**Industrial Internship Report on**

**”** **Music Player Application using GUI”**

**Prepared by**

**Name :- Vikram Singh Kaviya**

|  |
| --- |
| *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.  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 9](#_Toc139702811)

[2.5 Glossary 10](#_Toc139702812)

[3 Problem Statement 11](#_Toc139702813)

[4 Existing and Proposed solution 12](#_Toc139702814)

[5 Proposed Design/ Model 13](#_Toc139702815)

[5.1 High Level Diagram (if applicable) 13](#_Toc139702816)

[5.2 Low Level Diagram (if applicable) 13](#_Toc139702817)

[5.3 Interfaces (if applicable) 13](#_Toc139702818)

[6 Performance Test 14](#_Toc139702819)

[6.1 Test Plan/ Test Cases 14](#_Toc139702820)

[6.2 Test Procedure 14](#_Toc139702821)

[6.3 Performance Outcome 14](#_Toc139702822)

[7 My learnings 15](#_Toc139702823)

[8 Future work scope 16](#_Toc139702824)

# Preface

Summary of the whole 6 weeks’ work.

About need of relevant Internship in career development.

Brief about Your project/problem statement.

Opportunity given by USC/UCT.

How Program was planned



Your Learnings and overall experience.

Thank to all (with names), who have helped you directly or indirectly.

Your message to your juniors and peers.

# 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. Github
2. Youtube

## Glossary

|  |  |
| --- | --- |
| Terms | Acronym |
| User | User enter’s the location of the music directory |

# Problem Statement

In the assigned problem statement :

The task with developing a music player application using Java Swing. The application should provide a user-friendly interface for playing, pausing, stopping, and managing a collection of audio tracks. The application should also support basic functionality such as playlist management and volume control.

# Existing and Proposed solution

Provide summary of existing solutions provided by others, what are their limitations?

What is your proposed solution?

What value addition are you planning?

## Code submission:- (https://github.com/Vikram-Singh-Kaviya/-Music-Player-Application-using-GUI)

## Report submission:-

# Proposed Design/ Model

Given more details about design flow of your solution. This is applicable for all domains Students can cover it after they have their algorithm implementation. There is always a start, intermediate stages and then final outcome.

## High Level Diagram (if applicable)

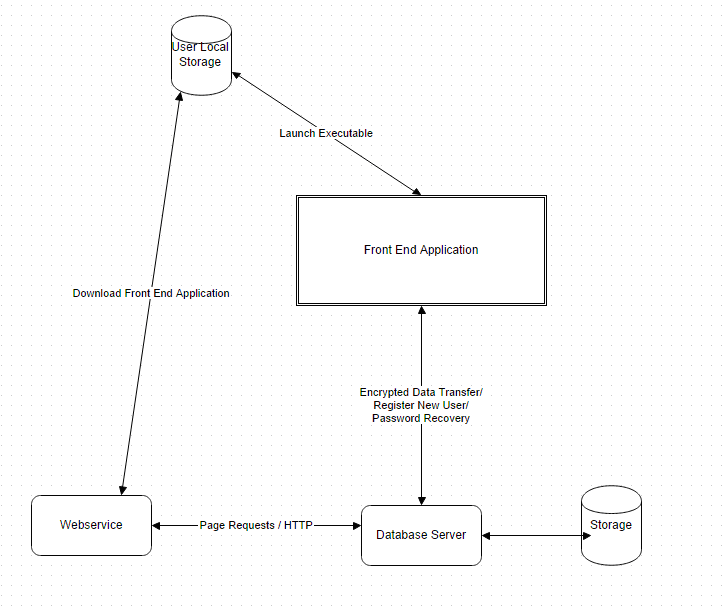
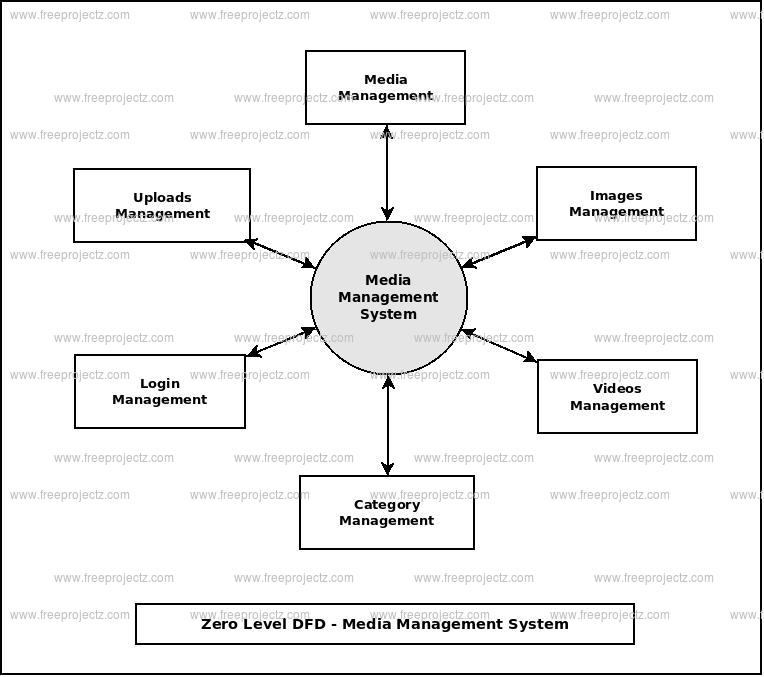


Figure 1: HIGH LEVEL DIAGRAM OF THE SYSTEM

## Low Level Diagram (if applicable)



## Interfaces (if applicable)

Update with Block Diagrams, Data flow, protocols, FLOW Charts, State Machines, Memory Buffer Management.

# Performance Test

This is very important part and defines why this work is meant of Real industries, instead of being just academic project.

Here we need to first find the constraints.

How those constraints were taken care in your design?

What were test results around those constraints?

Constraints can be e.g. memory, MIPS (speed, operations per second), accuracy, durability, power consumption etc.

In case you could not test them, but still you should mention how identified constraints can impact your design, and what are recommendations to handle them.

## Test Plan/ Test Cases

Test plan contains name , description , collection a chosen test case, builds, test result, milestones, tester assignments and priority definition.

**Test Execution:-**

1. Test execution is available when.
2. A Test Specification is written.
3. A Test Plan is created.
4. Test Case Suite (for the Test Plan) is defined.
5. A Build is created.

## Test Procedure:-

Software Testing is the process of executing a program or system with the intent of finding error. The scope of software testing offen includes examination of code as well execution of that code in various conditions.

1. Component or unit testing
2. System testing
3. Acceptance testing

## Performance Outcome

The performance outcome of a music player application can be evaluated through several key criteria. Firstly, the responsiveness of playback controls—such as play, pause, stop, and track navigation—needs to be swift and seamless, ensuring minimal delay between user actions and actual playback. Additionally, the audio quality during playback is paramount, demanding clarity and distortion-free output. The application's resource utilization, including memory and CPU usage, must be monitored to prevent excessive strain on the system. Efficient playlist management, involving quick additions and removals of tracks, is essential for a smooth user experience, particularly when handling larger playlists. The user interface's responsiveness and fluidity play a significant role, as transitions and animations should be smooth without perceptible lag. Effective error handling for issues like missing audio files or unsupported formats contributes to a reliable user experience. Concurrency and multithreading management must be sound to avoid thread-related problems. Startup time and performance with large playlists are also critical aspects to evaluate. Compatibility across different platforms and Java versions ensures consistent performance, while considerations for battery consumption on mobile platforms are important. Ultimately, the application's scalability, memory management, and optimization through code profiling contribute to achieving a positive user experience. Regular testing, optimization, and user feedback play pivotal roles in maintaining and enhancing the music player application's performance outcomes.

# My learning:-

It's great that you're looking to learn from your project! Developing a music player application using Java Swing provides an excellent opportunity to enhance your programming skills. Here's what you can potentially learn from this project:

1. \*\*Java Programming:\*\* You'll gain a deeper understanding of Java programming concepts, object-oriented principles, and how to work with various Java libraries and APIs for user interfaces, audio playback, and more.

2. \*\*Swing Framework:\*\* Building a user interface using Java Swing introduces you to GUI components, event handling, layout managers, and other GUI-related concepts. This experience will be useful for creating other desktop applications as well.

3. \*\*User Interface Design:\*\* Designing an intuitive and visually appealing user interface is crucial for user satisfaction. You'll learn about creating an engaging UI layout, using icons, buttons, sliders, and other components effectively.

4. \*\*Event Handling:\*\* Implementing playback controls and playlist management involves event-driven programming. You'll learn how to handle user interactions, respond to button clicks, and manage the application's behavior.

5. \*\*Audio Playback:\*\* Integrating an audio playback engine exposes you to working with audio files, decoding formats, managing playback states, and ensuring a seamless audio experience.

6. \*\*File Management:\*\* Handling playlists involves managing files, file paths, and metadata. You'll learn about file I/O operations and data structures for managing track information.

7. \*\*Error Handling:\*\* Dealing with potential errors, such as missing files or unsupported formats, helps you understand the importance of robust error handling and providing meaningful user feedback.

8. \*\*Optimization:\*\* As you work with a larger codebase, you'll learn about code organization, optimizing performance, identifying bottlenecks, and profiling tools for analysis.

9. \*\*User Experience:\*\* Creating a smooth and responsive application enhances the user experience. You'll learn about user-centric design principles and how to prioritize usability.

10. \*\*Problem-Solving:\*\* Throughout the development process, you'll encounter challenges and obstacles that require creative problem-solving skills. This experience hones your ability to find solutions and troubleshoot issues.

11. \*\*Project Management:\*\* Completing a project involves planning, organizing tasks, setting milestones, and managing your time effectively. You'll learn project management skills that can be applied to future endeavors.

12. \*\*Continuous Learning:\*\* Technology evolves, and there's always something new to learn. As you explore libraries, tools, and techniques, you'll develop a mindset of continuous learning.

Remember that learning often comes from both successes and challenges. Embrace the journey, seek help when needed, and enjoy the satisfaction of building a functional application from scratch. Your learning from this project will serve as a solid foundation for future programming endeavors.

# 

# 8.Future work scope

The completion of your Java Swing-based music player application lays the groundwork for an array of exciting future possibilities. One avenue of exploration could involve enhancing the user interface with advanced design elements, like customizable themes and animated transitions, for a more visually appealing and immersive experience. You might consider incorporating an audio equalizer to empower users with control over sound frequencies, elevating the quality of their audio playback. In the realm of playlist management, features like drag-and-drop track reordering, auto-generation of playlists based on genres, and support for multiple saved playlists could elevate user convenience. The integration of a search bar and filters might assist users in swiftly locating and playing specific tracks. Integrating online streaming services or APIs could extend the application's capabilities, enabling users to enjoy a diverse range of music sources. Exploring the possibility of a dynamic audio visualizer could add a captivating visual dimension to the music playback experience. Creating user profiles, cross-platform compatibility, mobile versions, and cloud integration are avenues for extending accessibility and convenience. The incorporation of social sharing, audio effects, gesture control, machine learning-based recommendations, and accessibility features would contribute to the application's sophistication and inclusiveness. Furthermore, a commitment to regular bug fixes, updates, and user-driven improvements ensures that the application remains polished, relevant, and responsive to user needs. As you venture into these exciting future prospects, your music player application could evolve into a feature-rich, engaging, and indispensable tool for music enthusiasts.