Flokey - Synopsis

By Raunak M, Aryan N and Vishal S 12C

# Objective of Flokey:

Flokey is a notes and personal data organizing program combining the r*aw power of data organization and filtering using databases* with an *easy User Interface for taking notes*.

Born primarily out of a need to organize Organic Chemistry notes, Flokey is especially good at organizing webs of data that need to be filtered and sorted in many different ways.

# Advantages and Unique Selling Points of Flokey:

* **Tags**: Tag data with custom tags and sort + filter based on those tags. Example:  
  Tag important reactions and easily filter based on those tags before an exam.
* **Hierarchy**: Create collapsible headings within headings and so on to create a hierarchy similar to a flowchart to visualise your data/notes.
* **Extremely malleable**: organize and reorganize data on demand. Example:

Data in Chem 1: organized by *1) Type of Compound 2) Reactions of Compound*, whereas in Chem 2 it’s the reverse order. Flokey allows you to do this by dragging and dropping to rearrange headings.

## Chem 1 Hydrocarbons

##### Methods of Preparation

##### Properties

#### Alkyl Halides

##### Methods of Preparation

##### Properties

#### Alcohol

##### Methods of Preparation

##### Properties

### Chem 2

#### Methods of Preparation

##### Hydrocarbons

##### Alkyl Halides

##### Alcohol

#### Properties

##### Hydrocarbons

##### Alkyl Halides

##### Alcohol

* **Personal Wikipedia**: Note pages are connected to other relevant note pages, creating an easily traversable document.
* **Lightweight and Free**: Programs like Word, OneNote and flowchart programs like draw.io, etc. are considerably heavy programs and not ideal for taking large amounts of notes. Note taking programs with comparable features require subscriptions and are considerably more expensive.

# Hardware, Software Component:

* Hardware Components:
* Software Components:
  1. Python + MySQL: To store links to data and tags + Use SQL to search and filter through data + Django: To develop a back-end for the website.

## SQL + Python : backend

* **Link to data** is identified by **tags**. Ex, Reaction A has tags: Alkyl Halide, Alcohol and Methods of Preparation, Free radical. 0 if tag is not applicable, and 1 if tag is applicable.
* **Tags are grouped together** => RX, ROH, etc are Functional Groups, Methods of Preparation and Properties are Reactions.
* **Group of tabs has a hierarchy** => First organized by Functional Group then MoP and Prop. And vice versa

=>Data Mapping

| Link | Alkane | Alkene | Alkyne | RX | ROH | ROR | Free radical | MOP | Property |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 |
| B | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |

=>Tag Groups

| Alkane | Functional Group |
| --- | --- |
| Alkene | Functional Group |
| Alkyne | Functional Group |
| RX | Functional Group |
| ROH | Functional Group |
| ROR | Functional Group |
| MOP | Reaction |
| Properties | Reaction |
| Free radical | Mechanism |

=>Group Order

| Functional Group | 1 |
| --- | --- |
| Characteristic | 2 |
| Mechanism | Null (we don’t create a subheading) |

* 1. Tkinter: For frontend

# Shortcoming / Implement Later, Scope and Additional features

* Shortcomings
  + 1. UI is simple, not too fancy
    2. Security vulnerabilities
* Implement later
  + 1. Security fixes
    2. Online and available to others
    3. Support organic chemistry diagrams similar to ChemDoodle