**Project Task 1**

Andre Burger - 577387

Devon van Heerden - 577334

Werner Janse van Rensburg - 577930

Marizaan Joubert - 577344

Wian Joubert - 577737

**ApexCare SolutionsProject Management Plan**

**Table of Contents**

[Introduction 2](#_Toc178590332)

[Gantt Chart 2](#_Toc178590333)

[Risks 2](#_Toc178590334)

[Work Breakdown Structure 3](#_Toc178590335)

[Conclusion 3](#_Toc178590336)

# Introduction

# Gantt Chart

# Risks

|  |  |  |  |
| --- | --- | --- | --- |
| **Risk Description** | **Likelihood that risk occurs** | **Impact if risk occurs** | **Mitigation Strategy** |
| System Downtime | LOW | HIGH | Implement redundant servers, backup systems, and real time monitoring systems to detect, prevent and resolve the issues resulting in downtime. |
| Poor incident categorization | MEDIUM | MEDIUM | Regular training of the AI model will reduce the likelihood of incorrect categorization and improve its accuracy when sorting queries. |
| Inaccurate data integration | MEDIUM | HIGH | Implement data validation and error handling |
| Privacy and Security Breaches | LOW | HIGH | Make use of strong security protocols (encryption, authentication, vulnerability assessment, regular security updates) |
| Scalability and performance risks as user amount increases | LOW | HIGH | Design the system with scalability in mind by using a cloud infrastructure |
| Mobile App compatibility | MEDIUM | HIGH | Various mobile platforms and devices will be tested to ensure compatibility. |
| Bugs after deployment | HIGH | MEDIUM | After the system goes live, bug-tracking software will monitor and fix issues. |
| AI bias | LOW | MEDIUM | Developers will regularly review the AI models manually to ensure that there are no biases in the data scheduling |
| Faulty prioritization of service requests | MEDIUM | MEDIUM | Update priority settings regularly.  Manually review service requests that are critical to the system. |
| Issues when depending on a third-party service | HIGH | HIGH | Fallback plans like backup API’s, and alternative service providers will be utilized in case of a third-party service crashing. |
| Data loss when integrating updates or migrations | LOW | HIGH | Backing up all data before an update/migration. Do a smaller scale test before involving the entire system. Have rollback procedures in place. |
| Inadequate user training | MEDIUM | MEDIUM | Have user guides and tutorials available to users so that they can utilize the system to its full ability. Having a simple and straightforward UI will also aid in the ease of use of the system |
| Incorrect Resource Allocation | MEDIUM | HIGH | Utilize tools to constantly monitor and update resources. This will ensure that the entire team stays aware of how everything is progressing. |
| Increasing operational costs | MEDIUM | HIGH | Regularly monitor and review costs to optimize resource allocation and accurately predict future expenses, ensuring effective budget management. |
| Inaccurate prediction of demands | LOW | MEDIUM | Make use of predictive models to analyse historical data to generate accurate forecasting demands. |
| Poor Disaster Recovery Strategy | LOW | HIGH | Develop and test disaster recovery plans to minimize downtimes when a system failure occurs. |

# Work Breakdown Structure

# Conclusion