**Course Project Analysis – Workers' Compensation Digital Case Management System  
IS315, Fall Semester**

**Amani Mulaudzi, Angela Mast, Marina Perdiguero and Pawinee Sukasem**

**December 1, 2024**

**Professor Lynn Kuluva**

**Park University**

# Table of Contents

[I. Executive Summary 3](#_heading=h.d95296lip8wq)

[II. Introduction 4](#_heading=)

[III. Planning the Information System Project 5](#_heading=h.lkt1xc2vwtvk)

[IV. Feasibility Study 7](#_heading=)

[V. Project Plan 16](#_heading=h.3mygp82j3d9c)

[VI. JAD Workshop 31](#_heading=h.o82nn6foi2qa)

[VII. Discussion of the Current System 33](#_heading=h.980z7yfe61rw)

[VIII. Systems Analysis 35](#_heading=h.cmredyj4e24e)

[IX. Use Cases 38](#_heading=h.nlallppswcl)

[X. Process Model 53](#_heading=h.baictivqjlzp)

[XI. Data Model 81](#_heading=h.qri469r70n54)

[XII. Conclusion 99](#_heading=h.mm0vfcl9h82c)

[References 100](#_heading=h.mtppmhcb5o15)

[Appendix A - Gantt 101](#_heading=h.vh5mv015svsj)

[Appendix B - PERT Chart 102](#_heading=h.toihbgm1tjss)

# I. Executive Summary

# The South Dakota Department of Labor’s Workers’ Compensation Division is struggling with a physical file system. Searching for claims in space constraining file cabinets is a long and cumbersome task for the clerks. Addition of new claims causes storage issues, and it is overall a long process that can affect customer and clerk satisfaction. The addition of a digital system to manage claims would provide an exponentially faster experience for the department of labor and the claimants. There will be profit due to the lesser need of clerks because of the efficient system. This could save the department a substantial amount of money and time.

# II. Introduction

The South Dakota Department of Labor, Workers’ Compensation division is currently using a paper based system to process injury claims from workers across the state. Although the current system is functional, it is far from efficient. Massive file cabinets are used to store different claims, medical reports, and other necessary documents based on the year and case number. Some folders could be filled with documents, and others have only one or two. The proposition to transition to an online database system could be a tremendous improvement for the Workers’ Compensation division.

# III. Planning the Information System Project

## Information System Request

The South Dakota Department of Labor's Workers' Compensation division is requesting a new information system in order to improve efficiency.

## Project Sponsor

Kelly Jones, Head of HR of the South Dakota Department of Labor, Worker’s Compensation division

## Business Need

The current paper-based system for handling and finding workers' compensation claims is slow and inefficient. This causes delays in answering questions and makes the clerks' work more difficult.

## Business Requirements

Digital Claims Submission: Implement a secure online platform for submitting and managing injury claims and related documents.

Electronic Document Management: Convert existing paper files to digital format and store them in a searchable database.

Advanced Search Functionality: Enable quick retrieval of claims using various search parameters.

Security and Compliance: Ensure the system adheres to legal and privacy regulations.

User-Friendly Interface: Develop an accessible interface for both clerks and external users.

System Integration: Ensure compatibility with existing state IT systems.

## Business Value

Increased productivity, lower expenses, better security and accuracy, better data management, higher customer satisfaction, and scalability for future requirements are all examples of the business value. This solution improves overall operations and stakeholder trust by streamlining procedures, reducing errors, and offering workers and businesses faster, more dependable service. The department now uses a paper-based procedure that is time-consuming, prone to errors, and expensive. The digital system will provide better data management, increase customer satisfaction, lower expenses, improve data accuracy and security, and improve efficiency. Furthermore, it will be flexible and expandable to meet changing requirements, resulting in more dependable and efficient operations.

## Special Issues or Constraints

Since dealing with medical records, the database/system should have high security measures. This will be a full transition from paper to electronic with no existing electronic database.

## Project Planning Summary

A team of a project manager, systems analyst, database administrator, software developers, UI/UX designer, IT security specialist, QA testers, trainers, and support personnel will be needed to complete the System Proposal. The approximate cost is between $195,000 and $300,000. Project start-up, requirement collecting, system design, development, testing, training, deployment, and final review are important responsibilities. These procedures guarantee careful design, development, and deployment of the new information system.

# IV. Feasibility Study

## Technical Feasibility

***Can We Build It?***

Current Status: The project team has limited experience with the business functions currently supported by the department's paper-based system, as there is minimal digital integration. Transitioning to a fully digital solution will require the development of new procedures and processes to align with the system's capabilities and business needs. (Dennis et al., 2022).

#### Familiarity with Application:

* **Current Status:** There is very little digital integration in the department's paper-based existing system. Using a fully digital solution necessitates the introduction of new procedures and processes (Watson, 2013).
* **Risk:** There is less knowledge with comparable software inside the department due to the large change from paper to digital, which raises the risk. This risk can be reduced, though, if the development team has prior expertise with applications based on the government or insurance industry.

#### Familiarity with Technology:

* **Current Status:** Digital document management, databases, and web-based apps are all part of the suggested solution. The learning curve can be high if the department has never used similar technologies before (Dennis et al., 2022).
* **Risk:** If the internal team lacks background in database administration, cybersecurity, or web development, there is a greater chance of failure. This risk can be decreased by working with seasoned vendors or employing knowledgeable IT specialists.

#### Project Size:

* **Current Status:** The project entails creating a safe internet platform, integrating a database system, and digitizing a large number of records. This project has many different parts and is quite large-scale (Watson, 2013).
* **Risk:** Because of their complexity, larger projects inherently include higher risk. Effective risk management will need careful project management, which includes phased implementation and regular milestones.

#### Compatibility:

* **Current Status:** There isn't much modern technology that can be merged with the manual, paper-based method because of this. The recommended system must be secure, accessible to authorized users, and compatible with other state systems.
* **Risk:** There is a moderate level of risk because the new system may have to interface with other governmental or medical systems. Ensuring interoperability with existing protocols and standards will be crucial. Two such roadblocks are data migration and ensuring that the new system conforms with all relevant legal and regulatory requirements (Dennis et al., 2022).

### *Conclusion:*

### Although the suggested system is theoretically possible, there are certain risks associated with it because of the magnitude of the project and lack of experience with the technology and application. Through meticulous planning, collaborating with knowledgeable experts, and guaranteeing compatibility with current systems, these risks can be reduced. Overall, with the right knowledge and resources, the system can be developed, however it might require upgrading or purchasing additional technical resources to satisfy the new specifications (Watson, 2013).

## Economic Feasibility

***Development costs***

There will be some costs during the development of this system. Firstly, there is labor cost. We will have to pay programmers, system analysts, project managers, and so forth during the duration of development. Data entry will also be costly, as the system needs to have information transferred from physical paper to a database. Hardware will be a cost, since little to none exists for this type of project because it is currently being run on a physical file system. Software will also be an expense, and whether it is outsourced or created in house will determine the price. ***Annual operating costs***

To keep the system running, there should be an assigned team for keeping software up to date and working as expected. This team (which could be small) will be an expense due to labor costs, but will provide maintenance for the system. There might be an annual cost to use licensed software if the system uses outsourced software. Lastly, training staff on using the system/training on certain aspects of the development technical skills will be a cost (Singal et al., 2023). This will be annual because of new hires and system updates.

***Annual benefits***

Some tangible benefits include faster service, which provides for more clients and claims to be processed in the same amount of time. Since more claims can be done in the same amount of time, there would not need to be as many employees to provide the same rate of service. The transition to an online format will result in reduction of staff due to this increased rate. Another tangible benefit would be that this online database can hold more clients. Since physical files have a finite limitation for how much data can be stored, having an online database can allow for more business easily. Another annual benefit would be the reduction of paper and paper storage costs due to the transition to an online format.

***Intangible benefits and costs***

An intangible benefit would be the large amount of storage available after going paperless. Another benefit would be accessibility - to find a client’s records, there is not a requirement anymore to be in the building. Improved customer service and enhanced reputation would be intangible benefits. This would be due to the faster and more reliable service. There would also be better security due to the online format rather than the physical one. Files would not go missing without a trace with an online system.

|  | Devel | Year 1 | Year 2 | Year 3 | Year 4 | Year 5 | Total |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Tangible Benefits: |  |  |  |  |  |  |  |
| Reduction of Staff |  | $250,000 | $250,000 | $250,000 | $250,000 | $250,000 | $1,250,000 |
| Reduced paper storage costs |  | $5,000 | $5,000 | $5,000 | $5,000 | $5,000 | $25,000 |
| Ability to hold more Clients |  | $15,000 | $15,000 | $15,000 | $15,000 | $15,000 | $75,000 |
| Improved accuracy (fewer errors) |  | $10,000 | $10,000 | $10,000 | $10,000 | $10,000 | $50,000 |
| Total Tangible Benefits |  | $280,000 | $280,000 | $280,000 | $280,000 | $280,000 | $1,400,000 |
|  |  |  |  |  |  |  |  |
| Development Costs: |  |  |  |  |  |  |  |
| Labour | $420,000 |  |  |  |  |  | $420,000 |
| Hardware Purchase | $150,000 |  |  |  |  |  | $150,000 |
| Data Entry | $20,000 |  |  |  |  |  | $20,000 |
| Software | $80,000 |  |  |  |  |  | $80,000 |
| Total Development Cost | $670,000 |  |  |  |  |  | $670,000 |
|  |  |  |  |  |  |  |  |
| Operational Costs |  |  |  |  |  |  |  |
| Software Maintenance |  | $5,000 | $5,000 | $5,000 | $5,000 | $5,000 | $25,000 |
| Software Licensing |  | $1,000 | $1,000 | $1,000 | $1,000 | $1,000 | $5,000 |
| Training and Development |  | $20,000 | $20,000 | $20,000 | $20,000 | $20,000 | $100,000 |
| Total Operational Costs |  | $26,000 | $26,000 | $26,000 | $26,000 | $26,000 | $130,000 |
|  |  |  |  |  |  |  |  |
| Total Costs | $670,000 | $26,000 | $26,000 | $26,000 | $26,000 | $26,000 | $800,000 |
| Total Benefits - Total Costs | -$670,000 | $254,000 | $254,000 | $254,000 | $254,000 | $254,000 | $600,000 |
| Cumulative Net Cash Flow | -$670,000 | -$416,000 | -$162,000 | $92,000 | $346,000 | $600,000 |  |
|  |  |  |  |  |  |  |  |
| Return on Investment (ROI) |  |  |  |  |  |  | 75% |
|  |  |  |  |  |  |  |  |
| Payback Period | 3.64 years |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| Intangible Benefits: |  |  |  |  |  |  |  |
| Improved customer satisfaction due to faster service |  |  |  |  |  |  |  |
| Better data security and accessibility |  |  |  |  |  |  |  |
| Reduced risk of lost or misplaced files |  |  |  |  |  |  |  |
| Enhanced reputation of the agency for modern practices |  |  |  |  |  |  |  |

## Organizational Feasibility

***Is the project strategically aligned with the business?***

Yes, the project is strategically aligned with the goals of the South Dakota Department of Labor, specifically within the Workers’ Compensation Division. The primary objective of this department is to ensure fair and timely handling of workers' compensation claims, which is crucial for supporting injured workers in the state. Moving from a paper-based system to a digital one will greatly enhance the efficiency of the division. This transition will not only reduce the time spent managing and retrieving files but will also improve overall service quality by enabling faster responses to inquiries.

***Project champions:***

* **Strategic Alignment:** The South Dakota Department of Labor’s Workers’ Compensation Division is currently struggling with an outdated paper-based system that slows down processes and makes it challenging to provide quick and efficient support to injured workers. Moving to a digital case management system, where reports are submitted online and documents are stored electronically, is perfectly aligned with the department's goals. This transition will lead to quicker responses, reduced operational delays, and improved customer service. By modernizing their operations, the department can ensure that workers’ compensation claims are handled more effectively and in a timely manner, thus fulfilling its mission more efficiently.
* **Advocacy and Support:** Key leaders in the department who recognize the challenges of the current paper system will be crucial for advocating the digital transition. These project champions will ensure that the necessary resources, both financial and human, are dedicated to the project. Their role is essential in driving the project forward, maintaining focus on the goals, and overcoming any potential resistance to the change. Their involvement will be key to pushing the project to completion and ensuring a smooth transition from the old system to the new.

***Senior management:***

* **Decision-Making and Resource Allocation:** The success of this project is heavily dependent on the support from senior management. They need to approve the transition to a digital system and ensure the allocation of necessary funds and resources for development and implementation. Senior management’s endorsement is critical not only for securing the budget but also for signaling to the entire department that this project is a top priority. Their involvement will ensure that the project aligns with broader organizational goals and that it receives the attention and resources necessary to succeed.

***Users:***

* **Efficiency and Usability:** The primary users of the new system will be the clerks and administrative staff who currently manage the paper files. The current system requires a time-consuming manual process of locating files from large cabinets and cross-referencing spiral notebooks, which can result in delays and errors. The new digital system will allow users to query information quickly by typing in a claimant’s name or other identifying information, saving time and reducing errors. By involving these users in the design process, the system can be tailored to meet their needs, ensuring that it is user-friendly, which will lead to high adoption rates and greater operational efficiency.

***Conclusion:***

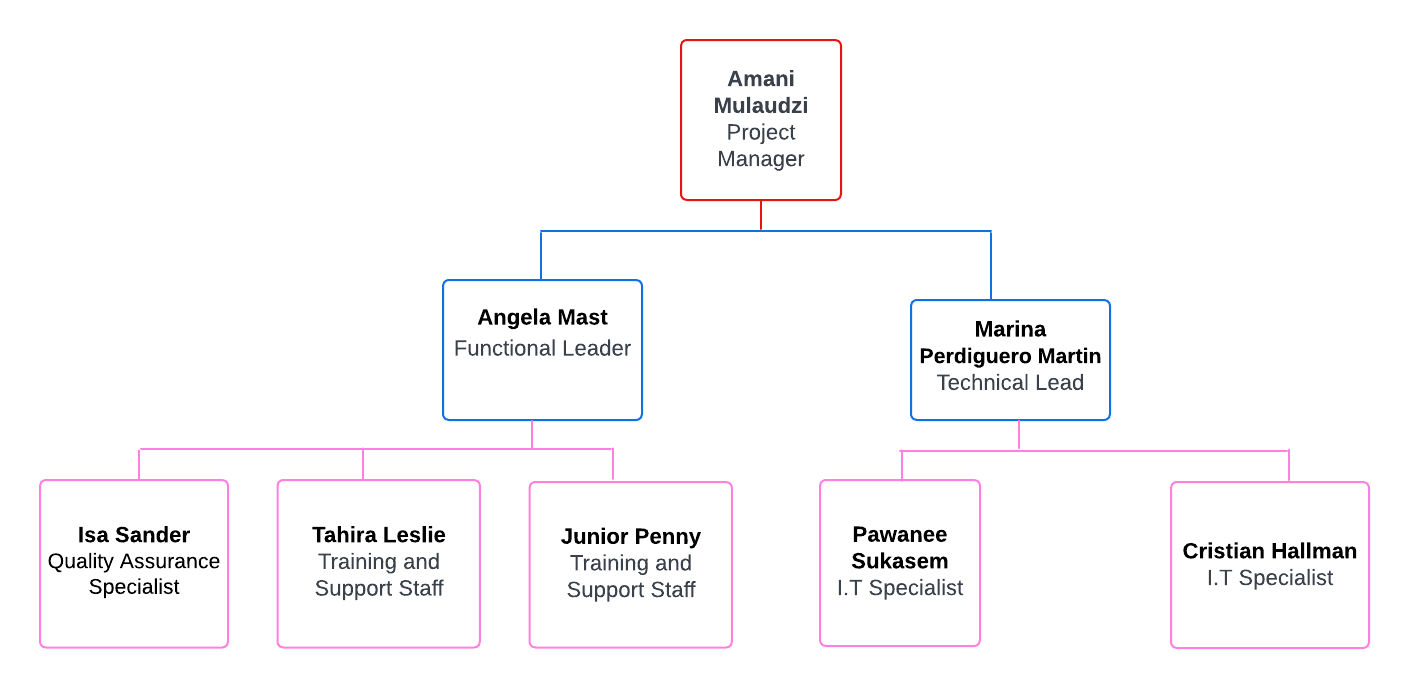
From an economic perspective, the long-term benefits of this digital transition outweigh the initial development and implementation costs. Increased efficiency, lower operational costs, and improved service delivery will provide significant returns on investment. With strong support from project champions and senior management, along with input from the end-users, the project is well-positioned for success. The digital system will modernize the Workers’ Compensation Division’s processes, allowing the department to better fulfill its mission of providing fair and timely support to injured workers. Economically, this project is a sound investment that will benefit the department and its stakeholders for years to come.

# V. Project Plan

## WorkPlan

| Phase | Task | Task Name | Assigned  To | Duration  (days) | Dependencies | Start  Date | Finish  Date |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 |  | Project Plan and Feasibility Study | Marina Perdiguero Martin | 20 |  | 8/13/2024 | 9/8/2024 |
|  | 1.1 | System Request | Marina Perdiguero Martin | 18 |  | 8/15/2024 | 9/8/2024 |
|  | 1.2 | Feasibility Study | Marina Perdiguero Martin | 17 | 1.1 | 8/16/2024 | 9/8/2024 |
|  | 1.3 | Technical feasibility | Junior Penny | 17 | 1.2 | 8/17/2024 | 9/8/2024 |
|  | 1.4 | Economic feasibility | Junior Penny | 17 | 1.3 | 8/18/2024 | 9/8/2024 |
|  | 1.5 | Organizational feasibility | Junior Penny | 17 | 1.4 | 8/19/2024 | 9/8/2024 |
|  | 1.6 | Workplan | Isa Sander | 12 | 1.5 | 8/23/2024 | 9/8/2024 |
|  | 1.7 | Staffing Plan | Isa Sander | 7 | 1.6 | 8/30/2024 | 9/8/2024 |
|  | 1.8 | Standard List | Isa Sander | 7 | 1.7 | 8/31/2024 | 9/8/2024 |
|  |  |  |  |  |  |  |  |
| 2 |  | System Requirements | Amani Mulaudzi | 12 | 1 | 9/8/2024 | 9/22/2024 |
|  | 2.1 | Functional Requirements | Amani Mulaudzi | 11 | 1.8 | 9/9/2024 | 9/22/2024 |
|  | 2.2 | Nonfunctional Requirements | Amani Mulaudzi | 11 | 2.1 | 9/10/2024 | 9/22/2024 |
|  | 2.3 | Business Requirements | Amani Mulaudzi | 6 | 2.1 | 9/16/2024 | 9/22/2024 |
|  | 2.4 | System Requirements | Cristian Hallman | 6 | 2.2 | 9/16/2024 | 9/22/2024 |
|  | 2.5 | User Requirements | Cristian Hallman | 4 | 2.2 | 9/18/2024 | 9/22/2024 |
|  |  |  |  |  |  |  |  |
| 3 |  | Use Case Analysis | Pawinee Sukasem | 12 | 2 | 9/22/2024 | 10/6/2024 |
|  | 3.1 | Create Use Cases | Pawinee Sukasem | 10 |  | 9/24/2024 | 10/6/2024 |
|  | 3.2 | Explanation of Use Cases elements and relationships | Pawinee Sukasem | 9 | 3.1 | 9/25/2024 | 10/6/2024 |
|  | 3.3 | Use Cases and Primary Output | Tahira Leslie | 6 | 3.1 | 9/30/2024 | 10/6/2024 |
|  |  |  |  |  |  |  |  |
| 4 |  | Process Model | Angela Mast | 17 | 3 | 10/6/2024 | 10/27/2024 |
|  | 4.1 | Level 0 Data Flow Diagram | Angela Mast | 16 | 3.3 | 10/7/2024 | 10/27/2024 |
|  | 4.2 | Level 1 Data Flow Diagram | Angela Mast | 11 | 4.1 | 10/14/2024 | 10/27/2024 |
|  | 4.3 | Process Description | Amani Mulaudzi | 8 | 3.3,4.1 | 10/17/2024 | 10/27/2024 |
|  |  |  |  |  |  |  |  |
| 5 |  | Data Model | Isa Sander | 12 | 4 | 10/27/2024 | 11/10/2024 |
|  | 5.1 | Entity Relationship Diagrams | Isa Sander | 11 | 4.3 | 10/28/2024 | 11/10/2024 |
|  | 5.2 | Data Dictionary | Isa Sander | 11 | 4.3 | 10/28/2024 | 11/10/2024 |
|  | 5.3 | Entity Descriptions | Tahira Leslie | 7 | 5.2 | 11/1/2024 | 11/10/2024 |
|  | 5.4 | Attribute Descriptions | Tahira Leslie | 7 | 5.3 | 11/1/2024 | 11/10/2024 |
|  | 5.5 | Relationship Descriptions | Marina Perdiguero Martin | 7 | 5.4 | 11/1/2024 | 11/10/2024 |
|  | 5.6 | Data Normalization | Marina Perdiguero Martin | 4 | 5.5 | 11/6/2024 | 11/10/2024 |
|  |  |  |  |  |  |  |  |
| 6 |  | System Proposal | Junior Penny | 17 | 5 | 11/10/2024 | 12/1/2024 |
|  | 6.1 | Development of System Proposal | Angela Mast | 16 | 1-5 | 11/11/2024 | 12/1/2024 |
|  | 6.2 | Presentation to Management | Angela Mast | 16 | 6.1 | 11/11/2024 | 12/1/2024 |

## Staffing Plan



## Standards List

1. Documentation Standards
   * Date and Project Name: All documents related to the "Too Much Paper" project must display the project name and date in the header to facilitate easy document tracking and management.
   * Margins: Document margins should be set to 1 inch on all sides for a proper format and easy readability.
   * Version Control: All documents must follow strict version control. Each version should have a recorded number and change log to avoid confusion during updates.
   * Digital Submission: All documents must be submitted through a secure online system. Medical reports and other documents can be sent in PDF or digital formats to ensure quick and efficient document management.
   * Storage: All documents must be stored in a secure, searchable digital database to reduce the burden of handling paper documents.
2. Coding Standards
   * Header Information: Each code module must include a header with the author’s name, last update date, and a description of the code’s purpose, making it easier for future maintenance and tracking.
   * Indentation: Code must use clear indentation, particularly for loops, if-else conditions, and case statements. Proper indentation helps with readability and reduces the risk of errors.
   * Commenting: Comments should be included every 5 lines of code to help others understand how the code works. These comments will assist with future maintenance and improvements.
3. Procedural Standards
   * Task Progress Reporting: Task progress must be recorded in the Project Work Plan, with updates every Tuesday and Thursday after class to keep track of ongoing work.
   * Classroom Meetings: Project updates and team meetings will occur every Tuesday and Thursday after class to discuss progress, solve problems, and update the status of tasks.
   * Change Management: Any changes to system requirements or tasks must be approved by the instructor or group leader before proceeding to prevent unplanned impacts on the main project plan.
4. Specification Requirement Standards
   * Program Name: The name and details of the program being developed must be clearly stated to ensure everyone understands the program’s purpose.
   * Business Rules: Clearly define the business rules that will be included in the system, such as verifying medical information and handling employee claims.
   * Data Security: Since the system handles medical and claim data, it must meet high security standards according to privacy laws to prevent unauthorized access to sensitive information.
   * Pseudocode: Pseudocode should be written to explain the program’s logic clearly before actual coding takes place.
   * Due Date: Each phase’s due date must be clearly specified according to the Project Work Plan to ensure timely completion and proper project control.
5. User Interface Design Standards
   * User-Friendly Interface: The system’s interface must be designed for ease of use, considering both employees and external users, and must provide quick access to reduce processing time.
   * Label Formatting: Labels and input fields must be in bold, left-aligned, and followed by a colon for clarity and readability.
   * Tab Order: The tab movement order must be set to move from the top left to the bottom right to ensure efficient navigation.
   * Hot Keys: Every editable field must have hotkeys to allow users to access and update data quickly and conveniently.
   * Search Functionality: The system must include efficient search functionality, allowing users to search by name, address, or event date, enabling employees to access information quickly without wasting time searching manually.
6. Use cases
   * Format: The use cases should follow the casual use case format which consists of multiple paragraphs which cover the main success scenario and alternative success and failure scenarios. This format is a simple, less formal way of documenting use cases and it is used when the complexity of the system or task is low or when a detailed description is unnecessary.
   * Elements: The information included in the use cases should include:
     + Use Case Name:The title of the use case should clearly indicate the action or process being documented
     + ID: A unique identifier for each use case (e.g., UC-001), which helps track and organize multiple use cases.
     + Priority: Indicates the importance of the use case in the system's overall functionality (e.g., High, Medium, Low).
     + Brief Description: A short summary explaining the purpose of the use case and what it aims to achieve.
     + Actor: Identifies the primary users or systems involved in the use case, such as "Clerk," "Medical Provider," or "System."
     + Trigger: The event that initiates the use case, such as a user submitting a claim or the system detecting incomplete information.
     + Type: Specifies whether the trigger is external (user-initiated) or temporal (time-based event).
     + Preconditions: Lists the conditions that must be met before the use case can be executed, such as user login and system availability.
     + Normal Course: Describes the step-by-step process for successfully completing the use case. This includes actions taken by users and the system's responses.
     + Information for Steps: Details the inputs required at each step and the outputs produced by the system.
     + Postconditions: The state of the system after the use case has been successfully executed, such as an updated claim or a notification sent.
     + Exceptions: Lists potential issues that could arise during the execution of the use case and how the system handles these issues.
     + Summary of Input and Output: A summary of the inputs received from users or other sources, and the outputs generated by the system. This includes data such as claim IDs, reports, and notifications.
   * Examples: The use cases document should include cases like Update Claim Information, Notify Medical Provider of Missing Information ,Search Claims by Name and Date of Injury ,Notify Employer of Claim Submission ,Upload Medical Reports, Submit Injury Claim ,Archive old claims and Generate claim report.
7. Data Flow Diagrams (DFD)
   * All the DFDs should be developed in the Lucidchart application.
   * All the shapes should follow the Data Flow elements in said application.
   * There must be one DFD context level, one DFD level 0 and as many DFD level 1 as processes.
   * Every primitive process in DFD level 1 should have a process description with the process name, number, short description, process description and notes.
8. Entity Relationship Diagrams (ERD)
   * All the ERDs should be developed in the Lucidchart application.
   * All the shapes should follow the Entity Relationship elements in said application.
   * There must be one ERD for the process explaining the details with all the entities needed.
   * The relationships between the entities must be specified using the arrows provided in the Lucidchart application.
   * Every entity, attribute and relationship should have a process description with the name, description, additional information according to each type and notes.

## Risk Assessment

**1. - Budget Overruns Due to Unforeseen Development Costs**  
Likelihood of risk:Medium probability of risk

Potential impact: This risk could lead to an increase in project costs by a significant margin, potentially exceeding the original budget by 20-40%. Factors contributing to these additional expenses include unexpected issues during development, the need for extra resources, extended timelines, and unforeseen complexities in the technology or scope changes. These cost overruns could affect the project's delivery schedule, quality, and stakeholder confidence.

Mitigation:To mitigate this risk, a contingency fund should be built into the initial budget to cover unexpected expenses. This ensures financial flexibility. It's also important to closely monitor the budget throughout the project's duration by conducting regular reviews and adjusting the financial plan as needed. Proactively identifying any deviations from the budget early allows the team to implement corrective measures, such as adjusting project scope or reallocating resources to stay within the budget.

**2.- Inaccurate File Tagging and Classification**

Likelihood: Medium. When transitioning from a manual, paper-based system to a digital solution, there is a risk that files may be incorrectly tagged, misclassified, or not properly indexed in the database. Human error during data entry or incorrect scanning of documents could lead to misfiled claims.

Potential Impact: High. Incorrectly classified files could delay claim processing, as employees may struggle to locate the correct file or information when queried. This could lead to inefficiency, lost productivity, and frustration for both employees and claimants. In the worst-case scenario, it could result in missed deadlines for important legal or medical reports.

Mitigation: Implement automated file classification tools, such as Optical Character Recognition (OCR), to reduce manual errors. Perform regular audits of digital files to ensure proper tagging and classification. Involve experienced staff in the quality control process to verify that files are accurately digitized and organized.

**3.- Inadequate Project Timeline Estimation**Likelihood: Medium probability of risk

Potential impact: This risk may result in project delays, with tasks taking longer than originally planned. It could impact the project's overall progress and lead to rushed work, reducing quality. Additionally, it may cause team stress and stakeholder dissatisfaction due to unmet deadlines. The project's completion could be delayed by weeks or even months, impacting both costs and resource allocation.

Mitigation:To mitigate this risk, involve an experienced project management team that can assess the project's complexity and develop a realistic timeline. This should include buffer periods for testing, troubleshooting, and unforeseen issues. Regularly reviewing and adjusting the timeline based on project progress will also help ensure the project remains on track without sacrificing quality.

**4.- Data Breach or Unauthorized Access During Development**Likelihood**:** High probability of risk

Potential impact**:** A data breach could severely damage the project's reputation, resulting in financial losses, legal penalties, and a loss of stakeholder trust. Additionally, it may require costly remediation, including system fixes, data recovery, and increased security measures. The project could also face significant delays due to the time needed to address these issues.

Mitigation: To mitigate this risk, implement robust security protocols from the start. This includes using encryption for all data transfers, ensuring secure access management with strong authentication controls, and conducting periodic security audits to identify and resolve vulnerabilities. Additionally, enforcing strict data handling policies and regular employee training will help protect sensitive information throughout the development process.

**5.- Inadequate Security Testing of the Digital System**Likelihood: Medium probability of risk

Potential impact:Inadequate security testing could lead to severe breaches, resulting in data loss, financial damage, and loss of user trust. The organization may face legal and regulatory consequences, along with costly post-launch fixes, which can delay the project further and increase overall costs.

Mitigation: To mitigate this risk, perform comprehensive security testing before the system's deployment. This should include penetration testing to identify potential attack vectors and vulnerability assessments to address any weaknesses. Regular security reviews during development phases will also help ensure that any security gaps are resolved promptly.

**6.- Lack of Familiarity with Digital Systems**

Likelihood: High probability of risk

Potential impact: This risk could result in extended development time and may increase project costs by up to 30%. The reasons for this increase include the time required for the team to learn and adapt to the new system, additional costs for training, hiring external consultants with technical expertise, potential rework to correct errors, and modifications to meet required standards that might not be achieved initially.

Mitigation: It’s essential to allocate time and resources for upfront training on digital document management and databases. Working with experienced consultants will also be beneficial. Phasing the implementation can help identify and address issues early, reducing the overall risk.

# VI. JAD Workshop

| **Requirements Workshop – JAD Session** |
| --- |
| Type of Meeting: JAD Methodology |
| Project:South Dakota Department of Labor, Workers’ Compensation division new claim software |
| Date of Session: 9/10/2024 |
| Location of Session: South Dakota Department of Labor Conference Room |
| Facilitator: Amani Mulaudzi |
| Scribe: Isa Sander |
| Timekeeper: Tahira Leslie |
| Attendees: Junior Penny, Isa Sander, Cristian Hallman, Pawinee Sukasem, Tahira Leslie,  Marina Perdiguero, Angela Mast, Amani Mulaudzi |
| Please Read: System Request and Project Background |
| Please Bring: Laptop, this agenda, good ideas |
| **Agenda**   | Time Box | Item | Description | | --- | --- | --- | | 8:00 - 8:10 | Introduction | Amani introduces attendees and the scenario and shares agenda and rules. | | 8:10 - 8:15 | Statement of project objective | Marina states the objectives of the project. | | 8:15 - 8:25 | Purpose of the meeting | Amani will explain that we will determine the  functional and nonfunctional requirements of the project. | | 8:25 - 9:25 | Current Workflow Analysis | Pawinee discusses the current process for handling injury claims, including document storage and retrieval. | | 9:25 - 10:15 | Identify key functional requirements | Angela discusses and Isa documents the functional requirements for the new digital system. | | 10:15 - 11:00 | Identify key nonfunctional requirements | Angela discusses and Isa documents the nonfunctional requirements for the new digital system. | | 11:00 - 11:30 | Break |  | | 11:30 - 12:00 | Reduction of staff | Amani will lead a discussion on the reduction of staff and how it will affect the business. | | 12:00 - 1:00 | Financial Factors | Junior Penny states financial considerations and factors for the development of the system. | | 1:00 - 1:45 | Security and Access Control | Cristian Hallman discusses security measures needed for sensitive information (e.g., secure login, data encryption, access roles). | | 1:45 - 2:15 | User needs | Marina discusses the user needs and requirements. | | 2:15 - 2:45 | Break |  | | 2:45 - 3:30 | System Integration | Pawinee discusses system integration and possible risks and solutions. | | 3:30 - 3:50 | Next steps | Confirm and assign follow-up tasks | | 3:50 - 4:00 | Wrap-up | Thanks, and adjournment | |
|  |

# VII. Discussion of the Current System

**Summary of the Current System**

The Workers' Compensation section of the South Dakota Department of Labor currently uses a system that primarily relies on physical paper files to manage claims. This leads to a number of operational inefficiencies, especially in terms of updating and retrieving case data. The procedure is laborious and error-prone because clerks must manually search through all the files for case files based on missing information such as names or dates of injury. Further delays have been caused by the physical storage problems resulting from the high volume of paper files, which are kept in big filing cabinets.

Moreover, clerks must first take a message, physically obtain the file, and then call the person back whenever a claimant or company calls to find out the status of a claim. This procedure is laborious and inefficient, which increases staff burden and the time it takes to respond to questions. These inefficiencies are made worse by the complexity and variation in case file sizes (some tiny, others enormous with multiple medical reports).

**Definition of Improvements Needed**

**Digital Claims Management:** The suggested upgrade calls for a completely digital claims management system to take the place of the current paper-based one. With the use of this system, claims may be electronically filed, monitored, and processed, doing away with the requirement for human retrieval and physical storage.

**Faster Data Retrieval:** By querying the system using claimant data (such as name and date of injury), clerks should be able to swiftly obtain claim details. The system should retrieve pertinent documents in a matter of seconds. This would cut down on how long it takes to respond to questions from claimants.

**Online Submission gateway:** To enable direct submission of claims and reports through the system by medical providers and claimants, an online gateway ought to be implemented. Clerks would handle less paperwork as a result, and everyone would receive real-time information.

**Electronic Document Management:** To simplify document management and retrieval, medical records and other documents should be submitted and kept electronically as faxed digital files or uploaded PDFs. Large filing cabinets would also be unnecessary as a result, freeing up actual office space.

**Automated alerts:** In order to minimize manual follow-ups by clerks, the system should automatically send alerts to claimants and other stakeholders when significant developments occur (such as claim approval or requests for more documentation).

**Better Security and Compliance:** To guarantee that sensitive information is safeguarded and that the department complies with local, state, and federal laws like HIPAA, the digital system must include strong security features like data encryption, access limits, and audit trails.

**Data Analysis and Reporting:** The department should be able to monitor KPIs like caseloads and claim processing times thanks to the system's reporting features. Better decision-making and resource allocation would result from this.

# VIII. Systems Analysis

| **Systems Analysis** |
| --- |
| Prepared by: Amani Mulaudzi, Angela Mast, Pawinee Sukasem, Marina Perdiguero |
| Functional requirements:   * Process Oriented:   + The system must allow the ability to upload records online through pdf or fax, as well as medical records from health professions.   + The system must have the ability to search individuals’ names and be able to access their records.   + The system must be able to check the claim statues of an individual.   + The system must be able to document dates when records are uploaded   + The system must be able to document claim data and record information about all claims an individual has made. * Information Oriented:   + The system must be able to hold a total of 20000 + 5% of growth per year of files.   + The system will archive the files that have not been accessed in 6 years.   + The system must be able to hold information about an individual's claim status.   + The system must provide access to claim data, allowing clerks to retrieve claims from previous years for reference or reporting.   + The system must categorize documents within each individual's file by type and allow users to filter or sort documents based on type and date. |
| Nonfunctional requirements:   * Operational:   + The hardware and software that are currently in use must work with the system.   + The system needs to offer a straightforward interface for medical practitioners to submit reports, and it should be easy to use even for clerks with no training.   + The system needs to function with several browsers and operating systems (Mac, Windows, Chrome, Edge, etc.) * Performance:   + It should take no more than five seconds to query a record by name or claim number.   + Except during planned maintenance, the system should be accessible for record uploading and querying around-the-clock.   + Up to 600 concurrent users should not cause performance issues for the system. * Security:   + The system must log all user activities, including login attempts, data access, and modifications, for auditing and compliance purposes.   + The system’s database is only available to employees at South Dakota Department of Labor, Worker’s compensation division.   + The system’s file uploading system will only be available to verified medical professionals and the individuals who are filing claims.   + Backups must be encrypted and performed daily, with a disaster recovery plan that ensures system recovery within 2 hours of a security-related incident. * Cultural and political:   + The system needs to abide by federal and state laws pertaining to the processing ›and storage of private medical and personal data.   + Multilingual support should be included in the solution to guarantee accessibility for South Dakotan users who do not speak English.   + Without requiring a lot of rewriting, the system should be built to support modifications to labor and compensation legislation in the future. |

# 

# IX. Use Cases

# 

| **Use Case Name:** Update Claim Information | | | | | | **ID:** UC-1 | | | | **Priority:** High | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Brief Description:** The admin manages user roles and access permissions in the system | | | | | | | | | | | | | |
| **Actor:** Clerk, Admin | | | | | | | | | | | | | |
| **Trigger:** New information needs to be added to an existing claim.  **Type:** External | | | | | | | | | | | | | |
| **Preconditions:**   1. The clerk is authenticated via the system login. 2. The claim must already exist in the system. 3. The system is online and operational. | | | | | | | | | | | | | |
| **Normal Course:** 1.0 Process for Updating a Claim in the System 1. The clerk provides the claim details of the claim that need to be updated.  2. The system sends the claim information to the claims database to search  3. The system sends the clerk the matched claim information  4. The clerk saves the updates, and the system confirms that the claim has been successfully updated. | | | | | | | | **Information for Steps:**  ← Claim details  → Claim information  → Claim search results  → Confirmation message | | | | | |
| **Postconditions:**   1. The claim is successfully updated with new information, and the system confirms the changes. 2. The updated claim is now accessible for future queries. 3. A log of the update, including the timestamp and clerk's ID, is created for auditing. | | | | | | | | | | | | | |
| **Exceptions:**  E1. System Downtime:   1. Occurs if the system experiences downtime or technical issues during the update process. 2. The system displays an error message and the clerk is unable to make updates.   E2. Invalid Data Entry:   1. Occurs if the clerk enters invalid or incomplete data during the update. 2. The system prompts the clerk to correct the invalid fields before proceeding.   E3. Insufficient Permissions:   1. Occurs if the clerk does not have permission to update certain claim fields. 2. The system displays an access error, preventing the update of restricted fields. | | | | | | | | | | | | | |
| **Summary of Input and Output** | | | | | | | | | | | | | |
| **Inputs**  Claim details | | **Source**  Clerk | | **Outputs**  Claim information  Claim search results  Confirmation message | | | | | | | | **Destination**  Clerk  Clerk  Clerk | |

| **Use Case Name:** Notify Medical Provider of Missing Information | | | | | | **ID:** UC-2 | | | | **Priority:** Medium | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Brief Description:** This use case involves the system automatically notifying a medical provider if essential information or documents are missing from their submission. The goal is to streamline the reporting process and ensure that claims can be processed without unnecessary delays**.** | | | | | | | | | | | | | |
| **Actor:** Medical Provider | | | | | | | | | | | | | |
| **Trigger:** When a medical report submission contains incomplete information or missing documents  **Type:** External | | | | | | | | | | | | | |
| **Preconditions:**   1. The medical provider must have submitted a medical report via the system. 2. The system must be able to identify missing or incomplete fields and must have access to the provider's contact information. | | | | | | | | | | | | | |
| **Normal Course:**  1.0 Process for Handling Missing Medical Report Information   1. The medical provider submits the medical report details to the system that automatically reviews the report for completeness, checking if all required fields and attachments are present. 2. If information is missing, the system automatically generates a notification detailing the missing required fields. 3. The system will send the resubmission instructions. 4. The medical updates the report and sends the updated submission. 5. The system sends a confirmation message that the updated fields have been saved. | | | | | | | | **Information for Steps:**    ← Medical report details  → Required fields  → Resubmission instructions  → Updated submission  → Confirmation message | | | | | |
| **Postconditions:**   1. The medical provider must have submitted a medical report through the system for an existing claim. 2. The system must be functioning properly, able to detect incomplete information or missing documents. 3. The provider's contact information (email or phone number) must be available in the system for notification. | | | | | | | | | | | | | |
| **Exceptions:**  E1. Incorrect Contact Information:   1. Occurs when the medical provider's contact details are incorrect or outdated. 2. The system prompts the clerk to verify and update the provider's information.   E2. Provider System Downtime:   1. Occurs if the provider's email system or network is unavailable. 2. The system retries after a delay or notifies the administrator. | | | | | | | | | | | | | |
| **Summary of Input and Output** | | | | | | | | | | | | | |
| **Inputs**  Medical report details | | **Source**  Medical provider | | **Outputs**  Required fields  Resubmission instructions  Updated submission  Confirmation message | | | | | | | | **Destination**  Medical Provider  Medical Provider  Medical Reports Database  Medical Provider | |

| **Use Case Name:** Search Claims by Name and Date of Injury | | | | | | **ID:** UC-3 | | | | **Priority:** Medium | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Brief Description:** A clerk searches the claims database for records by entering a claimant’s name and date of injury, allowing for quick retrieval of relevant claim information. | | | | | | | | | | | | | |
| **Actor:** Clerk | | | | | | | | | | | | | |
| **Trigger:** The clerk initiates the search process by entering the claimant’s name and date of injury.  **Type:** External | | | | | | | | | | | | | |
| **Preconditions:**   1. The clerk must be logged into the secure system with the appropriate role-based permissions to search the claims database. 2. The clerk must know the employee’s name and injury date. | | | | | | | | | | | | | |
| **Normal Course:**  1.0 Search Claim by Name and Date of Injury   1. Claimant or representative provides name and date of injury. 2. Clerk enters the provided name and date of injury into the system. 3. System searches the database for matching claims based on the provided details and displays them to the clerk. 4. Clerk selects the appropriate claim from the search results. 5. System retrieves the full details of the selected claim. 6. System sends the information desired to the clerk | | | | | | | | **Information for Steps:**  ← Claims name and Date of injury  → Claims name and Date of injury  → Claim search results  ← Selected claim ID  ← Selected claim information  → Selected claim information | | | | | |
| **Postconditions:**   1. Claim details are retrieved and displayed to the clerk. 2. The clerk provides the requested claim status to the caller. | | | | | | | | | | | | | |
| **Exceptions:**  E1. No Claims Found:   1. Occurs if no claims match the search criteria (Step 5). 2. Clerk is prompted to adjust the search parameters (e.g., by refining the date range or spelling of the name).   E2. Database Unavailable:   1. If the claims database is offline, the system displays an error message and the clerk is unable to complete the search. | | | | | | | | | | | | | |
| **Summary of Input and Output** | | | | | | | | | | | | | |
| **Inputs**  Claims name and Date of injury  Selected claim ID  Selected claim information | | **Source**  Clerk  Clerk  Claims Database | | **Outputs**  Claims name and Date of injury  Claim search results  Selected claim information | | | | | | | | **Destination**  Claims Database  Clerk  Clerk | |

| **Use Case Name:** Notify Employer of Claim Submission | | | | | | **ID:** UC-4 | | | | **Priority:** Medium | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Brief Description:** This use case describes how the system automatically notifies an employer when an employee submits a new injury claim. | | | | | | | | | | | | | |
| **Actor:** Employee | | | | | | | | | | | | | |
| **Trigger:** New claim submission by an employee.  **Type:** External | | | | | | | | | | | | | |
| **Preconditions:**   1. A new claim has been submitted by the employee. 2. The employer’s contact information is available in the system. | | | | | | | | | | | | | |
| **Normal Course:**  1.0 Notify Employer of Claim Submission   1. Employee submits a new injury claim through the system 2. System stores new claim in the claims database 3. System retrieves the employer’s contact information 4. System prepares a notification message for the employer 5. System sends the notification to the employer 6. System logs the notification in the claim record | | | | | | | | **Information for Steps:**  ← New injury claim  → New claim record  ← Employer contact details  → Notification message  → Notification message  → Notification log | | | | | |
| **Postconditions:**   1. Employer is notified of the claim submission. 2. Notification is logged in the claim's history. | | | | | | | | | | | | | |
| **Exceptions:**  E1. Employer’s Contact Information Not Found:   1. Occurs if the system is unable to locate the employer’s contact details (Step 3 of the normal course). 2. The system notifies the claims clerk to manually contact the employer and update the contact information in the database.   E2. Notification Sending Failed:   1. If the email or other notification to the employer fails (Step 5), the system retries a certain number of times and notifies the clerk if unsuccessful. | | | | | | | | | | | | | |
| **Summary of Input and Output** | | | | | | | | | | | | | |
| **Inputs**  New injury claim  Employer contact details | | **Source**  Employee  Claims database | | **Outputs**  New claim record  Notification message  Notification log | | | | | | | | **Destination**  Claims database  Employer  Claims database | |

| **Use Case Name:** Upload Medical Reports | | | | | | **ID:** UC-5 | | | | **Priority:** High | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Brief Description:** This use case showcases how medical providers can upload medical reports into this system via digital file or pdf file. | | | | | | | | | | | | | |
| **Actor:** Medical Provider | | | | | | | | | | | | | |
| **Trigger:** When a report needs to be uploaded to a client’s records  **Type:** External | | | | | | | | | | | | | |
| **Preconditions:**   1. The medical provider must be logged in and authorized to use the system. 2. The medical provider must be following HIPAA guidelines when uploading records. 3. The medical provider has already converted the records into a pdf or digital file. | | | | | | | | | | | | | |
| **Normal Course:**  1.0 Upload Medical Reports   1. The medical provider uploads records via a submission box. 2. System checks if the files are in pdf or other acceptable formats. 3. The medical provider confirms submission via the “submit” button. 4. The system saves the records under the patient’s profile with the date of submission. | | | | | | | | **Information for Steps:**  ← Medical report files    → Approval of file type    ← Uploaded reports | | | | | |
| **Postconditions:**   1. The medical reports are stored into the database. 2. The medical reports are given a unique ID, along with the date and time of submission being stored. | | | | | | | | | | | | | |
| **Exceptions:**  E1. No claim associated with patient   1. The patient is found 2. There is no claim associated with them 3. There will not be a prompted submission page. | | | | | | | | | | | | | |
| **Summary of Input and Output** | | | | | | | | | | | | | |
| **Inputs**  Medical report files  Uploaded records | | **Source**  Medical Provider  Medical Provider | | **Outputs**  Uploaded reports | | | | | | | | **Destination**  Medical Reports Database | |

| **Use Case Name:** Submit Injury Claim | | | | | | **ID:** UC-6 | | | | **Priority:** High | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Brief Description:** This use case describes how a client can submit an injury claim through the system. | | | | | | | | | | | | | |
| **Actor:** Employee client | | | | | | | | | | | | | |
| **Trigger:** When client needs to file an injury claim  **Type:** External | | | | | | | | | | | | | |
| **Preconditions:**   1. The employee must be able to navigate to the claims submission portal. 2. The employee must be in the acceptable timeframe of when the injury occurred in order to file a claim. | | | | | | | | | | | | | |
| **Normal Course:**  1.0 Submit Injury Claim   1. Employee selects the forum for a new claim submission and completes it with their injury information 2. Employee may upload relevant documents if they are available 3. Client ID and details are uploaded to the claims database 4. The system authorizes that all necessary fields in the forum are complete and appropriate | | | | | | | | **Information for Steps:**  ← Injury information  ← Relevant documents  → Client ID and details  → Approval submission | | | | | |
| **Postconditions:**   1. Claim is saved in the database and has a Claim ID associated with it 2. Employee is now notified of the approval of the submission via notification on webpage 3. Employee is notified of approval through their email | | | | | | | | | | | | | |
| **Exceptions:**  E1. Employee not authorized.   1. If the employee is not within the authorization or eligibility to complete a new claim, they will be denied access.   E2. File types are not accepted.   1. Submissions will be denied until accepted types of files are uploaded. | | | | | | | | | | | | | |
| **Summary of Input and Output** | | | | | | | | | | | | | |
| **Inputs**  Injury Information  Relevant documents | | **Source**  Employee  Employee | | **Outputs**  Client ID and details  Approval submission | | | | | | | | **Destination**  Claims Database  Employee | |

| **Use Case Name:** Archive old claims | | | | | | **ID:**UC-7 | | | | **Priority:** Low | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Brief Description:** Archive claims older than 6 years. | | | | | | | | | | | | | |
| **Actor:** System | | | | | | | | | | | | | |
| **Trigger:** The system will store a claim 6 years or older into an archive system | | | | | | | | | | | | | |
| **Type:** Temporal | | | | | | | | | | | | | |
| **Preconditions:**   1. Claims are older than 6 years from the current date. 2. The system has access to the archive storage location and sufficient storage capacity. | | | | | | | | | | | | | |
| **Normal Course:**  1.0 Archiving file   1. The system runs a scheduled task to identify claims that are 6+ years old. 2. The system retrieves the list of eligible claims and compiles them for archiving. 3. The system runs a scheduled task to identify medical reports that are 6+ years old. 4. The system retrieves the list of eligible medical reports and compiles them for archiving. | | | | | | | | **Information for Steps:**    ← Claim Date  → Claims older than 6 years  ← Medical Report Date      →Medical Reports older than 6 years | | | | | |
| **Postconditions:**   1. Claims older than 6 years are archived and marked as such in the primary system, improving database efficiency and reducing clutter. | | | | | | | | | | | | | |
| **Exceptions:**  E1. System unavailable   1. If the archive storage system is unavailable (e.g., due to network failure), the system logs the failure and retries at the next scheduled run. | | | | | | | | | | | | | |
| **Summary of Input and Output** | | | | | | | | | | | | | |
| **Inputs**  Claim Date  Medical Report Date | | **Source**  Claims Medical  Reports Database records | | **Outputs**  Claims older than 6 years  Medical Reports older than 6 years | | | | | | | | **Destination**  Archive Database  Archive Database | |

| **Use Case Name:** Generate claim report | | | | | | **ID:** UC-8 | | | | **Priority:** Medium | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Brief Description:** Generate a report summarizing the details of a claim, including the employee’s information, injury details, medical reports, and claim status. | | | | | | | | | | | | | |
| **Actor:** Clerk | | | | | | | | | | | | | |
| **Trigger:** The user (clerk or administrator) needs to generate a claim report, either for reviewing claim details, sharing with another party, or submitting as part of a case resolution. | | | | | | | | | | | | | |
| **Type:** External | | | | | | | | | | | | | |
| **Preconditions:**   1. The claim exists in the system and contains all necessary information (employee details, injury report, medical reports, claim status). | | | | | | | | | | | | | |
| **Normal Course:**  1.0 Generating claim report   1. The user logs into the system with appropriate access rights. 2. The user searches for the specific claim by either entering the claim ID or searching by employee details. 3. The user searches for the specific claim by either entering the Medical ID or searching by medical report details 4. The system compiles the report, including the employee’s personal details, injury details, medical report attachments, and current claim status and The report is displayed to the user for review. | | | | | | | | **Information for Steps:**    ← User credentials  ← Claim information    ← Medical reports      → Claim Report | | | | | |
| **Postconditions:**   1. A full or partial claim report is generated and available for download or print. | | | | | | | | | | | | | |
| **Exceptions:**  E1. System failure   1. The system fails to generate the report due to a system error.   E2. Corrupted file   1. If a medical report file is corrupted, the system alerts the user and skips that document in the report generation. | | | | | | | | | | | | | |
| **Summary of Input and Output** | | | | | | | | | | | | | |
| **Inputs**  User credentials  Claim information  Medical reports | | **Source**  Clerk  Claims Database  Medical Reports Database | | **Outputs**  Claim Report | | | | | | | | **Destination**  Clerk | |

# 

# X. Process Model

**DFD Context Level**

# 

# 

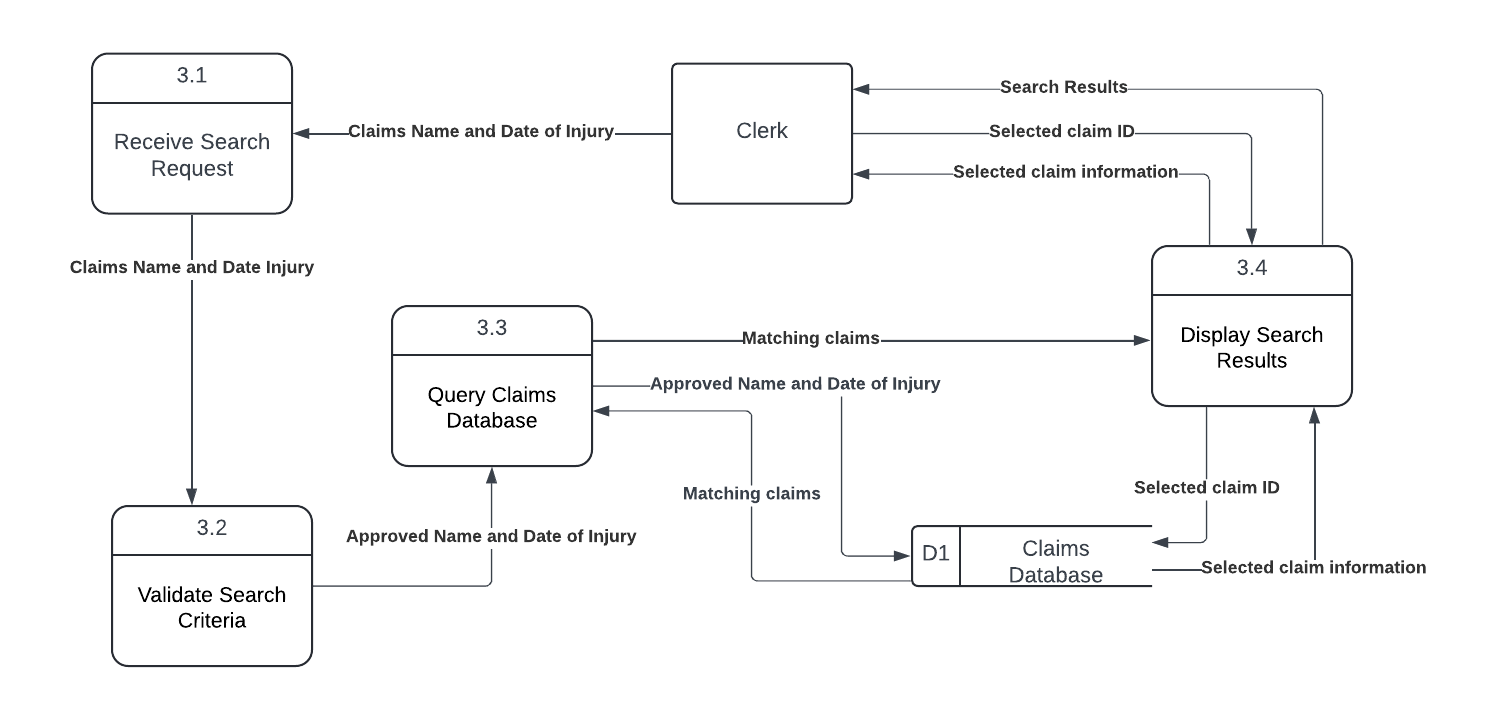
# DFD Level 0

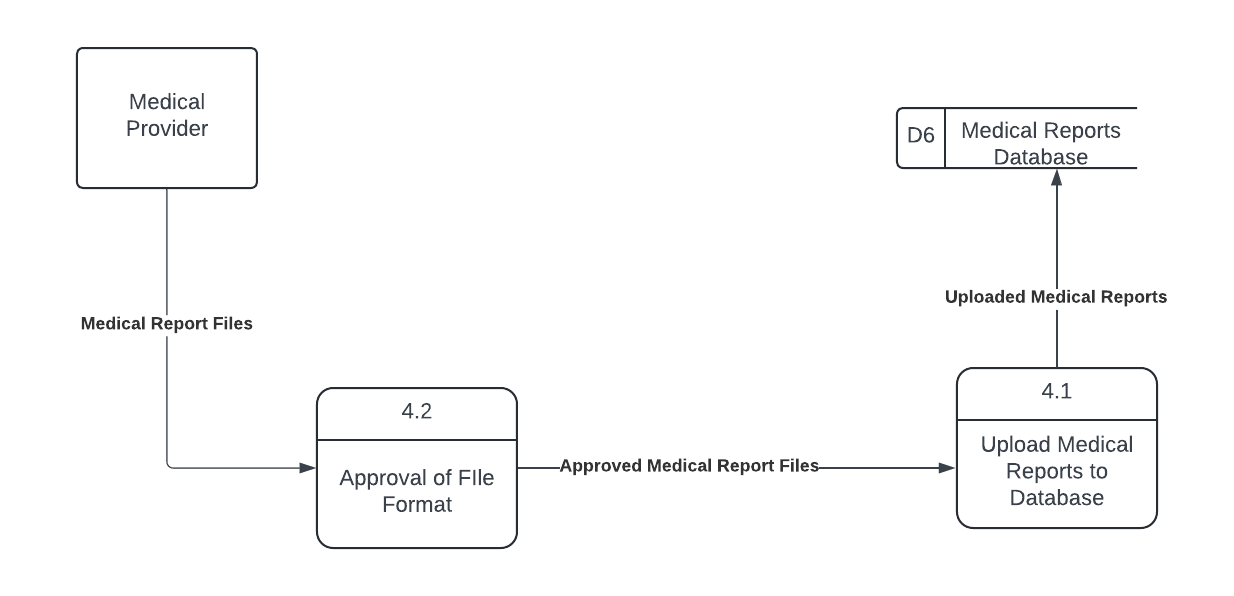
# DFD Level 1

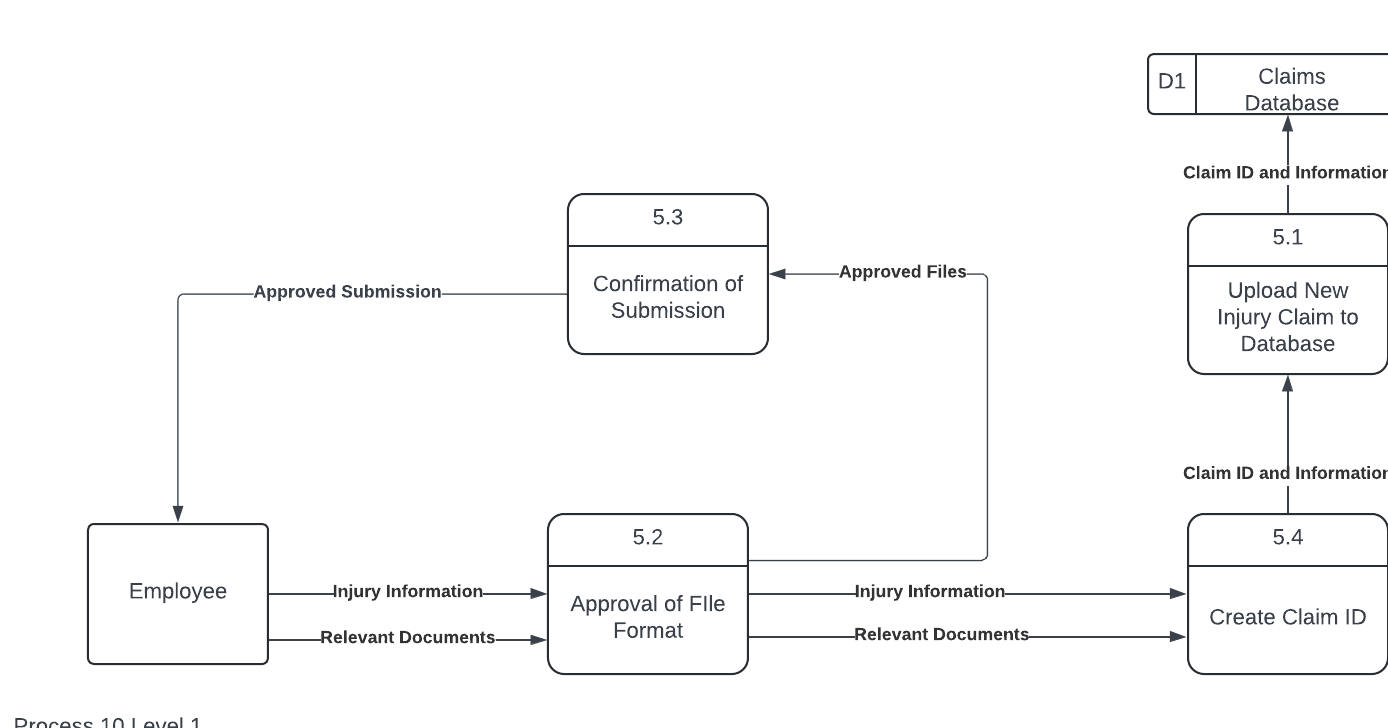
# 

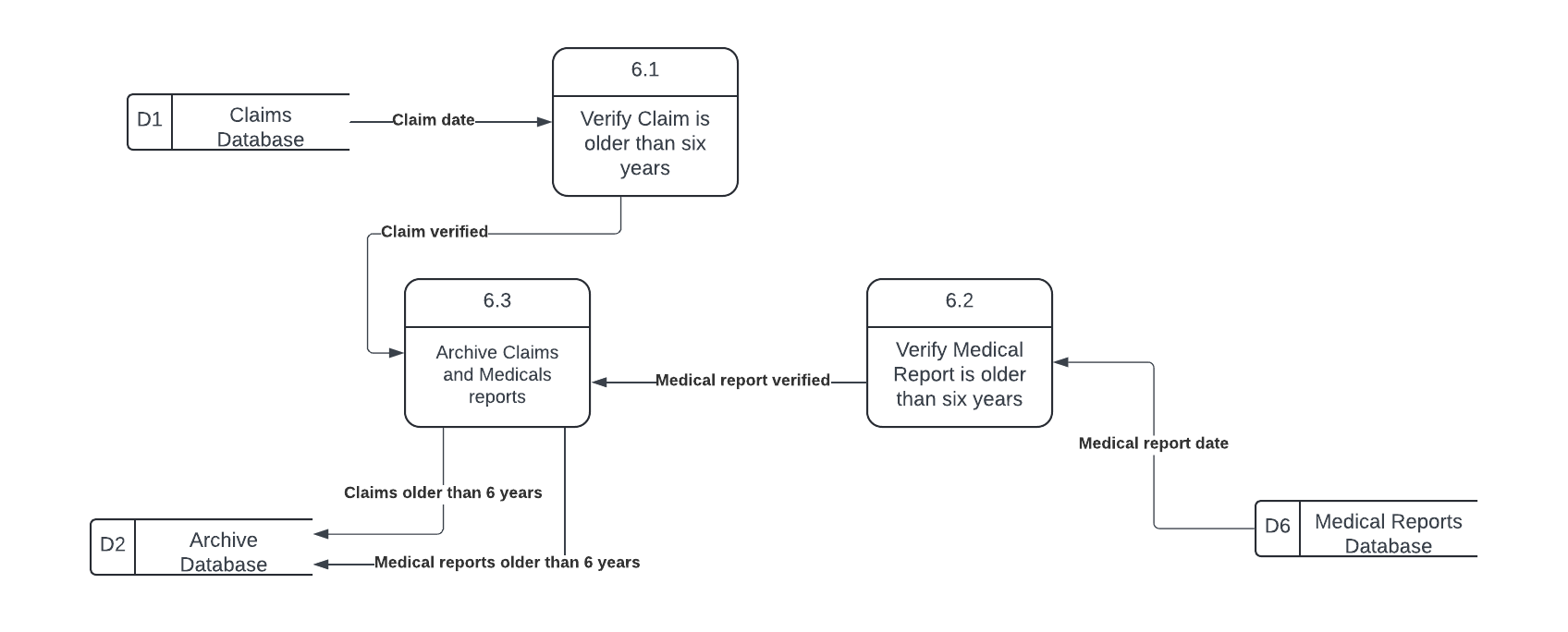
# 

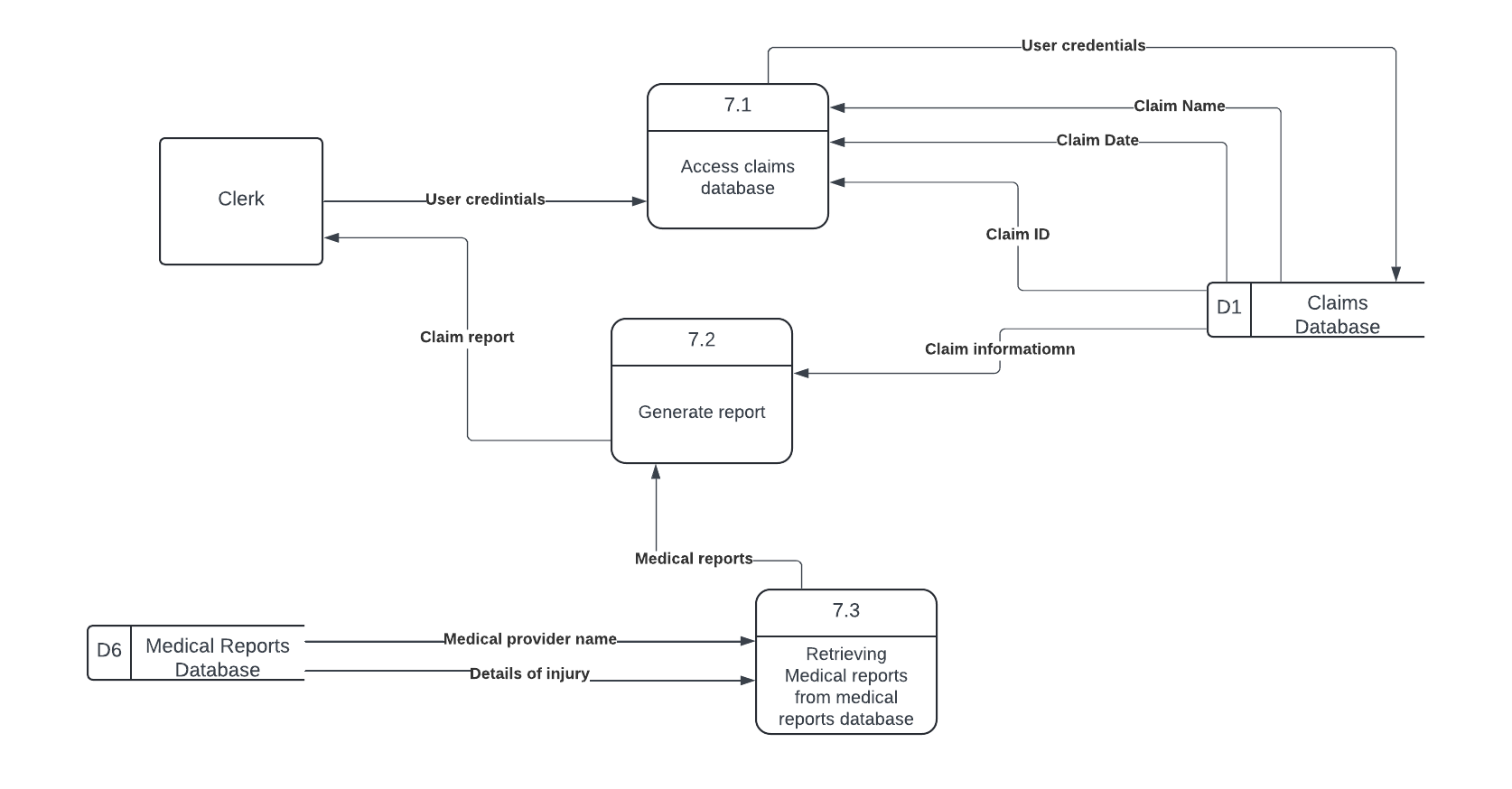
# 

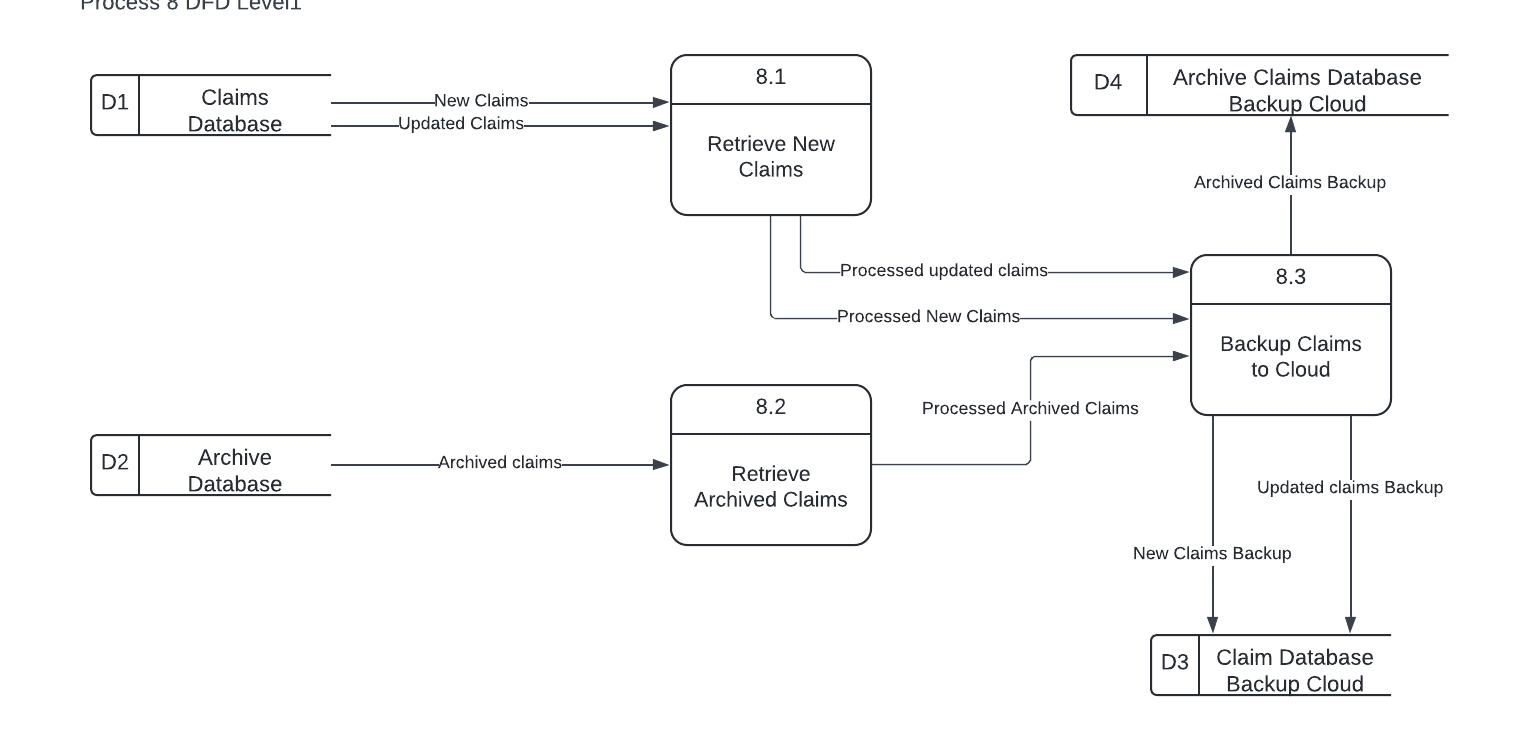


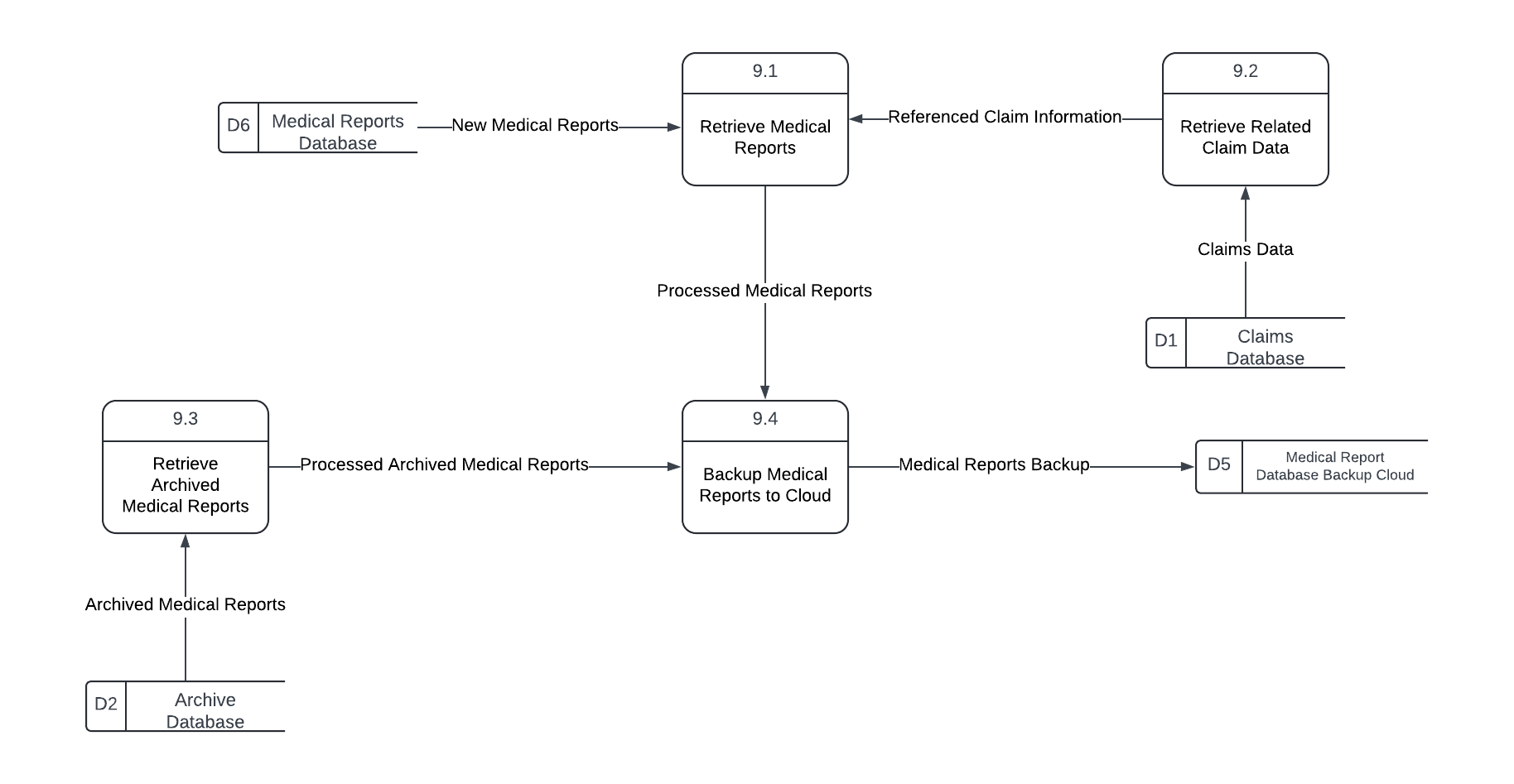


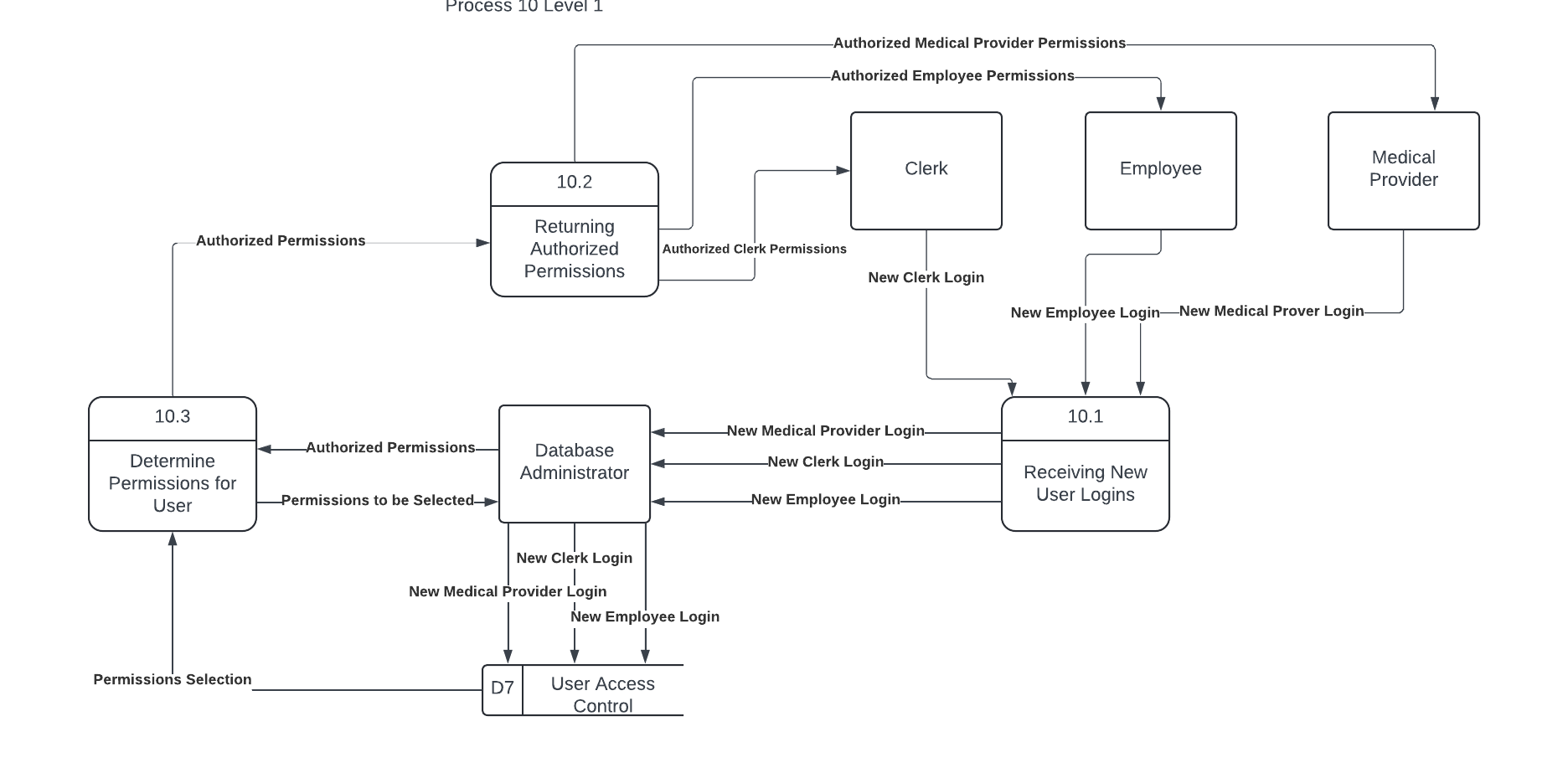


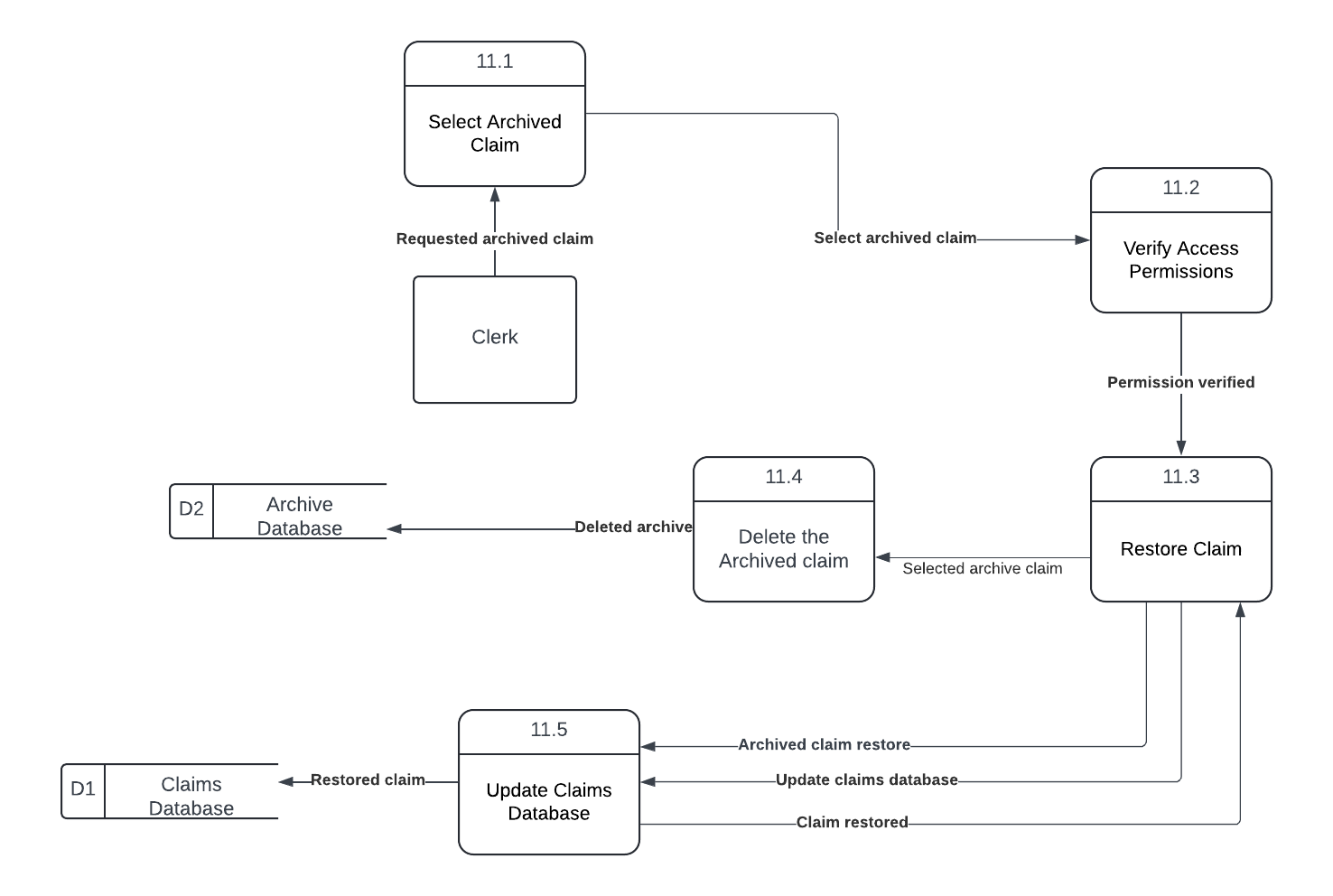












**Process Descriptions**

| **Process Description** |
| --- |
| Process Name: Receive Update Request |
| Process Number: 1.1 |
| Short Description: This process receives and processes the update request from the clerk. |
| Process Description:  Receive the update request from the clerk containing claim details.  Validate that the request is properly formatted and contains necessary details.  Forward the update details to the Validate Update Criteria process. |
| Notes: Ensures all necessary information is provided for the update process. |

| **Process Description** |
| --- |
| Process Name: Validate Update Criteria |
| Process Number: 1.2 |
| Short Description: This process verifies that the update request meets all necessary criteria. |
| Process Description:  Receive update details from the Receive Update Request process.  Check the details to ensure they meet predefined update criteria.  If criteria are met, forward validated data to the Update Claims Database process.  If criteria are not met, send an error message back to the clerk. |
| Notes: Prevents incomplete or incorrect updates from being processed. |

| **Process Description** |
| --- |
| Process Name: Update Claims Database |
| Process Number: 1.3 |
| Short Description: This process updates the Claims Database with validated information. |
| Process Description:  Receive validated update data from the Validate Update Criteria process.  Update the Claims Database with the new information.  Send an updated success message to the clerk. |
| Notes: Reflects updated claim information in the database for future reference. |

| **Process Description** |
| --- |
| Process Name: Check Report Submission |
| Process Number: 2.1 |
| Short Description: This process verifies the initial submission of a medical report by the provider. |
| Process Description:  Receive medical report details from the Medical Provider.  Retrieve relevant fields from the Medical Reports Database.  Validate that the report meets submission requirements.  Forward report data to the Identify Missing Information process if necessary. |
| Notes: Reports are checked for completeness based on required fields. |

| **Process Description** |
| --- |
| Process Name: Identify Missing Information |
| Process Number: 2.2 |
| Short Description: This process identifies any missing information in the submitted report. |
| Process Description:  Receive report data from the Check Report Submission process. Analyze report fields to identify missing or incomplete information. Mark missing information and forward it to the Send Resubmission Instructions process. If all information is present, send confirmation to the provider. |
| Notes: This step ensures all required information is included before processing |

| **Process Description** |
| --- |
| Process Name: Send Resubmission Instructions |
| Process Number: 2.3 |
| Short Description: This process sends instructions to the provider to complete missing fields in the report. |
| Process Description:  Receive missing information details from the Identify Missing Information process.  Compile instructions for resubmission based on identified missing data.  Send resubmission instructions back to the Medical Provider.  Notify the provider to complete and resubmit the report. |
| Notes: Instructions are customized to the specific missing fields identified. |

| **Process Description** |
| --- |
| Process Name: Updated Submission |
| Process Number: 2.4 |
| Short Description: This process updates the medical report submission with additional data provided. |
| Process Description:  Receive updated report submission from the Medical Provider.  Verify the additional data meets all submission requirements.  Store the updated report in the Medical Reports Database.  Send a confirmation message to the provider once submission is accepted. |
| Notes: The updated report replaces any prior incomplete submission. |

| **Process Description** |
| --- |
| Process Name: Receive Search Request |
| Process Number: 3.1 |
| Short Description: This process receives a search request from the clerk for claims data based on the claim name and date of injury. |
| Process Description: In this process, the clerk initiates a search request by providing specific search criteria, including the claimant's name and date of injury. The system receives this request, validates the input criteria for completeness and format, and then forwards the request to the relevant processes for retrieving matching claims data. Once validated, the request is prepared to be processed by the "Search Claims by Name and Date of Injury" function, which will query the appropriate databases. |
| Notes:   * This process includes input validation to ensure search criteria are properly formatted. * If the input criteria are incomplete or improperly formatted, the system prompts the clerk to correct the input. * After validation, the process routes the request to subsequent processes that handle data retrieval. |

| **Process Description** |
| --- |
| Process Name: Validate Search Criteria |
| Process Number: 3.2 |
| Short Description: This process validates the search criteria provided by the clerk to ensure they meet system requirements. |
| Process Description: The Validate Search Criteria process checks the search request input for completeness, accuracy, and proper formatting. It verifies that both the claimant's name and date of injury fields are filled in and adhere to the required format (e.g., name is a string of text, date is in MM/DD/YYYY format). If the criteria do not meet these requirements, an error message is generated, prompting the clerk to correct and re-enter the search information. Once validation is successful, the search request is cleared to proceed to the next step in the claims retrieval process. |
| Notes:   * This process prevents invalid or incomplete search criteria from progressing through the system. * Error handling includes prompts for the clerk to correct specific fields as necessary. * Only valid search requests proceed to the data retrieval stage. |

| **Process Description** |
| --- |
| Process Name: Query Claims Database |
| Process Number: 3.3 |
| Short Description: This process retrieves claims information from the Claims Database based on the validated search criteria. |
| Process Description: The Query Claims Database process involves sending a search request to the Claims Database using the validated search criteria (claimant's name and date of injury). The system scans the database for records that match the search parameters. If matching records are found, relevant claim information, such as claim status, claim number, and other pertinent details, are retrieved. This data is then compiled and prepared for display or further processing. If no matching records are found, a "No Results" message is generated to inform the clerk. |
| Notes:   * This process assumes that the search criteria have been validated in the previous step. * The retrieved data is formatted to ensure compatibility with the output display. * Error handling includes a "No Results" message if no records match the search criteria. |

| **Process Description** |
| --- |
| Process Name: Display Search Results |
| Process Number: 3.4 |
| Short Description: This process presents the search results from the Claims Database to the clerk in an easily readable format. |
| Process Description: The Display Search Results process takes the retrieved claim information from the Query Claims Database process and formats it for clear presentation on the clerk's interface. This includes displaying key details such as claim number, claimant's name, date of injury, claim status, and any other relevant information. The results are arranged in a logical order for easy reference, with options for the clerk to view additional details if available. If no matching records are found, a "No Results Found" message is shown to inform the clerk. |
| Notes:   * The results are displayed in a user-friendly layout with clear headings and organized fields. * If there are multiple matching records, they are displayed in a list format for easy navigation. * The process supports error handling by displaying a "No Results Found" message when applicable. |

| **Process Description** |
| --- |
| Process Name: Upload Medical Reports to Database |
| Process Number: 4.1 |
| Short Description: This process describes how a medical provider can upload medical reports to the medical reports database data store. |
| Process Description:  Approved format medical reports are retrieved from 4.2  The approved format medical reports are uploaded to the D6 Medical Reports Database |
| Notes:  If the database is full, the medical reports will not be uploaded and an error message will display. |

| **Process Description** |
| --- |
| Process Name: Approval of File Format |
| Process Number: 4.2 |
| Short Description: This process checks the format of the medical reports and verifies it is in an acceptable format. |
| Process Description:  Medical provider uploads medical report files into the system.  Medical report files types are checked to verify current format.  IF Medical report files are not in the correct format  THEN Submission button will be unavailable and error message will appear  ELSE Medical report files are approved and in the correct format  The approved medical files are then sent to process 4.1 to upload them to the database |
| Notes:  Picture files, music files, and video files are not an allowed format.  PDF files, excel files, and digital fax files are an allowed format. |

| **Process Description** |
| --- |
| Process Name: Upload New Injury Claim to Database |
| Process Number: 5.1 |
| Short Description: This process describes how the new injury claim is uploaded to the claims database. |
| Process Description:  Claim ID, injury information, and relevant documents are all retrieved from process 5.4  The claim ID, injury information, and relevant documents are uploaded into the claims database D1 data store. |
| Notes:  If the claims database is full, the injury claim will not be submitted and an error message will appear. |

| **Process Description** |
| --- |
| Process Name: Approval of File Format |
| Process Number: 5.2 |
| Short Description: This process approves or denies the file format of new claim documents |
| Process Description:  Injury information and relevant documents are retrieved from the Employee.  IF the documents are not in an acceptable format  THEN the system will not allow submission, and an error message that mentions the documents that are in incorrect format will appear.  ELSE the documents are in an acceptable format and proceed to the next step.  The approved files are sent to process 5.3  The injury information and relevant documents are sent to process 5.4 |
| Notes: The amount of documents retrieved from the Employee can be large or small, but must not exceed over 100 different documents. |

| **Process Description** |
| --- |
| Process Name: Confirmation of Submission |
| Process Number: 5.3 |
| Short Description: This process prompts the employee to confirm their submission of claim material. |
| Process Description:  The approved files are retrieved from process 5.2  Approval of submission is then prompted to confirm the submission of their approved files  IF the employee does not approve the submission  THEN files will not be uploaded and will not go to process 5.4  ELSE proceed to process 5.4 |
| Notes:  Submission is required for the files to be uploaded into the claims database |

| **Process Description** |
| --- |
| Process Name: Create Claim ID |
| Process Number: 5.4 |
| Short Description: This process describes how a claim ID is created based on the approved injury information and relevant documents |
| Process Description:  Injury information and relevant documents are retrieved from process 5.2  A new claim ID is created  Claim ID and information is then sent to process 5.1 to be uploaded to the database |
| Notes:  The claim ID is unique for each claim, but all documents in a given claim will have the same claim ID. |

| **Process Description** |
| --- |
| Process Name: Verify Claim is older than 6 years |
| Process Number: 6.1 |
| Short Description: Checks if the claim date is older than six years. |
| Process Description: The process retrieves the claim date from the Claims Database and verifies if the claim is older than six years. If the claim meets this criterion, it moves to the next step for further verification. |
| Notes: This process ensures that only claims older than six years are considered for archiving. |

| **Process Description** |
| --- |
| Process Name: Verify Medical Report is older than six years |
| Process Number: 6.2 |
| Short Description: Checks if the medical report date is older than six years. |
| Process Description: This process retrieves the medical report date from the Medical Reports Database and verifies if it is older than six years. If the medical report meets this criterion, it is marked as verified for further archiving. |
| Notes: This process ensures that only medical reports older than six years are considered for archiving. |

| **Process Description** |
| --- |
| Process Name: Archive Claims and Medical Reports |
| Process Number: 6.3 |
| Short Description: Archives verified claims and medical reports. |
| Process Description: After the claim and medical report have both been verified as older than six years, this process archives them. The claims and medical reports are moved from the Claims Database and Medical Reports Database to the Archive Database. |
| Notes: This process completes the archival of all records older than six years for compliance and storage optimization. |

| **Process Description** |
| --- |
| Process Name: Access claims database |
| Process Number: 7.1 |
| Short Description: This process allows a clerk to retrieve relevant claim data from the Claims Database using user credentials and claim-specific information. |
| Process Description: The clerk inputs their credentials along with a claim ID, claim name, or claim date to search the Claims Database. The system retrieves the necessary claim information and sends it to the clerk for further use. |
| Notes: Secure access is required through user credentials.  Only authorized personnel can retrieve claim details. |

| **Process Description** |
| --- |
| Process Name: Generate Report |
| Process Number: 7.2 |
| Short Description: This process generates a comprehensive report based on the information retrieved from the Claims Database and medical reports. |
| Process Description: Using the information provided by the Claims Database and medical reports, this process compiles a detailed claim report. The report includes claim details, dates, medical provider information, and injury descriptions. The final report is sent to the clerk. |
| Notes: Input data is sourced from both the Claims and Medical Reports databases.  The report generation follows predefined templates and standards. |

| **Process Description** |
| --- |
| Process Name: Retrieving Medical reports from medical reports database |
| Process Number: 7.3 |
| Short Description: This process retrieves necessary medical information from the Medical Reports Database. |
| Process Description:Upon request, the system accesses the Medical Reports Database using medical provider names and injury details. It retrieves relevant reports and forwards them to the report generation process (7.2) . |
| Notes: Access to the Medical Reports Database may be restricted to ensure privacy and compliance. |

| **Process Description** |
| --- |
| Process Name: Retrieve New Claims |
| Process Number: 8.1 |
| Short Description: This process gathers newly added claims from the Claims Database for backup purposes. |
| Process Description: The Retrieve New Claims process identifies and extracts claims that have been added or modified since the last backup. It queries the Claims Database to gather this data, focusing only on records marked as "new" or "updated." Once retrieved, this data is prepared for transfer to the designated backup storage location to ensure that recent claims information is securely stored and available for recovery if needed. |
| Notes:   * This process runs periodically to ensure all new claims are backed up in a timely manner. * The process only retrieves records with a "new" or "updated" status to minimize redundancy in backups. * Any issues retrieving claims are logged for review to maintain data integrity in backups. |

| **Process Description** |
| --- |
| Process Name: Retrieve Archived Claims |
| Process Number: 8.2 |
| Short Description: This process retrieves archived claims data from the Archive Database to include it in the backup. |
| Process Description: The Retrieve Archived Claims process accesses the Archive Database to gather claims that have been archived. This process retrieves all relevant archived claim records and prepares them for secure transfer to the Archive Claims Database Backup Cloud. The purpose is to ensure that older, archived claims are also backed up and available for long-term storage, allowing for a complete data recovery if needed. |
| Notes:   * This process only targets records from the Archive Database, ensuring that active claims remain unaffected. * Retrieval occurs during scheduled backup operations to avoid impacting system performance. * Any issues with data retrieval are logged for auditing and resolution to ensure comprehensive backup coverage. |

| **Process Description** |
| --- |
| Process Name: Backup Claims to Cloud |
| Process Number: 8.3 |
| Short Description: This process securely backs up claim records to the Claims Database Backup Cloud. |
| Process Description: The Backup Claims to Cloud process involves transferring claim records, including both new and archived claims, to the Claims Database Backup Cloud. The process initiates after gathering data from the Claims Database and the Archive Database. It ensures that all recent claims, along with archived records, are copied to a secure cloud-based storage solution. The data is encrypted during transfer and upon storage in the cloud to maintain security and compliance with data protection standards. The backup process is monitored to verify successful data transfer, with error logs recorded if any issues occur. |
| Notes:   * The backup operation is scheduled to run during off-peak hours to optimize system resources. * Data encryption is used to protect sensitive information during transit and in the cloud. * If any backup errors occur, they are logged, and the process is re-initiated to ensure all data is securely backed up. |

| **Process Description** |
| --- |
| Process Name: Retrieve Medical Reports |
| Process Number: 9.1 |
| Short Description: This process gathers recent and relevant medical reports from the Medical Reports Database for backup purposes. |
| Process Description: The Retrieve Medical Reports process queries the Medical Reports Database to collect recent and newly updated medical report records. These records are marked for inclusion in the scheduled backup, ensuring that all recent medical documentation is preserved. The process specifically retrieves data added or modified since the last backup and prepares it for secure transfer to the Medical Report Database Backup Cloud, guaranteeing that important medical data is readily available for recovery. |
| Notes:   * The process selectively retrieves only records marked as “new” or “updated” to avoid redundancy in backups. * Retrieval is scheduled during designated backup times to minimize impact on system performance. * Any issues encountered during retrieval are logged for review and resolution to maintain the integrity of the backup. |

| **Process Description** |
| --- |
| Process Name: Retrieve Related Claim Data |
| Process Number: 9.2 |
| Short Description: This process gathers relevant claim data from the Claims Database that is associated with specific medical reports for backup. |
| Process Description: The Retrieve Related Claim Data process accesses the Claims Database to collect claim records that are linked to medical reports being backed up. This ensures that both medical and related claim information are stored together in the backup, preserving the contextual relationship between claims and associated medical documentation. The process identifies claim records based on references or connections to the retrieved medical reports and prepares this data for transfer to the designated backup location. |
| Notes:   * This process only retrieves claim data directly associated with specific medical reports to maintain efficiency and data relevance. * Retrieval occurs as part of the overall backup process to ensure consistency across databases. * Any retrieval errors or mismatches are logged and reviewed to ensure comprehensive backup integrity. |

| **Process Description** |
| --- |
| Process Name: Retrieve Archived Medical Reports |
| Process Number: 9.3 |
| Short Description: This process retrieves archived medical report data from the Archive Database to include it in the backup. |
| Process Description: The Retrieve Archived Medical Reports process queries the Archive Database to collect archived medical report records. This ensures that older medical reports, which may still be relevant for historical or legal purposes, are preserved in the backup. Once retrieved, these records are prepared for secure transfer to the Archive Claims Database Backup Cloud, providing comprehensive data storage for both current and historical medical reports. |
| Notes:   * This process focuses solely on archived medical reports, preventing duplication with active records. * Data retrieval occurs during scheduled backup times to optimize system performance. * Any retrieval issues are logged for troubleshooting, ensuring a complete and reliable backup process. |

| **Process Description** |
| --- |
| Process Name: Backup Medical Reports to Cloud |
| Process Number: 9.4 |
| Short Description: This process involves securely transferring and storing medical reports in a cloud-based system to ensure data integrity and accessibility. |
| Process Description: The Backup Medical Reports to Cloud process transfers current medical report records from the local database to a secure cloud storage system. This process ensures that all medical reports are safely stored, protecting against data loss due to hardware failure or other incidents. Before the backup, each report is verified for accuracy and encrypted to safeguard sensitive patient information during the transfer. Once successfully uploaded, a verification step confirms the integrity of the data, ensuring that the backup is both comprehensive and reliable. |
| Notes:   * Regular backups are scheduled weekly to keep medical report data current and reduce the risk of data loss. * Reports are encrypted prior to transfer to comply with health data regulations, ensuring patient confidentiality. * Any issues encountered during the backup process are logged for troubleshooting to maintain backup reliability. |

| **Process Description** |
| --- |
| Process Name: Retrieving New User Logins |
| Process Number: 10.1 |
| Short Description: This process retrieves new logins from all external entities and provides them to the database administrator so he may give them the permissions accordingly. |
| Process Description:  New login information is retrieved from the Clerk, Employee, and Medical Provider  IF the information matches one of an existing account  THEN the login information will be flagged with existing account tag before proceeding.  The new login information is then sent to the Database Administrator |
| Notes:  This process accounts for all external entities, but it is rare that they will happen all at the same time. |

| **Process Description** |
| --- |
| Process Name: Returning Authorized Permissions |
| Process Number: 10.2 |
| Short Description: This process returns the login information to each respective entity with permissions granted. |
| Process Description:  The authorized permissions are retrieved from process 10.3  The permissions are then sent back to the Employee, Medical Provider, and Clerk |
| Notes: After this process is conducted, each entity should now be able to login and have the acceptable accessibility needed. |

| **Process Description** |
| --- |
| Process Name: Determine Permissions for User |
| Process Number: 10.3 |
| Short Description: This process describes how the database administrator determines the accessibilities of each login request. |
| Process Description:  Permissions to be selected are sent to the Data administrator to choose what permissions each login should have.  The Data Administrator authorizes permissions and sends back to the process  The authorized permissions are sent to process 10.2 |
| Notes:  The database administrator can choose what permissions each should have based on their role in their companies. |

| **Process Description** |
| --- |
| Process Name: Select Archived Claim |
| Process Number: 11.1 |
| Short Description: This process allows the selection of an archived claim for potential restoration. |
| Process Description:  Receive request to restore an archived claim. Display list of archived claims available for selection. Allow users to select the specific archived claim for restoration. Send selected claim data to the Verify Access Permissions process. |
| Notes: The list of archived claims is pulled from the Archive Database. |

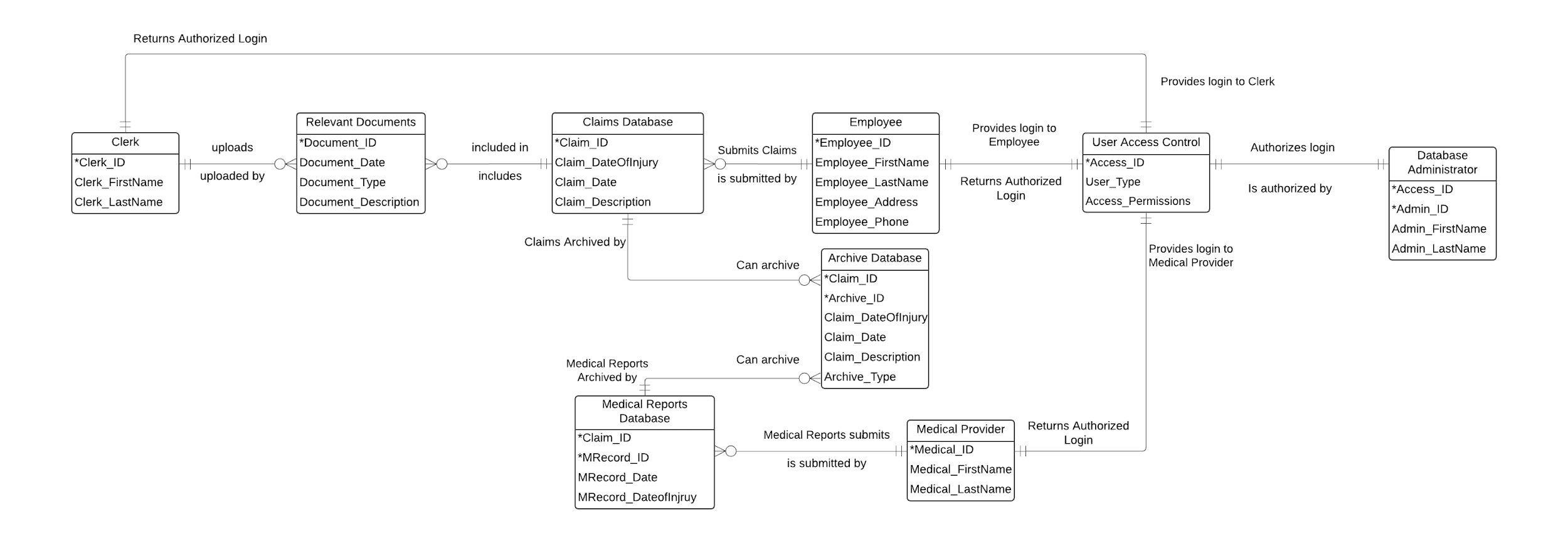
| **Process Description** |
| --- |
| Process Name: Verify Access Permissions |
| Process Number: 11.2 |
| Short Description: This process checks if the user has permission to restore the selected archived claim. |
| Process Description:  Receive selected claim data from the Select Archived Claim process.  Verify the user’s permissions to access and restore the claim.  If permissions are valid, forward to the Restore Claim process.  If permissions are not valid, send an error message back to the user. |
| Notes: User permissions are verified based on pre-defined system access levels. |

| **Process Description** |
| --- |
| Process Name: Restore Claim |
| Process Number: 11.3 |
| Short Description: This process retrieves and restores the selected archived claim from the archive database |
| Process Description:  Receive verified claim data from the Verify Access Permissions process.  Retrieve the archived claim information from the Archive Database.  Mark the claim as restored and send it to the Update Claims Database process.  Flag the original archive entry as deleted or moved for tracking purposes. |
| Notes: The retrieved claim is marked as active in the Claims Database. |

| **Process Description** |
| --- |
| Process Name: Delete the Archived Claim |
| Process Number: 11.4 |
| Short Description: This process deletes the archived claim from the Archive Database after it has been selected for restoration. |
| Process Description:  Receive the selected archived claim from the Restore Claim process.  Verify that the claim is marked for deletion after restoration.  Delete the archived claim from the Archive Database.  Send a confirmation of deletion to the Update Claims Database process. |
| Notes: This ensures that only restored claims are removed from the Archive Database to prevent duplication. |

| **Process Description** |
| --- |
| Process Name: Update Claims Database |
| Process Number: 11.5 |
| Short Description: This process updates the Claims Database with the restored claim and confirms its active status. |
| Process Description:  Receive restored claim data from the Restore Claim process.  Insert the restored claim into the Claims Database as an active record.  Mark the claim as "Restored" to distinguish it from newly submitted claims.  Send confirmation of successful update to the appropriate process. |
| Notes: The restored claim will be treated as an active claim, and the database will be updated accordingly. |

# XI. Data Model

**Entity Relationship Diagram**

**Data Dictionary**

| **Entity** |
| --- |
| Entity Name: Clerk |
| Entity Description: User who has authorized access to the claims system. Clerks are responsible for managing and searching claims, generating reports, and assisting with claim processing. |
| Attributes:  \*Clerk\_ID  Clerk\_FirstName  Clerk\_LastName |
| Notes: Clerks have varying permission levels depending on their role within the organization, which is managed through the User Access Control system. |

| **Attribute** |
| --- |
| Attribute Name: Clerk\_ID |
| Attribute Description: A unique identifier for each clerk in the system. |
| Values and Meanings: Numeric or alphanumeric, system-generated. |
| Notes: This attribute is used as the primary key for the Clerk entity. |

| **Attribute** |
| --- |
| Attribute Name: Clerk\_FirstName |
| Attribute Description: The first name of the clerk. |
| Values and Meanings: Text, includes first name. |
| Notes: Used for identification and display in reports and system logs. |

| **Attribute** |
| --- |
| Attribute Name: Clerk\_LastName |
| Attribute Description: The last name of the clerk. |
| Values and Meanings: Text, includes last name. |
| Notes: Used for identification and display in reports and system logs. |

| **Entity** |
| --- |
| Entity Name: Relevant Documents |
| Entity Description: The Relevant Documents entity stores all documents that are associated with the claims submitted by employees. These documents may include medical reports, accident reports, and any other documentation related to the employee’s claim. The documents are uploaded by clerks and are linked to specific claims in the Claims Database. |
| Attributes:  \*Document\_ID  Document\_Date  Document\_Type  Document\_Description |
| Notes: This entity helps keep track of all documents tied to a claim, ensuring that all required paperwork is properly linked, stored, and easily retrievable. |

| **Attribute** |
| --- |
| Attribute Name: Document\_ID |
| Attribute Description: A unique identifier for each document in the system. |
| Values and Meanings: Numeric or alphanumeric, system-generated. |
| Notes: This attribute is used as the primary key for the Relevant Document entity. |

| **Attribute** |
| --- |
| Attribute Name: Document\_Date |
| Attribute Description: The date when the document was uploaded to the system. |
| Values and Meanings: Date format (YYYY-MM-DD). |
| Notes: Used to track when documents are uploaded for archiving and reference purposes. |

| **Attribute** |
| --- |
| Attribute Name: Document\_Type |
| Attribute Description: The type of document, such as medical report, accident report, or other relevant files. |
| Values and Meanings: Possible values could include "Medical Report," "Accident Report," "Witness Statement," "Photos," etc. |
| Notes: This attribute helps categorize the documents for easier access and management. |

| **Attribute** |
| --- |
| Attribute Name: Document\_Description |
| Attribute Description: A brief description of the document’s contents. |
| Values and Meanings: Text-based description that outlines the key points of the document. |
| Notes: This field allows clerks and administrators to quickly identify the document's purpose without opening the file. |

| **Entity** |
| --- |
| Entity Name: Claims Database |
| Entity Description: It stores all injury claims submitted by employees. It includes details about each claim, such as the date of injury, the date the claim was filed, and a description of the claim. This database is essential for tracking and processing claims and is linked to other entities, such as Medical Reports and Archived Claims. |
| Attributes:  \*Claim\_ID  Claim\_DateOfInjury  Claim\_Date  Claim\_Description |
| Notes: Claims in this database are subject to archival after a specified period and are linked to related medical reports. |

| **Attribute** |
| --- |
| Attribute Name: Claim\_ID |
| Attribute Description: A unique identifier for each claim in the system. |
| Values and Meanings: Numeric or alphanumeric, system-generated. |
| Notes: This attribute serves as the primary key for the Claim Database entity and is used to link claims with medical reports and other related data. |

| **Attribute** |
| --- |
| Attribute Name: Claim\_DateOfInjury |
| Attribute Description: The date on which the injury occurred. |
| Values and Meanings: Date format (YYYY-MM-DD). |
| Notes: Used for claim tracking and as a key piece of information for reports and archival processes. |

| **Attribute** |
| --- |
| Attribute Name: Claim\_Date |
| Attribute Description: The date on which the claim was filed. |
| Values and Meanings: Date format (YYYY-MM-DD). |
| Notes: Used to track the timeline of the claim process; important for legal and administrative purposes. |

| **Attribute** |
| --- |
| Attribute Name: Claim\_Description |
| Attribute Description: A brief description of the claim, detailing the circumstances and nature of the injury. |
| Values and Meanings: Text, includes relevant details about the injury. |
| Notes: Used by clerks and medical providers to understand the context of the claim; this field may be referenced when generating claim reports. |

| **Entity** |
| --- |
| Entity Name: Employee |
| Entity Description: The Employee entity represents individuals who submit claims within the claims management system. This entity captures key details about each employee involved in the claims submission process. |
| Attributes:  \*Employee\_ID  Employee\_FirstName  Employee\_LastName  Employee\_Address  Employee\_Phone |
| Notes: This entity is linked to the Claims Database entity to show who submits each claim. |

| **Attribute** |
| --- |
| Attribute Name: Employee\_ID |
| Attribute Description: Unique identifier assigned to each employee |
| Values and Meanings: Numeric or alphanumeric code that distinguishes each employee within the system. |
| Notes: Primary key for the Employee entity, used to link employees with the claims they submit. |

| **Attribute** |
| --- |
| Attribute Name: Employee\_FirstName |
| Attribute Description: First name of the employee |
| Values and Meanings: Text field containing the employee’s first name. |
| Notes: Used to identify employees by name within the system. |

| **Attribute** |
| --- |
| Attribute Name: Employee\_LastName |
| Attribute Description: Last name of the employee |
| Values and Meanings: Text field containing the employee’s last name. |
| Notes: Used to identify employees by name within the system. |

| **Attribute** |
| --- |
| Attribute Name: Employee\_Address |
| Attribute Description: Residential address of the employee. |
| Values and Meanings: Text field formatted to include street address, city, state, and postal code (STREET; CITY, STATE ABBREVIATION, POSTAL CODE). |
| Notes: Optional field; relevant for record-keeping or correspondence. |

| **Attribute** |
| --- |
| Attribute Name: Employee\_Phone |
| Attribute Description: Personal phone number of the employee. |
| Values and Meanings: Text field formatted to include the phone number. ((XXX) XXX - XXXX) |
| Notes: Optional field; relevant for record-keeping or correspondence. |

| **Entity** |
| --- |
| Entity Name: Archive Database |
| Entity Description: The Archive Database entity stores archived claims within the system. It is used for managing the archiving process and tracking the status and type of each archived claim. |
| Attributes:  \*Archive\_ID  \*Claim\_ID  Claim\_DateofInjury  Claim\_Date  Claim\_Description  Archive\_Type |
| Notes: This entity is related to both the Claims Database and Medical Reports Database to manage archived claims and medical records effectively. |

| **Attribute** |
| --- |
| Attribute Name: Archive\_ID |
| Attribute Description: Unique identifier for each archived record |
| Values and Meanings: Numeric or alphanumeric code that uniquely identifies each entry in the Archive Database. |
| Notes: Primary key for the Archive Database entity. |

| **Attribute** |
| --- |
| Attribute Name: Claim\_ID |
| Attribute Description: Unique identifier that links each archived record to a specific claim. |
| Values and Meanings: Numeric or alphanumeric code corresponding to a claim in the Claims Database. |
| Notes: Acts as a foreign key, allowing reference to claims from the Claims Database. |

| **Attribute** |
| --- |
| Attribute Name: Claim\_DateOfInjury |
| Attribute Description: The date on which the injury occurred. |
| Values and Meanings: Date format (YYYY-MM-DD). |
| Notes: Foreign key from the claims database, used for claim tracking and as a key piece of information for reports and archival processes. |

| **Attribute** |
| --- |
| Attribute Name: Claim\_Date |
| Attribute Description: The date on which the claim was filed. |
| Values and Meanings: Date format (YYYY-MM-DD). |
| Notes: Foreign key from the claims database, used to track the timeline of the claim process; important for legal and administrative purposes. |

| **Attribute** |
| --- |
| Attribute Name: Claim\_Description |
| Attribute Description: A brief description of the claim, detailing the circumstances and nature of the injury. |
| Values and Meanings: Text, includes relevant details about the injury. |
| Notes: Foreign key from the claims database, used by clerks and medical providers to understand the context of the claim; this field may be referenced when generating claim reports. |

| **Attribute** |
| --- |
| Attribute Name: Archive\_Type |
| Attribute Description: Specifies the category or nature of the archived claim |
| Values and Meanings: Predefined text values such as "Closed Claim", "Expired", "Settled", etc. |
| Notes: Helps in categorizing archived claims for easier retrieval and management. |

| **Entity** |
| --- |
| Entity Name: User Access Control |
| Entity Description: This entity manages access permissions based on user types. |
| Attributes:  \*Access\_ID  User\_Type  Access\_Permissions |
| Notes: This entity is critical for managing and authorizing different levels of access to the system. |

## 

| **Attribute** |
| --- |
| Attribute Name: Access\_ID |
| Attribute Description: Unique identifier for each access control entry. |
| Values and Meanings: Numeric or alphanumeric values unique to each record. |
| Notes: Primary key of the entity. |

| **Attribute** |
| --- |
| Attribute Name: User\_Type |
| Attribute Description: Designates the type of user associated with the access permissions. |
| Values and Meanings: Clerk, Employee, Medical Provider, Administrator. |
| Notes: Defines roles and determines access permissions. |

| **Attribute** |
| --- |
| Attribute Name: Access\_Permissions |
| Attribute Description: Specifies permissions for accessing different parts of the database. |
| Values and Meanings: Permissions can vary, such as 'Read', 'Write', 'Update', 'Archive'. |
| Notes: Permissions are configured based on the User\_Type. |

| **Entity** |
| --- |
| Entity Name: Database Administrator |
| Entity Description: Manages the overall database system and authorizes logins. |
| Attributes:  \*Access\_ID  \*Admin\_ID  Admin\_FirstName  Admin\_LastName |
| Notes: Holds the highest level of access and authorization responsibilities within the database. |

## 

| **Attribute** |
| --- |
| Attribute Name: Access\_ID |
| Attribute Description: Linked identifier for access permissions from User Access Control. |
| Values and Meanings: Matches Access\_ID in User Access Control for consistency. |
| Notes: Ensures that administrators have the appropriate permissions. |

| **Attribute** |
| --- |
| Attribute Name: Admin\_ID |
| Attribute Description: Unique identifier for each administrator. |
| Values and Meanings: Numeric or alphanumeric values, unique per administrator. |
| Notes: Primary key for the Database Administrator entity. |

| **Attribute** |
| --- |
| Attribute Name: Admin\_FirstName |
| Attribute Description: The first name of the database administrator. |
| Values and Meanings: Text field for administrator first names. |
| Notes: Identifies the individual responsible for database administration. |

| **Attribute** |
| --- |
| Attribute Name: Admin\_LastName |
| Attribute Description: The last name of the database administrator. |
| Values and Meanings: Text field for administrator last names. |
| Notes: Identifies the individual responsible for database administration. |

| **Entity** |
| --- |
| Entity Name: Medical Reports Database |
| Entity Description: This entity stores the medical reports filed. This is where the medical reports are stored and retrieved from. |
| Attributes:  \*Claim\_ID  \*MRecord\_ID  MRecord\_Date  MRecord\_DateofInjury |
| Notes: Claim\_ID is a foreign key from the claims database, and MRecord\_ID is a primary key. |

| **Attribute** |
| --- |
| Attribute Name: Claim\_ID |
| Attribute Description: Identifier that links a claim to a specific ID for ease of retrieval and documentation. It can be combined with medical records so certain records for a claim can be linked together. |
| Values and Meanings: Numeric or alphanumeric code corresponding to a claim in the Claims Database. |
| Notes: This is a foreign key for the Medical Reports Database from the claims database. |

| **Attribute** |
| --- |
| Attribute Name:MRecord\_ID |
| Attribute Description: Identifier of a medical record that gives a specific ID to a record for storage and retrieval. |
| Values and Meanings: Numeric or alphanumeric code corresponding to a medical record in the Medical Reports Database. |
| Notes: MRecord\_ID is a primary key of the Medical Reports Database entity. |

| **Attribute** |
| --- |
| Attribute Name: MRecord\_Date |
| Attribute Description: This attribute records the date that the medical report was filed on. |
| Values and Meanings: A date type value (YYYY-MM-DD). |
| Notes: This can make searching for medical reports easier if only the date is known. |

| **Attribute** |
| --- |
| Attribute Name: MRecord\_DateofInjury |
| Attribute Description: Stores the date of the actual injury that is in the medical report. |
| Values and Meanings: A date type value. (YYYY-MM-DD). Separate from MRecord\_Date, since that is the date of the filing, not the date of the actual injury. |
| Notes: This can help a clerk search for a claim or medical report if only the date of the injury is known. |

| **Entity** |
| --- |
| Entity Name: Medical Provider |
| Entity Description: This is the health professional who is responsible for uploading medical reports to the medical report database. |
| Attributes:  \*Medical\_ID  Medical\_FirstName  Medical\_LastName |
| Notes: The primary key of this entity is Medical\_ID. |

| **Attribute** |
| --- |
| Attribute Name: Medical\_ID |
| Attribute Description: Stores a unique medical ID given to each health professional to document who completed filing of medical reports. |
| Values and Meanings: Numeric or alphanumeric code corresponding to a medical provider in the Medical Provider entity. |
| Notes: This is the primary key of the Medical Provider entity. |

| **Attribute** |
| --- |
| Attribute Name: Medical\_FirstName |
| Attribute Description: This is the first name of the medical provider who completed the filing of medical records. |
| Values and Meanings: String format of first name. |
| Notes: Medical\_FirstName can be found from the Medical\_ID, since they should both be linked to the same medical provider. |

| **Attribute** |
| --- |
| Attribute Name: Medical\_LastName |
| Attribute Description: This is the last name of the medical provider who completed the filing of medical records. |
| Values and Meanings: String format of first name. |
| Notes: Medical\_LastName can be found from the Medical\_ID, since they should both be linked to the same medical provider. |

| **Relationships** |
| --- |
| Relationship Name: Uploads |
| Relationship Description: A Clerk uploads zero or many Relevant Documents. Each Relevant Document is uploaded by one and only one Clerk. |
| Attached Entities:  Clerk  Uploaded by - Min: One, Max: One  Relevant Documents  Uploads - Min: 0, Max: Many |
| Notes: Clerks may upload multiple documents, but each document is associated with only one Clerk. |

| **Relationships** |
| --- |
| Relationship Name: Included in |
| Relationship Description: A Claim includes zero or many Relevant Documents. Each Relevant Document is included in one and only one Claim Database. |
| Attached Entities:  Relevant Documents  Included in - Min: 0, Max: Many  Claims Database  Included - Min: One, Max: One |
| Notes: A claim may have no attached documents or multiple documents related to it. |

| **Relationships** |
| --- |
| Relationship Name: Submits Claims |
| Relationship Description: An employee submits zero or many claims to the Claims Database, Claims are submitted by one and only one employee |
| Attached Entities:  Employee  Submits - Min: 0, Max: Many  Claims Database  Is submitted by - Min: One, Max: One |
| Notes: An employee does not need to submit any claims to the claims database, but they can submit many if needed. |

| **Relationships** |
| --- |
| Relationship Name: Claims Archived by |
| Relationship Description: The claims database can archive zero or many claims, and the archive database’s claims are archived by one and only one claims database. |
| Attached Entities:  Claims Database  Can archive - Min: 0, Max: Many  Archive Database  Archived by - Min: One; Max: One |
| Notes: There are other processes required to retrieve claims from the archive database. |

| **Relationships** |
| --- |
| Relationship Name: Medical Reports Archived by |
| Relationship Description: The medical reports database can archive zero or many claims, and the archive database’s medical reports are archived by one and only one medical reports database. |
| Attached Entities:  Medical Reports Database  Can archive - Min: 0, Max: Many  Archive Database  Archived by - Min: One; Max: One |
| Notes: There are other processes required to retrieve reports from the archive database. |

| **Relationships** |
| --- |
| Relationship Name: Submits Medical Reports |
| Relationship Description: A medical provider submits zero or many medical reports to the Medical Reports Database, Medical Reports from the Medical Reports Database are submitted by one and only one medical provider |
| Attached Entities:  Medical Reports Database  Submits - Min: 0, Max: Many  Medical Provider  Is submitted by - Min: One, Max: One |
| Notes: A medical provider does not need to submit any medical reports to the medical reports database, but they can submit many if needed. |

| **Relationships** |
| --- |
| Relationship Name: Provides login to Employee |
| Relationship Description: One and only one user access control provides login to one and only one employee and one and only one employee returns authorized login to one and only one user access control |
| Attached Entities:  Employee  Provides login - Min: One, Max: One  User Access Control  Returns authorized login - Min: One, Max: One |
| Notes: This relationship is important for security and access management, ensuring that each Employee has a unique, traceable login credential and that access is controlled at the individual level. |

| **Relationships** |
| --- |
| Relationship Name: Provides login to Medical Provider |
| Relationship Description: One and only one user access control provides login to one and only one medical provider and one and only one medical provider returns authorized login to one and only one user access control |
| Attached Entities:  Medical Provider  Provides login - Min: One, Max: One  User Access Control  Returns authorized login - Min: One, Max: One |
| Notes: This relationship is important for security and access management, ensuring that each Medical Provider has a unique, traceable login credential and that access is controlled at the individual level. |

| **Relationships** |
| --- |
| Relationship Name: Provides login to Clerk |
| Relationship Description: One and only one user access control provides login to one and only one clerk and one and only one clerk returns authorized login to one and only one user access control |
| Attached Entities:  Clerk  Provides login - Min: One, Max: One  User Access Control  Returns authorized login - Min: One, Max: One |
| Notes: This relationship is important for security and access management, ensuring that each Clerk has a unique, traceable login credential and that access is controlled at the individual level. |

| **Relationships** |
| --- |
| Relationship Name: Authorizes login |
| Relationship Description: One and only one User Access Control authorizes login to one and only one database administrator, while each user access control must be authorized by one and only one Database Administrator. |
| Attached Entities:  User Access Control  Authorizes login - Min:One, Max: One  Database Administrator  Is authorized by - Min: One, Max: One |
| Notes: This setup is critical for security, as it enables centralized control over who can access the system, with clear responsibility assigned to individual Database Administrators. |

# XII. Conclusion

The benefits of developing the online database system for the South Dakota Department of Labor’s Workers’ Compensation Division greatly outweigh the costs. The savings from reduced labor costs and paper storage will be approximately $1,275,000 over the first five years of implementation. There will be a decrease of human errors and increased storage that may hold more employees. Also, the physical storage system will be eliminated.

# In conclusion, while the initial development costs are substantial, the long-term benefits, including cost savings, improved operational efficiency, and enhanced customer satisfaction, make this system a sound investment.

# References

AltexSoft. (2023, November 30). *Functional and nonfunctional requirements specification*. <https://www.altexsoft.com/blog/functional-and-non-functional-requirements-specification-and-types/>

Dennis, A., Wixom, B. H., & Roth, R. M. (2022). *Systems analysis and design*. John Wiley & Sons, Inc.

Kanjilal, J. (2024, March 8). *Overview of the JAD methodology*. Developer.com. <https://www.developer.com/project-management/jad-methodology/#:~:text=JAD%20sessions%20should%20have%20a,expect%20quick%20resolutions%20from%20them>.

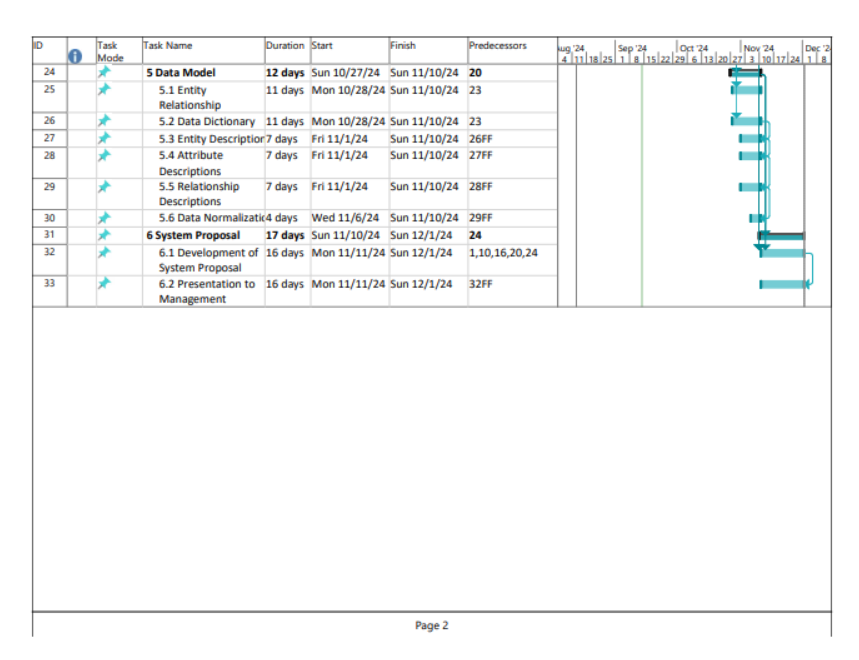
Lucidchart Team. (n.d.). *What is a data flow diagram*. Lucidchart.<https://www.lucidchart.com/pages/data-flow-diagram>

Singal, S. K., Goel, V., Nautiyal, H., & Papantonis, D. E. (2023). *Economic feasibility*. Economic Feasibility - an overview | ScienceDirect Topics. <https://www.sciencedirect.com/topics/engineering/economic-feasibility>

Watson, R. (2013). Information Systems. Retrieved from <http://solr.bccampus.ca:8001/bcc/file/5f0cf1f4-2c31-4424-804e-e8d57075a5fe/1/Information%20Systems.pdf>

# 

# Appendix A - Gantt



# Appendix B - PERT Chart

