**Internship Report on ”File-Organizer”**

**Prepared by**

**Bishika Pokharel**

**Date: 11/20/2024**

*Executive Summary*

This report provides an overview of the **Industrial Internship** facilitated by **Upskill Campus** and **The IoT Academy** in collaboration with their industrial partner, **UniConverge Technologies Pvt. Ltd. (UCT)**. The internship revolved around solving a real-world industrial problem statement provided by UCT, and we were tasked with completing the project and delivering a comprehensive report within four weeks. My Project name is “**File-Organizer in Phyton**”.

## 1 Project Overview

My "File Organizer in Python" project aimed to develop a Python-based solution to tackle the challenge of managing disorganized files within a directory. The **File Organizer** automates file categorization into pre-defined folders based on file extensions or types, such as documents, images, videos, and more. The tool enhances productivity and efficiency by keeping directories organized and easy to navigate. Key functionalities of the File Organizer included:

1. **Automated File Categorization**: Files are automatically sorted into appropriate folders based on their extensions.
2. **Customizable Configuration**: Users can define and modify folder categories to suit their needs.
3. **Support for a Wide Range of File Types**: The tool handles documents, images, videos, audio, archives, and more.
4. **Scalability**: The program can handle both small and large volumes of files efficiently.

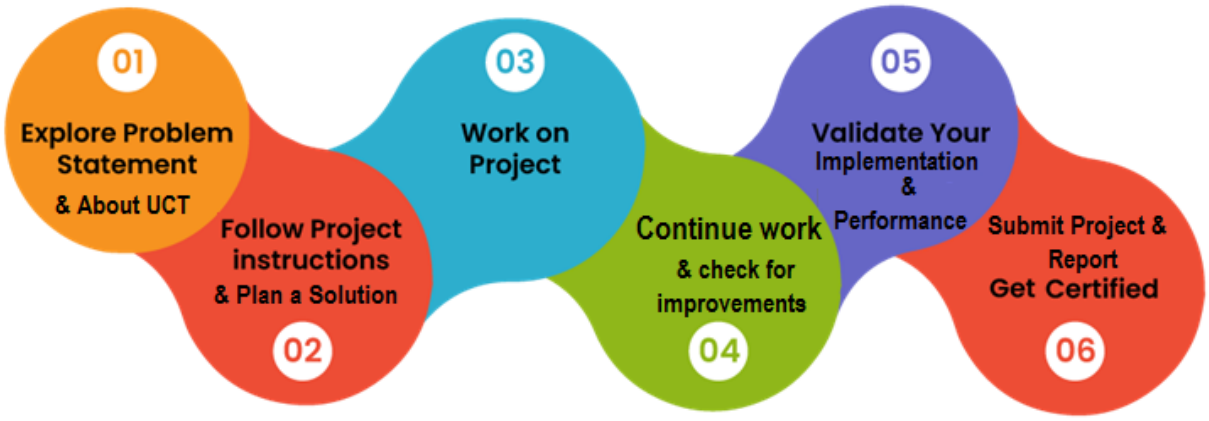
This project involved critical phases such as requirement analysis, design, development, testing, and

| |
|---|
| final deployment. |

## TABLE OF CONTENTS

## 1 Preface

Internships are pivotal in bridging the gap between academic learning and professional practice. They provide students and young professionals with a platform to gain hands-on experience, develop practical skills, and prepare for real-world challenges in their chosen career paths. A relevant internship is not just a stepping stone but a transformative experience that accelerates career growth. By gaining industry exposure, honing skills, and building networks, individuals are better equipped to achieve their professional aspirations.



# 2 Introduction

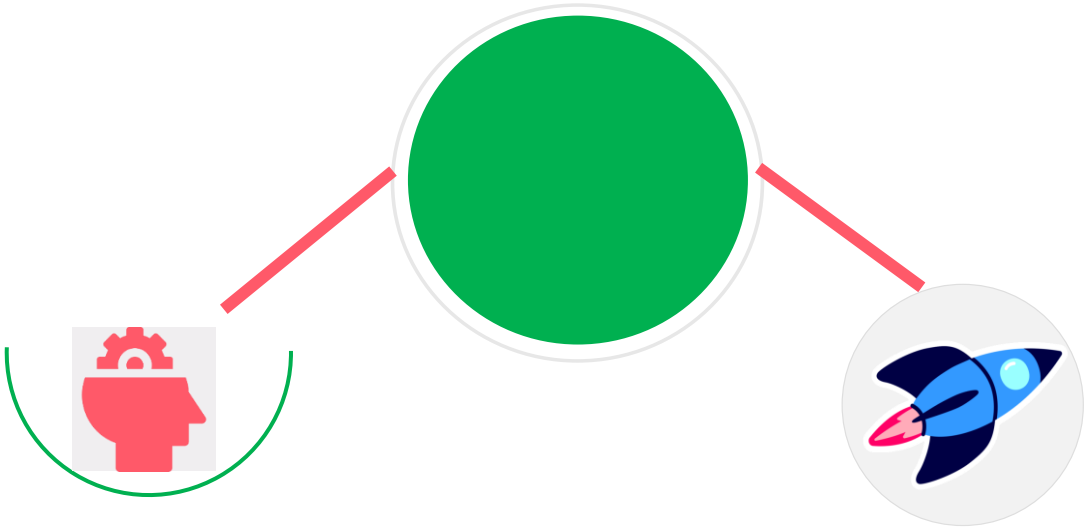
## 2.1 About UniConverge Technologies Pvt Ltd

The UniConverge Technologies Pvt Ltd, founded in 2013, operates in the Digital Transformation sector, offering industrial solutions with a primary emphasis on sustainability and return on investment (RoI). To develop its products and solutions, the company utilizes a range of cutting-edge technologies, including the Internet of Things (IoT), Cyber Security, Cloud Computing (AWS, Azure), Machine Learning, Communication Technologies (4G/5G/LoRaWAN), as well as Java Full Stack, Python, and Front-End development.

## 2.2 About upskill Campus (USC)

upskill Campus along with The IoT Academy and in association with Uniconverge technologies has facilitated the smooth execution of the complete internship process.

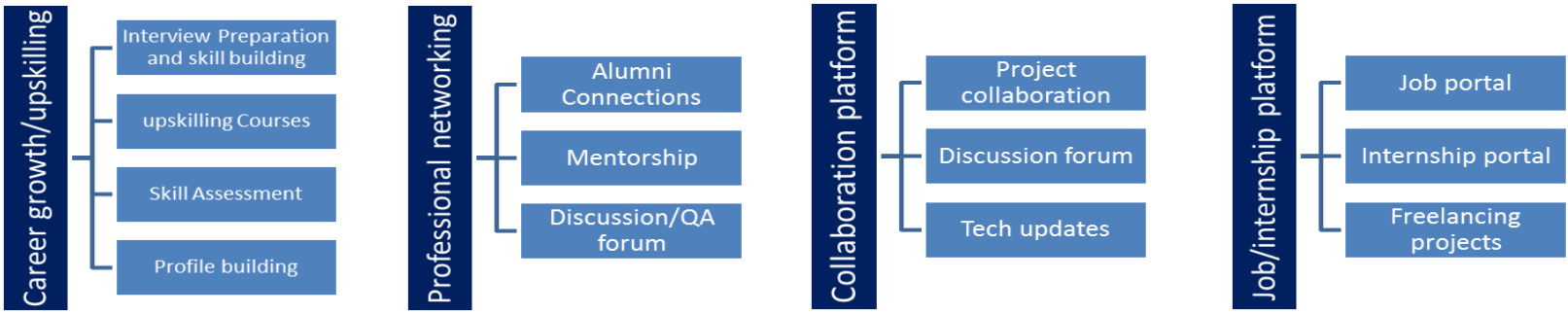
USC is a career development platform that delivers **personalized executive coaching** in a more affordable, scalable and measurable way.



Seeing need of upskilling in self paced manner along-with additional support services e.g. Internship, projects, interaction with Industry experts, Career growth Services

upSkill Campus aiming to upskill 1 million learners in next 5 year

https://www.upskillcampus.com/




### 2.3 The IoT Academy

The IoT academy is EdTech Division of UCT that is running long executive certification programs in collaboration with EICT Academy, IITK, IITR and IITG in multiple domains.

### 2.4 Objectives of this Internship Program

The objective for this internship program was to

☛ To get practical experience of working in the industry.

☛ to solve real-world problems.

☛ to have improved job prospects.

☛ to have an Improved understanding of our field and its applications.

☛ to have Personal growth like better communication and problem-solving.

☛ To Build Confidence

☛ To Enhance Employability

☛ To explore Career and Networking Opportunities

- To understand Industry-Specific Requirement
- To develop Skill

# 3 Brief About the Project

My "File Organizer in Python" project addresses the common challenge of managing disorganized files within a directory. Maintaining an organized structure becomes a critical necessity in today's digital era, where individuals and organizations frequently handle large volumes of files. The lack of organization can lead to inefficiencies, time loss, and confusion when retrieving or managing files.

The **File Organizer** was developed as a Python-based solution to automate the categorization of files into specific folders based on their extensions or file types. It aims to simplify directory management and improve productivity by ensuring files are neatly organized and easy to navigate.

## 3.1 Problem Statement

The disorganization of files in personal and professional environments often leads to:

i. **Difficulty in File Retrieval**: Scattered files make it challenging to locate specific documents, images, videos, or other file types.

ii. **Decreased Productivity**: Time spent on searching or managing files manually reduces overall efficiency.

iii. **Lack of Scalability**: With growing volumes of files, manual organization becomes increasingly impractical.

## 3.2 Proposed Solution

The **File Organizer** offers a robust and scalable solution by:

- Automatically sorting files into categorized folders (e.g., "Images," "Documents," "Videos") based on their extensions.
- Providing customization options to define folder categories and file types.
- Supporting a wide range of file formats, ensuring adaptability across diverse use cases.
- Handling large directories with ease, ensuring scalability for personal and professional environments.

By implementing this project, I aimed to deliver a practical tool that can be used by individuals and organizations to save time and maintain a structured file system.

# 4 Proposed Model/ Design

The design flow of the **File Organizer** project was structured to address the problem of disorganized files through a systematic and efficient solution. The development process was divided into logical stages, ensuring clarity and proper implementation. Here's an overview of the design flow:

---

- **1. Start: Requirement Analysis**
- **Understanding the Problem**:
    - The need to automate the organization of files in directories was identified.
    - Common challenges such as cluttered folders, difficulty in locating files, and inefficiency were outlined.
- **Defining Objectives**:
    - Automate file categorization based on extensions.
    - Provide flexibility for customization of categories.
    - Ensure scalability to handle large volumes of files.

---

- **2. Intermediate Stages**

## 4.1 Design of File Type Mapping

- A dictionary (FILE_TYPE_MAPPING) was created to define file categories and their corresponding extensions:

FILE_TYPE_MAPPING = {

"Images": [".jpg", ".jpeg", ".png", ".gif"],

"Documents": [".pdf", ".docx", ".txt"],

"Videos": [".mp4", ".mkv", ".avi"],

"Audio": [".mp3", ".wav"],

"Archives": [".zip", ".rar", ".7z"],

```
}
```

- This allowed dynamic allocation of files to appropriate folders.

## 4.2 Algorithm Design

- **Input**: Directory path (default: current working directory).
- **Process**:
    1. Scan the directory for files.
    2. Identify the file extension.
    3. Match the extension to a category in FILE_TYPE_MAPPING.
    4. Create a corresponding folder (if not already existing).
    5. Move the file to the respective folder.
- **Output**: An organized directory with files sorted into categorized folders.

## 4.3 Implementation

- Python libraries such as os and shutil were used for file handling:
    - os: To navigate the directory and check file extensions.
    - shutil: To move files between directories.
- The script was modularized into functions for clarity and reusability:
    - scan_directory(): Scans the directory and retrieves files.
    - categorize_files(): Matches files with categories.
    - move_files(): Moves files to the appropriate folders.

## 4.4 Testing and Debugging

- **Test Cases**:
    - Test with directories containing diverse file types.
    - Test with empty directories.

- Test edge cases, such as files without extensions or unsupported extensions.

- **Debugging**:
  - Ensured proper folder creation.
  - Handled exceptions for files with no read/write permissions.
  - Validated that existing files in destination folders were not overwritten.

---

## 4.5 Final Outcome

- **End Result**:
  - An organized directory with files categorized into folders such as "Images," "Documents," "Videos," etc.
  - Customizable configurations to adapt to different user needs.
- **User Experience**:
  - Users provided a seamless experience with clear outputs and logs indicating successful organization.
  - The system handled errors gracefully, ensuring robustness and reliability.

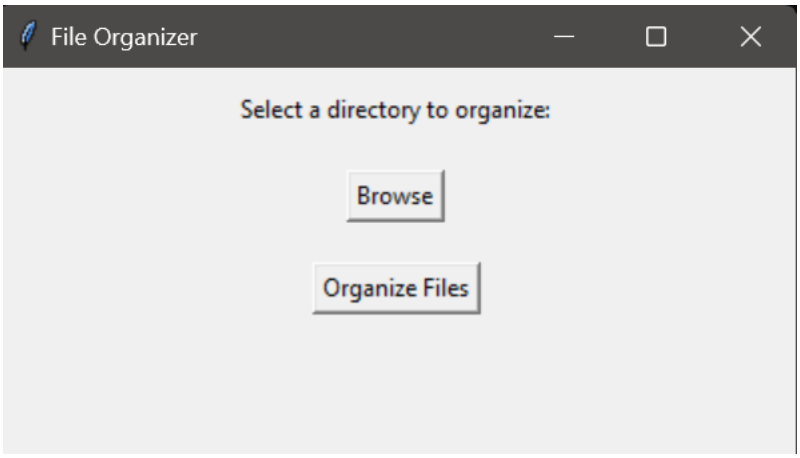
---

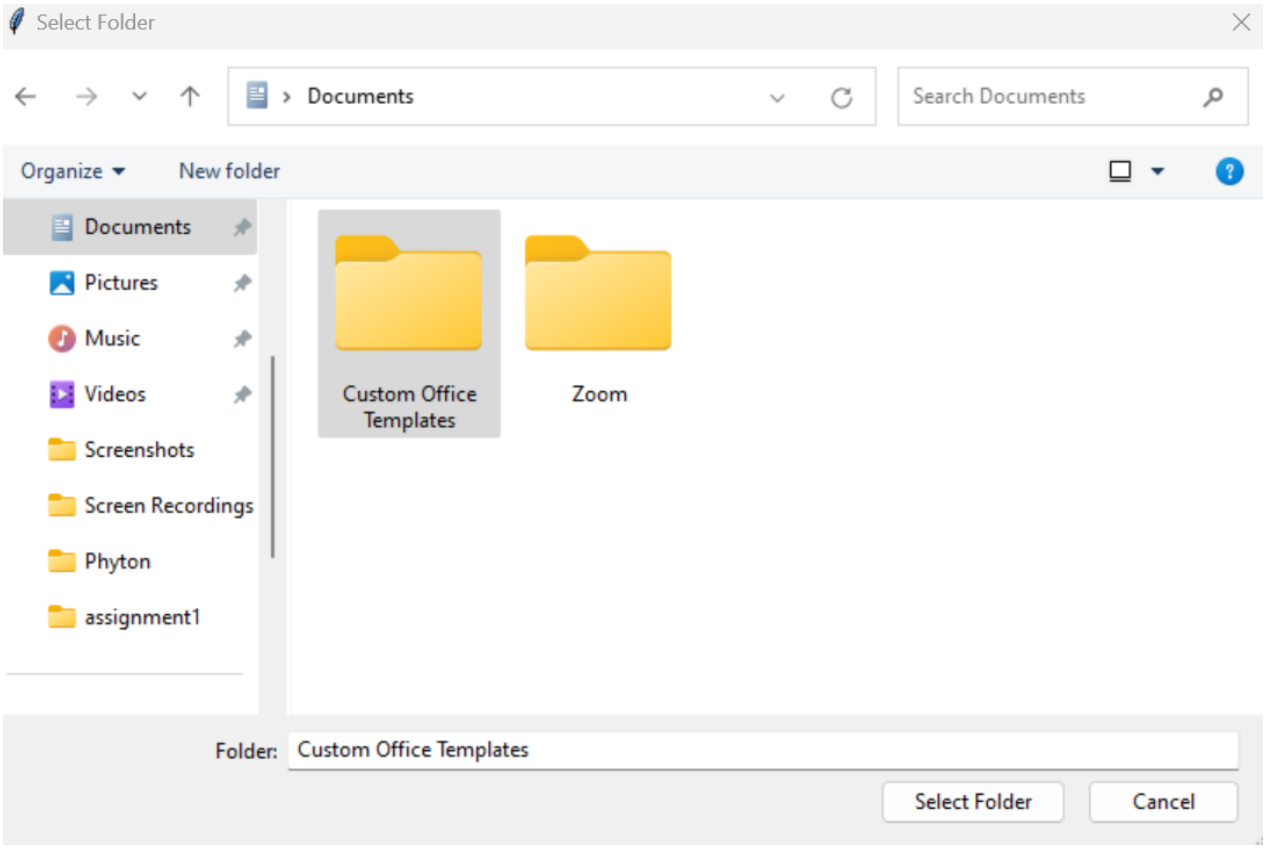
---

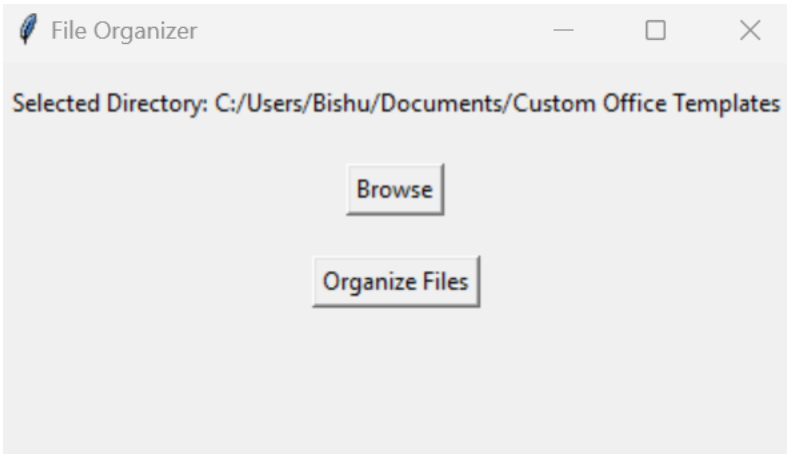
## 5 High-Level Diagram

1. **Start**: User specifies the directory to organize.
2. **Scan**: Directory is scanned for files.
3. **Categorize**: Files are matched to their respective categories based on extensions.
4. **Organize**: Files are moved to categorized folders.
5. **End**: Organized structure is displayed.

This structured design flow ensured that the **File Organizer** was both efficient and user-friendly, making it a practical tool for solving the problem of file disorganization.




Select Folder
Documents
Search Documents
Organize
New folder
Documents
Pictures
Music
Videos
Screenshots
Screen Recordings
Phyton
assignment1
Custom Office Templates
Zoom
Folder:
Custom Office Templates
Select Folder
Cancel


File Organizer
Selected Directory: C:/Users/Bishu/Documents/Custom Office Templates
Browse
Organize Files

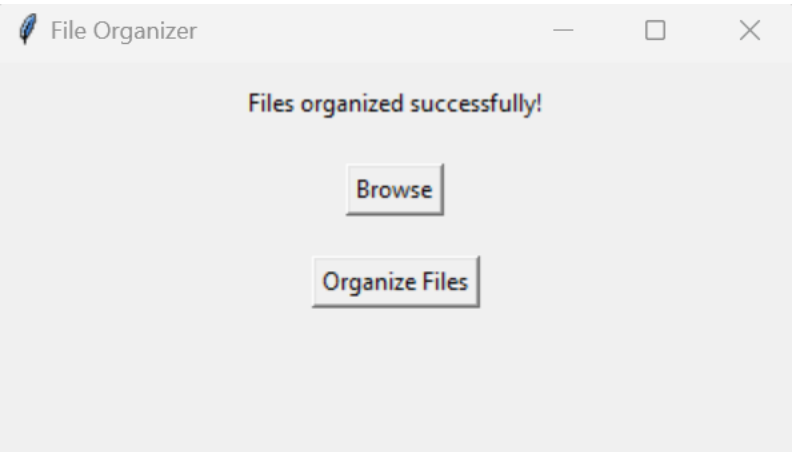


Figure 1: HIGH-LEVEL DIAGRAM OF THE SYSTEM

# 6 My Learnings and Overall Experience

During my internship at **Upskill Campus** and **The IoT Academy** in collaboration with **UniConverge Technologies Pvt. Ltd. (UCT)**, I gained invaluable insights into the practical applications of programming and problem-solving in an industrial environment. Working on the project **"File Organizer in Python"** enhanced both my technical and professional skills.

- **Key Learnings**

1. **Technical Skills**:
    - Mastery of Python programming and file handling techniques.
    - Exposure to project design, development, and testing methodologies.
    - Understanding of automation principles to address real-world problems.
2. **Problem-Solving Skills**:
    - Translating abstract problem statements into actionable solutions.
    - Designing scalable and efficient systems for large datasets.
3. **Soft Skills**:
    - Improved time management by working within tight deadlines.
    - Effective communication and collaboration in a team setting.
4. **Professional Insights**:
    - Understanding the importance of user-centric design in project development.
    - Exposure to industry expectations and best practices in software development.

Overall, this internship has been a transformative journey, providing a blend of learning and hands-on experience that will significantly contribute to my career growth.

---

## 7 Acknowledgments

I extend my heartfelt gratitude to all who supported me during this internship:

- **Upskill Campus and The IoT Academy**: For organizing and facilitating this enriching internship program.
- **UniConverge Technologies Pvt. Ltd. (UCT)**: To provide the real-world problem statement and guidance throughout the project.

## 8 Future work scope

The File Organizer project presents significant opportunities for future enhancements aimed at improving both functionality and usability. Potential improvements include advanced file categorization through AI-driven sorting and the development of a user-friendly Graphical User Interface (GUI) featuring drag-and-drop capabilities. Integration with cloud services such as Google Drive and Dropbox would further streamline the organization of local and cloud-based files. Moreover, implementing features for duplicate file management and offering customization options for categorization based on creation date or file size could greatly enhance the user experience. Incorporating multi-language support, ensuring cross-platform compatibility, and developing a mobile application would broaden its appeal. The addition of reporting and analytics features, along with scheduling and real-time monitoring options, would automate various processes, transforming the File Organizer into a comprehensive tool suitable for both personal and professional use.

**9 Code submission (Github link): https://github.com/bckapokharel/Project_File-Organizer.git**

**10 Report submission (Github link) : https://github.com/bckapokharel/Project_File-Organizer.git**