.9.2 Usability Risk
  - 7.9.3 People-related Risk
- 7.10 Quality Assurance
- 7.11 Monitoring & Control
  - 7.11.1 Timeline Planning and Task Decomposition
  - 7.11.2 Requirements Traceability Matrix (RTM)
  - 7.11.3 Identification of Major Project Risks
  - 7.11.4 Perform Quality Control
  - 7.11.5 Review and Status Meetings
- 8 Risk Management Plan
  - 8.1 Version History
  - 8.2 Introduction
    - 8.2.1 Purpose of the Risk Management Plan
  - 8.3 Risk Management Procedure
  - 8.4 Tools and Practices
  - 8.5 Risk Management Plan Approval
- 9 Software Design Maintainability Report
  - 9.1 Design Strategies
    - 9.1.1 The Planning Phase Before Development
    - 9.1.2 The Process of Developing
    - 9.1.3 Correction by Nature
    - 9.1.4 Enhancement by Nature
    - 9.1.5 Maintainability Practices
  - 9.2 Architectural Design Patterns
  - 9.3 Software Configuration Management Tools
    - 9.3.1 MediaWiki
    - 9.3.2 GitHub
    - 9.3.3 Google Drive
- 10 Software Configuration Management Plan
  - 10.1 Identification
    - 10.1.1 Document overview
    - 10.1.2 Abbreviations and Glossary
    - 10.1.3 References
      - 10.1.3.1 Project References
      - 10.1.3.2 Standard and regulatory References
    - 10.1.4 Conventions
  - 10.2 Organization
  - 10.3 Configuration identification
  - 10.4 Configuration control
    - 10.4.1 Change Management
  - 10.5 Configuration support activities
    - 10.5.1 Configuration Status Accounting
      - 10.5.1.1 Evolutions traceability
      - 10.5.1.2 Setting up Configuration status
      - 10.5.1.3 Configuration status diffusion
      - 10.5.1.4 Configuration status records storage
    - 10.5.2 Configuration audits
    - 10.5.3 Reviews
    - 10.5.4 Configuration management plan maintenance
- 11 Change Management Plan
  - 11.1 Version History
  - 11.2 Introduction
    - 11.2.1 Purpose Of The Change Management Plan
  - 11.3 Change Management Process
    - 11.3.1 Change Request Process Flow Requirements
    - 11.3.2 Change Request Form and Change Management Log
    - 11.3.3 Change Request Form and Change Management Log
    - 11.3.4 Change Request Form and Change Management Log
  - 11.4 1. Change Control Board
  - 11.5 Responsibilities
- 12 Release Plan
  - 12.1 Revision History
  - 12.2 Introduction
  - 12.3 Referenced Documents
  - 12.4 Overview
  - 12.5 Assumptions, Constraints, Risks
    - 12.5.1 Assumptions
    - 12.5.2 Constraints
    - 12.5.3 Risks
  - 12.6 Release Approach
    - 12.6.1 Rationale
    - 12.6.2 Release Strategy
      - 12.6.2.1 Release Content
      - 12.6.2.2 Release Schedule
      - 12.6.2.3 Release Impacts
      - 12.6.2.4 Release Notification
- 13 Test Plan
  - 13.1 Test Plan Identifier
  - 13.2 Introduction
  - 13.3 Test Items
  - 13.4 Features To Be Tested

- 13.5 Features Not To Be Tested
- 13.6 Approach
- 13.7 Item Pass/Fail Criteria
- 13.8 Suspension Criteria And Resumption Requirements
- 13.9 Test Deliverables
- 13.10 Testing Tasks
- 13.11 Environmental Needs
- 13.12 Responsibilities
- 13.13 Staffing And Training Needs
- 13.14 Schedule
- 13.15 Risks And Contingencies
- 13.16 Approvals
- 14 Test Cases and Requirements Test Coverage Report
- 15 CMMI Level 2 Definition
  - 15.1 Executive summary
  - 15.2 Description
  - 15.3 Level 2 KPIs
    - 15.3.1 Requirement Management (REQM)
    - 15.3.2 Project Planning (PP)
    - 15.3.3 Project Monitoring and Control (PMC)
    - 15.3.4 Process and Product Quality Assurance (PPQA)
    - 15.3.5 Configuration Management (CM)
    - 15.3.6 Measurement and Analysis (MA)
    - 15.3.7 Supplier Agreement Management (SAM)
  - 15.4 Generic goals and practices
  - 15.5 Specific goals and practices
  - 15.6 Approvals
  - 15.7 CMMI audit checklist
- 16 Meeting Minutes
  - 16.1 Meeting 1
  - 16.2 Meeting 2
  - 16.3 Meeting 3
  - 16.4 Meeting 4
  - 16.5 Meeting 5
  - 16.6 Meeting 6
  - 16.7 Meeting 7
- 17 Presentation Slides

## Team Information

### Project Manager

#### Vaish Arjun

As the Project Manager, Arjun leads the planning, and division of workload. He will also oversee the progress made on the project as well as the management of budget.

**Email:** ARJUN014@e.ntu.edu.sg, **Phone number:** 8348 1255

### QA Manager

#### Kiran Mac Milin

As the Quality Assurance Manager, Mac is in charge of the overall product and process quality by implementing appropriate QA processes.

**Email:** MACMILIN001@e.ntu.edu.sg, **Phone number:** 8872 8067

### QA Engineer

#### Roxas Abby Maurea Imus

As the Quality Assurance Engineer, Abby will devise and conduct the test plans to ensure that the product is up to standards and will meet the customer's expectation.

**Email:** ABBYMAUR001@e.ntu.edu.sg, **Phone number:** 9068 2219

### Lead Developer

#### Asok Kumar Gaurav

As the Lead Developer, Gaurav is responsible for overall technical components of the project.

**Email:** ASOK001@e.ntu.edu.sg, **Phone number:** 9811 4906

### Front-End Developer

#### Arya Shashwat

As the Front-End Developer, Shashwat is responsible for implementing the visual components of the product where the users can interact with.

**Email:** SHASHWAT002@e.ntu.edu.sg, **Phone number:** 9087 3378

## Back-End Developer

### Chandna Divvij

As the Back-End Developer, Divvij is in charge of handling the architecture and the database of the product.

**Email:** DIVVIJ001@e.ntu.edu.sg, **Phone number:** 9897 5410

## Release Engineer/Manager

### Koh Liang Jing

As the Release Manager, Liang Jing will create baselines and integrate changes for delivery. She will also manage the releases of the product.

**Email:** LKOH015@e.ntu.edu.sg, **Phone number:** 8484 4627

# Project Proposal

## Executive Summary

Mental wellness is a much discussed topic in modern society. There are many elements such as academics, jobs and relationships that can cause stress and hamper a person's mental health. Being in control of one's mental faculties and identifying the challenges towards mental wellness enables people to handle challenges, build strong relationships and enjoy life. [1]

There are a lot of resources available to help one cope with mental stress. These include talking to professionals and therapeutic chatbots that mimic a real personal interaction. These chatbots make use of Natural Language Processing in order to understand the user's statements and give appropriate responses. They also provide easy access to people who can't see a professional.

Given the ease of use of chatbots, they are quite popular. However, they lack a personal touch to their interactions and quite often their responses are generic. This discourages user participation and leaves them feeling dissatisfied. Given the advent of AI and new machine learning libraries being launched every other day, the quality of interaction with an autonomous chatbot can be improved manifold.

In an effort to create more interactive and human-like chatbots, our team plans to create a website featuring the improved version of their counterparts. We would use facial recognition to detect a user's emotional state and provide a virtual counsellor to have a conversation with. Our goal is to use Natural Language Processing to make the interaction as human-like as possible.

## Statement of Problem

One of the very first chatbots called Eliza was developed in 1964 at MIT. It used pattern matching to give responses. Artificial Intelligence has come a long way since then and now we have machines that can actually contextualize a user's statements. However, this sophisticated AI at times still can't perfectly mimic a human conversation. Therapeutic chatbots use filler sentences like "Tell me more" or "That's very interesting" often. These generic responses give off the feeling that one is talking to a machine rather than a counsellor.

There is a lot of information to be extracted about a person's mental state from their face. Facial recognition and computer vision have made it possible for a computer to gauge a user's emotions through a camera. Yet designers often neglect this particular data and put all their focus on making algorithms for conversations. As a result, the user is less engaged with a chatbot that is less inclined to customize the conversation based on the user's mood.

It is important to create a platform for users to be able to express their emotions, both through texts and facial expressions. A chatbot makes the platform more interactive by engaging in the user's emotion as well as recommending healthy ways to cope with the emotions.

## Objectives

This document proposes a website that allows a user to have a virtual conversation with a therapeutic chatbot for an improved mental state. The objectives are: -

1. Using face recognition to detect a user's emotional state
2. Creating a chatbot driven by machine learning and NLP to have a human-like conversation

## Face Recognition

In this project, we plan to use facial recognition to detect a user's emotional state. This could be sadness, anger and loneliness among others. The customer's emotional state would then be used to provide them with a targeted therapeutic chatbot whose responses would be tailored to handling that particular emotion. We also plan to incorporate mini games to boost the user's mood.

A person's face is assigned various markers such as sloping of the brow and corners of the mouth that contribute to the detection of emotions. Through mathematical modelling of datasets of faces, this process can be done with a high degree of accuracy. For optimum results, an accuracy of at least 95% would be ideal.

## Natural Language Processing (NLP)

NLP is a branch of Artificial Intelligence that describes how computers understand human language. This can be used to interpret a user's intent properly and accurately. Our project would feature a chatbot that can understand a user's response and reply appropriately using NLP.

## Technical Approach

### Plan of Action

We would start the project by splitting the team in 3 main groups - Frontend, Backend and Machine Learning. The front end team would work with React and Bootstrap to create a visually appealing and user friendly interface, while the backend team would work with Flask to create a backend server. They would also use Unity to create the mini games for the website. The machine learning team is further divided into two parts, the first one uses OpenCV and TensorFlow for emotion detection, and the second team uses Azure Bot framework and LUIS (Language Understanding) to accomplish Natural Language Processing for the chatbot.

Each of the three components is very important as everything affects the user experience. The interaction should be extremely intuitive and smooth and the user should feel satisfied after the conversation with the chatbot.

## **Customer Needs**

The need for this application stemmed from the emergence of the Coronavirus Pandemic (COVID-19) in 2019. Government agencies around the world made the executive decision to shut their borders, put their country into lockdown and quarantine people in their homes to limit the spread of the virus. Globally, the amount of calls to mental health hotline has increased drastically as well. With lock down in place, people might feel restricted and start to feel that they have no control over their situation hence leading to a build up in stress. [3] Which leads to an increased need for an outlet for people to improve their mental wellness. Hence, the creation of an AI chatbot to improve mental health.

An online chatbot also eliminates the need to see a professional in person. As our application would be available 24/7, people that might not have the time to see a professional can do so on our application at any time without any interruptions. Furthermore, most people with mental health issues might be scared to open up to another human being. As it can be overwhelming to open yourself up and not be able to predict the outcome of that openness. Which is eliminated using a chatbot as there will be absolutely no judgement. In addition, professional therapy is known to be expensive and not everyone has the financial ability to do so, thus, our application will be able to reach out to those in need without them worrying about anything else.

## **Target Specifications**

The target specification will be derived from the needs of the customer and together with the technology that will be put in place. There will be a table for the specification and its targeted results. We will be doing timely checks to see if the specifications are met and up to standards.

The customer needs that our application targets are listed as follows:

1. Users who are hesitant to go to Well-Being Centres: As mentioned above, we plan to implement an online chatbot which can help the user to overcome the feeling of hesitation of seeing and communicating with an actual person.
2. Users who want to save money: A counseling session in Singapore can cost around 60-70 SGD on average. This can be considered expensive for some people. Our free application eliminates the monetary need and helps the user focus on their mental health without any financial burden.
3. Users who want to search for well being centres close to them: Our application also provides details of actual therapy centres and professionals to which the user can contact in case they wish to. The user can see all the centres in their vicinity and then choose to book an appointment according to their convenience.
4. Users who want positivity: The aim of our application is to spread positivity. We incorporate mini games along with embedded YouTube videos with the aim of spreading smiles and positivity.

## **Technology Consideration**

Web Technologies	<b>React JS</b> – open source JavaScript framework used for designing user interfaces and state management with a high level of abstraction. This would be used to design the website's front-end.
	<b>Bootstrap</b> – a front-end framework featuring a vast library of pre-built UI components to choose from (compatible with React). It would be used along with react for front-end development.
Back-end	<b>Flask</b> – a python based web framework featuring tools and libraries used for the deployment of a back-end system.
AI and Machine Learning	<b>Face recognition</b> <ul style="list-style-type: none"> <li>- <b>OpenCV</b> (Open Source Computer Vision Library) is a Python based open source computer vision and machine learning software library.</li> <li>- <b>Tensorflow</b> is a python friendly open source library for numerical computation that makes machine learning faster and easier.</li> </ul> <p>OpenCV and tensorflow would be utilized for face recognition and emotion detection.</p>
	<b>NLP - LUIS</b> (Language Understanding) is a cloud-based conversational AI service that applies custom machine-learning intelligence to a user's conversational, natural language text to predict overall meaning and provide suitable responses.  <b>Chatbot - Azure Bot Framework</b> is an open-source SDK with tools for end-to-end chatbot development.
Game Development	<b>Unity</b> - Unity is a cross platform game development engine. We would use it to create mini games and load the content onto the website.

## System Architecture/Platform

The diagram below represents the system architecture that our team plans to build. It comprises three main elements - the front-end, back-end and the users.

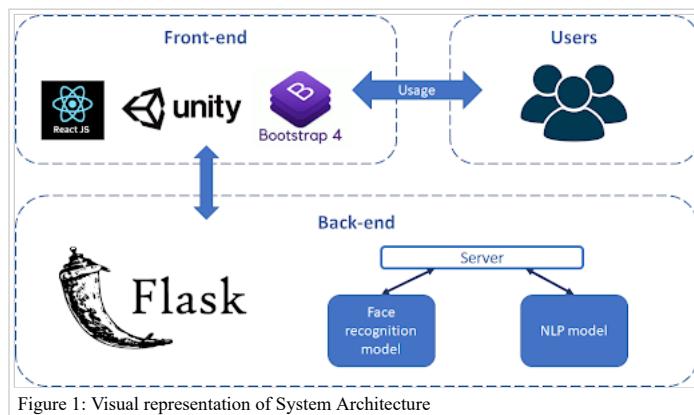


Figure 1: Visual representation of System Architecture

1. **Web development** - This would be carried out mostly in JavaScript with the help of React JS and Bootstrap. The website would feature a user friendly and intuitive interface.
2. **Back-end** - The backend server would be built on the Flask framework with Python. It would be responsible for running the face recognition model for emotion detection as well as the NLP model for generating responses to the user's conversation.
3. **Game Development** - Our frontend team would use Unity to develop mini games for the user. This content would then be loaded on the website.
4. **Development Tools** - For coding and development we would use Visual Studio Code. It is an open source code editor developed by Microsoft which can support both Python and JavaScript. This makes it ideal for us to use as a team.
5. **Version Control** - Our development team plans to use Git for version control in this project. This would enable us to coordinate our parts and track changes in the code over time. This could be done by initializing a repository on GitHub enabling us to merge all our codes from time to time. We also hold weekly meetings and commit our week's work for others to review. It helps us stay up to date with everyone's work and test any new code.

**6. Wiki** - Our team would create a wiki site to collaborate and share knowledge. This would also help us to keep a track of our progress.

## Project Management

Attached below is a Gantt Chart that we have created according to the weekly deliverables that are put in place. The deliverables for week 1 do not have any dependencies and we have split the workload according to our strengths amongst our team of 7. From there onwards we would be tackling each deliverable based on our assigned roles. We have assigned a team member to make sure that the team is on track when it comes to delivery of the required assignments, we intend to have timely warnings to warn us of impending schedule slips.

For the first 4 weeks of the project, we will be focused more towards the planning because we believe as a team that project planning is at the utmost importance. As we move to complete the weekly deliverables, each team member will be responsible for areas within their assigned roles.

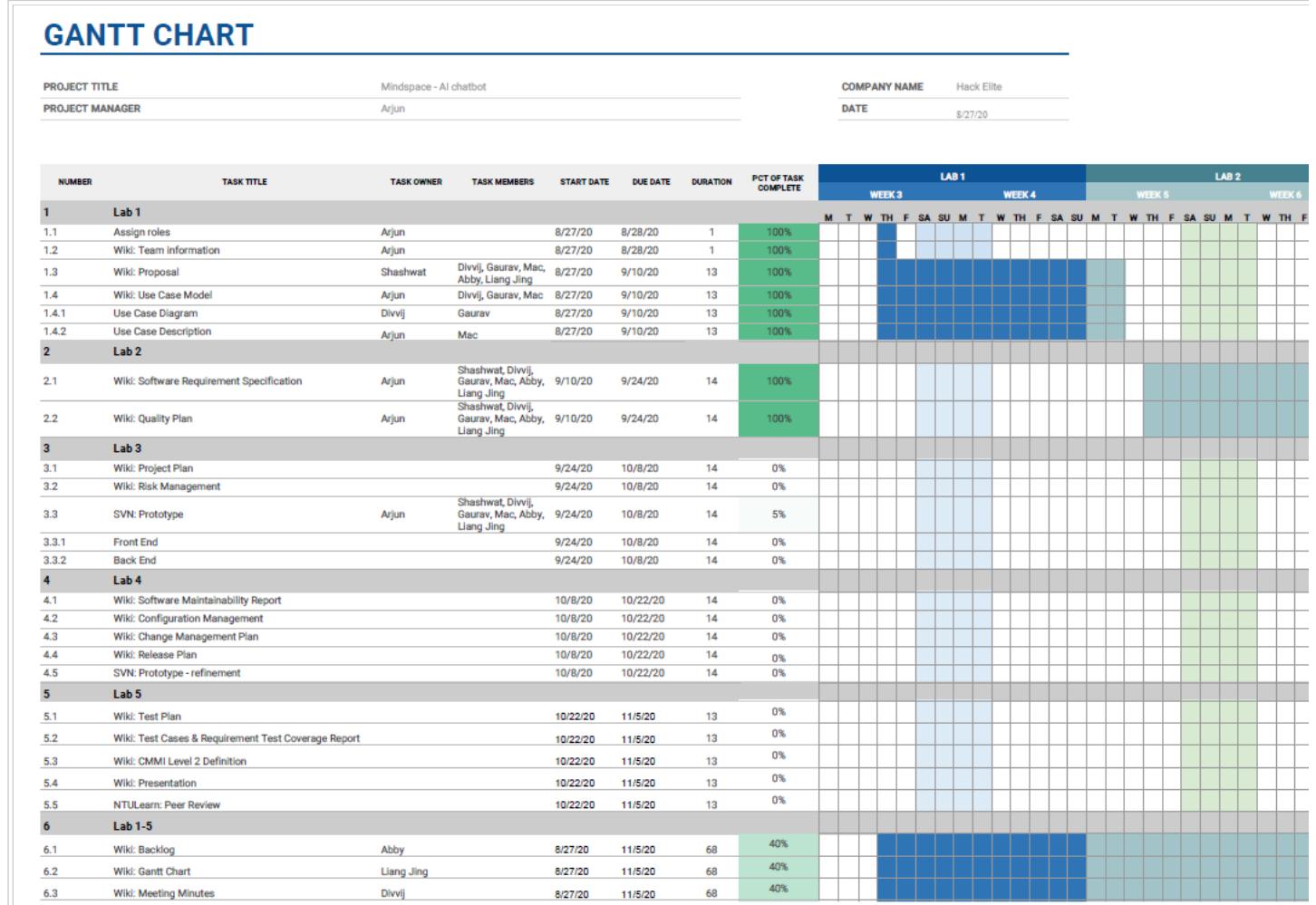


Figure 2: Gantt chart for the project. The solid bars indicate the portions of the tasks that we have accomplished.

## Deliverables

A web application called ‘MindSpace’ will be presented. The crux of MindSpace revolves around a chatbot that uses Artificial Intelligence and Natural Language Processing to determine the mood/emotion of the user and recommends ways in which the user can improve their mood. In the current day and age, mental wellbeing is considered a very integral part of every individual’s life regardless of whether they suffer from mental illnesses or not. The app aims to improve a person’s mood and improve their productivity.

The application will use technology like Facial Emotion Recognition to identify a user’s mood. The chatbot will combine different factors like Facial Emotion and the text-based chat to determine how the user is feeling. We will make use of Cognitive Behaviour Therapy (CBT) methods to mitigate negative emotions. The various therapy methods include Video & Game therapy, Guided Meditation Therapy etc. Our algorithm will consider the user’s feelings holistically and then recommend an appropriate mode of therapy.

The end-goal of this application is to provide a platform for people from all walks of life to lead better lives by being mindful. We plan to complete the first build of the physical prototype by week 7. It would comprise a rough interface of what the finished product would look like and test a basic connection with the backend server. The machine learning team would set up the core libraries needed and fit the appropriate models.

We would test the React code using the Jest library. It is capable of running parallel unit tests and provides a detailed description of the code coverage. Our aim is to get at least 85% code coverage to reduce errors.

## Communication and Coordination with Sponsor

We would have weekly interactions with the sponsor and discuss our progress and our plans for the upcoming week. Our project manager Arjun would head these meetings, take inputs from the sponsor and demo our prototype build for that duration.

# Week	Type of Meeting	Agenda	Points to Discuss
3	Zoom Call	Team introduction	<ul style="list-style-type: none"> <li>• Introduce the sponsor to the project team</li> <li>• Elaborate each team members' roles and responsibilities</li> </ul>
4	Email	Deliverables	<ul style="list-style-type: none"> <li>• Sponsor to receive email regarding all the deliverables and documents for the project</li> <li>• Include progress so far on the documents</li> <li>• Request guidance from sponsor on points of ambiguity</li> </ul>
5	Zoom Call and Email	Completed Deliverables and Prototype	<ul style="list-style-type: none"> <li>• Email sponsor all the completed documents and project proposal</li> <li>• Request inputs on areas of improvement</li> <li>• Discuss specifications of the prototype to be built and the quality plan</li> </ul>
6	Email	Progress of the Prototype	<ul style="list-style-type: none"> <li>• Update sponsor about the progress of the prototype</li> <li>• Mention any hurdles/change of plan</li> </ul>
7	Zoom Call	Prototype demo and Future Development	<ul style="list-style-type: none"> <li>• Demo the completed prototype via zoom call</li> <li>• Take design inputs from the sponsor to implement in the next iteration</li> <li>• Decide on a project development lifecycle (suggest waterfall methodology)</li> </ul>
8	Email	Progress update and Project Plan	<ul style="list-style-type: none"> <li>• Email a complete progress breakdown to the sponsor</li> <li>• Include the completed project plan</li> </ul>
9	Email	Update	<ul style="list-style-type: none"> <li>• Update sponsor about the project's progress</li> <li>• Request for feedback and inputs</li> </ul>
10	Zoom Call	Release Plan and Demo	<ul style="list-style-type: none"> <li>• Discuss release plan with sponsor</li> <li>• Demo the first release prototype</li> </ul>
11	Zoom Call	Test Plan	<ul style="list-style-type: none"> <li>• Explore test cases and possible hurdles with sponsor</li> <li>• Decide on the required code coverage aiming for</li> </ul>

## Team Qualifications

1. **Arya Shashwat:** Over 5 years of programming experience in C++ and python. Proficient in front-end frameworks like React Js and Bootstrap. Knowledge of Node Js for back-end.
2. **Koh Liang Jing:** Programmer with around 6 years of experience. Proficient in Java, C# and Python. Knowledge in mobile programming such as Android and in data analytics such as SAS, R, Python and Tableau.
3. **Roxas Abby Maurea Imus:** Proficient in C#, Java, C, HTML, CSS.

Knowledge in both iOS and Android development.

1. **Kiran Mac Milin:** Experienced in programming with C, C++, Java, Python. Proficient in Natural Language Processing techniques and Machine Learning (Pursued URECA based on NLP). Knowledge of Dart programming language to create mobile applications.
2. **Vaish Arjun:** Programming experience of around 4 years in both C++ and Python. Proficient in C, Visual Basic, Java, HTML, CSS and Javascript. Knowledge of Dart programming language to create mobile applications.
3. **Chandna Divvij:** Experienced programmer in Python, C++ and Java, with knowledge in C, HTML, CSS, JavaScript, SQL and Dart as well. Also proficient in using Firebase and Google Analytics.
4. **Asok Kumar Gaurav:** Proficient in Python, Java, C, C++, html, CSS, javascript, and SQL. Experienced in Flask Backend Framework. Knowledge of tensorflow, keras, OpenCV.

## Conclusion

Having established the importance of mental health and the need for a digital system that mimics the role of a professional counsellor, our project establishes a system to help with this issue. We would use emotion detection and Natural Language Processing to produce a superior chatbot that is able to have quality conversations without giving generic responses.

Our team would build a website to take in the detected emotion and provide a customized chatbot that suits the user. It would also feature some mini games to improve the user's mood.

## References

- [1] World Health Organization, "Mental Health: Strengthening our response," <https://www.who.int/en/news-room/fact-sheets/detail/mental-health-strengthening-our-response> (30 March 2018)
- [2] Josh Barkin, "When Bots Fail At Conversation," <https://medium.com/@joshbarkin/when-bots-fail-at-conversation-d7419605f5cc> (20 August 2016)
- [3] Channel News Asia, "COVID-19: Worries about pandemic see more calls to mental health helplines" <https://www.channelnewsasia.com/news/singapore/covid-19-fear-toll-mental-health-hotline-anxiety-singapore-12631710> (15 April 2020)

## Appendix A: Résumés of Team Members

### Shashwat Arya

+65 90873378 | [shashwat.arya5@gmail.com](mailto:shashwat.arya5@gmail.com)

<https://www.linkedin.com/in/shashwat-arya-149984171>

#### EDUCATION

Nanyang Technological University, Singapore	Aug 2018 – Jul 2022 (expected)
Bachelor of Engineering in Computer Science	
<ul style="list-style-type: none"> <li>• Relevant modules: <b>Business Finance, Data Science, AI and machine learning; Databases, Algorithms, Data Structures, Object-oriented programming</b></li> </ul>	
University of Wisconsin-Madison, United States	Jan 2020 – May 2020

#### Exchange Program

- Relevant modules: **Matrices and machine learning, Software Engineering, Computer Graphics**

#### WORK EXPERIENCE

Nanyang Technological University, Singapore	Aug 2019 – July 2020
Undergraduate Research Assistant	
<ul style="list-style-type: none"> <li>• Coordinated with my <b>professor</b> to build a <b>cross platform app</b> on <b>react-native</b> platform using <b>AWS</b></li> <li>• Aggregated several <b>major shopping platforms</b> to provide an all-purpose shopping app to customers</li> </ul>	
Centre for IT Services, Nanyang Technological University	2018 – 2019 (6 months)

#### Student Assistant

- **Collaborated** with other students and staff to work on different **technical projects**
- Created content and edited web pages utilizing **HTML** and **SharePoint** for NTU's website

#### PROJECTS AND CO-CURRICULARS ACTIVITIES

Data Science and Machine Learning project	Apr 2019
<ul style="list-style-type: none"> <li>• <b>Accurately</b> linked anomaly years to socio-economic conditions from past <b>120 years of Olympics dataset</b></li> <li>• Derived insights using <b>regression</b> and <b>anomaly detection</b> and implemented <b>matplotlib, seaborn</b> python libraries for visualizations</li> </ul>	
React-Native Run Tracking App	Jan 2020 – May 2020
<ul style="list-style-type: none"> <li>• Created a <b>cross platform app</b> for running and tracking your live run with stats</li> <li>• <b>Designed the UI</b> working in a <b>team of 6</b> with inputs from a client</li> <li>• Implemented <b>react-native</b> and <b>Node.js</b> for development and <b>Firebase</b> for database</li> </ul>	
Technical Director, IEEE student club, NTU	Aug 2019 – Dec 2019

- Organized an **All Singapore Hackathon** attended by **250 people**
- **Lead a team of 3** to organize technical workshops on **python, C++** for several hundred people

#### SKILLS AND AWARDS

Programming - Python, Java, C++, JavaScript, SQL, HTML, C
<ul style="list-style-type: none"> <li>• <b>Frameworks</b> - React, React-Native, Node.js, Sci-kit learn, Seaborn, Matplotlib</li> </ul>
<ul style="list-style-type: none"> <li>• <b>Rakuten Open Innovation Challenge</b> – Top 5</li> </ul>
<ul style="list-style-type: none"> <li>• <b>Languages spoken</b> – English(native), Hindi(native), German(intermediate)</li> </ul>

## Gaurav Asok Kumar

Email: Gauravasokkumar10@gmail.com  
 Phone: +65 98114906  
[www.linkedin.com/in/gaurav-asok-kumar-583672179/](https://www.linkedin.com/in/gaurav-asok-kumar-583672179/)

GitHub:[GAK1729](https://github.com/GAK1729)  
 LinkedIn:

### SKILLS / INTERESTS

Programming Skills: Python, Java, C++, C, HTML/CSS, JavaScript,  
 Deep Learning: Neural networks, Convolutional Neural networks, LSTM using Keras/Tensorflow  
 Python libraries: Pandas, scikit-learn, seaborn, matplotlib, Keras, Tensorflow,

### ACADEMIC PROJECTS

<u>Deep Learning Project</u>	Jun 2020 – Jul 2020
Image Classification in Kaggle FIRE Dataset	
<ul style="list-style-type: none"> <li>- <a href="https://www.kaggle.com/phylake1337/fire-dataset">https://www.kaggle.com/phylake1337/fire-dataset</a></li> <li>- Detecting presence of Fire in images, and classifying whether images have fire or not</li> <li>- Modelled Convolutional Neural Network with Accuracy of 97%</li> </ul>	
<u>Data Analytics</u>	Jun 2020 – Jul 2020
Shopee Code League COMPETITION: ORDER BRUSHING [Data Analytics]	
<ul style="list-style-type: none"> <li>- Participated in Shopee Code League Data Analytics Competition</li> <li>- Got 90% Accuracy, and shopee Score of 20/24</li> </ul>	
<u>Data Science Project</u>	Jan 2019 – Mar 2019
Visualizing Olympic history dataset (Group Project)	
<ul style="list-style-type: none"> <li>- Visualized and analysed the Olympics data set from Kaggle using python and pandas</li> <li>- Used Random forest to predict winners using age, height, and weight of athlete</li> </ul>	
<u>Deep Learning</u>	Jun 2020 – Jul 2020
Image classification using TensorFlow and Keras	
<ul style="list-style-type: none"> <li>- Participated in Shoppe Kaggle competition to identify and classify images</li> <li>- Performed data extraction and processing</li> <li>- Developed Convolutional Neural Network model using Keras and Tensorflow to extract key features from the image data set</li> </ul>	
<u>Software Engineering</u>	Jun 2020 – Jul 2020
Web development and Data analysis	
<ul style="list-style-type: none"> <li>- Developed Health Website that predicts the pollution levels on different regions in Singapore for the next 10 days</li> <li>- Website made using HTML, JavaScript ES6, and Flask Python backend</li> <li>- Demonstration of the project <a href="https://youtu.be/sVZNPE9-MYg">https://youtu.be/sVZNPE9-MYg</a></li> </ul>	

### Python Programming Aug 2018 – Nov 2018

NTU Campus canteen Mapping (Group Project CZ1003)	
<ul style="list-style-type: none"> <li>- A python program that recommends a canteen in NTU for user, based on user's meal preferences. User can input location, and view the shortest path from the user location to the selected canteen.</li> </ul>	

### Java Programming Jan 2019 – Mar 2019

Cinema UI (Group Project CZ2002)

Developed Java code for recommending and booking cinema tickets

### WORK EXPERIENCE

#### Birthassist Dec 2019 – Jan 2020

##### Software Engineering internship

- Interned at Birthassist – a Software company in Hyderabad, India
- Learned and Applied Flask Python to help in coding
- Website: <https://birthassist.in/>

**Divvij CHANDNA**

HP: +65 98975410 | Email: divvij001@e.ntu.edu.sg  
 (LinkedIn: <https://www.linkedin.com/in/divvij/>  
 GitHub: <https://github.com/divvijchandna>)

**EDUCATION****Nanyang Technological University, Singapore****Aug 2018 — Jun 2022 (Expected)**

- Bachelor of Engineering (Computer Science)
- Current CGPA: 4.14
- Expected Honours (Distinction)

**Vasant Valley School, New Delhi****Apr 2016 — May 2018**

- CBSE All India Senior School Certificate Examination (Science)
- Final Percentage: 94.4%

**WORK EXPERIENCE****Tech Mahindra Ltd., Noida, India****May 2019 — Jul 2019****Project Trainee, Intern**

- Investigated new and emerging software applications within the Blockchain industry to design, select, implement and use.
- Formulated a “Go to Market” plan for various use cases of blockchain by identifying the scope of the field and the current state.

**Bellurbis Technologies Pvt. Ltd., Gurgaon, India****Jun 2016 — Aug 2016****Android App Developer, Intern**

- Computed basic functions using Java to assist in the development of various Android projects under the guidance of the app development team.
- Learned the basics of Android Studio, a software used for Android app development.

**ACADEMIC PROJECTS****Nanyang Technological University, Singapore****Software Engineering Project****Mar 2020 — May 2020****Title: HDBFinder**

- Built an Android application using Flutter to search for HDBs and apartments in Singapore using a government database in a team of 5.
- Used firebase as a backend to store user details and HDB details.

**Software Systems Analysis and Design Project****Feb 2020 — May 2020****Title: Pizza Rush**

- Developed an educational game using Flutter on Android and iOS to help students study Mathematics in a team of 4.
- Built the frontend of the application and designed diagrams showing the system architecture.

**SKILLS / INTERESTS**

**Languages:** English (fluent), Hindi (fluent), Sanskrit (intermediate), German (beginner)

**Programming Skills:** C++, Python, C, Java, HTML, CSS, SQL, Bootstrap 4, Javascript, Dart (Flutter), Firebase, Google Analytics, Google Tag Manager

**Software Application Skills:** Microsoft Office 2019

**Online Courses:** Mathematics for Machine Learning: Linear Algebra (by Imperial College London on Coursera), IBM Blockchain Foundation for Developers (by IBM on Coursera), Front-End Web UI

**Frameworks and Tools:** Bootstrap 4 (by HKUST on Coursera), Google Analytics for Beginners (by Google), Google Tag Manager Fundamentals (by Google)

**Mac Milin KIRAN**

HP: +65 88728067 | Email: [macmilinkiran@gmail.com](mailto:macmilinkiran@gmail.com)  
[www.linkedin.com/in/macmilin](https://www.linkedin.com/in/macmilin)

**EDUCATION**

**Nanyang Technological University, Singapore** Aug 2018 – Jul 2022 (Expected)

- Bachelor of Engineering (Computer Science)
- Specialization: Cybersecurity & Artificial Intelligence
- Minor (Expected): Public Policy and Global Affairs

**ACADEMIC PROJECTS**

**Nanyang Technological University, Singapore**

**Undergraduate Research Program**

Sep 2019 – Jun 2020

**Title: Estimating the effects of Trump's Tweets: A Computational Approach**

- Analysed effects of Trump's tweets on traditional American media using Sentiment Analysis, Natural Language Processing, Vector Autoregression.
- Recognized by International Conference of Undergraduate Research (ICUR) to present abstract at the regional conference.

**NTU PEAK Leadership Program**

Mar 2020

**Title: Student Consulting with Changi Airport Group**

- Brainstormed innovative solutions with team of 5 to prevent Foreign Object & Debris Damage on Changi Airport Runway.
- Received one-on-one mentorship from company representatives for leadership development.
- Presented the case and solutions to a panel of top company executives.

**Module Projects**

**Title: HDB Finder**

Jan 2020 – Apr 2020

- Worked in a group of 5 to develop a mobile application using Flutter to search and get details of HDB rentals.
- Extracted and processed data from the government website and developed UI.

**Title: Regression Analysis for World Happiness Report**

Mar 2019 – Apr 2019

- Performed data analysis on factors affecting happiness score of countries using Python.
- Performed linear regression, ridge regression, lasso regression using k-fold cross-validation.

**WORK EXPERIENCE**

**Robert Bosch (South East Asia) Pte Ltd.**

**Corporate Research Intern**

Jun 2020 – Aug 2020

- Completed a study on Vector Autoregression to predict the output variables.
- Utilized isolation forest technique and z-score method to identify anomalies.
- Performed anomaly detection and imputed outlier values into the data to increase efficiency of Torque calculation & prediction in automated machinery & tools.

**Programming Skills:** Python, Java, C, C++, Dart, MS SQL Server, MySQL, Machine Learning

**Software Application Skills:** Microsoft Office 2019, Tableau

**Koh Liang Jing**  
8484 4627 | liangjing.gg@gmail.com  
[www.linkedin.com/in/kohliangjing](https://www.linkedin.com/in/kohliangjing)  
<https://github.com/liangjingga/>

---

<b>Education</b>	<b>Nanyang Technological University, Singapore</b> Bachelor of Computer Science AUGUST 2018 – MAY 2022 (expected)
	<b>Nanyang Polytechnic</b> Diploma in Business Intelligence and Analytics with Merit APRIL 2014 - MAY 2018 Activities and Societies: Student Union and Astronomy
<b>Work Experience</b>	<b>ST Electronics (Info-comm system)</b> Sharepoint Developer (Intern) SEPTEMBER 2017 - NOVEMBER 2017 (3 months) <ul style="list-style-type: none"><li>Enhanced company portal using SharePoint and to customize workflow forms</li><li>Designed the new company portal and move over the old forms to the new portal</li><li>Converted company manual forms to e-forms and other improvements</li><li>Gathered business requirements</li><li>Developed the workflow development of the e-forms</li></ul> IDE used: SharePoint Language used: C#, javascript
<b>IT Skills</b>	HTML, Javascript, Cascading Style Sheets, Java, PHP, Android Studio, Python, C#, SQL, Tableau, SAS, R Microsoft SharePoint, Microsoft Access

---

**ABBY MAUREA ROXAS**  
Permanent Resident | Phone: +65 9068 2219 |  
Email: abbymaur001@e.ntu.edu.sg

## EDUCATION

---

**Nanyang Technological University, Singapore** Aug 2018 – May 2022  
Bachelor of Engineering in Computer Science (Expected)

**Nanyang Polytechnic, Singapore** Apr 2015 – Mar 2018  
Diploma in Information Technology with Merit

## ACADEMIC PROJECTS (OPTIONAL)

---

**Nanyang Technological University, Singapore**

**Software Engineering** Aug 2019 – Dec 2019  
• Acquired knowledge on Software Development Life Cycle  
and built an Android application.

**Object-Oriented Design & Programming** Jan 2019 – May 2019  
• Developed a Java application with OOP Concept

**Nanyang Polytechnic, Singapore**

**Asian Culinary Institute Mobile Application** Nov 2017 – Feb 2018  
• Translated from Objective-C to Swift using XCode.

**Application Security and Project** Apr 2017 – Oct 2017  
• C#, ASP.NET and SQL to build an application that comprises hashing, encryption, administrative logging of activities, users' management, etc.

## SKILLS

---

**Digital Skills**

- Proficient in skillsets such as Java, C#, Swift, ASP.NET, SQL, HTML, CSS, Python used in Data Analysis
- Experienced in Microsoft Office products.
- Able to do basic design and photoshop.

**Languages**

- Proficient in English and Tagalog

**Arjun Vaish**  
**Email:** [arjun.vaish25@gmail.com](mailto:arjun.vaish25@gmail.com)  
 Phone: +65 83481255

## WORK EXPERIENCE

- **Information Technology Intern** June 2020- August 2020  
**DHL Express, Singapore**
  - Worked closely with the Head of Key Accounts to automate the processes for different departments in DHL.
  - Made use of Excel,VBA, Python to carry out the automation.
  - 2.13 Full-Time Equivalent (FTE) was saved during the duration of the internship post automation.
- **Undergraduate Student Researcher** August 2019- July 2020  
**Nanyang Technological University, Singapore**
  - Invitation-only programme to expose outstanding NTU undergraduates to pursue independent research under the supervision of a professor for an 11-month period.
  - Working under Asst. Prof. Arijit Khan on 'Bitcoin/Blockchain Network Mining'.

## EDUCATION

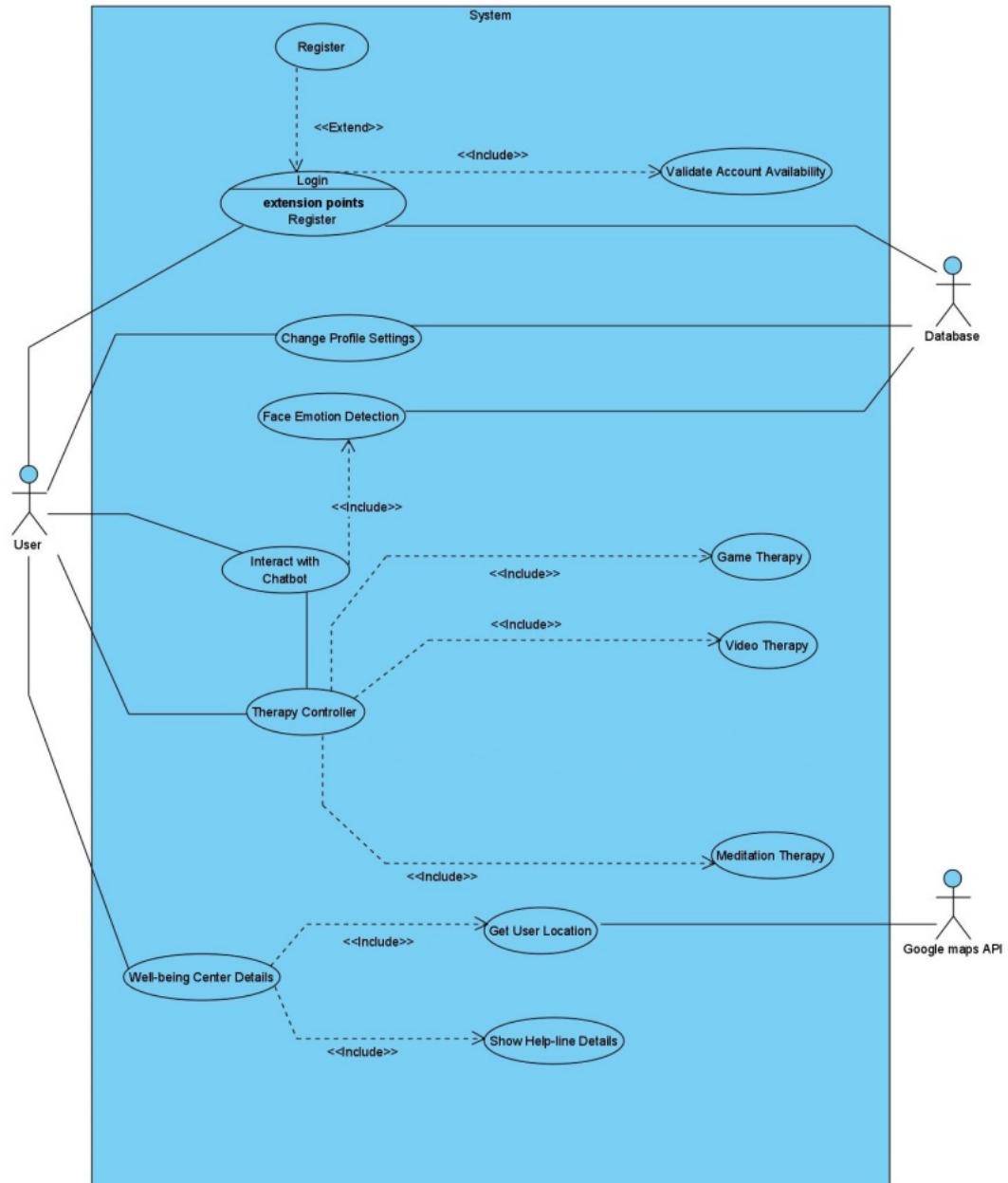
- **Nanyang Technological University, Singapore** Aug 2018- Jun 2022(Expected)
  - Bachelor of Engineering (Computer Science)
  - Minor in Business
  - CGPA: 4.51/5
  - Specialization: Data Science and Analytics, Artificial Intelligence

## ACADEMIC PROJECTS

- **Software Engineering Project** Jan 2020- May 2020  
**Title: Health Application**
  - Created a health application using HTML, CSS3, JavaScript giving details of pollutants level's (PSI,PM2.5 and UVI), dengue clusters in Singapore and COVID-19 cases throughout the world as part of a course group project.
  - Used Machine Learning to predict pollution levels of the next 14 days.
- **Software System Analysis and Design Project** Jan 2020- May 2020  
**Title: Quiz Application**
  - Created a quiz mobile application using Flutter as part of course group project.
  - Implemented leader board functionality which allowed players to compete with one another.
- **Object Oriented Design and Programming Project** Aug 2019- Dec 2019  
**Title; Cinema MOBLIMA Application**
  - Created a cinema UI in Java using Object Oriented Programming concepts as part of a course group project.
  - Implemented functionality of adding new movies and seats in various theatres across Singapore.

## Use Case Model

### Use Case Diagram



## Use Case Descriptions

Use Case ID:	1.0		
Use Case Name:	Sign In / Create Account		
Created By:	Vaish Arjun	Last Updated By:	
Date Created:	04/09/2020	Date Last Updated:	

Actor:	User
Description:	User is given the option to create an account. While creating the account, the account availability is checked and credentials are stored.
Preconditions:	<ul style="list-style-type: none"> <li>1. New user must be trying to access the app.</li> <li>2. User must have Wifi connected to his/her device.</li> <li>3. The application should be running on a web browser.</li> </ul>
Postconditions:	<ul style="list-style-type: none"> <li>1. Users can see their personalized dashboard.</li> <li>2. Users can interact with the chatbot which also makes use of facial recognition to detect emotions.</li> <li>3. Users can undertake various forms of therapy such as Video Game Therapy, Meditation and Sleep Therapy..</li> <li>4. Users can view the contact details for various well being centers in their vicinity.</li> <li>5. Users can update their personal particulars and profile details.</li> </ul>
Priority:	High

Frequency of Use:	1-4 Times
Flow of Events:	<ol style="list-style-type: none"> <li>1. User opens the web application.</li> <li>2. User is given the option to create an account on the homepage of the web app.</li> <li>3. User presses the respective icon.</li> <li>4. Users enter their personal particulars.</li> <li>5. The user account is created.</li> <li>6. The user views a personalized dashboard.</li> </ol>
Alternative Flows:	<ol style="list-style-type: none"> <li>1. The user who already has an account can login.</li> <li>2. The user can continue as a guest.</li> </ol>
Exceptions:	There is no network connection.
Includes:	Validate Account Availability
Special Requirements:	<ol style="list-style-type: none"> <li>1. The web application should be responsive.</li> <li>2. The Internet should be connected to the device.</li> </ol>
Assumptions:	User account is not created
Notes and Issues:	-

Use Case ID:	2.0
Use Case Name:	Validate Account Availability

Created By:	Vaish Arjun	Last Updated By:	
Date Created:	04/09/2020	Date Last Updated:	

Actor:	User
Description:	System can validate the user by sending email to the respective email address and complete creation of the account.
Preconditions:	<p>1. User is new to the application.</p> <p>2. User has WI-FI connected to his/her device.</p> <p>3. User must have entered all the details to create the account.</p>
Postconditions:	A code will be sent to the email address for verification.
Priority:	Medium
Frequency of Use:	1-4 Times
Flow of Events:	<p>1. User enters his personal particulars and health information to create the account.</p> <p>2. A mail is sent to the user containing a verification code.</p> <p>3. User can click on the link or enter the code to complete the verification.</p>
Alternative Flows:	-
Exceptions:	1. User enters the code which doesn't match. Then the user is sent back to step 1 with the error message "Please try again".

Includes:	-
Special Requirements:	The Internet should be connected to the device.
Assumptions:	-
Notes and Issues:	-

Use Case ID:	3.0		
Use Case Name:	Interact with the Chatbot		
Created By:	Vaish Arjun	Last Updated By:	
Date Created:	04/09/2020	Date Last Updated:	

Actor:	User
Description:	User is given the option to interact with the Chatbot. The Chatbot offers features such as facial recognition for emotion detection, and also various therapy options including Video Game, Meditation and Sleep therapy.
Preconditions:	<ol style="list-style-type: none"> <li>1. The web app should be open on a device connected to an internet.</li> <li>2. User selects the Chatbot Option.</li> <li>3. User allows camera to take photo for emotion analysis. .</li> <li>4. The application should be running on a web browser.</li> </ol>

Postconditions:	<ol style="list-style-type: none"> <li>1. Users can interact with the chatbot.</li> <li>2. Users get details about their mood based on their facial expression and suggestions to improve it further.</li> <li>3. Users get various Therapy options to choose from..</li> </ol>
Priority:	High
Frequency of Use:	0-15 Times
Flow of Events:	<ol style="list-style-type: none"> <li>1. User has the option to predict their emotions using facial recognition.</li> <li>2. The camera will take a photo of the person and use Neural Networks for emotion analysis.</li> <li>3. On the other hand, the user can choose from a wide range of therapies such as Video Game, Meditation and Sleep therapy..</li> </ol>
Alternative Flows:	Can view after sign in or continue as a guest
Exceptions:	There is no network connection.
Includes:	Face Emotion Detection
Special Requirements:	The Internet should be connected to the device.
Assumptions:	-
Notes and Issues:	-

Use Case ID:	4.0
--------------	-----

Use Case Name:	Display the dashboard.		
Created By:	Vaish Arjun	Last Updated By:	
Date Created:	05/09/2020	Date Last Updated:	

Actor:	User
Description:	Users can view the dashboard giving a snippet of the various functionalities offered by the web application. .
Preconditions:	1. The web app should be open on a device connected to an internet. 2. The application should be running on a web browser.
Postconditions:	The user gets to view a dashboard which is personalized if the user has an account. The user can see the features of the web app and the navigation bar.
Priority:	Medium
Frequency of Use:	0-4 Times
Flow of Events:	1. The user presses 'Sign In' or 'Continue as guest' on the homepage. 2. User views the dashboard with the navigation bar.
Alternative Flows:	The user creates an account and then signs in his/her account to view the personalized dashboard.
Exceptions:	-
Includes:	-

Special Requirements:	The internet should be connected to the device.
Assumptions:	-
Notes and Issues:	-

Use Case ID:	5.0		
Use Case Name:	Display Well Being Center Details		
Created By:	Vaish Arjun	Last Updated By:	
Date Created:	05/09/2020	Date Last Updated:	

Actor:	User
Description:	Users can access various wellbeing centres in Singapore on the basis of proximity from their current location. They can also get details of all Centres present in Singapore .
Preconditions:	<ol style="list-style-type: none"> <li>1. The web app should be open on a device connected to an internet.</li> <li>2. The user presses 'Details of Well Being Centers' in the navigation bar.</li> <li>3. The application should be running on a web browser.</li> </ol>
Postconditions:	<ol style="list-style-type: none"> <li>1. The user is able to view the details of wellbeing centres in Singapore .</li> </ol>

Priority:	Medium
Frequency of Use:	0-5 Times
Flow of Events:	<ol style="list-style-type: none"><li>1. The user continues as a guest or signs in to view the dashboard.</li><li>1. User clicks "Well Being Centers" in the navigation bar.</li><li>2. The user views the contact details and addresses of the centers .</li></ol>
Alternative Flows:	-
Exceptions:	-
Includes:	Get User Location, Show Help Line Details
Special Requirements:	The internet should be connected to the device.
Assumptions:	-
Notes and Issues:	-

Use Case ID:	6.0		
Use Case Name:	Display Therapy options		
Created By:	Kiran Mac Milin	Last Updated By:	
Date Created:	05/09/2020	Date Last Updated:	

Actor:	User
Description:	Users are able to view various therapy options like Meditation Therapy, Sleep Therapy and Video & Game Therapy. The system must rank the options using an algorithm to display the most appropriate form of therapy according to the user's input.
Preconditions:	<ol style="list-style-type: none"> <li>1. User must have Wifi connected to his/her device.</li> <li>2. Application must be running on web browser.</li> <li>3. User must have completed all the screening/ diagnostic questions with the chatbot.</li> </ol>
Postconditions:	<ol style="list-style-type: none"> <li>1. Users are able to view the different therapy options.</li> <li>2. The therapy options will be ranked by the system.</li> <li>3. The system will use an algorithm to rank which methods are most effective for the patient according to their conditions.</li> </ol>
Priority:	High
Frequency of Use:	1-4 Times

Flow of Events:	<ol style="list-style-type: none"> <li>1. User completes all the screening questions with the chatbot.</li> <li>2. User is given the options to choose their preferred method of lightening their mood with rankings by system.</li> <li>3. User presses the respective icon.</li> <li>4. User is taken to the respective therapy page.</li> </ol>
Alternative Flows:	<ol style="list-style-type: none"> <li>1. User can restart their chat with the chatbot to resubmit answers to the screening questions.</li> </ol>
Exceptions:	-
Includes:	Sleep therapy, Meditation Therapy, Video & Game therapy pages
Special Requirements:	<ol style="list-style-type: none"> <li>1. The Internet should be connected to the device.</li> </ol>
Assumptions:	-
Notes and Issues:	-

Use Case ID:	7.0		
Use Case Name:	Video & Game Therapy Page		
Created By:	Kiran Mac Milin	Last Updated By:	
Date Created:	05/09/2020	Date Last Updated:	

Actor:	User
Description:	Users are able to choose whether they would like video or game therapy. For video therapy, the user will be shown a series of calming and soothing videos so that the mind of the user is at ease. For game therapy, the system will link the users to some soothing & satisfying mini games to calm the mind.
Preconditions:	<ol style="list-style-type: none"> <li>1. User must have Wifi connected to his/her device.</li> <li>2. Application must be running on a web browser.</li> <li>3. User must have completed all the screening/ diagnostic questions with the chatbot.</li> <li>4. User must have chosen 'Video &amp; Game Therapy'.</li> </ol>
Postconditions:	<ol style="list-style-type: none"> <li>1. Users are able to view the option between Video &amp; Game therapy.</li> <li>2. On choosing video therapy, system must display a series of soothing videos.</li> <li>3. On choosing game therapy, system must allow the user to start the calming games.</li> </ol>
Priority:	High

Frequency of Use:	1-4 Times
Flow of Events:	<ol style="list-style-type: none"> <li>1. User completes all the screening questions with the chatbot.</li> <li>2. User chooses 'Video &amp; Game Therapy'.</li> <li>3. User presses the icon for video therapy.</li> <li>4. User is taken to the Videos page where there are a number of soothing videos for the user to view.</li> </ol>
Alternative Flows:	<ol style="list-style-type: none"> <li>1. User completes all the screening questions with the chatbot.</li> <li>2. User chooses 'Video &amp; Game Therapy'.</li> <li>3. User presses the icon for game therapy.</li> <li>4. User is taken to the games page where there are a number of soothing mini games for the user to play.</li> </ol>
Exceptions:	-
Includes:	Watch Embedded Videos, Play mini games
Special Requirements:	<ol style="list-style-type: none"> <li>1. The Internet should be connected to the device.</li> </ol>
Assumptions:	-
Notes and Issues:	-

Use Case ID:	8.0		
Use Case Name:	Meditation Therapy		
Created By:	Kiran Mac Milin	Last Updated By:	
Date Created:	05/09/2020	Date Last Updated:	

Actor:	User
Description:	Once the user chooses meditation therapy, the system takes the user to a page consisting of a series of guided meditation therapy. The system will start giving instructions by displaying the text and providing a voice over. The meditation will also be timed. Once the user completes a part of the meditation technique, they can click 'Proceed' to go to the next page. The system decides on a 'level' of meditation according to what the chatbot has diagnosed. The recommended meditation level is shown; the user can also switch to a different level based on their comfort.
Preconditions:	<ol style="list-style-type: none"> <li>1. User must have Wifi connected to his/her device.</li> <li>2. Application must be running on a web browser.</li> <li>3. User must have completed all the screening/ diagnostic questions with the chatbot.</li> <li>4. User must have chosen 'Meditation Therapy'.</li> </ol>
Postconditions:	<ol style="list-style-type: none"> <li>1. Users are able to view the page to start Meditation Therapy.</li> </ol>
Priority:	High
Frequency of Use:	1-4 Times

Flow of Events:	<ol style="list-style-type: none"><li>1. User completes all the screening questions with the chatbot.</li><li>2. User chooses 'Meditation Therapy'.</li><li>3. User presses the 'Start' button to begin.</li><li>4. User is taken through the meditation process step-by-step.</li></ol>
Alternative Flows:	-
Exceptions:	-
Includes:	-
Special Requirements:	<ol style="list-style-type: none"><li>1. The Internet should be connected to the device.</li></ol>
Assumptions:	-
Notes and Issues:	-

Use Case ID:	9.0		
Use Case Name:	Sleep Therapy		
Created By:	Kiran Mac Milin	Last Updated By:	
Date Created:	05/09/2020	Date Last Updated:	

Actor:	User
Description:	
Preconditions:	<ul style="list-style-type: none"> <li>5. User must have Wifi connected to his/her device.</li> <li>6. Application must be running on a web browser.</li> <li>7. User must have completed all the screening/ diagnostic questions with the chatbot.</li> <li>8. User must have chosen 'Sleep Therapy'.</li> </ul>
Postconditions:	<ul style="list-style-type: none"> <li>2. Users are able to view the page to start Sleep Therapy.</li> </ul>
Priority:	High
Frequency of Use:	1-4 Times
Flow of Events:	<ul style="list-style-type: none"> <li>1. User completes all the screening questions with the chatbot.</li> <li>2. User chooses 'Sleep Therapy'.</li> <li>3. User presses the 'Start' button to begin.</li> </ul>

Alternative Flows:	-
Exceptions:	-
Includes:	-
Special Requirements:	1. The Internet should be connected to the device.
Assumptions:	-
Notes and Issues:	-

Use Case ID:	10.0		
Use Case Name:	Face Emotion Detection		
Created By:	Kiran Mac Milin	Last Updated By:	
Date Created:	05/09/2020	Date Last Updated:	

Actor:	User
Description:	The user must take a picture so that the system can scan their mood. This is implemented through Facial Emotion Recognition.
Preconditions:	<ol style="list-style-type: none"> <li>1. User must have Wifi connected to his/her device.</li> <li>2. Application must be running on a web browser.</li> <li>3. User must have logged in successfully.</li> <li>4. User must have clicked on 'Start Session'.</li> </ol>
Postconditions:	<ol style="list-style-type: none"> <li>1. System will have identified the user's mood.</li> <li>2. Chatbot will continue to ask questions to gauge how the user is feeling.</li> </ol>
Priority:	High
Frequency of Use:	1-4 Times
Flow of Events:	<ol style="list-style-type: none"> <li>1. User logs in successfully.</li> <li>2. User chooses 'Start Session'.</li> <li>3. User presses the 'Scan to show mood' button.</li> </ol>

Alternative Flows:	-
Exceptions:	-
Includes:	-
Special Requirements:	1. The Internet should be connected to the device.
Assumptions:	-
Notes and Issues:	-

## Backlog

### Meeting 1

Task Description	Team Member Name:	Team Leader	Member 1	Member 2	Member 3	Member 4	Member 5	Member 6	Average Task Priority	Average Task Difficulty	Average Task Deadline (days)	Task Assignee	Task Helpers	Task Allocation Method	Has the task been completed successfully?	Actual Days used to Complete the Task	Assignee's Mood after Completing this Task	Average Quality F
member's mood during the this week's meeting:		Excited - Excited - Excited - Excited - Excited - Excited -							-	-	-	-		-	-	-	-	
Project Proposal (After Task Completed)			-	9 -	-	-	-	8 -	-	-	-	Arjun	-	Shashwat, Arjun, Mac, Liang Jing, Abby	Through Discussion	Yes	6	Happy ▼ 8.5
Estimated Task Priority (0-10)			-	10 -	-	-	-	8 -	-	-	-							
Estimated Task Difficulty (0-10)			-	7 -	-	-	-	5 -	-	-	-							
Estimated Time Required for the Task (1-14 Days)			-	6 -	-	-	-	4 -	-	-	-							
Team Member's Confidence about the Task (0-10)			-	10 -	-	-	-	7 -	-	-	-							
Task Quality Evaluation (After Task Completed)			-	-	-	10 -	9 -	-	-	-	-	Arjun	-	Divvij, Gaurav	Through Discussion	Yes	5	Happy ▼ 9.5
Use Case Diagram (After Task Completed)			-	-	-	5 -	5 -	-	-	-	-							
Estimated Task Priority (0-10)			-	-	-	5 -	5 -	-	-	-	-							
Estimated Task Difficulty (0-10)			-	-	-	3 -	3 -	-	-	-	-							
Estimated Time Required for the Task (1-14 Days)			-	-	-	3 -	3 -	-	-	-	-							
Team Member's Confidence about the Task (0-10)			-	-	-	10 -	10 -	-	-	-	-							
Task Quality Evaluation (After Task Completed)			9 -	10 -	-	-	-	-	-	-	-	Arjun	-	Arjun, Mac	Through Discussion	Yes	5	Excited ▲ 9.5
Use Case Description (After Task Completed)			5 -	5 -	-	-	-	-	-	-	-							
Estimated Task Priority (0-10)			5 -	5 -	-	-	-	-	-	-	-							
Estimated Task Difficulty (0-10)			5 -	5 -	-	-	-	-	-	-	-							
Estimated Time Required for the Task (1-14 Days)			3 -	3 -	-	-	-	-	-	-	-							
Team Member's Confidence about the Task (0-10)			10 -	10 -	-	-	-	-	-	-	-							
Backlog	Task Quality Evaluation		-	-	-	-	-	-	-	-	-	Arjun	-	Abby, Liang Jing	Through Discussion	Yes	1	Happy ▼ 5.5
Estimated Task Priority (0-10)			-	-	-	-	-	-	-	-	-							
Estimated Task Difficulty (0-10)			-	-	-	-	-	-	-	-	-							
Estimated Time Required for the Task (1-14 Days)			-	-	-	-	-	-	-	-	-							
Team Member's Confidence about the Task (0-10)			-	-	-	-	-	-	-	-	-							
Project Discussion	Task Quality Evaluation		9 -	8 -	9 -	9 -	8 -	9 -	9 -	9 -	9 -	Arjun	-	Arjun, Divvij, Gaurav, Liang Jing, Mac, S	Through Discussion	Yes	2	Happy ▼ 8.71428
Estimated Task Priority (0-10)			10 -	10 -	10 -	10 -	10 -	10 -	10 -	10 -	10 -							
Estimated Task Difficulty (0-10)			8 -	8 -	8 -	8 -	8 -	8 -	8 -	8 -	8 -							
Estimated Time Required for the Task (1-14 Days)			1 -	1 -	1 -	1 -	1 -	1 -	1 -	1 -	1 -							
Team Member's Confidence about the Task (0-10)			10 -	10 -	10 -	10 -	10 -	10 -	10 -	10 -	10 -							

### Meeting 2

Task No.	Task Description	Team Member Name:	Team Leader	Member 1	Member 2	Member 3	Member 4	Member 5	Member 6	Average Task Priority	Average Task Difficulty	Average Task Deadline (day)	Task Assignee	Task Helpers	Task Allocation Method	Has the task been completed successfully?	Actual Days used to Complete the Task	Assignee's Mood after Completing this Task	Avg Quality		
			Team member's mood during the this week's meeting:	Excited	Excited	Excited	Excited	Excited	Happy	Excited	Happy	Excited	-	-	-	-	-	-			
1	Quality Plan	Team Member's Confidence about the Task (0-10)	Task Quality Evaluation (After Task Completed)	10 -	9 -	9 -	10 -	9 -	10 -	10 -	9 -	10 -	9.571428571	6.571428571	5.857142857	Arjun	- q, Gaurav, Mac, Abby, LiangJing, Divyij	Yes	6	Excited	9.5714
			Estimated Task Priority (0-10)	10 -	9 -	9 -	10 -	9 -	10 -	10 -	9 -	10 -	-	-	-	-	-	-			
			Estimated Task Difficulty (0-10)	9 -	8 -	8 -	9 -	7 -	8 -	9 -	8 -	9 -	-	-	-	-	-	-			
			Estimated Time Required for the Task (1-14 Days)	7 -	5 -	5 -	5 -	6 -	6 -	6 -	5 -	6 -	-	-	-	-	-	-			
2	Software Requirements	Team Member's Confidence about the Task (0-10)	Task Quality Evaluation (After Task Completed)	10 -	9 -	10 -	10 -	9 -	10 -	10 -	9 -	10 -	9.571428571	7.714285714	6.285714286	Arjun	- q, Gaurav, Mac, Abby, LiangJing, Divyij	Yes	6	Excited	9.7142
			Estimated Task Priority (0-10)	10 -	9 -	10 -	10 -	9 -	10 -	10 -	9 -	10 -	-	-	-	-	-	-			
			Estimated Task Difficulty (0-10)	9 -	8 -	8 -	9 -	7 -	8 -	9 -	8 -	9 -	-	-	-	-	-	-			
			Estimated Time Required for the Task (1-14 Days)	7 -	6 -	6 -	6 -	7 -	7 -	8 -	6 -	7 -	-	-	-	-	-	-			

## Meeting 3

Task No.	Task Description	Team Member Name:	Team Leader	Member 1	Member 2	Member 3	Member 4	Member 5	Member 6	Average Task Priority	Average Task Difficulty	Average Task Deadline (days)	Task Assignee	Task Helpers	Task Allocation Method	Has the task been completed successfully?	Actual Days used to Complete the Task	Assignee's Mood after Completing this Task	Avg Quality			
			Team member's mood during the this week's meeting:	Excited	Excited	Excited	Happy	Excited	Happy	Excited	Happy	Excited	-	-	-	-	-	-				
1	Project Plan	Team Member's Confidence about the Task (0-10)	Task Quality Evaluation (After Task Completed)	10 -	9 -	9 -	10 -	9 -	10 -	10 -	9 -	10 -	9.428571429	7.671428571	6.367142856	Arjun	- q, Gaurav, Mac, Abby, LiangJing, Divyij	Yes	7	Excited	9.8671	
			Estimated Task Priority (0-10)	10 -	9 -	9 -	10 -	9 -	10 -	10 -	9 -	10 -	-	-	-	-	-	-				
			Estimated Task Difficulty (0-10)	9 -	8 -	8 -	9 -	7 -	8 -	9 -	8 -	9 -	-	-	-	-	-	-				
			Estimated Time Required for the Task (1-14 Days)	7 -	5 -	5 -	5 -	6 -	6 -	6 -	5 -	6 -	-	-	-	-	-	-				
2	Risk Management Plan	Team Member's Confidence about the Task (0-10)	Task Quality Evaluation (After Task Completed)	10 -	9 -	10 -	10 -	9 -	10 -	10 -	9 -	10 -	9.285714286	7.142857143	6.587142857	Arjun	- q, Gaurav, Mac, Abby, LiangJing, Divyij	Yes	7	Excited	10	
			Estimated Task Priority (0-10)	10 -	9 -	9 -	10 -	9 -	10 -	10 -	9 -	10 -	-	-	-	-	-	-				
			Estimated Task Difficulty (0-10)	9 -	8 -	8 -	9 -	7 -	8 -	9 -	8 -	9 -	-	-	-	-	-	-				
			Estimated Time Required for the Task (1-14 Days)	7 -	6 -	6 -	5 -	7 -	8 -	7 -	6 -	7 -	-	-	-	-	-	-				
3	Charter	Team Member's Confidence about the Task (0-10)	Task Quality Evaluation (After Task Completed)	9 -	8 -	10 -	10 -	9 -	10 -	10 -	9 -	10 -	9.666666667	9	9	Arjun	- Divyij, Mac	rough Discuss	Yes	7	Excited	10
			Estimated Task Priority (0-10)	8 -	7 -	9 -	10 -	8 -	10 -	10 -	9 -	10 -	-	-	-	-	-	-				
			Estimated Task Difficulty (0-10)	7 -	6 -	8 -	9 -	5 -	7 -	8 -	7 -	8 -	-	-	-	-	-	-				
			Estimated Time Required for the Task (1-14 Days)	5 -	4 -	6 -	7 -	3 -	5 -	6 -	5 -	6 -	-	-	-	-	-	-				
4	Initial Requirements	Team Member's Confidence about the Task (0-10)	Task Quality Evaluation (After Task Completed)	10 -	9 -	10 -	10 -	9 -	10 -	10 -	9 -	10 -	9	9.5	9	Arjun	- Arjun	rough Discuss	Yes	7	Excited	10
			Estimated Task Priority (0-10)	10 -	9 -	9 -	10 -	9 -	10 -	10 -	9 -	10 -	-	-	-	-	-	-				
			Estimated Task Difficulty (0-10)	9 -	8 -	8 -	9 -	7 -	8 -	9 -	8 -	9 -	-	-	-	-	-	-				
			Estimated Time Required for the Task (1-14 Days)	7 -	6 -	6 -	5 -	7 -	8 -	7 -	6 -	7 -	-	-	-	-	-	-				
5	Backend	Team Member's Confidence about the Task (0-10)	Task Quality Evaluation (After Task Completed)	9 -	8 -	9 -	10 -	9 -	10 -	10 -	9 -	10 -	9	8.8	4.8	Arjun	- Abby, LiangJing	rough Discuss	Yes	7	Excited	10
			Estimated Task Priority (0-10)	8 -	7 -	8 -	9 -	6 -	7 -	8 -	7 -	8 -	-	-	-	-	-	-				
			Estimated Task Difficulty (0-10)	7 -	6 -	7 -	8 -	5 -	6 -	7 -	6 -	7 -	-	-	-	-	-	-				
			Estimated Time Required for the Task (1-14 Days)	5 -	4 -	5 -	6 -	3 -	4 -	5 -	4 -	5 -	-	-	-	-	-	-				
6	Frontend	Team Member's Confidence about the Task (0-10)	Task Quality Evaluation (After Task Completed)	10 -	9 -	10 -	10 -	9 -	10 -	10 -	9 -	10 -	9	10	9	Arjun	- Gaurav	rough Discuss	Yes	7	Excited	10
			Estimated Task Priority (0-10)	10 -	9 -	9 -	10 -	9 -	10 -	10 -	9 -	10 -	-	-	-	-	-	-				
			Estimated Task Difficulty (0-10)	9 -	8 -	8 -	9 -	7 -	8 -	9 -	8 -	9 -	-	-	-	-	-	-				
			Estimated Time Required for the Task (1-14 Days)	7 -	6 -	6 -	7 -	5 -	6 -	7 -	6 -	7 -	-	-	-	-	-	-				
7	Release plan	Team Member's Confidence about the Task (0-10)	Task Quality Evaluation (After Task Completed)	10 -	9 -	10 -	10 -	9 -	10 -	10 -	9 -	10 -	9	10	8	Arjun	- Shadow	rough Discuss	Yes	10	Excited	9
			Estimated Task Priority (0-10)	10 -	9 -	9 -	10 -	9 -	10 -	10 -	9 -	10 -	-	-	-	-	-	-				
			Estimated Task Difficulty (0-10)	9 -	8 -	8 -	9 -	7 -	8 -	9 -	8 -	9 -	-	-	-	-	-	-				
			Estimated Time Required for the Task (1-14 Days)	7 -	6 -	6 -	7 -	5 -	6 -	7 -	6 -	7 -	-	-	-	-	-	-				

## Meeting 4

Task No.	Task Description	Team Member Name:	Team Leader	Member 1	Member 2	Member 3	Member 4	Member 5	Member 6	Average Task Priority	Average Task Difficulty	Average Task Deadline (days)	Task Assignee	Task Helpers	Task Allocation Method	Has the task been completed successfully?	Actual Days used to Complete the Task	Assignee's Mood after Completing this Task	Avg Quality				
			Team member's mood during the this week's meeting:	Excited	Excited	Excited	Excited	Excited	Happy	Excited	-	-	-	-	-	-	-	-					
1	Test Plan	Team Member's Confidence about the Task (0-10)	Task Quality Evaluation (After Task Completed)	10 -	9 -	9 -	10 -	9 -	10 -	10 -	9 -	10 -	9	7.714285714	6.367142856	Arjun	- q, Gaurav, Mac, Abby, LiangJing, Divyij	Yes	8	Excited	10		
			Estimated Task Priority (0-10)	10 -	9 -	9 -	10 -	9 -	10 -	10 -	9 -	10 -	-	-	-	-	-	-					
			Estimated Task Difficulty (0-10)	9 -	8 -	8 -	9 -	7 -	8 -	9 -	8 -	9 -	-	-	-	-	-	-					
			Estimated Time Required for the Task (1-14 Days)	7 -	6 -	6 -	7 -	5 -	6 -	7 -	6 -	7 -	-	-	-	-	-	-					
2	Test Cases and Requirements	Team Member's Confidence about the Task (0-10)	Task Quality Evaluation (After Task Completed)	10 -	9 -	10 -	10 -	9 -	10 -	10 -	9 -	10 -	9	8	7.871428571	6.367142856	Arjun	- q, Gaurav, Mac, Abby, LiangJing, Divyij	Yes	6	Excited	10	
			Estimated Task Priority (0-10)	10 -	9 -	9 -	10 -	9 -	10 -	10 -	9 -	10 -	-	-	-	-	-	-					
			Estimated Task Difficulty (0-10)	9 -	8 -	8 -	9 -	7 -	8 -	9 -	8 -	9 -	-	-	-	-	-	-					
			Estimated Time Required for the Task (1-14 Days)	7 -	6 -	6 -	7 -	5 -	6 -	7 -	6 -	7 -	-	-	-	-	-	-					
3	CMMI level 2 definition	Team Member's Confidence about the Task (0-10)	Task Quality Evaluation (After Task Completed)	10 -	9 -	9 -	10 -	9 -	10 -	10 -	9 -	10 -	9	8.714285714	8.285714286	6.587142857	Arjun	- q, Gaurav, Mac, Abby, LiangJing, Divyij	Yes	6	Excited	9.4285	
			Estimated Task Priority (0-10)	10 -	9 -	9 -	10 -	9 -	10 -	10 -	9 -	10 -	-	-	-	-	-	-					
			Estimated Task Difficulty (0-10)	9 -	8 -	8 -	9 -	7 -	8 -	9 -	8 -	9 -	-	-	-	-	-	-					
			Estimated Time Required for the Task (1-14 Days)	7 -	6 -	6 -	7 -	5 -	6 -	7 -	6 -	7 -	-	-	-	-	-	-					
4	Presentation slides	Team Member's Confidence about the Task (0-10)	Task Quality Evaluation (After Task Completed)	10 -	9 -	9 -	10 -	9 -	10 -	10 -	9 -	10 -	9	9.333333333	7.166666667	6.333333333	Arjun	- Mac	rough Discuss	Yes	6	Excited	9.8333
			Estimated Task Priority (0-10)	10 -	9 -	9 -	10 -	9 -	10 -	10 -	9 -	10 -	-	-	-	-	-	-					
			Estimated Task Difficulty (0-10)	9 -	8 -	8 -	9 -	7 -	8 -	9 -	8 -	9 -	-	-	-	-	-	-					
			Estimated Time Required for the Task (1-14 Days)	7 -	6 -	6 -	7 -	5 -	6 -	7 -	6 -	7 -	-	-	-	-	-	-					

Months into the coronavirus pandemic with no current end in sight, many people find themselves experiencing various degrees of mental health deterioration for a myriad of different reasons, such as fears of catching the virus, retrenchment due to economic loss caused by the pandemic, and the inability to go about regular social activities that were often used as a form of escapism. Not only are people undergoing anxiety and vulnerability attributed to the pandemic, we are also still faced with regular elements for stress, such as academics, jobs and relationships.

With everything that is going on in the world, mental health might be put on the back burner as reaching out for professional help may be unattainable due to cost, lockdown restrictions, or general inability to open up your feelings. Hence, there is an urgent need to bring mental health care solutions in the palms of people and it is through our proposed mental wellness application called MindSpace.

## Overall Description

In essence, MindSpace is a website which provides an interface for people to interact intuitively and smoothly with a chatbot to help improve mental wellness. MindSpace will also provide an interface to detect a user's emotional state. The chat-bot will be powered by Machine Learning together with Natural Language Processing(NLP) to ensure that the chatbot is as human-like as possible as it can seem.

The full-stack of the website will be developed using React, Bootstrap, Flask and NodeJS. The technologies used for detecting emotions and facial recognition would be OpenCV and NLP will be using NLP-NLTK.

## Investigation & Analysis Methodology

### System Investigation

The AI facial detection system detects the user's emotion by recognising the human emotion displayed by the user. Hence, afterwards the details of the user emotion will be passed to the AI chatbot. If the system is unable to detect the emotion due to camera permission issues, the user is also able to manually input their current emotion. The AI chatbot will then move on to the therapy session where user's input is also processed along the way to detect their ongoing emotions, which will help direct the AI on how to continue the therapy session. Only the emotions before and after the therapy session will be recorded, and the content of the session will not be recorded so that we can maintain the user's privacy. User's emotions will be recorded in our database where our system will be able to record the user's progress and suggest the appropriate therapy accordingly.

### Analysis Methodology

#### Feasibility study and requirements elicitation

Organize a requirement elicitation team to watch a therapist do their job, noting down on what information the therapist asks for when targeting a certain emotion. In addition, the team should examine the documents on how each emotion should be tackled and on existing competing products. Lastly, the team would arrange current therapist customers to test our prototype. A Feasibility and Risk Assessment study will be conducted to determine which solution(s) are most appropriate based upon the feedback of the session.

#### System analysis and requirements specification

##### Perform an analysis of the problem using object-oriented techniques

An external view of the enterprise model of the mindscape system including user registration, validation of account, changing of profile settings and the desired features of the system will be developed using Unified Modeling Language (UML). This System Requirement Specifications documents will form part of the documentation for the project. Some desired features of the new system includes: The ability to recognise user's facial emotion, to interact with the chatbot for therapy, to provide video therapy, game therapy and meditation therapy session.

#### Scope and Limitations

**Requirement analysis** - The scope of the requirement analysis includes system I/O description, user requirement definition, functional and security requirement. The limitations of the requirement analysis includes having the system require a registered account for the user to access the functions available. Additionally, the information required to create an account includes only username and password. Where the other fields asking for additional user's details are deemed optional, as some of the users might be afraid that their identity will be discovered. Hence to protect their identity, they have the options to not fill in their personal information. The functional requirement of the system includes creating an account, logging in to a previously registered account, interacting with the chatbot, providing video, game and meditation therapy.

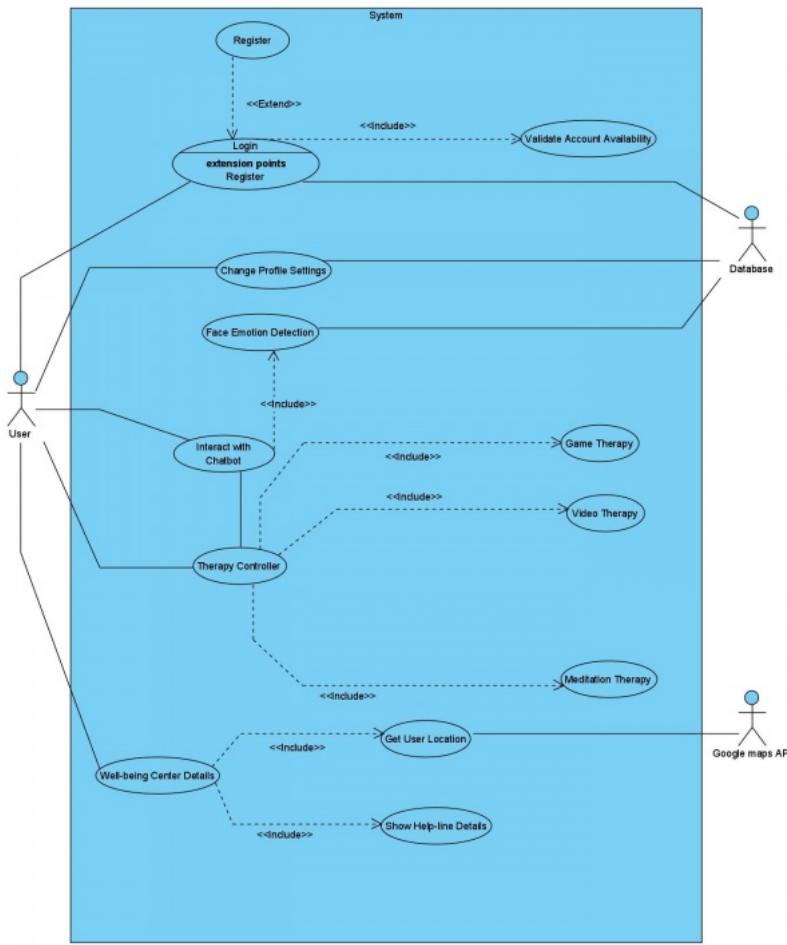
**Data analysis** - The scope of data analysis includes data collection process, data validation, data storage, manipulation and retrieval. Privacy of data should be the most important requirement when dealing with data. It must be impossible to save the chat content so users will feel safe when chatting with our chatbot, hence they would be more willing to use the system as we are able to guarantee their privacy. Exposing their vulnerable emotions is already hard for them, privacy would be their topmost priority when using an application like this. Data collected should be only necessary information for the therapy session to work (example, their emotion before and after the therapy) but it should not collect data that is considered sensitive to the user (example, their chat content with the AI chatbot).

**Process analysis** - The scope of process analysis includes data/process flow analysis, process decomposition and system interfaces. The system should be only accessible to users that are registered on our application. Registered users should be able to enter our system through the login page. When a user enters the system they are able to choose between; interacting with the chatbot, choosing their therapy needed, changing their profile settings or viewing their nearest wellbeing centre details. If the user chooses to interact with the chatbot, it will start asking if the user allows the camera to detect their facial expression. If not, they can manually input their emotions as well.

**Web application architecture** - The scope of web application architecture includes the web application information structure, usability, user interface design, interaction and application implementation. The user interface should use colours that trigger calm and relaxing emotions that will help the user in progress of their therapy. The user interface should be user friendly and easy to navigate around.

#### Object-oriented design using UML

A detailed object-oriented design for the system will be developed. UML will be used again for the graphical representation and documentation of the design.



## Prototyping

The Lead Developer and Front-End Developer will create a prototype based on the requirements gathered and a quick design that is agreed upon among the team. The prototype will be a working example of the system, to be used for demonstration and proof of concept purposes only. It will include all the interfaces that end-users will come across. Hence, the prototype will be presented to potential users, getting reviews and suggestions on the interface. Afterwards, the Lead Developer and Front-End Developer are able to use the reviews and suggestions to refine the prototype accordingly, creating a prototype that will satisfy the user's requirements. So that during the development of the product, there will not be additional time and resources spent on changing the interface of the system.

## Constraints

### Scalability

Flask which is being used to design the backend is capable of being scaled to incorporate a large number of servers. However our team would be running a live server on our PC, hence the total number of users would be limited to the system's processing power. If the system needs to handle more users, servers would have to be set up or other cloud services would be needed.

### Open Source

All the frontend, backend and machine learning libraries being used are open source. Although this saves on our development expenses, open-source software is less reliable than proprietary software.

### Development Costs

Since the entire project is built on open-source library and tools, the development cost would be negligible, presuming no unforeseen hurdles or additional costs.

### Project Schedule

The project would take 11 weeks to complete including a prototype and production build. The timeline includes weekly interaction with the sponsor to discuss progress and future plans.

## Operational Requirements

### Help Desk Support

System users have a 24x7 access to email assistance for questions that are technical in nature, such as, slow or sluggish system response time, incompatible browser features, application errors, system downtime inquiries, account lock-out assistance, etc.

### Application Services and Technical support

Programmers and application developers will have access to source code to address bugs or system enhancements as deemed necessary. Network Administrator and DBA support is also required to maintain a 24x7 system uptime.

## **Administration Features**

System security and access levels are provided in the online system. There are varying levels of system access and functional authority. Each user's access is limited to his/her own records. Only authorized system administrator(s) have access to all user's records.

### **Chatbot System independent of Emotion Detection**

The chatbot system will remain operational and its functionality will be complementary but independent from the user emotion detection functionality. At any one time, students may use either the chatbot system or the emotion detection feature only, but not both.

### **System hardware fail over and routine back up**

Computer operations center will handle system hardware tasks such as data tape back-up, hardware maintenance, fail over, scheduled system patches and maintenance.

### **Audit Trail**

System audit trails are an inherent part of all user registrations. Among others, the interaction and the respective action taken with each user is captured.

## **Functional Requirements**

The AI Chatbot system is "self-service style" system that address the user's wellbeing needs.

### **AI Chatbot System for Mental Wellbeing**

The user can access the application to chat with a chatbot for improving their mental wellbeing. All system (browser) interfaces are based on ISO accepted industry standards for the WWW. Among others the online chatbot system will have the following functionalities:

#### **Registration/ Login**

1. The database must validate the user upon login.
2. Email will be sent to validate the user's account on first-time registration.
3. Subsequent logins will be validated on entering a password.

#### **Profile Settings**

1. Change User Name
2. User Authentication/Change Password
3. Change Email Address

#### **Chatbot Interaction**

1. Determine User's Mood
  1. Using Natural Language Processing Techniques, text analytics
  2. Facial Emotion Recognition
2. Therapy Recommendation
  1. System will recommend and rank different therapy methods
  2. Therapy methods include:
    1. Video Therapy
    2. Games Therapy
    3. Guided Meditation Therapy

#### **Wellness/Wellbeing Centres**

1. Get user's location
2. Recommend nearby mental wellness centres
3. Provide a list of hotlines

## **Input Requirements**

### **User identifier key and user access**

Each user is assigned a unique username upon registration. The user must know this. This identifying key maps to all his/her record information in the system.

### **Facial Recognition**

Each user must show the chatbot how they are feeling. The system will take a picture and analyze the facial expressions to determine their mood.

## **Process Requirements**

The following are among the inherent requirements that the web application 'MindSpace' must be able to handle.

### **Performance**

1. Time taken for the application to load must be less than 3 seconds.
2. The application must be able to support 1000 concurrent users at least.

### **Responsiveness**

1. The average response time to any state transition made by the user must not be more than 0.5 seconds.
2. The maximum response time between click and reaction must be not more than 1 seconds.

## Usability

1. 90% of the users must be able to understand the application interface and use it on their first try.
2. Dialog boxes and messages must be sent, received and displayed to assist the overall user experience.

## Data Validation

1. Data error from the user's end and from the back-end database-processing end must be gracefully handled.
2. Data validation and error-handling routines must be a part of the application.

## Data Security

1. Personal user data must only be collected, used or disclosed with the individual's knowledge and consent.
2. User's location must only be tracked with their permission and when they're active on the application.

## Data Repository

1. The application will use Flask as the back-end with an SQLite database as the main repository of data.

## Output Requirements

### Confidentiality and Privacy

Just like going for physical therapy or counselling sessions where doctors and medical professionals are bound by patient confidentiality, protecting the privacy of our users is paramount. Conversations with the chatbot to ease mental stress will not be recorded and saved, it will be discarded immediately after the session finished. Users can also be rest assured that no data will be saved if they use our face recognition and emotion detection feature.

## Hardware Requirements

### Network

Wired and wireless network infrastructure is required for the web application to load.

### User Computers

Users can use either Mac, Unix or Windows computers with a working network to enter the web application.

## Software Requirements

### Client Operating System

PC operating system: -

- MAC OS
- Windows
- UNIX

Mobile operating system: -

- iOS
- Android

### Client Application

Client can access the website on any modern web browser that supports JavaScript. Most common ones are: -

- Chrome 85.0 (release: 08-09-2020)
- Firefox 80.0 (release: 10-08-2020)
- Safari 14.0 (release: 16-09-2020)
- Edge 85.0 (release: 15-01-2020)

### Network System

Network protocols required for communication: -

- HTTP
- FTP
- TCP/IP

### Software Tools and Libraries

Web technologies: -

- React Js - frontend design
- Flask - backend
- Node.js - backend

NLP and machine learning: -

- OpenCV - face recognition
- NLTK - NLP

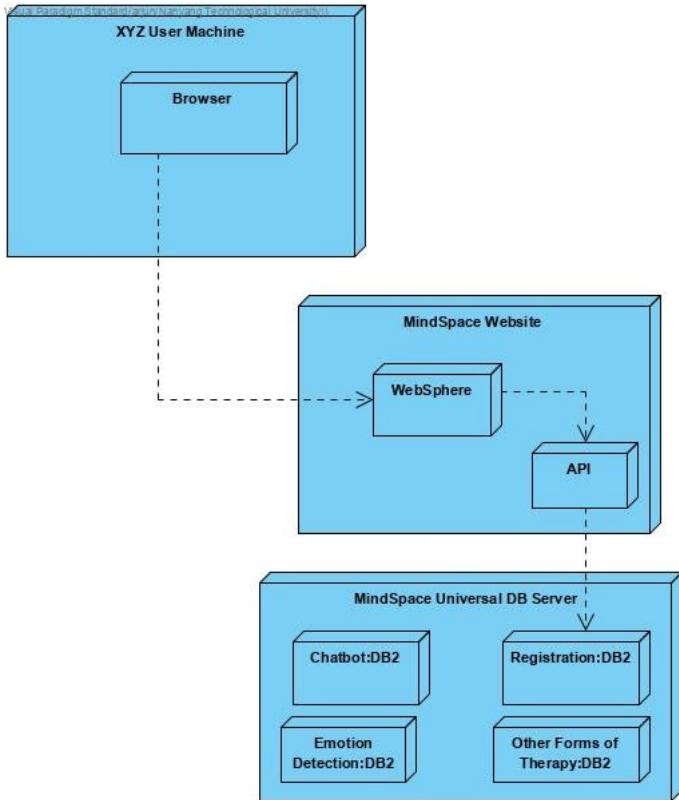
## Backend Server

The backend server supported by Flask and Node.js would be locally hosted on a PC. Since no mainframe server or cloud services are being used, the entire processing burden falls on the PC. This could be a possible bottleneck if the chosen PC does not have the adequate processing power.

## Testing

Testing would be done using the Node library Jest. The code must have at least 85% code coverage to deduce errors. This would be accomplished by running parallel unit tests.

## Deployment Requirements



## Quality Plan

### Signature Page

Prepared by: Whole Team Date: 20/09/2020

Reviewed by 1: Kiran Mac Milin Date: 21/09/2020

Reviewed by 2: Kiran Mac Milin Date: 22/09/2020

Approved by : Vaish Arjun Date: 24/09/2020

### Document Change Record

Revision	Description of Change	Approved By	Date
1.0	Initial Template	Arjun	23/09/2020
2.0	Add Purpose and Scope, Problem Reporting and Corrective Action	Arjun	25/09/2020
3.0	Add Reference Documents, Tools Techniques and Methodologies	Arjun	29/09/2020
4.0	Add Management, Media Control	Arjun	30/09/2020
5.0	Add Supplier Control, Documents	Arjun	1/10/2020
6.0	Add Standards, Practices, Conventions and Metrics, Record Collection, Maintenance, and Retention	Arjun	5/10/2020
7.0	Add Software Reviews, Training	Arjun	8/10/2020
8.0	Add Risk Management , SQA Plan Change Procedure and History	Arjun	11/10/2020

## Purpose and Scope

### Purpose

The purpose of this Software Quality Assurance (SQA) Plan is to establish the goals, processes, and responsibilities required to implement effective quality assurance functions for the MindSpace project. The Software Quality Assurance Plan provides the framework necessary to ensure a consistent approach to software quality assurance throughout the project life cycle. It defines the approach that will be used by the QAM and Software Quality (SQ) personnel to monitor and assess software development processes and products to provide objective insight into the maturity and quality of the software. The systematic monitoring of products, processes, and services will be evaluated to ensure they meet requirements and comply with policies, standards, and procedures, and ISO standards.

## Scope

The purpose of SQA is to ensure that the software developed does not deviate from the original intended product. SQA is also concerned to identify any errors, omissions, inconsistencies, and alternatives, enhancements or improvements that can be made at any stage of development.

Our intended product, MindScape which will provide users a place to communicate with a human-like chatbot to ease mental health stresses and help recognize their own emotions through an emotion detection feature in our project.

We intend to only use open-source libraries to develop our product, including the tools that will allow us to develop the chatbot and the emotions recognition, which are Natural Language Toolkit and OpenCV respectively.

## Reference Documents

- IEEE STD 730-2002, IEEE Standard for Software Quality Assurance Plans ([http://standards.ieee.org/reading/ieee/std\\_public/description/se/730-2002\\_desc.html](http://standards.ieee.org/reading/ieee/std_public/description/se/730-2002_desc.html))
- ISO IEC 90003:2004 Software Standard (<http://praxiom.com/iso-90003.htm>)
- Project Proposal
- Use Case Description
- System Requirement Specifications

## Management

This section describes the management organizational structure, its roles and responsibilities, and the software quality tasks to be performed.

### Management Organisation

The Quality Assurance Manager (QAM) would take charge of the implementation of the quality assurance system.

### Project Management

The project manager would approve the following: -

- Approving the system requirement specification document
- The overall time scale for the project
- Choosing the software development methodology and lifecycle
- Choosing the software development techniques and tools used
- Allocation of roles within the team

### Assurance Management

The QAM provides Project Management with visibility into the processes being used by the software development teams and the quality of the products being built. The QAM is independence from the project and the software developers.

## Tasks

This section summarizes the tasks (product and process assessments) to be performed during the development of software. These have been identified based on the project plan and inputs from the sponsor.

### Product Assessments

The following product assessments will be conducted by SQ personnel: -

- Website user interface
- Chatbot responses
- Facial emotion detection

### Process Assessments

The following process assessments will be conducted by SQ personnel: -

- Robust connection of the website with the backend
- High accuracy of machine learning models

## Roles and Responsibilities

This section describes the roles and responsibilities of each assurance person assigned to the Project.

### QAM

The QAM's responsibilities include, but are not limited to: -

- Ensure the project team has the required tools and resources
- Guide the SQ team and conduct SQ assessments and activities regularly
- Resolve any SQ development issues/concerns timely to maintain quality
- Resolve any concerns from the sponsor

## **Software Quality Personnel**

Responsibilities include, but are not limited to:

- Develop and maintain the project software quality assurance plan
- Assist the Quality Assurance Manager in conducting SQ activities
- Help resolve any development issues/risks

## **Documents**

### **Purpose**

This section identifies the minimum documentation governing the requirements, development, verification, validation, and maintenance of software that falls within the scope of this software quality plan. Each document below shall be assessed (reviewed) by SQ personnel.

### **Minimum Document Requirements**

- Project Proposal - Use Case Model - System requirement specification - Risk Management - Project Plan - Release Plan - Test Plan - Configuration Management Plan
- Change management plan - Release Plan

## **Standards, Practices, Conventions and Metrics**

### **Purpose**

This section highlights the standards, practices, quality requirements, and metrics to be applied to ensure a successful software quality program.

### **Software Quality Program**

These practices and conventions are tools used to ensure a consistent approach to software quality for all programs/projects.

### **Functionality**

It is an essential purpose of any service or product that is why the product is designed.

### **Maintainability**

It addresses the capability to identify and fix a fault in a software component. It is also known as supportability.

### **Reliability**

This characteristic defines the capability of the system to maintain its service levels under defined conditions for defined periods of time after software system is put in service as per its specifications.

### **Usability**

It is the ease of use for a given function. It is important to evaluate how simple it is for users to utilize the application.

## **Standard Metrics**

The following standard metrics are the minimum planned metrics that will be collected, reported, and maintained in the area of software quality assurance:

1. Effectiveness of Exception handling
2. Program Size
3. No. of Error Messages
4. Length of user manual
5. Cyclomatic Complexity
6. Class Coupling and Cohesion

## **Software Reviews**

### **Purpose**

This section identifies the number and type of system/subsystem reviews and engineering peer reviews that will be supported by the SQ Personnel. The project milestone chart, and the SQ Personnel resource levels determine the reviews that are supported.

### **Minimum Software Reviews**

For each review, SQ will assess the review products to assure that review packages are being developed according to the specified criteria, the review content is complete, accurate, and of sufficient detail, and Requests for Action are captured, reviewed, and tracked to closure. In addition, SQ will assess the processes used to conduct the reviews to determine if appropriate personnel are in attendance, correct information is presented, entry and exit criteria are met, and appropriate documents are identified for update.

The following software reviews will be assessed by SQ:

- Project Plan Review
- Software Requirements Specification Review
- Software Design Document Review
- Code Review
- Test Plan Review
- Acceptance Review

## Test

SQ personnel will assure that the test management processes and products are being implemented per Test Plan. This includes all types of testing of software system components as described in the test plan, specifically during integration testing (verification) and acceptance testing (validation). SQ personnel will monitor testing efforts to assure that test schedules are adhered to and maintained to reflect an accurate progression of the testing activities. SQ will assure that tests are conducted using approved test procedures and appropriate test tools, and that test anomalies are identified, documented, addressed, and tracked to closure. In addition, SQ will assure that assumptions, constraints, and test results are accurately recorded to substantiate the requirements verification/validation status. SQ personnel will review post-test execution related artifacts including test reports, test results, problem reports, updated requirements verification matrices, etc. SQ will ensure that the user feedback on the chatbot is collected in regular intervals. Suggestions from the user feedback will be taken seriously, in order to improvise the performance quality of the chatbot. The game and video section will be updated regularly. Moreover, the SQ will ensure regular performance consultation by mediation experts to ensure the quality of the meditation section is maintained.

## Problem Reporting and Corrective Action

SQ personnel generate, track, and trend assessment findings and observations in our centralized documents for Reporting and Corrective Action System. The location for the documents to report and correct is located in our shared Google Drive specifically for this project.

We will have meetings at the end of every product and process reviews to decide as a team on the most appropriate corrective actions, if any part of our product or process is not up to standards it will be documented in the centralized document with written specification as to when the problem was discovered as well as the corrective actions to be taken to fix the problems. The time taken for corrective issues will be taken into an account to update the whole project schedule to ensure that we do not have a schedule slip.

## Tools, Techniques and Methodologies

SQ personnel will require access to the following:

### Software Quality Tools

- Microsoft Office tools (i.e., Word, Excel, and PowerPoint)
- Visual Studio - React JS
- Visual Studio- flask
- Unity

### Media Control

The SQ deliverables would be documented in Microsoft Word as well as Google Docs for team collaboration. The team would be granted server space on Nanyang Technological University's Wiki site to store and review all the deliverables. See Section 12 for additional details on the collection and retention of key records.

### Supplier Control

[Not applicable for this project]

## Record Collection, Maintenance, and Retention

SQ personnel will maintain records that document assessments performed on the project. Maintaining these records will provide objective evidence and traceability of assessments performed throughout the project's life cycle. SQ personnel will maintain electronic copies of all assessment reports and findings. SQ Project folders will contain the assessment work products such as completed checklists, supporting objective evidence, and notes. The table below identifies the record types that will be collected, as well as the Record Custodian and Retention period:

**Record Collection Table**

Record Title	Record Custodian	Record Retention
SQA Assessments	SQ Personnel	6 Months
SQA Checklists	SQ Personnel	6 Months
Deliverable Defects	SQ Personnel	6 Months

## Training

SQ personnel have fundamental knowledge in the following areas through prior experience, training, or certification in methodologies, processes, and standards:

- Audits and Reviews (Assessments)
- Risk Management
- Software Assurance
- Configuration Management
- Software Engineering
- ISO 9001, ISO 9000-3
- CMMI
- Verification and Validation

## Risk Management

SQ personnel will assess the project's risk management process and participate in monthly risk management meetings and report any software risks to the QAM and the project manager.

Possible risks

Technology Risk

- Poor Accuracy of the NLP model

- Poor Accuracy of Face Recognition model
- Slow performance of Face Recognition model
- unable to integrate front end and back end
- Software break down with multiple users

#### Usability Risk

- UI not easily learnable by the users
- Chat bot not able to provide the required solution to user

#### Estimation Risk

- Wrong estimation to time to complete project

## SQA Plan Change Procedure and History

SQ personnel are responsible for the maintenance of this plan. It is expected that this plan will be updated throughout the life cycle to reflect any changes in support levels and SQ activities. Proposed changes shall be submitted to the Quality Assurance Manager (QAM), along with supportive material justifying the proposed change.

## Project Plan

### Revision History

Version	Implemented By	Revision Date	Approved By	Approval Date	Reason
1.0	Abby	22/09/2020	Arjun	23/09/2020	Added Introduction, Risk Management
2.0	Arjun	22/09/2020	Arjun	24/09/2020	Added Schedule
2.1	Arjun	23/09/2020	Arjun	24/09/2020	Added Work Packages
3.0	Divvij	24/09/2020	Arjun	25/09/2020	Added Best Practice Checklist
4.0	Gaurav	24/09/2020	Arjun	26/09/2020	Added Product Checklist
5.0	Liang Jing	25/09/2020	Arjun	26/09/2020	Added Project Organization , Quality Assurance
5.1	Liang Jing	26/09/2020	Arjun	27/09/2020	Added Team Communication
6.0	Mac	26/09/2020	Arjun	27/09/2020	Added Process Definition ,Monitoring & Control
7.0	Shashwat	27/09/2020	Arjun	28/09/2020	Added Project Estimates
7.1	Shashwat	28/09/2020	Arjun	28/09/2020	Added Cost Estimates

## Introduction

### Project Overview

MindSpace is a website that helps users to improve on their mental wellness by allowing them to interact with an AI chatbot that is able to respond intuitively and smoothly as if they were in a physical counselling session. Given an access to the user's camera function on their computer or where they have accessed the website, MindSpace will also be able to detect their emotional state and mood as an output to the user's camera feed.

### Project Description and Scope

The basis of MindSpace is to help alleviate the stress felt by the users through chatting with an AI agent which is trained to respond as human-like as possible through Natural Processing Language. Especially extraordinary times like this when a pandemic is occurring, stress is at an all-time high due to certain uncertainties in our lives such as job stability and our fear for health. Spending money to reach out for help will be put in the backburner, hence there is a need to make mental health solutions accessible to everyone.

The user can pour their feelings into the chatbot and it will process the user's input and derive the meaning of the input, in return the agent will respond accordingly.

Apart from the human-like responses from the AI chatbot, the website will also be able to recognize the user's mental state and mood with a facial input from the user.

However, the AI chatbot must not be treated as a replacement for professional help as it will not provide any sort of diagnosis as it merely helps to suggest therapy methods to alleviate stress. If stress and mental health deteriorate, the user can find hotlines from relevant agencies in MindSpace as well as Mental Wellness Centres closest to them.

The system shall include all necessary user interfaces to smoothly and intuitively determine the user's state and moods based on facial recognition and emotion detection and textual inputs in the chatbot.

## Project Organization

### Team Structure

The following is the list of executive roles, as required by CMM level 3.

- Project Manager: Vaish Arjun
- QA Manager: Kiran Mac Milin
- QA Engineer: Roxas Abby Maurea Imus
- Lead Developer: Asok Kumar Gaurav

- Front-End Developer: Arya Shashwat
- Back-End Developer: Chandna Divvij
- Release Engineer/Manager: Koh Liang Jing

## Roles & Responsibilities

### Project Manager: Vaish Arjun

- leads the planning
- handle the division of workload
- oversee the project progress
- management of budget

### QA Manager: Kiran Mac Milin

- in charge of the overall product and quality processes
- implement appropriate QA processes

### QA Engineer: Roxas Abby Maurea Imus

- devise and conduct the test plans
- ensure that the product is up to standards
- ensure that the product will meet the customer's expectation

### Lead Developer: Asok Kumar Gaurav

- responsible for overall technical components of the project

### Front-End Developer: Arya Shashwat

- responsible for implementing the visual components of the product

### Back-End Developer: Chandna Divvij

- in charge of the architecture of the product
- handle the database of the product

### Release Engineer/Manager: Koh Liang Jing

- create baselines
- integrate changes for delivery
- manage the releases of the product

## Team Communication

Our communication channels include the following:

- Bi-weekly face to face meetings are held on every alternate Thursdays.
- Group announcements and updates are sent through WhatsApp
- While Zoom discussions are held weekly, either on Sundays/Mondays.
- Split up into subgroups as necessary, in order to work more co-operatively on specific problems.

## Process Definition

Minspace intends to use the **Scrum Agile Framework** methodology throughout the AI Chatbot system. Scrum is an **agile software development framework** centred around moving projects forward via short-term blocks of work called sprints, which are usually confined to two-week intervals. Teams working with this framework are self-organizing and not top-down or hierarchical in nature.

The approach produces ongoing release cycles, each featuring small, incremental changes from the previous release. At each iteration, the product is tested. The Agile model helps teams identify and address small issues on projects before they evolve into more significant problems, and engage stakeholders and get their feedback throughout the development process.

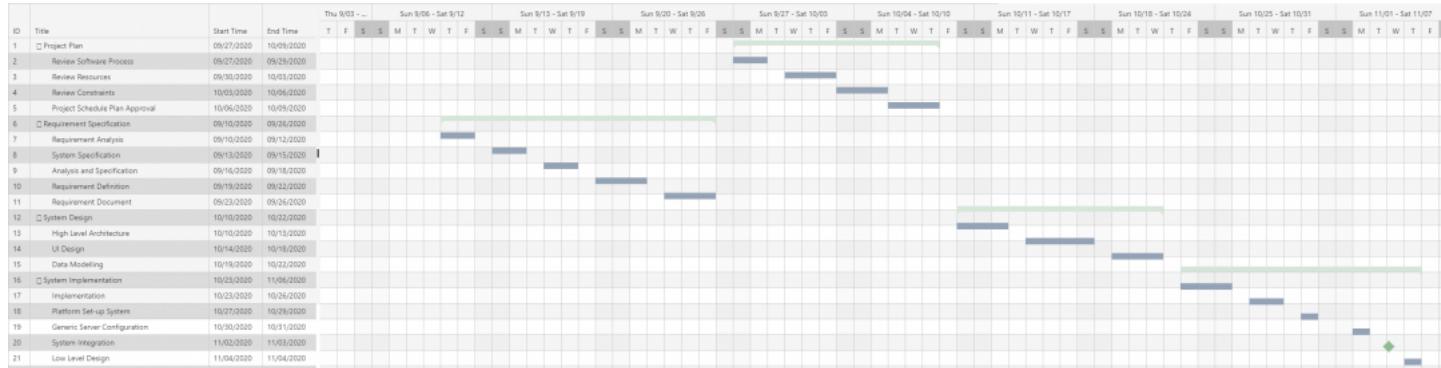
The scrum framework is heuristic; it's based on continuous learning and adjustment to fluctuating factors. It acknowledges that the team doesn't know everything at the start of a project and will evolve through experience. Scrum is structured to help teams naturally adapt to changing conditions and user requirements, with re-prioritization built into the process and short release cycles so your team can constantly learn and improve.

This methodology is more flexible than the traditional Waterfall SDLC due to repeated iterations involving design, coding, unit testing, integration, and quality assurance. The Waterfall SDLC is not a viable choice due to the short timeline available for AI Chatbot project to reach delivery quality.

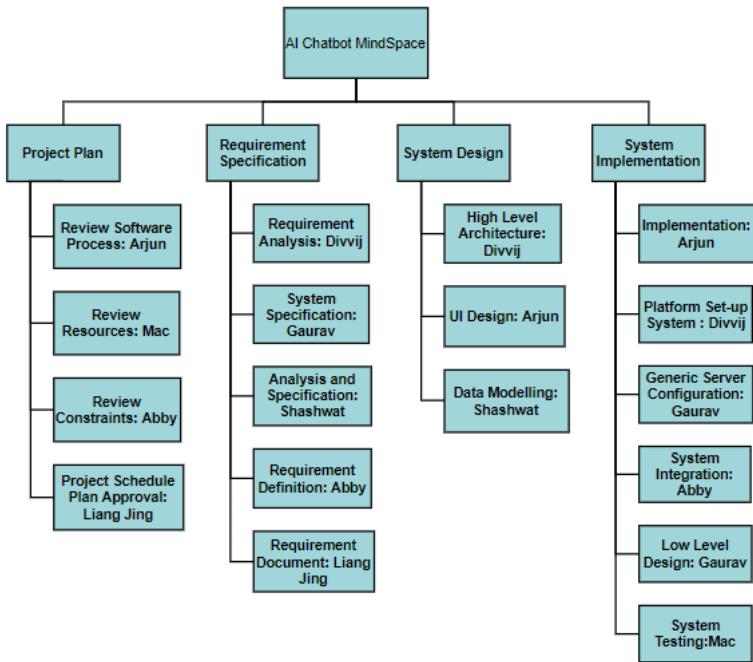
Minspace has chosen to avoid such methodologies as Spiral because of concerns over the short timeline. Should design procedures, for example, need to be revisited within the first release date, it is likely that the project will overshoot its critical schedule. Minspace intends to deliver the first iteration of the functionality on the System Delivery date indicated in the Estimations section of this document. After further interactions with stakeholders, further iterations should occur as necessary.

## Schedule

### Activity Dependencies and Schedule



## Work Breakdown Structure



## Work Packages

The entire project work is broken down by the important phases of the software development life cycle. They include the following:

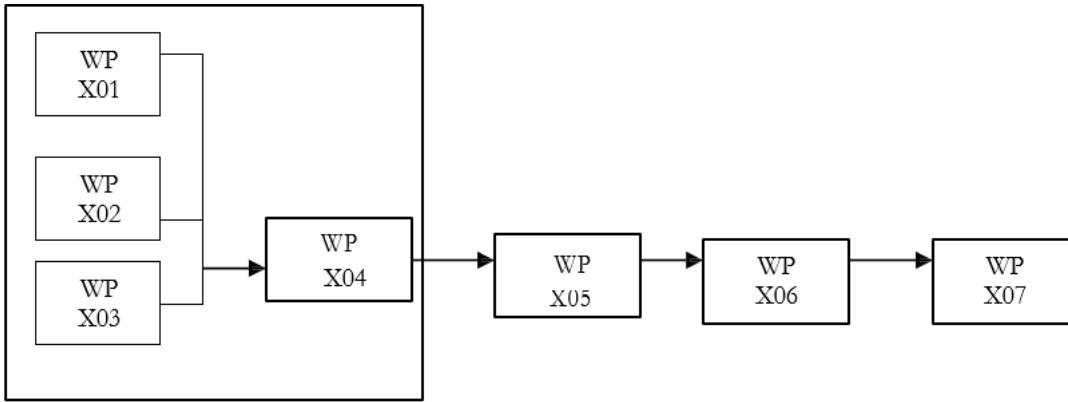
1. Project Plan
2. Requirement Specification
3. User Interface
4. Technical Architecture
5. Data Modeling
6. Coding & Unit Testing
7. Integration & Quality Assurance

## Activity Dependencies

The following table describes the dependencies of the deliverable work packages:

Work Package Number	Work Package Description	Duration	Dependencies
X01	Work Package	7 Days	--
X02	Requirement Specification	7 Days	--
X03	User Interface	7 Days	--
X04	Technical Architecture	12.1 Days	X01,X02,X03
X05	Data Modeling	7 Days	X04
X06	Coding & Unit Testing	16.2 Days	X05
X07	Integration & System Testing	16.2 Days	X06

The following Activity Network Diagram describes the above in more graphical detail:



Note that work package X05 is dependent on all work packages encapsulated by the larger boxes linked to its left. For instance, WP X05 may not start until WP X01-X04 has been finished.

## Work Package Details

Work packages are listed below. A team member, indicated in bold, has been assigned as primarily responsible for each work package and will coordinate that package

<b>Project</b>	AI Chatbot Mindspace System
<b>Work Package</b>	X01— Project Plan (1 of 7)
<b>Assigned To</b>	Vaish Arjun, Chandna Divvij, Asok Kumar Gaurav, Kiran Mac Milin, Arya Shashwat, Roxas Abby Maurea Imus, Koh Liang Jing
<b>Effort</b>	7PD
<b>Start Date</b>	Sunday, 27/09/20
<b>Purpose</b>	To determine an introductory overview of the project, to be refined in later work packages.
<b>Inputs</b>	None
<b>Activities</b>	This work package includes providing a brief overview of the project, its objectives, and a set of proposed project deliverables throughout the development of the software cycle. The people responsible for this work package will also be transcribing ideas brought up in the group meeting discussion into a formal report.
<b>Outputs</b>	A written document of the Project Plan Introduction.

<b>Project</b>	AI Chatbot Mindspace System
<b>Work Package</b>	X02— Requirement Specification (2 of 7)
<b>Assigned To</b>	Kiran Mac Milin, Koh Liang Jing, Roxas Abby Maurea Imus, Vaish Arjun, Arya Shashwat, Asok Kumar Gaurav, Chandna Divvij
<b>Effort</b>	7PD
<b>Start Date</b>	Thursday, 10/09/20
<b>Purpose</b>	To establish a common understanding between the customer and the software project team of the customers' requirements to be addressed by the project
<b>Inputs</b>	Customer's requirements
<b>Activities</b>	Identify “the customer”, interview customer, write and inspect customer requirement and build requirements.
<b>Outputs</b>	A written document of the requirement specification.

<b>Project</b>	AI Chatbot Mindspace System
<b>Work Package</b>	X03— User Interface (3 of 7)
<b>Assigned To</b>	<b>Roxas Abby Maurea Imus</b> , Arya Shashwat, Chandna Divvij, Vaish Arjun, Asok Kumar Gaurav, Koh Liang Jing, Kiran Mac Milin
<b>Effort</b>	7PD
<b>Start Date</b>	Thursday, 16/10/20
<b>Purpose</b>	To build the user interface between the system and the customer, to make it easy use, and friendly to the customer
<b>Inputs</b>	User information
<b>Activities</b>	To get the user information, user request, display the dialog between system and user, display the result of request
<b>Outputs</b>	User Interface

<b>Project</b>	AI Chatbot Mindspace System
<b>Work Package</b>	X04— Technical Architecture (4 of 7)
<b>Assigned To</b>	<b>Koh Liang Jing</b> , Arya Shashwat, Vaish Arjun, Chandna Divvij, Asok Kumar Gaurav, Roxas Abby Maurea Imus, Kiran Mac Milin
<b>Effort</b>	12.1PD
<b>Start Date</b>	Thursday, 18/10/20
<b>Purpose</b>	To do the <u>high level</u> architecture design
<b>Inputs</b>	Project Plan Work Packages (X01 to X03 inclusive).
<b>Activities</b>	High level design entails defining the architecture of the software system and identifying the various components and how they are inter-related to and interactive with each other. Designers also need to decide on the software and hardware infrastructures, such as what operating system on which the software is built, the language used to implement the software, and so on. Design topics including maintainability, portability, and reusability will be addressed here as well.
<b>Outputs</b>	High Level Design and Architectural Specification.

<b>Project</b>	AI Chatbot Mindspace System
<b>Work Package</b>	X05— Data Modeling (5 of 7)
<b>Assigned To</b>	<b>Arya Shashwat</b> , Kiran Mac Milin, Chandna Divvij, Asok Kumar Gaurav, Koh Liang Jing, Roxas Abby Maurea Imus, Vaish Arjun
<b>Effort</b>	7PD
<b>Start Date</b>	Saturday, 20/10/20
<b>Purpose</b>	To build the project's database
<b>Inputs</b>	Project Plan Work Packages (X01 to X05 inclusive).
<b>Activities</b>	Analyze the data flow relationships, entity relationships
<b>Outputs</b>	A written document of the data modeling

<b>Project</b>	AI Chatbot Mindspace System
<b>Work Package</b>	X06— Coding & Unit testing (6 of 7)
<b>Assigned To</b>	<b>Chandna Divvij, Arya Shashwat, Kiran Mac Milin, Asok Kumar Gaurav, Koh Liang Jing, Roxas Abby Maurea Imus, Vaish Arjun</b>
<b>Effort</b>	16.2PD
<b>Start Date</b>	Sunday, 25/10/20
<b>Purpose</b>	To implement the system as per the requirements specification and other associated documents. This work package includes such additional activities as preliminary unit testing.
<b>Inputs</b>	Project Plan Work Package X06.
<b>Activities</b>	Programmers will implement the modules according to the design specifications noted in the Specification document.
<b>Outputs</b>	Source code and header files

<b>Project</b>	AI Chatbot Mindspace System
<b>Work Package</b>	X07— Integration & System Testing (7 of 7)
<b>Assigned To</b>	<b>Asok Kumar Gaurav , Vaish Arjun ,Arya Shashwat, Kiran Mac Milin, Chandna Divvij, , Koh Liang Jing, Roxas Abby Maurea Imus</b>
<b>Effort</b>	16.2PD
<b>Start Date</b>	Tuesday 05/11/20
<b>Purpose</b>	To identify and fix logical and syntactical errors produced during the implementation of the <u>System</u> , and setting up drivers and stubs to see how the module responds to various inputs. Black box testing as well as white box testing might be conducted to check for logical errors. All the testing procedures will be documented in the Test Plan report. If problems are found, they will be noted and fixed at the earliest possible time.
<b>Inputs</b>	Project Plan Work Package X07.
<b>Activities</b>	The Integration testing team may try to simulate how a user might interact with the system. Similar to Unit Testing, Integration Testing may require the development of stubs and drivers as well, but here this is more geared towards the higher (overall system) level. Testers may also examine issues such as system performance and integrity. Heuristics assessment plays an important role in this work package, as intelligence components will define eventual system success.
<b>Outputs</b>	A test report.

## Project Estimates

### Code Size Estimation using Function Points

Unadjusted function points are calculated based on the parameters described below. This is used to calculate the code size to estimate the adjusted function points.

#### Unadjusted Function Points

The functionalities of MindSpace chatbot are summarized below:

User:

- Login/register an account on the website
- Emotion detection via computer vision
- Able to chat with a therapeutic chatbot customized to mood
- Play interactive games to boost morale
- Watch mood boosting meditation videos

Backend:

- Emotion detection trained model
- NLP model to automate replies
- Login verification

The measure of unadjusted function points is based on five primary component elements of these functions: Inputs, Outputs, Inquiries, Logical Files, and Interfaces. Each element ranges from Low Complexity, Medium Complexity to High Complexity. The detailed evaluation of the complexity is as follows:

Inputs:

- User login/registration: The user would enter a username and password to login or register an account (for new user) and the credentials would be stored in a database.
- Facial image: Input a photo through the webcam before every user to get an analysis of the user's mood and train the chatbot accordingly.
- Messages to the chatbot: While interacting with the provided chatbot, the user would have to supply the responses in text form to converse.
- Game input: While playing the mood boosting games, input is needed when required.

## Outputs:

- Emotion prediction: Based on the image supplied by the user, the face recognition model would predict the current emotional state.
- Chatbot replies: The NLP model would analyze the message typed by the user and generate an appropriate response to be sent back.

## Inquiries:

- Allow the user to browse different types of meditation video
- Allow the user to browse mini games

## Logical Files:

- User authentication: A database of user credentials referenced during attempts of login or new user registration.
- NLP word bank: Collection of responses/interlinked words used to train the NLP model.

## Interfaces:

- Backend connection: Connecting the backend (Flask) to the frontend (React JS) through their APIs.
- Authentication verification: Accessing the user credential database to verify login/registration.

## Summary of above analysis:

Element	Complexity	Detail			
Inputs	Low	User login/registration			
	Medium	Facial Image			
	Low	Messages to the Chatbot			
	Medium	Game Input			
Outputs	Medium	Emotion Prediction			
	High	Chatbot Replies			
Logical Files	Medium	User Authentication Database			
	Medium	NLP Word bank			
Inquiries	Low	Browse Videos			
	Low	Browse Games			
Interfaces	High	Backend Connection			
	Medium	Authentication Verification			

## Calculation of Unadjusted Function Points:

Characteristic	Low	Medium	High		
Inputs	2	$\times 3$	2	$\times 4$	0
Outputs	0	$\times 4$	1	$\times 5$	1
Inquiries	2	$\times 3$	0	$\times 4$	0
Logical Files	0	$\times 7$	2	$\times 10$	0
Interfaces	0	$\times 5$	1	$\times 7$	1
Unadjusted FP	12		40		17
Total=L+M+H	69				

## Adjusted Function Points

Influence Factors	Score	Detail
Data Communications	5	Application is more than a front-end, and supports more than one type of teleprocessing communications protocol.
Distributed Functions	5	Distributed processing and data transfer are online and in both directions.
Performance	4	Response time or throughput is critical during all business hours. No special design for CPU utilization was required. Processing deadline requirements with interfacing systems are constraining.
Heavily used	4	Some security or timing considerations are included.
Transaction rate	2	Daily peak transaction period is anticipated.
On-line data entry	0	More than 30% of transactions are interactive data entry
End-user efficiency	3	Four to five of the efficiency designs are included
On-line data update	0	Online update of major internal logical files is included.
Complex processing	5	Any one of the complex components
Reusability	3	The application was specifically packaged and/or documented to ease re-use, and the application is customized by the user at source code level.
Installation Ease	0	No special considerations were stated by the user but special setup is required for installation.

Operational Ease	5	Effective start-up, back-up, and recovery processes were provided, but no operator intervention is required (count as two items).
Multiple sites	3	User requirements do not require considering the needs of more than one user/installation site.
Facilitate change	2	Flexible query and report facility are provided that can handle complex requests, for example, and/or logic combinations on one or more internal logical files (count as three items).
Total score	41	

Influence Multiplier = Total score  $\times$  0.01 + 0.65 = 41  $\times$  0.01 + 0.65 = 1.06

Adjusted FP = Unadjusted FP  $\times$  Influence Multiplier = 69  $\times$  1.06 = 73.14

Scoring (0 – 5)
0 = No influence
1 = Insignificant influence
2 = Moderate influence
3 = Average influence
4 = Significant influence
5 = Strong influence

#### Lines of Code

Most of our coding would be done in Python (Flask, Machine learning) and JavaScript (React JS). According to Pabipedia, each Function Point requires 20 lines of code if the application is implemented using Python. Since Python and JavaScript are very similar in their implementation, we are using the same value for both. Therefore, we have: Lines of Code = 73.14 FP  $\times$  20 LOC/FP = 1463 LOC

#### Efforts, Duration and Team Size Estimation

To estimate the effort and duration required for the project, we use function points as the basis to calculate Effort, Duration, Team size and finally the schedule. The estimates are expanded to account for project management and extra contingency time to obtain the total average effort estimates. From these averages, the duration of each work package in working days is estimated based on the following calculations.

- Working days include 5 days in a week.
- Effort = Size / Production Rate = (1463 LOC) / (30 LOC/PD) = 48.7 PD
- Duration = 3  $\times$  (Effort) 1/3 = 3  $\times$  (48.7) 1/3 = 10.95 Days
- Initial schedule = 10.4 Days / 4 days a week = 2.73 Weeks
- Team size = Effort / Duration = 48.7 PD / 10.95 D = 5 Persons
- Working hours include 8 hours in a working day.
- Total person-hours (PH) = 48.7 PD  $\times$  8 hours = 390 PH

#### Distribution of Effort

1990's Industry Data	Work Package	Distribution	Estimates
Preliminary Design 18 %	Project Plan	9%	35.1
	Requirement Specification	9%	35.1
Detailed Design 25 %	User Interface	7%	27.3
	Technical Architecture	11%	42.9
	Data Modeling	7%	27.3
Code & Unit Testing 26 %	Code & Unit testing	21%	81.9
	Online Documentation	5%	19.5
Integration & Test 31 %	Integration & Quality Assurance	31%	120.9
	Extrapolated total effort		390
	2% for project management		7.8
	3% for contingency		11.7
	Total effort		409.5

#### Cost Estimates

##### Hardware

Hardware Equipment	Total Price
Individual Team Member's Notebook and Desktops	\$0.00

##### Software

Software Equipment	Total Price
Window OS	\$0.00
MacOS	\$0.00

##### Other Requirements

Others	Total Price
OpenCV, open-sourced	\$0.00
NLTK, open-sourced	\$0.00

Flask, back-end	\$0.00
Wiki, collaborative sharing	\$0.00

## Product Checklist

The plan is that the items listed below will be delivered on the stated deadlines.

Project Deliverable	Estimated Deadline
Project Plan	Oct 4 <sup>th</sup> , 2020
Requirements Specification	Oct 18 <sup>th</sup> , 2020
Design Document	Oct 25 <sup>th</sup> , 2020
Module/System Test Plan	Nov 4 <sup>th</sup> , 2020
System Release (Demo)	Nov 10 <sup>th</sup> , 2020

## Best Practice Checklist

Practice	
Document what we do; all documentation must be in a standardized format.	✓
Pay attention to requirements, check for ambiguity, completeness, accuracy, and consistency. The requirement documentation must contain a complete functional specification.	✓
Keep it simple. Complexity management is one of the major challenges. Strive to: <ul style="list-style-type: none"> <li>▪ Minimize interfaces between modules, procedures and data.</li> <li>▪ Minimize interfaces between people, otherwise exponential communication cost</li> <li>▪ Avoid fancy product functions, design as long as the functionality meets the customer requirements</li> </ul>	✓
Require Visibility. We must see what we build otherwise we can measure the progress and take management action. This includes: the manager must have good communication with his or her employees; require developers to make code available for review; review design for appropriateness.	✓
Plan for continuous change. We must: <ul style="list-style-type: none"> <li>▪ All manuals designs, test, source code should have revision numbers and dates revision history comments, change marks to indicate the changes</li> <li>▪ New revisions should be approved before being made and checked for quality and compliance after being made</li> </ul>	✓
Don't underestimate. We must be careful to obtain accurate estimates for: time, effort, overhead, meeting time, and especially effort on integration, testing, documentation and maintenance.	✓
Code reviews are a much more efficient method to find software defects. Plan and manage code reviews between team members.	✓
Software testing will use both black box and white box testing. It will involve unit, functional, integrating and acceptance testing.	✓

## Risk Management

Besides the general risks that entails any given project, the following risks have been identified for the MindSpace project. The risk are categorized into technological risk, usability risk as well as people-related risk.

### Technological Risk

#### Poor Accuracy of the NLP model

Impact: AI Chatbot unable to formulate an appropriate response.

Severity : High

Probability : Varies, accuracy can be measured using a method in the NLPT library

Risk Reduction : Familiarize with the models that you are capturing to be able to interpret it accurately.

#### Poor Accuracy of Face Recognition Model

Impact : Emotion detected may be inaccurate

Severity : High

Probability : 25%, depends on a lot of factors such as lighting, resolution of image etc.

Risk Management : Give recommendations of the quality of the image to take.

#### Slow Performance of Face Recognition Model

Impact : Slow image processing and slow output to the user

Severity : Medium

Probability : 20%, depending on the resolution of the image.

Risk Management : Ensure good resolution of the image to increase performance.

### **Inability to Integrate Front-End and Back-End**

Impact : Unable to deliver complete project

Severity : High

Probability : 10%

Risk Management : Try to converge parts of the projects earlier before the deadline.

### **Usability Risk**

#### **User Interface Low Usability**

Impact : Users might not want to utilize the website to its full potential.

Severity : Medium

Probability : 20%

Risk Management : Adhere to the Human-Computer Interaction guidelines for better usability.

#### **Chatbot not able to provide required solution to the user**

Impact : Defeats the main goal of the project.

Severity : High

Probability : 20%, linked to the poor accuracy of the NLP

Risk Management : same as improving the accuracy.

### **People-related Risk**

#### **Problems co-ordinating schedule amongst group**

Impact Severity: Moderate

Probability: 40%

Impacts: Different schedules and timetables amongst the team makes it hard to set a time for a meetup.

Risk Reduction: Conducting short and intense meetings rather than a longwinded meeting.

## **Quality Assurance**

The project will achieve quality assurance by following the standard set by the team. The specific procedures and details shall be provided in the Quality Plan. Specific test procedures and details shall be provided in the Module/System Test Plan.

In addition, Mindspace shall make use of two testing methodologies:

- Unit Testing involves testing system components individually.
- In-Place Testing involves testing of the whole system as a unit.

Furthermore, this methodology will be used to test the important aspect of the Mindspace:

- System Function will be tested to ensure that software flaws are eliminated.

This methodology makes broad use of realistic test cases. Detailed test data is an important part of the final project delivery. During testing phase, we shall provide a comprehensive and detailed subset of the needed data for testing purposes. We will validate code using realistic scenarios. In addition, extreme cases will be used to ensure that the system behaves correctly in degenerate cases.

## **Monitoring & Control**

Project controls are a series of tools that help keep a project on schedule. Combined with people skills and project experience, they deliver information that enables accurate decision making. The project control process mainly focuses on:

Many procedures are required in order to be able to successfully monitor the progress of a software project. Some of the most important are:

### **Timeline Planning and Task Decomposition**

This document outlines an estimated timeline for the project. A reasonably accurate timeline can be assembled by hierarchically decomposing tasks into measurable subcomponents and estimating requirements for each. At the same time, this decomposition can assist in task assignment and balancing. Throughout the implementation phase, these subcomponents can allow for fine-grained measurement of progress. Project subcomponents and timeline estimates are included in the Estimates and Work Breakdown Structure sections of this document.

### **Requirements Traceability Matrix (RTM)**

This maps, or traces, the project's requirements to the deliverables. The matrix correlates the relationship between two baseline documents. This makes the project's tasks more visible. It also prevents new tasks or requirements being added to the project without approval. This makes the project's tasks more visible. It also prevents new tasks or requirements being added to the project without approval.

### **Identification of Major Project Risks**

Early identification of major risks to the project allows for the placement of preventative measures before problems can develop. Major risks have been identified in the Risk Management section of this document, along with the measures being taken to avoid them.

### **Perform Quality Control**

Quality control activities check the quality attributes of the delivered outputs. For instance, the product of a project might meet the budget and schedule targets. But the quality requirements might not meet the customers' expectations. In this case, the project will be considered as failed as well. Therefore, performing quality control is important.

### **Review and Status Meetings**

These further analyse problems, finding out why something happened. They can also highlight any issues that might happen later.

## **Risk Management Plan**

### **Version History**

Version	Implemented By	Revision Date	Approved By	Approval Date	Reason
1.0	Arjun	22/09/2020	Arjun	23/09/2020	Added Introduction
2.0	Gaurav	22/09/2020	Arjun	24/09/2020	Added Risk Management Procedure
2.1	Gaurav	24/09/2020	Arjun	25/09/2020	Added Risk Response Planning
3.0	Divvij	25/09/2020	Arjun	26/09/2020	Added Tools and Practices

## **Introduction**

### **Purpose of the Risk Management Plan**

A risk is an event or condition that, if it occurs, could have a positive or negative effect on a project's objectives. Risk Management is the process of identifying, assessing, responding to, monitoring, and reporting risks. This Risk Management Plan defines how risks associated with the "MindSpace" project will be identified, analyzed, and managed. The Plan outlines how risk management activities will be performed, recorded, and monitored throughout the lifecycle of the project and provides templates and practices for recording and prioritizing risks. It is less costly to mitigate risks to prevent them from triggering (to be proactive) than it is to deal with issues that arise if the risk does trigger (to be reactive). Unmanaged risks can easily prevent a project from achieving objectives or even cause it to fail to succeed. Risk management is important during project initiation, planning, and execution; well-managed risks significantly increase the likelihood of project success. The Risk Management Plan is created by the project manager in the Planning Phase of the CDC Unified Process and is monitored and updated throughout the project. The intended audience of this document is the project team, project sponsor, and management.

## **Risk Management Procedure**

### **2.1 PROCESS**

The project manager will ensure that a five-step process is followed in order to respond to potential risks. The first step is to identify the plausible risk. The legal, environmental, market, and regulatory risks are considered and its impact on the project will be thoroughly analyzed. Since the chatbot is dealing with critical topics like psychology, and mental health, potential risks due to non-compliance with local legal authorities will be avoided. The second step is to analyze the risk. This includes analyzing, the scope of the risk internally, and the effects of the risk on the business, internal technical management, and human resource management. Once the risk has been localized, the far-reaching negative repercussions of the risk will be notified to the top management and the specific departments involved. The third step is to evaluate and to rank the risk. The risks will be prioritized based on the severity of the risk. Low-risk cases will be handled internally without much interference from the top management. High-risk cases that might potentially cause severe catastrophic losses will be dwelt with greater caution, after involving the top management, project manager, and the stakeholders involved. The fourth step is to treat the risk. Every risk needs to be eliminated or contained such that its negative impact is minimized. This will either be done internally or by contacting experts externally. The fifth step is to monitor and review the risk. Monitoring is essential to keep a close watch on the risk. The project manager working with the project team and project sponsors will ensure that risks are actively identified, analyzed, and managed throughout the life of the project. Risks will be identified as early as possible in the project so as to minimize their impact. The steps for accomplishing this are outlined in the following sections. The project manager will serve as the Risk Manager for this project.

### **2.2 RISK IDENTIFICATION**

Risk identification will involve the project team, appropriate stakeholders, and will include an evaluation of environmental factors, organizational culture, and the project management plan including the project scope. Careful attention will be given to the project deliverables, assumptions, constraints, WBS, cost/effort estimates, resource plan, and other key project documents. A Risk Management Log will be generated and updated as needed and will be stored electronically in the project library.

### **2.3 RISK ANALYSIS**

All risks identified will be assessed to identify the range of possible project outcomes. The qualification will be used to determine which risks are the top risks to pursue and respond to and which risks can be ignored.

#### **2.3.1 Qualitative Risk Analysis**

The probability and impact of occurrence for each identified risk will be assessed by the project manager, with input from the project team using the following approach:

Probability

- High – Greater than 70% probability of occurrence
- Medium – Between 30% and 70% probability of occurrence
- Low – Below 30% probability of occurrence

#### Impact

- High – Risk that has the potential to greatly impact project cost, project schedule or performance
- Medium – Risk that has the potential to slightly impact project cost, project schedule or performance
- Low – Risk that has relatively little impact on cost, schedule or performance

Risks that fall within the RED and YELLOW zones will have risk response planning which may include both a risk mitigation and a risk contingency plan.

#### 2.3.2 Quantitative Risk Analysis

Analysis of risk events that have been prioritized using the qualitative risk analysis process and their affect on project activities will be estimated, a numerical rating applied to each risk based on this analysis, and then documented in this section of the risk management plan.

#### 2.4 RISK RESPONSE PLANNING

Each major risk (those falling in the Red & Yellow zones) will be assigned to a project team member for monitoring purposes to ensure that the risk will not “fall through the cracks”. For each major risk, one of the following approaches will be selected to address it:

- Avoid – eliminate the threat by eliminating the cause
- Mitigate – Identify ways to reduce the probability or the impact of the risk
- Accept – Nothing will be done
- Transfer – Make another party responsible for the risk (buy insurance, outsourcing, etc.)

For each risk that will be mitigated, the project team will identify ways to prevent the risk from occurring or reduce its impact or probability of occurring. This may include prototyping, adding tasks to the project schedule, adding resources, etc. For each major risk that is to be mitigated or that is accepted, a course of action will be outlined for the event that the risk does materialize in order to minimize its impact.

#### 2.5 RISK MONITORING, CONTROLLING, AND REPORTING

The level of risk on a project will be tracked, monitored and reported throughout the project lifecycle. A “Top 10 Risk List” will be maintained by the project team and will be reported as a component of the project status reporting process for this project. All project change requests will be analyzed for their possible impact to the project risks. Management will be notified of important changes to risk status as a component to the Executive Project Status Report.

### Tools and Practices

A Risk Log will be maintained by the project manager and will be reviewed as a standing agenda item for project team meetings.

Each risk will be also be constantly monitored by the project manager on a weekly basis to ensure everything is under control.

### Risk Management Plan Approval

The undersigned acknowledge they have reviewed the Risk Management Plan for the Mindspace project. Changes to this Risk Management Plan will be coordinated with and approved by the undersigned or their designated representatives.

Signature: Mac Date: 27/09/2020

Print Name: Kiran Mac Milin

Title: Risk Management Plan 7.1

Role: QA Manager

Signature: A.Vaish Date: 28/09/2020

Print Name: Vaish Arjun

Title: Risk Management Plan 7.1

Role: Project Manager

## Software Design Maintainability Report

### Design Strategies

#### The Planning Phase Before Development

First of all, we would analyze and predict the future possible improvements that we might have to implement to Mindspace in the future. For example, if our web applications are widely used, we would have to scale it up to handle the higher traffic rate. For scalability, we would need to work on the backend of the applications which includes the database of the accounts, chatbot, and facial recognition to handle the appropriate amount of data coming in from each user. Second of all, Mindspace values our user experience the most since it will be an application that helps people emotionally and mentally, user experience will be our topmost priority. Hence our application uses the Model-View-Control (MVC) model as our architectural model. This will be highly beneficial for our Graphic User Interface (GUI) design as it allows us to change and add functions easily without affecting the other functions.

#### The Process of Developing

We will be testing our web application in a small test-driven development. Due to safe distancing measures in place in response to COVID-19, we are unable to have a large amount of our target audience to test our product. Instead, team members will fill in the role of testers and provide continuous feedback on the design and usability of our web application.

## Correction by Nature

There will be two measures that will be put in place to ensure product correctness, corrective and preventive measures.

- Corrective Maintainability

After testing and reviews, issues that have been identified will go through the corrective procedure.

- Preventive Maintainability

Features implemented will be tested independently before converging, making errors be detected easily and thus be fixed.

## Enhancement by Nature

We will enhance our application while testing the application. And this is what we will look out for:

- Adaptive Maintainability

Ability to run in multiple Operating Systems and Environment.

- Perfective Maintainability After product delivery, quickly detect an error and correct it, reducing maintenance costs and time required.

## Maintainability Practices

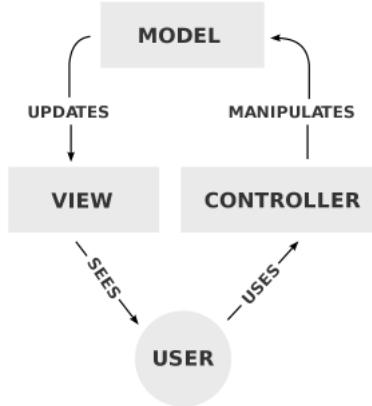
To uphold quality in both process and product, we have implemented the following maintainability practices over the course of our project:

- Readable Code through individual efforts.
- shared google drive for important documents.
- proper and timely update of the MediaWiki page for consistency.

## Architectural Design Patterns

The design pattern used for our project is the MVC Framework which consists of three main components. The following are the components used:

1. **Model** - The Model component includes all the data-related logic that the user works with. This can represent any business logic related data or data that is being transferred between the View and Controller components. In our project, this includes the mini games made on Unity, the chatbot interface and other user details.
2. **View** - The View component is used for the User Interface of the application. It consists of code for all the different pages on our web application that the final user interacts with.
3. **Controller** - Controllers act as an interface between Model and View components which process all the business logic and incoming requests, manipulate data using the Model component and interact with the Views to render and display the final output. For example, the Chatbot controller will handle the user inputs from the Chatbot View and update the database using the Chatbot model.



## Software Configuration Management Tools

This is where we will discuss on version control management, and tracking on who made what changes and when.

### MediaWiki

MediaWiki is a free and open-source application. This service is used as it is easy for beginners to pick up. There are many FAQs provided which can teach users the functions required by the users. There is a wide range of functions which allows users to create their information in different styles. It also allows users to concurrently edit the page at the same time. Hence, editing of the page will not result in a loss of information.

### GitHub

GitHub is a source code hosting platform using the distributed version control and source code management Git. GitHub is chosen for its familiarity and support provided by various IDE applications. GitHub also supports issue tracking similar to a ticketing system. Whether it's a software bug, code enhancement or documentation, users can open an issue, label them appropriately and assign them for other team members to resolve. All users involved will receive timely updates on the progress of the issue.

### Google Drive

Google Drive service is used as file storage and for the backup of documents initially created. This service allows users to share and store files within the group easily. Furthermore, Google Drive allows users to edit documents using the full functionality of applications like Google Docs, Google Sheets etc. This service allows users to edit documents concurrently and supports version control.

# Software Configuration Management Plan

## Identification

The overall objective of a Configuration Management (CM) Plan is to document and inform project stakeholders about CM with the project, what CM tools will be used, and how they will be applied by the project to promote success.

## Document overview

The Mindspace CM Plan defines the project's structure and methods for • Identifying, defining, and baselining configuration items (CI) • Controlling modifications and releases of CIs • Reporting and recording status of CIs and any requested modifications • Ensuring completeness, consistency, and correctness of CIs • Controlling storage, handling, and delivery of the CIs

The intended audience of the CM Plan is the project manager, project team, project sponsor and any senior leaders whose support is needed to carry out communication plans.

## Abbreviations and Glossary

<b>Acronym</b>	<b>Definition</b>
ABL	Allocated Baseline
AS	Acquisition Strategy
CBL	Concept Baseline
CCB	Configuration Control Board
CDR	Critical Design Review
CDRL	Contract Data Requirements List
CI	Configuration Item
CIL	Configuration Item List
CM	Configuration Management
CML	Configuration Management Library
CMP	Configuration Management Plan
CMN	Common
COR	Contracting Officer Representative
COTS	Commercial Off The Shelf
CPP	Configuration Management PVCS Procedures
CR	Change Request
CSA	Configuration Status Accounting
ELC	ERA Life Cycle Document
EOP	Executive Office of the President
ERA	Electronic Records Archives
FBL	Functional Baseline
FCA	Functional Configuration Audit
FOC	Full Operational Capability
HWCI	Hardware Configuration Item
ICWG	Interface Control Working Group
IEEE	Institute of Electrical and Electronics Engineers
IRD	Interface Requirements Document
ITSS	Information Technology Support Services
IV&V	Independent Verification and Validation
LMC	Lockheed Martin Corporation
MCD	Master Configuration Document
MP	Metrics Plan
NARA	National Archives and Records Administration
OCR	Operations Change Request
Ops CCB	Operations Configuration Control Board
PAT	Product Acceptance Test
PBL	Product Baseline
PCA	Physical Configuration Audit
PD	Program Director
PDR	Preliminary Design Review
PM	Project Manager
PMO	Program Management Office
PMP	Program Management Plan
PSD	Program Support Division Director
PVCS	Polytron Version Control System
QA	Quality Assurance
QM	Quality Management
QMP	Quality Management Plan
RD	Requirements Document
RM	Release Manager
SCM	Software Configuration Management
SDR	System Design Review
SOP	Standard Operating Procedure

SRR	System Requirements Review
Std	Standard
SyRS	System Requirements Specification
TBD	To Be Determined
TEP	Technical Review Process
TRA	Training Needs Assessment
UCM	Unified Change Management
WBS	Work Breakdown Structure

## References

The following documentation was used to support the development of this document using the versions listed below.

### Project References

- Project Plan
- Project Proposal
- Quality Plan

### Standard and regulatory References

The standards and guidelines used in the preparation of this document are listed below

- IEEE Std. 610.12-1990, IEEE Standard Glossary of Software Engineering Terminology
- IEEE Std. 828-2005, IEEE Standard for Software Configuration Management Plans
- OMB Circular A-130, Management of Federal Information Resources

## Conventions

A naming convention, combined with labels, is used to uniquely identify the CIs and Work Products that are placed under CM control for the ERA program. The naming convention uniquely identifies each item and its different versions.

The naming convention for the CIs is organized into two (2) parts: CI type and the type of file. File names consist of a combination of the file or CI title with an underscore between each word in the title and file type extension

E.g: Quality Plan would be stored as Quality\_Plan.doc

## Organization

The Software configuration for this project will be managed by the following members of the development team

- The Software Configuration Manager (SCM)
- The Project Manager
- The Quality Manager
- The Lead Developer

### 2.1 Activities and responsibilities

Activities When Setting Up the Project	Person responsible
Identify the configuration items	SCM
Install the bug repository tool and set up the database	SCM
Install the software configuration repository tool and set up the database	SCM
Manage and structure the reference space	SCM
Define the configuration processes	SCM

<b>Activities during the project lifecycle</b>	<b>Person responsible</b>
Export components for modification, test or delivery	SCM
Set under control validated components	
up the database	SCM
Create version, write version delivery document	SCM
Approve reference configurations	Project manager
Verify version to be delivered and authorise deliveries	Project manager
Backup spaces	SCM
Do configuration audits	Quality Manager
Inspect configuration records	Quality Manager
Archive reference version	SCM

<b>Manage versions and archives</b>	<b>Person responsible</b>
Manage configuration records	SCM
Install the software configuration repository tool and set up the database	SCM
Produce reports and statistics	SCM
Manage reference space and its access control list	SCM
Manage spaces backup and archive media	SCM
Manage quality reports	Quality Manager

### 2.1.1 Decisions process and responsibilities

Responsibilities during reviews, audits and approvals are listed

At the end of an activity of the project:

<b>Activities</b>	<b>Person responsible</b>
Do a configuration freeze	SCM
Present a configuration state of the components impacted by the activity	SCM
Present a documentation state of the components impacted by the activity	SCM

During a configuration management process audit:

<b>Activities</b>	<b>Person responsible</b>
Do the configuration management process audit	Project Manager
Present the records of the configuration management process	SCM
Present the quality records of the configuration management process	Quality Manager
Present the records of the documentation management process	SCM

## Configuration identification

### 1. Identification rules of configuration items

#### a. Identification of a configuration item

- The identification of a configuration item is done using the following identification scheme:
  - XXX\_Va.b
  - Where:
    - “Va.b” is the version of the configuration item

#### b. The version number of a configuration item

The attribution of a version number is a prerequisite to any delivery of any configuration item. This number shall be incremented before a new delivery, if the product or its documentation were modified.

The definition rules of a version number are the following:

- Major edits call for a new major version (a)
  - Adding or removing significant sections, functions or features
  - Redevelopment of document
  - New Design
  - Major change in user functionality

- Minor edits call for a new subversion (b)
  - Editing significant sections, functions or features
  - Change formatting
  - Bug Fix

## 2. Identification rules of documents

**a. Description of documents identifiers** The identification of documents is described below:

**XXX\_<document type>\_<document number>\_<revision index>**

where:

- "document type" is Foo for FOO documents and BAR for bar documents,
- "document number" is an incremental number, with a separate list for each document type,
- "revision index" designates the approved iteration of the document. The revision index is V1 for the first iteration, V2 for the second and so on.

### b. Definition and evolution of the revision index

The attribution of a revision index is a prerequisite to any delivery of a document or file. This index shall be incremented before the diffusion of a modified document.

The definition rules of a revision index are the following:

- The revision index is incremented for a major edit
  - Adding or removing significant sections, functions or features
  - Redevelopment of document

## 3. Identification rules of a media

### a. Internal identification

A media can be identified as a CDROM, a thumb drive or a hard drive

The identification of a media is described below:

**<configuration item identification>/<media>/<volume>**

where:

- "media" is the media number,
- "volume" is an incremental number to distinguish the media if the delivery contains more than one media.

## 4. Reference configuration identification

Each reference configuration is defined by:

- An identifier,
- Its content listed in the corresponding Version Delivery Description document,
- The acceptance or validation reviews associated to the building of the reference configuration.

A reference configuration is established for each design review and each test review of the project.

## 5. Configuration Baseline Management

Describe what baselines are to be established. Explain when and how they will be defined and controlled.

Examples of baselines :

- functional baseline (FBL), which describes the system functional characteristics;
- allocated baseline (ABL), which describes the design of the functional and interface characteristics,
- product baseline (PBL), which consists of completed and accepted system components and documentation that identifies these products.

## Configuration control

Changes to software components will happen only when a bug is found or when the performance is not optimal. The following process will ensure the application functions properly if any of the changes are made.

## Change Management

Changes are bound to occur in any software project, no matter simple nor complex. Changes could be due to user demands or due to errors and bugs. And during every change, all of the software components must still work according to requirements. The process to control the changes to the baseline and for tracking the change implementation is as follows:

Example for problem resolution:

1. The change request form containing the details of software errors are submitted by users
2. The development team reviews the change request form and evaluates the impact and the extent of the modification
3. When the change request form is approved, the developers will then assess the cost, feasibility and priority of the changes
4. The change control board then review the changes and decides whether to implement it or not
5. If the change is approved, then the change will be implemented in a new branch of the system

Example for multiple configuration:

1. The change request form containing the details of software errors are submitted by users
2. The development team reviews the change request form and evaluates the impact and the extent of the modification
3. When the change request form is approved, the developers will then assess the cost, feasibility and priority of the changes
4. The change control board then review the changes and decides whether to implement it or not
5. If the change is approved, then the change will be implemented in a new branch of the system

## Configuration support activities

### Configuration Status Accounting

Configuration Status Accounting (CSA) is the process to record, store, maintain and report the status of configuration items during the software lifecycle. All software and related documentation should be tracked throughout the software life.

### Evolutions traceability

The traceability of modifications of items given their types:

- Document: The modification sheet number identifies the origin of the modification. The modified paragraphs in the document are identified, if possible, by revision marks.
- Source file: The software configuration management tool records, for each source file or group of source files, a comment where is described the modification.
- Configuration item: The Version Delivery Description of the article identifies the modification sheet included in the current version.

The modification sheet describes the modifications done to the components with enough precision to identify the modified parts.

### Setting up Configuration status

The SCM sets up the state of all versions and of each configuration article with:

- The label,
- The version number,
- The creation date of the VDD,

The SCM writes the VDD.

### Configuration status diffusion

The SCM and the quality manager write the VDD.

### Configuration status records storage

The records are stored in a configuration folder, which contains:

- The requests sorted by record number,
- The software documents,
- The VDD's,
- The configuration states sorted chronologically.

### Configuration audits

Software configuration audits ensure that the final application functions as promised and satisfies all specifications and requirements i.e. the delivered software complies with control standards, audit and report the changes made. Configuration audit methods used are –

- Functional Configuration Audit – A CM method examination used to verify, test, inspect and analyze that the final product has met the requirements specified in the functional baseline documentation.
- Physical Configuration Audit – This is a CM examination of the implemented software against its technical documentation. It encapsulates a detailed examination of drawings, design documents and specifications to ensure that the documentation set is ready to support the post development process.

### Reviews

Conducting technical reviews are necessities for any software projects to ensure its conformity, accuracy and completeness and if the delivery fails to tick the three things that I've mentioned, corrective actions must occur. Before making the product of any SDLC stage immutable, technical reviews and auditing needs to be performed to ensure that requirements are being followed and that the product is on track. Performing technical reviews on release or change branches are also important to make sure that it is ready to be merged onto the main branch.

Our software configuration manager will be in charge of overseeing the processes under the Software Configuration Management, such as overseeing the baselining process, identifying configuration items, change control, status accounting and audits and reviews. As a whole, the build and release of our product. They will be overseeing every revision that was made on the source code.

### Configuration management plan maintenance

The CM Manager is responsible for the development and maintenance of this plan. The CMP will continue to be updated as needed throughout the entire lifecycle of the project to ensure the relevance and adequacy of the CMP to plan and manage CM activities. Scheduled updates to the CMP will follow the projected dates listed in WBS. The CMP is under CM control.

## Change Management Plan

### Version History

Version	Implemented By	Revision Date	Approved By	Approval Date	Reason
1.0	Arjun	20/10/2020	Arjun	21/10/2020	Added Introduction
2.0	Mac	21/10/2020	Arjun	22/10/2020	Added Change Request Process Flow Requirements
2.1	Gaurav	22/10/2020	Arjun	24/10/2020	Added Change Request Form and Change Management Log
2.2	Divvij	24/10/2020	Arjun	25/10/2020	Added Evaluating and Authorizing Change Requests
3.0	Liang Jing	26/10/2020	Arjun	28/10/2020	Added Responsibilities

## Introduction

### Purpose Of The Change Management Plan

The Change Management Plan documents and tracks the necessary information required to effectively manage project change from project inception to delivery. The Change Management Plan is created during the Planning Phase of the project. The purpose of the Change Management process is to control the lifecycle of all changes, enabling beneficial changes to be made with minimum disruption to IT services.

The objectives of the Change Management process are to:

- Respond to the customer's changing business requirements while maximizing value and reducing incidents, disruption and re-work
- Respond to the business and IT Requests for Change that will align the services with the business needs
- Ensure that changes are recorded and evaluated, and that authorized changes are prioritized, planned, tested, implemented, documented and reviewed in a controlled manner
- Ensure that changes to configuration items are recorded in the Configuration Management System
- Optimize overall business risk
- It is often correct to minimize business risk, but sometimes it is appropriate to knowingly accept a risk because of the potential benefit
- Changes should be managed to:
  - Optimize risk exposure (supporting the risk profile required by the business)
  - Minimize the severity of any impact and disruption
  - Achieve success at the first attempt
- Ensure that all stakeholders receive appropriate and timely communication about change so that they are aware and ready to adopt and support the change

Its intended audience is the project manager, project team, project sponsor, and any senior leaders whose support is needed to carry out the plan.

## Change Management Process

### Change Request Process Flow Requirements

The Change Request Process Flow is outlined below:



**1) Create & Log the Request for Change (RFC)** A Request for Change is typically created by the individual, process or business unit requiring the change. Depending on the type of change, an RFC record will contain varying information necessary to make decisions for authorization and implementation of the change, including, identifying information, a description, configuration item incurring the change, a reason for the change, requestor's contact information, type of change, timeframe, costs, back out plan and business justification.

#### 2) Review Request for Change (RFC)

Each Request for Change should be reviewed and prioritized by the change authority for business practicality. These requests can be rejected and returned to the submitter or management as notification or in request of more detail. These unapproved changes should be monitored and closed as needed.

#### 3) Evaluate the Change

Evaluating the change to assess the impact, risk and benefits to IT services are critical in order to avoid unnecessary disruption to business operations. For certain types of changes, such as major changes, a formal change evaluation takes place by the change evaluation process and is documented in a Change Evaluation Report. The impact assessment will consider the impact on the application infrastructure, user experience, other services – both IT and non-IT services, implementation resources and currently scheduled changes in the changelog.

#### 4) Approve/Authorize the Change

Change requests commonly require authorization prior to implementation and each change requires authorization from the appropriate authority level depending on the type of change (strategic, tactical, operational).

#### 5) Coordinate Implementation

Once authorized, a change request or the change record is handed over to the release and deployment process for coordination and collaboration with the application management teams for building, testing and deploying the change. Each change should have remediation plans prepared in the case of an implementation failure. Once building and testing are complete, release and deployment should notify the change manager of the results and suggested implementation requirements. The Change Manager should schedule each CHANGE based on the suggested implementation requirements and the management of business risk. The Change Manager using a Forward Schedule of Changes (FSC) or Change Schedule will communicate to all stakeholders upcoming changes that may impact them. The FSC along with projected service outages (PSO), or expected deviations in service availability, will be taken into consideration when coordinating change implementation. Release and Deployment will be responsible for the implementation and coordination of training needs.

#### 6) Review and Close Change Request

Upon completion of the change, a Post Implementation Review (PIR), which is a review of the detail implementation results, should take place to confirm the change has successfully achieved its objectives. If successfully implemented, and the change was associated with fixing and error in service all associated problems and known errors should be closed. If not successful, the remediation plan should be activated appropriately.

A Change Management policy should also be defined to support the process. This policy might include, defining what an emergency change is; the implied benefit of the process; encouraging a change and ITIL friendly business culture, establishing roles and responsibilities for various change management activities, restricting change management access to authorized staff, risk management and performance measurement.

### Change Request Form and Change Management Log

Change Request Form:

Element	Description
Date	The date when the CR was created
CR#	Identification number assigned by the Change Manager
Title	A brief description of the change request
Description	Description of the desired change, the impact, or benefits of a change should also be described
Submitter	Name of the person completing the CR Form who can answer questions regarding the suggested change
Phone	Phone number of the submitter
E-Mail	Email of the submitter
Product	The product that the suggested change is for
Version	The product version that the suggested change is for
Priority	A code that provides a recommended categorization of the urgency of the requested change (Values given in following section)

Change Management Log:

Element	Description
Team Name	Name of the team working on the project
Project Manager	Name of the person in charge of the planning and execution of the project
Last Updated Date	The last date when the CML was updated
Date Created	The date when the CML was created
Version	The version of the CML
CR#	Identification number assigned by the Change Manager
Escalation Status	Shows status of escalation to the CCB (Yes/No)
Description	Description of the desired change
Steps Required	Steps to take for the change to occur
Impact	A value that shows the impact of the change (Insignificant, Tolerable, Serious, Catastrophic)
Submitter	Name of the person completing the CR Form who can answer questions regarding the suggested change
Date of Request	The date when the CR was created
Change type	Type of change required (Values given in following section)
Status	Status of the CR (Values given in following section)
Priority	A code that provides a recommended categorization of the urgency of the requested change (Values given in following section)

### Change Request Form and Change Management Log

Priority Criteria:

Priority	Description
Critical	<ul style="list-style-type: none"> <li>▪ Malfunctioning of the entire application</li> <li>▪ Major changes</li> <li>▪ The changes must be made as soon as possible</li> <li>▪ Drastic change in the market environment</li> </ul>
High	<ul style="list-style-type: none"> <li>▪ Certain parts of the application are not functioning properly</li> <li>▪ Changes can be implemented quickly</li> <li>▪ Change on the recommendation of senior board member</li> </ul>
Medium	<ul style="list-style-type: none"> <li>▪ No major changes to the system</li> <li>▪ Changes are necessary</li> <li>▪ Major bug fix</li> </ul>
Low	<ul style="list-style-type: none"> <li>▪ Minor Changes</li> <li>▪ Rectification can be delayed</li> <li>▪ Minor bug fix</li> </ul>

Change Types:

Type	Description
Scope	Change that affects the scope
Time	Change that affects the time
Duration	Change that affects the duration
Cost	Change that affects the cost
Resources	Change that affects the resources
Deliverables	Change that affects the deliverables
Product	Change that affects the product
Process	Change that affects the process
Quality	Change that affects the quality

Status Types:

Status	Description
Open	Entered/Opened but not yet approved or assigned
Work in Progress	CR approved, assigned, and work is progressing
In Review	CR work is completed and in final review prior to testing
Testing	CR work has been reviewed and is being tested
Closed	CR work is completed, has passed all tests, and updates have been released.

## Change Request Form and Change Management Log

Priority Criteria:

Priority	Description
Critical	<ul style="list-style-type: none"> <li>▪ Malfunctioning of the entire application</li> <li>▪ Major changes</li> <li>▪ The changes must be made as soon as possible</li> <li>▪ Drastic change in the market environment</li> </ul>
High	<ul style="list-style-type: none"> <li>▪ Certain parts of the application are not functioning properly</li> <li>▪ Changes can be implemented quickly</li> <li>▪ Change on the recommendation of senior board member</li> </ul>
Medium	<ul style="list-style-type: none"> <li>▪ No major changes to the system</li> <li>▪ Changes are necessary</li> <li>▪ Major bug fix</li> </ul>
Low	<ul style="list-style-type: none"> <li>▪ Minor Changes</li> <li>▪ Rectification can be delayed</li> <li>▪ Minor bug fix</li> </ul>

Change Types:

Type	Description
Scope	Change that affects the scope
Time	Change that affects the time
Duration	Change that affects the duration
Cost	Change that affects the cost
Resources	Change that affects the resources
Deliverables	Change that affects the deliverables
Product	Change that affects the product
Process	Change that affects the process
Quality	Change that affects the quality

Status Types:

Status	Description
Open	Entered/Opened but not yet approved or assigned
Work in Progress	CR approved, assigned, and work is progressing
In Review	CR work is completed and in final review prior to testing
Testing	CR work has been reviewed and is being tested
Closed	CR work is completed, has passed all tests, and updates have been released.

## 1. Change Control Board

Caption text

Role	Name	Contact	Description
Project Manager	Vaish Arjun	83481255	<ul style="list-style-type: none"> <li>■ The project manager is one of the highest levels of authorities in the group</li> <li>■ Role of ensuring that the project stays on track and all deadlines are met</li> <li>■ Has authority to overrule a change</li> <li>■ Considers the financial and economic impact of the proposed changes</li> </ul>
QA Manager	Kiran Mac Milin	86829306	<ul style="list-style-type: none"> <li>■ Quality assurance engineer is to be consulted to make sure that the changes meet the coding practices and standards</li> <li>■ Considers the quality and regulatory impacts of the proposed changes</li> </ul>
Lead Developer	Asok Kumar Gaurav	98114906	<ul style="list-style-type: none"> <li>■ Has to ensure that the Software development is on track</li> <li>■ Considers the technical, and algorithmic feasibility and implementability of the proposed changes</li> </ul>
Release Engineer/Manager	Liang Jing	84844627	<ul style="list-style-type: none"> <li>■ Has lower level of authority, but works closely with the project manager</li> <li>■ Considers the scheduling impact of the proposed changes</li> </ul>

## Responsibilities

Caption text

Role	Name	Contact	Description
Project Manager	Arjun	83481255	Overall delivery of product
QA Manager	Mac	86829306	Overall technical lead, responsible for technical aspects of product release
QA Engineer	Abby	90682219	Mobile app or Web app front-end programming. Participate in the entire SDLC, generating work products including documentation, source code, unit and integration tests
Lead Developer	Gaurav	98114906	Server, application and database programming. Participate in the entire SDLC, generating work products including documentation, source code, unit and integration tests
Front-end Developer	Shashwat	90873378	Overall product and process quality, implementation of QA processes
Back-end Developer	Divvij	98975410	Devise test plans, conduct tests
Release Engineer/Manager	Liang Jing	84844627	Create baselines and build and integrate changes for delivery. Manage releases of product (or prototype in our case).

## Release Plan

### Revision History

Version	Date	Organization/Point of Contact	Description of Change
1.0	23/10/2020	Mac	Added Introduction
2.0	24/10/2020	Gaurav	Added Referenced Documents
3.0	25/10/2020	Divvij	Added Overview
4.0	26/10/2020	Abby	Added Assumptions, Constraints and Risks
5.0	27/10/2020	Shashwat	Added Release Approach
5.1	28/10/2020	Arjun	Added Release Notification

### Introduction

This release plan details the software build of MindSpace application to date along with the completed functionalities. For the customer, this document provides a potential reference to look forward to in our completed application. It would present a detailed summary of all upcoming releases and their impact. The development team can utilize this plan to prioritize tasks and keep track of their progress. They can decide the release versions based on the urgency and significance of subsequent tasks. The release information of the first version is given below.

Application: MindSpace

Release Version: 1.1.0

Date Released: 15 October 2020

### Referenced Documents

Document Name	Issuance Date
MindSpace Project Proposal	12/09/2020
MindSpace Project Plan	21/09/2020
MindSpace System Requirement Specification	29/09/2020
MindSpace Quality Plan	29/09/2020

### Overview

Mental health is something that is very important to every human being, but it is a topic that is sometimes overlooked due to its stigma in society. There are many people who suffer from mental health problems but are afraid to open up and talk to people about it. Very often people try to ignore their problems because they feel uncomfortable sharing their feelings with their family and friends or taking external help.

MindSpace is an application created to provide a virtual solution for mental health issues. The main feature of MindSpace is the chatbot, Chatty, which is created using Natural Language Processing and Artificial Intelligence. It is meant to be an alternative to a therapist, having a human and personal touch to it. Chatty can give solutions to the user's problems such as depression and anxiety or just be a companion to talk to. MindSpace includes other features as well, including games for stress relief and facial recognition for mood detection.

Though other similar solutions exist, MindSpace gives more of a human touch to the user and acts as a better replacement to a real therapist. Keeping in mind software maintainability as well as ease of refactoring and programming, MindSpace will be developed as a web app using React JS.

## Assumptions, Constraints, Risks

### Assumptions

It is assumed that in order for our programme to work, the open-sourced libraries that we have used are working, functional and fits the current needs of our program as well as assuming that our chatbot is trained extensively that will result in the responses being apt and concise.

### Constraints

The most common constraints include budget, resources, and technical aspects. We do not have any constraints in the aspects of budget and resources, since most of our resources are open-sourced.

There are 3 constraints issues that we had identified and they are scalability, the usage of open-source libraries, and project schedule. Firstly, our back-end deployment is through flask and we'll be running the program locally, if there is a need for multiple users to access the programme, we will have to deploy it on a server. Secondly, since most of our resources used in the project are open-source, they are generally less reliable than proprietary software. Lastly, project schedule, we are only given 11 weeks to complete the project from requirement analysis to production build.

### Risks

In the project planning document, we have identified 3 categories of risks, technological risk, usability risk as well as people-related risk.

1. For technological risk, there are 4 risks identified
  1. Poor accuracy of the NLP model. It occurs when the AI chatbot is unable to respond appropriately, it can be mitigated through training the AI agent extensively so that it is able to interpret and reply appropriately.
  2. Poor Accuracy of Face Recognition Model occurs when the model does process the user's image properly due to the user's environment(bad room lighting etc), it can be mitigated through giving suggestions on how the image should be captured.
  3. Slow image processing and slow output of emotion detected, it occurs when there is a latency between capturing the face data, and the output of the emotions. It can be mitigated through finding the best resolution of the image to ensure fast processing of the image.
  4. Inability to integrate front-end and back-end on time, it can be mitigated through early and frequent converging of parts.
2. For Usability Risk there are 2 risks Identified
  1. Low Usability, results in the users inability to fully utilize the website to its full potential. It can be mitigated by following the Human-Computer Interaction.
  2. Inappropriate chatbot responses, it can be mitigated by training the AI agent extensively.
3. For People-risk there is 1 risk Identified
  1. Scheduling amongst the group due to timetable conflict, it can be mitigated by performing shorter and intense meetings.

## Release Approach

### Rationale

The development team will follow SCRUM, a subset of Agile Software Development, as the software development lifecycle methodology. As prescribed by SCRUM, weekly meetings are held with a different agenda to furnish a semi-deployable product as described in the "Project Proposal".

The SCRUM process has been chosen as it provides extra adaptability and flexibility to the project. It entails the development and testing of specific functionalities of the application in short development periods called sprints. Any errors arising in a week's sprint can be easily fixed in the subsequent sprint. Additionally, any changes proposed by the sponsor or users can be quickly integrated without backtracking progress.

### Release Strategy

The Release Strategy we employed is known as the Phased Function Rollout. It is a process where we will implement functionalities in separate modules that are part of the entire system independently until the implementation of the system is complete. The incremental approach adopted allows the monitoring of the development progress and raise any issues that occur in the early implementation stages. Information learned from the early implementations of the system could be further applied to guide the rest of the process which results in fewer issues upon release.

A significant amount of time and resources have been invested in developing and implementing the core functionalities of MindSpace. In addition, we have predicted that this app would be widely employed by people who tend to shy away from speaking up or seeking help when needed.

The Major Release includes the core and basic functionalities of MindSpace which includes components like the AI Chatbot and the live Emotion Detection.

For the Minor Releases, an enhancement to the user experience, interactions of the system, and the variety of mini-games we offer will be included. Future Releases of the app will mainly consist of Minor Releases as well as Revisions.

Major Releases occur when there is a chance to upgrade the chatbot/ emotion detection framework which can cause incompatibility issues.

Minor Releases involve significant bug fixes and minor improvements of the web application. Revisions occur when there are little bugs or errors that require fixing.

### Release Content

Based on the MindSpace Software Requirement Specification, the system provides use cases and the users' requirements listed below:

Use Cases:

1. The user must be able to login to the app, create a new account or continue as a guest if they do not want to create an account
2. The user can chat with our AI Chatbot
3. User can use our Emotion Detection Feature to see what they are feeling
4. User can play a variety of mini-games in order to relieve stress
5. User can get a guide on how to perform various therapeutic exercises in order to make hem feel better
6. Users can see the location of the nearest parks which can help them to de-stress.

#### **MindSpace Release Version 0.0.0 - Development Phase**

The development of MindSpace will include all the functionalities mentioned above except for 6. which is "Users can see the location of the nearest parks which can help them to de-stress."

#### **MindSpace Release Version 1.0.0 - Ready for Production / Official Release / Major Release**

The official release of the MindSpace application. Feature 6. which is "Users can see the location of the nearest parks which can help them to de-stress." will also be included in this version. All required functionality based on the Use Cases defined by the developers has been tested and vetted by the QA Engineers and approved by the QA Manager. With this release, the users can use the application which allows us to gather valuable feedback and further improve on the features of MindSpace.

#### **MindSpace Release Version 1.1.0 - 1.Y.Z - Continuous Maintenance and Improvements**

The Minor Release / Patches of MindSpace will involve improving the current user interface and include another mini game. These improvements enhance the user experience while using the system.

#### **Release Schedule**

The Major Release of MindSpace - Version 1.0.0 will be completed by 7th October 2020 as mentioned in the Project Plan Document, with the basic and core functionalities implemented. Any subsequent releases may take up to 1 month to complete. The subsequent releases will focus on maintenance as well as implementing new functionalities based on the feedback gathered by the users.

#### **Release Impacts**

Every time there is a new release, the system impacts associated with the release and user processes that are to be modified will be identified. Any modifications to the system functionalities might affect the future development of the system, therefore it is required for project manager to oversee the release, as his or her responsibilities include monitoring the progress of the team members. The project manager has to be able to foresee any future consequences from each release.

#### **Release Notification**

When a release version has been generated, the respective stakeholders will be notified as stated below. Methods of how the stakeholder will be notified, what information will be included in each notification and prior to release, the timeframes for receipt of the information, will all be included.

Stakeholders	Method of Providing Notification	Information included in notification	Timeframes for receipt of notification
Users	Through system	What changes has been added	3 days prior to version release
Teammates	Emails	What changes are made, including the bugs to be fixed and also reasons why changes were made and user statistics.	Immediately once decision to implement the new release version

## **Test Plan**

### **Test Plan Identifier**

This is the test plan on MindSpace, developed by the team. This is the first version of the test plan and it is currently at level 1 as when new features are added in, the developers will enhance the test plan according to the changes made. This is a main plan for the entire MindSpace project.

### **Introduction**

The test plan is designed to specify the scope, approach, resources and schedule of all the testing activities of the project, MindSpace. The plan identifies the items & features to be tested, the types of testing to be performed, the personnel responsible for testing, the resources & schedule required to complete testing and the risk associated with the plan.

This plan will include the main testing plans. The tests carried out will be: Unit testing, Integration testing, System testing and User acceptance testing.

- Unit testing: Test cases created by the quality management team will be used to test each unit
- Integration testing: Units which have been tested will be tested together to discover faults in integration
- System test: Independent external testers will be requested to use the application and give feedbacks
- User acceptance test: Possible end-users will be asked to use the application and their feedback will be taken into consideration

### **Test Items**

### **Features To Be Tested**

S.No.	Feature	Risk Level
1	Registering onto MindSpace	Low
2	Logging into MindSpace	Low
3	Accessing Chatbot	Low
4	As a user, chatting with the Chatbot and getting appropriate responses	Medium
5	Accessing Different Therapy Methods	Low
6	Viewing the meditation videos	Low
7	Viewing breathing exercises videos	Low
8	Viewing Positive Affirmations	Low
9	Playing therapeutic games	High
10	As a user, using the emotion recognition feature	High

## Features Not To Be Tested

- 1. The minigame when Unity is not initialized.
- 2. The parks to be updated when the Location Services are off.
- 3. The facial recognition feature if the user does not use it for long(>1 hour).
- 4. The user entering unrelated things in the chatbot( chatbot will respond by saying that it does not comprehend).

## Approach

Tests will be conducted according to the documented test cases given in the Test Cases and Requirements Test Coverage Report. The QA engineers will create test runs and the tester will execute the given tests and mark each case as Pass/Fail. The tester should also record the results after running each test case in the report.

Tests marked as Fail should be looked into by developers in charge of that particular feature. They should fix the errors occurring to correct the system. The QA Engineers will review the test runs in the Report and report back to the QA manager accordingly. The test cases will be reviewed by the QA managers before they are finally marked as passed. When the QA manager deems testing to be complete, a test report will be submitted to the project manager who takes a final look to see if the application is working as it should and marks testing as complete.

## Item Pass/Fail Criteria

For each test case, the proper input and output will be given to the tester. The test case will only be marked as passed if the result matches exactly with the ideal output defined in the test case. If the output is not exactly the same, the test case will be deemed to have failed.

If it is unclear that the output matches the ideal result, the QA manager will be consulted who can also consult the project manager if necessary.

The completion criteria for this plan are:

- 1. 100% of unit testing cases are complete
- 2. 90% of integration testing cases are complete and 10% cases have minor defects
- 3. System Testing is done with 20 external testers
- 4. At least 100 users have given their feedback for user acceptance testing

## Suspension Criteria And Resumption Requirements

### 1. Suspension Criteria

- If the team members report that there are 45% failed test cases, then testing will be suspended until the development team fixes all the failed cases.
- The lead developer will be informed about the test results.
- Product manager will consult the lead developer, and release manager for a revised schedule to fix the failed cases.

### 2. Exit Criteria

- Specifies the criteria that denote successful completion of a test phase
- run rate is mandatory to be 100% unless a clear reason is given
- Pass rate is 75%, achieving pass rate is mandatory

## Test Deliverables

The deliverables included in this Test Plan are:

1. Test Cases
2. Test report
3. Revision Logs
4. Defect logs with solutions implemented

## Testing Tasks

The following activities must be completed:

1. Test plan prepared
2. Items to be tested are identified
3. Identify the method of conducting tests

4. Personnel assigned to each test case
5. Conduct testing
6. Fix bugs and errors which are discovered
7. Create defect logs
8. Create test report

## Environmental Needs

The following environmental needs must be satisfied for testing:

- Unity should be installed on the computer in order to run the Word Search minigame
- NPM installed on the user's machine
- The camera should be working in order for the Emotion Detection model to work.

## Responsibilities

The Project Manager / Develop Lead takes on the role of the Test Manager. The Test Manager is responsible for facilitating the testing project, coordinating availability and schedule of testers and training them as needed. Each tester should understand the expectations on the completion date and level of quality. The Test Manager should also communicate any risks to the team.

## Staffing And Training Needs

Testing can be done by one tester. The tester should conduct testing on the entire applications. The tester assigned should have basic knowledge of using chatbots and facial recognition software. It is an added benefit for the user to have basic knowledge on mental health to verify the answers given by the chatbot.

## Schedule

After integrating the frontend and backend, MindSpace application would be thoroughly tested to ensure it doesn't crash or give unanticipated errors. The testing phase is key after development and would last about 2 weeks. The planned testing strategies are as follows-

- Frontend back backend would be tested separately and again after successful integration.
- The QA manager would check the quality of testing and keep progress on track with reference to the backlog. If some bug arises that compromises functionality, it would be given higher priority.
- The development team would keep each other informed about all the bugs fixed through the group's IM channel before pushing their changes to the CMS.

## Risks And Contingencies

Some possible risks to the project and ways to mitigate them are as follows-

Risk	Contingency
Failure of integration with Backend	Possible problems could arise integrating a Python based backend (Flask) with a JavaScript based frontend (React js). If the errors arising cannot be solved by the development team, another backend such as FireBase can be used instead
Pushing of erroneous code to version control	A team member could mistakenly push code ridden with errors to the version control software. To mitigate this, the code would first be tested by the development lead before committing it.
Shortage of testers	The project could have a shortage of people to test. If this happens, other members of the development team as well as the manager can pitch in to ease the burden.

## Approvals

## Test Cases and Requirements Test Coverage Report

**Test Case 1**

Test Case #: 1	Test Name: gameword.js
System: MindSpace	Sub-System: Word Search Puzzle
Designed By: Quality Assurance Team	Design Date: 13 October 2020
Executed By: Quality Engineer	Execution Date: 13 October 2020
Short Description: Testing if the game loads and works	

<b>Pre-Conditions</b>	
1	User must have an internet connection
2	User must be logged in

<b>Step</b>	<b>Action</b>	<b>Expected System Response</b>
1	User enters the 'word search puzzle' page	The word search puzzle page opens up and loads the game.
2	Drag and hold your mouse across wrong alphabet	The alphabet's background will turn red, indicating that it is not the word from the list. Afterwards, it will turn back to its default state.
3	Drag and hold your mouse across the right word	The alphabet' background will turn to green and stay that way throughout the game.

<b>Post-Conditions</b>	
1	The words that are found are all struck off

**Test Case 2**

Test Case #: 2	Test Name: mediatation.js
System: MindSpace	Sub-System: Meditation
Designed By: Quality Assurance Team	Design Date: 13 October 2020
Executed By: Quality Engineer	Execution Date: 13 October 2020
Short Description: Testing the if the Meditation page shows the correct output	

<b>Pre-Conditions</b>	
1	User must have an internet connection
2	User must be logged in

<b>Step</b>	<b>Action</b>	<b>Expected System Response</b>
1	User enters the 'Meditation Routine' page	The Meditation page loads and show the appropriate meditation videos
2	Click on one of the videos	The video plays the correct output

<b>Post-Conditions</b>	
1	The appropriate videos are displayed

**Test Case 3**

Test Case #: 3	Test Name: breathing.js
System: MindSpace	Sub-System: Breathing Exercise
Designed By: Quality Assurance Team	Design Date: 13 October 2020
Executed By: Quality Engineer	Execution Date: 13 October 2020
Short Description: Testing the if the Breathing Exercise page shows the correct output	

<b>Pre-Conditions</b>	
1	User must have an internet connection
2	User must be logged in

<b>Step</b>	<b>Action</b>	<b>Expected System Response</b>
1	User enters the 'Breathing Exercise' page	The Breathing Exercise page loads and show the appropriate breathing exercise videos
2	Click on one of the videos	The video plays the correct output

<b>Post-Conditions</b>	
1	The appropriate videos are displayed

**Test Case 4**

Test Case #: 4	Test Name: positive.js
System: MindSpace	Sub-System: Positive Affirmation
Designed By: Quality Assurance Team	Design Date: 13 October 2020
Executed By: Quality Engineer	Execution Date: 13 October 2020
Short Description: Testing the if the Positive Affirmation page shows the correct output	

<b>Pre-Conditions</b>	
1	User must have an internet connection
2	User must be logged in

<b>Step</b>	<b>Action</b>	<b>Expected System Response</b>
1	User enters the 'Positive Affirmation' page	The Positive Affirmation page loads and show the appropriate text

<b>Post-Conditions</b>	
1	The appropriate texts are displayed

**Test Case 5**

Test Case #: 5	Test Name: parks.js
System: MindSpace	Sub-System: Suggestion of parks near you
Designed By: Quality Assurance Team	Design Date: 13 October 2020
Executed By: Quality Engineer	Execution Date: 13 October 2020
Short Description: Testing the if the Parks page shows the correct output	

<b>Pre-Conditions</b>	
1	User must have an internet connection
2	User must be logged in

<b>Step</b>	<b>Action</b>	<b>Expected System Response</b>
1	User enters the 'Parks' page	The Parks page loads and show the appropriate park details

<b>Post-Conditions</b>	
1	The appropriate park details are displayed

**Test Case 6**

Test Case #: 6	Test Name: Test Letter Buttons in Hangman.js
System: MindSpace	Sub-System: Games
Designed By: Quality Assurance Team	Design Date: 25 October 2020
Executed By: Quality Engineer	Execution Date: 26 October 2020
Short Description: Testing the effect of pressing any letters from the alphabet. There will be two outcomes if the letter is a part of the word to guess or when it is not.	

<b>Pre-Conditions</b>	
1	User must be in the 'Guess the Emotion' page

<b>Step</b>	<b>Action</b>	<b>Expected System Response</b>
1	Click any of the 26 letters	<ul style="list-style-type: none"> <li>• If the letter is a part of the word, it will be input in the bar</li> <li>• Else if it wrong the incorrect guess counter is incremented to 1.</li> </ul>

<b>Post-Conditions</b>	
1	Incremented wrong letter count or the correct answer is displayed in the word bar.

**Test Case 7**

Test Case #: 7	Test Name: Test Wrong Letter Choice Counter in Hangman.js
System: MindSpace	Sub-System: Games
Designed By: Quality Assurance Team	Design Date: 25 October 2020
Executed By: Quality Engineer	Execution Date: 26 October 2020
Short Description: Testing whether the wrong letter chosen counter is incremented up to 6, since there can only be 6 tries.	

<b>Pre-Conditions</b>	
1	User must be in the 'Guess the emotions' page

<b>Step</b>	<b>Action</b>	<b>Expected System Response</b>
1	Incorrectly guess the word by randomly clicking any of the 26 letters to get 6 wrong	'Incorrect guesses: n' text in the Hangman.js page, n gets incremented at every wrong choice

<b>Post-Conditions</b>	
1	Text change to specify how many times the user has gotten the wrong letter choice.

**Test Case 8**

Test Case #: 8	Test Name: Toggle to other minigames from 'Hangman.js'
System: MindSpace	Sub-System: Games
Designed By: Quality Assurance Team	Design Date: 25 October 2020
Executed By: Quality Engineer	Execution Date: 26 October 2020
Short Description: Testing whether clicking on other games from 'Guess the Emotion' would direct you to the games.	

<b>Pre-Conditions</b>	
1	User must be in the 'Guess the emotions' page

<b>Step</b>	<b>Action</b>	<b>Expected System Response</b>
1	Click on 'Word search' Button	The game 'Word search' gets displayed.
2	Click back arrow to go back to 'Guess the Emotion'	'Guess the Emotion' game gets displayed.
3	Click on 'Lights Off' Button	The game 'Lights Off' gets displayed.

<b>Post-Conditions</b>	
1	Ability to toggle from games to games from 'Guess the Emotion' page.

**Test Case 9**

Test Case #: 9	Test Name: Dynamic Link for Emotions Search in "Hangman.js"
System: MindSpace	Sub-System: Games
Designed By: Quality Assurance Team	Design Date: 25 October 2020
Executed By: Quality Engineer	Execution Date: 26 October 2020
Short Description: Testing whether the dynamic link to the emotion is created correctly.	

<b>Pre-Conditions</b>	
1	User must be in the 'Guess the emotions' page

<b>Step</b>	<b>Action</b>	<b>Expected System Response</b>
1	Play the game as instructed	- Generated dynamic link appears at the end of the game
2	Click the link generated link	- Opens a new tab that google searches the 'emotions' that the user had to guess.

<b>Post-Conditions</b>	
1	A new tab is opened with the emotion word.

**Test Case 10**

Test Case #: 10	Test Name: Test the Cells reaction on 'Board.js'
System: MindSpace	Sub-System: Games
Designed By: Quality Assurance Team	Design Date: 25 October 2020
Executed By: Quality Engineer	Execution Date: 26 October 2020
Short Description: Testing the effect of clicking a Cell on the Board.	

<b>Pre-Conditions</b>	
1	User must be in the 'Lights Out' page

<b>Step</b>	<b>Action</b>	<b>Expected System Response</b>
1	Clicking a Cell	<p>The adjacent lights are toggled on or off depending on their current state.</p> <ul style="list-style-type: none"> <li>• Adjacent cell's current state On -&gt; toggled off</li> <li>• Adjacent cell's current state Off -&gt; toggled on</li> </ul>

<b>Post-Conditions</b>	
1	Appropriate cell response should be shown.

**Test Case 11**

Test Case #:11	Test Name: Toggling all cells off on 'Board.js'
System: MindSpace	Sub-System: Games
Designed By: Quality Assurance Team	Design Date: 25 October 2020
Executed By: Quality Engineer	Execution Date: 26 October 2020
Short Description: Testing the goal of the game.	

<b>Pre-Conditions</b>	
1	User must be in the 'Lights Out' page

<b>Step</b>	<b>Action</b>	<b>Expected System Response</b>
1	Complete the game as instructed	'Congratulations you won' text appears.

<b>Post-Conditions</b>	
1	Displays a text when you toggle all the lights off

**Test Case 12**

Test Case #: 12	Test Name: chatbotQuestions.js
System: MindSpace	Sub-System: Chatbot
Designed By: Quality Assurance Team	Design Date: 15 October 2020
Executed By: Quality Engineer	Execution Date: 16 October 2020
Short Description: Testing the questionnaire functionality of the chatbot	

<b>Pre-Conditions</b>	
1	User must have an internet connection

<b>Step</b>	<b>Action</b>	<b>Expected System Response</b>
1	User types a message	Chatbot starts up with welcome message and questions
2	User answers questions	Chatbot detects user's mood and stores the data

<b>Post-Conditions</b>	
1	Chatbot starts regular conversation with the user

**Test Case 13**

Test Case #: 13	Test Name: chatbot.js
System: MindSpace	Sub-System: Chatbot
Designed By: Quality Assurance Team	Design Date: 16 October 2020
Executed By: Quality Engineer	Execution Date: 17 October 2020
Short Description: Testing the conversational functionality of the chatbot	

<b>Pre-Conditions</b>	
1	User must have an internet connection
2	User must have completed the initial questionnaire previously

<b>Step</b>	<b>Action</b>	<b>Expected System Response</b>
1	User types a message	Chatbot responds appropriately and/or gives prompts for user to select
2	User selects prompt	Chatbot gives information regarding the prompt

<b>Post-Conditions</b>	
1	User ends conversation with chatbot

### Test Case 14

Test Case #: 14	Test Name: emotion.js
System: MindSpace	Sub-System: Emotion Detection
Designed By: Quality Assurance Team	Design Date: 5 October 2020
Executed By: Quality Engineer	Execution Date: 10 October 2020
Short Description: Testing the functionality of the Emotion Detection Feature	

<b>Pre-Conditions</b>	
1	User must have an internet connection

<b>Step</b>	<b>Action</b>	<b>Expected System Response</b>
1	Click on the Emotion Detection icon	A window with camera opens automatically showing the emotion of the user real time. The 3 emotions that can be deciphered are happy, neutral and sad.

<b>Post-Conditions</b>	
1	User closes the Camera Window

## CMMI Level 2 Definition

### Executive summary

#### Purpose

This document called CMMI can be used to guide process improvement across a project, division, or an entire organization. CMMI defines the following maturity levels for processes: Initial, Managed, Defined, Quantitatively Managed, and Optimizing.

#### Summary of definition

The project must be Process dependent on individuals.

### Description

Capability Maturity Model Integration (CMMI) is a model used to give appraisals to the organizations and companies which allow third-parties to see the maturity level of their development processes. It can also be used by organizations and companies to improve on their processes. In the CMMI model, there consists five maturity levels which are:

- Maturity Level 1: Initial
- Maturity Level 2: Managed
- Maturity Level 3: Defined
- Maturity Level 4: Quantitatively Managed
- Maturity Level 5: Optimizing

Each maturity level has a set of key process areas associated with it. If an organization or company is rated to be at a given level, it must have achieved all the key process areas in that level and prior. For this project, we will look into the following four process areas that are found in Maturity Level 2 that is:

1. Requirements Management
2. Project Planning
3. Software Quality Assurance
4. Software Configuration Management

## Level 2 KPIs

### Requirement Management (REQM)

The purpose of Requirements Management is to manage the requirements of the project's products and product components and to identify inconsistencies between those requirements and the project's plans and work products.

### Project Planning (PP)

The purpose of Project Planning is to establish and maintain plans that define project activities.

### Project Monitoring and Control (PMC)

The purpose of Project Monitoring and Control is to provide an understanding of the project's progress so that appropriate corrective actions can be taken when the project's performance deviates significantly from the plan.

### Process and Product Quality Assurance (PPQA)

The purpose of Process and Product Quality Assurance (PPQA) is to provide staff and management with objective insight into processes and associated work products.

### Configuration Management (CM)

The purpose of Configuration Management (CM) is to establish and maintain the integrity of work products using configuration identification, configuration control, configuration status accounting, and configuration audits.

### Measurement and Analysis (MA)

The purpose of Measurement and Analysis (MA) is to develop and sustain a measurement capability that is used to support management information needs.

### Supplier Agreement Management (SAM)

The purpose of Supplier Agreement Management (SAM) is to manage the acquisition of products from suppliers.

## Generic goals and practices

**Commitment to perform:** It describes the action taken by the organization to ensure that the process is established and will endure

Generic practices: Establish organizational policies

**Ability to perform:** It describes the preconditions that must be managed in the project or organization to implement the software process

Generic practices: Arrange resources, organizational structures and provide adequate training

**Activities to perform:** It describes the roles and procedures needed to implement a key process area

Generic practices: Establish plans and procedures, perform the intended, tracking the progress, and taking corrective actions if necessary

**Measurements and analysis:** It describes the need to measure the process and analyse the measurement

Generic practices: Examples includes measurements that are taken to measure the process and analyzation of the measurement

**Verify implementation:** It describes the steps to ensure the activities performed are in line with the established process

Generic practices: Keep reviews and audits by management

## Specific goals and practices

### Requirement Management (REQM)

- Specific goal
  - SG 1 Manage requirements
- Specific practices
  - SP 1.1 Understand Requirements
  - SP 1.2 Obtain Commitment to Requirements
  - SP 1.3 Manage Requirements Changes
  - SP 1.4 Maintain Bidirectional Traceability of Requirements
  - SP 1.5 Ensure Alignment Between Project Work and Requirements

### Project Planning (PP)

- Specific goal
  - SG 1 Establish Estimates
  - SG 2 Develop a Project Plan
  - SG 3 Obtain Commitment to the Plan
- Specific practices
  - SP 1.1 Estimate the Scope of the Project
  - SP 1.2 Establish Estimates of Work Product and Task Attributes
  - SP 1.3 Define Project Lifecycle Phases
  - SP 1.4 Estimate Effort and Cost
  - SP 2.1 Establish the Budget and Schedule
  - SP 2.2 Identify Project Risks
  - SP 2.3 Plan Data Management

- SP 2.4 Plan the Project's Resources
- SP 2.5 Plan Needed Knowledge and Skills
- SP 2.6 Plan Stakeholder Involvement
- SP 2.7 Establish the Project Plan
- SP 3.1 Review Plans that Affect the Project
- SP 3.2 Reconcile Work and Resource Levels
- SP 3.3 Obtain Plan Commitment

#### **Project Monitoring and Control (PMC)**

- Specific goal
  - SG 1 Monitor the Project Against the Plan
  - SG 2 Manage Corrective Action to Closure
- Specific practices
  - SP 1.1 Monitor Project Planning Parameters
  - SP 1.2 Monitor Commitments
  - SP 1.3 Monitor Project Risks
  - SP 1.4 Monitor Data Management
  - SP 1.5 Monitor Stakeholder Involvement
  - SP 1.6 Conduct Progress Reviews
  - SP 1.7 Conduct Milestone Reviews
  - SP 2.1 Analyze Issues
  - SP 2.2 Take Corrective Action
  - SP 2.3 Manage Corrective Actions

#### **Process and Product Quality Assurance (PPQA)**

- Specific goal
  - SG 1 Objectively Evaluate Processes and Work Products
  - SG 2 Provide Objective Insight
- Specific practices
  - SP 1.1 Objectively Evaluate Processes
  - SP 1.2 Objectively Evaluate Work Products
  - SP 2.1 Communicate and Resolve Noncompliance Issues
  - SP 2.2 Establish Records

#### **Configuration Management (CM)**

- Specific goal
  - SG 1 Establish Baselines
  - SG 2 Track and Control Changes
  - SG 3 Establish Integrity
- Specific practices
  - SP 1.1 Identify Configuration Items
  - SP 1.2 Establish a Configuration Management System
  - SP 1.3 Create or Release Baselines
  - SP 2.1 Track Change Requests
  - SP 2.2 Control Configuration Items
  - SP 3.1 Establish Configuration Management Records
  - SP 3.2 Perform Configuration Audits

#### **Measurement and Analysis (MA)**

- Specific goal
  - SG 1 Align Measurement and Analysis Activities
  - SG 2 Provide Measurement Results
- Specific practices
  - SP 1.1 Establish Measurement Objectives
    - Resources, People, Facilities, and Techniques.
  - SP 1.2 Specify Measures
    - Information Needs Document, Guidance, Reference, and reporting
  - Specify Data Collection and Storage Procedures
    - Sources, Methods, Frequency and Owners
  - SP 1.4 Specify Analysis Procedures
    - Rules, Alarms, SPC and Variance
  - SP 2.1 Obtain Measurement Data
    - Actual, Plan, Automatic and Manual
  - SP 2.2 Analyze Measurement Data
    - Evaluate, Drill Down, RCA.
  - SP 2.3 Store Data and Results
    - Store, Secure, Accessible, History, and Evidence.
  - SP 2.4 Communicate Results
    - Information Sharing, Dashboards, Up to Date, Simple, and interpret.

#### **Supplier Agreement Management (SAM)**

- Specific goal
  - SG 1 Establish Supplier Agreements
  - SG 2 Satisfy Supplier Agreements
- Specific practices
  - SP 1.1 Determine Acquisition Type
  - SP 1.2 Select Suppliers
  - SP 1.3 Establish Supplier Agreements
  - SP 2.1 Execute the Supplier Agreement
  - SP 2.2 Accept the Acquired Product
  - SP 2.3 Ensure Transition of Products

## Approvals

Version	Implemented By	Revision Date	Approved By	Approval Date	Reason
1.0	Arjun	26/09/2020	Arjun	26/09/2020	CMMI Level 2 Definition Plan draft
2.0	Divvij	27/09/2020	Arjun	28/09/2020	Level 2 KPAs
2.1	Mac	28/09/2020	Arjun	29/09/2020	Supplier Agreement Management (SAM)
3.0	Liang Jing	29/09/2020	Arjun	30/09/2020	Generic goals and practices
4.0	Arjun	1/10/2020	Arjun	3/10/2020	Specific goals and practices

## CMMI audit checklist

KPAs	Specific Goal	Specific Practice	COMPLETED
Process Area Requirements Management	Manage Requirements	1.1 Understand Requirements 1.2 Obtain Commitment to requirements 1.3 Manage Requirements changes 1.4 Maintain Bidirectional Traceability of requirements 1.5 Ensure Alignment between project work and requirements	✓
Project planning	Establish Estimates	1.1 Estimate the Scope of the project 1.2 Establish Estimates of Work Product and Task Attributes 1.3 Define Project Lifecycle Phases 1.4 Estimate Effort and Cost	✓
	Develop a Project Plan	2.1 Establish the budget and Schedule 2.2 Identify Project Risks 2.3 Plan Data Management 2.4 Plan Project's Resources 2.5 Plan Needed knowledge and Skills 2.6 Plan Stakeholder involvement 2.7 Establish the project plan	✓
	Obtain Commitment to the Plan	3.1 Review Plans that affect the project 3.2 Reconcile Work and resource levels 3.3 Obtain Plan commitment	✓
Project Monitoring and Control	Monitor Project Again Plan	1.1 Monitor Project Planning Parameters 1.2 Monitor Commitments 1.3 Monitor Project Risks 1.4 Monitor data management 1.5 monitor Stakeholder involvement 1.6 Conduct Progress Reviews 1.7 Conduct Milestone Reviews	✓
	Manage Corrective Action to Closure	2.1 Analyse Issues 2.2 Take Corrective Action 2.3 Manage Corrective Actions	✓
Process and Product Quality Assurance (PPQA)	Objectively Evaluate Processes and Work Products	1.1 Objectively Evaluate Processes 1.2 Objectively Evaluate Work Products	✓
	Provide Objective Insight	2.1 Communicate and Resolve noncompliance issues 2.2 Establish Records	✓
Configuration Management (CM)	Establish Baselines	1.1 Identify Configuration items 1.2 Establish a Configuration Management System 1.3 Create or Release baselines	✓
	Track and Control changes	2.1 Track Change Requests 2.2 Control Configuration items	✓
	Establish Integrity	3.1 Track Change Requests 3.2 Perform configuration and establish measurement objectives	✓
Measurement and	Align Measurement	1.1 Resources, people, Facilities,	✓

Analysis (MA)	and Analysis Activities	and techniques 1.2 Specify measures 1.3 Specify data collection and storage procedures 1.4 Specify Analysis Procedures	▼
	Provide Measurement Results	2.1 Obtain Measurement data 2.2 Analyse measurement data 2.3 Store Data and results 2.4 Communicate results	✓
Supplier Agreement Management (SAM)	Establish Supplier Agreements	1.1 Determine Acquisition Type 1.2 Select suppliers 1.3 Establish supplier agreements	✓
	Satisfy Supplier Agreements	2.1 Execute the supplier agreement 2.2 Accept the acquired product 2.3 Ensure transition of products	✓

## Meeting Minutes

### Meeting 1

## Meeting Minutes

<b>Subject</b>			
Hack Elite - First Group Meeting			
<b>Date, Time (duration) and Venue</b>			
27/August/2020 10:30 – 12:30 am SWLAB3			
<b>Attendees</b>	<b>Non-Attendees</b>		
Gaurav, Shashwat, Arjun, Liang Jing, Abby, Mac, Divvij			
<b>Chaired by Vaish Arjun</b>			
<b>Last meeting minutes have been reviewed</b>	Yes		
<b>Progress Updates</b>			
Task	Problem/Issue/Progress	Solution/Action	Taken by & deadline
<b>Task 1</b>	Decide team name	Team name – Hack Elite	Everyone
<b>Task 2</b>	App ideas	AI Face Recognition, Health Consultation, Location letter dropping, AI chatbot for therapy	Everyone, 03/09
<b>Task 3</b>	Team roles	Divide the team into different roles according to their strengths.	Everyone, 10/09
<b>Task 4</b>	Project Features	AI Chatbot -emotion detection (face) -mood detection from chat -chat bot recommendations -using nlp and ai	Everyone

<b>Task 5</b>	Project Proposal		Shashwat, 03/09
<b>Task 6</b>	Use Case diagram		Divvij, Gaurav, 03/09
<b>Task 7</b>	Use Case description		Arijun, Mac, 03/09
<b>Task 8</b>	Backlog		Liang Jing, Abby, 03/09
<b>The next meeting will be held</b>		06/09, 10:30, SWLAB3	
<b>This minutes have been agreed by all attendees</b>		Signed by chair	

## Meeting 2

## Meeting Minutes

<b>Subject</b>			
Hack Elite - Second Group Meeting			
<b>Date, Time (duration) and Venue</b>			
06/September/2020 2:00 – 3:00 pm	Zoom		
<b>Attendees</b>	<b>Non-Attendees</b>		
Gaurav, Shashwat, Arjun, Liang Jing, Abby, Mac, Divvij			
Chaired by Vaish Arjun			
Last meeting minutes have been reviewed	Yes		
<b>Progress Updates</b>			
Task	Problem/Issue/Progress	Solution/Action	Taken by & deadline
<b>Task 1</b>	Create Wiki Accounts		Everyone, 07/09
<b>Task 2</b>	Changes to project proposal	Team member information (resume, skills)	Everyone, 09/09
<b>Task 3</b>	Choose frontend/backend for app	React JS for frontend, Flask for backend	Everyone, 10/09
<b>Task 4</b>	Changes to use case diagram		Divvij, Gaurav
<b>Task 5</b>	Team roles	Shashwat – front end Abby – qa engineer Mac – qa manager Divvij – back end	Everyone, 09/09

		Arjun – project manager Gaurav – lead developer Liang Jing – release engineer	
<b>Task 6</b>	Division of work for project	Front end – Shashwat Back end – Gaurav NLP – Mac, Divvij Face Recognition – Arjun Games – Abby, Liang Jing	Everyone, 09/09
<b>The next meeting will be held</b>			10/09, 10:30, SWLAB3
<b>These minutes have been agreed by all attendees</b>			Signed by chair

## Meeting 3

## Meeting Minutes

<b>Subject</b>			
Hack Elite - Third Group Meeting			
<b>Date, Time (duration) and Venue</b>			
10/September/2020 10:30 – 12:30 am	SWLab 3		
<b>Attendees</b>	<b>Non-Attendees</b>		
Gaurav, Shashwat, Arjun, Liang Jing, Abby, Mac, Divvij			
<b>Chaired by Vaish Arjun</b>			
<b>Last meeting minutes have been reviewed</b>	Yes		
<b>Progress Updates</b>			
Task	Problem/Issue/Progress	Solution/Action	Taken by & deadline
<b>Task 1</b>	System Requirement Specification	Abby - topic 3, 10 Liang Jing - topic 4, 11 Shashwat - topic 5, 12 Arjun - topic 6, 13 Mac - topic 7, 8 Divvij - topic 9	Everyone except Gaurav, thursday

<b>Task 2</b>	Quality Plan	Abby - topic 1, 8 Liang Jing - topic 2, 9 Shashwat - topic 3, 10 Arjun - topic 4, 11 Mac - topic 5, 12 Divvij - topic 6, 13 Gaurav - topic 7, 14, 15	Everyone, thursday
<b>Task 3</b>	Start prototyping	Start coding next week after meeting	Everyone
<b>The next meeting will be held</b>			18/09
<b>These minutes have been agreed by all attendees</b>			Signed by chair

## Meeting 4

## Meeting Minutes

<b>Subject</b>			
Hack Elite - Fourth Group Meeting			
<b>Date, Time (duration) and Venue</b>			
18/September/2020 05:00 – 06:00 pm Zoom			
<b>Attendees</b>			
Gaurav, Shashwat, Arjun, Liang Jing, Abby, Mac, Divvij			
<b>Chaired by Vaish Arjun</b>			
<b>Last meeting minutes have been reviewed</b>	Yes		
<b>Progress Updates</b>			
Task	Problem/Issue/P rogress	Solution/Action	Taken by & deadline
<b>Task 1</b>	System Requirement Specification	Complete	

<b>Task 2</b>	Quality Plan	Complete	
<b>Task 3</b>	Prototyping roles	Front end – Shashwat Back end – Gaurav NLP – Mac, Divvij Face Recognition – Arjun Games – Abby Therapy – Liang Jing  Main features to be implemented by the next lab. UI, Chatbot, backend, face recognition (maybe)	Everyone 8/10
<b>The next meeting will be held</b>			24/09
<b>These minutes have been agreed by all attendees</b>			Signed by chair

## Meeting 5

## Meeting Minutes

<b>Subject</b>			
Hack Elite - Fifth Group Meeting			
<b>Date, Time (duration) and Venue</b>			
<ul style="list-style-type: none"> <li>· 24/September/2020 10:30 - 12:30 am</li> <li>· SWLAB3</li> </ul>			
<b>Attendees</b>			
<ul style="list-style-type: none"> <li>· Gaurav, Shashwat, Arjun, Liang Jing, Abby, Mac, Divvij</li> </ul>			
<b>Chaired by Vaish Arjun</b>			
<b>Last meeting minutes have been reviewed</b>	Yes		
<b>Progress Updates</b>			
Task	Problem/Issue/Progress	Solution/Action	Taken by & deadline
<b>Task 1</b>	Project Plan	Division of roles: Abby - 1, 8 (5 with Shashwat) Liang Jing - 2, 9 Mac - 3, 10 Arjun - 4 Shashwat - 5 Gaurav - 6	Everyone 01/10

		Divvij - 7	
<b>Task 2</b>	Risk Management	Division of roles: Arjun - 1 Gaurav - 2 Divvij - 3	Arjun, Gaurav, Divvij 01/10
<b>The next meeting will be held</b>			01/10
<b>These minutes have been agreed by all attendees</b>			Signed by chair

## Meeting 6

## Meeting Minutes

<b>Subject</b>			
Hack Elite - Sixth Group Meeting			
<b>Date, Time (duration) and Venue</b>			
9/October/2020 5:30 - 6:30 pm Zoom			
<b>Attendees</b>			
Gaurav, Shashwat, Arjun, Liang Jing, Abby, Mac, Divvij			
<b>Chaired by Vaish Arjun</b>			
<b>Last meeting minutes have been reviewed</b>	Yes		
<b>Progress Updates</b>			
Task	Problem/Issue/Progress	Solution/Action	Taken by & deadline
<b>Task 1</b>	Chatbot	Bot complete, need to add more data	Divvij and Mac 13/10

<b>Task 2</b>	Frontend	Most pages complete, minor changes required for UI	Shashwat, Liang Jing 13/10
<b>Task 3</b>	Games	Mostly complete, minor changes required	Abby 13/10
<b>Task 4</b>	Facial Recognition	Further training and integration to front end required	Arjun 13/10
<b>Task 5</b>	Backend	Working to integrate with other features	Gaurav 13/10
<b>The next meeting will be held</b>			15/10
<b>These minutes have been agreed by all attendees</b>			Signed by chair

## Meeting 7

## Meeting Minutes

<b>Subject</b>			
Hack Elite - Seventh Group Meeting			
<b>Date, Time (duration) and Venue</b>			
<ul style="list-style-type: none"> <li>• 26/October/2020 5:30 - 6:30 pm</li> <li>• Zoom</li> </ul>			
<b>Attendees</b>	<b>Non-Attendees</b>		
<ul style="list-style-type: none"> <li>• Gaurav, Shashwat, Arjun, Abby, Mac, Divvij</li> </ul>	<ul style="list-style-type: none"> <li>• Liang Jing</li> </ul>		
<b>Chaired by Vaish Arjun</b>			
<b>Last meeting minutes have been reviewed</b>	Yes		
<b>Progress Updates</b>			
Task	Problem/Issue/Progress	Solution/Action	Taken by & deadline

<b>Task 1</b>	Chatbot	Shorter replies	Divvij and Mac 28/10/2020
<b>Task 2</b>	Frontend	1. Incorporate the change made by abby 2. Profile page and navbar 3. Changing the facial recognition button	1. Liang Jing 2. Shashwat, 3. Liang Jing 27/10
<b>Task 3</b>	Video Demo	Complete the video demo	1. Arjun 13/10
<b>Task 4</b>	Test Cases	Complete test cases (Template on drive)	1. Everyone 28/10
<b>Task 5</b>	Pear Evaluation	Complete Pear Evaluation	1. Everyone 13/10
<b>The next meeting will be held</b>			15/10
<b>These minutes have been agreed by all attendees</b>			Signed by chair

## Presentation Slides

**MindSpace**  
**The AI Chatbot**

CZ3002: Advanced Software Engineering

Group Name: HackElite

# Team Members



NAME	MATRIC NO.	ROLE
Vaish Arjun	U1822157E	Project Manager
Kiran Mac Milin	U1823120L	Quality Manager
Roxas Abby Maurea Imus	U1822611H	Quality Assurance Engineer
Koh Liang Jing	U1820617C	Release Manager
Asok Kumar Gaurav	U1823850F	Lead Developer
Arya Shashwat	U1822436J	Front-End Developer
Chandna Divvij	U1822949J	Back-End Developer

## Outline of Presentation



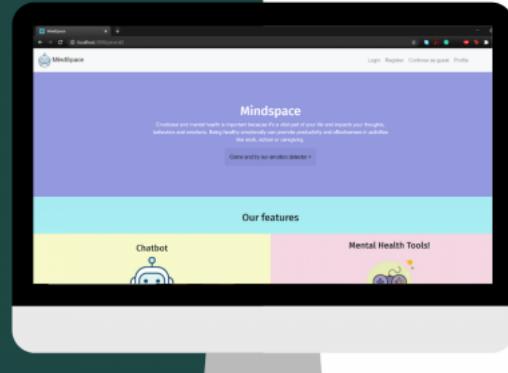
- PRODUCT INTRODUCTION
- DESIGN FOR MAINTAINABILITY
- SOFTWARE QUALITY ASSURANCE
- PROJECT MANAGEMENT
- RISK MANAGEMENT

## PRODUCT INFORMATION

1

What is MindSpace?

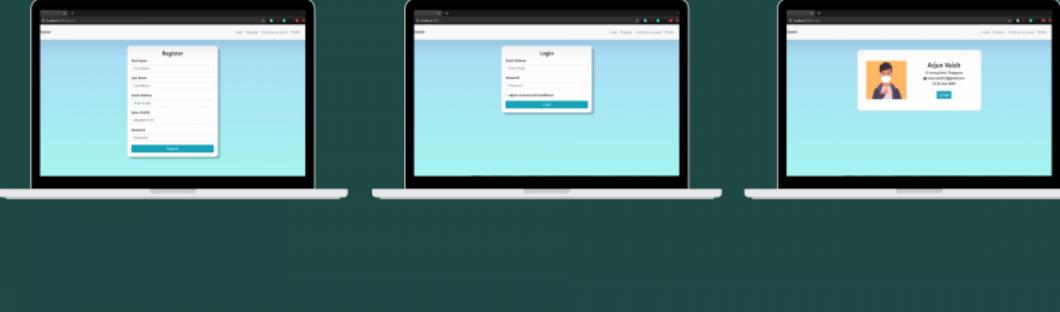
A web application for people to improve their mood and talk to an AI chatbot about mental health issues.





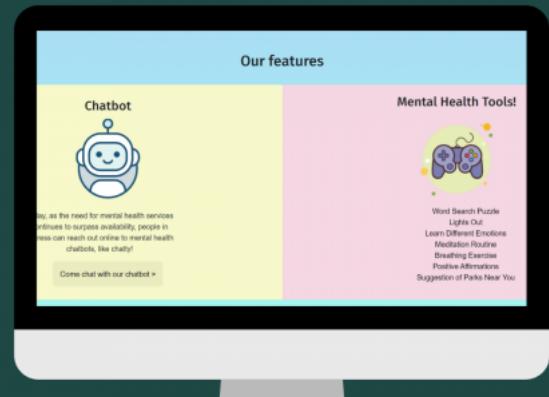
## Features

Register, Login, Profile



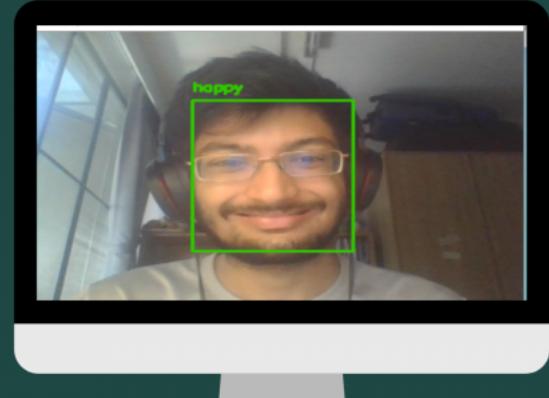
## Features

Overview



## Features

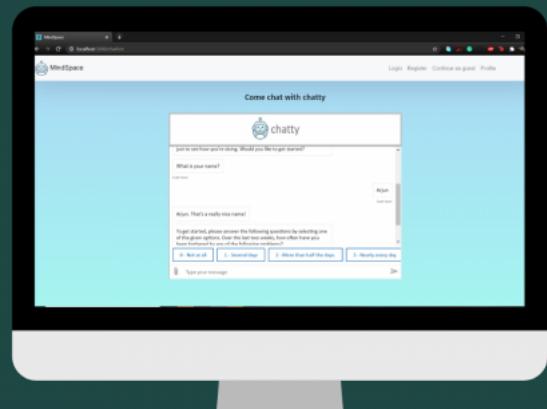
Facial Emotion  
Recognition





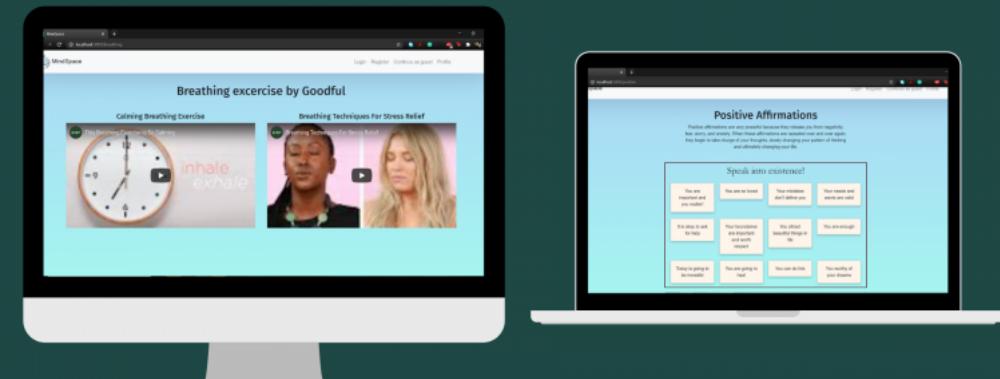
# Features

## Chatbot



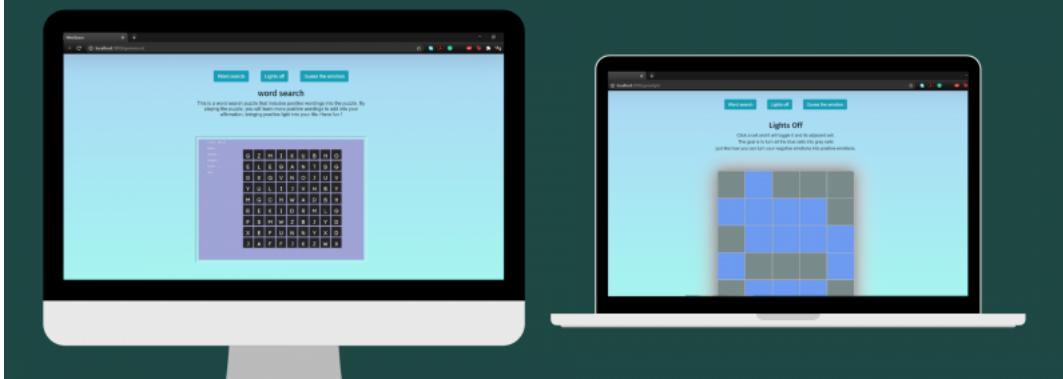
# Features

## Mood Lifting Exercises



# Features

## Mood Lifting Exercises



2

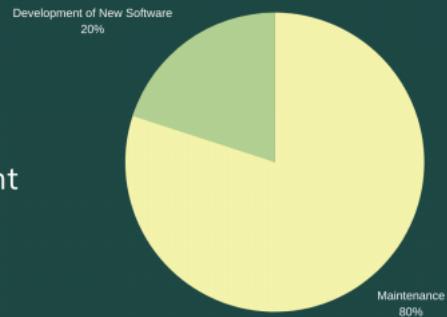
## DESIGN FOR MAINTAINABILITY



## Cost



80% - Maintenance  
20% - New Software Development



## Technical Considerations



**Front End:** ReactJS & Bootstrap

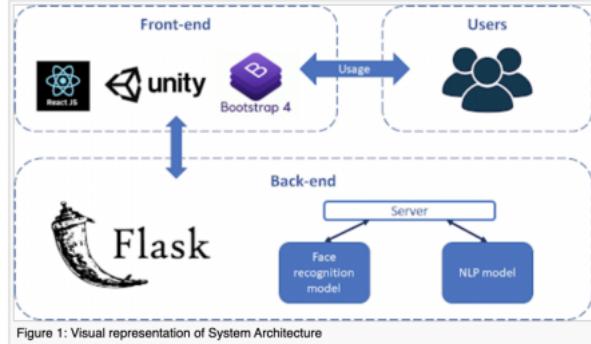
**Back End:** Flask

**AI & Machine Learning:** OpenCV & Tensorflow

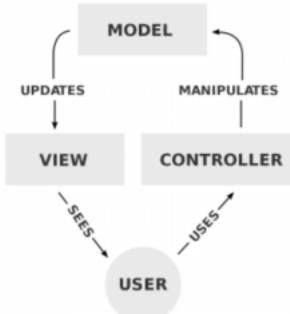
**NLP:** MS LUIS & Azure Bot Framework

**Game Development:** Unity

# Technical Considerations



## Design Pattern - MVC Structure



- Faster Development Process
- Facilitates Multiple Views
- Modification does not affect entire Model
- SEO Friendly Platform

## Software Management

SCM Tools





# Software Maintenance

## CORRECTIVE MECHANISM

Used NPM to update packages when library changed.

## PREVENTIVE MECHANISM

ReactJS allows easy change of components.

## ADAPTIVE MECHANISM

Virtual environment to run the back end files, including flask and TensorFlow. Allowed easy adaptation to changing users.

## PERFECTIVE MECHANISM

ReactJs or React is a declarative, efficient, and a flexible JavaScript library for building user interfaces.



3

# SOFTWARE QUALITY ASSURANCE

## Technical Review



- Defect detection process that has peers and technical specialist as part of the review process.
- Report is prepared with the list of issues that needs to be addressed.
- Following product assessments will be conducted:
  - Website user interface
  - Chatbot responses
  - Facial Emotion Recognition



# Management Review

By having a management review, we can ensure that all the goals and constraints are met.

## Metrics

### Product Evaluation

- Work product size (LOC, Pages)
- Preparation time before the review.
- Rework time to fix defects.
- Duration of the review process.

### Process Evaluation

- Improvement of review process effort and timings.
- The number of reviews.
- Estimated project time saved.
- Cost of quality metrics for review.

### Formal Reviews

- Defined entry and exit criteria.
- Checklist to be used by reviewers.
- Deliverables like reports and evaluation sheets.

# Software Quality Planning

Functionality

Maintainability

Reliability

Usability



# Software Reviews

The following Software Reviews will be assessed by the SQ Personnel:

- Project Plan Review
- Software Requirements Specification Review
- Software Design Document Review
- Code Review
- Test Plan Review
- Acceptance Review





# Testing

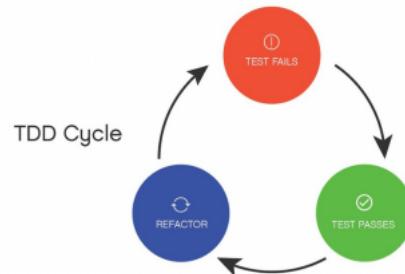
The following Tests will be carried out by the SQ Personnel:

- Unit Testing
- Integration Testing
- System Testing
- User-acceptance Testing

# Testing

Methodology:

- RED: Write a unit test and execute it. The test must fail (red) since there's no code under test yet.
- GREEN: Write code that passes (green) the previously written test.
- REFACTOR: Remove redundancy and enhance the written code to make it clean, reusable, and presentable. While refactoring, the functionality must not be changed, so the test still passes (keep it green).



4



# PROJECT MANAGEMENT

# Project Organization

Project Manager - Arjun Vaish  
 Quality Assurance Manager - Mac Milin Kiran  
 Lead Developer - Gaurav Asok Kumar  
 Release Manager - Liang Jing Koh  
 Quality Assurance Engineer - Abby Maurea Imus Roxas  
 Front-end Developer - Shashwat Arya  
 Back-end Developer - Divvij Chandra



## PROJECT ESTIMATION

### FUNCTION POINTS

Element	Complexity	Detail
Inputs	Low	User login/registration
	Medium	Facial Image
	Low	Messages to the Chatbot
	Medium	Game Input
Outputs	Medium	Emotion Prediction
	High	Chatbot Replies
Logical Files	Medium	User Authentication Database
	Medium	NLP Word bank
Inquiries	Low	Browse Videos
	Low	Browse Games
Interfaces	High	Backend Connection
	Medium	Authentication Verification

## PROJECT ESTIMATION

### UNADJUSTED F.P

Characteristic	Low	Medium	High	
Inputs	2	$\times 3$	2	$\times 4$
Outputs	0	$\times 4$	1	$\times 5$
Inquiries	2	$\times 3$	0	$\times 4$
Logical Files	0	$\times 7$	2	$\times 10$
Interfaces	0	$\times 5$	1	$\times 7$
<b>Unadjusted FP</b>	12		40	17
<b>Total=L+M+H</b>	69			



# ADJUSTED F.P.



**Influence Factors = 14**

**Total Score = 41**

**Influence Multiplier = 1.06**

**Adjusted FP = 73.14**

**Lines of Code = 1463 LOC**



# DISTRIBUTION OF EFFORT



1990's Industry Data	Work Package	Distribution	Estimates
Preliminary Design 18%	Project Plan	9%	35.1
	Requirement Specification	9%	35.1
Detailed Design 25%	User Interface	7%	27.3
	Technical Architecture	11%	42.9
Code & Unit Testing 26%	Data Modeling	7%	27.3
	Code & Unit testing	21%	81.9
Integration & Test 31%	Online Documentation	5%	19.5
	Integration & Quality Assurance	31%	120.9
<b>Extrapolated total effort</b>			390
2% for project management			7.8
3% for contingency			11.7
<b>Total effort</b>			409.5



# ACTIVITY DEPENDENCIES

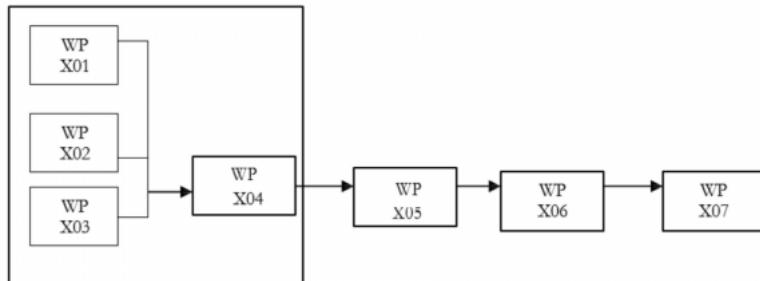


Work Package Number	Work Package Description	Duration	Dependencies
X01	Work Package	7 Days	-
X02	Requirement Specification	7 Days	-
X03	User Interface	7 Days	-
X04	Technical Architecture	12.1 Days	X01, X02, X03
X05	Data Modeling	7 Days	X04
X06	Coding & Unit Testing	16.2 Days	X05
X07	Integration & System Testing	16.2 Days	X06



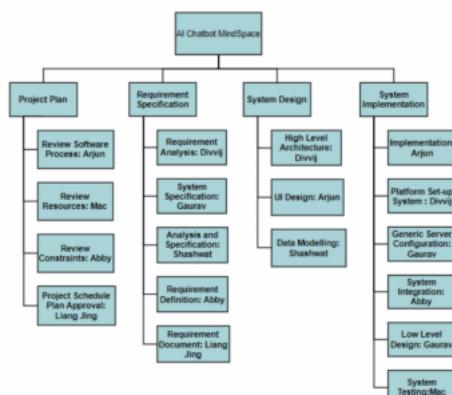
## PROJECT ESTIMATION

## ACTIVITY DEPENDENCIES



## PROJECT ESTIMATION

## WORK BREAKDOWN STRUCTURE



## PROJECT ESTIMATION

## PROJECT TIMELINE



## GANTT CHART



5



## RISK MANAGEMENT

Identifying any potential problems before it takes place and have a solution to address them and limit any negative impact.

Ability to mitigate any inevitable potential problems that are to occur.

Ensure that everyone on the team charts the right path forward when dealing with risks.

For recording purposes, every potential risk identified are logged into our Risk Management document.



## OBJECTIVES OF RISK MANAGEMENT



### RISK MANAGEMENT

## RISK MANAGEMENT APPROACH



### 1 IDENTIFY THE RISK

Expansive task and one that should occur throughout the SDLC.

### 2 ANALYZING THE RISK

Discuss with the team on the impact of the risk occurring, how the risk will arise, and its probability.

### 3 EVALUATING THE RISK

After analyzing all the risks, they need to be ranked in order of their impact severity.

### 4 TREATING THE RISK

The development team needs to brainstorm on the methods to eliminate or contain the risk.

### 5 MONITORING AND REVIEWING THE RISK

Some risks cannot be eliminated and hence, they must be monitored continuously.



# IDENTIFYING THE RISK



## EVALUATING THE RISK THROUGH 3 CATEGORIES:

- Technological Risk
- Usability Risk
- Estimation Risk



# OUR PROJECT'S IDENTIFIED RISKS



### TECHNOLOGICAL RISKS

- Poor Accuracy of the NLP model
- Poor Accuracy of Face Recognition model
- Slow performance of Face Recognition model
- Inability to integrate front end and back end Software break down with multiple users

### USABILITY RISKS

- UI may not be easily learnable by the users.
- Chatbot may not be able to provide the desirable answers to users.

### ESTIMATION RISK

- Wrong estimation of time to complete project.



# ANALYZING THE RISK



Qualitative Risk Analysis was the method that we adapted for our project.

We assess our identified risks by its probability of occurring and its impact.

Probability	Impacts				
	Very low (1)	Low (2)	Medium (3)	High (4)	Very high (5)
Very high (5)	5	10	15	20	25
High (4)	4	8	12	16	20
Medium (3)	3	6	9	12	15
Low (2)	2	4	6	8	10
Very low (1)	1	2	3	4	5



## ANALYZING THE RISK



RISK MANAGEMENT

Probability	Impacts				
	Very Low (1)	Low (2)	Medium (3)	High (4)	Very High (5)
Very High (5)					
High (4)					
Medium (3)				Poor Accuracy of NLP model (12)	
Low (2)			User Interface Low Usability (6) Problems co-ordinating schedule amongst group (6)	Poor Accuracy of Face Recognition Model (8)	
Very Low (1)			Slow Performance of Face Recognition Model (3)	Inability to Integrate Front-End and Back-End (4) Chatbot not able to provide required solution to the user (4)	



## EVALUATING THE RISK



RISK MANAGEMENT

After analyzing the Risks, we need to evaluate it according to its severity.

### RANKING OUR RISKS

- Poor Accuracy of NLP model (12).
- Poor Accuracy of Face Recognition Model (8).
- User Interface Low Usability (6).
- Problems Coordinating schedule amongst the development group (6).
- Inability to Integrate Front-End and Back End (4).
- Chatbot not able to provide required solution to the user (4).
- Slow Performance of Face Recognition Model (3)



## TREATING THE RISK



RISK MANAGEMENT

### FOUR WAYS WE CAN TREAT THE RISKS

1. Avoidance
2. Mitigation
3. Acceptance
4. Transfer



RISK MANAGEMENT

# MONITORING AND REVIEWING THE RISKS



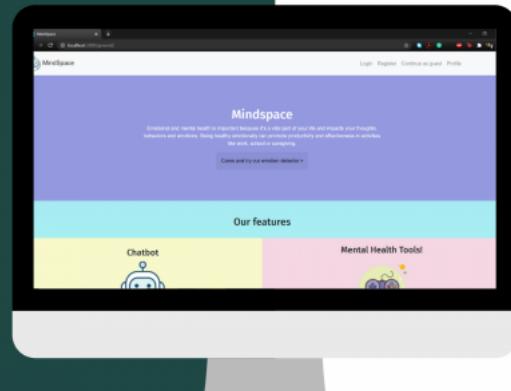
## CONTANTLY IDENTIFY, ANALYZE AND PLAN FOR NEW RISKS

1. Timeline Planning & Task Decomposition
2. Project Risk reviews at all team meetings/ status meetings.
3. Regular Test runs

## MONITORING PREVIOUSLY IDENTIFIED RISKS

1. Re-prioritize risks depending on probability and complexity.
2. Perform Quality Control.

## DEMO



# THANK YOU! ANY QUESTIONS/ COMMENTS?

CZ3002: Advanced Software  
Engineering

Project: MindSpace

Team: HackElite

Retrieved from "http://155.69.100.27/3002S12021\_HackElite/index.php?title=Main\_Page&oldid=497"

- 
- This page was last modified on 29 October 2020, at 03:09.
  - This page has been accessed 517 times.