TERNOPS DEMO 2

Celebrity Biographies &
their Achievements

CSCI-2040U

03.20.2025

Agenda

Demonstration

Project Progress

Build Files

Version Control

Retrospective

Q & A

1 Demonstration

```
to_csv(file_path, index=Fal

eck if the path is a file or
os.path.exists(images_path)
 if os.path.isdir(images_pat
     # If it's a directory r
     try:
         shutil.rmtree(image
         print(f"Successfull
     except Exception as e:
         print(f"Error delet
 elif os.path.isfile(images_
     # If it's a file remove
     try:
         os.remove(images_pa
         print(f"Successfull
     except Exception as e:
         print(f"Error delet

rint a success message
nt(f"Successfully deleted th

d some functions to ease GUI
y_image_to_folder(image_path
reate new folder if it doesn
path = r"src\\Data\\Images\\
not os.path.exists(newpath):
 os.makedirs(newpath)
```

2 Project Progress

```
pFrame = CTkFrame(mainFrame, width=
opFrame.pack(fill="x", side="top")

User info
serLabel = CTkLabel(topFrame, text=u
serLabel.pack(side="left", padx=40)
TODO Profile Picture
profilePicture = CTkImage(dark_image
profilePictureLabel = CTkLabel(mainF

f userAccount.get_user_name() == "Gu
   # Login / Register buttons
   loginButton = CTkButton(topFrame,
   loginButton.pack(side="right", pa
   registerButton = CTkButton(topFra
   registerButton.pack(side="right")
lse:
   # Logout button
   logoutButton = CTkButton(topFrame
   logoutButton.pack(side="right", p


 Add Celebrity
ddButton = CTkButton(topFrame, text=
ddButton.pack(side="right", padx=90)

 Searchbar
earchbar = CTkEntry(topFrame, width=
earchbar.pack(side="right", pady=20)

 Scrollable frame to house list of c
crollFrame = CTkScrollableFrame(main
crollFrame.pack(fill="both")

 Load the celebrities from CSV file
lebrities = functions.load_celebri
```

Must Have

- Add a homepage that displays trending celebrities.
- GUI element to filter celebrities by keyword.
- Implement functionality to filter celebrities.

Must Have

- GUI buttons to add, edit, and delete.
- GUI buttons to log in & log out.
- GUI text field to search for celebrities.

Must Have

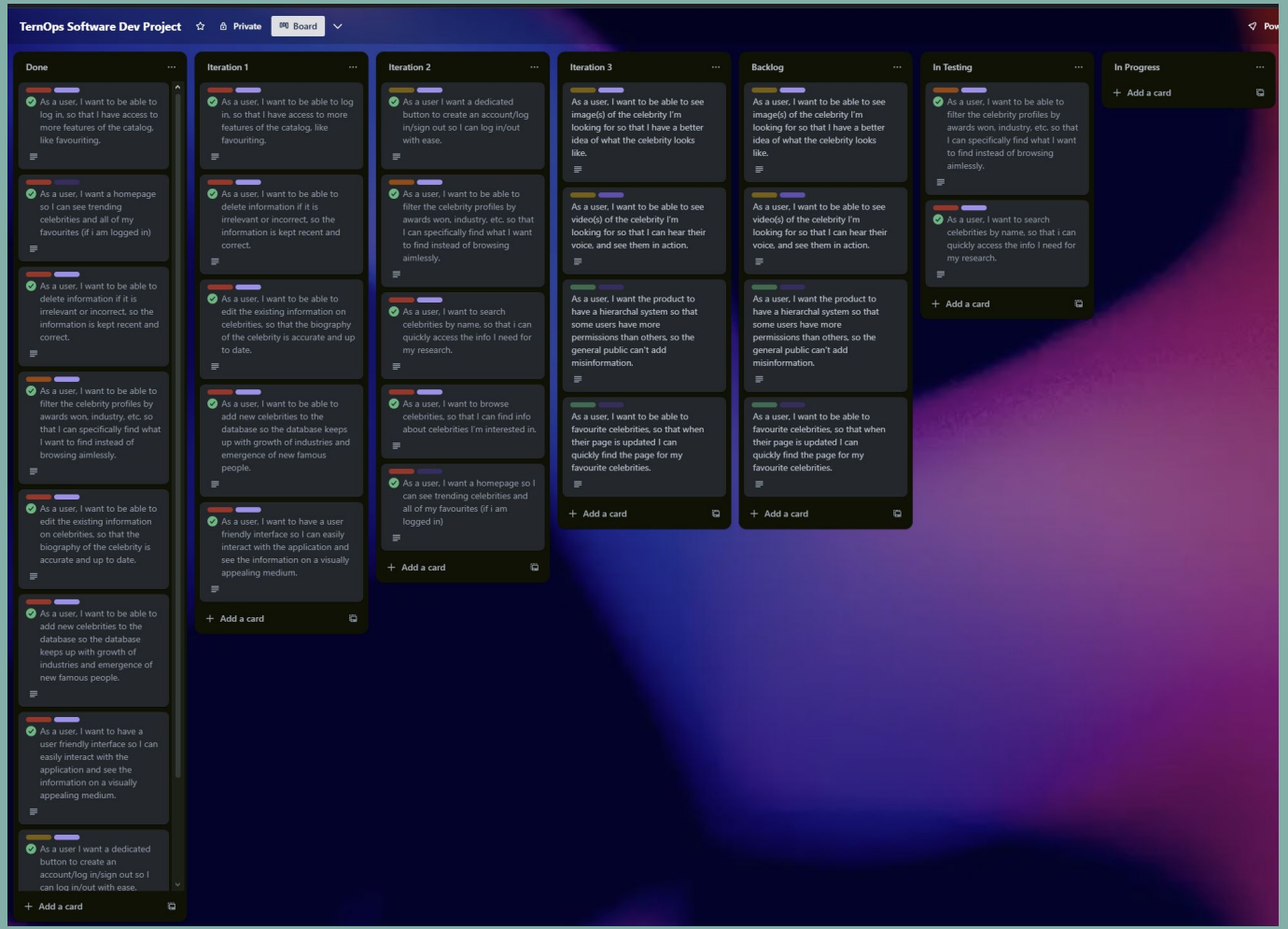
- Implement functionality to search for celebrities.
- Implement functionality to delete celebrities/info.
- Implement functionality to edit celebrities info.

Iteration 3 tasks

- **View Celebrity Images & Videos.**
- **Log in/ Log out admin permissions.**
- **Filter celebrity profiles by awards, etc.**
- **Favourite celebrities.**

```
if ! command -v python &>/dev/null; then
    echo "Python3 is required but not installed. Please install Py
    exit 1
fi

# Step 2: Check if pip is installed
if ! command -v pip &>/dev/null; then
    echo "pip is required but not installed. Please install pip."
    exit 1
fi

# Step 3: Set up the virtual environment
echo "Setting up virtual environment..."
python -m venv venv

# Step 4: Activate the virtual environment
echo "Activating virtual environment..."
source venv/Scripts/activate

# Step 5: Install dependencies
echo "Installing dependencies from requirements.txt..."
pip install -r requirements.txt

# Step 6: Run the GUI application
echo "Running the application..."
python src/gui.py

# Step 7: Deactivate the virtual environment after running
deactivate
```

TRELLO BOARD



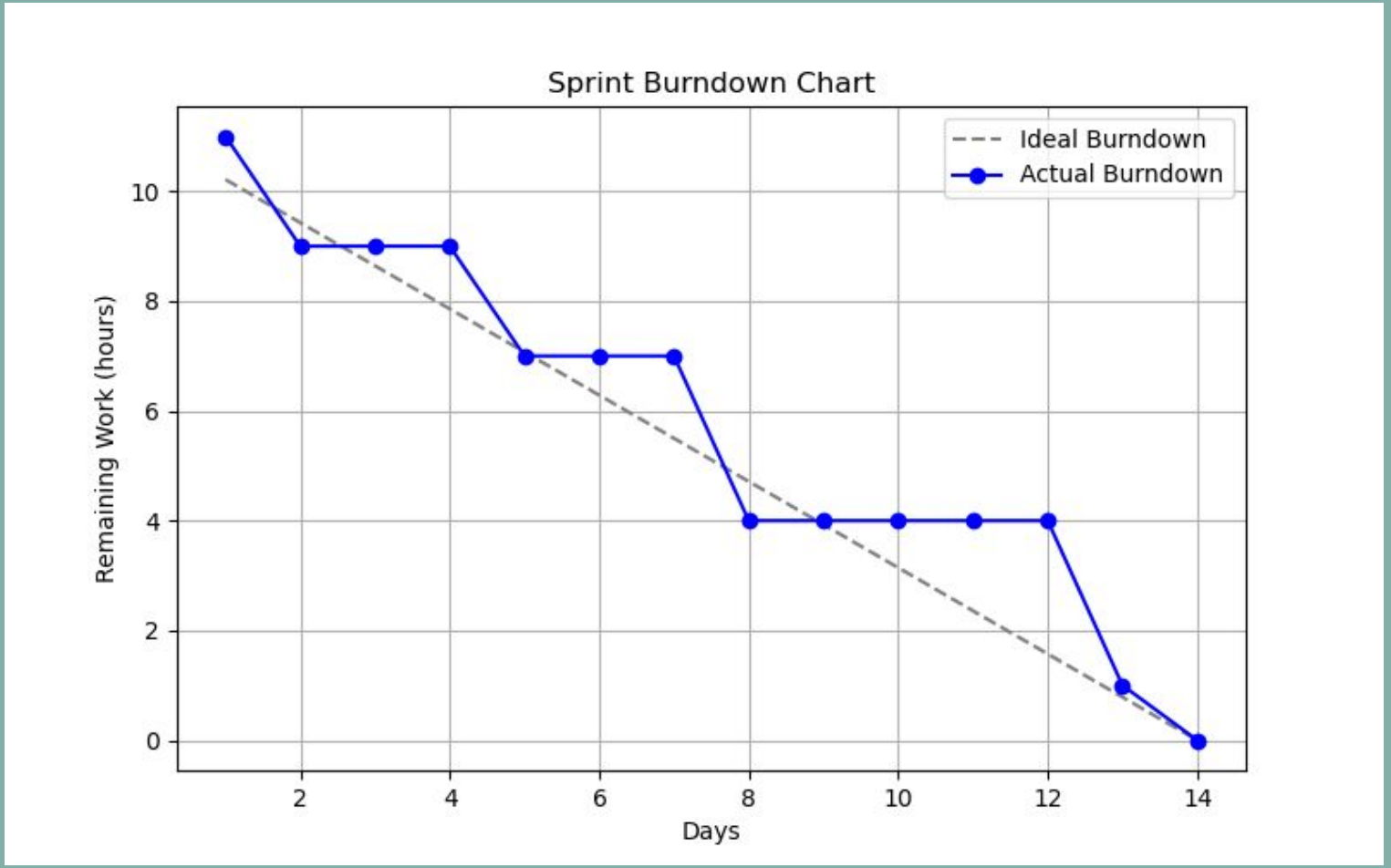
TERNOPS

VELOCITY CALCULATION

Iteration length = 14 days x 1.5 working hours = 21 hours
Total Estimated work = 11 hours = 7 days
Days required = 7 days / 0.7 (estimated velocity) = 10 days
True Velocity = 11 hours /21 hours = 0.5238 (52.4%)

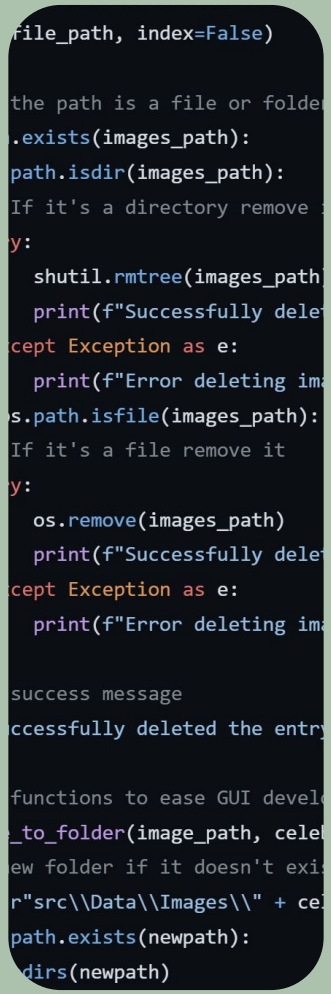
In this iteration, we reached a 52.4% velocity which is much faster than previously, this is because of the adjusted working hours per day. 8 Hours was inaccurate for the amount of time actually present after other responsibilities. This velocity is still low however midterms played a role throughout this iteration leading to us being slightly behind schedule.

BURN DOWN CHART



TERNOPS

3 Build Files



Build File

```
#!/bin/bash

# Exit immediately if a command exits with a non-zero status
set -e

# Step 1: Check if Python3 is installed
if ! command -v python &>/dev/null; then
    echo "Python3 is required but not installed. Please install Python3."
    exit 1
fi

# Step 2: Check if pip is installed
if ! command -v pip &>/dev/null; then
    echo "pip is required but not installed. Please install pip."
    exit 1
fi

# Step 3: Set up the virtual environment
echo "Setting up virtual environment..."
python -m venv venv

# Step 4: Activate the virtual environment
echo "Activating virtual environment..."
source venv/Scripts/activate

# Step 5: Install dependencies
echo "Installing dependencies from requirements.txt..."
pip install -r requirements.txt

# Step 6: Run the GUI application
echo "Running the application..."
python src/gui.py

# Step 7: Deactivate the virtual environment after running
deactivate
```

README File

Running the MVP

Basic setup to run the MVP:

To automatically run MVP using Git Bash, follow the steps below:

1. **Clone the Repository**
   Open a termianl and clone this repository to your local machine using the command:

```
git clone <repository-url>
```

2. **Open the Repository in terminal**
   Navigate to the project folder and open it in your terminal by the following command:

```
cd <repository name>
```

3. **Launch the MVP**
   In the terminal and run the following command to launch the MVP and automatically install all necessary libraries and install the required dependencies (it may take some time to run the MVP based on if your computer already has the necessary requirments or not):

```
./build.sh
```

To manually run MVP, follow the steps below:

1. **Clone the Repository**
   Clone this repository to your local machine using:

```
git clone <repository-url>
```

2. **Open the Repository in terminal**
   Navigate to the project folder and open it in your preferred IDE.

3. **Install Dependencies**
   Open a terminal and run the following command to install the required dependencies:

```
pip install -r requirements.txt
```

4. **Import Necessary Libraries**
   This will automatically install all the necessary libraries for the MVP to run.

5. **Run the GUI**
   Locate and open the `gui.py` file under the `src` folder.

6. **Launch the MVP**
   Run the `gui.py` file to start the MVP application.

4 Version Control

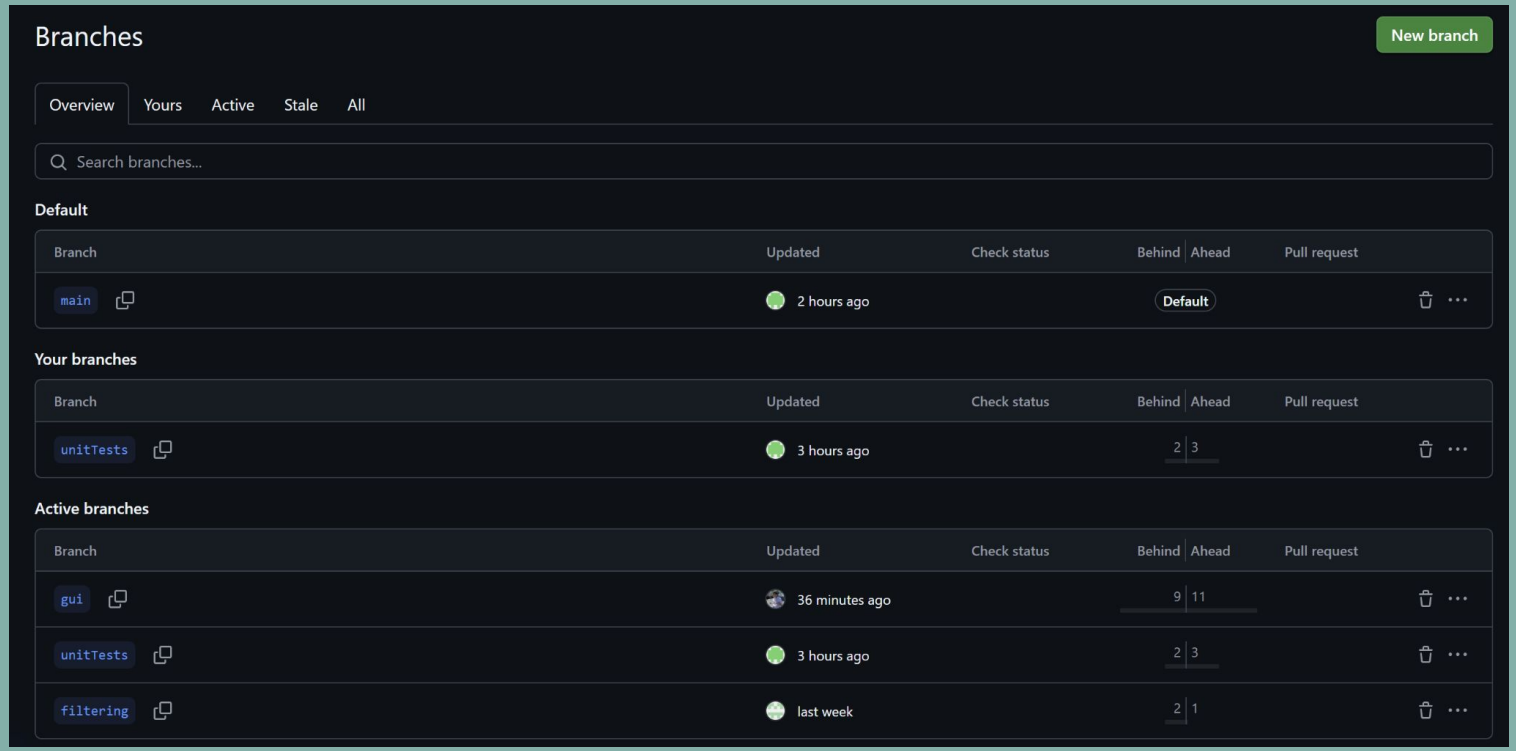
REPO COMMITS

TERNOPS

REPO COMMITS

TERNOPS

Branches



TERNOPS

5 Retrospective

ITERATION 2 RETROSPECTIVE

Went Well

- Better communication
- Ability to individually complete tasks during a busy time

Improvements

- More brief meetings
- More frequent pushing + pull requests to keep everyone on the same page

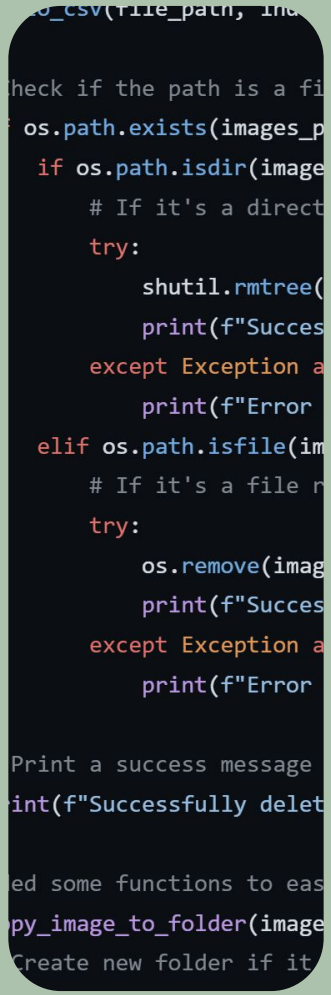
Challenges

- Writing tests
- Time management delays leading to crunch near the end

Resolution

- Ask clarification from team members
- Learn more about pytest

6 Questions and Answers



```
if userAccount.get_user_name() == "Guest":
    # Login / Register buttons
    loginButton = CTkButton(topFrame, text="Login", command=lambda: showSignInFrame("Login to Account"), width=60)
    loginButton.pack(side="right", padx=15)
    registerButton = CTkButton(topFrame, text="Register", command=lambda: showSignInFrame("Register Account"), width=60)
    registerButton.pack(side="right")
else:
```

MATTEO DEANGELIS GERALDO
RYAN COONES
TOMASZ PUZIO
ERICA PATEL
NOLAN TSANG

THANK YOU

TERNOPS