**Industrial Internship Report on**

**Project Title: Quiz Game  
Prepared by: Lucky Gautam**

|  |
| --- |
| *Executive Summary* |
| This report provides details of the Industrial Internship provided by upskill Campus and The IoT Academy in collaboration with Industrial Partner UniConverge Technologies Pvt Ltd (UCT).  This internship was focused on a project/problem statement provided by UCT. We had to finish the project including the report in 6 weeks’ time.  My project was a Python-based quiz game. The objective was to build a functional CLI-based quiz system that interacts with the user, evaluates answers, and stores scores.  This internship gave me a very good opportunity to get exposure to Industrial problems and design/implement solution for that. It was an overall great experience to have this internship. |

**TABLE OF CONTENTS**

1. [Preface 3](#_Toc139702806)
2. [Introduction 4](#_Toc139702807)

[2.1 About UniConverge Technologies Pvt Ltd 4](#_Toc139702808)

[2.2 About upskill Campus 8](#_Toc139702809)

[2.3 Objective 9](#_Toc139702810)

2.4 [Reference](#_Reference)..................................................................................................................................10

[2.5 Glossary 10](#_Toc139702812)

1. [Problem Statement 11](#_Toc139702813)
2. [Existing and Proposed solution 12](#_Toc139702814)

1. [Proposed Design/ Model 14](#_Proposed_Design/_Model)
2. [Performance Test](#_Performance_Test)............................................................................................................................15

6.1 [Test Plan/ Test Cases](#_Test_Plan/_Test).................................................................................................................15

6.2 [Test Procedure](#_Test_Procedure)..........................................................................................................................16

[6.3 Performance Outcome 16](#_Toc139702822)

1. [My learnings](#_My_learnings)....................................................................................................................................17
2. [Future work scope](#_Future_work_scope)...........................................................................................................................18

# Preface

This internship helped me practically apply my Python skills. It gave me a structured learning path to design, implement, test, and finalize a small-scale but complete project. The quiz game project gave exposure to user interaction design, input validation, file handling, and modular coding.

Over the course of 6 weeks, I learned how to break down a real-world problem, plan the solution, and develop it step by step using Python. The project helped strengthen my foundation in logic building, clean coding practices, and testing real user inputs. I now feel more confident in handling small software projects independently.

I am thankful to my mentors and coordinators at upskill Campus and UCT who guided us with weekly tasks and provided clear direction. It's not just about submitting a report, but about building something useful and real.



# Introduction

## About UniConverge Technologies Pvt Ltd

A company established in 2013 and working in Digital Transformation domain and providing Industrial solutions with prime focus on sustainability and RoI.

For developing its products and solutions it is leveraging various**Cutting Edge Technologies e.g. Internet of Things (IoT), Cyber Security, Cloud computing (AWS, Azure), Machine Learning, Communication Technologies (4G/5G/LoRaWAN), Java Full Stack, Python, Front end**etc.



1. UCT IoT Platform **(****)**

**UCT Insight** is an IOT platform designed for quick deployment of IOT applications on the same time providing valuable “insight” for your process/business. It has been built in Java for backend and ReactJS for Front end. It has support for MySQL and various NoSql Databases.

* It enables device connectivity via industry standard IoT protocols - MQTT, CoAP, HTTP, Modbus TCP, OPC UA
* It supports both cloud and on-premises deployments.

It has features to  
• Build Your own dashboard  
• Analytics and Reporting  
• Alert and Notification  
• Integration with third party application(Power BI, SAP, ERP)  
• Rule Engine

 

1. **Smart Factory Platform (****)**

Factory watch is a platform for smart factory needs.

It provides Users/ Factory

* with a scalable solution for their Production and asset monitoring
* OEE and predictive maintenance solution scaling up to digital twin for your assets.
* to unleased the true potential of the data that their machines are generating and helps to identify the KPIs and also improve them.
* A modular architecture that allows users to choose the service that they what to start and then can scale to more complex solutions as per their demands.

Its unique SaaS model helps users to save time, cost and money.

 

1.  based Solution

UCT is one of the early adopters of LoRAWAN teschnology and providing solution in Agritech, Smart cities, Industrial Monitoring, Smart Street Light, Smart Water/ Gas/ Electricity metering solutions etc.

1. Predictive Maintenance

UCT is providing Industrial Machine health monitoring and Predictive maintenance solution leveraging Embedded system, Industrial IoT and Machine Learning Technologies by finding Remaining useful life time of various Machines used in production process.



## 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

<https://www.upskillcampus.com/>

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



## 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.

## Objectives of this Internship program

The objective for this internship program was to

 ☛ get practical experience of working in the industry.

 ☛ to solve real world problems.

 ☛ to have improved job prospects.

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

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

## Reference

[1] Python Official Docs (<https://docs.python.org/>)

[2] W3Schools Python Tutorial (<https://www.w3schools.com/python/>)

[3] YouTube – Tech with Tim: Python Projects

## Glossary

|  |  |
| --- | --- |
| Terms | Acronym |
| CLI | Command Line Interface |
| GUI | Graphical User Interface |
| IDE | Integrated Development Environment |
| GitHub | Code hosting and version control platform |
| Scoreboard File | Text file where scores are saved |

# Problem Statement

The assigned problem statement for the internship was to **develop a Python-based command-line Quiz Game** that allows users to attempt multiple-choice questions on various topics. The game must:

* Display questions with multiple answer options
* Accept user input and validate it
* Track and display the user’s score based on correct answers
* Store each user's name and score in a file (scoreboard.txt)
* Display a scoreboard showing all previous scores

The project is intended to strengthen programming logic, modular thinking, user interaction design, and file handling in Python.

# Existing and Proposed solution

**Existing Solutions:**

There are several online quiz platforms and mobile applications like **Kahoot**, **Quizizz**, and **Google Forms** that allow users to participate in quizzes. However, these platforms are either:

* Built for mass usage with server-side complexity,
* Dependent on the internet,
* Not open-source or beginner-friendly, or
* Too complex for basic learners to understand or build themselves.

Also, most of these platforms don’t provide access to their internal code logic, making it hard for students to learn from them.

**Proposed Solution:**

My proposed solution is a **simple, offline, CLI-based Python quiz game**. It is:

* Lightweight and beginner-friendly
* Fully offline and requires only Python to run
* Developed using basic Python concepts like lists, loops, functions, and file handling
* Easy to modify or extend (e.g., add more questions or create categories)

The program stores user scores in a local text file and displays a scoreboard after every quiz attempt.

**Value Addition:**

* Helps beginners understand how to build interactive programs in Python
* Demonstrates real-world usage of input(), try-except, functions, and file I/O
* Scoreboard functionality adds a competitive and replayable element
* Can be enhanced further with GUI (tkinter) or database integration

## Code submission

(<https://github.com/LuckyG05/Upskillcampus/blob/main/quizgame.py>)

## Report submission

## (<https://github.com/LuckyG05/Upskillcampus/blob/main/quizgame_LuckyGautam_USC_UCT.pdf.docx>)

# Proposed Design/ Model

The proposed solution follows a **simple linear and modular flow**, designed using basic Python functions. The design ensures that each part of the quiz game — from user interaction to score saving — is handled cleanly and logically.

**Design Flow:**

1. **Start**

* User is welcomed to the game
* Asks for name and confirmation to play

1. **Load Questions**

* Questions are stored in a list inside the code (or can be loaded from a file/JSON)
* Each question has 4 options and the correct answer index

1. **Quiz Loop**

* Each question is displayed one-by-one
* User is prompted to enter a valid option (1 to 4)
* Input is validated using try-except blocks

1. **Scoring System**

* For each correct answer, the score increases by 1
* Incorrect answers display the correct one
* Score is shown after every question

1. **Score Saving**

* Final score is saved in a file (scoreboard.txt) with the user's name
* If the file doesn’t exist, it is created automatically

1. **Scoreboard Display**

* After the quiz ends, all past scores are displayed from scoreboard.txt

1. **End**

* Game thanks the user and exits

# Performance Test

This project, while simple, simulates real-world software principles: user interaction, validation, data persistence, and modularity. The performance testing focused on ensuring that the game functions reliably across different inputs, handles errors gracefully, and saves data consistently — just as expected in real industrial scenarios.

| **Constraint** | **Addressed in Design?** | **How It Was Handled** |
| --- | --- | --- |
| **Input Accuracy** | Yes | Used try-except and input range checks |
| **Data Loss** | Yes | Scores are saved using append mode in a .txt file |
| **Crash Handling** | Yes | Handled invalid input (e.g., non-numeric answers) |
| **Ease of Use** | Yes | Simple CLI prompts and feedback for users |
| **Code Maintainability** | Yes | Used modular functions for each task |

## Test Plan/ Test Cases

| **Test Case #** | **Description** | **Expected Result** | **Status** |
| --- | --- | --- | --- |
| TC1 | User enters correct answer | Score increases by 1 | Pass |
| TC2 | User enters wrong answer | Correct answer is displayed | Pass |
| TC3 | User enters invalid input (e.g., 'abc') | App does not crash; handles gracefully | Pass |
| TC4 | User score is saved | scoreboard.txt is updated | Pass |
| TC5 | Show previous scores | Past scores are displayed correctly | Pass |

## Test Procedure

1. Launch the program using python quizgame.py

2. Enter various types of inputs:

* Correct values (1–4)
* Wrong values (e.g., 5, 0)
* Invalid (e.g., letters or symbols)

3. Complete the quiz

4. Verify:

* Real-time score is displayed
* Final score is saved in the scoreboard
* Past scores appear on screen after the game ends

## Performance Outcome

* The game performs smoothly under different input scenarios.
* There were no crashes observed, even with invalid input.
* Data is consistently saved and displayed.
* Response time for each operation (question load, input read, file write) is instantaneous due to the lightweight nature of the application.
* The program can be easily extended with GUI or category filters without breaking its base logic.

# My learnings

This internship gave me hands-on experience in applying my Python knowledge to a real-world problem. Through the 6-week project, I learned how to:

* Break down a problem into logical modules and plan the structure of a program
* Implement clean and reusable code using Python functions
* Use **file handling** effectively to store and retrieve data
* Validate user input using try-except blocks to prevent program crashes
* Write and test a complete application from scratch
* Manage time and milestones week-by-week, simulating an actual software development cycle
* Prepare documentation and maintain a GitHub repository professionally

These learnings will be extremely useful in my career. They not only strengthen my Python programming foundation but also improve my confidence in working on future projects — whether academic, industrial, or personal. The experience has also helped me understand the importance of planning, user-friendly design, and testing.

# Future work scope

Due to time limitations and the focus on keeping the project simple and beginner-friendly, several enhancements could not be implemented. However, the current project design allows for easy expansion in the future. Some ideas for future improvements include:

* **GUI Implementation**: Transitioning the quiz game from CLI to GUI using Python's tkinter for better visual interaction.
* **Question Bank Storage**: Using a .json or .csv file to store and load quiz questions dynamically instead of hardcoding them.
* **Quiz Categories and Levels**: Allowing users to select topics (e.g., Science, History) or difficulty levels (Easy, Medium, Hard).
* **User Authentication**: Adding login/signup to maintain personalized score history.
* **Web Version**: Developing a web-based version using Flask or Django for broader accessibility.

These features would increase the reusability, interactivity, and professionalism of the quiz game.

***Thank you to the team at upskill Campus and UniConverge Technologies Pvt Ltd for this opportunity.***