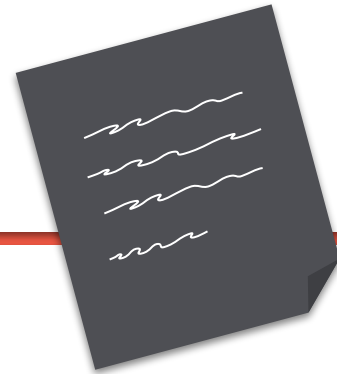
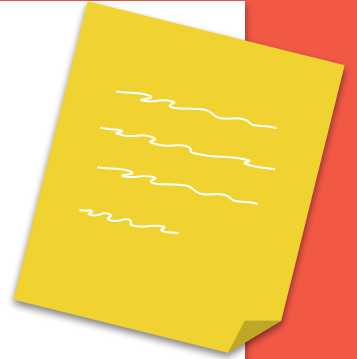


# PyPASS

## Password Manager

Offline, Feature-rich and easy-to-use!

**Under the guidance of Prof. Bhaswati Sahoo**



# Table of Contents



**01**

## About the Project

Introducing the project , and it's inspiration.

**03**

## Technologies Used

Technical aspects and requirements.

**05**

## Sneak Peek

A look into the application and its features.

**02**

## Project Objectives

Purpose of making this application.

**04**

## Architecture

Basic app design and use-case diagram.

**06**


## Our Team

People involved.



# ABOUT THE PROJECT

# 01



PyPASS is a Password manager which helps people manage their passwords offline. People can be irresponsible as they forget their password, design bad passwords, or are not interested in security, using the same passphrases for each and every account is dangerous as it creates one point of failure.

We, here, are tackling some of the everyday problems including:

1. *Cyber security concerns*
2. *Absence of strong and secure passwords*
3. *Hard to remember those passwords*





# Introduction

## **What is a password manager?**

A password manager is an encrypted piece of software or a program that allows users to store, generate, and manage their passwords for local applications and online services, as well as other informations in one convenient location with one master password.

( In simpler terms, A book of all your passwords is locked by a master key that only you know. )



Not only it helps to keep your sensitive data and credentials safe, but it can also generate unique and strong passwords. In this case, you don't have to keep reusing the same ones across your devices and platforms.



# Project Objectives

02



How is the product going help us?



# Why Password manager!

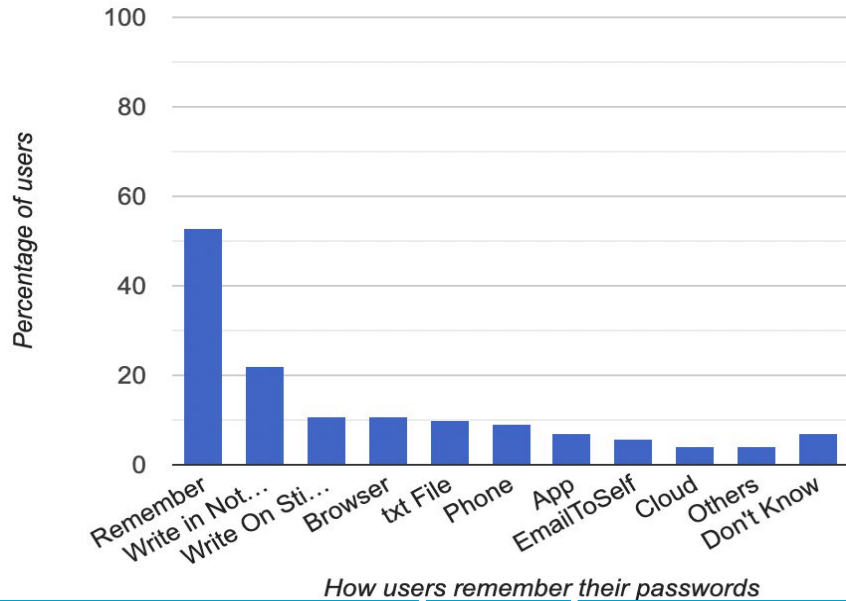
The aim is to make a GUI based python program to have easy access to your data. You just need:

- To remember your master password
- Not use that password anywhere else.

All your details will be stored on your device in an encrypted CVS file.



Kaspersky 2016 Survey with 12500+ responses



# Kaspersky Survey

Important takeaways from the survey:

- 1) More than 50% of users use unsafe methods of saving their passwords
- 2) Less than 10% actually use a dedicated password manager app.



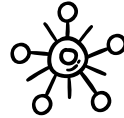
# Technologies Used

03 



## **IDE (VS Code)**

Provides  
comprehensive  
environment for  
development



## **Python 3.7**

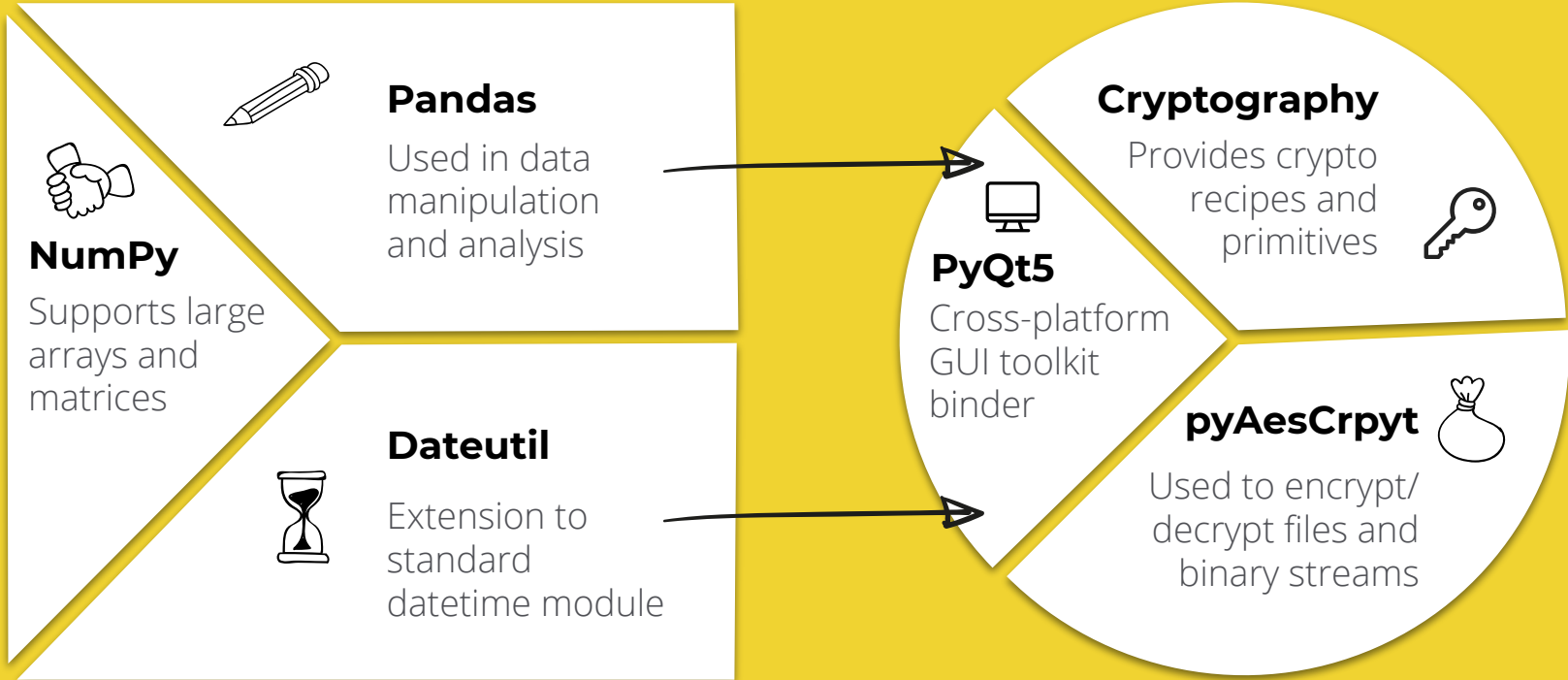
General  
purpose, high  
level, interpreted  
language



## **Encryption (AES 256) Hashing (SHA512)**

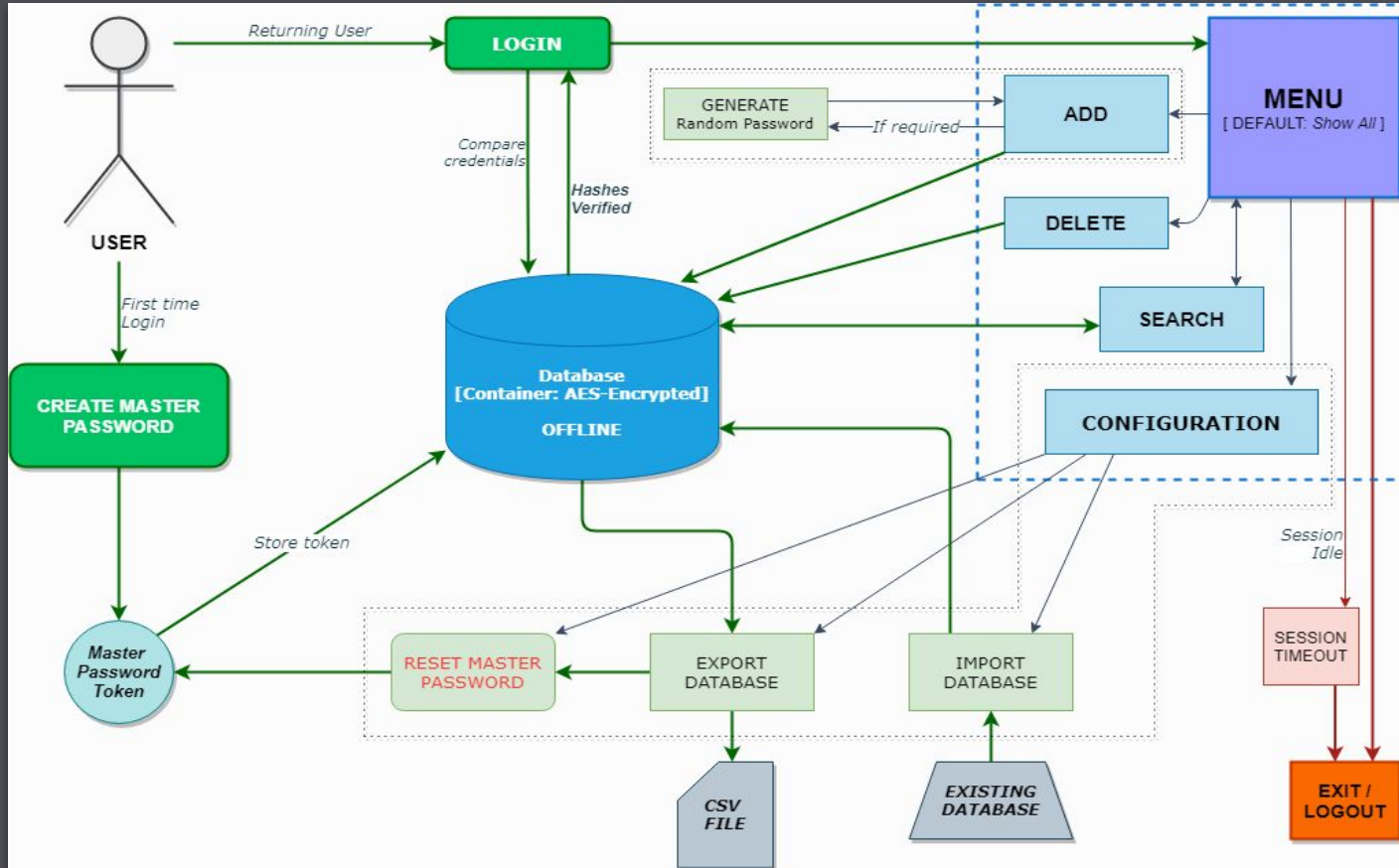
Encoding  
information,  
mapping  
arbitrary data

# Python Libraries Required



# Architecture

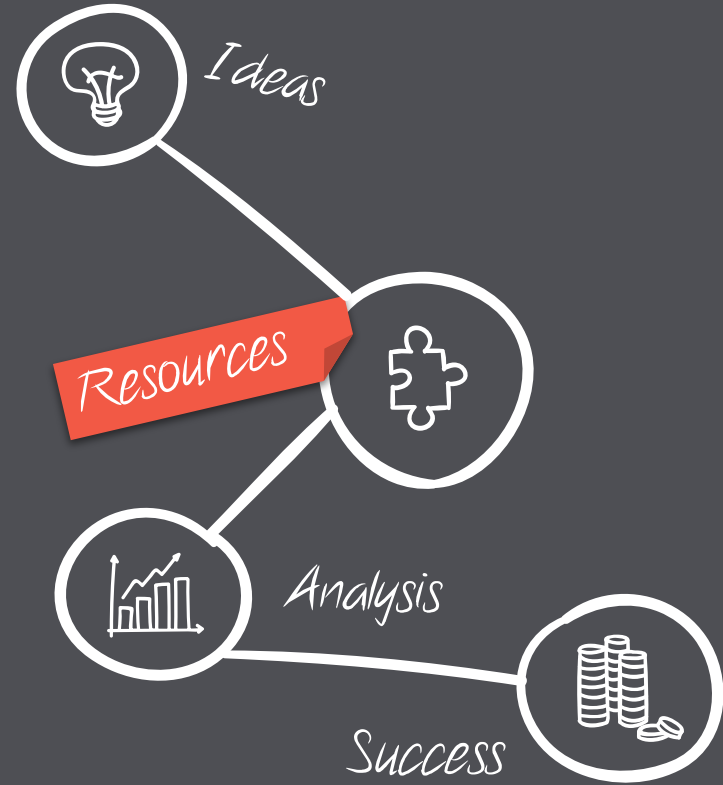
04



# 05

## Sneak Peek

A quick look at the key features of our application.



# Initial Login

**Master password:** The one password that will be used to access your details.

**Login:** Create Username and password that will be used to access the data

The image displays two windows from the PyPASS application. The top window, titled "PyPASS - Create Master", is for creating a master password. It features three input fields: "Username" with the value "Dex", "Password" with masked characters "•••••", and "Repeat Pass" with the value "qwerty". A "Save" button is located at the bottom right. The bottom window, titled "PyPASS 1.0.0 - Login", is for logging in. It has "Username" and "Password" fields, both containing masked input. The "Username" field has "Dex" and the "Password" field has "•••". There is a small "x" icon in the password field's clear button. An "OK" button is at the bottom. Below the button, a message reads: "Please fill the Username and password to LOGIN".

PyPASS - Create Master

Create the Master Password

Username Dex

Password •••••

Repeat Pass qwerty

Save

PyPASS 1.0.0 - Login

Welcome to < PyPASS >

Username Dex

Password •••

OK

Please fill the Username and password to LOGIN

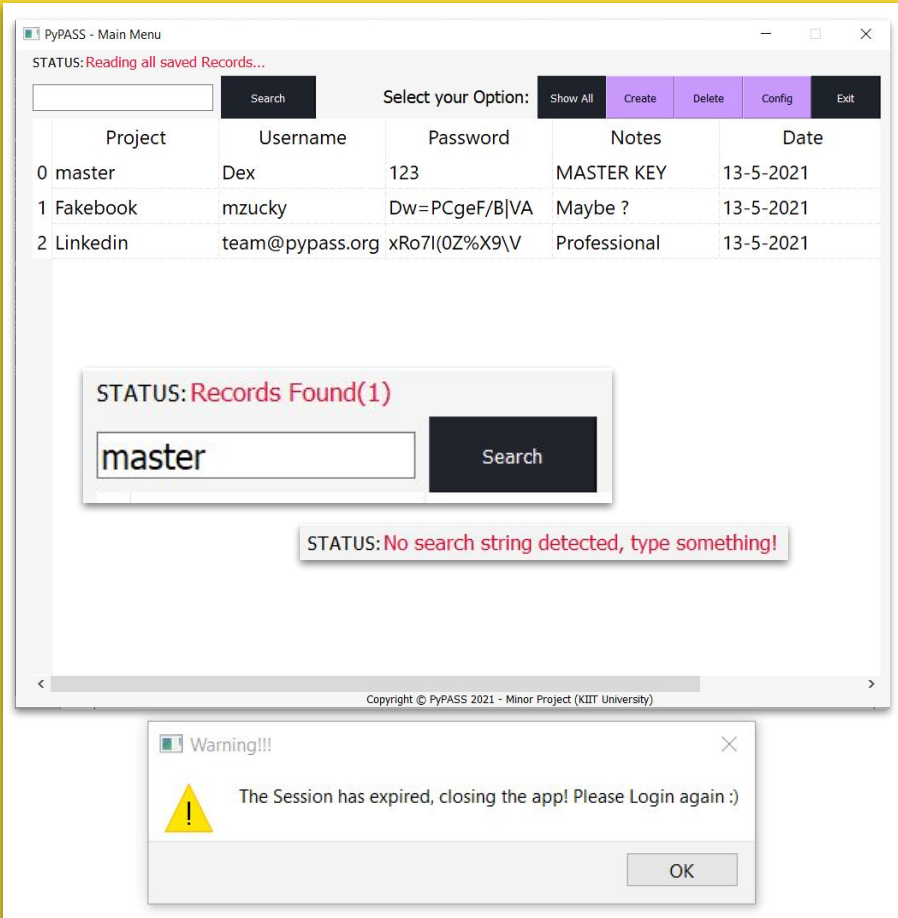
# Main Menu

## + Session Time Limit

**Search:** Look through the database for a specific keyword

**Show All:** Displays all the stored entries from the dataframe

**Session Timeout:** Logs out the user and exits the application after a preset time of inactivity in the app window



# Further Options

## ( Add/Delete Password + Config)

**Add entry:** Add an entry to the database.

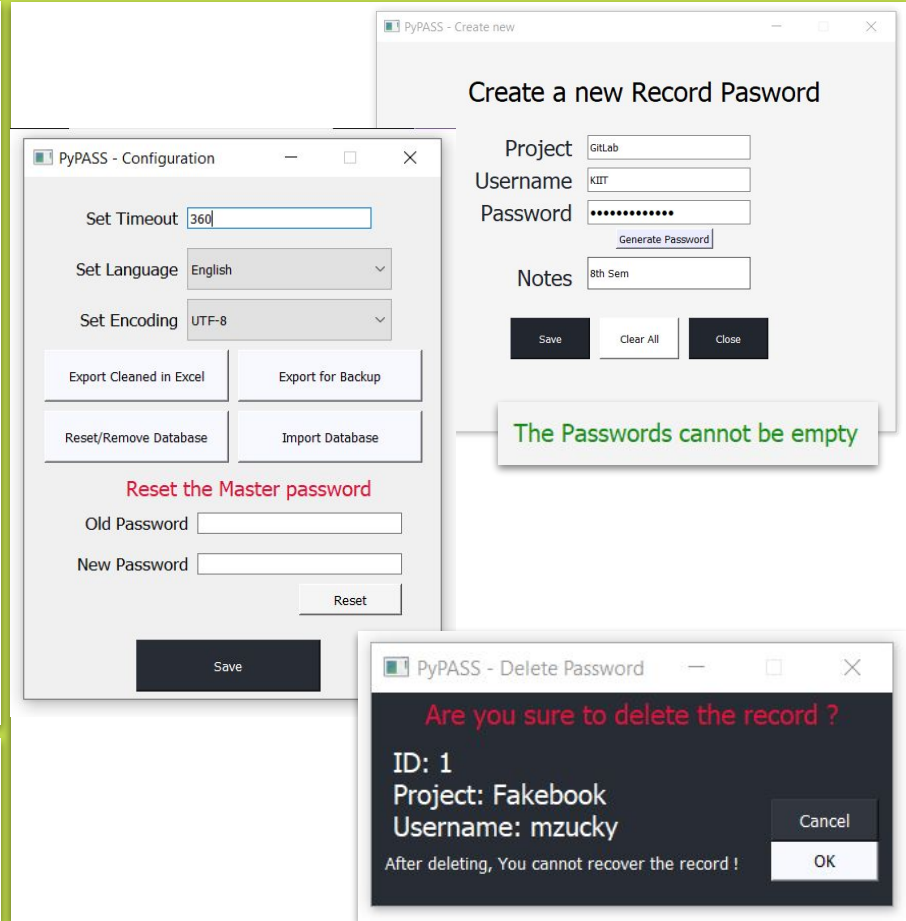
**Delete entry:** Delete the entry from the database

**Generate password:** Generates a random string of pre-specified length(12)

**Reset master:** Reset your master password and change it

**Export CSV:** Saves your data in a spreadsheet format.

**Import Database:** Reads data from a spreadsheet file and save the data in encrypted form.

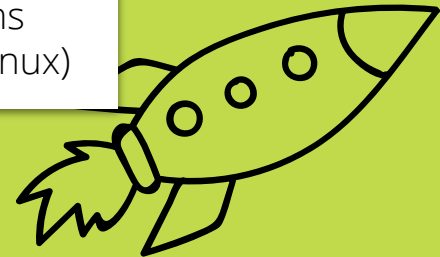


# Future Updates



Upgrading app security,  
adding features such as  
**2FA Authentication,**  
**Backup** and **Recovery.**

Converting this into a  
**cloud application** for  
easy access, compatible  
across all platforms  
(Windows, MacOS, Linux)





# Our Team

**Saket Pandey**

**1806413**

IT 2022  
KIIT University

**Tushar Abir**

**1806532**

IT 2022  
KIIT University



**Sambeet Pani**

**1806512**

IT 2022  
KIIT University

**Ashwin Anand**

**1806371**

IT 2022  
KIIT University

**Devyansh Singh**

**1806375**

IT 2022  
KIIT University

# Thanks!



CREDITS: This presentation template was created by **Slidesgo**, including icons by **Flaticon**, infographics & images by **Freepik**.

© Copyrights Reserved  
PyPASS 2021