Project Data Tracking and Visualization System and Interface

For South Fork Design Group

Chris Young
Chris McCleod
Dallin Fairbanks
Bonnie McDougal

Table of Contents

Executive Summary

South Fork Design Group is a structural design and engineering firm in Rexburg, Idaho that services clients and produces drafts and designs for new homes, commercial buildings, garages, sheds, and other projects. Ray McDougal, the owner and manager of South Fork, has seen the need for storing past project data because understanding their project data will lead to increased profitability and efficiency for future projects at South Fork.

In this report, we analyze the current systems at South Fork Design and recommend a solution that can increase profitability, efficiency, and the business process at South Fork.

Findings and Recommendations

Throughout our analysis, we worked with Ray to identify current functions in the South Fork Design system that can be improved. These are detailed further in the report. Our primary recommendations include:

- Implementation of a centralized database to hold and track project data, employee information, and client contact information
- A project-entry system that facilitates the collection and entering of data, as well as creating and updating employee and client information
- A new system to track and maintain employee labor hours while discontinuing the use of separate applications

Our analysis of the current system of South Fork Design includes diagrams, descriptions, and models below to help understand the main areas for improvement in the system. Additionally, our proposed

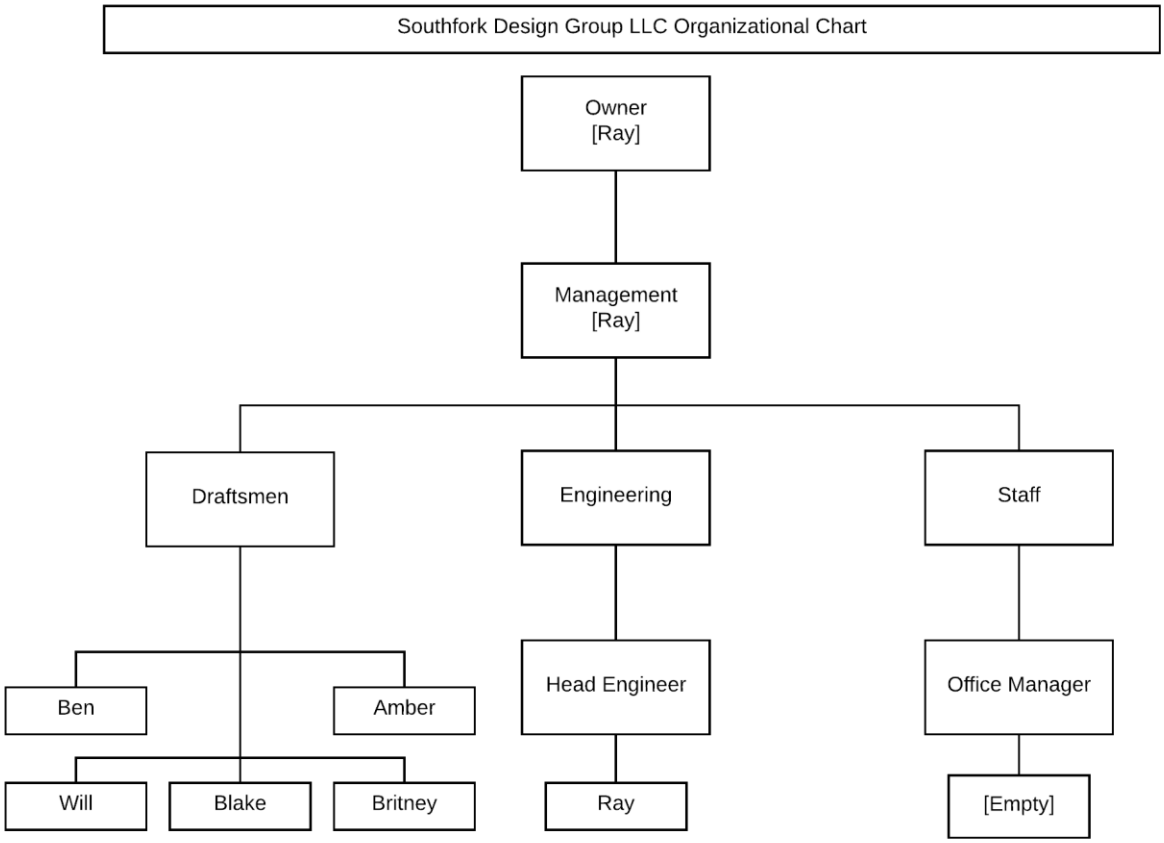
solution, description, and implementation plan are detailed further below. We recommend South Fork Design implements these solutions to increase profitability, efficiency, and growth in the future.

Business Background

After graduating from Utah State University with a degree in environmental engineering, Ray McDougal worked for several years at the Idaho National Laboratory (INL). However, after a shift in career focus and a move to Rexburg, Idaho, Ray founded a structural design and engineering business named South Fork Design Group (South Fork).

Ray started his business with one employee: himself. As his clientele grew, he hired other people as draftsman and eventually stepped away from the design responsibilities to focus on the engineering and business management of South Fork. Today, he works as the head engineer and manager of South Fork. He has five draftsmen to design the plans for the project sand work with clients: Will, Blake, Amber, Brittany, and Ben. He also normally has one office manager, who helps schedule meetings, keep track of projects and clients, and write up the payroll report. Currently, he is in between office managers and is seeking another manager. The current business organizational structure is pictured below.

South Fork has upgraded its design software and provides clients with 3-D modelling and demonstrations of their future homes or investments. As a business, South Fork takes pride in having a good construction background, which makes its deliverables sound and easily transferable to external contractors for construction.



Business Case

The business case portion of this report includes a description of South Fork's problem, the scope of the solution, and a feasibility analysis for the solution.

Business Problem

The greatest issue faced by South Fork is that Ray has no effective means of storing and utilizing data. Due to this data deficit, Ray is unable to set bid prices as accurately as he would like on new projects, track what kinds of projects are most financially advantageous, or effectively track and reward employee performance. With a well-designed database, Ray will be able to use data to meet the current needs of the company.

Scope

The scope of this project is to create a new system that will allow South Fork Design to effectively collect and analyze data to more efficiently and effectively run business operations. This includes five main deliverables with associated documentation and implementation plans. First, a new database design will be created to store the data that the company wants to track. There is no existing database so a new database will be designed from scratch and an Entity Relationship Diagram will be created as the first deliverable to show how the data will be stored.

The second and third deliverables will be a class and sequence diagram. These diagrams will illustrate how the new system will operate in order to work with the newly-designed database. The class diagram will be used to describe how the code will be created and kick-start the future creation of the actual program. The sequence diagram will be created to show how the new classes will interact to display and analyze what will be collected. This program will allow the company to track data about projects, employees, and the work process, and then organize and display the requested information. After these changes, an updated data flow diagram will be made to reflect the changes that the new system will introduce with the flow of data and how that data will now be tracked and accessed.

The fifth and final deliverable will be user interface mockups. These mockups will visually describe how the system might look and feel. Included in these mockups will be screens that demonstrate how data will be added, viewed, updated, and deleted from the system. Mockups will give a visual representation of how the system will work for the owner, the draftsmen and the office manager. All parts of this project will contribute to the overall goal of allowing Southfork Design to be able to use data in a way that will benefit the company in many ways.

Feasibility Analysis

Based off the current system and the future system requirements, we consider this project to be highly feasible. We evaluated this according to

- economic feasibility,
- technical feasibility, and
- organizational feasibility

The following table illustrates the feasibility level we determined with each of the areas

Feasibility Area	Feasibility Level
Economic	High
Technical	High
Organizational	High

We came to these conclusions based off of the risks involved in each area and the resources and methods available to South Fork to manage risk.

Economic Feasibility

We broke our project up into the following different major components:

- Database build
- System build
- User interface build
- Testing and deployment

We evaluated our costs assuming each person could work two hours per day on this project at a rate of $40/hour. The costs for each component are as follows:

Component	Cost
Database build	80
System build	640
User interface build	480
Testing and deployment	1,040
Total cost	$2,240

The total cost of the project would come out to be $2240. Considering South Fork's net income of $115,000 in 2018, we consider this to be highly feasible.

Additionally, we recognize that by using this new system to create more accurate budgeting and cost drivers, South Fork would save a significant amount of funds. From January to October 2019, South Fork awarded **$4581.04** in bonuses to its draftsman for completing projects under the estimated hours. With a more accurate budgeting system, South Fork could anticipate a decrease in this number.

Through this new system, South Fork will also be able to recognize more profitable projects and clients and better market to them. This will help increase revenue.

However, we are unsure about the future costs that South Fork would experience over time through upkeep of the system. However, we anticipate the system requiring little upkeep unless Ray decides to start tracking new cost drivers.

Overall, we are looking at an initial investment of $2240, with potential average monthly savings of up to $381.75, as well as incalculable increased revenue and savings for South Fork. The upkeep costs would also be minimal.

Due to this, we consider the overall economic feasibility to be low.

Technical Feasibility

Based off the current skills of the team in MVC, databases, and database design, we consider this to be a technically feasible system. The technical risks involved with this project involve integration with South Fork's current website, security of the system, and long-term durability of the system.

Risks Involved

The technical risks involved with this project involve

- Failed integration with South Fork's current website
- Low security of the system
- Lack of long-term functionality of the system

We evaluate these risks using the table below.

	Impact	Likelihood	Lack of Forewarning	Overall
Failed website integration	H	L	L	L
Insecure system	M	M	L	M

Lack of long-term functionality	M	L	H	M

Currently, our team does not have experience integrating an MVC program to another website. This would impact the company greatly if South Fork couldn't access its system for project and hours tracking. However, through research and consultation with peers and professors, the team could learn this and implement a solid integration, so the likelihood of this happening is low. Overall, we consider this to be a low risk.

Additionally, our team does not have much experience or background with security in a system. The risk of the system being insecure and therefore prone to hackers is at a medium level, and considering the team's experience with security, the likelihood is at a medium level as well. As a team, we will use HTML helpers where possible to boost security within the system, but this lack of experience does present a medium overall risk.

We are working to design and build a system that can be used for years to come. However, if Ray decides he wants to track new potential cost drivers or make other changes to the system, he would be unable to do that himself and would need to hire someone to make these changes. This would have a medium level impact on the company, because while South Fork wouldn't be able to measure new drivers, it would still have a significant amount of data to work with for the tracking. The likelihood of this happening is low, but we can't anticipate these changes either.

Overall, we consider the economic risk to be at a low level, and the feasibility to be high.

Risk Management

To prevent these risks, the team will be deliberate in its research and learning of website integration. We will take the time to discuss with professors and peers to ensure good understanding.

As a preventative action to needed changes within the current system for data, we have asked Ray as well as his draftsman the things they perceive to be cost drivers as well as things that Ray would like to look into in the future that he could perceive having an impact. We anticipate these potential cost drivers being beneficial towards Ray's business decisions for years to come.

Organizational Feasibility

This project would be a great benefit to South Fork as an organization. Ray has struggled with his inaccurate hour budgeting system for years now but has been unable to develop an accurate change himself. He has high hopes for a system to track this data and help with his overall management of South Fork. However, his draftsmen's use of the system and Ray's understanding of the system upon implementation present risks.

Risks Involved

Overall, we consider the risks involved with the organizational feasibility to be low. These risks include

- Inaccurate data inputted by draftsmen for a short time
- Inaccurate data inputted by draftsmen for an extended period of time
- Improper use of data by Ray

The table below assesses each of these risks according to impact, likelihood, lack of forewarning (how much warning we would or would not have of this occurring), and overall risk.

	Impact	Likelihood	Lack of Forewarning	Overall
Short-term inaccurate data	M	H	L	L
Long-term inaccurate data	H	L	L	L
Improper use of Data	H	L	M	L

As the new system is implemented, we would expect that at first the draftsmen would make mistakes when inputting data, but hope that the draftsmen would adjust quickly, resulting in accurate data inputted consistently within a minimal amount of time. The inaccurate data in the short term would have a medium-level impact on the company, since with time, the data would become more and more accurate. However, the likelihood of this happening is high.

Long-term inaccurate data input would have a high negative impact on the company. We anticipate something like this occurring due to employee apathy and attitude. The likelihood of this is low considering the draftsmen's attention to detail and consistency already practiced in the current system.

Another possible risk involves Ray's use of the data. If Ray doesn't know how to use the data he gathers to adjust cost drivers and make business decisions, then the system won't have helped Ray in his current situation, resulting in a high negative impact. However, we consider the likelihood of this

to be low, as Ray invests a lot of time into understanding his business and is willing to adjust business strategies according to business needs.

As demonstrated by the matrix, we consider all of these risks to be low for South Fork. However, South Fork could take preventive measures to manage these risks.

Risk Management

To prevent the risk of short-term inaccurate data, we recommend that South Fork takes ample time to train draftsmen on the system. We also recommend sending out reminders and notifications to draftsmen to remind them to input data as needed.

Long-term inaccurate data is most likely to occur due to a poor attitude around the new system. South Fork should strive to create a good company culture around change to help prevent this risk.

Ray should be sure to consult with business professionals to learn how best to utilize the data he gathers from the new system. This will help ensure that the data is used to help create more accuracy in budgeting and bidding.

Current System Overview

As part of the overview of the current systems, we cover the

- Current system project lifecycle
- Current system software and documents
- Current system diagrams

Current System Project Lifecycle

When a client first contacts South Fork with a project, the office manager schedules a first meeting between the client and one of the draftsmen. They meet, and after receiving the specifications for the structure, the draftsman will give a bid to the client, which includes a dollar amount and an effort estimate (in hours) for the project.

This bid can be given at the initial client meeting, which Ray reports increases the chance of a client committing to the project. However, if a draftsman is unsure of a bid amount, he or she will close the meeting and meet with Ray to evaluate a bid price. After meeting with Ray, the draftsman will send the client an email with the bid details.

If the client accepts the project, he or she pays a deposit with either cash, card, or check. The client signs a contract through a software called Docuign, and the office manager inputs the project details into an Excel project tracking spreadsheet. South Fork payments are managed through a software called Square.

Once this data is inputted, the draftsman begins the design of the project using the structural design software. The draftsman keeps in contact with the client throughout this project and will meet with the client throughout the design process to ensure that the structure is up-to-expectations. The client will revise as needed.

After completing the draft of the project, the draftsman prints out the plans and gives them to Ray, who then ensures that the project will be structurally sound. If revisions are needed, Ray will mark

these revisions in red on the plans and send them back to the draftsman for revision. If the plans are up-to-specifications, then the office manager will contact the client with a bill for the final payment.

After the client makes the final payment, the plans are printed and sent to the client. The project is then removed from the project tracking spreadsheet.

Under South Fork's current business process, clients are charged and can pay through a system called square. Employees keep track of their hours on each project through an application called Toggl. Currently, South Fork does not have a system for tracking historical project data.

Project bids and effort estimates are made based on the square footage of the structure. However, Ray says that this current system is inaccurate, and that other factors besides square footage drive the hours spent on the project.

If an employee finishes the project under the budgeted hours, he or she receives a bonus. However, there is no penalty if the project goes beyond the budgeted hours.

Current System Software and Documents

The current system of South Fork implements the following:

- Apps, such as Toggl and Square
- Microsoft Excel spreadsheets
- DocuSign

These features will be discussed more in detail below.

Toggl: Toggl is a time tracking application that offers online time tracking and reporting services. Toggl tracks time based on assigned tasks and projects through either an interactive task timer or manual entry. South Fork Design primarily uses Toggl to log employee hours on current assigned tasks.

Square: Square is a free point-of-sale application that provides all the necessary functions of running a business. Square provides an inventory management system, payment methods, secure transactions, invoicing, and payment processing. South Fork utilizes Square for transactions data and secure payments.

Microsoft Excel: Excel is a spreadsheet application that can hold and analyze data. Currently, South Fork tracks projects and the project queue within Excel. The office manager adds and updates projects as more information is given or as projects are completed.

DocuSign: DocuSign is a web-based software that allows parties or organizations to automate how agreements are prepared, delivered, and signed. South Fork uses DocuSign to send contracts to clients and make agreements for project deliverables.

South Fork also has a current website that displays past projects, available services, a question and answer page, and other items.[1]

Current System Diagrams

After interviewing Ray McDougal, we created the following diagrams and models to visualize the current system of South Fork Design:

1. Activity Diagram, modeling the interaction between engineers, draftsmen, and clients at South Fork Design.
2. Context Level Data Flow Diagram (DFD), showing the main data components between entities at South Fork.
3. Level-0 DFD, which details the flow of data and current data stores (like Toggl and Square)
4. Level-1 DFD(s), which further outline the process of assigning draftsmen to projects and collecting payment information

[1] (South Fork Design Group 2015)

These diagrams are included and described below.

Activity Diagram

The activity diagram depicts the project lifecycle as described previously. We included four separate swim lanes in this diagram to model the transfer of roles throughout the life of the project. The diagram can be found in Appendix A.

Context Level Data Flow Diagram

The data flow diagrams for the current system are used to depict how plans, financial data, and employee hours are transferred and recorded throughout the business. These diagrams have three external entities: client, engineer, and draftsman. To help with readability, we split the client into two separate entities, but they represent the same client.

See the context level diagram in Appendix A.

Level-0 Data Flow Diagram

The level-0 diagram includes the following processes:

1.0 Receive Project Requirements

2.0 Determine Bid Price

3.0 Record Project Information

4.0 Gather Financial Data

5.0 Select Project

6.0 Validate Plans

7.0 Record Hours

8.0 Revise and Engineer Draft

9.0 Send Final Plans

10.0 Prompt Final Payment

11.0 Determine Employee Bonus

The level-0 diagram also presents several data stores:

- Square
- Excel Project Lineup
- Toggl

We recommend using this diagram along with the activity diagram to better understand the chronological process that goes with the data flow.

See the level-0 diagram in Appendix A.

Level-1 Data Flow Diagrams

The level-1 diagrams further depict the following systems

4.0 Gather Financial Data

5.0 Select Project

7.0 Record Hours

11.0 Determine Employee Bonus

See these diagrams in Appendix A.

Current System Analysis

While the current system at South Fork Design has been in use for several years, we identified the following features of the current system that are certain to result in long-term financial fees, inefficiency, and employee dissatisfaction:

1. Use of Toggl app
2. Excel Project Sheet
3. Inconsistent data storage
4. Data tracking
5. Employee bonus calculation

These problems will be discussed in further detail below.

Use of Toggl App

As previously discussed and modeled in the Level-0 diagram provided in Appendix A, the Toggl app allows employees to record labor hours on assigned projects. South Fork currently has 6 draftsman who use the Toggl app to track their hours.

Screenshots of the Toggl interface can be found in Appendix A.

Up to this point, South Fork has had no previous problems while using Toggl. However, Ray intends to expand South Fork Design to include additional engineers and draftsman.

Toggl allows for 5 users on its free plan. Each additional user would incur membership costs, which Ray wishes to avoid.

Toggl Pricing

Toggl offers a free plan and three paid plans — Starter, Premium, and Enterprise. Paid plans range from $10 to $20+ per user per month. A discount of $1 to $2 (for Starter and Premium, respectively) per user per month is given to users who subscribe yearly. Priority support is available with their Enterprise plan.

Plans	Monthly Pricing ($/User/Month)	Annual Pricing ($/User/Year)	Team Members
Free	$0	$0	5
Starter	$10	$108	Unlimited
Premium	$20	$216	Unlimited
Enterprise	Customized	Customized	Unlimited

As long as South Fork Design continues to use the services of Toggl, they are unable to expand in number unless a pricing plan is selected, or a new service app is found to use. Ray is reluctant to select a pricing plan for a service that usually has no charge. Unfortunately, other top-rated time tracking applications like Harvest, Freshbooks, actiTIME, and TMetric also have pricing plans for either a similar or greater amount than Toggl.[2]

Thus, Ray is currently unable to expand South Fork Design because the current system constrains him to a maximum amount of employees without purchasing additional use of time-tracking software.

Excel Project Sheet

Another limiting feature of the South Fork Design system is the use of Microsoft Excel as a database. Currently, all the projects are tracked in an Excel file. This means project details like square footage, timeline, location, bid price, assigned employees, etc. Again, while South Fork has not yet

[2] (Shelton 2019)

encountered any major problems while using their Excel file to track projects, Excel is not the ideal option for tracking projects.

A major portion of the Office Manager's (OM) time is spent maintaining the Excel spreadsheet. Manual entry, updating, saving, and fixing errors could all be eliminated from OM's job if South Fork instead implements a relational database to manage their project tracking. The OM could instead manage the database and ensure minimal errors in data entry and increase data integrity. This would increase employee satisfaction in the long-term.

Another disadvantage of using an Excel spreadsheet is that it is generally not portable. The file is stored on the OM computer and if travel takes Ray or other employees elsewhere in Idaho, they are unable to view the current projects and any new projects that have been added to the system without first uploading the entire spreadsheet to either a memory drive or some type of third-party cloud storage site.

A screenshot of the excel spreadsheet can be found in Appendix A.

Inconsistent Data Storage

Another disadvantage and potential problem in the current South Fork Design system is the lack of a centralized location for data storage. Currently, multiple platforms are in use to store data. Toggl tracks employee labor hours, while Square houses financial transactions and data, and the Excel spreadsheet contains all the project tracking data.

While the use of multiple platforms has not yet impacted South Fork Design, integrating a centralized database for tracking both projects and employee hours will increase data integrity at South Fork. Currently, it is both time-consuming and inefficient for Ray to analyze data for projects because he has to first locate the project on the Excel spreadsheet, then go to Toggl to find the recorded hours for that specific project, then enter Square to find the related pricing and payments sent by the client. South Fork Design will continue to use Square to manage all client financial data, however, integrating both the Excel spreadsheet for projects and Toggl into a centralized database will enable Ray to better find needed data without constant back and forth between applications.

Data Tracking

Similarly, because South Fork Design lacks a centralized data storage system, Ray is unable to identify and track specific cost-drivers in project lifecycle. Ray wants to be able to track project data in order to better identify cost drivers. With the appropriate cost drivers, Ray can allocate resources and better price projects that include different features like porches, garages, etc. Without the necessary data, Ray is unable to change his pricing scheme, although he suspects different drivers than the ones currently used.

While the Excel spreadsheet is flexible and easy to use and access, any user can enter any type of data within it. Thus, if the OM changes or other people enter data, there is no guarantee that data will be consistent or if it will even be formatted the same. Thus, Ray is unable to analyze and track project data.

If South Fork implements a relational database to store project data, Ray will be better able to track his data and identify relevant cost drivers. Understanding relevant cost drivers will ultimately increase profitability and efficiency for South Fork Design.

Employee Bonus Calculation

Another disadvantage to the current data storage system is the employee bonus calculation. Currently, bonuses are awarded on the basis of finishing a project below budgeted hours. However, Ray understands that this is not the best way to award employee bonuses. As previously mentioned, more than $4,000 have already been awarded in employee bonuses for this year alone. Ideally, Ray would like to award a yearly bonus based on overall employee profitability, because current methods to calculate budgeted hours have been largely inaccurate.

With the current system, Ray is unable to switch to this method of bonuses because of the current data storage system in place. Ray cannot track employee profitability because it would require hours of sifting through Toggl data trying to match up employee labor hours to completed projects. Also, as previously mentioned, some projects have different cost drivers that have yet to be identified until a better data storage system is implemented. As such, some projects will, by necessity, take longer than budgeted hours suggest because of special features that the client requests. Thus, not all projects have fairly-distributed bonuses.

Implementing a better data storage system will allow more precise calculations of budgeted hours because Ray will have the data to analyze past project trends. As a result, bonuses will be better and more fairly-distributed because appropriate cost-drivers will be used in price scheming.

New System Proposal

As part of this project, we propose a new system that will involve a database, a program, a time-tracking feature, and a user interface. In this section, we will explore the system requirements and project plan.

System Requirements

We propose a system that will implement a centralized database to track project data, employee labor hours, client data, and other project aspects (like structure type, positions, statuses, and project types). We also propose the design and build of a website that will access this database and will allow employees to login, view assigned projects, view efficiency metrics, and allow Ray to assign projects to draftsmen. In addition, the office manager will be able to login and create, update, and delete project, client, and employee data. Finally, employees will also be able to 'clock in' and 'clock out'; the labor hours will be recorded and entered by project in the database.

The new system will be comprised of features from the previous system like Square (financial invoicing application) and new features that will include additional levels of security, employee logins, and effectively enter data to the database.

These subsystems are pieces of the larger South Fork Design system and will be described in greater detail below:

- Receive project details
- Enter project specifications
- Select project
- Record labor hours

Data Flow Diagram

The data flow for the new system will be slightly different and a bit more detailed than the current system. To illustrate this, we have provided context level, level-0, level-1, and level-2 diagrams.

Context Level

The context level diagram for South Fork's proposed system shows how data will interact and be passed with external entities (such as client, office manager, draftsman, and engineer). The diagram has minor changes from the current system. However, the success of the proposed system depends on how well the new system processes are implemented, in addition to the efficiency and usability of the new database.

The context level diagram can be found in Appendix B.

Level-0

Major systems and processes will continue to be detailed further below; however, a brief overview and additional context will be explained concerning the Level-0 diagram. A client sends a proposed project via email or call to the system. Included are project details and information and requirements that the client would wish to be implemented in the project. Project specifications are received by the office manager, and in process 2.0, the office manager will login to the system and then subsequently enter project data into the newly-implemented database.

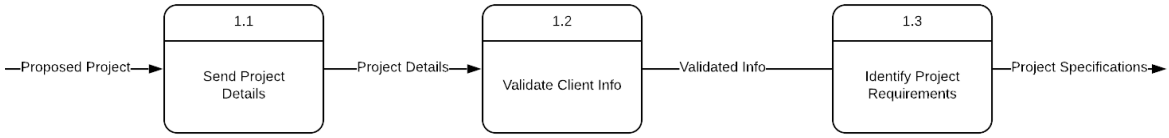
The system will then generate a bid price and validate the price with the client. If the client approves, a contract will be sent to the database, a deposit will be paid via Square, then a draftsman will be

assigned to oversee the project. If the client does not approve, then plans can be adjusted, and a new bid price can be generated to send to the client.

The client will create the draft in process 6.0, then record their labor hours in 7.0. Plans are passed to the engineer for review and edits, and then final plans are passed back to the client. If the client approves, additional financial data is sent and the final payment is made via Square.

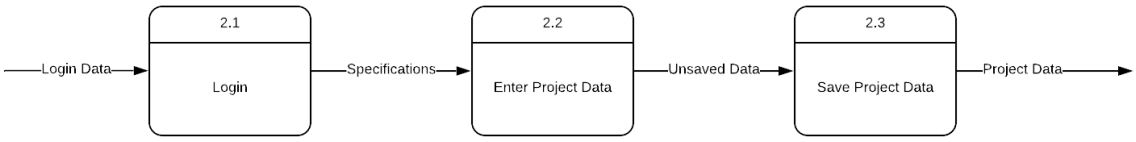
The level-0 diagram can be found in Appendix B.

1.0 Receive Project Details (Level-1)



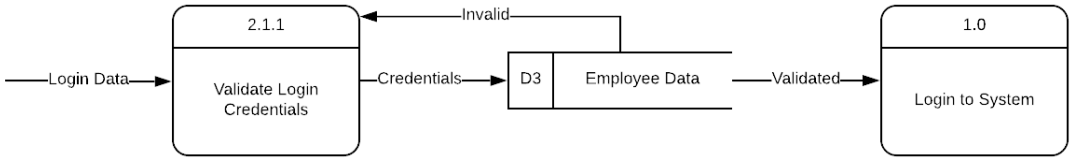
The proposed project is sent to the system and project details are received by the system. The client info is validated to make sure that all of the contact details are in place and that client can be consulted if further information is needed. Once client contact information is validated, project requirements are isolated and identified for further use within the system.

2.0 Enter Project Data (Level-1)



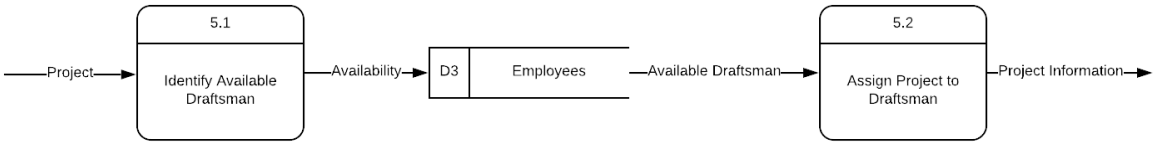
The office manager will enter login credentials to the system. Once validated, the office manager can then create, update, or delete projects. In this context however, the office manager will enter the project information into the database, and then save the project in the database.

2.1 System Login (Level-2)



In order to login to the system, the office manager will enter their login credentials to the system. The system will then validate the credentials by checking them against the data in the employee database. (For diagramming purposes, we labeled 'Employees' as D3 when the 'Employees' database is actually just a table in the centralized 'Projects' database). If credentials are valid, the system will redirect to the main menu of the system. If credentials are not validated, then the system will redirect to the login page.

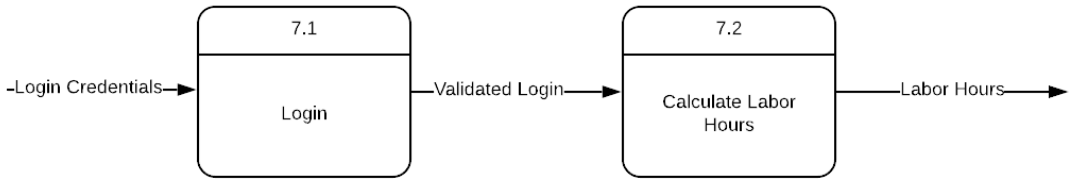
5.0 Select Project (Level-1)



The newly entered project will be assigned to an available draftsman through the following steps. The system (assuming the office manager is part of the system) will check for the availability of draftsman by searching the database and then assigning the project to a draftsman that isn't overwhelmed with

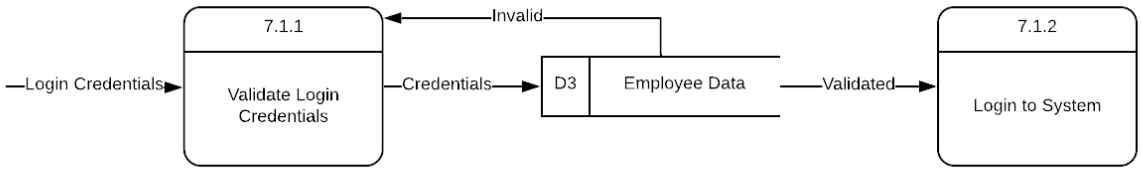
current projects. If no draftsman are currently available, then the project will wait in the project queue.

7.0 Record Labor Hours (Level-1)



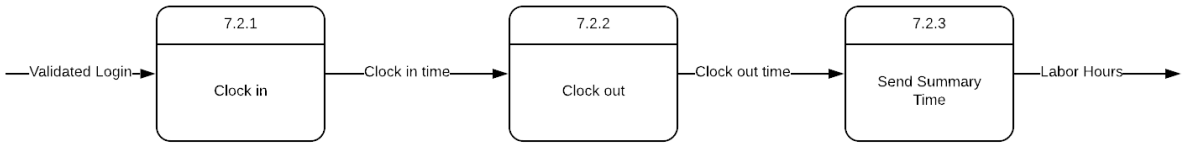
To record labor hours, employees can first login to the system and then choose to record their hours via a clock in and clock out function.

7.1 Login to System (Level-2)



This process is essentially similar 2.1 as described earlier.

7.2 Calculate Labor Hours (Level-2)



After employee login credentials are validated, employees will be redirected to the main menu and they can then record their labor hours by 'punching' a clock in and clock out button. The system will be careful to not allow multiple clock ins or outs, as well as only allowing the maximum 'in' time to be 10 hours. The summary time period will then be calculated and pushed to the database.

Database Design

To help increase accuracy in cost drivers, budgeting hours, and bidding; we plan to develop a database to store the necessary project, client, and employee data at South Fork.

The database will have seven different tables:

- Employees
- Projects
- Clients
- Positions
- Project Types
- Structure Types
- Statuses

Employees Table

The employees table will be used to store employee information and includes the following fields with their respective data types:

Field	Datatype	Name in Database
Employee ID	int	EmployeeID
First name	VARCHAR	EmpFirstName
Last name	VARCHAR	EmpLastName
Email	VARCHAR	EmpEmail
Phone number	VARCHAR	EmpPhone
Wage	float	EmpWage
Password	VARCHAR	EmpPassword

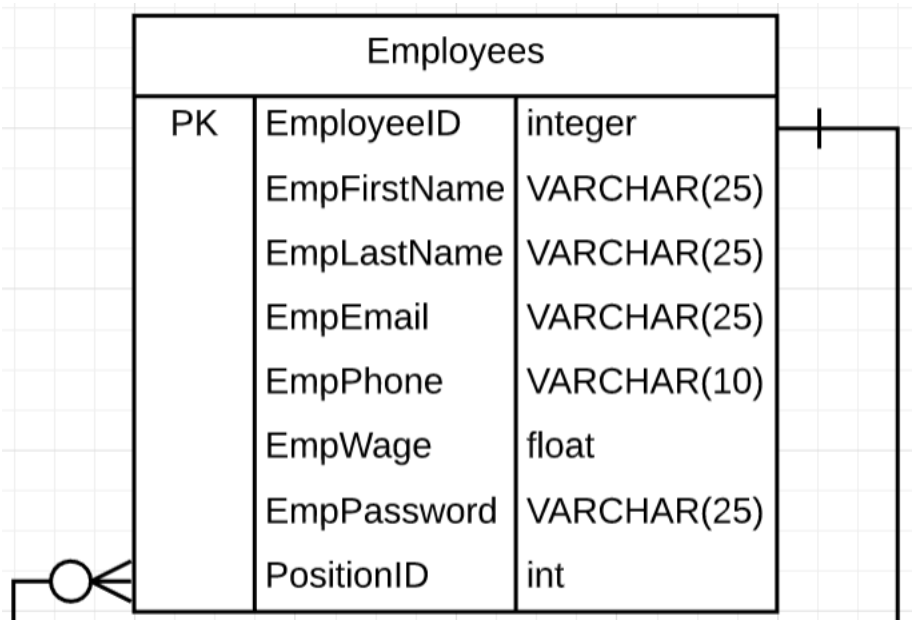
Employee position (FK)	int	PositionID

The ID field will be an auto-incremented integer used to promote integrity and usability in the database.

The email and password fields will be used for authentication to login to dashboards and assigned projects.

The PositionID field references the positions table to keep track of an employee's role in the company.

The image below shows the employees table as it appears in the ERD.



Projects Table

The projects table will be used for storing project data, including data for potential cost drivers. It will have the following fields and data types:

Field	Datatype	Name in Database
Project ID	int	ProjectID
Budgeted hours	float	BudgetedHours
Actual hours	float	ActualHours
Budgeted calendar days	float	BudgetedCalDays
Actual calendar days	float	ActualCalDays
Deposit date	date	DepositDate
Full payment date	date	PaidDate
Date plans delivered	date	DeliveredDate
Bid price	float	BidPrice
Final project price	float	ProjectPrice
Deposit paid	boolean	DepositPaid
Paid in full	boolean	PaidInFull
Square footage	float	SquareFoot
Size of deck	float	DeckSize
Existence of a hand-framed living space roof	boolean	HandFramedLivRoof
Project location in snow country	boolean	SnowCountry
Insulated concrete forms in structure	boolean	InsConcreteForms
Client ID for the project (FK)	int	ClientID
Employee ID for the project (FK)	int	EmployeeID
Structure type ID for the project (FK)	int	STypeID
Project type ID for the project (FK)	int	PTypeID

Status ID for the project (FK)	int	StatusID

The ProjectID field will be an autoincremented primary key used to track the data and preserve data integrity.

The following fields will be used to track budgeted and actual hours and calendar days of a project:

- BudgetedHours
- ActualHours
- BudgetedCalDays
- ActualCalDays

While we could have made both budgeted fields calculated fields, we decided not to, because these numbers could possibly change throughout the life of the project as client specifications change.

The following fields will be used to keep track of the dates involved with the project:

- DepositDate
- BeginDate
- CompletionDate
- PaidDate

The following fields will be used to keep track of the pricing and payments of the project:

- BidPrice
- ProjectPrice
- DepositPaid
- PaidInFull

The bid price could also be a calculated field, but as we mentioned before with the budgeted fields, this field could change throughout the life of the project, so we decided to keep the hard-coded field in here instead.

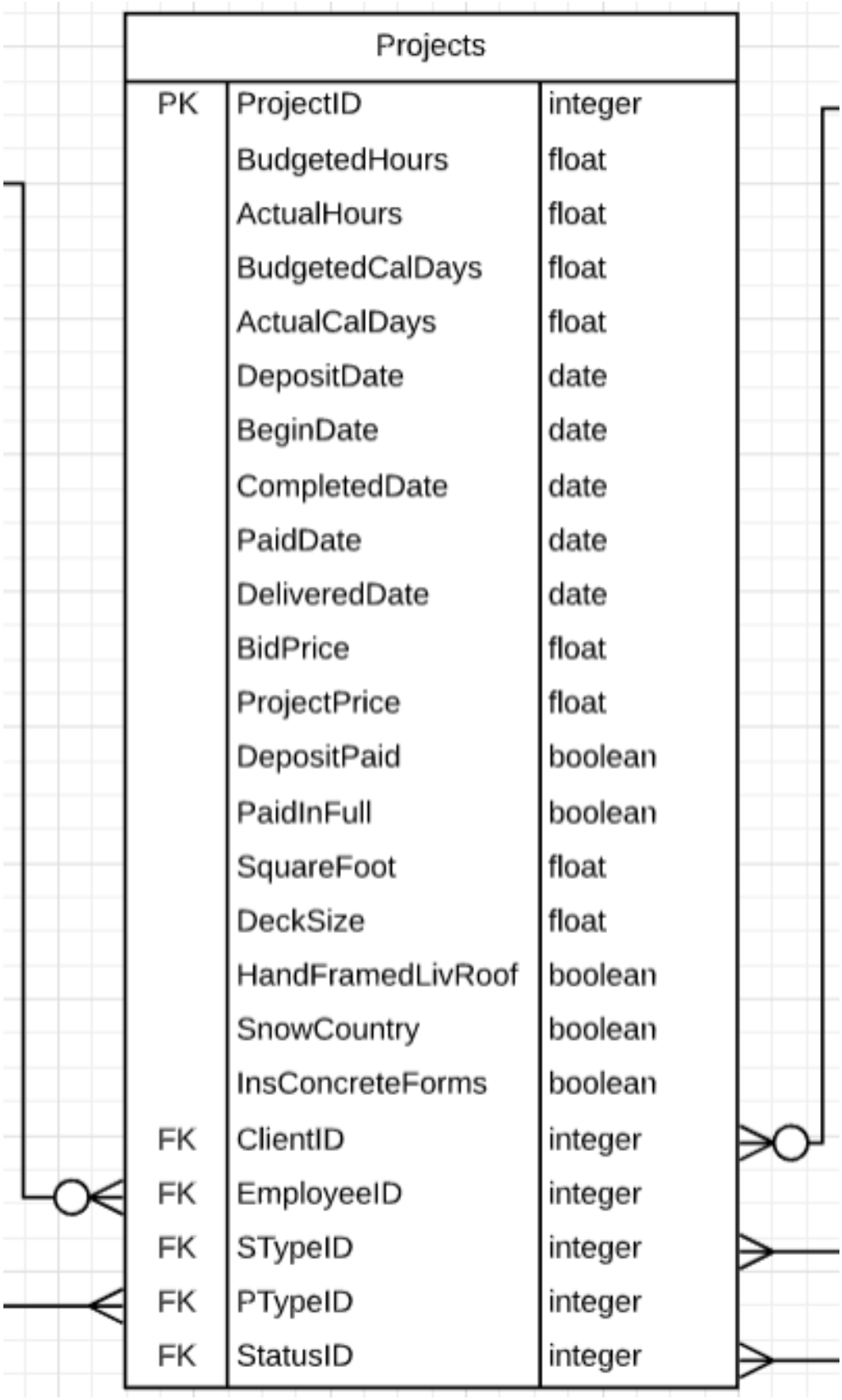
The following fields will be used to track potential cost drivers of a project:

- SquareFoot
- DeckSize
- HandFramedLivRoof
- SnowCountry
- InsConcreteForms

The following keys are foreign keys that will refer to other tables:

- ClientID
- EmployeeID
- STypeID
- PTypeID
- StatusID

The image below shows the projects table as it appears in the ERD.


Projects
PK ProjectID integer
BudgetedHours float
ActualHours float
BudgetedCalDays float
ActualCalDays float
DepositDate date
BeginDate date
CompletedDate date
PaidDate date
DeliveredDate date
BidPrice float
ProjectPrice float
DepositPaid boolean
PaidInFull boolean
SquareFoot float
DeckSize float
HandFramedLivRoof boolean
SnowCountry boolean
InsConcreteForms boolean
FK ClientID integer
FK EmployeeID integer
FK STypeID integer
FK PTypeID integer
FK StatusID integer

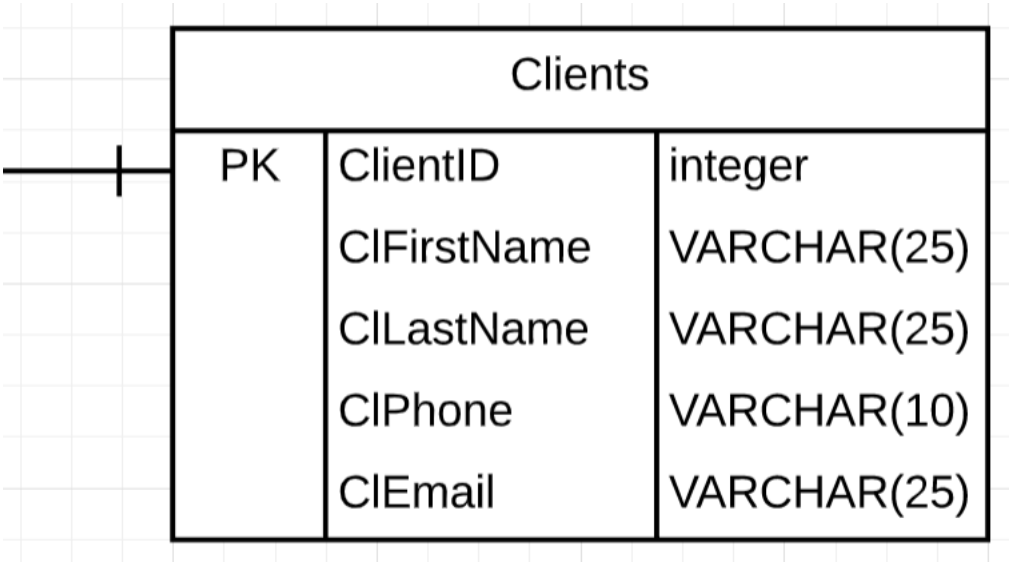
Clients Table

The clients table will be used to keep track of client information and will include the following fields:

Field	Data type	Name in database
ID of the Client	int	ClientID
Client first name	VARCHAR	CFirstName
Client last name	VARCHAR	CLastName
Client phone number	VARCHAR	CPhone
Client email	VARCHAR	CEmail

The ClientID field will be an auto-incremented field used as a primary key.

The Clients table as it is in the ERD is shown below.



Positions Table

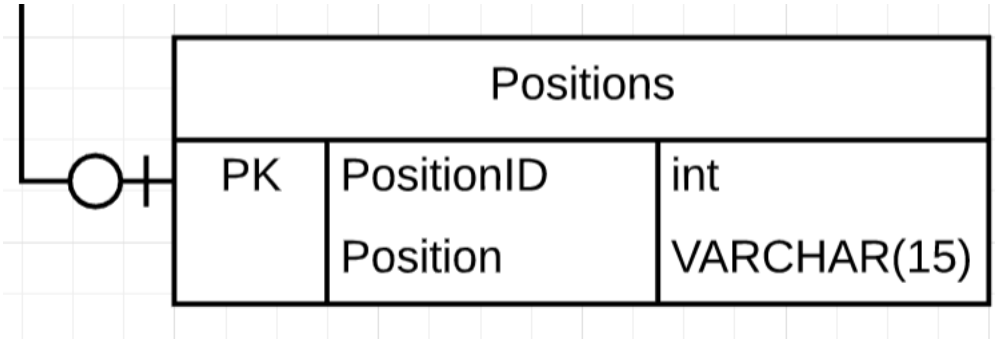
The Positions table is a lookup table where the employee position/title will be stored. It will have the following fields.

Field	Data type	Name in database
Position ID	int	PositionID
Position description	VARCHAR	Position

The different position records will include:

- Engineer
- Draftsman
- Office Manager

The table as it appears in the ERD is shown below.



ProjectTypes Table

The ProjectTypes table is a lookup table for storing the type of project. This will include the following fields:

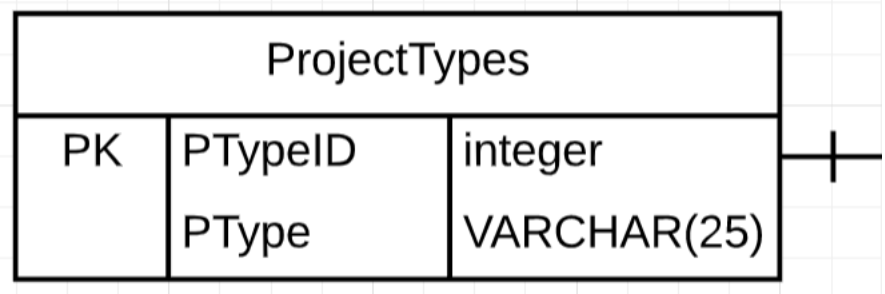
Field	Data Type	Name in Database
Project Type ID	integer	PTypeID

Project type description	VARCHAR	PType

The PTypeID field will be an autoincremented field for the primary key. The PType field will hold the description of the project type and will hold the following records:

- Engineering-only project
- Design-only project
- Full project

We hope to track project types to better measure the profitability of different types of projects. The table from the ERD is shown below.



StructureTypes Table

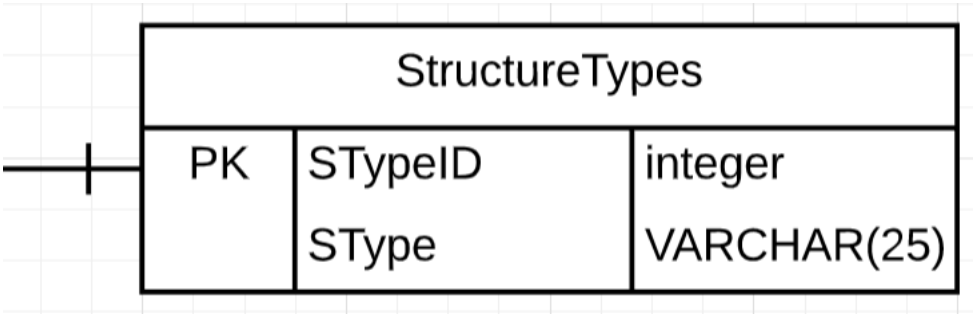
The StructureTypes table is a lookup table. It's fields are listed below.

Field	Data type	Name in database
Structure type ID	integer	STypeID
Structure type description	VARCHAR	SType

The STypeID field will be an autoincremented primary key, and the SType field will contain the description. The SType field will include the following records:

- Addition
- Commercial
- Residential

As South Fork tracks this data, they will be able to better determine profitability and cost drivers in the company. The table as shown in the ERD is depicted below.



Statuses Table

The Statuses table will be used to track the projects and which projects still need finished. It will include the following fields

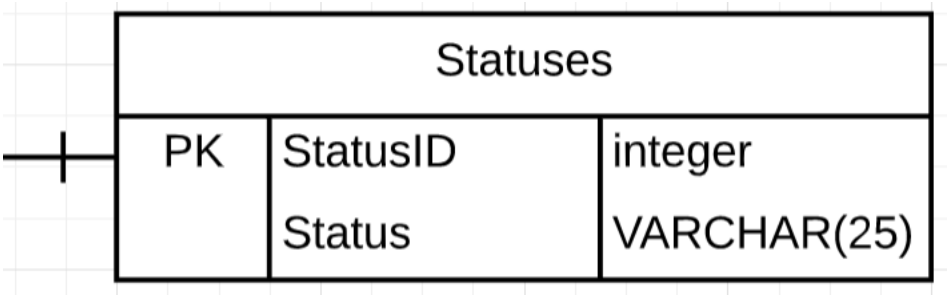
Field	Data type	Name in Database
Status ID	integer	StatusID
Status description	VARCHAR	Status

The StatusID field will be an autoincremented primary key. The Status field will be a description field and will include the following records

- Unassigned, not started
- Assigned, not started

- In progress
- Completed
- Delivered

The table as it appears in the ERD is shown below:



The complete ERD is in Appendix A.

Program

If built and implemented correctly, the system will automate, centralize and facilitate many tasks performed by the company as it stands today.

Primarily, the system will be designed to interface with the new database; it will pull data from the database to generate dynamic reports that will aid with the strategic planning of the company and it will write to the database, storing information about new projects, clients, and employees.

Saving Data

Unless new data can be easily and accurately stored in the database, it is unlikely that this project will yield the benefits it promises. For that reason, the MVC web application our team will design will have create, delete, and edit methods on every relevant table of the database.

Adding Projects

For SouthFork Design, drafting for clients is the entire business. The system must have methods that facilitate the creation of new projects and the modification of old projects. As the project price, assigned designer, project status, and cost drivers can change with time, all of these must be editable.

Adding and Modifying Employees

As the company grows, it is essential that it be able to expand to accommodate changes in the number of employees; therefore, the system must also have functionality to add, remove, and edit employees to properly adjust to changes in personnel.

Tracking Time

Because Ray is deciding to branch away from commercial time tracking software like Toggl, the new system must also be capable of recording employee hours. These hours will not only contribute to employee pay, but also, the service will automatically store this data with the associated project, thereby reducing the required work for employees to transfer data between unintegrated systems.

Data Interpretation

While creating, using, and editing data in an organized database will aid the company in storing its data, these things won't help provide any insight into what the company can do to increase profitability or efficiency. In addition, at this time, the designers are inexperienced with data analytics, SQL, Tableau, and other methods of data visualization and it is unlikely that those circumstances will change within the foreseeable future. In order for the database to fulfill its purpose properly, dynamic data displays will be created in the MVC web application. These dashboards will

be designed to display relevant information about project costs versus benefits and employee profitability. In addition, they must organize and display data in such a way that individuals unfamiliar with data analytics can interpret them and make strategically appropriate decisions based on them.

Hassle-free Modification of Dashboard

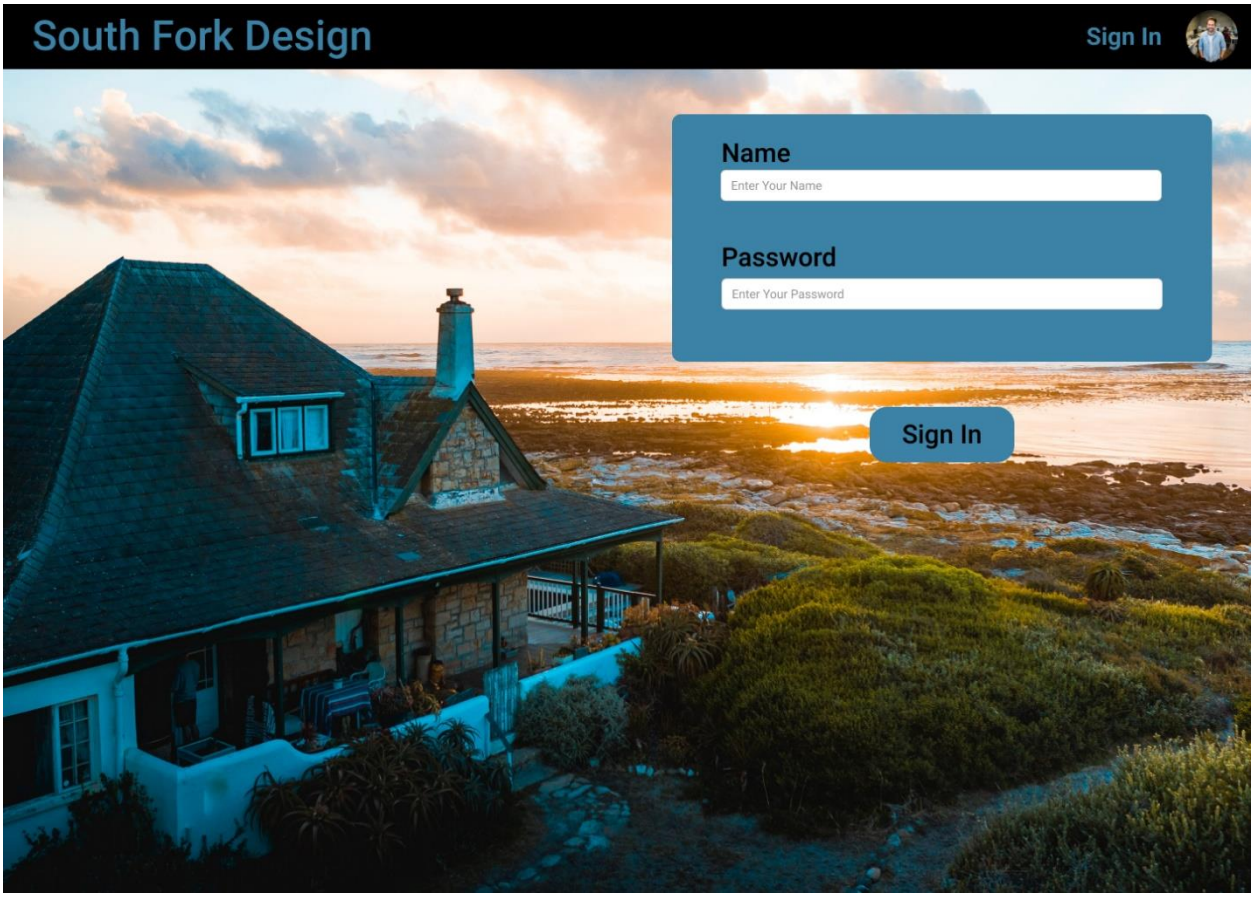
Because our team is unlikely to perfectly anticipate every data need of South Fork for the near future, it is probable that Ray will want these dashboards in the new system to be modified in the future. The system should be designed in such a way that those modifications are easy to implement so that, as the company and its employees become more familiar with the system and their own data, they can make low-cost modifications to reports that more accurately illustrate the trends that will assist them in running their business more effectively

User Interface

One of the final steps for this project to be successful is to create a way for those who don't understand the technical models we create to understand the recommended implementations. The best way to represent our changes is with user interface mock-ups of what a new system might look like. The following models demonstrate how a user will interact with the new system to easily record, access, view and analyze the needed data that will be collected. This step is vital because you can have all the information in the world but if you don't know how to interact with it, then it is useless. With a well-designed-and-tested user interface, each person who interacts with the system will be able to access the data that they need to positively influence the business processes, increase productivity, and maximize revenue. Here are some of the mockups that were created.

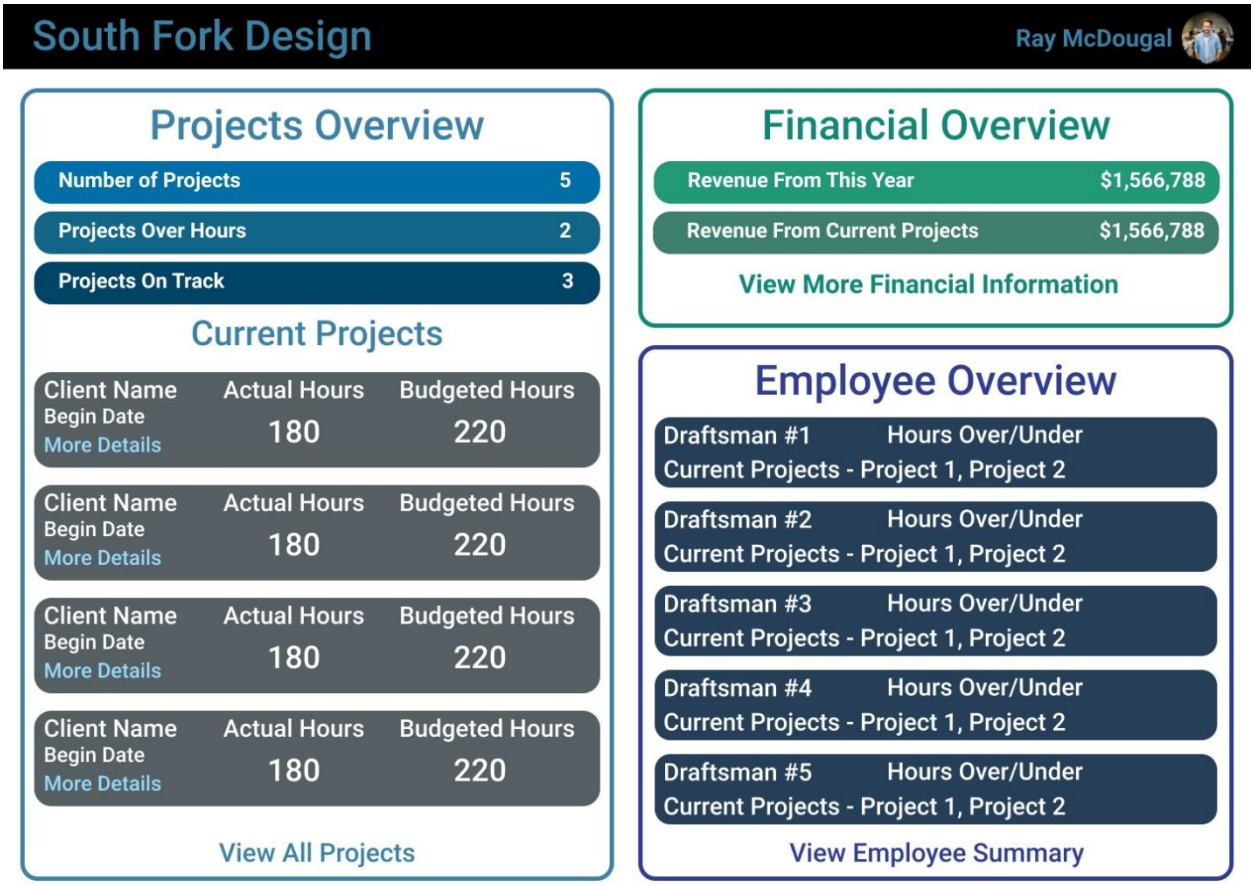
Sign-in Page

The sign-in page displayed below demonstrates the security that will be needed for the program. Sensitive financial and client data will be stored in the system so security will have to be key when development begins. The sign-in page is also aesthetically designed to make the experience of using the software pleasant from the beginning.



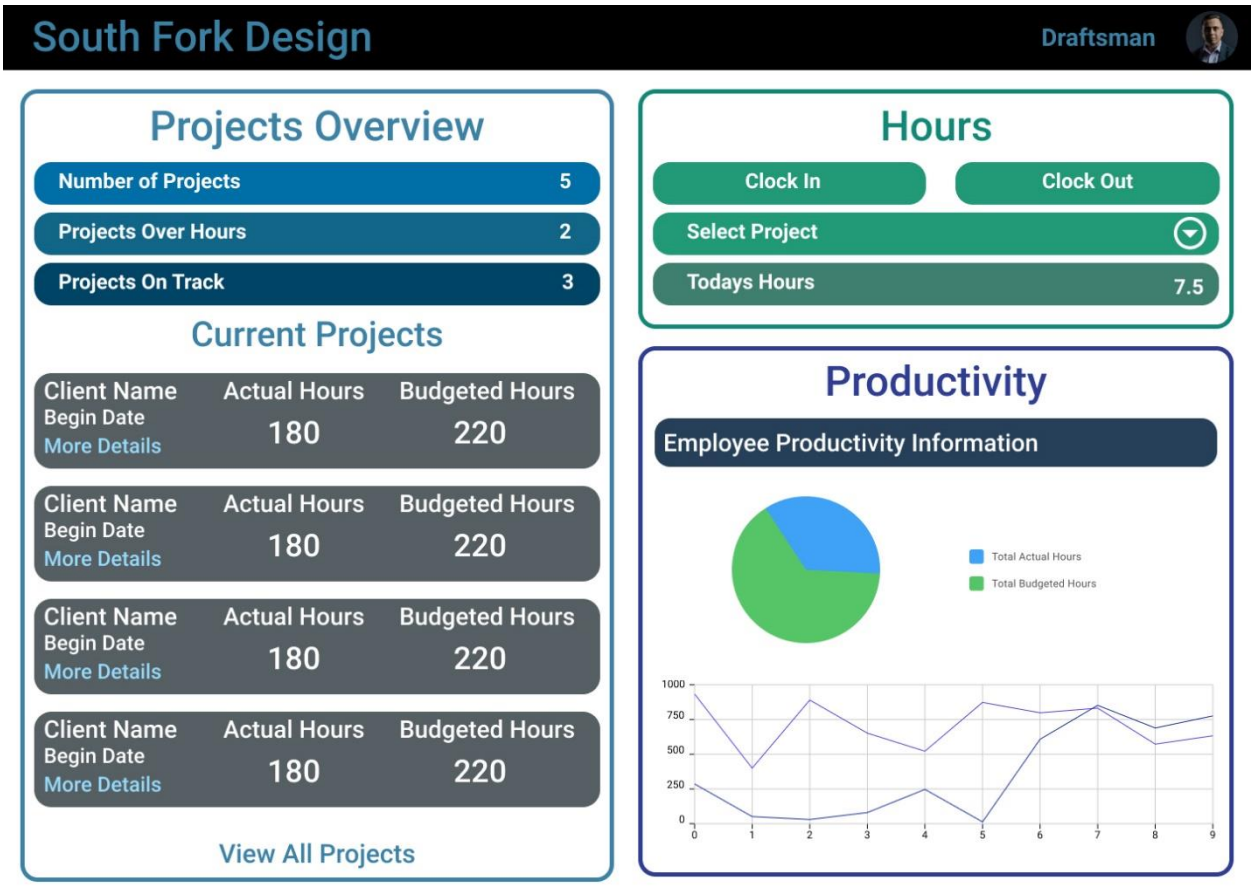
Ray's Landing Page

Ray is the owner of the company; therefore, he needs the most important information easily accessible from a dashboard. This mockup shown below gives him immediate information about which projects are ongoing, how his employees are performing and overall financial information about the company. Each of these sections gives a short overview and has easy access links to get to the more detailed analytical information. The end goal of this dashboard is to bring the most important information immediately to the owner of the company.



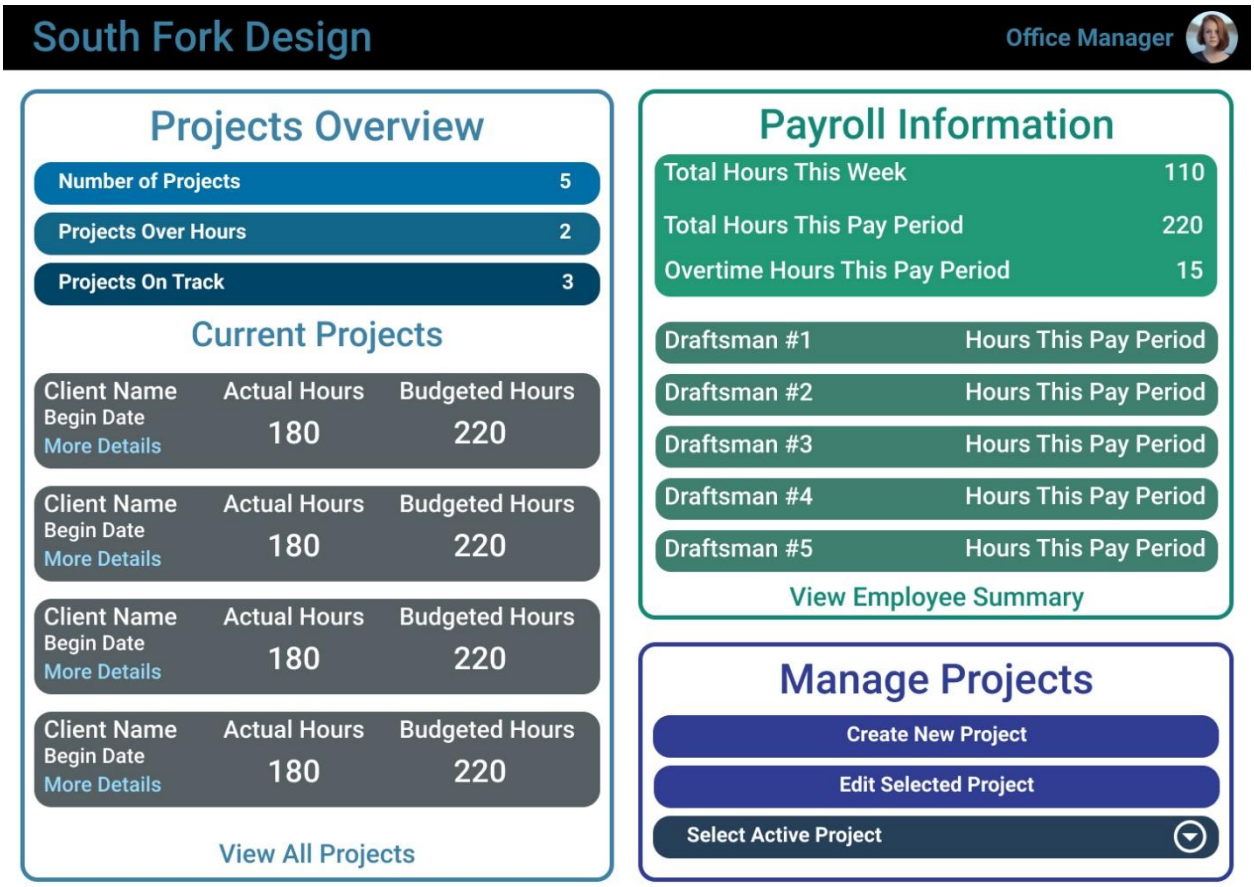
Draftsman Landing Page

The draftsman dashboard serves the same purpose as Ray's dashboard but is draftsman-specific and brings only the most important information about projects, time, and productivity to the draftsman. The project overview below allows quick access to summaries as well as important information about each of the projects that the draftsman is assigned to work on. The hours view allows the draftsmen to clock in and out of projects and view basic information about the time that they work. Finally, the productivity section allows for each draftsman to see what areas they are excelling in and where they can improve as well as holding each draftsman accountable for the quality and efficiency of their work.



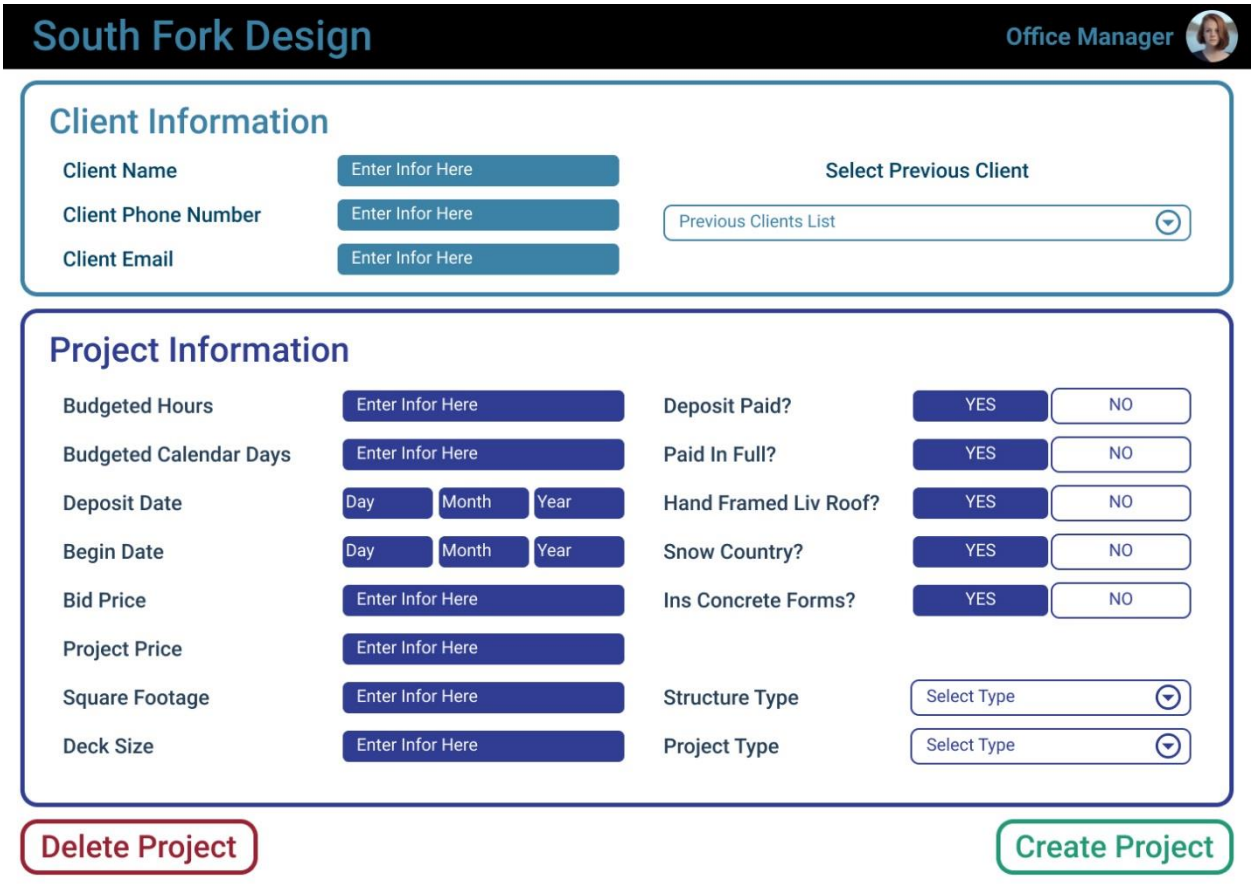
Office Manager Landing Page

The Office Manager oversees the day-to-day information in the company. Their dashboard shown below allows them to easily track which projects are ongoing as well as create, view, update and delete projects. This allows them to make sure that the information in the database is always up to date. They also have responsibility for payroll information which they can easily view in the payroll information section.



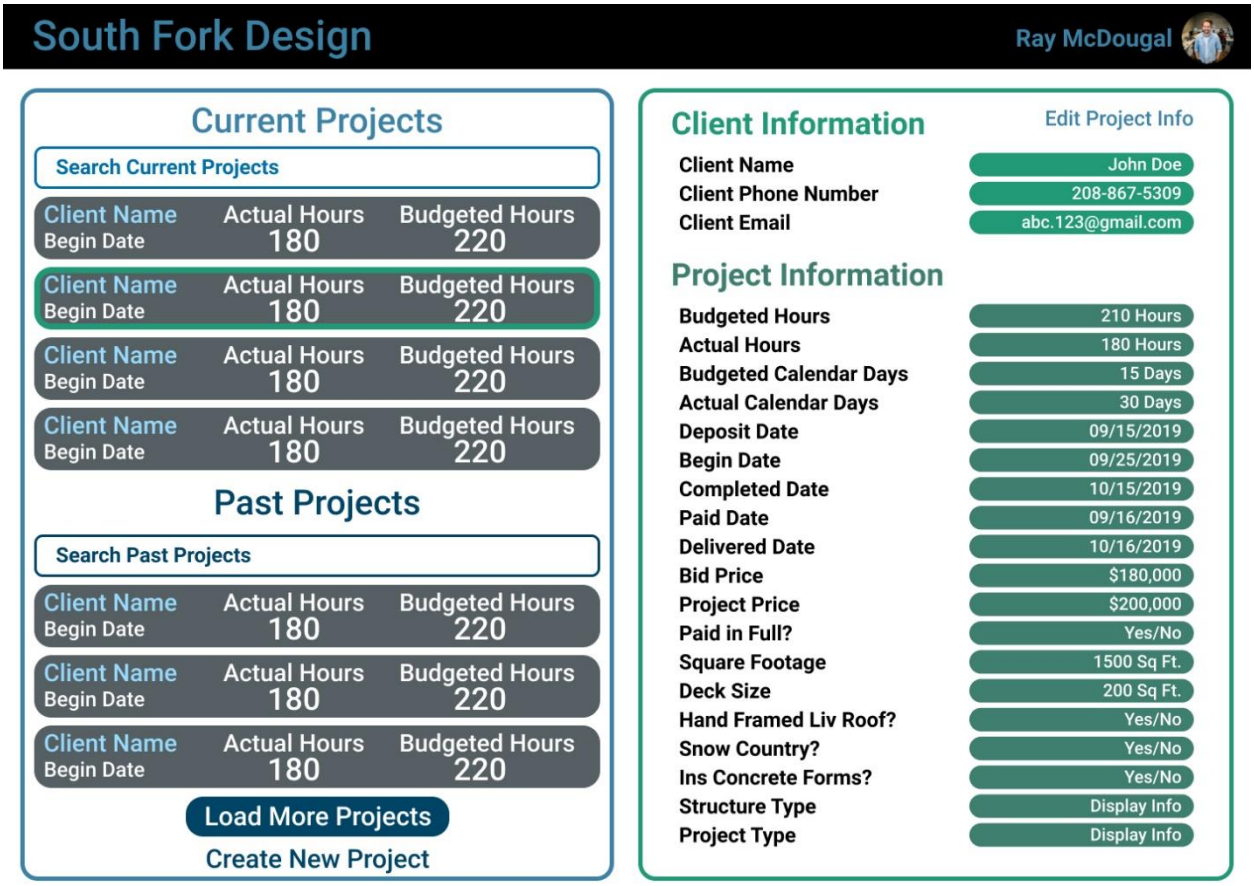
Create, Read, Update, and Delete Page

The create, read, update and delete project page displayed below can be viewed by any of the employees for the purpose of keeping the database up to date. Typically, the create and delete functions will be performed by the office manager while the read and update functions will be done by Ray or the draftsmen. This page is critical because it is where the data tracking starts. When all the fields are filled out, the goal of our project is met and information about the projects will be effectively stored. This information can then be used later to help the company maximize profit and employee productivity.



Detailed Project View Page

When an employee wants more than just the summary information about a project, they can view this page shown below. In this detailed project view, they can search through and view all current and past projects. When they select their desired project, they will be able to view all the stored information about that project. Any information that is collected about a project from start to finish can be found here for reference. This allows any employee access to information about their current and past work that they can easily reference at any time.

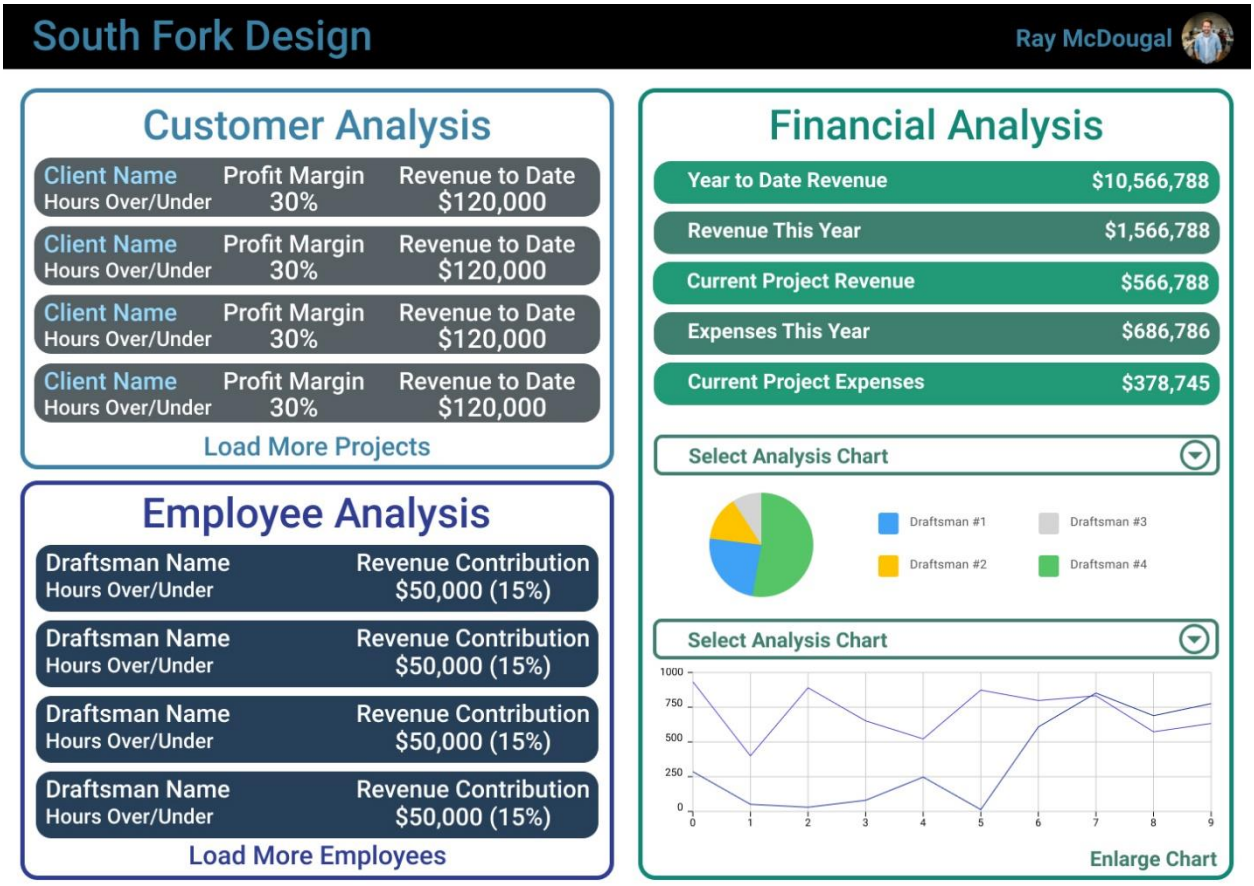


Employee Detail Page

The Employee detail page shown below is for the use of the owner, Ray. This page allows him to view information about each employee including their contact information, which projects they are working on and how much they get paid. He can also select an employee to view more statistics about their productivity, including their revenue contribution, hours ratio (budgeted vs. actual), and other charts or statistics that will help him analyze each employee's performance.

Financial Analysis Page

This financial analysis page below provides all the information that the owner, Ray, needs to analyze the past, present, and future of his company. He can view profitability statistics organized by clients, employees, and projects. He can also view any number of financial statistics that he wishes to track. This page also focuses on displaying the information in many different forms of charts and visualizations. This will allow the owner to make informed financial decisions as well as identify the weak and strong points of his company.



Project Plan

The following screenshot from Microsoft Project shows the tasks of our project:

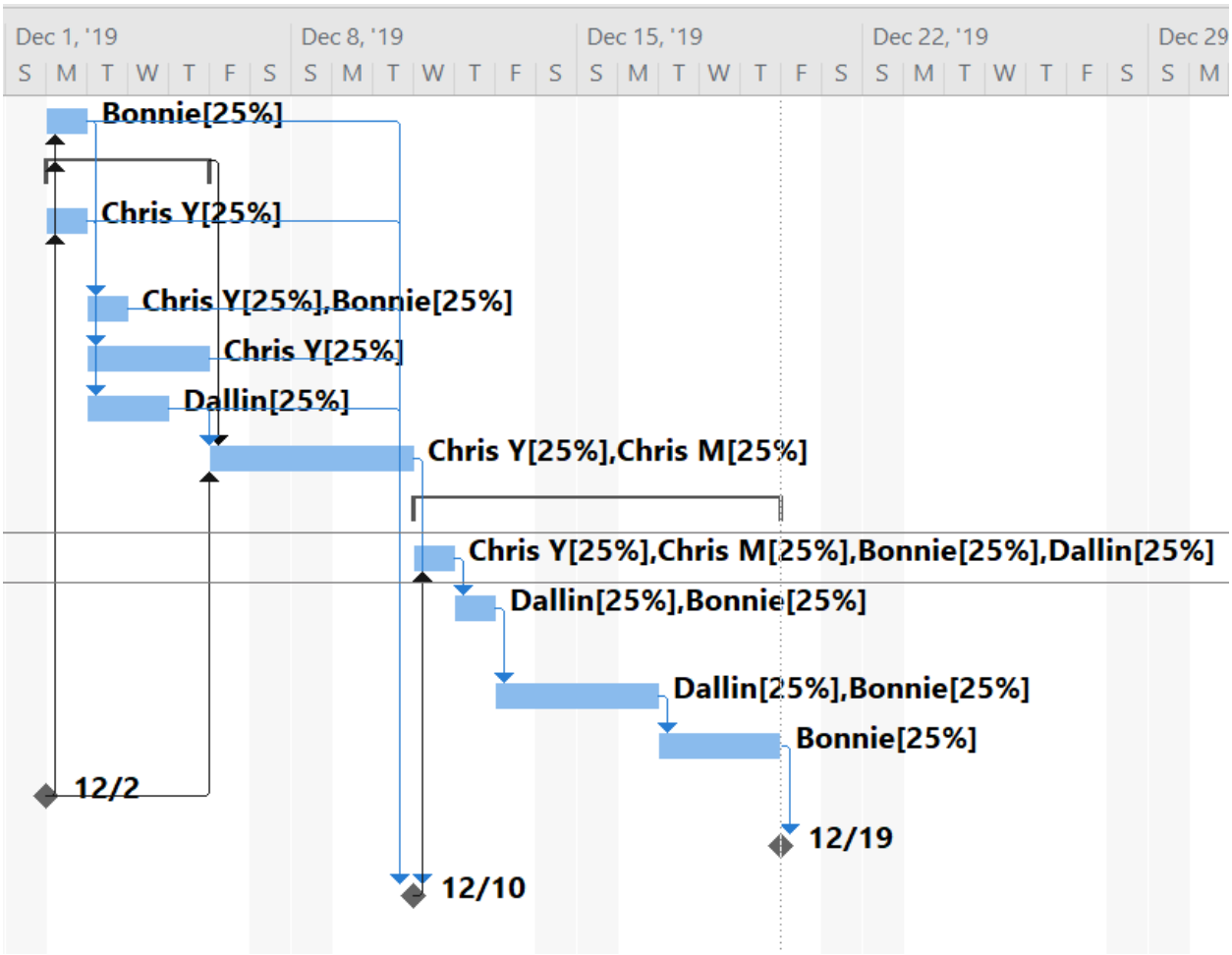
WBS	Task Name	Owner	Duration	Start	Finish	Predeces	Succe	Resource Names
1	Database Build	Bonnie	1 day	Mon 12/2	Mon 12	13	4,15	Bonnie[25%]
2	**System Build**	**Chris Y**	**4 days**	**Mon 12/2**	**Thu 12/**	**13**	**7**	
2.1	Build Models and Controllers	Chris Y	1 day	Mon 12/2/19	Mon 12/2/19	13	5,6,15	Chris Y[25%]
2.2	Connect to database	Chris Y	1 day	Tue 12/3/	Tue 12/	1	15	Chris Y[25%],Bonnie[25%]
2.3	Build Views	Chris Y	3 days	Tue 12/3/	Thu 12/	3	15	Chris Y[25%]
2.4	Write methods	Chris Y	2 days	Tue 12/3/	Wed 12	3	7,15	Dallin[25%]
3	Build UI	Chris M	3 days	Fri 12/6/1	Tue 12/	2,13,6	15	Chris Y[25%],Chris M[25%]
4	**Testing and deployment**	**Dallin**	**7 days**	**Wed 12/1**	**Thu 12/**			
4.1	Test with team	Dallin	1 day	Wed 12/1	Wed 12	15	10	Chris Y[25%],Chris M[25%],Bonnie[25%],Dallin[25%]
4.2	Add current data to database	Dallin	1 day	Thu 12/12/19	Thu 12/12/1	9	11	Dallin[25%],Bonnie[25%]
4.3	Deploy	Dallin	2 days	Fri 12/13/	Mon 12	10	12	Dallin[25%],Bonnie[25%]
4.4	Test with South Fork	Dallin	3 days	Tue 12/17	Thu 12/	11	14	Bonnie[25%]
5	Start		0 days	Mon 12/2	Mon 12		1,3,2,7	
6	Finish		0 days	Thu 12/19	Thu 12/	12		
7	o: finished system	Bonnie	0 days	Tue 12/10	Tue 12/	1,3,4,5,6,7	9	

This project will have four main components:

1. Database build
2. System build
3. UI Build
4. Testing and Deployment

Each component has been broken down into smaller tasks and assigned to different team members.

We plan to start this project on December 2, 2019, and the project would be finished on December 19th. The following image shows a timeline of the project as shown in Microsoft Project.


Dec 1, '19
S M T W T F S
Dec 8, '19
S M T W T F S
Dec 15, '19
S M T W T F S
Dec 22, '19
S M T W T F S
Dec 29
S M
Bonnie[25%]
Chris Y[25%]
Chris Y[25%],Bonnie[25%]
Chris Y[25%]
Dallin[25%]
Chris Y[25%],Chris M[25%]
Chris Y[25%],Chris M[25%],Bonnie[25%],Dallin[25%]
Dallin[25%],Bonnie[25%]
Dallin[25%],Bonnie[25%]
Bonnie[25%]
12/2
12/19
12/10

Works Cited

Shelton, Crystalynn. 2019. *10 Best Time Tracking Software 2019*. 21 Feb. https://fitsmallbusiness.com/best-time-tracking-software/.

South Fork Design Group. 2015. *Home*. http://www.southforkengineering.com/.

Appendices:

The appendices are divided into three sections:

- Current forms and reports
- Proposed forms and reports

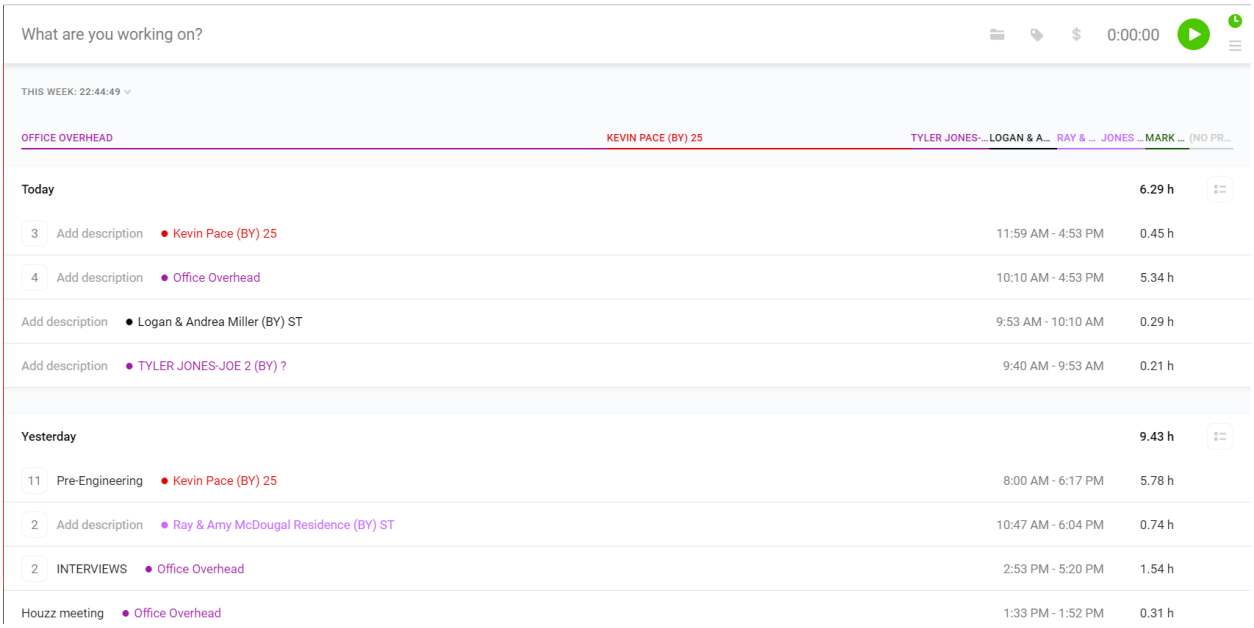
Appendix A: Current Forms and Reports

The current forms and reports include

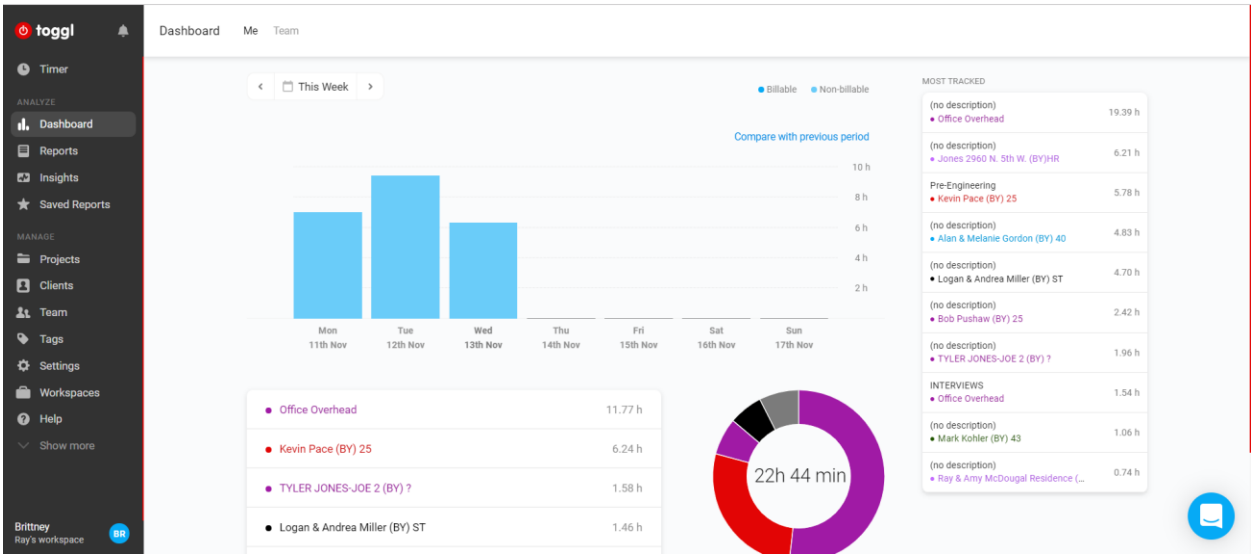
- Toggl interface screenshots
- Excel Spreadsheet
- Activity diagram
- Context level data flow diagram
- Level-0 data flow diagram
- Level-1 data flow diagram

Toggl Interface

The image below shows the hour tracking interface of Ray's draftsman, Brittney. This interface includes a timer system. They can also select which projects they are working on, which are shown through the color coding.



Toggl also has a dashboard capability to see the number of hours worked on individual projects. The screenshot below shows Brittany's weekly summary.

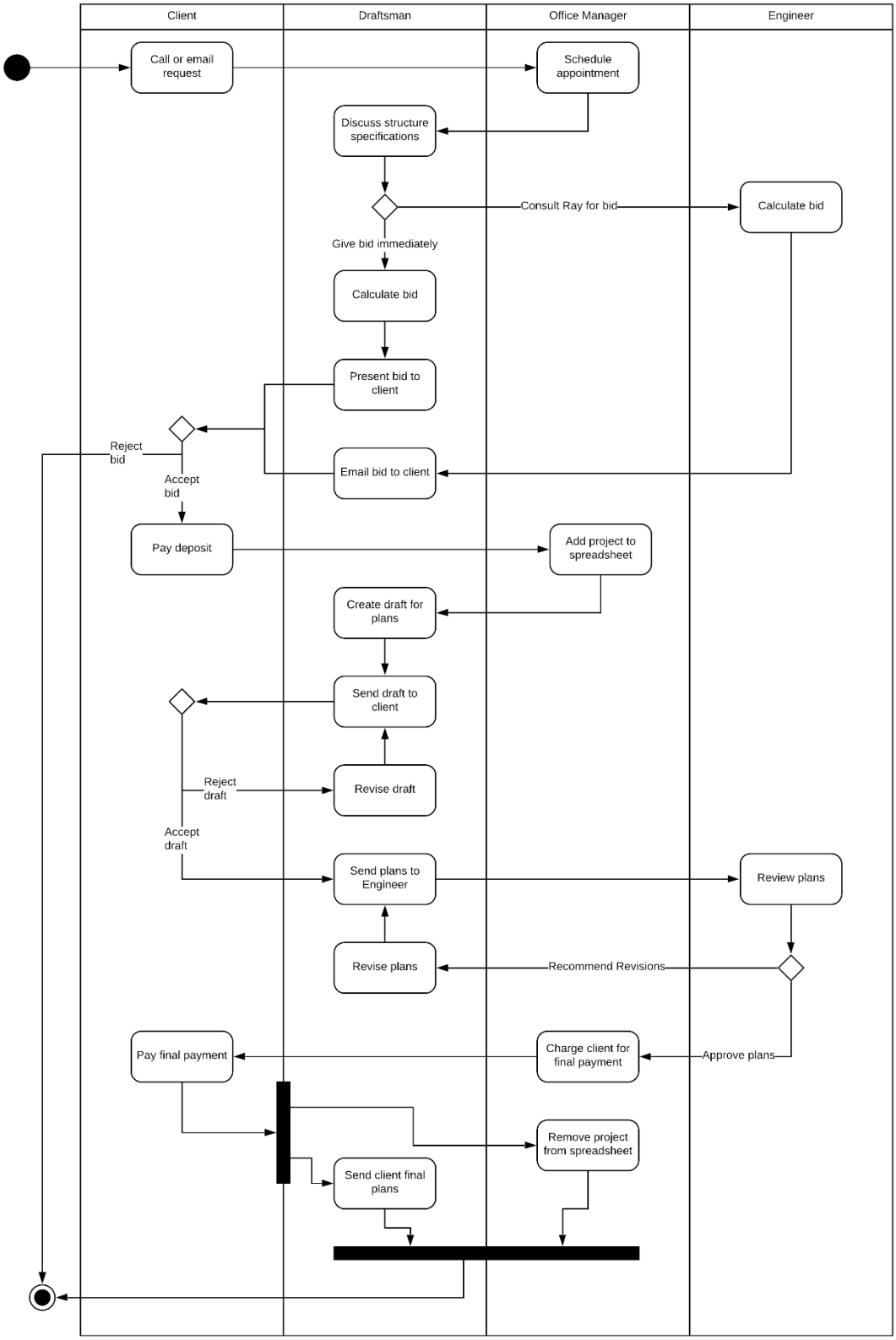


Excel Spreadsheet

The image below shows a screenshot of the excel spreadsheet used by South Fork to keep track of projects. They use colors to indicate months and status of projects. While the tracking is extensive, the spreadsheet leaves a significant amount of room for human error.

Client Name	Date Deposite Paid	Est. Start	Project Type	Hours Set: By Standard Or By Retainer	Hours Used	Hours Left	Plan Status	Next Preliminary Date	Done	Printed	# of Sets
Alan & Melanie Gordon		11.4.19	Standard	40			New Job				
Bauer Barn			Hourly	Hourly	6.6		In Progress				
Bob Pushaw	10.10.19	10.24.19	Standard	25			New Job				
Brian Hegsted			Standard	35	22.11	12.89	In Progress				
Greg Chatman (IW)			Standard	5.6	9.57	-3.97	Out for Final Review				
Harrison Overlay	09.06.19		Straight		8.09		Done-September		09.06.2019	Yes	4
Janet & Armond LaPine	11.16.18	11.30.18	Standard+Overage	23	27.51	-4.51	In Progress				
Jason Green : Res. Addition		9.23.19	Hourly		6.86		Out For Final Review				
Kevin Pace	9.19.19	10.3.19	Standard	25	6.83	18.17	In Progress				
Logan & Andrea Miller	NA	11.5.19	Straight				New Job				
Mark Kohler	09.17.19	10.1.19	Standard	43	5.85	37.15	In Progress				
Ray and Amy McDougal			Straight		14.41		In Progress				
Tyler Jones - IP Joe			Hourly		4		Out For Final Payment				
Tyler Jones - Jo 2			Hourly		10.9		In Progress				
Royce - Daybreak Rev.	???			10	11.57	-1.57	Hold		Hold		
Royce - Twin Rivers Cabin & Garage				59	24.73	34.27	Hold		Hold		
Bobbie & Brian Lehnhof	10.30.18	11.12.18	Standard	80	43.75	36.25	Hold		Hold		
Dayton Wall	6.4.19			35	34.74	0.26	Hold		Hold		
Lance Boyce Duplex	3.13.19		Standard	30	3.34	26.66	Hold		Hold		
Popovich Deck Changes			Hourly	Hourly	12.23		Hold		Done		
Bo Porter: Building 3 Framing Modification			Hourly	Hourly	1.41		Done-October		10.31.2019	Email	
Sean Southwick - Foundation Update			Hourly		3.37		Done-October		10.09.2019	Yes	4
Ted Erickson	8.27.19 PD in Full	9/10/2019	Standard	10	9.79	0.21	Done- October		10.02.2019	Yes	4
Bo Porter: Building 3	7.9.19		Hourly		36.01		Done-September		09.30.19	Yes	4
Gerald Stucki: Engineering		9.25.19	Straight		1.63		Done-September		09.26.2019	Yes	4
Danny Ricks	7.18.19		Standard	54	44.22	9.78	Done-September		09.16.2019	Yes	4
Ron Dye	8.1.19		Standard	10	14.39	-4.39	Done-September		09.04.2019	Yes	4
Jason Green 30x30			Standard	13	6.94	6.06	Done-August		08.16.2019		
Lori & Aaron Richards	3.8.19	3.19.19	Standard	49	14.14	34.86	Done-August		08.16.2019		
David White	2.18.19	3.4.19	Standard	30	41.38	-11.38	Done- August		08.14.2019		
Blundell Cabin		6.12.19	Straight	33	22.84	10.16	Done- August		08.05.2019		

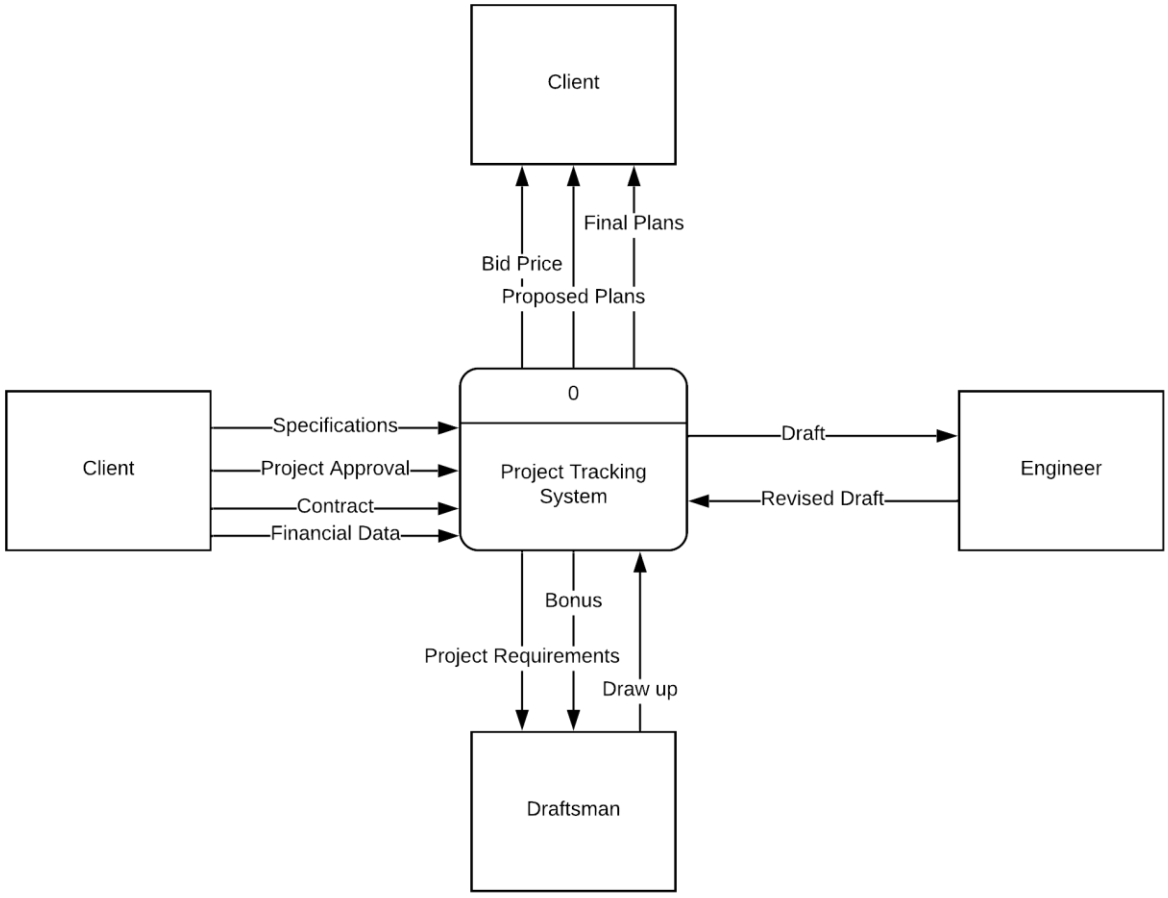
Current Activity Diagram



Activity Diagram Description

The client contacts South Fork Design via phone or email to submit a project request. The Office Manager receives the request and sets an appointment with an available draftsman to discuss project specifications and requirements. Then, the draftsman can either consult Ray to calculate a bid price, or calculate the bid immediately and present it to the client. If Ray calculates the bid price, he will email it to the client. After the client receives the bid price, they can either accept it, or reject it. If the client rejects the bid price, the process ends. If the client accepts the bid, then they pay a deposit and the Office Manager adds the project to the project spreadsheet tracker. Then, the draftsman creates a draft of the initial plans and sends them to the client for approval. If the client rejects the draft, the draftsman will revise the draft and send it back to the client for approval. If the client approves the draft, then the draftsman will send the draft plans to the engineer for reviewal and structural modifications. If the engineer recommends revisions, then the draftsman will revise the plans and send them to the engineer for approval. If the engineer approves the plans, then the Office Manager will charge the client for the final payment and then the client will make that payment. Then simultaneously, the Office Manager will remove the project from the tracking spreadsheet while the draftsman sends the final plans to the client. Then the process ends.

Current Context Level Diagram



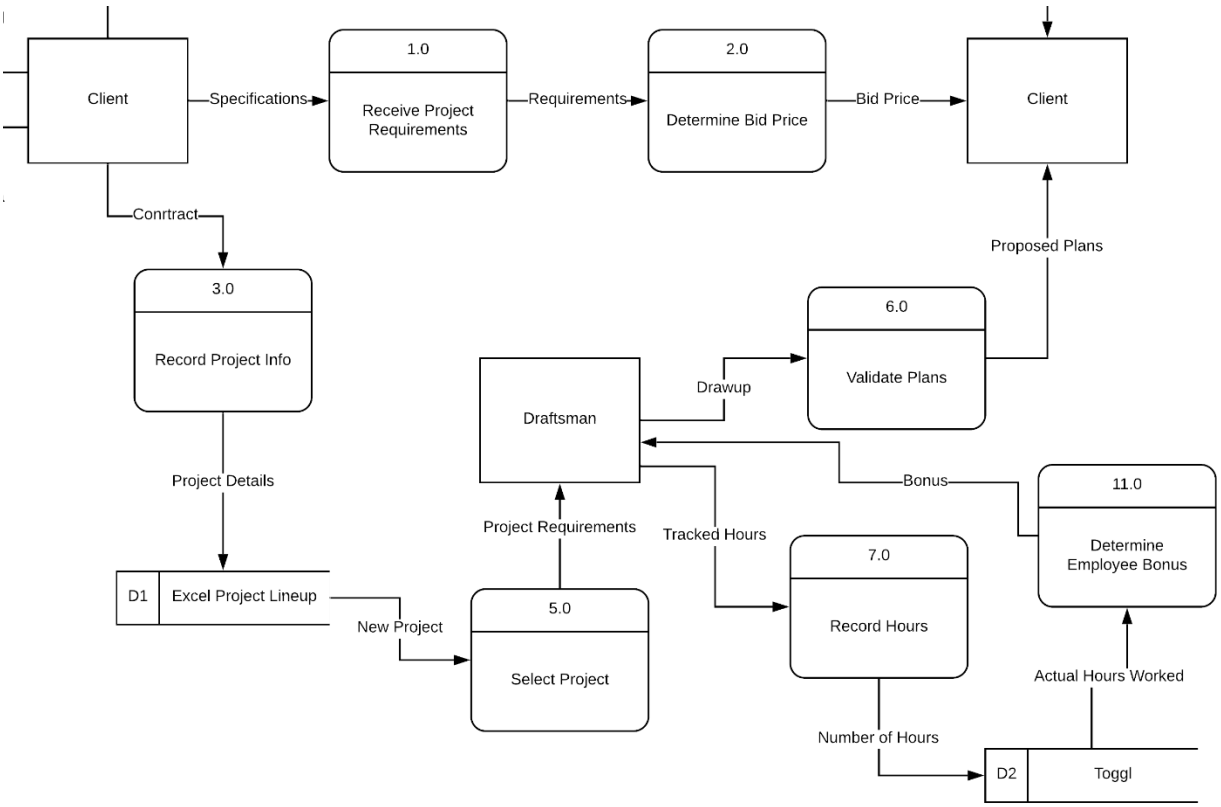
Context Diagram Description

The main flows of data within the Project Tracking System are as follows: the client sends project specifications to the system. A bid price is created and sent to the client. The client then signs and sends a contract to the system, finalizing the deal. The draftsman then receives the project requirements and creates a draw up which is then passed by the system to client as the proposed plans. The clients sends approval for those plans, which are then passed to the engineer as a draft. The engineer revises and edits the draft and passes the revised draft to the system. The final plans are sent to the client, and the client passes financial data to the system for the final payment.

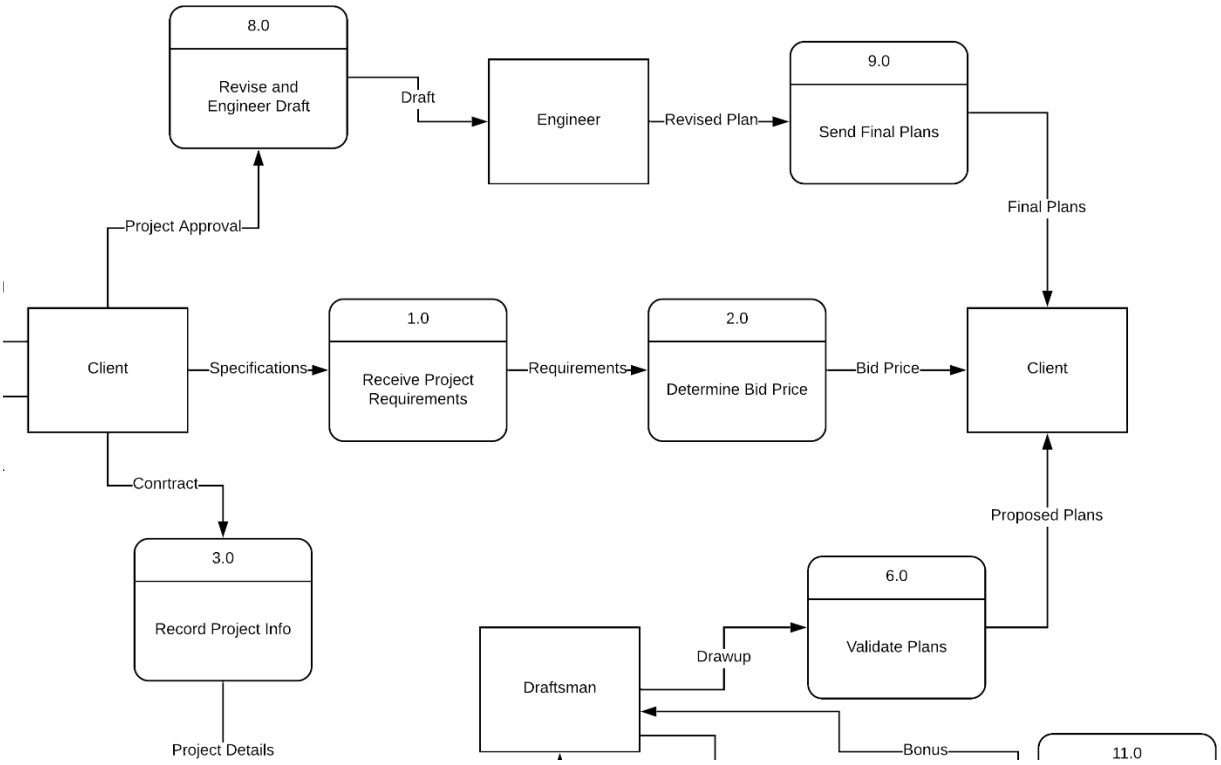
Current Level-0 Diagram

This diagram is broken into several pieces for readability. At the end of this document, we will show an image of the entire diagram to help piece it all together.

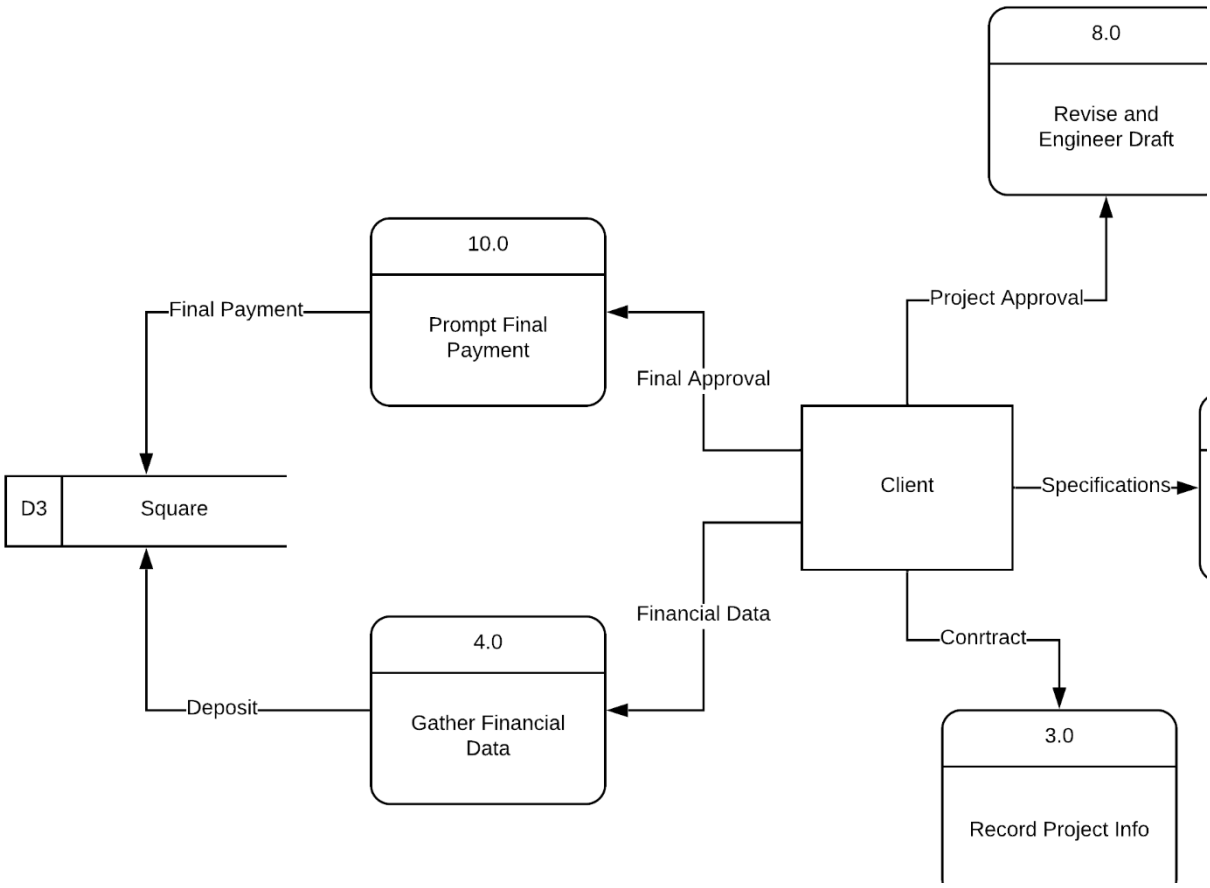
Systems 1.0, 2.0, 3.0, 5.0, 6.0, 7.0, 11.0



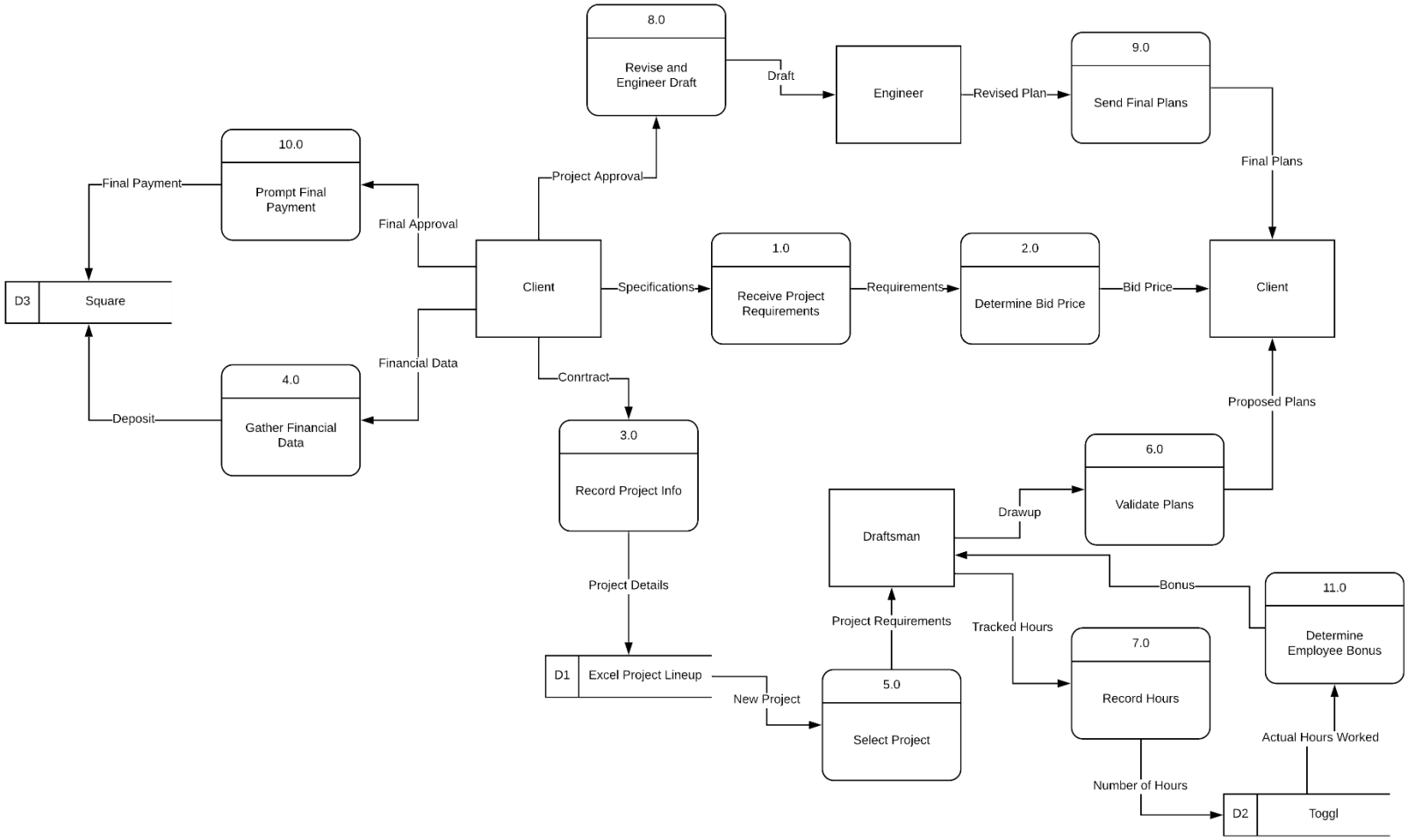
Systems 1.0, 2.0, 3.0, 6.0, 8.0, 9.0



Systems 3.0, 4.0, 8.0, 10.0



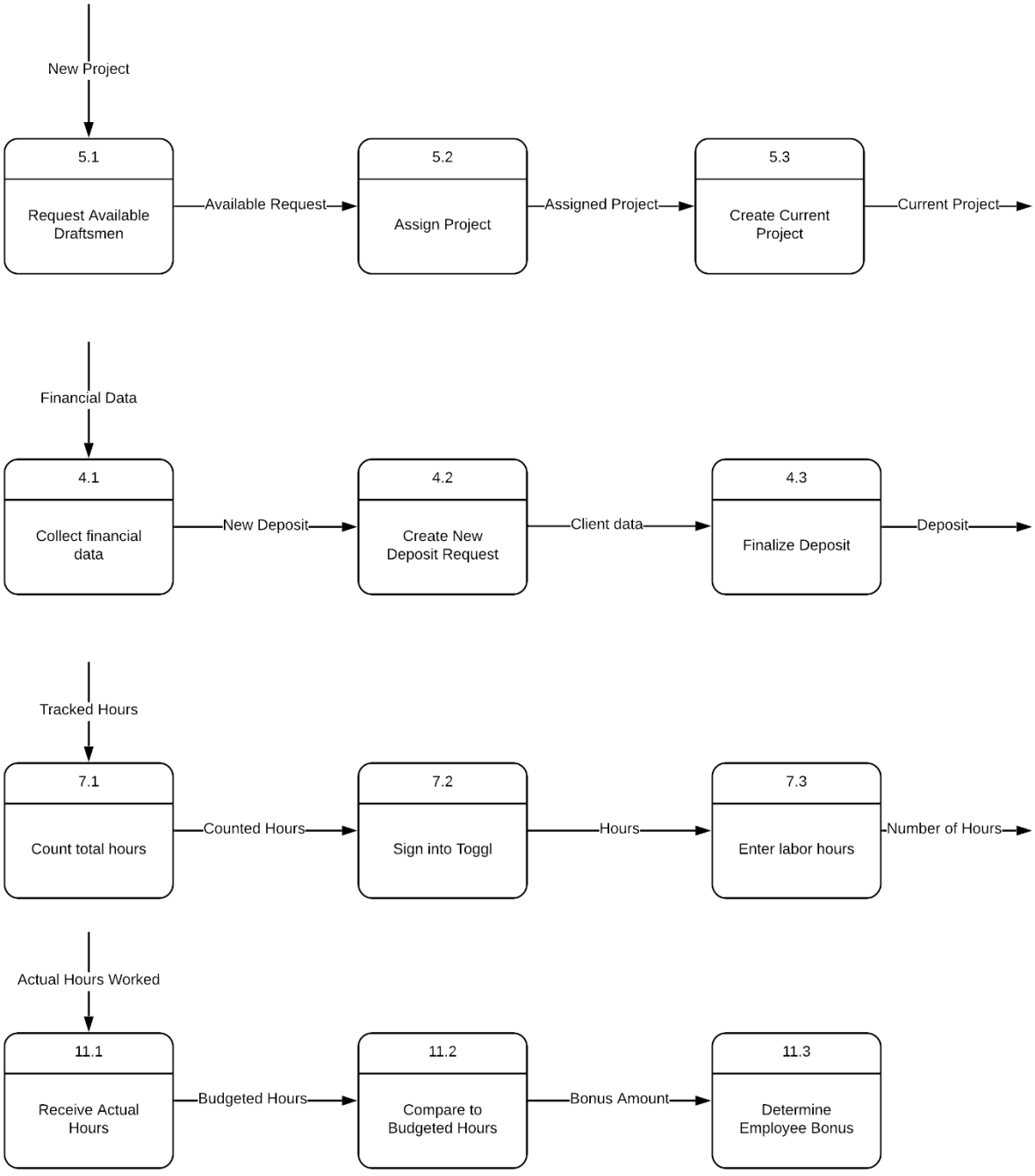
Full Diagram


8.0
Revise and Engineer Draft
Draft
Engineer
Revised Plan
9.0
Send Final Plans
Final Plans
10.0
Prompt Final Payment
Final Payment
Project Approval
Final Approval
Client
Specifications
1.0
Receive Project Requirements
Requirements
2.0
Determine Bid Price
Bid Price
Client
D3
Square
Financial Data
4.0
Gather Financial Data
Deposit
Conrtract
Proposed Plans
3.0
Record Project Info
6.0
Validate Plans
Drawup
Draftsman
Bonus
11.0
Determine Employee Bonus
Project Details
Project Requirements
Tracked Hours
7.0
Record Hours
D1
Excel Project Lineup
New Project
5.0
Select Project
Actual Hours Worked
Number of Hours
D2
Toggl

Level-0 Diagram Description

The client sends project specifications, which are received, and a bid price is calculated based on these initial requirements. The bid price is sent to the client, and once the client accepts the bid, the client sends financial data which is collected and a deposit is input into Square. Before the financial data is sent, the client also signs and sends a contract, which is used to record project information. Then, the project details are input into the Excel Project Lineup. A new project can then be selected and the project requirements are sent to a draftsman. The draftsman creates a draft and also records the number of labor hours into Toggl. The draft is then sent to be validated by the client. If the client validates the draft, the draft is sent to the engineer for structural and engineering-specific modifications. These revised plans are sent to the client for final approval. Once the client give final approval, the system prompts the client for the final payment, which is input via Square.

Current Level-1 Diagrams



Appendix B: Proposed Forms and Reports

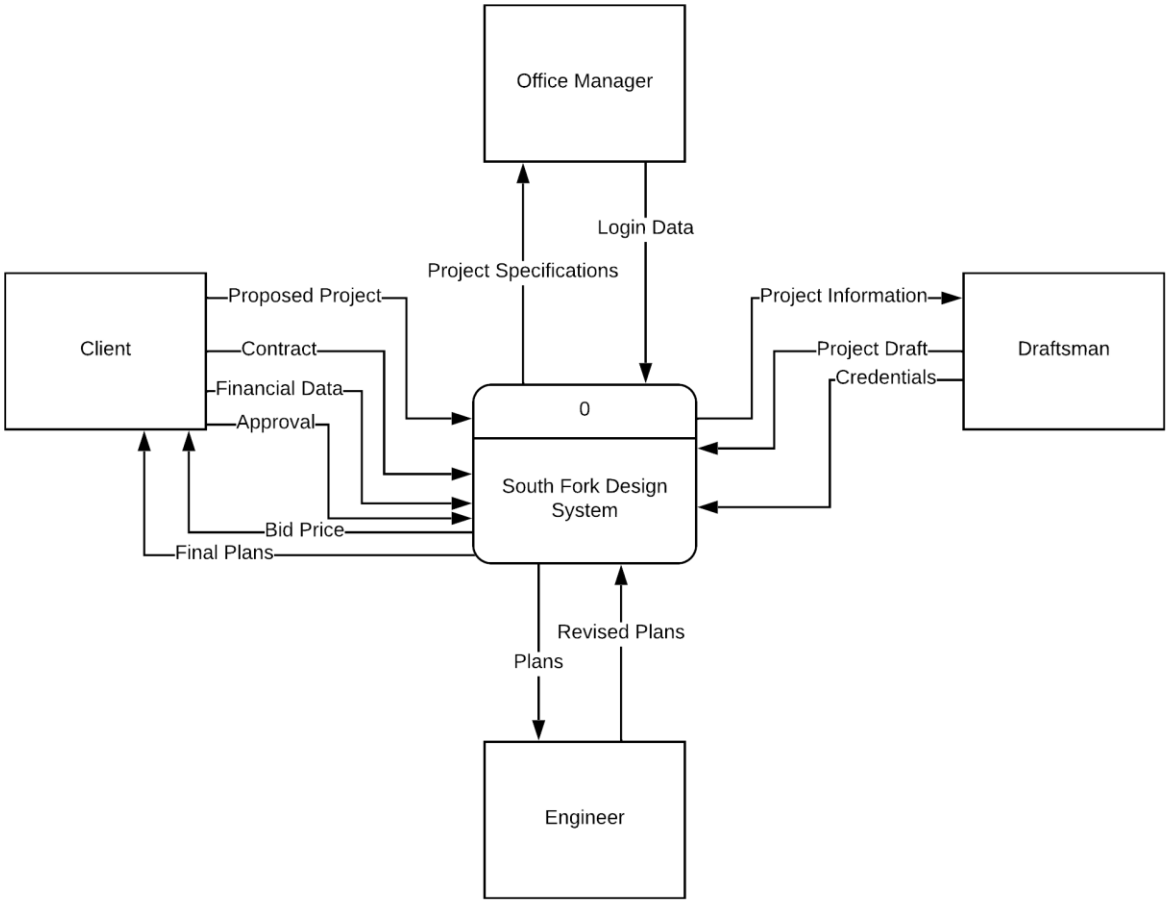
The proposed forms and reports include

- Data Flow Diagrams
- ERD
- Class Diagrams
- Sequence Diagrams

Data Flow Diagrams

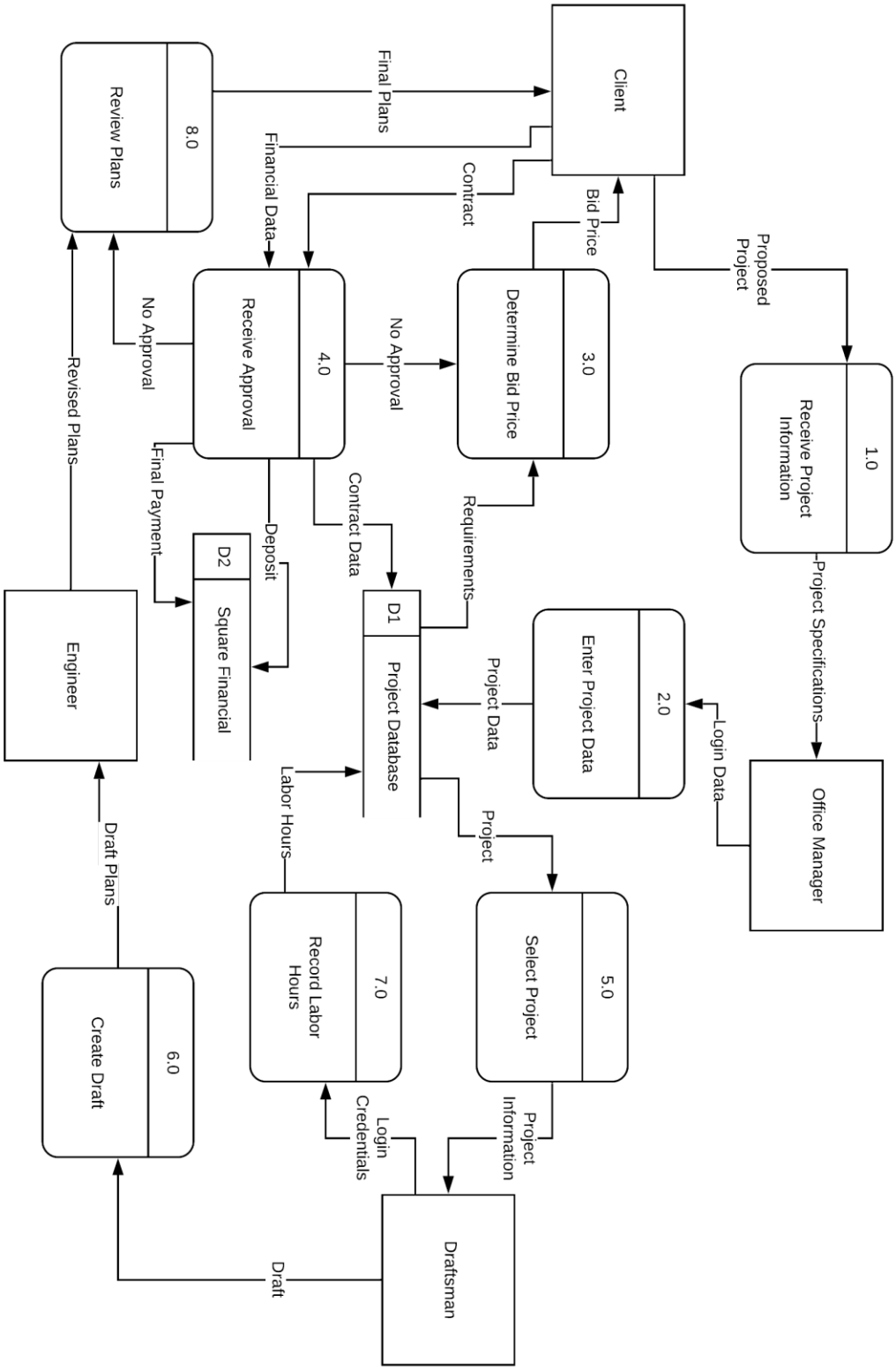
This section includes the context level and level-0 data flow diagrams for the new system.

Proposed Context Level



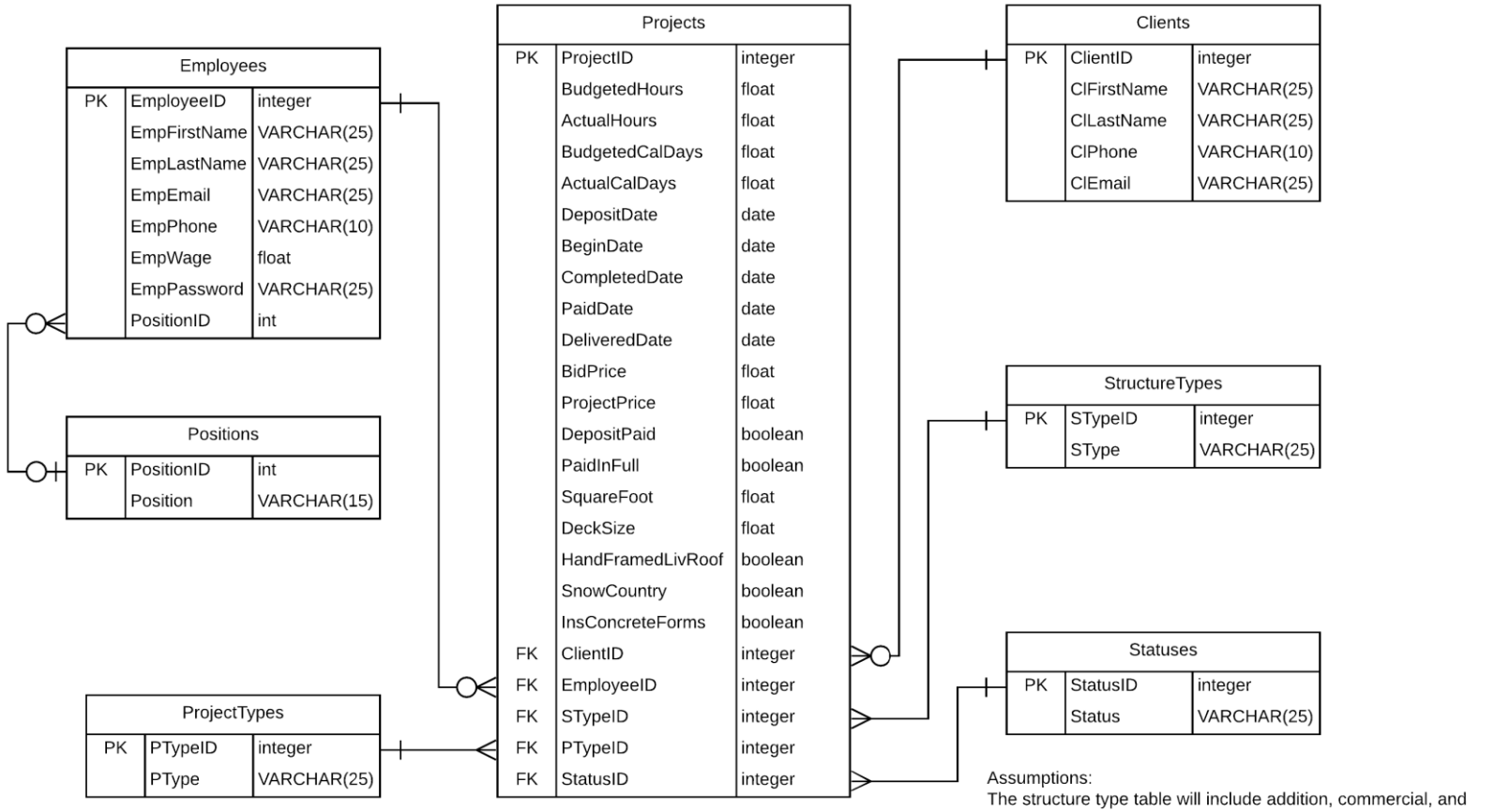
The description of this diagram can be found in the proposed system section.

Proposed Level-0 Diagram



The description of this diagram can be found in the proposed system section.

Proposed ERD



Assumptions:
The structure type table will include addition, commercial, and residential fields
The project type table refers to if the job was just an engineering project, design project, or full project
The status will include 1 for unassigned, not started, 2 for assigned, not started, 3 for in progress, 4 for completed, and 5 for delivered
The project price is the final price that the client pays. The bid price will be a calculated field, and thus not a field in the database

ERD Description

As explained in the proposed system section, the ERD contains seven tables to help keep track of projects, employees, and clients. The StructureTypes, Statuses, Positions, and ProjectTypes tables are all lookup tables to help with the project tracking. The assumptions in the ERD are also important to note:

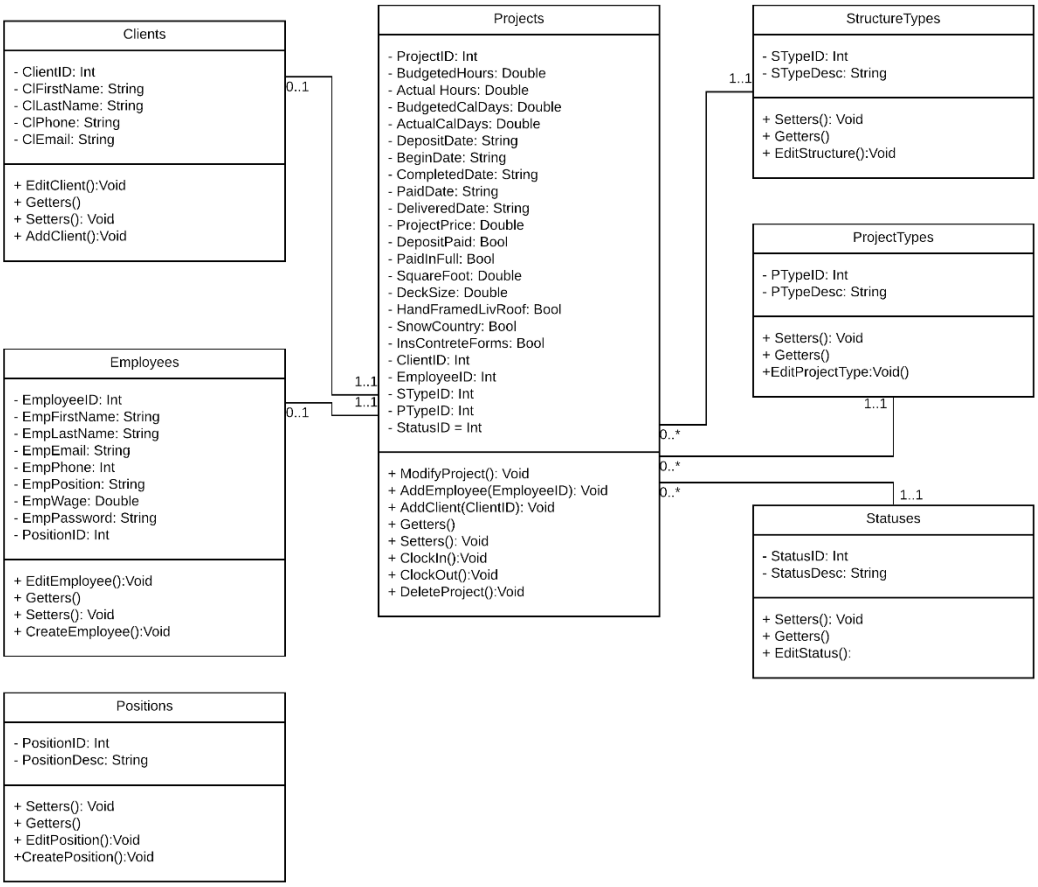
Assumptions:

The structure type table will include addition, commercial, and residential fields The project type table refers to if the job was just an engineering project, design project, or full project

The status will include 1 for unassigned, not started, 2 for assigned, not started, 3 for in progress, 4 for completed, and 5 for delivered

The project price is the final price that the client pays. The bid price will be a calculated field, and thus not a field in the database

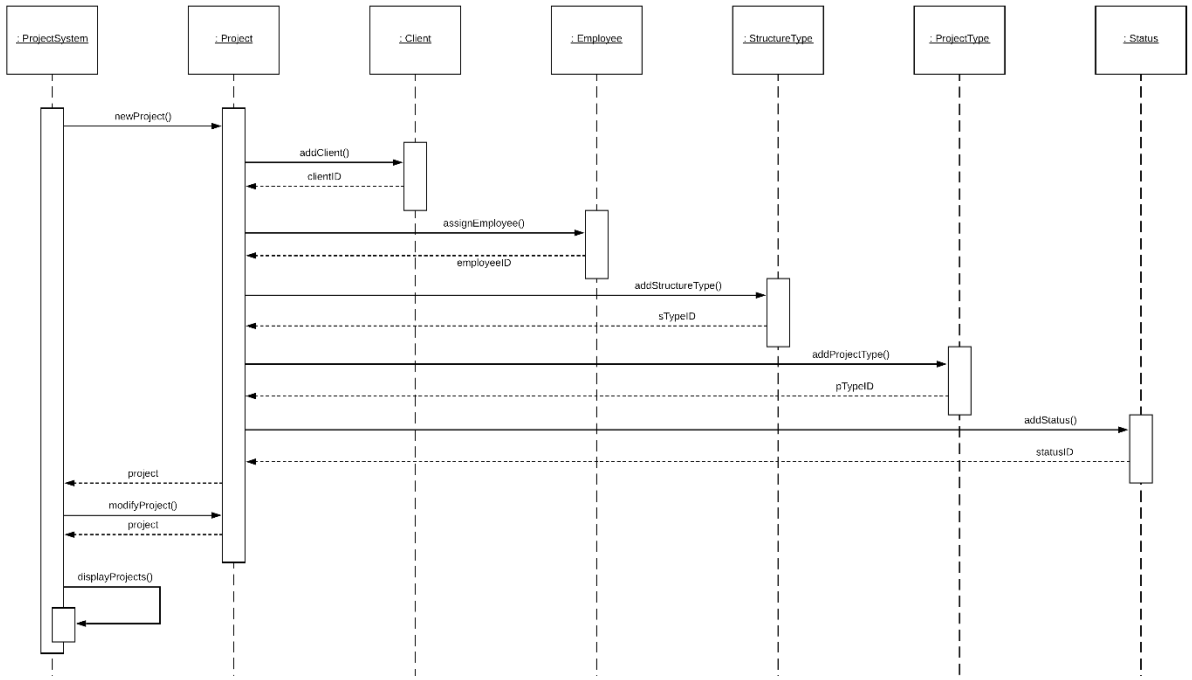
Proposed Class Diagram



Class Diagram Description

The class diagram follows a similar format to the ERD. To help promote security, we made all attributes within the classes private scope. These attributes are only accessible through the getter and setters available in the methods.

Proposed Sequence Diagram

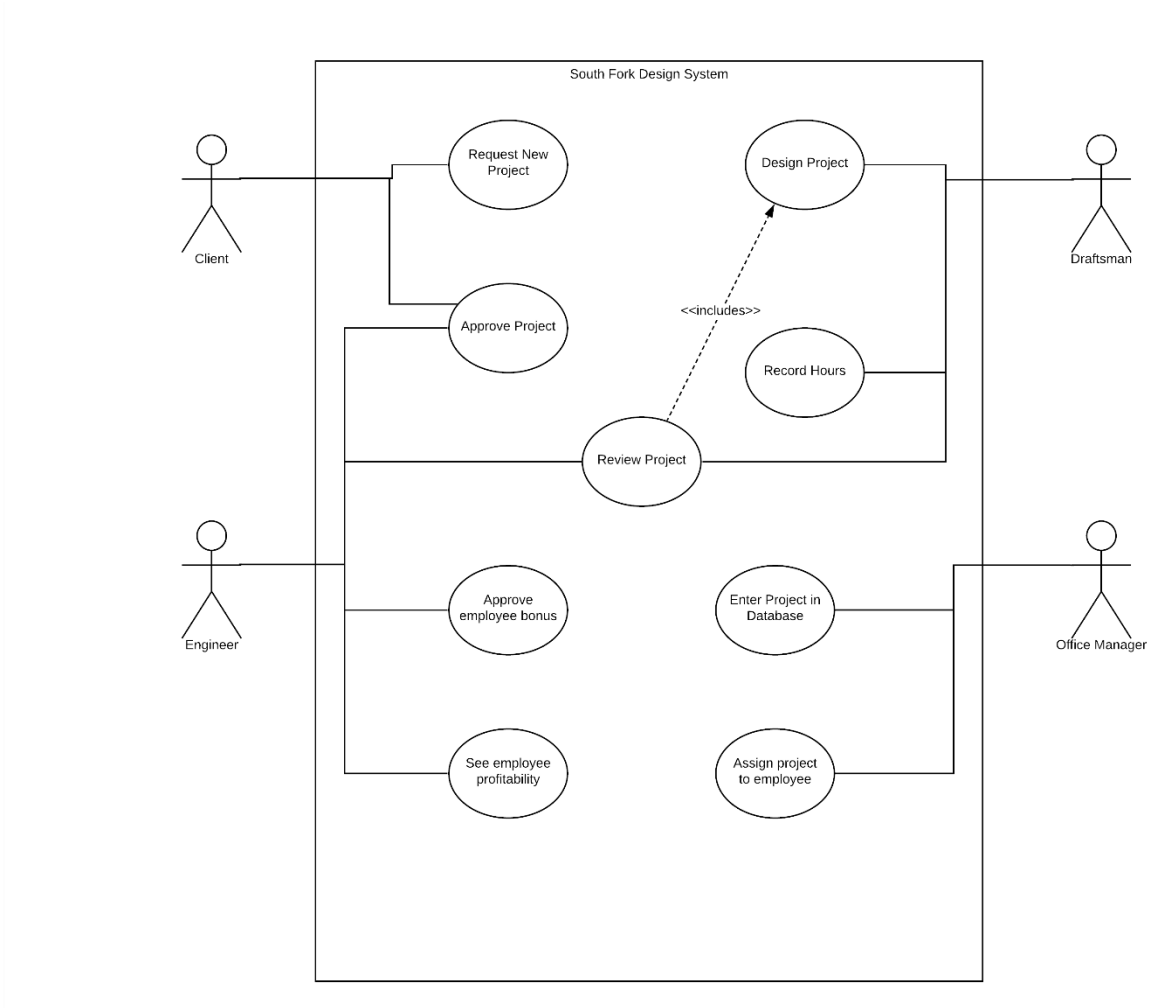


Sequence Diagram Description

As part of this diagram, we used the project system as the main lifeline of the project. Within that system, we have various different classes working to help build projects, including Project, Client, Employees, StructureType, ProjectType, and Status. As each class is built, the autoincremented primary key is returned to the system to update the data.

There is also capability to modify the projects and display them through this system.

Use Case Diagram



Use Case Diagram Description

This use case diagram holds four different roles:

- Client
- Engineer
- Draftsman
- Office Manager

The table below shows the different use cases for each role.

Role	Use cases
Client	Request new project Approve project
Engineer	Approve project Review project Approve employee hours See employee profitability
Draftsman	Design project Record hours Review project
Office manager	Enter project in database Assign project to employee