General Plan

1. **Personal information**
    - Title : Kanban Board
    - Name: Linh Ngo
    - Student Number: 906502
    - Degree Program:
    - Year of studies: Year 1
    - Date: 17/02/2021

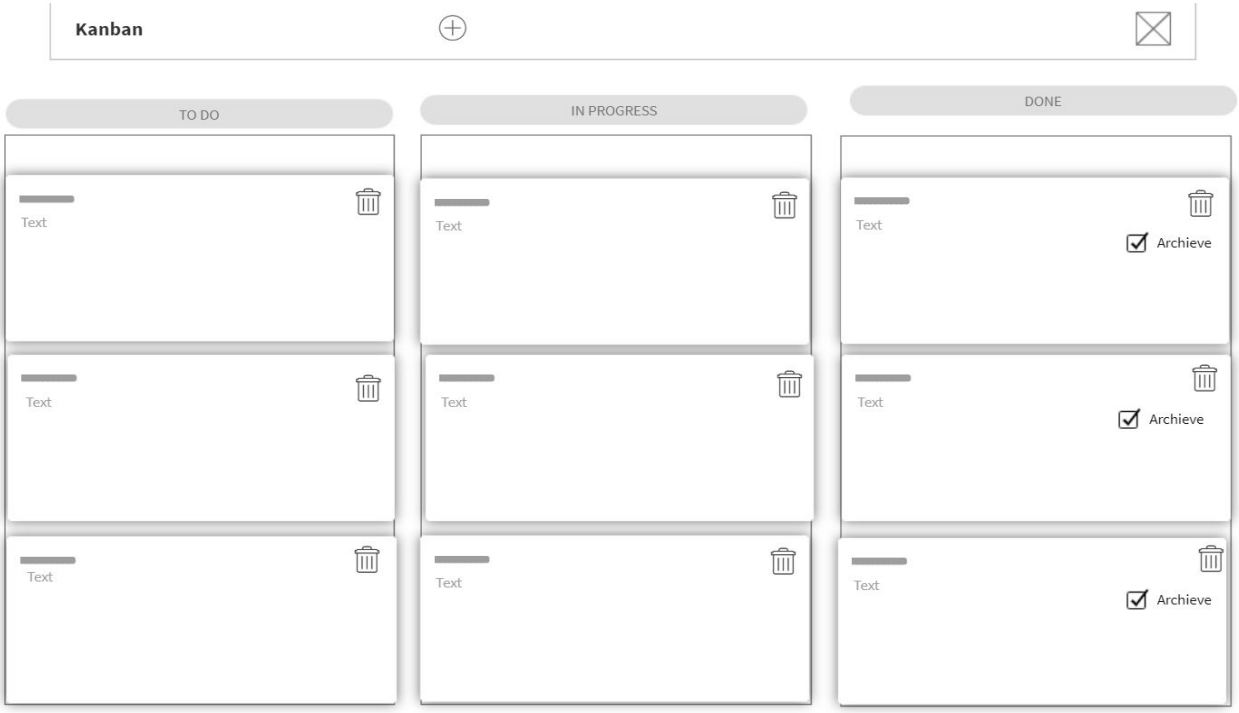
2. **General Description**

The goal of the project is to create a Kanban board with the graphical user interface which allows users to create and use Kanban Board and helps them in managing their personal tasks.

- The basic structure of the Kanban board has three columns: to-do, in progress, done.
- Some of the function that the users can do with the Kanban board is:
    - Drag-and-droppable card lists and cards between 3 columns with graphical user interface.
    - Add and remove cards to / from the board.
    - Archive and return cards from an archive pile.
- Cards include text content and time tracking (to-do, in progress, done).
- Card data is saved in and restored from external files. (human readable format)
- Cards have tags and can be filtered based on tags.
- The program supports multiple boards for a single user.

3. **Draft User Interface**

- The program communicates with the user through the graphical user interface. I intend to use the ScalaFX library to implement the GUI as the course provides some materials that I think might be helpful for my project.
- The user can add, remove, or archive cards by clicking into the buttons. ScalaFX offers an easy-to-use button component, so I am thinking of using them for GUI.
- TextField is an UI control that provides the user with an editable text field, by which the program can get inputs (texts) from users.
- The GUI will have three columns that represent three important phases of the board, and each rectangular box will be cards that users add to the board.



4. **Files and files format**
    - I am thinking of using a human readable format which is similar to the course exercise, to save datas of the cards.
    - Different names of the cards and properties of cards will be listed after the chunk header.
    - The character # must be the first character of the line to interpret the line as a block identifier between different cards.
    - The format is aimed to read one line at a time, as Scala library offers a compact way to read lines from a file.