iExtract iOS Album to Folder Conversion Tool

Team Information:

- **(Names: Roles):**
  - Sam Daughtry: Developer
  - Noah Gregie: UI/UX
  - Kevin Gustafson: Technical Director
  - Brendon Wong: QA Tester
- **GitHub Repo Link:** https://github.com/Gustakev/cs362-class-project
- **Communication Rules:**
  - We will communicate primarily through a Microsoft Teams group chat and our OSU associated emails. We may establish other means of communication amongst ourselves, but that is optional.
  - In terms of timeliness, as long as someone responds to a message sent via Teams or email within a few hours, it will be considered acceptable. However, there may be some times when quicker communication is expected and/or necessary to reach deadlines on time. During those times, it is preferable that team members respond to messages from other members within at least an hour, so that we may accomplish our goals on time.

Product Description:

- **Abstract:**
  - iExtract is a tool that helps people intelligently extract their photos and videos from their iPhones via reconstructing a folder structure and converting proprietary formats to common formats. This utility is dedicated to making iOS Photos app albums and collections easily convertible into folders en masse, which is not currently possible via any other tool than iMazing and a few other niche, monolithic tools. Additionally, iMazing has an expensive licensing model, as it provides many more features than just photo/video album to folder conversion, yet in a way that is not as focused.
- **Goal:**

    - The goal of this application is to solve the user experience issue regarding the either tedious, expensive, or iCloud-centric method of photo album and collection to local storage transfer that is available to iOS users at the moment. This app will facilitate a completely local transfer of albums and collections to local storage (converted into a folder-based format) on a user's PC, given they have created an unencrypted backup of their device, including their camera roll items.
- **Current Practice (Local Transfer Options):**
    - **iMazing:** The camera roll related features of iMazing closest resemble what iExtract is aiming to accomplish, but the issue with iMazing is that since it includes so many more features in addition to those ones, it has a high price tag and is subscription-based. This business model is not ideal for users that simply wish to convert their albums and collections into folders without having to manually reconstruct them from a raw dump, which is the kind of import method that the Windows Photos app provides for free.
    - **iTunes (Manually Copying User Created Folders of Photos/Videos from the iOS Files App):** This is a free method that anyone can use. However, the issues with this method are that it is entirely manual, tedious, wastes storage space on your phone during the process, and is prone to conversion failures, especially regarding live photos, which get converted into still images if not manually converted to videos before exporting to the Files App. Additionally, the album/collection structure must be entirely manually recreated by the user in order to preserve it, which is extremely tedious and infeasible for large numbers of albums, especially since in order to copy an album to your Windows computer, for example, you must first create a folder of the same name in the iOS Files App, then you must ensure you convert all live photos to videos and re-add them to the original album, and then you must copy all of the items from said album into the folder of the same name in the Files App. After that, you must use iTunes to access the folder in which you stored these items and copy its contents to your computer.
- **Novelty:**

    - This application provides a cheaper and more focused method to users who want to avoid iCloud and backup their important albums and collections en masse, locally, in an automated manner. This application, unlike the alternatives, will not require an account, will be entirely free (or very cheap compared to the $29.99 per device price of iMazing), and will have power-user features, such as album blacklisting and whitelisting, ensuring users only back up the photos they want. It will be functional cross-platform, as it will use Python, which will allow it to be used on Windows, Linux, and macOS, or anywhere that has a Python interpreter for the version we decide to use. Additionally, unlike some tools (iTunes) which only allow users to back up their items via a tedious process if their phone is still functional, this application will work on unencrypted/decrypted iOS backups, not active phones, which allows for much more flexibility in terms of use-cases.
- **Effects:**
    - If we are successful in creating this app in the given timeframe, there will finally be a free (or very inexpensive) tool that may be used by anyone to locally backup their iOS devices' photos and videos without the friction that currently exists, such as cost and tedium.
- **Technical Approach:**
    - We are planning to use Python and libraries that work across Windows, Linux, and macOS in order to create this app in such a way that it is functional across all the major desktop platforms. The main user experience will be a command line interface, but in order to reduce friction for less technical users, we will include a detailed list of commands that will be shown on-screen once the user enters "help". Additionally, we will be utilizing SQLite in our program to query the database iOS uses to track which albums and collections media belongs to.
- **Risks:**
    - The most significant risk to the success of this project is that it will be too difficult to understand the format in which Apple stores and organizes camera roll files, in regard to the metadata and album/collection structure that we wish to preserve using the folder-based backup system. To mitigate this risk, we will do

research into the existing documentation around how iOS stores camera roll files and their metadata, which is publicly available but perhaps difficult to understand.

Major Features (4+):

1. **Mass Album -> Folder Conversion**
2. **Mass Collection -> Folder Conversion**
3. **Folder & Collection Blacklisting (And Whitelisting) System**
4. **Proprietary File Type -> Common File Type Conversion Options**

Stretch Goals (2+):

1. **Extract Folders of Files from iOS Files App & Incorporate Blacklist (And Whitelist) Features**
2. **Custom Folder Creation Via SQL Filtering**
3. **Implement a GUI as a CLI Alternative**
4. **AI Descriptions of Content**

Notes:

- The major features should constitute a minimal viable product (MVP).
- You should aim to commit your living document to your project's repository latest by 1/15/2026, 11:59 PM.